ABSTRACT

Title of Dissertation: PLANNING AND SCHEDULING INTERRELATED ROAD NETWORK PROJECTS BY INTEGRATING CELL TRANSMISSION MODEL AND GENETIC ALGORITHM

Elham Shayanfar, Doctor of Philosophy, 2018

Dissertation directed by: Professor Paul Schonfeld

Department of Civil and Environmental Engineering

In systems with interrelated alternatives, the benefits or costs of each alternative depend on which other alternatives are selected and when they are implemented. System interrelations and uncertainties in various elements of transportation systems such as future demand, make it difficult to evaluate project impacts with analytical methods. This study proposes a general and modular framework for planning and scheduling interrelated infrastructure projects under uncertainties. The method should be general enough to address the planning problem for any

interrelated system in a wide range of applications. The goal is to determine which projects should be selected and when they should be implemented to minimize the present value of total system cost, subject to a cumulative budget flow constraint. For this purpose, the scheduling problem is formulated as a non-linear integer optimization problem that minimizes the present value of system cost over a planning horizon. The first part of this dissertation employs a simple traffic assignment model to evaluate improvement alternatives. The algorithm identifies potential locations within a network that needs improvements and considers multiple improvement alternatives at each location. Accordingly, a probabilistic procedure is introduced to select the optimal improvement type for the candidate locations. The traffic assignment model is used to evaluate the objective function and implicitly compute project interrelations, with a Genetic Algorithm (GA) developed to solve the optimization problem. In the second part of the dissertation, the traffic assignment model is replaced with a more detailed evaluation model, namely a Cell Transmission Model (CTM). The use of CTM significantly improves the model by tracking queues and predicating queue build-up and dissipation, as well as backward propagation of congestion waves. Finally, since GA does not guarantee global optimum, a statistical test is employed to test the optimality of the GA solution by estimating the probability of arriving at a better solution. In effect, it is shown that the probability of finding a better solution is negligible, thus demonstrating the soundness of the GA solution.

PLANNING AND SCHEDULING INTERRELATED ROAD NETWORK PROJECTS BY INTEGRATING CELL TRANSMISSION MODEL AND GENETIC ALGORITHM

by

Elham Shayanfar

Dissertation submitted to the Faculty of the Graduate School of the

University of Maryland, College Park, in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

2018

Advisory Committee:

Professor Paul Schonfeld, Chair/Advisor

Professor Lei Zhang

Associate Professor Qingbin Cui

Associate Professor Cinzia Cirillo

Professor Robert Windle

Dedication

To my beloved parents

for their endless love and support.

Acknowledgment

I would like to express my most sincere gratitude to my advisor, Dr. Paul Schonfeld, for his constructive guidance, and continuous support throughout the years. His passion for research, patience, commitment, kindness, and sheer devotion to his students are exemplary and will inspire me for the rest of my life.

My appreciation is extended to the committee members: Dr. Lei Zhang, Dr. Qingbin Cui, Dr. Cinzia Cirillo, and Dr. Robert Windle for their invaluable advisement and feedback on various aspects of this project.

I am also grateful to the Appalachian Regional Commission (ARC), especially Mr. Jason Wang for his support and interest in my work. Jason was not only an incredible supervisor, but the care and attention that he has bestowed upon me has been very precious to me.

I am thankful to Arta Moeini for his companionship and for the joyful moments that he has brought me in this taxing period. I would also like to thank my dear friends Elham, Arezoo, Bahar, and Alireza with whom I shared many wonderful memories throughout these years.

Most of all, I would like to thank my family: my father, who has always been my role model and always encouraged me to achieve my goals; my mother for her unconditional love and support; my sister who is my best friend and my brother who makes my life cheerful. Without their love and support, this work would not have been possible.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background

The problem of selecting transportation projects under budget constraints is a resource allocation problem which has been studied for decades. In early studies, the project selection problem was formulated as a simple linear and binary optimization problem (Lorie and Savage, 1955). In this case, there is some benefit and cost associated with each candidate project, and the objective function is formulated as a linear summation of benefits subject to the expenditure of projects bounded by a budget. This problem is well known as the knapsack problem, which is proved to be NP-hard (Crowder et al., 1983) and can be solved via dynamic programming or branch-and-bound methods (Martello and Toth, 1990). Although this formulation can be effectively solved by mathematical modeling and produce the optimal selection, it assumes that projects are completely "independent", which lacks any timing component, presuming that projects are implemented at about the same time.

In the real world, especially in transportation networks, the benefits and costs of projects are quite "interrelated". In other words, the benefits and costs of each individual project depend on whether and when some other projects are implemented. This is the case for most transportation networks since changes in network components shift the locations of bottlenecks and redistribute flows. Therefore, the total benefits from multiple projects are not a linear summation of the impacts from individual projects. Nemhauser and Ullman (1969) conducted one of the early studies dealing with project interdependencies. They proposed the following quadratic objective function:

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^{n} b_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij} x_i x_j \quad (1)$$

In this formulation d_{ij} represents the interaction between projects *i* and *j* having a positive value when projects are complementary and a negative value when they are competing. This is a binary, non-linear and non-separable knapsack problem which incorporates project interrelation. Variations of this method can be found in some recent works, such as Dickinson et al. (2001), Durango-Cohen and Sarutipand (2007), and Bhattacharyya et al. (2011).

Compared to the linear objective function, the quadratic objective function shown above enhances the flexibility of the project selection problem by incorporating project interrelations. However, the quadratic formulation has its own shortfalls. First, the pairwise dependencies (d_{ij}) do not fully represent the complex interrelations and miss some relations among alternatives. This is because the actual interrelations may extend beyond just the two-way interactions between project *i* and *j*. In fact, the interaction may go to third, fourth and even higher degrees. Second, the interrelations may be difficult to quantify even for pairwise interactions (i.e. estimate d_{ij} parameter for all pairs of projects) and the number of interactions requiring estimation explodes if we go beyond pairwise relations. Third, such methods ignore the timing aspect of project implementation and do not optimize the schedule of projects. The benefits associated with particular projects may be highly related to the times when they are implemented. Therefore, evaluating projects without considering their timing may yield misleading results.

In general, methods for analyzing mutually exclusive and independent alternatives are adequately addressed in the literature. However, no such general methods are found for analyzing interrelated alternatives. Even the methods that have been developed for analyzing interrelated alternatives in some specific applications have been incapable of dealing with enough interrelations and realistic problem features.

1.2 Problem Statement

Evaluating transportation infrastructure projects and determining which of them at what time should be implemented requires several criteria. Common evaluation practices use the linear summation of project impacts in the objective function which is later optimized. Nevertheless, these assumptions are inadequate since they disregard the interdependence due to non-linearly additive benefits, costs, budget constraints, constructability or operability requirements, and other possible factors. The selection and scheduling of projects with consideration of their interrelations is a challenging optimization problem, but its solution is very valuable as it has applications in various fields, including economics, finance, operations research, development, industrial engineering, and business administration. This research deals with constructing new roads and road expansion projects as an example of interrelated projects, however, the introduced methods may be used generally to analyze interrelated alternatives.

As traffic increases and links become congested, passenger and freight movements experience increasing travel times and delays. One obvious solution to this problem is constructing new lanes and creating additional capacity on the highly congested links. Then we must determine which links should be selected, in what order they should be implemented and when they should be funded to minimize the present worth of cost. One simple idea is to identify congested links and prioritize them according to their congestion level, i.e., volume/capacity ratio. However, even after adjusting for the relative costs of links, this approach does not yield the best solution as it disregards the interrelations among network links. In fact, changes in one link affect the flows on others and removing bottlenecks from some links may shift them elsewhere in the network. Thus, in sequencing a set of improvement projects we should consider their interrelations.

Conceptually, the first step of a project planning problem is the project evaluation which identifies candidate projects and evaluates their merits, often in terms of their benefits and costs. A second step selects which projects from among the considered set should be chosen for implementation. After evaluating and selecting a set of projects for improvement, a third step determines the order of projects and, finally, a fourth step determines the scheduled time for completion under budget limitations (Wang and Schonfeld 2005). Project selection and scheduling easily becomes a large optimization problem whose feasible region increases rapidly as the number of considered projects in the system grows. Considering a set of improvement projects for a given network, the objective is to find a project implementation sequence that minimizes the total system cost or maximizes the net benefits over the analyzed period. To date, several methods have been developed for scheduling interrelated projects. However, the number of studies on this topic is relatively low.

Another issue that complicates the project selection and scheduling is uncertainty. The presence of uncertainty in transportation systems causes new challenges in optimizing network investment decisions. Improving a transportation network requires a significant investment, and such investments are usually irreversible. Therefore, it is important to effectively plan and prioritize investments in a way that addresses present as well as uncertain future needs.

1.3 Research objective

The objective of this study is to is to develop a general framework for selecting, scheduling and sequencing of interrelated alternatives, and to demonstrate how a relatively simple method, namely a traffic assignment algorithm, can be efficiently used to evaluate the objective function of an investment planning optimization problem for a road network and thereby implicitly compute the relevant interrelations among all projects that are implemented at various times. Although traffic assignment is a simple and quick way to test the algorithm, it may be replaced with a mesoscopic or microscopic simulation model if more precise evaluations are sought. As such, this study examines a Cell Transmission Model which is capable of tracking queues in the network, predicting queue spillbacks and dissipation in a reasonable way which makes CTM suitable for modeling intersections. Although road and intersection expansion projects are the focus of this study, the proposed methodology should be applicable to general cases involving more complex systems. In fact, GAs can be effectively combined with an appropriate evaluation tool (e.g. microscopic simulation, simulation approximates, queuing or neural networks) specific to the problem, to solve the planning and scheduling problem for a variety of interrelated alternatives.

1.4 Research Approach

This study aims to propose a general technical approach for planning and scheduling interrelated infrastructure projects under uncertainties. The key to generality of the approach is to incorporate modularity into the method by connecting the application-specific "evaluation component" with the "optimization component". In this case it is important to separate the evaluation from the decision optimization component and connect them through data exchange rather than structural integration.

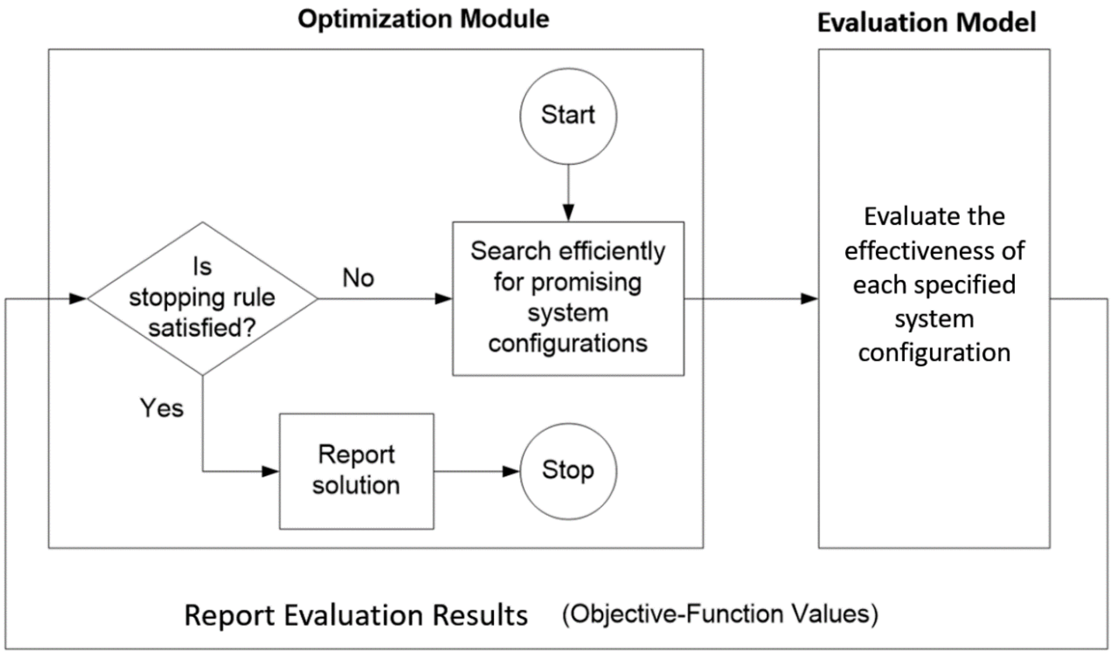


Figure 1.4-1 Optimization Model Separated from Evaluation Model

Evaluation Module: In general, the evaluation component of the planning and scheduling process is application-specific. Depending on the type of project, different evaluation models can be used to evaluate effects of any combination of projects, which accounts for the interrelations between projects. These evaluation models include: queuing models (Dai and Schonfeld 1998), queuing networks (Zhu et al. 1999), artificial neural networks (Wei and Schonfeld 1994), traffic assignment models (Tao and Schonfeld 2005, Tao and Schonfeld 2006, Tao and Schonfeld 2007, Shayanfar et al. 2016), mesoscopic models such as Cell Transmission models and hybrid combinations of analytic queuing and microsimulation models (Yang et al. 2009).

Optimization Module: On the other hand, the methods used for the decision optimization can be applied to a verity of interrelated systems regardless of their specific characteristics. These models can accept inputs from a wide range of application-specific models for evaluating the interrelated alternatives. Some of the methods explored in the literature include: Swapping Algorithms (Martinelli 1993), Branch and Bound (Wei and Schonfeld 1994), Simultaneous Perturbation Stochastic Approximation (Ting and Schonfeld 1998), Lagrangian Relaxation (Tao and Schonfeld 2005), Genetic Algorithms (Jong and Schonfeld 2001, Wang and Schonfeld 2008), Island Models (Tao and Schonfeld 2006, Tao and Schonfeld 2007), Simulated Annealing (Yang et al. 2009, Shayanfar et al. 2016) and Tabu Search (Shayanfar et al. 2016).

It should be noted that the methods used for evaluating systems with interrelated projects could be largely separated (i.e., be designed to be modularly independent) from the methods for optimizing the selection, sequencing and scheduling of such projects. In other words, various system evaluation methods could be mixed and matched with various optimization methods.

1.5 Dissertation Organization

To achieve the above research purposes, this dissertation will consist of the following eight chapters. The focus of each chapter is detailed below.

Chapter 1, "Introduction," presents background information, problem statement, research objective and the research approach used in this research.

Chapter 2, "Literature Review," focuses on reviewing research completed in recent years that is relevant to the topic of this dissertation. Particularly, recent literature on project selecting and scheduling techniques, evaluation methods for project interrelations, and optimization models are comprehensively review. Based on the review results, a summary and the potential contributions of this research work are addressed.

Chapter 3, “Methodology,” describes the general framework proposed by this research. The problem formulation including the objective function and constraints are described in this chapter. Also, the “Evaluation” and “Optimization” models developed in this study are presented in detail. Note that two separate Evaluation models including Traffic Assignment and Cell Transmission model were developed and tested in this study. The application results of each model are presented separately in two chapters.

Chapter 4, “Case Study Networks,” provides detailed information including maps, demand table, network characteristics, and interrelation example for two case studies implemented in the study. Specifically, the Sioux Falls and Anaheim network are described in this chapter.

In Chapter 5, “Traffic Assignment Application and Results,” a traffic assignment model is used to evaluate improvement alternatives. The algorithm identifies potential locations within a network that needs improvements and considers multiple improvement alternatives at each location. A probabilistic procedure is used to select the optimal improvement type for the candidate locations. This approach is applied in two case study networks for which the improvement projects include adding new links and adding lanes to existing links. At the end a statistical test is presented to test the goodness of the optimized result.

Chapter 6, “Cell Transmission Model,” presents a background on Cell Transmission mode, and describes the formulation, and network representation of this model.

Chapter 7, “CTM Application and Results,” replaces the Traffic Assignment in Chapter 5 with a CTM and applied the modified model to a similar case study in Chapter 5. The use of CTM significantly improves the model by capturing many important traffic phenomena such as queue build-up and dissipation, and backward propagation of congestion waves which allows to model intersections and consider intersection improvements.

The conclusions, and recommendations for application and future studies will be summarized in Chapter 8.

Chapter 2 Literature Review

2.1 Project Selecting and Scheduling

Decisions about planning and scheduling of projects are two major aspects of any business and engineering applications. In practice, decision makers have to allocate limited resources to multiple candidate projects (which is called resource assignment, or project planning/selection) and decide the starting/completing times for each selected project over a time horizon (called scheduling), in order to optimize some performance metric (e.g., maximize Net Present Value). Resource assignment and project scheduling jointly are challenging from a computational perspective (Lombardi and Milano 2012). This problem lies within the class of resource-constrained project scheduling problem, which is a well-known problem in operations research (Hartmann and Briskorn, 2010).

Consider the binary Knapsack Problem as an example of project selection. The objective is to maximize the total value of items in the knapsack

$$\max_{x_i \in \{0,1\}} \sum_i v_i x_i \quad (2)$$

subject to a capacity constraint

$$\sum_i w_i x_i \le W \,, \quad (3)$$

where v_i and w_i are value and cost of item i, respectively, and W is the maximum budget. Implicitly, this formulation assumes that projects can be evaluated independently, i.e., the objective is additive. The variant of the knapsack problem with one type of resource can be solved optimally by ranking items according to their value-cost ratio, v_i / w_i, and selecting them sequentially until the budget constraint is violated. This is similar to the practice of prioritizing candidate projects according to their benefit-cost ratios (BCR), which is commonly applied in practice. However, such a strategy fails when a project's benefit is affected by the acceptance or rejection of another project, i.e., when they are interdependent (Martinelli 1993).

In general, project prioritization is an important problem in transportation policy as projects require significant investments which are usually irreversible. Therefore, many studies in the literature address the problem of project prioritization under uncertainty. Sadeghi and Moghaddam (2016) propose a Data Envelopment Analysis (DEA) method to prioritize safety retrofit projects, in which uncertainty is considered in benefit and cost estimations. Xu et al. (2017) propose two types of indices, namely (i) spatial connectivity and accessibility, (ii) urban land development to prioritize future funding and construction of the planned high-speed rail corridors of China. Quadros et al. (2015) apply the Analytic Hierarchy Process (AHP) to identify the most relevant criterion in prioritizing the decisions of transportation infrastructure investments in Brazil. Machado-León et al. (2017) propose a methodological approach to analyze rail services performance in Algiers and identify the aspects that should be prioritized for improvements by combining an Importance-Performance Analysis and a decision tree model.

This research is in line with such studies to address the problem of project prioritization. However, it transcends the conventional prioritization practices by considering the timing of project implementation and demand uncertainties and focusing on the treatment of interrelations among projects in a network setting.

2.2 Evaluation of Project Interrelations

2.2.1 Dependence Matrix

One of the first works in the literature that considered interdependent alternatives was that of Markowitz (1952) on portfolio management. Since then more recent studies have addressed the problem of portfolio selection among interdependent projects (Cruz et al., 2014; Rebiasz et al., 2014; Li et al., 2016). In this study, a linear program was extended into a quadratic program with the inclusion of variances of returns for different stocks. The objective was to minimize the sum of purchase cost and interrelated risks

$$\max_{x \geq 0} c^T x + \frac{1}{2} x^T Q x \qquad (4)$$

subject to minimum return requirement

$$Ax \geq b, \qquad (5)$$

where Q denotes the variance/covariance matrix and b is the minimum return.

The consideration of project interdependence significantly complicates the model's structure because the combined costs and benefits for a set of projects are no longer equal to the sum of the costs and benefits, respectively, of individual projects. The dependence matrix, such as the Q matrix in the above formulation, is convenient in modeling interdependence between choices.

This method and its variants can also be found in more recent works. Dickinson et al. (2001) developed a model to optimize a portfolio of development improvement projects for the Boeing Company. The authors used a dependence matrix to quantify the interdependencies among projects. Then a non-linear, integer program model was developed to optimize the project selection. Sandhu (2006) introduced a dependency structure matrix that captured the project logistic interdependencies. Durango-Cohen and Sarutipand (2007) formulated a quadratic programming for optimizing maintenance and repair (M&R) policies for transportation infrastructure systems. The quadratic objective of their work included the pairwise economic dependencies capturing the costs and benefits of improving adjacent facilities. Dueñas-Osorio et.al (2007) studied the interdependence response of network systems to internal or external disruptions. They established interdependencies among network elements based on geographical proximity. Their work indicated that responses that are detrimental to networks are larger when interdependencies are considered after disturbances. Bhattacharyya et al. (2011) also considered n-way interdependencies in the Research and Development (R&D) project portfolio selection problem. Tofighian and Naderi (2015) developed a mixed integer linear program to formulate the selection and scheduling of projects maximizing total expected benefits. They also proposed an ant colony algorithm to optimize the objective function. This paper defined the interdependencies among projects with a simple dependence matrix, which is insufficient in capturing the full interrelations among projects in transportation networks and various other complex systems.

Two main issues arise from using a dependence matrix. First, as Disatnik and Benninga (2007) argue, the estimation and manipulation of a dependence matrix becomes computationally difficult as the project space grows. Second, the pairwise and n-way dependencies do not completely represent the complex interrelations among alternatives. Instead of a dependence matrix, complete system models, such as queueing approximations (Jong and Schonfeld, 2001), equilibrium assignment (Tao and Schonfeld, 2005), microsimulation (Wang and Schonfeld, 2008) and neural networks (Bagloee and Tavana, 2012), are better suited for modeling interrelations.

2.2.2 Complete System Models

Instead of a dependence matrix, complete system models, such as queueing approximations, equilibrium assignment, microsimulation, and neural networks are better suited for modeling interrelations.

In some studies, the objective function was evaluated with approximations of simulation including queuing models (Dai and Schonfeld 1998), queuing networks (Zhu et al. 1999), artificial neural networks (Wei and Schonfeld 1994), traffic assignment models (Tao and Schonfeld 2005, Tao and Schonfeld 2006, Tao and Schonfeld 2007, Shayanfar et al. 2016) and hybrid combinations of analytic queuing and microsimulation models (Yang et al. 2009). By 2005 it became computationally feasible to optimize the U.S. waterway network using microsimulation directly, without approximations, for long-term planning problems (Wang and Schonfeld 2005). These models are generally applicable for modeling truly complex systems and relations among infrastructure developments. The remainder of this section overviews some recent studies applying these models.

Queuing Approximation

Jong and Schonfeld (2001) developed a genetic algorithm and a simple approximation to solve project investment planning problem. They showed that GAs are very effective at searching for minimum cost highway alignments.

Artificial Neural Network

Wei and Schonfeld (1993) developed an algorithm which combined artificial neural network and a branch and bound algorithm to find the optimal or near optimal solution for scheduling interdependent projects. They proposed a multi-period network design model for selecting the best combination of improvement projects and schedules. Then they utilized the neural network approach to estimate total travel times corresponding to different project selection and scheduling. They applied their model to Calvert County highway system in southern Maryland to check the performance of their model. Bagloee and Tavana (2012) formulated the prioritization problem as a Traveling Salesman Problem (TSP) and incorporated a Neural Network (NN) to assess project interdependence. A heuristic algorithm with hybrid components was then used to search for the longest (most beneficial) path in the NN as a solution to the TSP.

Equilibrium Assignment Model

Tao and Schonfeld (2005) developed a Lagrangian heuristic for selecting interdependent projects under cost uncertainty. They developed a genetic algorithm for solving the Lagrangian problem and applied equilibrium assignment to evaluate the objective function.

Simulation

Wang and Schonfeld (2005) developed a waterway simulation model for evaluating lock operations over long analysis periods and then solved the problem of selecting, sequencing and scheduling interdependent projects with a genetic algorithm. Martinelli and Schonfeld (1995) developed an approximation to microsimulation model to evaluate lock improvements with consideration of their interrelation.

2.3 Project Selecting and Scheduling Optimization

2.3.1 Mathematical Programming

Studies in the existing literature mainly deal with the selection and scheduling of projects by assuming independence among them. Two approaches are commonly used for selecting and sequencing of independent projects. These are integer programming (Weingartner 1966; Cochran et al. 1971; Clark et al. 1984) and dynamic programming (Weingartner 1966; Nemhauser and Ullman 1969; Morin and Esogbue 1971; Erlenkotter 1973; Morin 1974). In a more recent effort, Li et.al (2013) proposed a hypergraph knapsack model to maximize the overall benefits for a sub collection of interdependent projects. For this purpose, a multi-commodity minimum cost network (MMCN) was developed to obtain traffic volume and speed to estimate benefits using a life cycle cost analysis method. Mollanejad and Zhang (2014) attempts to prioritize road improvements by accounting for equity issues into the interurban road network design problem. They did this by minimizing the total inaccessibility in the region by solving a mixed integer program. Chen et.al (2015) reformulated the mixed network design problem (MNDP) to simultaneously find both optimal capacity expansions of existing links and new link additions. The upper level aimed to minimize the network cost in terms of the average travel time via the expansion of existing links and the addition of new candidate links. The lower level was a dynamic user-optimal condition that could be formulated as a variational inequality problem. A surrogate based optimization framework was then proposed to solve the MNDP.

The drawbacks of these approaches include the difficulty of capturing the interrelations among projects and their inefficiency or even the infeasibility for solving large problems (Jong and Schonfeld 2001). The objective function for problems such as prioritizing interrelated projects has a surface that is “noisy” (i.e. containing numerous local optima) and non-convex. Moreover, as the number of candidate projects increases, the problem’s solution may soon exceed the capabilities of conventional mathematical optimization methods. Therefore, mathematical programming such as gradient-based search, integer programming and dynamic programming are incapable of solving the interrelated project-investment planning problems.

2.3.2 Heuristics/ Meta-Heuristics

As mentioned earlier, mathematical programming such as gradient-based search, integer programming and dynamic programming are not suitable for solving the interrelated project-investment planning problems. As a result, Heuristics and meta-heuristics, especially population-based methods such as GA, have gained popularity for solving problems without analytical objective functions because they can be relatively easily and efficiently distributed among multiple processors (Balamurugan, 2006; Haq and Kennan, 2006). Also, objectives evaluated from computer simulations, which are usually analytically intractable (i.e., discontinuous and non-differentiable) (Koziel et al. 2011), can easily be embedded directly into the heuristic optimization loop. The following are some of the recent studies that employed heuristic methods to solve the selecting and scheduling problem.

Bouleimen and Lecocq (2002) developed a simulated annealing algorithm for the resource constrained project scheduling problem. The objective of this model was to minimize total project duration. A new design was substituted the conventional SA search scheme which considered the specificity of the solution space of project scheduling problems. Tao and Schonfeld (2005) developed a lagrangian heuristic to solve the selection of interdependent projects under cost uncertainty. In this paper a genetic algorithm was developed to solve the lagrangian problem, and an equilibrium assignment was applied to evaluate the objective function. Mika et.al (2005) proposed two local search meta-heuristics, simulated annealing and tabu search to solve the multi-mode resource constrained project scheduling problem with discounted cash flows. The objective was set to maximize the net present value of all cash flows. Four payment models were considered in this study: lump-sum payment at the completion of the project, payments at activity completion times, payments at equal time intervals, and progress payments. They evaluated their model on a set of instance switches that were based on some standard test problems constructed by the ProGen project generator. Tao and Schonfeld (2007) developed variation of traditional genetic algorithms called island models to optimize the selection and scheduling of interrelated projects under resource constraints. Szimba and Rothengatter (2012) extended the classical benefit-cost analysis by integrating the occurrence of interdependence among the projects within an investment package. They addressed the interdependence problem by introducing a heuristic method to solve the large-scale problem with numerous projects. In this approach, the number of projects and their interrelations are reduced step by step in order to reduce the number of interdependence cases.

2.4 Summary

The above literature includes studies which employed both analytic methods and simulation models in various applications to solve the planning and prioritizing problem. Due to complex and combinatorial nature of the interrelated alternatives, heuristic algorithm was adopted as the dominant problem-solving approach. Table 2.4-1 provides a non-exhaustive summary of recent studies of the planning and scheduling of interrelated projects.

Table 2.4-1 Summary of selected relevant studies since 2000

Study	Objective	Optimization	Evaluation	Interrelation	Application
Lee and Kim 2000	Multi-Criteria	Goal program	Analytic network process	Pairwise comparisons	Information systems
Jong and Schonfeld 2001	Total system delay	Genetic algorithm	Analytical	Queueing approximation	Inland waterways

Dickinson et al. 2001	Net present worth	Nonlinear, integer program	Analytical	Dependency Matrix	Technology portfolio management
Tao and Schonfeld 2005	Net benefit	Lagrangian Relaxation Heuristic	Simulation	Equilibrium assignment model	Highway network
Wang and Schonfeld 2005	User benefit	Genetic algorithm	Simulation	Marginal cost increments	Inland waterways
Tao and Schonfeld 2006	User benefit	Island model	Simulation	Equilibrium assignment model	Highway network
Durango and Sarutipand 2007	Total cost	Quadratic program	Analytical	Pairwise dependencies	Generic transportation network
Wang et al. 2009	Net benefits	Simulation-based optimization	Simulation	Simulation	Waterway network maintenance
Bhattachary-ya et al. 2011	Multi-objective	Genetic algorithm	Analytical	Dependency matrix	R&D projects
Bagloee and Tavana 2012	Travel time	Hybrid meta-heuristics	Analytical	Neural network	Road projects
Li et al. 2013	Total benefits	Hypergraph knapsack	Simulation	Multicommo-dity minimum cost network	Tollway investments

Vilkkumaa et.al. 2014	Value of portfolio	Bayesian estimation	Analytical	Bayes estimates	Generic projects

After reviewing the literature, the main research gap in this area can be summarized as the following:

1) Although there are a number of application specific studies to deal with project interrelations, the literature lacks general methods for solving the selecting and scheduling of interrelated alternatives in a wide range of applications.
2) Estimating and using the marginal pairwise or n-way interrelations is rarely adequate. More complete system models, especially microscopic and mesoscopic models, are desirable for evaluating systems with project interrelations.
3) While uncertainties have been dealt mostly through sensitivity analyses, further efforts are justified to incorporate more sophisticated approaches for optimizing systems under different kinds of uncertainties.

2.5 Expected Contributions

Accounting the above observations, this research aims to contribute to the literature as the following:

1) This study proposes a general and modular framework for planning and scheduling interrelated infrastructure projects under uncertainties. The method should be general enough to address special problem characteristics and emerging issues in a wide range of applications, such as transportation networks, air transport systems, inland waterways, and areas beyond transportation.
2) Along with a general framework, we apply the proposed framework on two special case of road network projects and propose application-specific enhancements to this problem. In this sense, first a traffic assignment model is combined effectively with a GA for selecting and scheduling projects while capturing more interactions among projects (i.e. beyond previously considered pairwise interactions) and captures realistic features such as uncertainties in transportation systems.
3) In addition to the traffic assignment model which is a relatively simple method good for algorithm testing purposes, this study aims to use a more complex evaluation model to evaluate the objective function. For this purpose, I propose to use a Cell Transmission Model (CTM) which can be incorporated with the GA.
4) This study accounts for uncertainties in some important parameters such as: demand, project costs, and budget flow. For this purpose, a deterministic objective function is introduced and then is transformed into a stochastic one that combines different plausible scenarios.
5) the model is further developed to account for vehicle operation and safety costs. For this purpose, appropriate models are incorporated and added to the objective function to estimate the cost of fuel, tire, maintenance and repair along with the cost of crashes in the system.

6) In this study the budget constraint is formulated in a way to include possible internal funding from fuel taxes. In this case, throughout the analysis period, fuel taxes collected from users are added to an external budget in determining the overall investment budget. Additionally, this study proposes to add other constraints beyond budget including: construction time, precedence relations, and land availability.

7) Many realistic characteristics, beyond previous studies are considered in this research including:

    a) A multi period analysis (including peak and off-peak periods) to account for daily demand fluctuations.

    b) Adding new links as well as expanding the capacities of existing links.

    c) Multiple alternatives per location and the ability to select the best improvement alternative at each location

    d) Exponential growth of demand over time during the planning horizon.

Chapter 3 Methodology

3.1 General Framework

As stated earlier, the aim of this study is to propose a general technical approach for planning and scheduling interrelated infrastructure projects under uncertainties. To ensure the generality of the approach, we need to incorporate modularity into the method by connecting the application-specific evaluation component with the optimization component. In this sense, it is important to separate the evaluation from the decision optimization component and connect them through data exchange rather than structural integration. Figure 3.1-1 displays how the optimization module, in this case a GA can be combined with an application specific evaluation model. Note that in this framework the evaluation model feeds into the optimization loop and can be replaced with any suitable evaluation model for any type of projects.

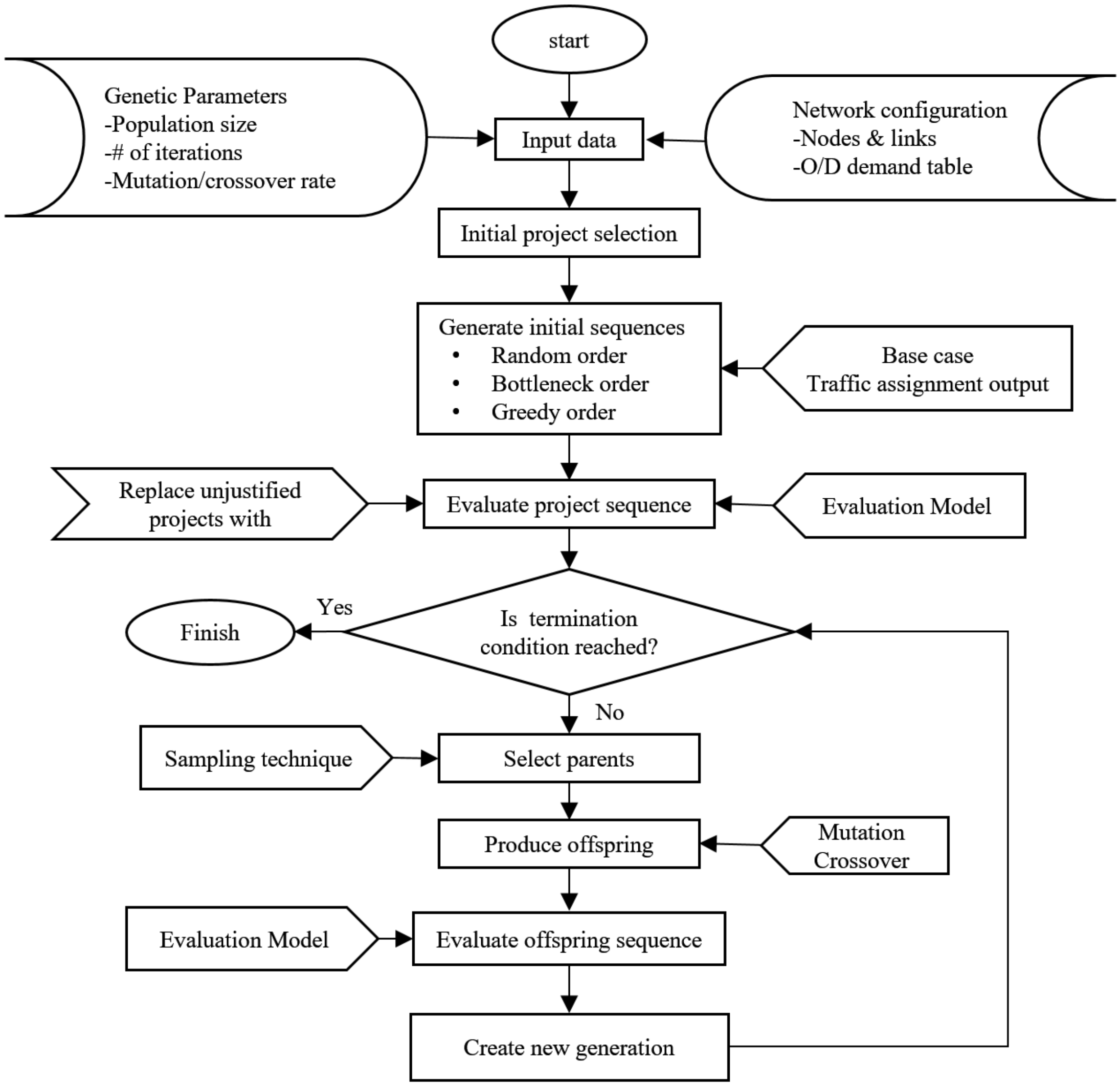


Figure 3.1-1 Framework of Optimization Process

3.2 Problem Formulation

The analysis in this paper focuses on (i) cost of travel time, (ii) vehicle operating cost, (iii) safety cost, and (iv) supplier cost. Therefore, the objective function is formulated to minimize the present value of total cost over planning horizon *T* bounded by some constraints. First, a deterministic and a stochastic objective function are introduced followed by the problem constraints including: budget flow, construction time, precedence relations, and land availability constraints. Next, detailed formulations are provided on how to estimate the cost components of the objective function including vehicle operating and safety costs.

3.2.1 Deterministic Objective Function

The mathematical formulation of the selection and scheduling problem can be quite complicated. One way to simplify the problem is to define decision variables as the completion time of each project. Let t_i be the time at which project *i* is completed, and *T* be the planning horizon (e.g. 30 years). Then, the set of t_is will determine the resulting project selection, sequence and schedule (Jong and Schonfeld, 2001). Let $x_i(t)$ be a binary variable that shows if project *i* is finished by time t:

$$\begin{cases} x_i(t) = 0 \quad if\ t < t_i \\ x_i(t) = 1 \quad if\ \ t > t_i \end{cases} \tag{6}$$

The problem is then formulated as:

$$min\, Z = \sum_{j=1}^{T} \left\{ \frac{1}{(1+r)^j} \left(\sum_{i=1}^{n_l} w_{ij} * v_t + \sum_{i=1}^{n_l} \{C_{vop(ij)} * VMT_{ij}\} + N_{cr(j)} * C_{Cr} \right) \right\} + \sum_{i=1}^{n_p} \frac{c_i x_i(t)}{(1+r)^t} \quad (7)$$

Where:

w_{ij} = travel time over link *i* in year *j*

v_t = value of time ($/hr)

n_l = total number of links

VMT_{ij}= vehicle miles traveled over link *i* in year *j*

$C_{vop(ij)}$= vehicle operating cost over link *i* in year *j* ($/veh.mi)

c_i = cost of project *i*

$N_{cr(j)}$= predicted number of crashes in year *j*

C_{cr}= crash cost for one predicted crash

n_p= number of projects

r= interest rate

The above formulation minimizes the sum of total user and supplier cost subject to a budget flow constraint, over a specified planning horizon. Note that project interrelations are not explicitly included in the objective function. As mentioned previously, the proposed method considers not only pairwise or slightly higher degrees of interrelation among alternatives, but all possible interactions among all alternatives throughout an entire network. The complete interrelations are captured by applying a full network model after each project implementation, which cannot be explicitly expressed in the objective function. In this setting, the user cost consists of travel time, vehicle operating, and safety costs while the supplier cost is the present value of implementation costs for all projects. This formulation improves the original form proposed by Jong and Schonfeld (2001) by including vehicle operating and safety costs as well as the project costs in the objective function. It is necessary to include project costs in the objective function since not all selected projects are guaranteed to fit in the budget and be implemented within the analysis period. In fact, some projects may be discarded from the sequence as they may become unjustified during the analysis. Another improvement to previous studies is to account for the possibility that candidate projects may become economically justified or unjustified after the implementation of other projects. This means that the set of candidate projects is not fixed during the analysis and is constantly updated. It is important to consider this possibility because project interrelations and effects of completing earlier projects affect the cost savings of future projects.

3.2.2 Stochastic Formulation

In long-term planning, decision makers are often confronted with the problem of uncertain information. In transportation systems some of the major sources of uncertainties include: future demand, project cost and available budget.

If we consider S plausible demand scenarios, then the stochastic formulation can be re-written as:

$$min\,Z = \sum_{s=1}^{S} P_s \left\{ \begin{array}{c} \sum_{j=1}^{T} \left\{ \left(\frac{1}{(1+r)^j} \left(\sum_{i=1}^{n_l} w_{ijs} * v_t + \sum_{i=1}^{n_l} \{ C_{vops(ij)} * VMT_{ijs} \} + N_{crs(j)} * C_{Cr} \right) \right\} \right) \\ + \sum_{i=1}^{n_p} \frac{c_i x_i(t)}{(1+r)^t} \end{array} \right\} \qquad (8)$$

In the above formulation S represents the set of scenarios, P_s denotes the probability of each scenario s, and the other parameters are the same as specified for Equation 7. The above formulation can be adapted to consider more scenarios regarding demand uncertainties, as in Sun and Schonfeld (2015).

3.2.3 Budget Constraint

Jong and Schonfeld (2001) apply a budget constraint to ensure that at any time t, $0 \leq t \leq T$, the cumulative expenditure on projects does not exceed the cumulative budget which is funded from "external" sources. In addition to their constraint, this paper considers an "internal" budget source for funding future projects. Within the analysis period, the "internal" fuel taxes collected from users are added to an external budget in determining the overall investment budget that is available. Other revenues collected from users can easily be added to the internal budget formulation. The external budget is assumed to "flow" uniformly over time in this analysis, but non-uniform budget flows can also be easily specified. The internal budget assumption is realistic, as fuel taxes and toll collections contribute substantially to highway improvement budgets. The following equation specifies how the internal budget is calculated:

$$b(t_i)_{internal} = VMT(t_{i-1}) * f_r * f_c * f_t \tag{9}$$

In the above formulation f_r, f_c, f_t denote fuel consumption rate (gal/vehicle.mile), fuel cost ($/gal), and gas tax rate (percentage of tax collected from dollar spent on gas) respectively. This formulation shows that the fuel taxes collected from period t_{i-1} contribute to the budget available in period t_i. More specifically, $VMT(t_{i-1})$ presents the vehicle miles travelled during the period in which project $i-1$ is completed. During this time, the fuel taxes collected from users are calculated and added to the budget for the next project.

Assuming that n_p is the number of candidate projects, for $0 \leq t \leq T$ the budget flow constraint is formulated as:

$$\sum_{i=1}^{n_p} c_i x_i(t) \leq \int_0^t b(t)_{external} + b(t)_{internal}\, dt \tag{10}$$

The left-hand side of the above formulation displays the total cost expended by time t, which should not exceed the cumulative budget available at that time.

3.2.4 Project Schedule

It is assumed here that projects should be funded sequentially rather than concurrently, with each successive project completed as soon as the cumulative budget permits, so that the cost savings from each completed project should start flowing as soon as possible. This in turn assumes that the cumulative budget constraint is binding, i.e. insufficient for all the available projects whose benefits exceed their costs. This situation generally prevails for transportation projects throughout the world.

It should be noted that since the cumulative budget constraint is assumed to be binding, the optimal completion time for all projects is uniquely determined for all projects in a given sequence. Thus, the optimized schedule (in continuous time rather than discrete time periods) is uniquely determined by the optimized sequence in conjunction with the cumulative budget.

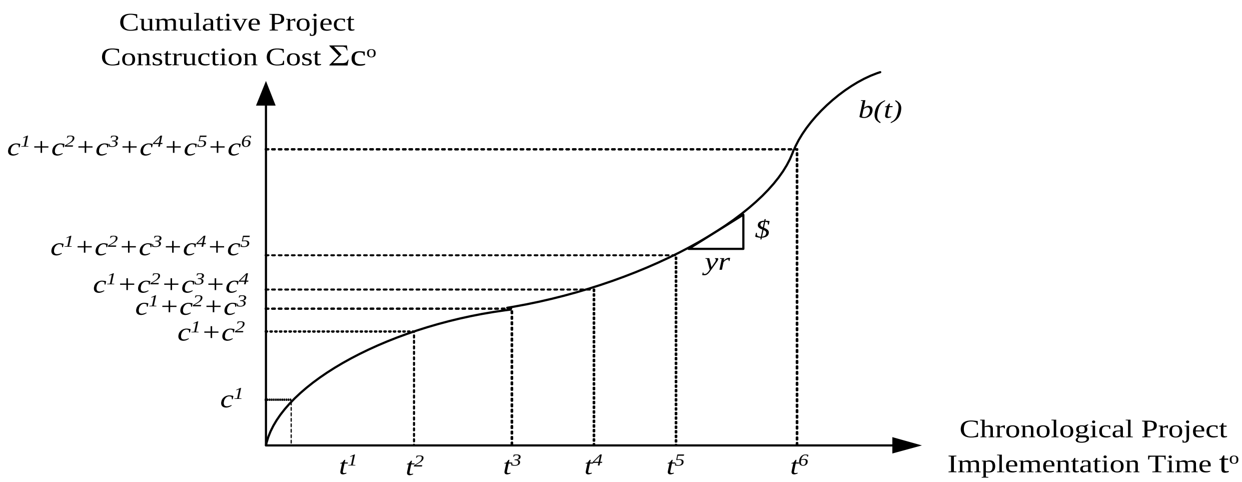


Figure 3.2-1 Project Completion Time Determined by Budget Flow and Project Costs

For any sequence, projects which are not funded within the specified analysis period (e.g. 20 years in this paper's numerical example), are effectively rejected. Construction periods that exceed the budget accumulation period of the respective project, and hence overlap with construction periods for other projects, can be considered without changing this formulation by assuming virtual borrowing. However, some modifications to the above formulation would be needed if resources other than budgets (e.g. construction equipment) were critical or if additional budget constraints (e.g. by region or type of projects) were applicable.

3.2.5 Vehicle Operating Cost

The cost of operating a vehicle on a given section is a function of costs for fuel, tires, and maintenance and repair. These costs are estimated as a function of average speed. Fuel consumption rate, tire wear rate, and maintenance and repair rate are formulated, respectively, in Equations 11 to 13 (HERS-ST technical report, 2005):

$$R_{fc} = 88.556 - 3.384 * \bar{S} + 1.7375 * \mathrm{G} + 0.053161 * \bar{S}^2 + 0.18052 * G^2 + 0.076354 * \bar{S} * \mathrm{G} \tag{11}$$

$$R_{tw} = 0.229 + 2.65 * 10^{-6} * \bar{S}^3 - 0.0403 * \ln(\bar{S}) + 0.076354 * \bar{S} * \mathrm{G} \tag{12}$$

$$R_{mr} = 48.4 + 0.00867 * \bar{S}^2 + 0.0577 * \bar{S} * \mathrm{G} \tag{13}$$

where

R_{fc} = fuel consumption rate (gallons/1000 miles)
R_{tw} = tire wear rate (% worn/1000 miles)
R_{mr} = maintenance and repair rate (% avg. cost/1000 miles)
$\bar{S}$ = average speed (miles/hour)
G = grade (%)

Figure 3.2-2, Figure 3.2-3, and Figure 3.2-4 show how fuel consumption rate, tire wear and maintenance and repair rate change with speed.

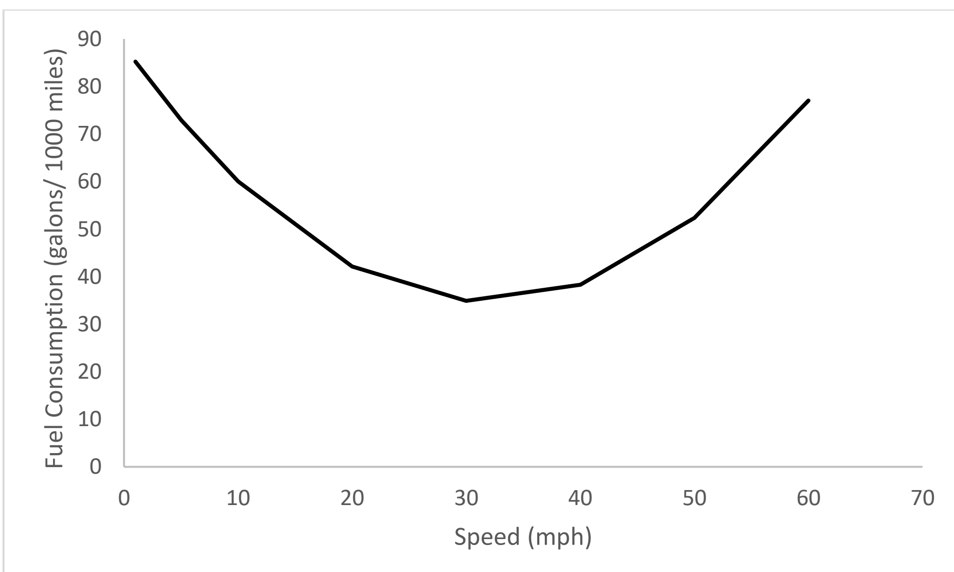


Figure 3.2-2 Fuel Consumption Rate V.S. Speed

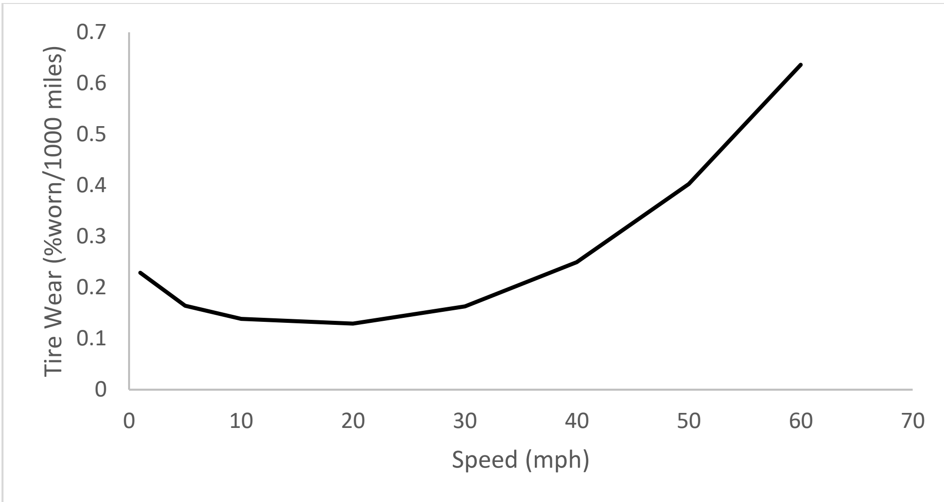


Figure 3.2-3 Tire Wear Rate V.S. Speed

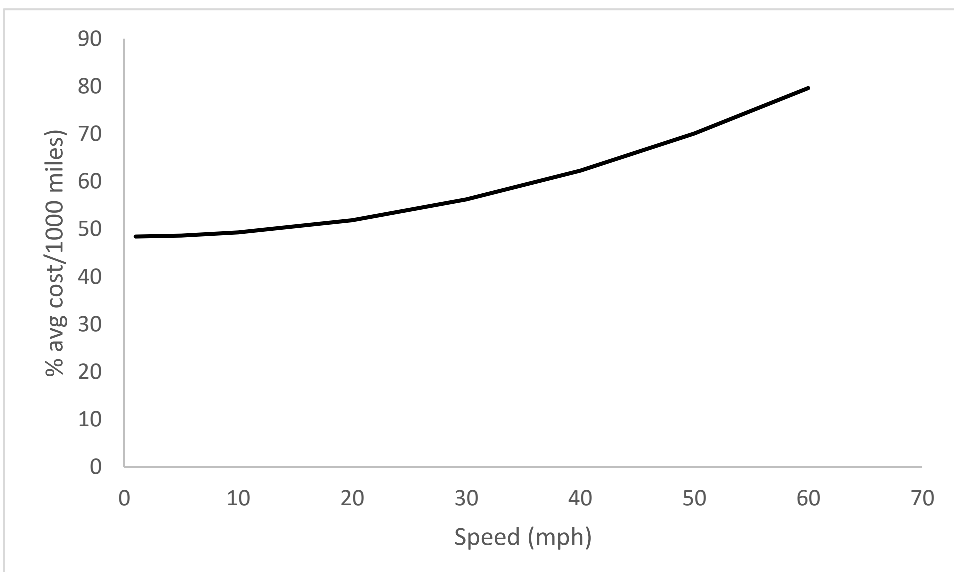


Figure 3.2-4 Maintenance and Repair Cost Rate V.S. Speed

The operating cost per vehicle-mile (C_{vop}) is estimated as the sum of the above cost components representing costs for fuel, tires, and maintenance and repair. The overall equation for combining these components is:

$$C_{vop} = (R_{fc} \times C_f + 0.01 \times R_{tw} \times C_t + 0.01 \times R_{mr} \times C_{mr}) * 0.001 \qquad (14)$$

where

C_{vop}= operating cost ($/veh. mile)

C_f = unit cost of fuel ($/gallon)

C_t = unit cost of tire ($/tire)

C_{mr}= unit cost of maintenance and repair

C_f, C_t, C_{mr} are, respectively, 2.1 ($/gallon), 105.8 ($/tire) and 151.1 ($/1000 mi). Prices are adjusted to 2015 dollars with the latest Consumer Price Index (CPI) provided by the Bureau of Labor Statistics.

3.2.6 Safety Cost

According to Highway Safety Manual (HSM, 2010) crash prediction models for two-lane and multi-lane roadway segments should include two analytical components: safety performance functions (SPFs) or baseline models and crash modification factors (CMFs). There are also calibration factors that adjust the predictions to a particular jurisdiction or geographical area. Here, we present two separate safety performance functions for two-lane and multi-lane roadway segments. The general form of the crash prediction models for roadway segments is shown in Equation 15. Equations 16 and 17 present the safety performance functions for two-lane and multi-lane roadway segments.

$$N_{cr} = C_r * N_{cr-spf} * (CMF_1 * \ldots * CMF_{12}) \tag{15}$$

$$N_{cr2-spf} = \text{AADT} * \text{L} * 365 * 10^{-6} * e^{-0.312} \tag{16}$$

$$N_{crm-spf} = \exp[-9.653 + 1.176 * \ln(AADT) + \ln(L)] \tag{17}$$

where:

N_{cr}= predicted number of crashes for a roadway segment per year.
$N_{cr2-spf}$= predicted number of crashes for two-lane roadway segments per year for nominal or baseline conditions;
$N_{crm-spf}$= predicted number of crashes for multi-lane roadway segments per year for nominal or baseline conditions;
C_r = calibration factor for roadway segments developed for use for a particular jurisdiction or geographical area.
CMF_n = crash modification factors for roadway segments.
AADT = average annual daily traffic (veh/day) on roadway segment;
L = length of roadway segment (mi).

In this model there are twelve CMFs which include CMF_1-lane width, CMF_2-shoulder width and type, CMF_3-horizontal curvature, CMF_4-super elevation deficiency, CMF_5- grade, CMF_6-driveway density, CMF_7-centerline rumble strip, CMF_8-passing lanes, CMF_9-two-way left-turn lane, CMF_{10}-roadside hazard rating, CMF_{11}-lighting, and CMF_{12}-automated speed enforcement. In this study, the values of CMF_2,..,CMF_{12} are 1.0, assuming their conditions remain the same before and after improvements. The only changing condition is lane width. Therefore, the algorithm estimates CMF_1 (lane width) for each segment from the following equation:

$$CMF_1 = (CMF_{ra} - 1) * P_{ra} + 1 \tag{18}$$

where:

CMF_1 = crash modification factor for the effect of lane width on total crashes.
CMF_{ra} = crash modification factor for related crashes (run-off-the-road, head-on, and sideswipe) calculated from Table 3.2-1.
P_{ra} = proportion of total crashes constituted by related crashes (with 0.574 as the default value).

Table 3.2-1 Values of CMF1 for Lane Width on Roadway Segments (HSM, Table 10-8)

Lane width (ft)	ADT<400 (veh/day)	ADT =400 to 2000 (veh/day)	ADT>2000 (veh/day)
9	1.05	1.05+0.000281*(ADT-400)	1.50
10	1.02	1.02+0.000175*(ADT-400)	1.30
11	1.01	1.01+0.000025*(ADT-400)	1.05
12	1.00	1.00	1.00

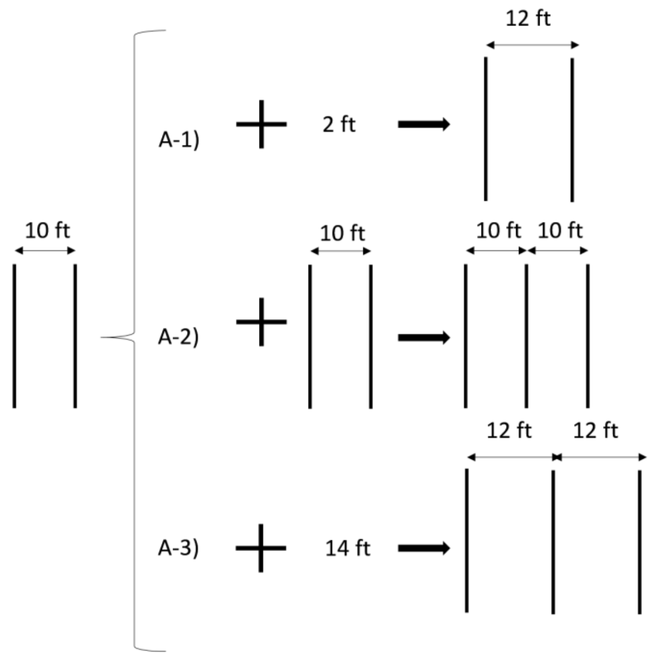
From Table 7-4 HSM (2010) (Societal Crash Costs by Severity) and Table 10-3 HSM (2010) (Default Distribution for Crash Severity Level), it is assumed that 32.1% of total crashes are “fatal and injury” (FI) and 67.9% are “property damage only” (PDO). Therefore, cost for one predicted crash ($Cost_{Cr}$) would be calculated as: 0.321*172,438 ($/FI crash) + 0.679*8,066 ($/PDO crash) = $60,830 / Crash. All costs are adjusted to 2015 dollars using an inflation factor from the latest Consumer Price Index (CPI) provided by the Bureau of Labor Statistics.

3.3 Design of Improvement Alternatives

The algorithm presented in this paper has the capability to consider multiple improvements over time at the same location as well as improvements throughout a network. These improvements include widening existing narrow lanes (from 10 ft. to 12 ft.), adding one or multiple narrow lanes (10 ft. wide) and adding one or multiple wide lanes (12 ft. wide). The alternatives considered for each link depend on the existing link characteristics, and are symmetric, i.e. the same for both directions of a link. According to the Highway Capacity Manual (HCM, 2010), lane widths under 12 ft. reduce travel speed, and thus also reduce operational capacity. In this case, it is assumed that the narrow and wide lanes are, respectively, 10ft. and12 ft. wide. According to HCM (2010), widening lanes from 10 ft. to 12 ft. would increase the capacity by 15%. The following list shows the set of improvement alternatives at each location:

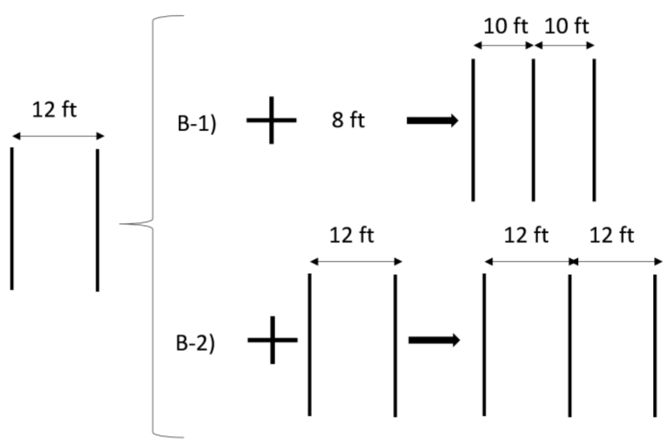
A. If the existing link has narrow lanes:

    1. Widen the existing lanes.
    2. Add one narrow lane.
    3. Widen existing lanes and add one wide lane.



B. If the existing link has wide lanes:

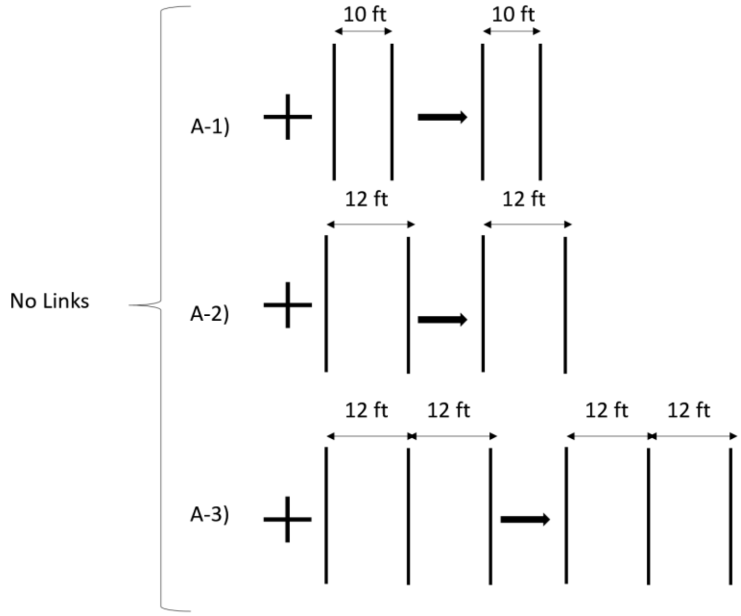
1. Add enough width for two narrow lanes. (In this case n wide lanes are transformed to n+1 narrow lanes.)
2. Add one wide lane.



C. If the there are no existing lanes (new development):

1. Add one narrow lane.
2. Add one wide lane.

3. Add two wide lanes.



For each case A, B and C the potential improvements are listed in increasing order of project costs. In this case the algorithm first evaluates the characteristics of each location in terms of existing narrow/wide lanes, and whether new lanes can be added. (In some locations new lanes cannot be added due to land availability constraints.) Then, based on the current condition, the above set of improvements are considered at each location.

Now there are two problems to be resolved. First, which links (locations) should be selected for improvement and in what sequence and when should those links be improved? Second, at each location, which alternative should be selected and implemented? The first problem is solvable by using the combination of the GA and the traffic assignment model (or CTM in future). This method will be explained further below.

One way to address the second problem is to compute the benefit-cost ratio of each alternative and select the best one at each location. This myopic search is quite prevalent in current practices. However, it disregards the interrelation among projects. A simple benefit-cost ratio in this case cannot capture the impact of selected projects on future project implementations. In other words, due to interrelations among network links, the alternative with lower benefit-cost ratio may be more beneficial if considered in the long run (over the entire analysis period) or in conjunction with other alternatives (e.g. in a series of links that remove all bottlenecks rather than just shifting them). Therefore, it seems preferable to consider all possible improvements at each location over the entire analysis period and allow the algorithm to evaluate them over the planning horizon. This means that the GA will both optimize the selection and sequence of projects among links in the network as well as optimize the selection of alternatives at each location, all within one optimization process. This will result in more search steps and increased computation time. In order to tackle this issue and guide the search process, we assign selection probabilities to each alternative based on project costs. This means that under each case the less costly alternatives have a higher probability of being selected. This is reasonable since in practice it is more desirable to start with less costly improvements, and later move to more expensive ones. More specifically, the selection probability of improvements at each location is inversely proportional to their relative costs. If M improvements are considered at one location, the probability of selecting each improvement *Pr(m)* is:

$$\Pr(m) = \frac{1/_{Cost(m)}}{\sum_{i=1}^{M}\left(1/_{Cost(i)}\right)} \tag{19}$$

3.4 Development of the Evaluation Model

3.4.1 User Equilibrium Traffic Assignment

Basic traffic assignment models are suitable methods for estimating the traffic-related attributes for unsaturated networks with steady flows. These attributes include travel time, traffic flow, speed and volume-capacity ratio over all the links in the network. This information is useful for estimating the cost savings resulting from capacity improvements and therefore supports a proper evaluation method for selection, sequencing and scheduling of projects. These cost savings mainly pertain to the travel time reduction for users and can be obtained by running the traffic assignment model at different stages of the Genetic Algorithm (GA) to compute w_{ij} and, hence, the objective function (Equation 7).

This convex combination algorithm is used here to evaluate improvement projects upon their implementation in the network. The Frank–Wolfe algorithm is an iterative first-order optimization algorithm for constrained convex optimization mostly used for solving traffic assignment problems. In each iteration, the Frank–Wolfe algorithm considers a linear approximation of the objective function, and moves slightly towards a minimizer of this linear function. A direction search is performed by solving a linear approximation of the objective function at each iteration. At this stage a step size is determined which leads the search towards that direction. Finally, the algorithm terminates when convergence criteria are met, which in this case is based on the similarity of successive solutions. In this case, the traffic assignment algorithm provides a relatively simple model for evaluating improvement projects (i.e. adding new links or expanding existing links), and estimating link travel time, speed, and volume.

Given a current travel time for link a, t_a^{n-1} the *n*th iteration of the convex combination algorithm is summarized as follows:

1. *Initialization*: all or nothing assignment assuming t_a^{n-1} which yields x_a^n.
2. *Updating travel time*: using a BPR function $t_a^n = t_a(x_a^n) = t_0(1 + 0.15\,(\frac{v}{c})^4)$.
3. *Direction finding*: all or nothing assignment considering t_a^n which yields auxiliary flow y_a^n.
4. *Line search*: find α that solves $min \sum_a \int_0^{x_a^n+\alpha(y_a^n-x_a^n)} t_a(\omega)d\omega$.
5. *Move:* set $x_a^{n+1} = x_a^n + \alpha_n(y_a^n\text{-}\ x_a^n),\ \forall\alpha$.
6. *Convergence test*: If a convergence criterion met, stop. Otherwise set n=n+1 and go to step 1.

3.4.2 Cell Transmission Model

Traffic assignment model is a relatively simple method which is good for algorithm testing purposes. However more complex methods are desirable for evaluating the objective function. In general, traffic operations models can be microscopic, mesoscopic or macroscopic. Microscopic simulations assume that the behavior of an individual vehicle is a function of the traffic conditions in its environment. Mesoscopic and macroscopic models assume that the aggregate behavior of sets of vehicles depends on the traffic conditions in their environment (Daganzo, 1994). More specifically, Mesoscopic models analyze transportation elements in small groups, within which elements are considered homogeneous while Macroscopic models deal with aggregated characteristics of transportation elements.

This study proposes to employ a mesoscopic model rather than a microscopic model for two reasons:

1. Although microscopic simulations usually monitor each vehicle’s destination, their assumptions are difficult to validate because humans’ behavior in real traffic (not in contrived “car-following” experiments) is difficult to observe and measure. This is unfortunate because for a simulation to work the microscopic details have to be just right.
2. At planning level, which is the case of this research, the networks are quite complex and the analysis is conducted over multiple years. Therefore, in most cases microsimulation is computationally expensive, and are more suitable to be used at operation level where networks are simpler, but more detail is required about the system performance.

In the last decades, the cell transmission model (CTM) by Daganzo (1994, 1995) has been used to capture shockwave and link propagation properties. CTM is a mesoscopic traffic model in which the physical length of roadways is divided into a number of cells which takes into account the traffic properties such as flow and density and thereby captures link spillovers and shockwave propagation. Ziliaskopoulos (2000) used the CTM to develop a system optimal problem as a linear programming. Ukkusuri and Waller (2008) compared the dynamic user equilibrium and dynamic system optimal by linear formulation with different objective functions.

This study aims to use a CTM model which is more complex than a traffic assignment model to evaluate the objective function. A Cell Transmission Model (CTM) has the ability to consider queuing in the network and capture saturation effects in congested traffic networks. This model will be explained in detail in chapter 4.

3.5 Development of the Optimization Model

In this problem, the objective function cannot be explicitly expressed in terms of the decision variable t_i. Therefore, it is not guaranteed to possess the desirable properties such as convexity and differentiability. In other words, the gradients of the objective function cannot be directly derived with respect to their corresponding decision variables. This makes it difficult to evaluate the objective function. Consequently, gradient-based research methods, integer and linear programming are ineffective in this case. Dynamic programming is also inapplicable since its independence and additive assumptions are violated in this problem. Moreover, as the number of candidate projects increases, the scope of the problem exceeds the capabilities of conventional mathematical optimization methods. Consequently, heuristic methods have become popular for solving such problems (Tao and Schonfeld, 2007; Wang and Schonfeld, 2008; Bagloee and Asadi, 2015). This study finds a GA very useful for effectively finding near-optimal solutions for such a large solution space and noisy objective function. The goal of this study is to employ the GA to solve the optimization model jointly with a traffic assignment model (and later with a Cell Transmission Model) which is used to evaluate the objective function. In other words, the GA optimizes the selection and sequence of projects while the traffic assignment model estimates variables such as travel time, speed and volume for evaluating the benefits and costs of projects. The final results determine which links should be selected for expansions and which new links should be added, in what order, and when they should be completed over the horizon period *T*.

3.5.1 Genetic Algorithm

A Genetic Algorithm (GA) is a metaheuristic method that mimics the process of natural selection and is a successful optimization method in a wide range of fields. GAs get a set of possible solutions called the population. Each individual in the population is specified by a string of encoded genes which is called a chromosome. In this process some individuals are selected to reproduce off springs and since each individual has a probability of selection according to its fitness value, better ("fitter") solutions have a higher opportunity of being selected. The selected solutions are then processed through a series of crossover and mutation operators which create offspring and change their attributes while maintaining the diversity of the population. Designing an appropriate GA can lead to an optimal or near optimal solution.

3.5.2 Solution Representation

The solutions are represented by the sequence of projects in which projects are implemented. In this setting, each project has to occur after all its predecessors and after all its successors. Figure 3.5-1 represents an example of a feasible solution.

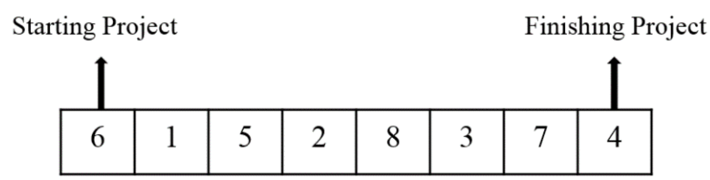


Figure 3.5-1 Example of a Feasible Solution

3.5.3 Initial population

In general, solutions of GAs are mostly represented by binary digits and the initial population is generated randomly. In this research, each individual in a population is defined by a string including a sequence of numbers each corresponding to a specific project. In addition to random order solutions, two other methods comprising greedy-order solutions and bottleneck-order solutions are used to create the initial population (Jong 2001). In greedy-Order solutions, projects are selected based on their benefit-cost ratio, regardless of their interrelations. In bottleneck-order solutions, projects are ranked based on the link volume-capacity ratio which describes the congestion severity over that link. This assumes that more congested links should have higher priority for being implemented.

3.5.4 Fitness function and parent selection

The fitness function is considered equivalent to the value of the objective function (NPV of total cost) and it is computed through the traffic assignment model. The selection probability is generally based on the value of the objective function in maximization problems. Therefore, in minimization problems the selection probability varies inversely with the objective function value. However, for preventing some undesirable properties of prematurity, a ranking method is applied instead (Wang 2001). In this method, the population is sorted with nonlinear ranking from the best to the worst. Then the selection probability of each chromosome is assigned according to its exponential ranking value considering the lowest fitness value equal to one (Michalewicz 1995). Let q be the selective pressure$\in [0,1]$, the selection probability is defined as follows:

$$P_i = c * q(1-q)^{i-1}, \quad c = 1/[1-(1-q)^{PopSize}] \tag{20}$$

Next, a roulette wheel approach is incorporated to select appropriate parents based on their selection probabilities (Michalewicz 1995). This process is conducted by spinning the roulette wheel *pop_size* times. Each time a random number *r [0,1]* is generated, then the i_{th} chromosome is selected such that $w_{i-1} < r \leq w_i$, where w_i is the cumulative probability for each chromosome.

3.5.5 Crossover and mutation

Then a crossover and a mutation operator are applied to reproduce offspring and create the new population. Common methods of mutation and crossover are not very efficient for sequencing problems since they construct many infeasible solutions with repetitive project numbers in one sequence. To avoid producing such solutions, some other genetic operators are employed to solve the project sequencing problem. These crossover and mutation operators are described below adapted from Wang (2001):

Crossover operators:

1. Partial Mapped Crossover (PMX)

   Proposed by Goldberg and Lingle (1985), this two-point crossover exchanges the sequence of projects between two random positions in the selected parents. Then a mapping mechanism is established to correct for the possible duplication of projects by replacing the repeated projects by their corresponding projects. Figure 3.5-2 illustrates this process.

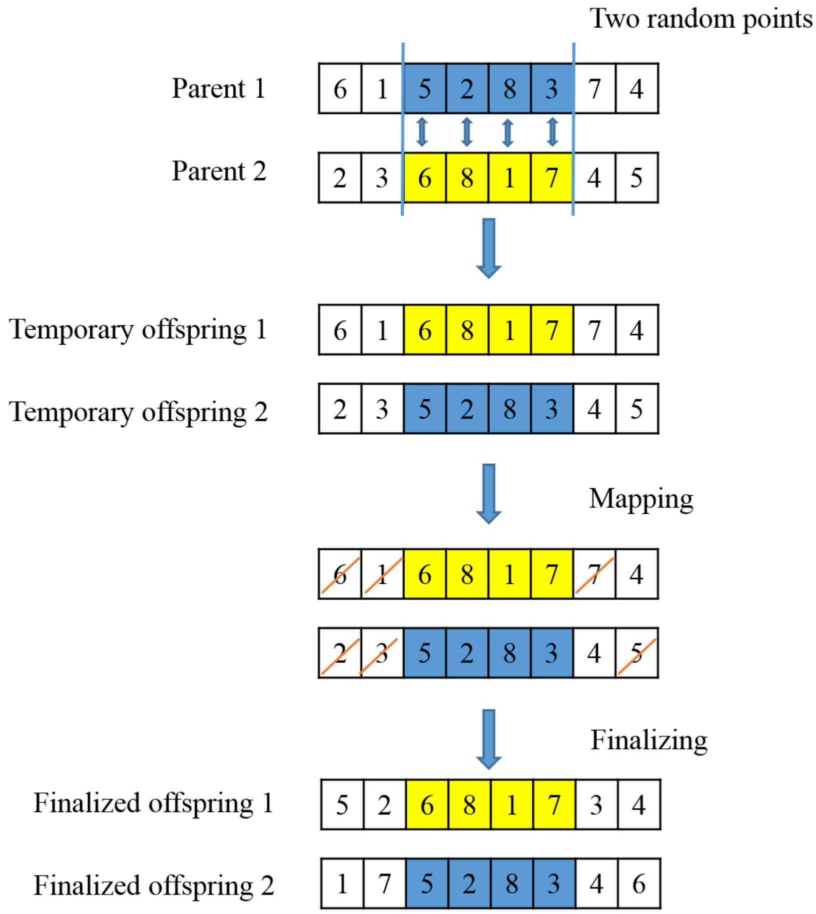


Figure 3.5-2 Example of Partial Mapped Crossover (PMX)

2. Position Based Crossover (PBX)

The PBX operator was proposed by Syswerda (1991). In this multi-point crossover, a set of random positions are selected from the first parent and copied to the same positions in the offspring. Then the projects that already exist in the offspring are deleted from the second parent and the rest are copied to the offspring with their original order. (Figure 3.5-3)

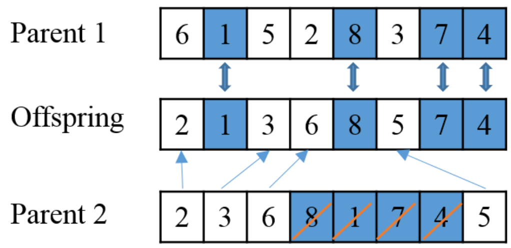


Figure 3.5-3 Example of Partial Based Crossover (PBX)

3. Order Crossover (OX)

   This two-point crossover operator was introduced by Davis (1985). This operator works by selecting two random points in the first parent and copying the sequence in between those points to the new offspring, keeping their original positions. The copied projects are deleted from the second parent and the remaining projects are inserted to the vacant positions in the offspring while keeping their order. (Figure 3.5-4)

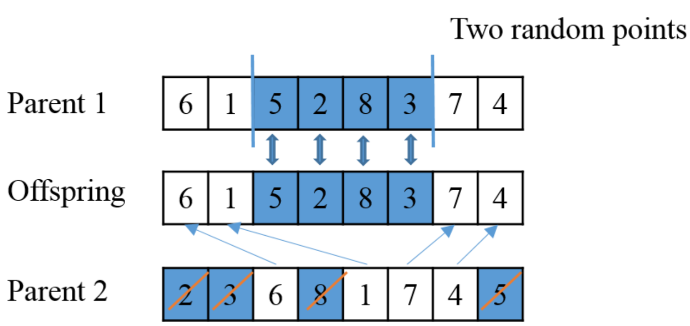


Figure 3.5-4 Example of Order Crossover (OX)

4. Order Based Crossover (OBX)

This operator also proposed by Syswerda (1991) is similar to PBX but imposes the selected positions in one parent on the corresponding projects in the second parent. (Figure 3.5-5)

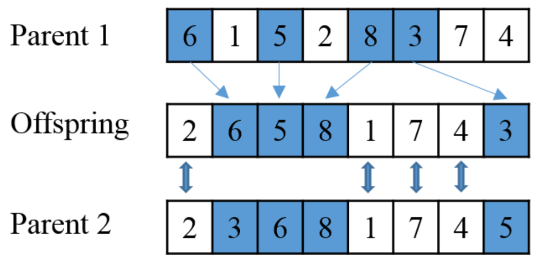


Figure 3.5-5 Example of Order Based Crossover (OBX)

5. Edge Recombination Crossover (ERX)

The edge recombination operator (ERX) is an operator that creates offspring exclusively by looking at the edges rather than the vertices. The idea here is to use as many existing edges, or node-connections, as possible to generate children. This operator is specifically useful when a genotype with non-repeating gene sequences is needed such as for the sequencing problem in this study. The method is introduced by Whitley et.al. (1989).

For each project *i*, the edge list consists of all other neighbor projects connected to project *i* from both parents. The construction of the offspring begins by selecting a project with the lowest number of edges. In case projects have equal number of edges, one of them is randomly chosen. The selected project is then crossed out from all the other edge lists, and the procedure continues by selecting the next project with the smallest number of edges until all projects are selected. Figure 3.5-6 shows an example of the ERX operator.

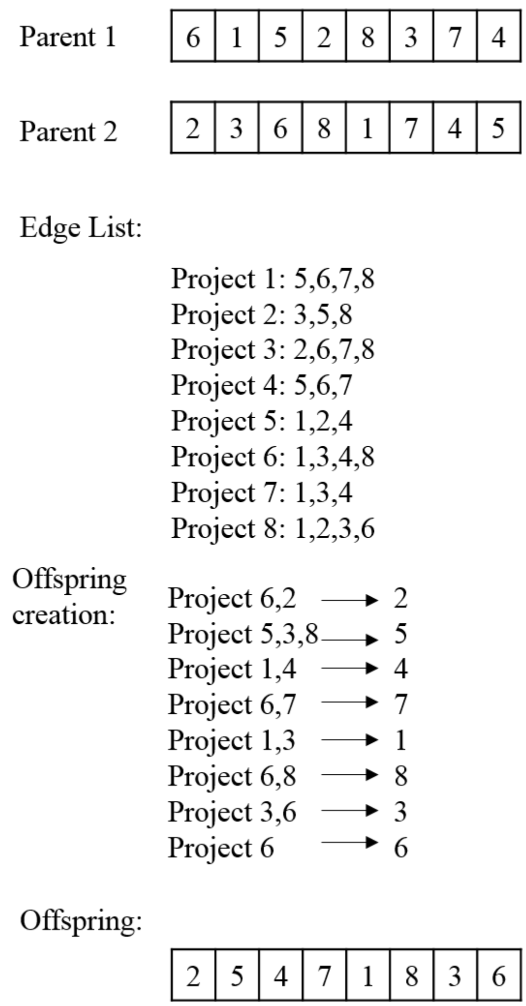


Figure 3.5-6 Example of Edge Recombination Crossover (ERX)

Mutation operators:

6. Insertion Mutation (IM)

   In this operator a project is randomly selected and is inserted to a random position. Other projects are shifted over while keeping their original sequence. (Figure 3.5-7 a)

7. Inversion Mutation (VM)

This operator selects two random positions and inverts the subsequence between those two points. The other projects keep their positions. (Figure 3.5-7 b)

8. Reciprocal Exchange Mutation (EM)

The EM operator simply exchanges the position of two random projects while other projects keep their original order. (Figure 3.5-7 c)

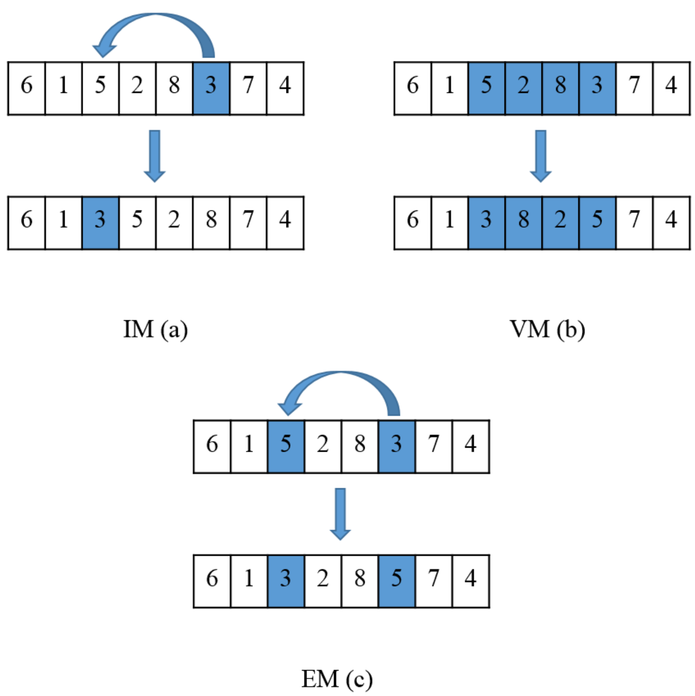


Figure 3.5-7 Example of Mutation Operators

The reproducing process randomly selects one operator and applies it on the selected parents.

3.5.6 Termination Criterion

In this case the termination criterion is based on the similarity of successive solutions. If the solution of the algorithm remains the same after 10 successive generations, then the algorithm stops.

Chapter 4 Case Study Networks

4.1 Sioux Falls Network

Figure 4.1-1 presents the Sioux Falls network consisting 24 nodes and 76 links which is used in LeBlanc et.al (1975). This network is suitable for testing user equilibrium and the metaheuristic algorithms. In this study a slightly different version of Sioux Falls is used which is displayed in Figure 4.1-2. Table 4.1-1 describes the hourly travel demand between each origin destination pair. These numbers are assumed as the peak hour demand and the off-peak travels is considered half of these values. It is also assumed that the demand increases exponentially as a function of time over the planning horizon as follows:

$$d_{ij}^{t} = d_{ij}^{0} * (1+r)^{t} \tag{21}$$

Where d_{ij}^{t} is the demand between origin i and destination j at time t, d_{ij}^{0} is the base demand for the ij O/D pair at time 0, and r is the growth rate.

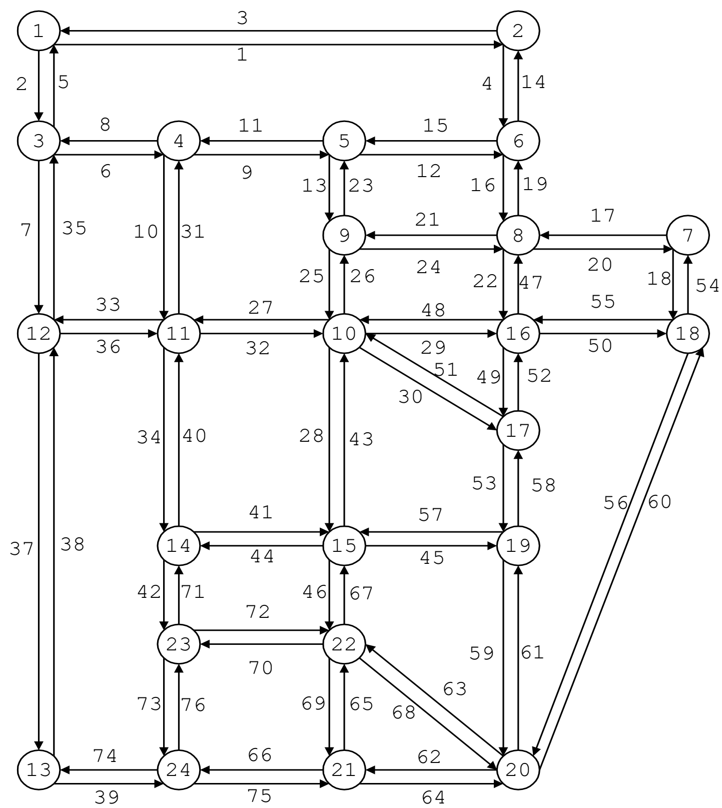


Figure 4.1-1 Sioux Falls Network (Le Blanc, 1975)

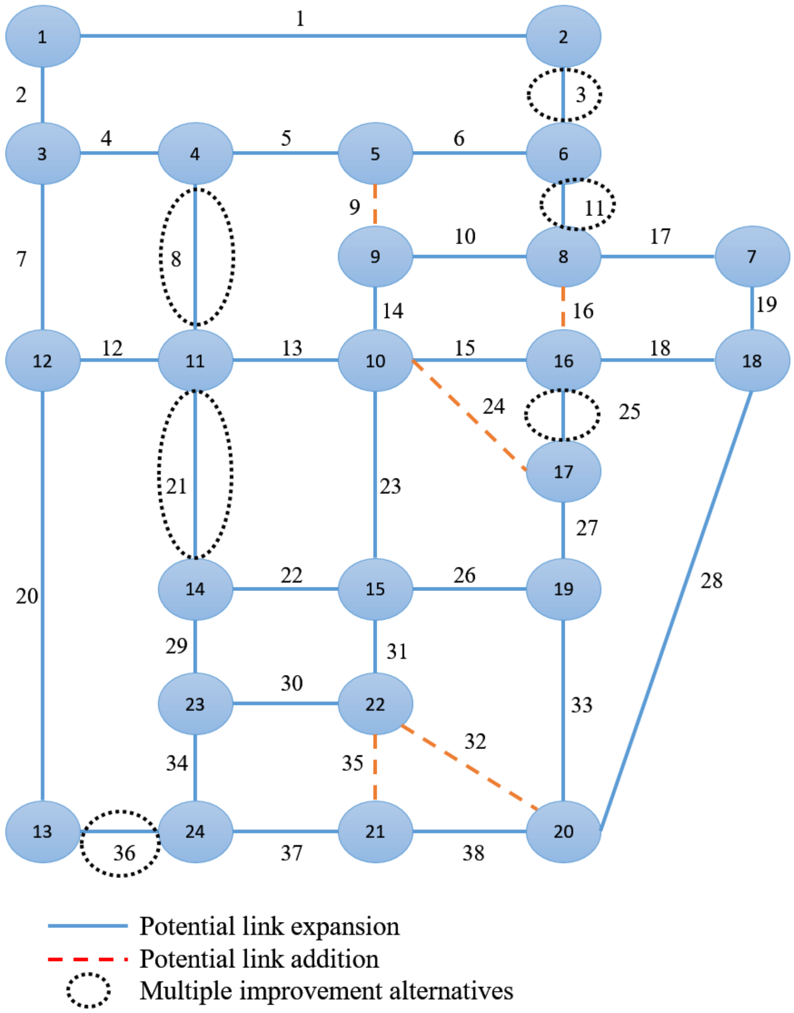


Figure 4.1-2 Sioux Falls Network Used in This Study

Table 4.1-1 Trip Table between Each Two Node Pairs (Vehicle per Hour)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	20	12	36	18	24	30	48	36	84	36	18	36	18	30	36	30	12	18	18	6	24	18	12
2	20	0	6	18	6	30	12	30	18	36	12	12	18	6	12	24	18	6	6	12	6	12	6	6
3	12	6	0	18	6	18	6	12	12	18	18	18	12	6	6	12	6	0	6	6	6	6	6	6
4	36	18	18	0	30	30	30	42	48	72	90	42	36	30	30	48	30	6	18	24	12	24	30	18

5	18	6	6	30	0	18	12	36	48	60	36	12	12	12	18	36	18	6	12	12	6	12	12	6
6	24	30	18	30	18	0	24	48	24	48	24	18	18	12	18	60	36	6	18	24	6	18	12	6
7	30	12	6	30	12	24	0	66	36	114	30	48	30	18	30	84	60	60	30	36	18	36	12	6
8	48	30	12	42	36	48	66	0	0	96	54	36	36	24	42	132	84	18	42	54	24	36	24	12
9	36	18	12	48	48	24	36	0	0	168	90	42	36	36	60	90	60	12	30	42	24	42	36	12
10	84	36	18	72	60	48	114	96	168	0	240	126	114	132	240	264	0	42	108	156	78	162	108	54
11	36	12	18	90	36	24	30	54	90	240	0	90	60	96	90	84	60	12	30	42	30	66	84	36
12	18	12	18	42	12	18	48	36	42	126	90	0	84	42	48	42	42	12	18	30	24	48	42	30
13	36	18	12	36	12	18	30	36	36	114	60	84	0	36	42	42	36	6	24	42	36	78	48	48
14	18	6	6	30	12	12	18	24	36	132	96	42	36	0	45	42	42	6	24	30	24	72	66	24
15	30	12	6	30	18	18	30	42	60	240	90	48	42	45	0	78	90	18	48	66	48	156	60	30
16	36	24	12	48	36	60	84	132	90	264	84	42	42	42	78	0	168	100	84	102	36	72	36	18
17	30	18	6	30	18	36	60	84	60	0	60	42	36	42	90	168	0	42	102	102	42	102	36	18
18	12	6	0	6	6	6	60	18	12	42	12	12	6	6	18	100	42	0	24	100	6	24	6	6
19	18	6	6	18	12	18	30	42	30	108	30	18	24	24	48	84	102	24	0	78	30	78	24	12
20	18	12	6	24	12	24	36	54	42	156	42	30	42	30	66	102	102	100	78	0	78	0	42	30
21	6	6	6	12	6	6	18	24	24	78	30	24	36	24	48	36	42	6	30	78	0	114	42	36
22	24	12	6	24	12	18	36	36	42	162	66	48	78	72	156	72	102	24	78	0	114	0	0	72
23	18	6	6	30	12	12	12	24	36	108	84	42	48	66	60	36	36	6	24	42	42	0	0	48
24	12	6	6	18	6	6	6	12	12	54	36	30	48	24	30	18	18	6	12	30	36	72	48	0

The Sioux Falls network illustrated in Figure 4.1-2 is selected for demonstrating the performance of the proposed algorithms. As mentioned earlier, this is not considered a realistic network since it mainly includes the city's major arterial roads and omits many characteristics of its transportation system. However, it has widely been used to examine and compare studies on networks (LeBlanc et.al 1975).

4.1.1 Project Interdependence

Conceptually, if the capacity increases in one link the network, congestion and average travel times tend to increase in other links that are "in series" with it and decrease in its "parallel" links. Table 4.1-2 shows a sample of the travel time savings from separate implementation of projects in the network. The second column presents the initial link travel times prior to project implementations while columns three to seven present the travel time reductions for single projects. Positive values indicate travel time reductions, while negative values show increases in travel time due to network interdependence. The bolded numbers indicate the travel time changes in the location of the expanded links. These numbers are relatively higher since the expanded links gain direct benefits after project implementation. Notably, the sum of all the cells in one column is not equal to the travel time changes on the links which are getting expanded. This, in effect, confirms the interrelation among links and the possible shifting of bottlenecks to surrounding links. Furthermore, the last column implies that the total system cost reduction from implementing two projects together is different from the sum of cost savings for the two individual projects, emphasizing that the cost saving of multiple projects is not a linear summation of their individual savings.

Table 4.1-2 Travel Time Reduction due to Link Expansion

link	Link travel time without projects	link travel time reduction (min/veh) expanding 2 & 5	expanding 4 & 14	expanding 6 & 8	expanding 10 & 31	expanding 13 & 23	expanding (2&5)&(4 &14)
1	3.594	0.006	-0.018	0.008	0.000	-0.004	-0.023
2	5.021	**1.115**	0.720	-0.408	-0.062	0.031	**1.659**
3	3.594	0.006	-0.018	0.008	0.000	-0.004	-0.023
4	10.356	2.338	**5.712**	3.468	0.658	-0.342	**7.240**
5	5.021	**1.115**	0.720	-0.408	-0.062	0.031	**1.659**
6	4.550	-0.346	0.570	**0.927**	-0.269	-0.144	0.736
7	2.618	0.016	0.041	0.032	0.021	0.041	0.052
8	4.550	-0.346	0.570	**0.927**	-0.269	-0.144	0.736
9	1.629	-0.093	0.166	-0.060	0.015	-0.206	0.164
10	7.374	-0.038	-4.221	-4.398	**2.803**	2.051	-2.308
11	1.629	-0.093	0.166	-0.060	0.015	-0.206	0.164
12	3.001	-0.344	0.340	-0.526	-0.272	-0.851	0.367
13	7.390	0.053	1.214	0.933	0.883	**2.392**	1.013
14	10.356	2.338	**5.712**	3.468	0.658	-0.342	**7.240**
15	3.001	-0.344	0.340	-0.526	-0.272	-0.851	0.367
16	7.882	-0.494	-1.261	-0.065	0.881	2.222	-2.483

17	1.796	0.013	0.089	0.028	0.015	0.018	0.087
18	1.312	0.000	0.001	0.001	0.000	0.000	0.001
19	7.882	-0.494	-1.261	-0.065	0.881	2.222	-2.483
20	1.796	0.013	0.089	0.028	0.015	0.018	0.087
21	7.333	0.691	1.464	1.392	0.647	0.267	1.475
22	4.369	0.151	0.788	0.999	0.500	0.378	0.537
23	7.390	0.053	1.214	0.933	0.883	**2.392**	1.013
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
75	3.287	0.090	0.189	0.164	-0.190	-0.173	0.139
76	1.431	-0.003	0.041	-0.087	0.050	0.033	0.079
Total travel time saving		9.753	15.656	8.054	13.632	18.037	25.216

4.2 Anaheim Network

In addition to Sioux Falls network which is fairly small, this method is also applied to the much larger Anaheim network, which is displayed in Figure 4.2-1. It has 416 nodes (of which 38 are origin/destination centroids), 914 links, and 1406 O-D pairs. All the network-related information is extracted from Bar-Gera (2011).

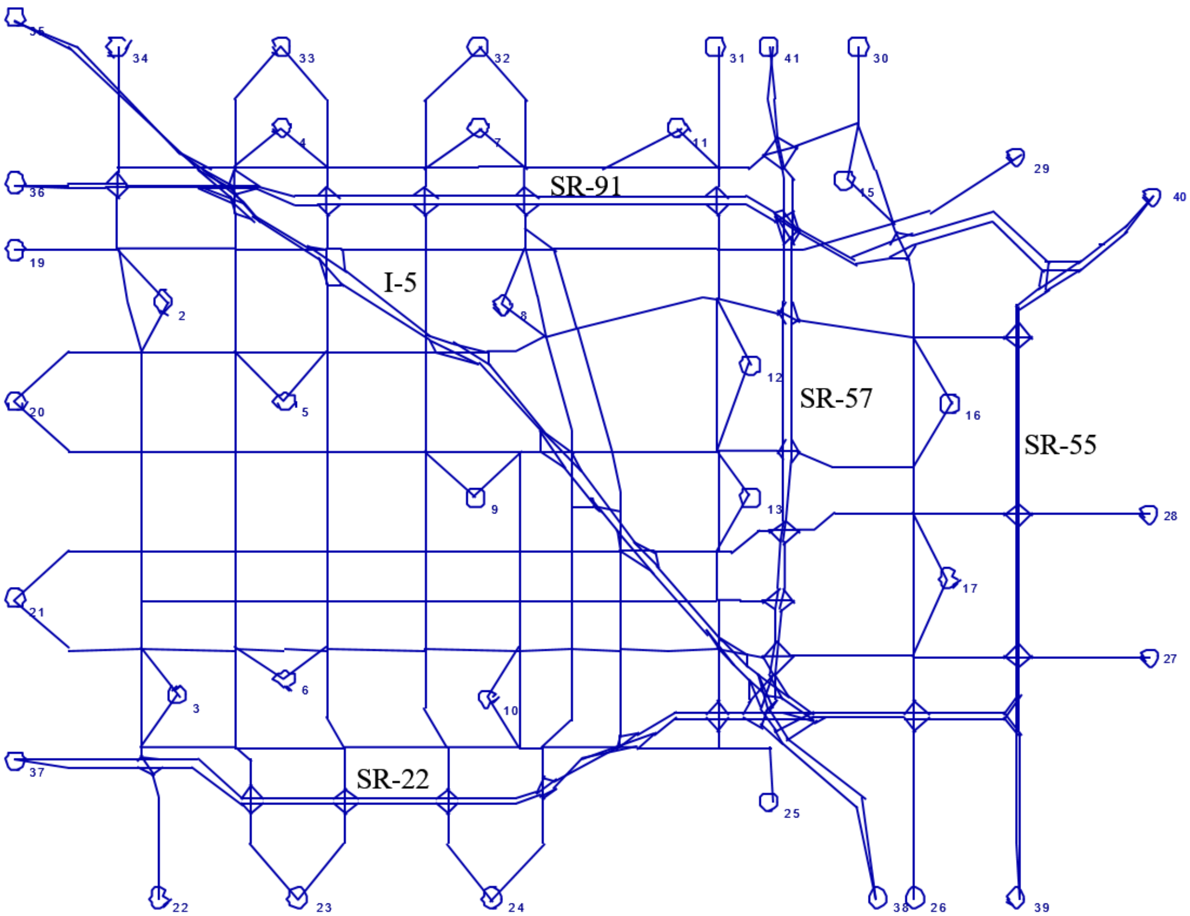


Figure 4.2-1 Anaheim Network

Chapter 5 Traffic Assignment Application and Results

In this paper, the Sioux Falls network (LeBlank et al., 1975) is used as a case study for the proposed model. This network consists of 24 nodes and 33 links as depicted in Figure 4.1-2, and is widely used in the literature for testing purposes. The main cost savings of link expansion projects are the reduced travel time, vehicle operating cost, and number of crashes for all the users. These parameters can be computed through the traffic assignment model by comparing the total system cost before and after project implementation. Next, we use the GA described in previous sections to find near optimal solutions for the sequence and schedule of selected projects. When optimizing, it is desired to find a sequence of projects which can be implemented within the planning horizon. Therefore, each project with a scheduled completion time after the planning horizon is eliminated from the sequence. Additionally, the projects with unacceptable marginal benefit-cost ratio are discarded form the sequence list during the evaluation stages and replaced by other justifiable projects.

In Figure 4.1-2, the dashed lines indicate the potential locations for adding new lanes. In this case there are three potential alternatives described in the section 3-2 (case C). The links surrounded by dashed circles indicate cases A and B with multiple alternatives at each location. The other links only have one potential improvement.

In this example, the narrow and wide lanes have a capacity of 1000 and 1150 vehicles/hour, respectively (HCM, 2010). In the following numerical examples, the equivalent annual cost of constructing roads is assumed to be 396,000 $/mile per foot of road width (Zhang et al., 2013).

Therefore, the cost of widening a lane, adding one narrow lane, and one wide lane are 792,000 ($/mile), 3,960,000 ($/mile) and 4,752,000 ($/mile), respectively.

The following table summarizes the potential improvements for all the links.

Table 4.2-1 Improvement alternatives for different links

Case	Link Number	Possible improvements
A	3, 11, 25	-Widen the existing lanes. -Add one narrow lane in each direction. -Widen existing lanes and add one wide lane.
B	8, 21, 36	- Add enough width for two narrow lanes. - Add one wide lane in each direction.
C	9, 16, 24, 32, 35	- Add one narrow lane in each direction. - Add one wide lane in each direction. - Add two wide lanes.
D	All other links	-Add one narrow lane in each direction.

The first step is to identify critical lanes with high volume-capacity ratios to form the first set of candidate projects. This is done by running the traffic assignment model with the given O/D demand matrix which is symmetric for O/D pairs. It is assumed that the improvement projects, whether expanded or newly added links, are implemented in both directions between two nodes. This makes sense because the demand matrix is symmetric and also because it saves costs to use mobilized construction equipment and other resources for both directions of a link. In this problem, the potential new links (shown by dashed lines in Figure 4.1-2) are actually existing links in the original network which are treated as potential new links. Initially, the algorithm considers zero capacity for these links and later examines whether and when they should be added to the network. It should be noted that a multi-period analysis is incorporated in this model to account for cyclical demand fluctuations during the day. While only peak and off-peak periods are presently considered, the number of periods per day can be easily increased.

After determining the initial set of candidates, the algorithm considers multiple improvement projects at each location based on Table 4.2-1. Then, all projects are investigated through a benefit-cost analysis to identify the economically beneficial projects and rank them based on their benefit-cost ratio. Thus, we can obtain two set of initial solutions, one based on volume-capacity ratio (bottleneck order solution) and the other based on benefit-cost ratio (greedy-order solution). These two set of initial solutions are later used as the initial population in the GA. Note that the optimal sequence of projects is different from the benefit-cost ratio and volume-capacity ratio ranked lists. These two sets of projects are only useful as good initial solutions for the GA.

5.1 Optimal Sequence and Schedule

The analysis begins by running the traffic assignment model to assess the travel times and traffic volumes before and after improvement projects. Then the GA is used to find the near-optimal solution for selecting and scheduling projects. At this stage the algorithm selects one improvement from a set of multiple improvement alternatives at each location following the procedure explained in section 3-3 and the probabilistic Equation 20. Ultimately, the GA yields the optimal project selection at each location, the order of their implementation, and the schedule of completing each one. In this study, we assume a 20-year planning horizon. That is, projects with scheduled completion time after the planning horizon are eliminated from the sequence. Table 5.1-1 presents the results from the stochastic model which yields the optimal selection of projects along with their order and schedule time. In this sequence, link 16 is a new link and the rest are lane addition projects. The results also indicate the optimal improvement type for each link which is also obtained from the GA results. As stated earlier, the stochastic model yields the optimized sequence which directly determines the optimized schedule. It is shown in the third column that this method is capable of determining the schedule of projects in continuous time.

Table 5.1-1 GA Optimal Sequence and Schedule

Project rank	**Project #**	**Improvement Type (on both directions)**	**Completion Time (year)**
1	25	Widen existing lanes	0.19
2	34	Add one narrow lane	1.13
3	36	Add one wide lane	4.60
4	14	Add one narrow lane	5.03
5	22	Add one narrow lane	7.17
6	16	Add a new link with a narrow lane	9.32
7	11	Widen existing lanes	9.94
8	15	Add one narrow lane	11.61
9	30	Add one narrow lane	13.35

10	3	Widen existing lanes	13.99
11	37	Add one narrow lane	15.90
12	2	Add one narrow lane	19.24
	PV of Total Cost×10^6($)	8917	

Figure 5.2-2 displays the accumulated cost over time broken down to travel time, vehicle operating and safety costs. It can be seen that most of the user cost pertains to travel time, then vehicle operating and safety costs.

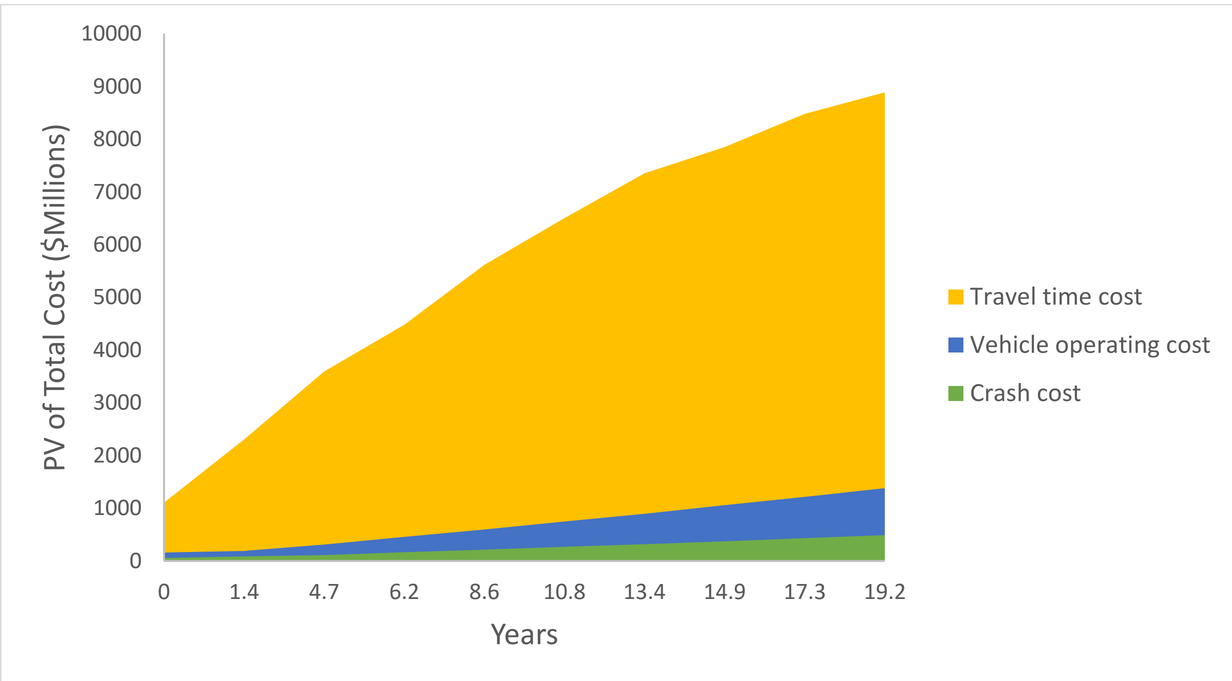


Figure 5.1-1 PV of Travel Time, Vehicle Operating and Safety Costs Over Time

Table 5.1-2 presents the results for GA, greedy-order, and bottleneck-order solutions. These results are in terms of the Present value (PV) of total costs that include: user travel time, vehicle operating, crash, and total cost. The results indicate lower costs for the results obtained by the GA (i.e. 9.03% less than the bottleneck-order and 8.06% less than the greedy-order solution). In fact, it is shown that BCR does not yield the optimal order of projects when project interrelations are considered.

Table 5.1-2 GA, Bottleneck-order and Greedy-order Solution Results

User Travel Time Cost ($)	**Vehicle Operating Cost ($)**	**Crash Cost ($)**	**Total Cost ($)**	**Cost Improvement by GA (%)**
		GA solution		
7,505,448,440	890,475,922	489,829,910	8,917,684,007	-
		Bottleneck order		
8,265,225,305	970,186,444	533,935,197	9,803,467,182	9.03%
		Greedy-order		
8,194,600,060	950,522,818	521,753,124	9,700,467,112	8.06%

Figure 5.1-2 illustrates the evolution of GA process. It indicates how the objective function converges to the optimized value. This optimization process is completed after the genetic search has stopped improving for 10 generations. In this figure, the value of the objective function is depicted for 100 generations.

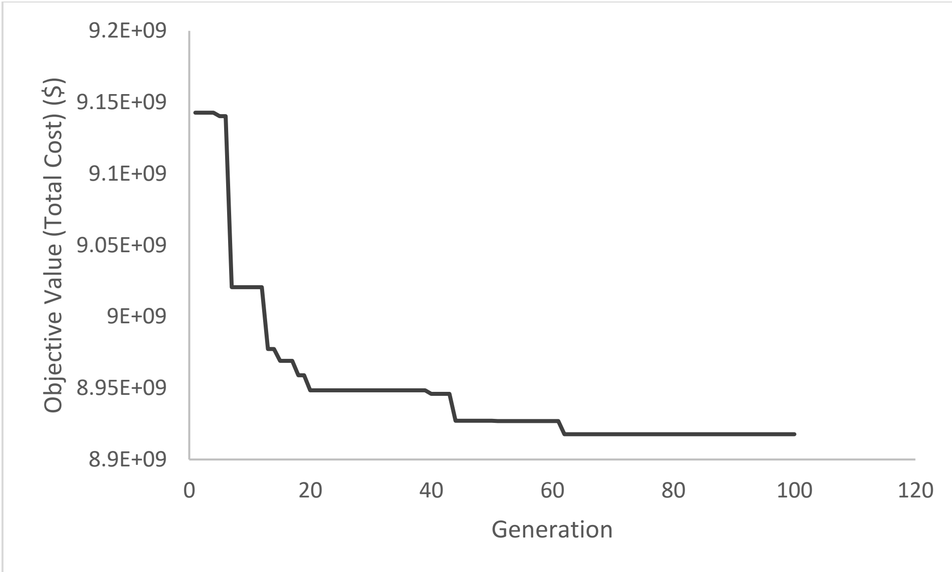


Figure 5.1-2 GA Evolution Process

5.2 Stochastic V.S. Deterministic Solution

A simpler and easier alternative to the stochastic program is to insert the average demand scenario into the deterministic program, which is smaller and can thus be solved in less time. Figure 5.2-1 displays the average demand growth compared to the three scenarios. We can see that the average demand is close to the medium growth rate and roughly between the low and high growth rate scenarios. With this average demand growth rate, we solve a deterministic version of the selection and sequencing problem whose objective is defined in Equation 7. However, with this approach the results are subject to the flaw of averages (Savage and Markowitz, 2009) and, hence, less reliable, as shown in De Neufville and Scholtes (2011).

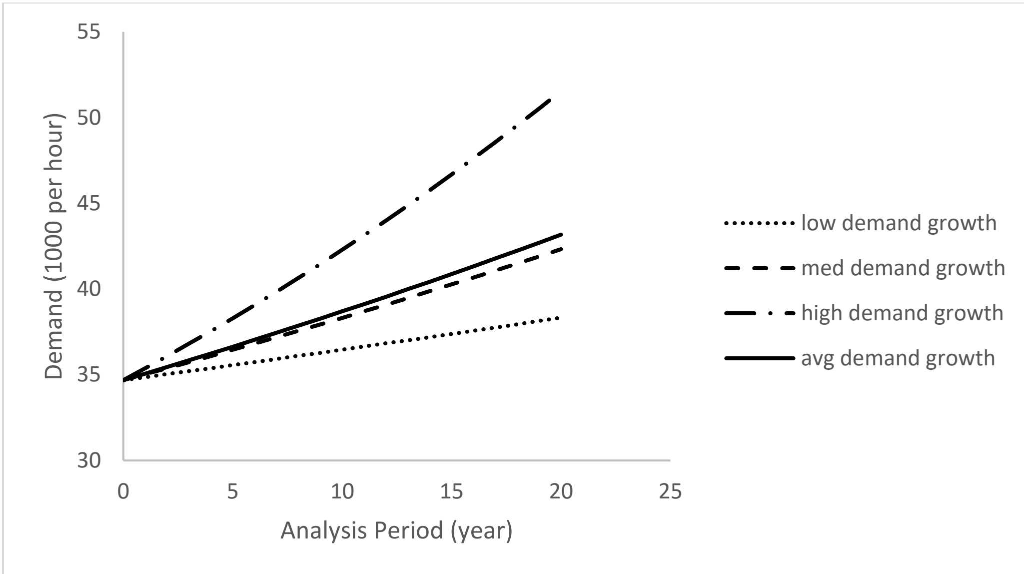


Figure 5.2-1 Demand in the average scenario V.S. demand in each scenario (low, med, high)

In order to compare the deterministic formulation with the stochastic one, the model is applied using the average demand growth rate through the deterministic formulation (Equation 7). Figure 5.2-2 shows the PV of total cost in the average scenario compared to each scenario. We can see that the total cost in the average scenario is skewed to the medium and low scenarios, underestimating the high cost of the high demand scenario. If decisions are made only based on the average scenario, some high costs are expected if the high demand scenario occurs. In fact, the results indicate that the total cost under the average demand growth scenario (solving only a deterministic program) is 7.5% above that using the proposed stochastic program. This shows that solving the stochastic model yields a better solution with a lower objective function i.e. yields a solution with a lower cost. Furthermore, the difference between the objective function value (PV of total cost) of the deterministic and stochastic program, which is called the Value of Stochastic Solution (VSS), is $669 million. This value shows the possible gain from using the stochastic model rather than using the expected value and solving the deterministic model. Failure to consider the full probability distribution instead of the average scenario is also called the “flaw of averages” (Savage and Markowitz, 2009).

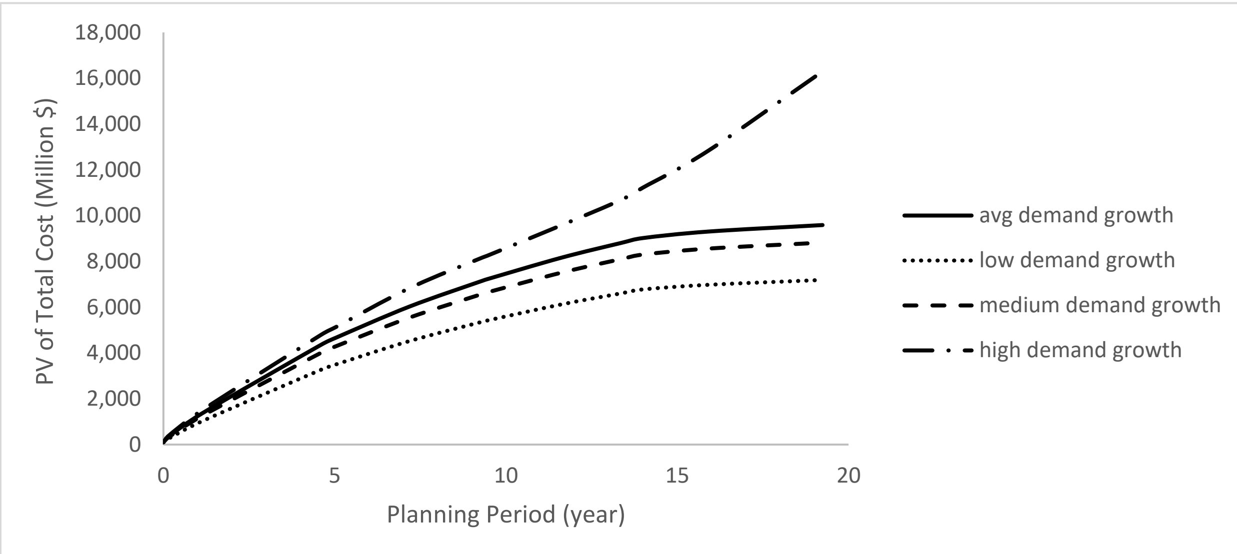


Figure 5.2-2 PV of total cost in the average scenario V.S. each scenario (low, med, high)

5.3 Computation Time

In this case, we tested the algorithm for 20, 40, 80 and 100 candidate projects. Table 5.3-1 compares CPU times for the Anaheim and Sioux Falls networks. It can be seen that a larger network significantly increases the CPU time. The results also indicate that the network size affects the CPU time much more than the number of projects. In this case, where the number of links in the Anaheim network is 12 times higher than Sioux Falls, the CPU time per generation becomes almost 115 times higher. This occurs because the traffic assignment algorithm has to evaluate the entire network regardless of the number of projects. Also, the number of generations for comparable precision is likely to increase with network size. In conclusion, this method is applicable to fairly large networks with numerous projects, but computational improvements would be desirable for analyzing very large networks.

Table 5.3-1 CPU Time per Generation (Sec)

Sioux Falls	Number of projects	5	10	15	20
	CPU time	51.65	91.26	149.25	161.53
Anaheim	Number of projects	20	40	80	100
	CPU time	10,472	12,764	16,897	18,533

5.4 Complete Enumeration Test

To evaluate the results of this algorithm, an exhaustive enumeration is carried out for the Sioux Falls network. Since the enumeration of the original problem with 20 candidate projects (i.e. 20! possible solutions) is lengthy and requires extensive computation time, this test is done for smaller problems with fewer projects. In this case, we consider four problems with 4, 5, 6 or 7 projects to be ranked. Each case is solved both by the GA and by a complete enumeration which evaluates each possible combination of projects and renders the exact solution. The results presented in Table 5.4-1 indicate that the GA yields the exact solution from enumeration in all four cases.

Table 5.4-1 Complete Enumeration Test

	Complete enumeration	**GA solution**

Number of projects	Solution space	Total system cost $* 10^6$	Optimal sequence	Total system cost $* 10^6$	Optimal sequence
4	4!=24	90980	3,2,1,4	90980	3,2,1,4
5	5!=120	94248	3,2,5,4,1	94248	3,2,5,4,1
6	6!=720	98009	3,2,5,4,1,6	98009	3,2,5,4,1,6
7	7!=5040	99301	3,2,5,4,1,6,7	99301	3,2,5,4,1,6,7

5.5 Statistical Test

One major limitation of meta-heuristics is that global optimality is almost never guaranteed, and it is challenging to assess the goodness of solutions obtained by evolutionary methods. However, for large problems where no globally optimal solution can be guaranteed, meta-heuristics can generate satisfactory solutions and their quality can be verified with statistical tests. This can be done by estimating probabilities of finding better solutions. For this purpose, first a sample of random solutions is created, and the objective function value (fitness value) is calculated for each solution. Next, an appropriate distribution function is fitted to the fitness values. Then, the cumulative probability of the solution obtained from the algorithm is computed based on the fitted distribution. In this case, it is desirable to obtain a very low probability to demonstrate the goodness of the solution.

For this test a random sample consisting of 100,000 solutions is created. After testing different distribution functions, the Lognormal (mu=22.997, sigma= 0.0238) distribution is found to best fit the sample. Figure 5.5-1 shows the fitted distribution and the data from random sampling. From this figure, it is evident that the GA solution (8917 $\times 10^6$ from Table 5.1-1) is located at the far left side of the diagram meaning that the GA solution has a lower objective function value than almost the entire sample. In other words, the solution found by the algorithm has a lower cost than any of the 100,000 random solutions in the distribution.

The next step is to calculate the cumulative probability of the best solution found by the GA (8917 $\times 10^6$ from Table 5.1-1) according to the Lognormal distribution: $p = F(x|\, \mu, \sigma) = F(8917 \times 10^6|22.997, 0.0238) = 1.568 \times 10^{-4}$. This can be derived from the following equation:

$$p = F(x|\, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^x \frac{e^{\frac{-(\ln(t)-\mu)^2}{2\sigma^2}}}{t}\, dt \tag{22}$$

This result implies that the probability of finding a solution better than the GA solution is vanishingly small, i.e. 2.834×10^{-5} . In other words, the GA solution dominates 99.999% of the random solutions in the distribution. Therefore, the solution found by the GA, although not guaranteed to be globally optimal, is very good compared to other possible alternatives in the solution space and the likelihood that significantly better solutions exist is negligible. Moreover, errors from imperfect optimization (i.e. deviations from global optimality) are likely to be greatly dominated by uncertainties in input parameters.

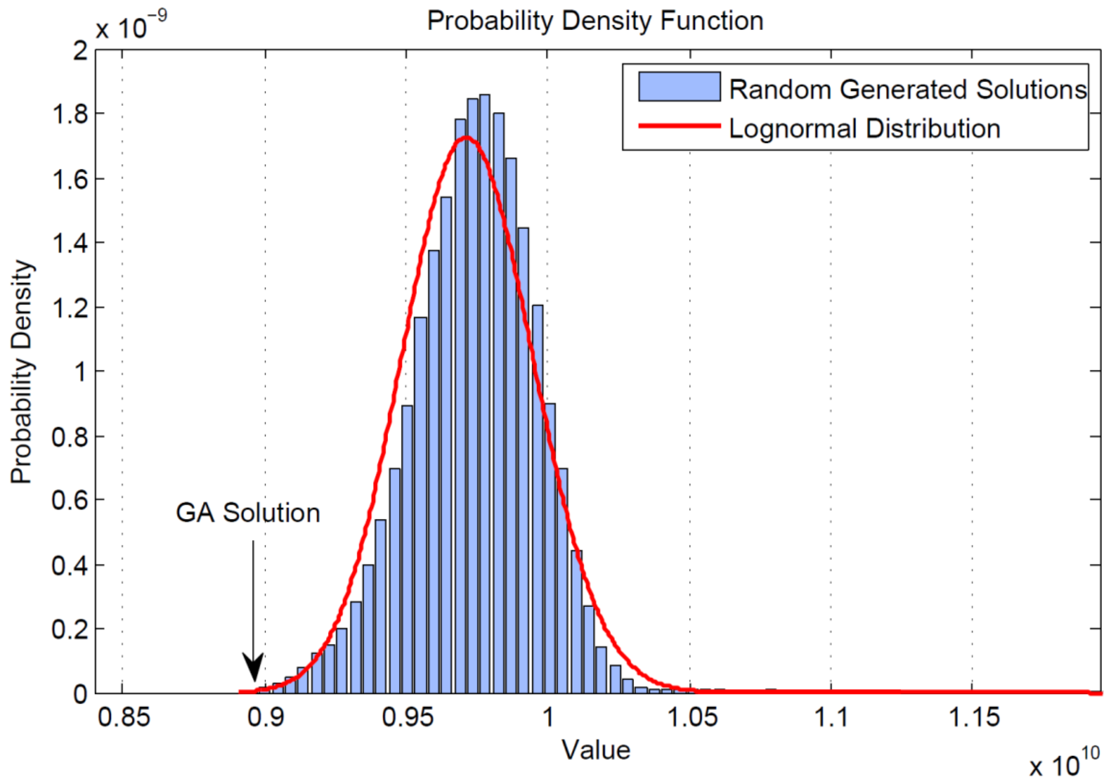


Figure 5.5-1 Fitted Lognormal Distribution

Chapter 6 Cell Transmission Model (CTM)

6.1 Background

So far, a traffic assignment model has been used to evaluate the objective function. In other words, a traffic assignment model is employed to obtain w_{ij} (travel time), VMT_{ij}(vehicle miles travelled), $N_{cr(j)}$(number of crashes), and other parameters to feed in to equations 7 and 8. As mentioned earlier, this study aims to develop and use a more detailed evaluation model specifically the Cell Transmission Model introduced by Daganzo (1994). The CTM discretizes the LWR model (or its simplified version) in both time and space, which is shown to be computationally efficient and easy to analyze yet capture many important traffic phenomena, such as queue build-up and dissipation, and backward propagation of congestion waves.

The CTM uses an algorithm that is consistent with the kinematic wave theory of traffic flow. This method assumes that the road has been divided into homogeneous sections (cells), *i*, whose lengths equal the distance traveled by free-flowing traffic in one clock interval. Under light traffic then, all the vehicles in a cell can be assumed to advance to the next with each tick of the clock; it is unnecessary to know where within the cell they are located. The state of the system at instant *t* is given by the number of vehicles contained in each cell, $n_i(t)$. To incorporate queuing, the following parameters are introduced for each cell:

$N_i(t)$: The maximum number of vehicles that can be present in cell *i* at time *t*

$Q_i(t)$: The maximum number of vehicles that can flow into cell *i* when the clock advances from *t* to *t + 1* (time interval *t*).

The first constant is the product of the cell's length and its "jam density"(k_j), and the second one is the minimum of the "capacity flows" (q_{max})of cells *i - 1* and *i*. As such, this method keeps track of the overall traffic state over time. The model is also able to keep track of the location of moving queues in the network, predicting queue spillbacks and dissipation in a reasonable way.

6.2 Similarity to a Hydrodynamic Model

The Lighthill, Whitham, Richards (LWR) hydrodynamic model with a density-flow (k-q) relationship is depicted in Figure 6.2-1 and can be expressed as:

$$q = min\{v_f k, q_{max}, \omega_c(k_j - k)\}, \qquad \text{for } 0 \leq k \leq k_j \qquad (23)$$

where v_f is the free-flow speed, k_j is the jam density and q_{max} is the maximum flow, and ω_c is the backward wave speed. Daganzo (1994) shows that the LWR equations for a single highway link can be approximated by a set of difference equations where current conditions (the state of the system) are updated with the tick of a clock.

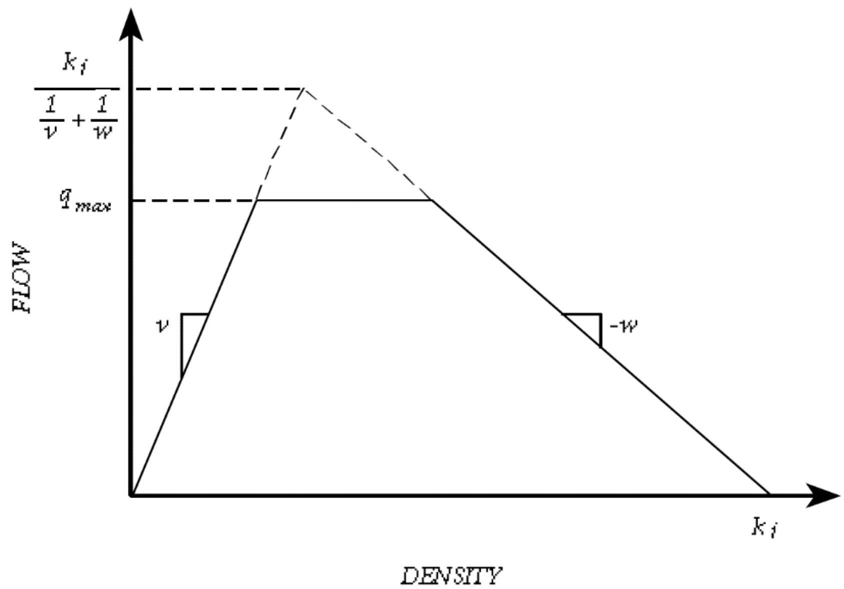


Figure 6.2-1 Flow-Density Relationship for CTM

The number of vehicles that can flow from cell *i -1* to cell *i* when the clock advances from *t* to *t+1*, $y_i(t)$, is assumed to be the smallest of three quantities:

$n_{i-1}(t)$: the number of vehicles in cell *i-1* at time *t*,

$Q_i(t)$: the capacity flow into cell *i* for time interval *t*,

$N_i(t)$- $n_i(t)$: the amount of empty space in cell *i* at time *t*.

This last quantity ensures that the vehicular density on every section of the road remains below jam density. In other words, if cells are numbered consecutively starting with the upstream end of the road from *i* = 1 to I, the inflow to cell *i* in the time interval (t, t+1) can be expressed as:

$$y_i(t) = min\{n_{i-1}(t)\ ,\ Q_i(t),\ \omega/v[\ N_i(t) -\ n_i(t)]\} \quad (24)$$

where the cell occupancy at time $t + 1$ equals its occupancy at time t, plus the inflow and minus the outflow:

$$n_i(t+1) = n_i(t) + y_i(t) - y_{i+1}(t) \tag{25}$$

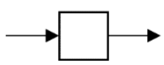
Note that these equations are a discrete approximation to the LWR hydrodynamic model explained above.

6.3 Network Representation

CELL

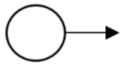
A cell is the smallest component of CTM. There are three types of cells:

1. Normal cell



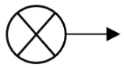
Vehicles can move into a normal cell stay there or move out of it from time period t to $t+1$. The characteristics of a normal cell include: capacity, the number of contained vehicles and the maximum flow rate.

2. Input cell



An input cell or "source" feeds vehicles in to the system and has an infinite number of vehicles ($n_i = \infty$). This cell may only be an abstract representation. The setting for this type of cell includes the number of discharged vehicles in to the system.

3. Output cell



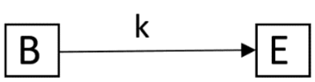
The output cell or "sink" cell that absorbs the traffic from the system should have infinite size ($N_i = \infty$). This cell could be abstract and not physically present in the network. The setting of this cell includes the number of vehicles that leave the system into this cell.

LINK

The links represent how cells are connected in the network. There are three types of links:

1. Ordinary link

The direct link is the simplest form of link in the CTM which connects two cells. In any interval, the flow over a direct link is the minimum of the output from upstream cell and the input from the downstream cell.



For an ordinary link the equivalent of equation 24 is:

$$y_k(t) = min\{n_B\ , \min[\,Q_B, Q_E],\ \omega/v[\,N_E - \ n_E]\} \qquad (26)$$

Where, y_k(t) is flow on link k from clock tick t to clock tick t+1. For simplicity, the time variable "t" is omitted from the right side of the above and in forthcoming expressions. It should be noted that any time-dependent quantities should be valued at "t", unless explicitly stated otherwise. The cell occupancy at time t+1 ($n_k(t+1)$ can be derived from equations 25.

A further simplification is desirable by the following equations:

$S_i(t) = \min[Q_i, n_i]$ (27)

$R_i(t) = \min[Q_i,\ \omega/v[\ N_i -\ n_i]]$ (28)

Where $S_i(t)$ is the maximum flows that can be sent and $R_i(t)$ is the maximum flows that can be received by cell *i* in the interval between *t* and *t + 1*,

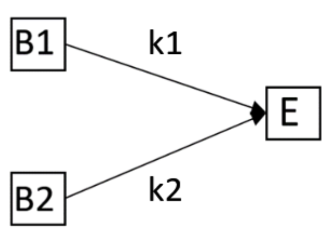
then we can write $y_i(t)$ in the more compact form:

$y_i(t) = min\{S_B >\ R_E\}$ (29)

That is, the flow on link "k" should be the maximum that can be sent by its upstream cell unless prevented to do so by its end cell.

We now explain the extensions of equation (25) for merge and diverge links.

2. Merge link



The merge link is provided when traffic from two cells enter the same cell. In this case, the volume is determined by the outputs of two upstream cells, the input of the downstream cell, and the proportion (spill back) of traffic between links. In this case the flow satisfies:

$y_{k1}(t) \leq S_{B1}$; $y_{k2}(t) \leq S_{B2}$ (30a)

$y_{k1}(t) + y_{k2}(t) \leq R_E$ (30b)

It is assumed that cells B1 and B2 send the maximum possible flow if cell E can receive it, therefore:

$y_{k1}(t) = S_{B1}$; $y_{k2}(t) = S_{B2}$, $\quad if \quad S_{B1} + S_{B2} \leq R_E$ (31)

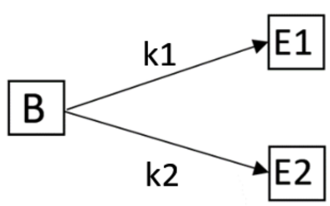
If the condition in (30) is not satisfied we will assume that a fraction (p1) of the vehicles come from B and the remainder (p2) from B2, where p1 + p2 = 1. Thus, we can write:

$y_{k1}(t) = mid\{S_{B1}, R_E - S_{B2}, p1 * R_E\}$ (32a)

$y_{k2}(t) = mid\{S_{B2}, R_E - S_{B1}, p2 * R_E\}$ (32b)

$if \quad S_{B1} + S_{B2} > R_E$ (32c)

3. Diverge link



The diverge link describes the case where vehicles from one cell enter two different cells. In this case, the volume is determined by the output of the upstream cell, the inputs of the downstream cells and the proportion of traffic between links. We assume that the proportions of S_B(t) going to cells E1 and E2 are q1 and q2 (q1+q2=1) which are exogenously determined, and that traffic flows in these proportions continuously between clock ticks. Therefore:

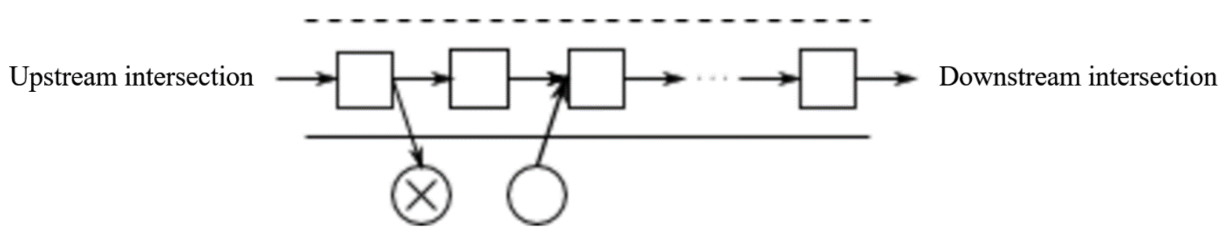
$y_B(t) = \min\{S_B, R_{E1}/q1, R_{E1}/q1\}$ (33a)

$y_{K1}(t) = q1 * y_B(t);\ y_{K2}(t) = q2 * y_B(t)$ (33b)

As it does for ordinary and merge links, equation (25) complete the set of equations needed to update the state of the system.

LANE

A lane represents a section of the traffic area and includes multiple cells and links. A lane has input flow, output flow, and it connects two intersections. The traffic area of a lane will be separated into several cells which are automatically connected after the parameters of the lane are given to the model. These parameters include length, number of lanes, free flow speed, maximum capacity, etc. Furthermore, a normal lane also includes an input cell and an output cell to represent its input and output flow.



INTERSECTION

The traffic intersection is the area where different traffic flows meet and conflict. A typical intersection consists of the input lanes, the output lanes and the conflict area. To describe the flows within the intersection, the conflict area should be separated into several cells. Then, different set of merges and diverges links are introduced to create different phases. An example of a two-phase intersection is displayed below. It is shown how links are set for East-West and North-South phases. There are 6 inner cells within an intersection and 6 links corresponding to each phase.

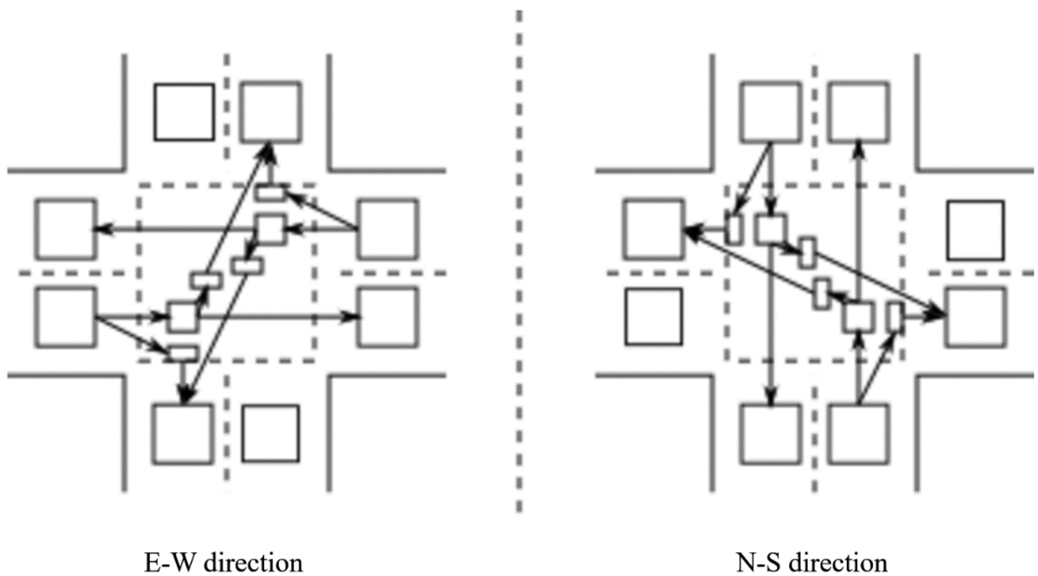


Figure 6.3-1 Two-Phase Intersection Cells and Links

According to Roess et.al. (2011), if it is assumed that the demands on intersections are known and the “critical lanes” can be identified, then the following equation could be solved to find a minimum acceptable cycle length:

$$C_{min} = \frac{N * t_L}{1 - (\frac{V_c}{3600/h})} \tag{34}$$

Where:

C_{min} = minimum cycle length (s)

N=number of phases

t_L=total loss time per phase (4 s/phase in this study)

V_c=sum of critical lane volumes (veh/h)

h=saturation headway (s/veh)

The "critical lane" concept involves the identification of specific lane movements that will control the timing of a given phase. During a simple two-phase signal, all E-W movements are permitted in one phase, and all N-S movements are permitted in another phase.

EXAMPLE

An example of CTM application for a small network is presented below. This network consists of four intersections and 24 lanes. A similar configuration will be used later on for the Sioux Falls network.

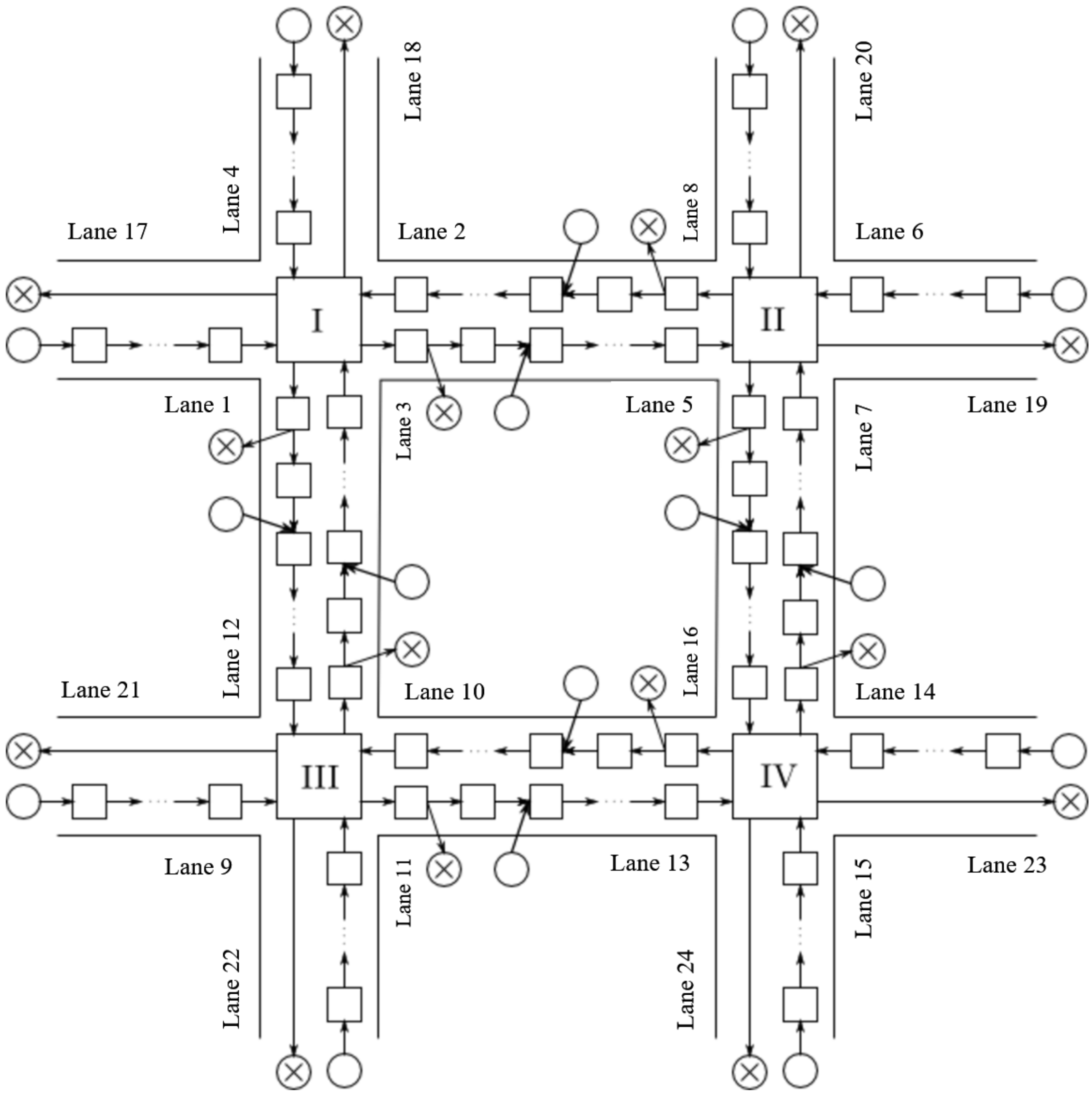


Figure 6.3-2 Example of Cell Transmission Application on a small network

Chapter 7 CTM Application and Results

In this chapter the goal is to replace the previous evaluation model (Frank-Wolf traffic assignment) with a more detailed model (CTM) and solve the similar problem of optimizing the selection and schedule of projects. The model is applied to the same Sioux Falls network with some alterations which were essential to apply CTM. These changes are described in more detail in the following section.

7.1 Initial Network Set-up

In this section the Cell Transmission model described in the previous chapter is applied to the Sioux Falls network. In this network each node presents a two-phased intersection. In order to be compatible with CTM, some characteristics are changed in this network compared to the one used in Chapter 5. First, the O/D demand table is modified in order to have volume/capacity ratio < 1 for all links. Unlike the Frank-Wolf algorithm, CTM does not allow a volume that exceeds capacity. The modified demand table is presented in Table 7.1-1. Second, the numbering of lanes is changed based on CTM order rules. It starts consecutively from intersection 1, numbering (i) east, (ii) west, (iii) north and (iv) south inflow lanes and moving to the next intersection. The new network with modified numbering is shown in Figure 7.1-1.

Table 7.1-1 Modified Demand Table

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*
1	0	18	18	16	8	11	7	22	8	9	16	8	16	8	27	16	7	5	8	8	3	11	8	5
2	18	0	2	6	2	11	2	11	3	3	4	4	6	2	1	9	3	2	2	4	2	4	2	2

3	18	2	0	6	2	6	1	4	2	2	6	6	4	2	1	4	1	0	2	2	2	2	2	2
4	16	6	6	0	11	11	5	15	9	6	32	15	13	11	3	17	5	2	6	9	4	9	11	6
5	8	2	2	11	0	6	2	13	9	5	13	4	4	4	2	13	3	2	4	4	2	4	4	2
6	11	11	6	11	6	0	4	17	4	4	9	6	6	4	2	22	6	2	6	9	2	6	4	2
7	7	2	1	5	2	4	0	12	3	5	5	9	5	3	3	15	5	11	5	6	3	6	2	1
8	22	11	4	15	13	17	12	0	18	9	19	13	13	9	4	48	15	6	15	19	9	13	9	4
9	8	3	2	9	9	4	3	18	0	8	16	8	6	6	5	16	5	2	5	8	4	8	6	2
10	9	3	2	6	5	4	5	9	8	0	22	11	10	12	22	24	11	4	10	14	7	15	10	5
11	16	4	6	32	13	9	5	19	16	22	0	32	22	35	8	30	11	4	11	15	11	24	30	13
12	8	4	6	15	4	6	9	13	8	11	32	0	30	15	4	15	8	4	6	11	9	17	15	11
13	16	6	4	13	4	6	5	13	6	10	22	30	0	13	4	15	6	2	9	15	13	28	17	17
14	8	2	2	11	4	4	3	9	6	12	35	15	13	0	4	15	8	2	9	11	9	26	24	9
15	27	1	1	3	2	2	3	4	5	22	8	4	4	4	0	7	4	2	4	6	4	14	5	3
16	16	9	4	17	13	22	15	48	16	24	30	15	15	15	7	0	30	18	30	37	13	26	13	6
17	7	3	1	5	3	6	5	15	5	11	11	8	6	8	8	30	0	8	18	18	8	18	6	3
18	5	2	0	2	2	2	11	6	2	4	4	4	2	2	2	18	8	0	9	18	2	9	2	2
19	8	2	2	6	4	6	5	15	5	10	11	6	9	9	4	30	18	9	0	28	11	28	9	4
20	8	4	2	9	4	9	6	19	8	14	15	11	15	11	6	37	18	18	28	0	28	54	15	11
21	3	2	2	4	2	2	3	9	4	7	11	9	13	9	4	13	8	2	11	28	0	41	15	13

22	11	4	2	9	4	6	6	13	8	15	24	17	28	26	14	26	18	9	28	54	41	0	48	26
23	8	2	2	11	4	4	2	9	6	10	30	15	17	24	5	13	6	2	9	15	15	48	0	17
24	5	2	2	6	2	2	1	4	2	5	13	11	17	9	3	6	3	2	4	11	13	26	17	0

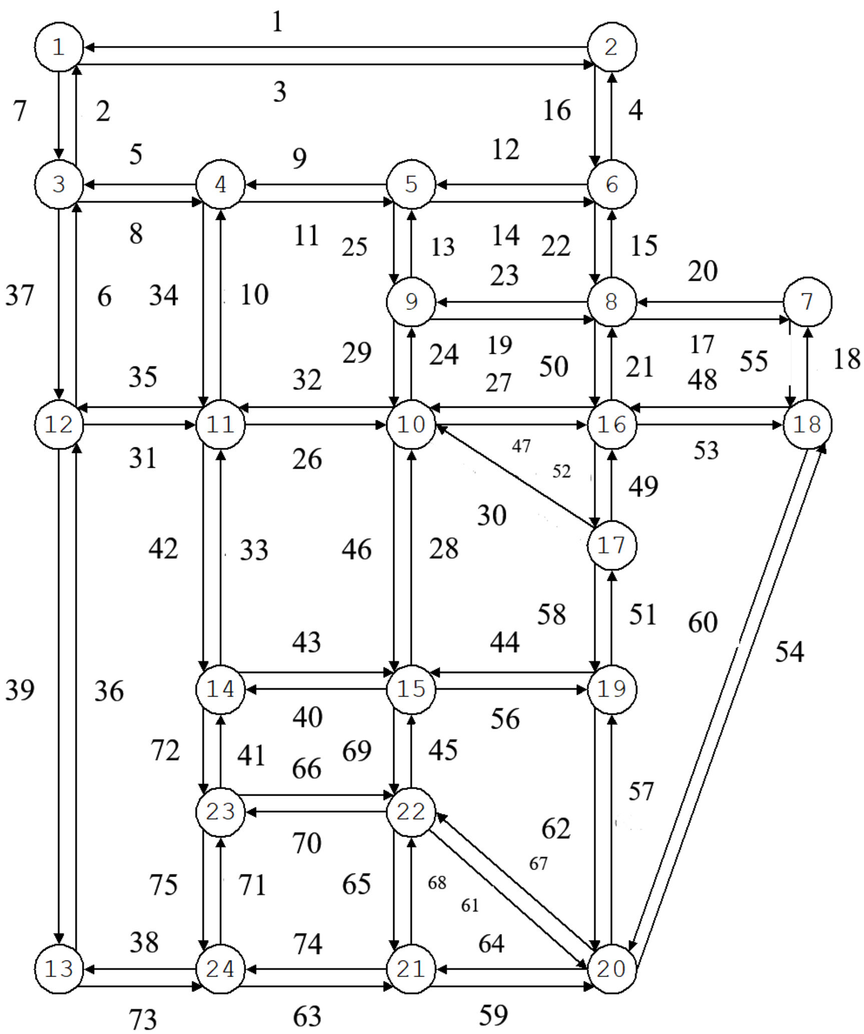


Figure 7.1-1 Modified Sioux Falls Network

Initial link attributes including flow and density are provided in Table 7.1-2 :

Table 7.1-2 Initial Link Flows and Densities

Link #	Flow (veh/h)	Density (veh/mi.ln)	Link #	Flow (veh/h)	Density (veh/mi.ln)	Link #	Flow (veh/h)	Density (veh/mi.ln)

1	1307	16.34044	**26**	1536	173.2249	**51**	887	205.673
2	1140	14.24884	**27**	3155	92.25792	**52**	1338	183.1067
3	1395	17.44329	**28**	3785	60.7459	**53**	2097	26.21804
4	1030	12.87447	**29**	776	211.2222	**54**	3705	64.77415
5	3614	45.16903	**30**	56	0.70282	**55**	3356	82.1842
6	3535	73.2683	**31**	1878	23.47339	**56**	1105	194.7354
7	3028	37.84825	**32**	2945	102.7371	**57**	1786	22.32115
8	3898	55.11182	**33**	1527	173.6687	**58**	1158	192.0866
9	1229	188.5685	**34**	376	4.696037	**59**	1247	15.58592
10	2771	34.63521	**35**	2818	35.22547	**60**	3833	58.36024
11	1944	152.7968	**36**	1970	151.5151	**61**	1753	21.91405
12	3067	38.33588	**37**	3650	67.49629	**62**	949	11.86159
13	2435	30.43786	**38**	2389	29.86544	**63**	1216	189.2008
14	3784	60.81826	**39**	1744	21.80504	**64**	2236	27.95485
15	846	207.7248	**40**	1488	18.60571	**65**	778	211.0888
16	3312	84.40222	**41**	2794	110.315	**66**	3430	78.4854
17	578	7.230776	**42**	3531	44.13872	**67**	1166	14.57373
18	2897	36.2155	**43**	3727	46.59138	**68**	823	208.8364
19	133	1.666874	**44**	1985	150.7569	**69**	1098	195.0758
20	912	204.3908	**45**	1502	174.8948	**70**	1078	13.47504
21	1943	152.8443	**46**	1055	13.18984	**71**	3942	49.28057
22	724	213.7881	**47**	2429	128.5733	**72**	2714	33.92908
23	171	2.132808	**48**	2694	115.296	**73**	1517	174.1411
24	843	207.8495	**49**	878	206.0913	**74**	1447	177.6746
25	3370	42.12574	**50**	2798	110.0838	**75**	3912	54.37932

7.2 CTM Set-up

The CTM described in the previous chapter is applied to the above network with 75 links and 20 two-phased intersections. All intersections are assumed to have two phases in E-W and N-S directions. It is also assumed that the nominal flow–density relationships of all cells are characterized by triangular fundamental diagrams shown in Figure 7.2-1.

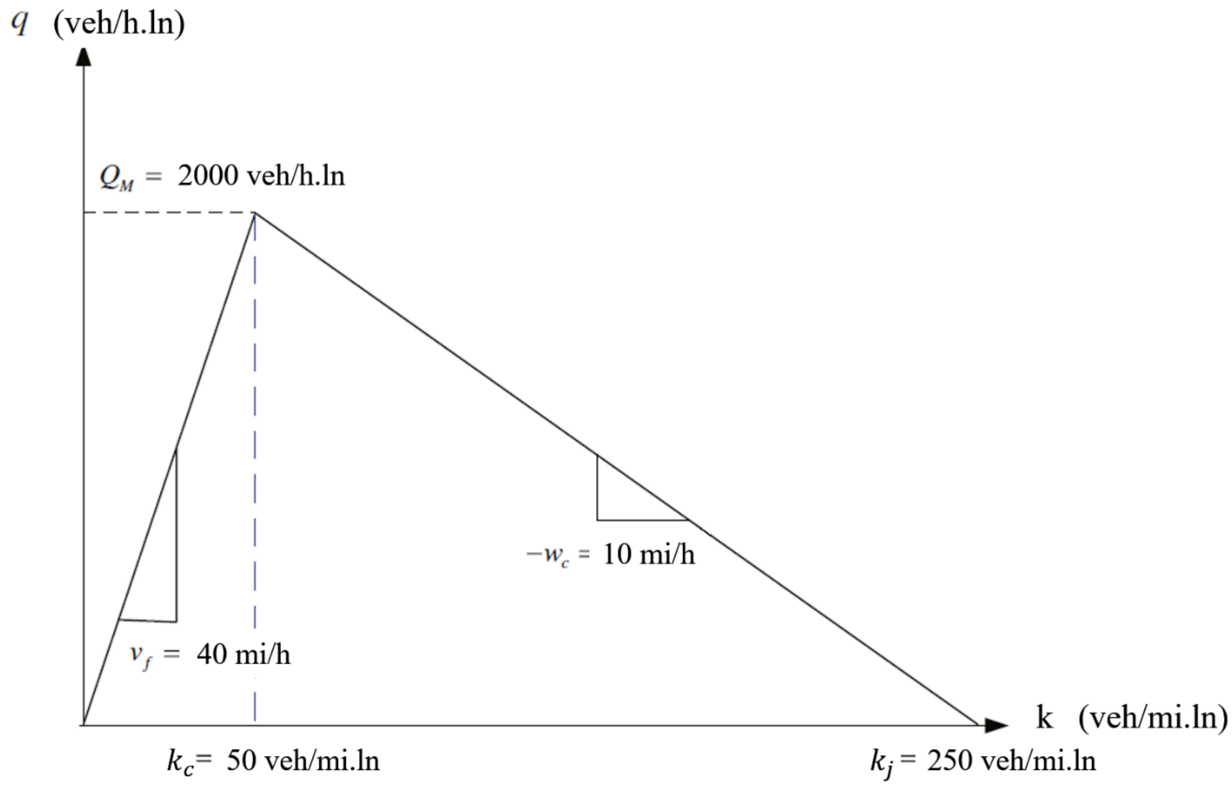


Figure 7.2-1 Fundamental Density-Flow Diagram for Each Cell

Table 7.2-1 indicates the main input parameters of CTM and their respective values.

Table 7.2-1 CTM Input Parameters

Parameter	**Value**
Free flow speed ($\boldsymbol{v_f}$)	40 (mi/h)
Spill back speed ($\boldsymbol{w_c}$)	10 (mi/h)
Jam density ($\boldsymbol{k_j}$)	250 (veh/ln.mi)
Maximum flow (($\boldsymbol{Q_m}$)	2000 (veh/h.ln)
Cell length ($\boldsymbol{L_c}$)	290 (ft)
Vehicle length ($\boldsymbol{L_v}$)	16 (ft)
Simulation interval ($\boldsymbol{dt}$)	5 (s)

The following provides a step-by-step summary description of CTM:

1. **Set inputs:** Define input parameters and specify their values. Input parameters include free flow speed, maximum flow, spill back speed, jam density, etc. as described in Table 7.2-1.
2. **Add links:** Add each link based on specified attributes such as length, saturation flow, and number of lanes as provided in Table 7.1-2. The model then divides the links to multiple cells and links.
3. **Add intersections and phases:** Add intersections and determine the number of phases.
4. **Set up initial link volumes:** Initial link volumes is determined using the traffic assignment model based on the demand table.
5. **Start CTM:** The simulation begins calculating the inflow to all cells from clock tick *t* to clock tick *t+1* (i.e. during time interval dt) based on equations 29 to 33. Then the cell occupancies are derived from equation 25 to update the state of the system. It should be noted that q1 and q2 from equation 33 determines the turning flows at each intersection. These parameters are considered exogenous and in this case are determined by traffic assignment.
6. **Obtain delays and densities:** At each time interval the model outputs the cell densities and delays at each intersection.

7.3 Identifying Link and Intersection Improvement Projects

In order to identify the most congested links and set up the initial set of candidate projects, we run the CTM for 40 intersection cycles (equal to approximately one hour of simulation time) on the unimproved network. The cell occupancy can be tracked at each step of the simulation which can yield the vehicle density for each link. The figures below show the density changes for all links during the analysis period.

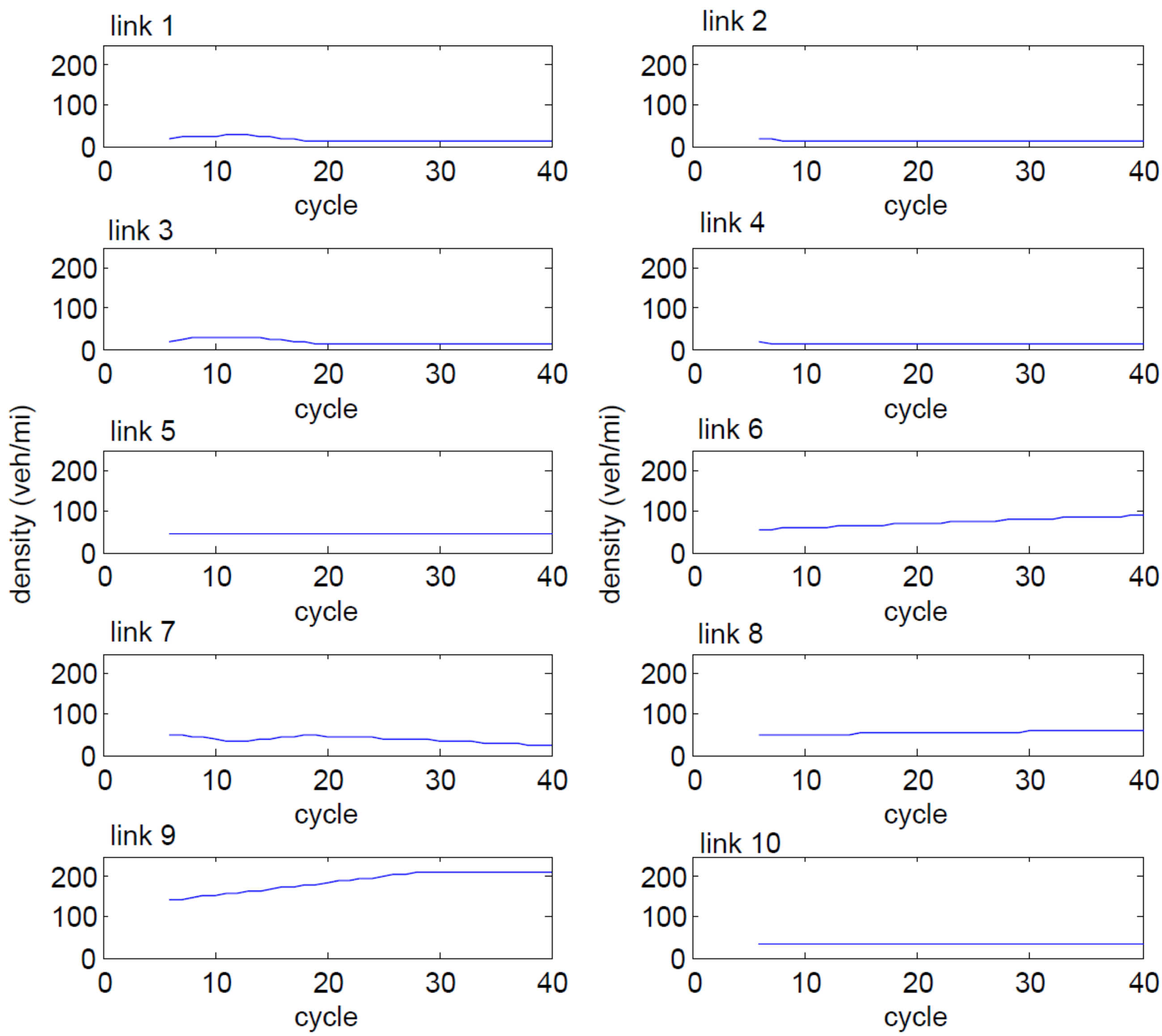


Figure 7.3-1 Initial Link Densities (Links 1-10)

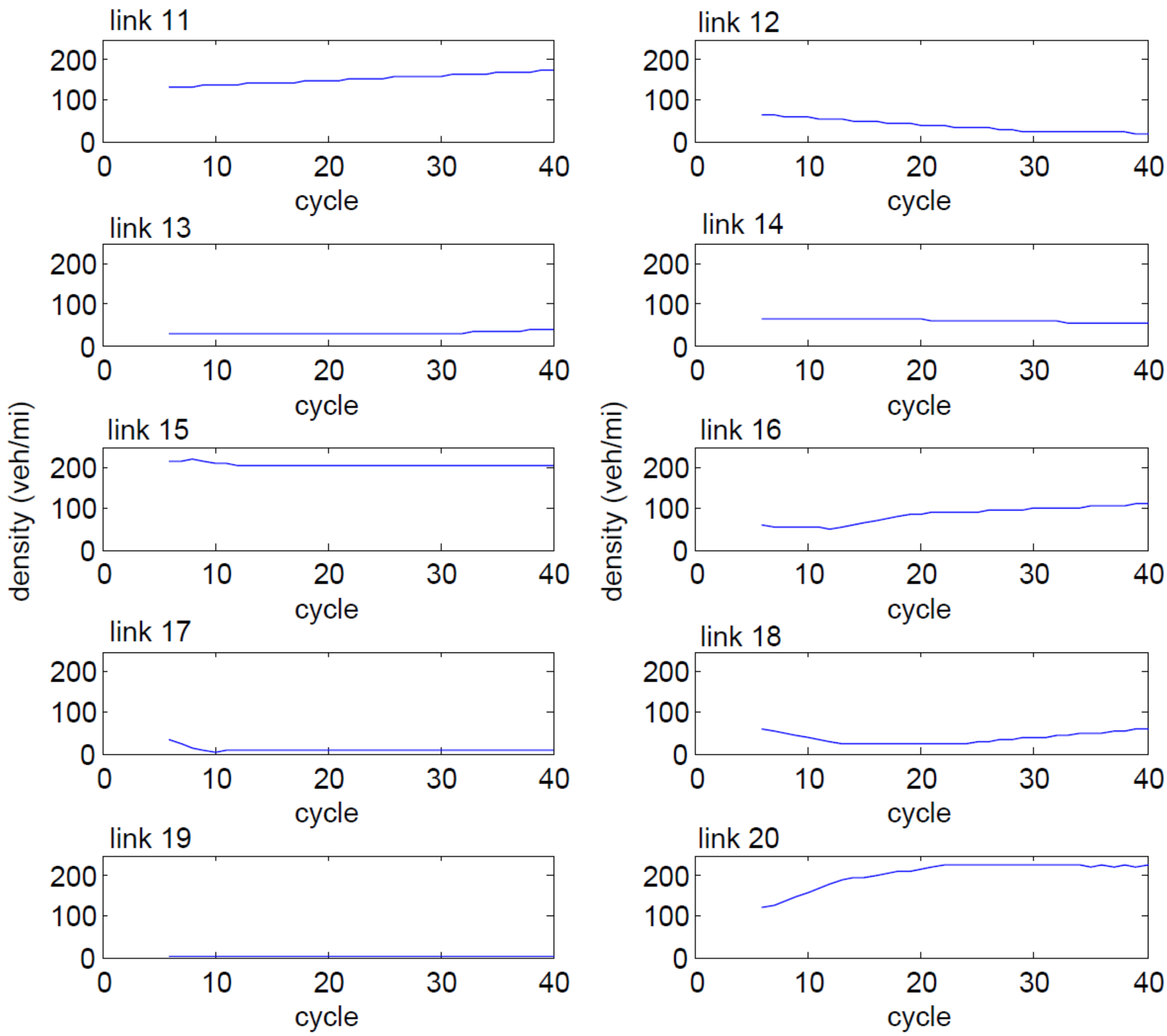


Figure 7.3-2 Initial Link Densities (Links 11-20)

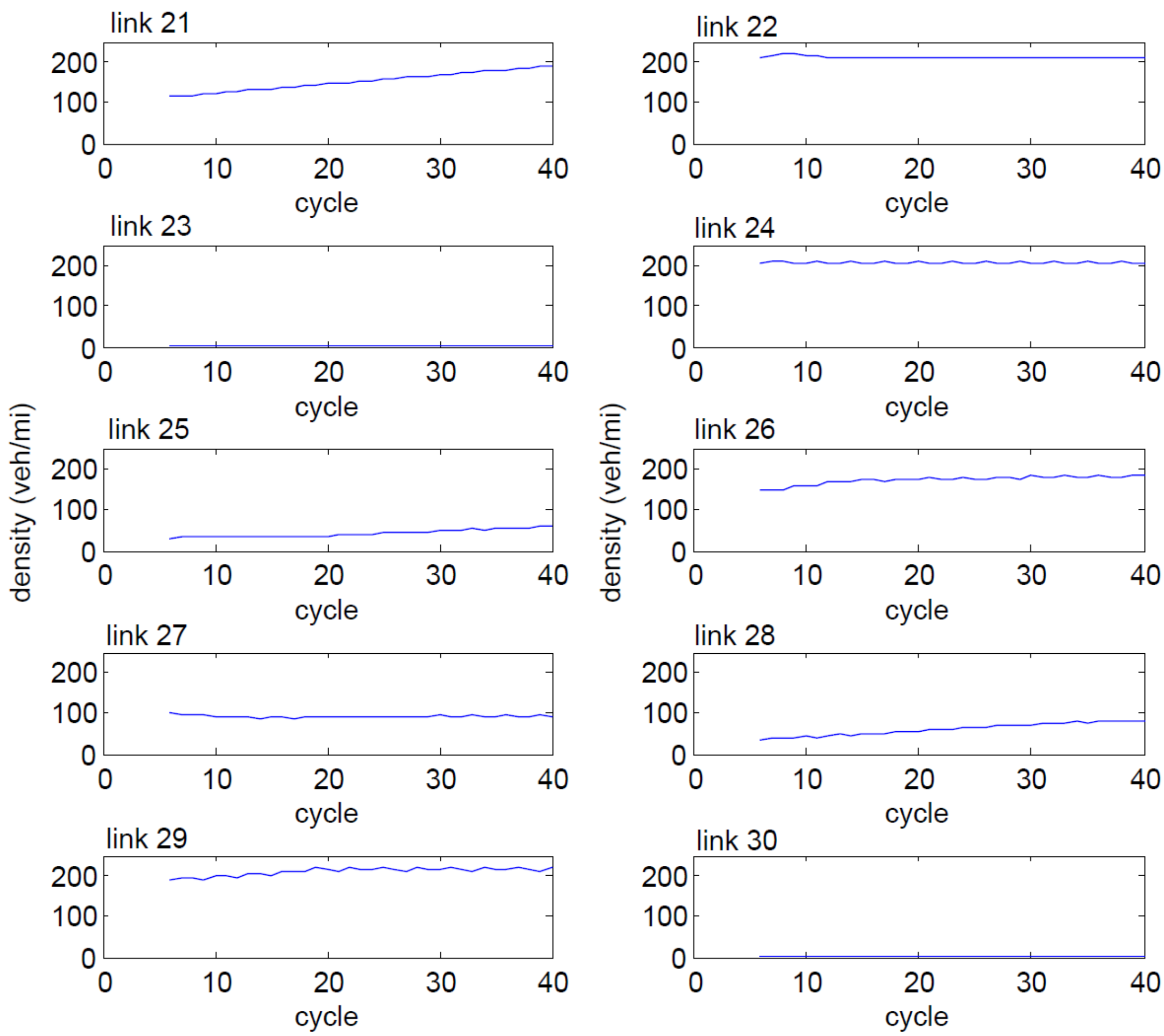


Figure 7.3-3 Initial Link Densities (Links 21-30)

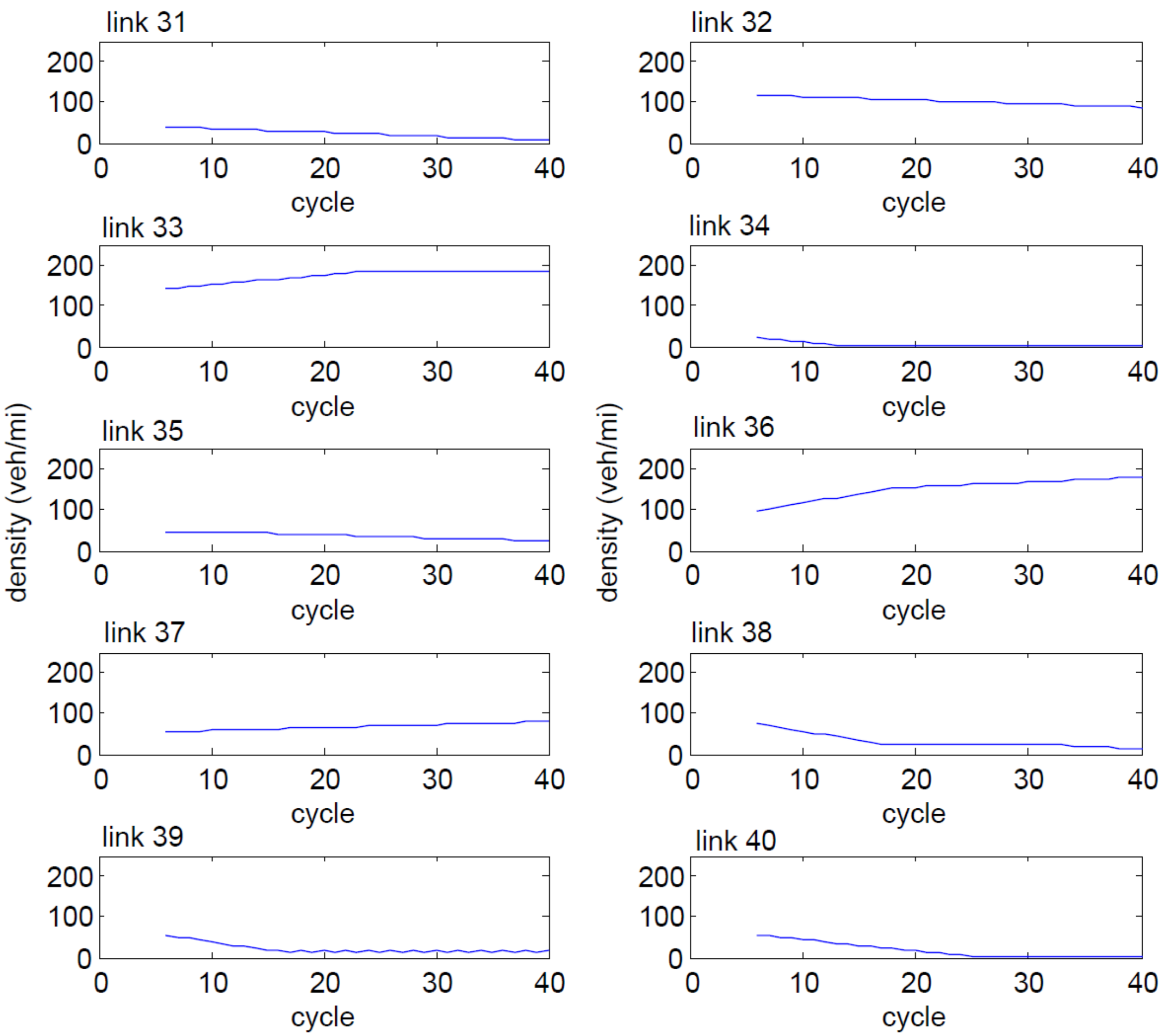


Figure 7.3-4 Initial Link Densities (Links 31-40)

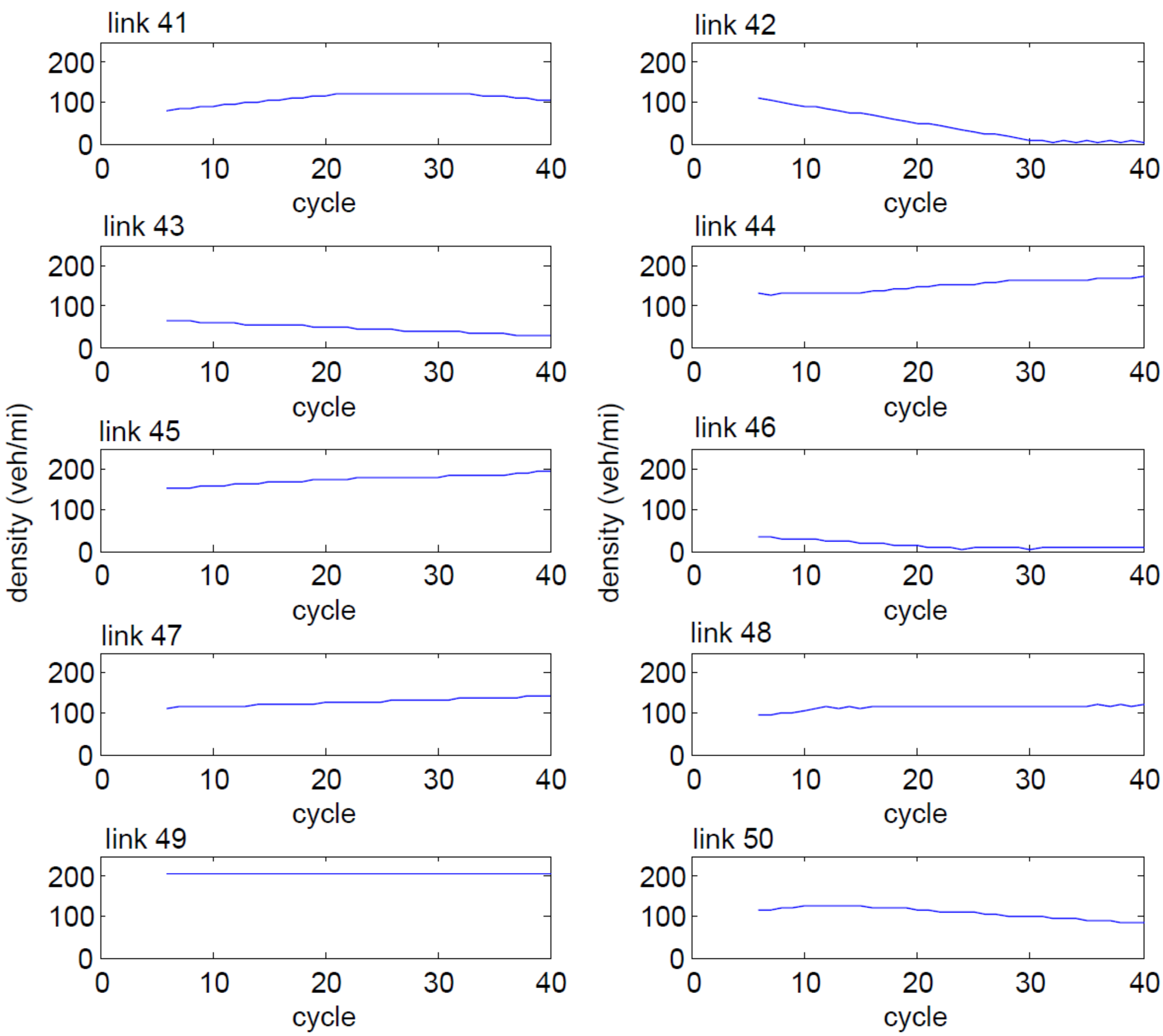


Figure 7.3-5 Initial Link Densities (Links 41-50)

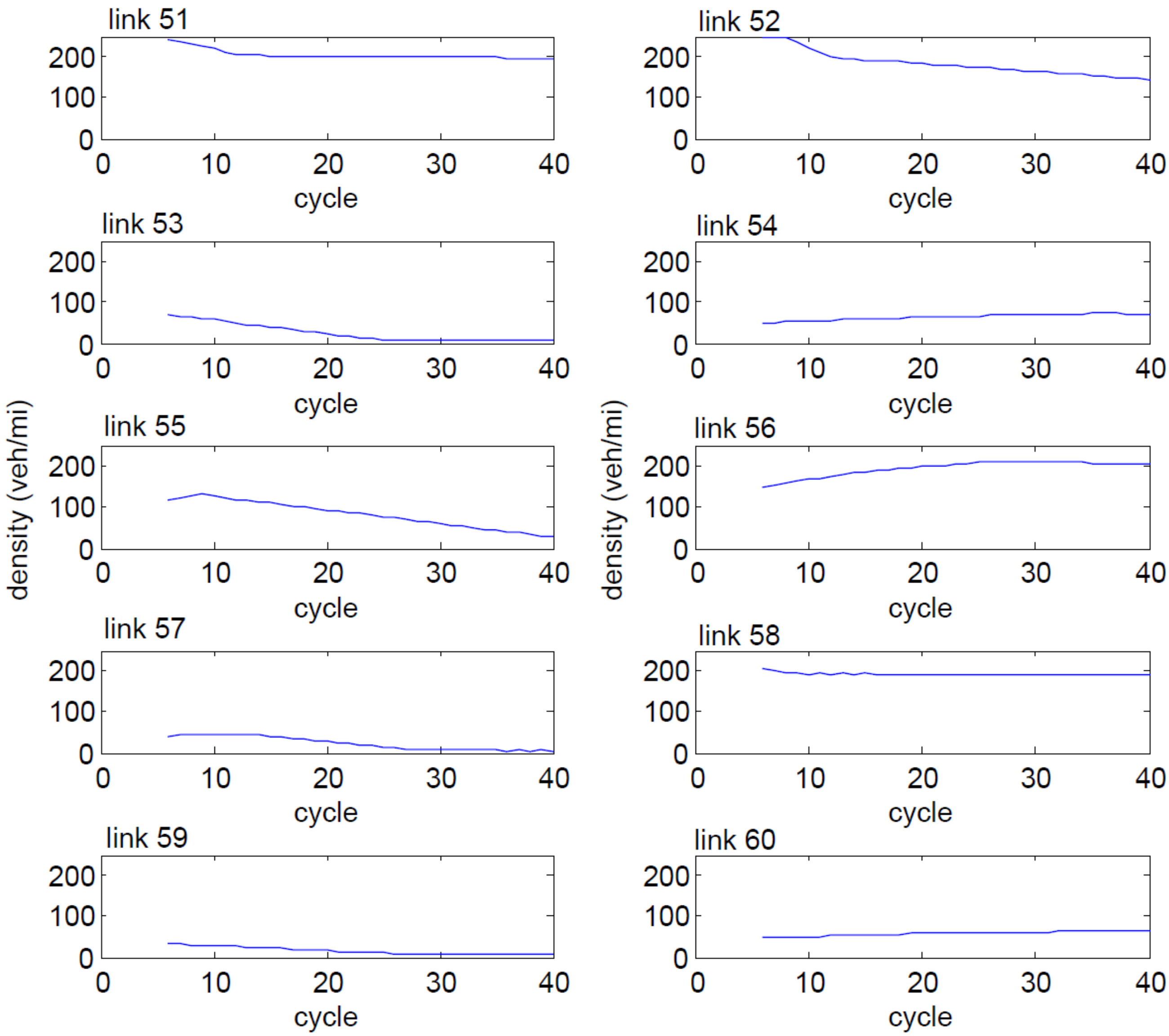


Figure 7.3-6 Initial Link Densities (Links 51-60)

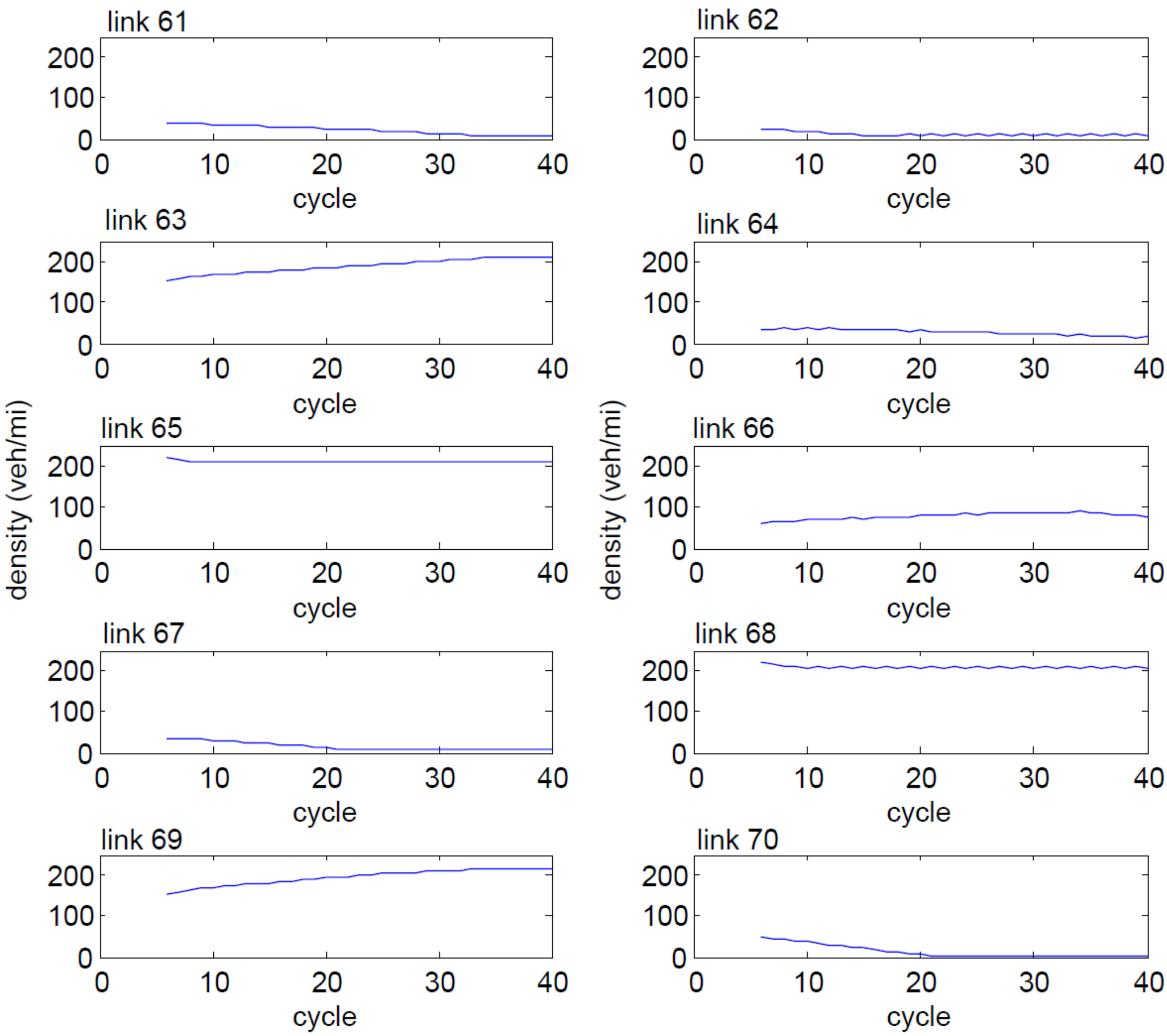


Figure 7.3-7 Initial Link Densities (Links 61-70)

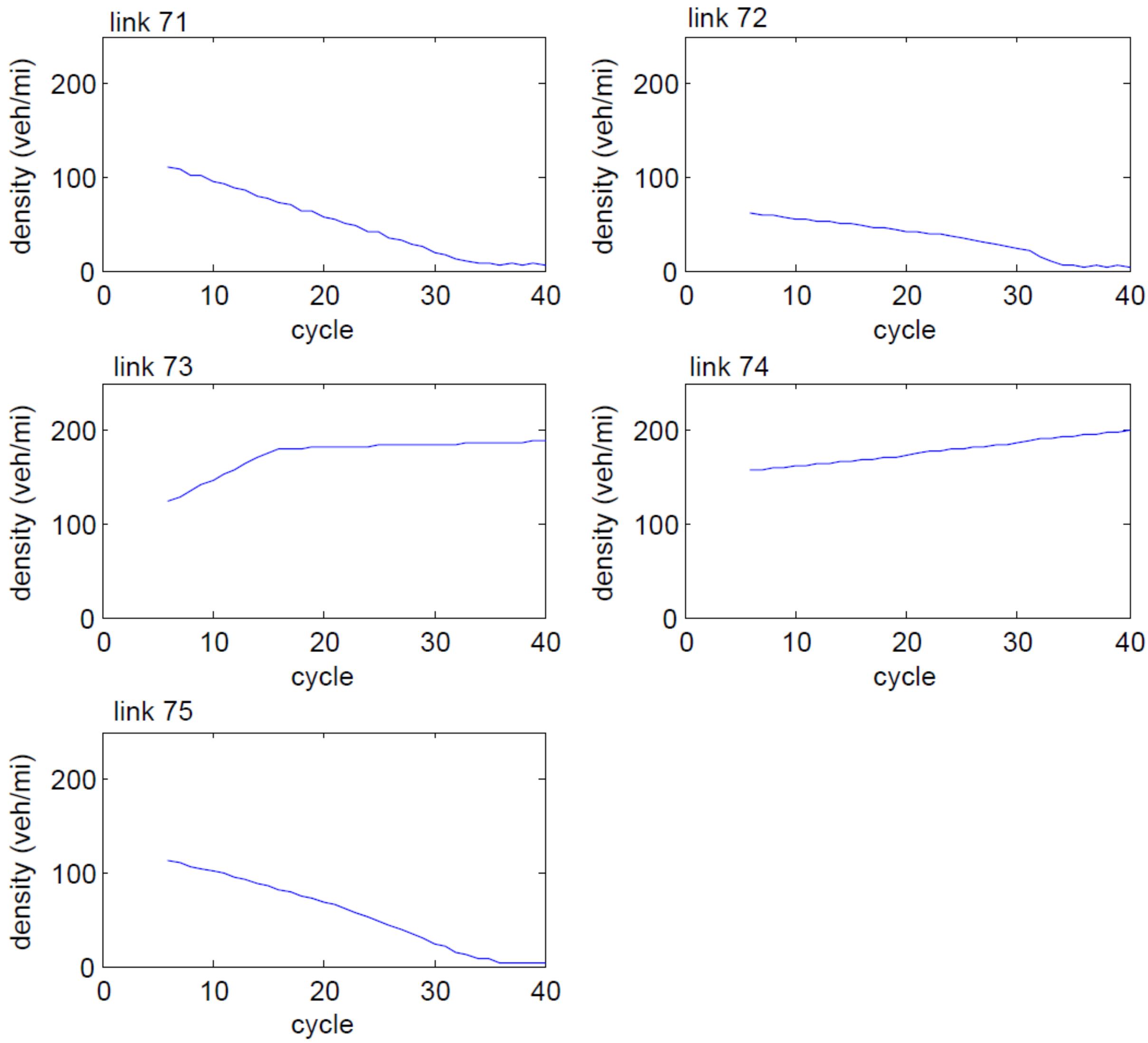


Figure 7.3-8 Initial Link Densities (Links 71-75)

From the above observations, the links with the highest average densities are selected as candidate projects for link improvements. In addition to link improvements, capacity improvements for intersections are also considered. That is, the upstream and downstream intersections of selected links are also considered for capacity improvements. The following figures display the “cyclic delay” at each intersection. This cyclic delay at each intersection is defined as the total time all vehicle are stopped in queue while waiting to pass through the intersection (veh.s).

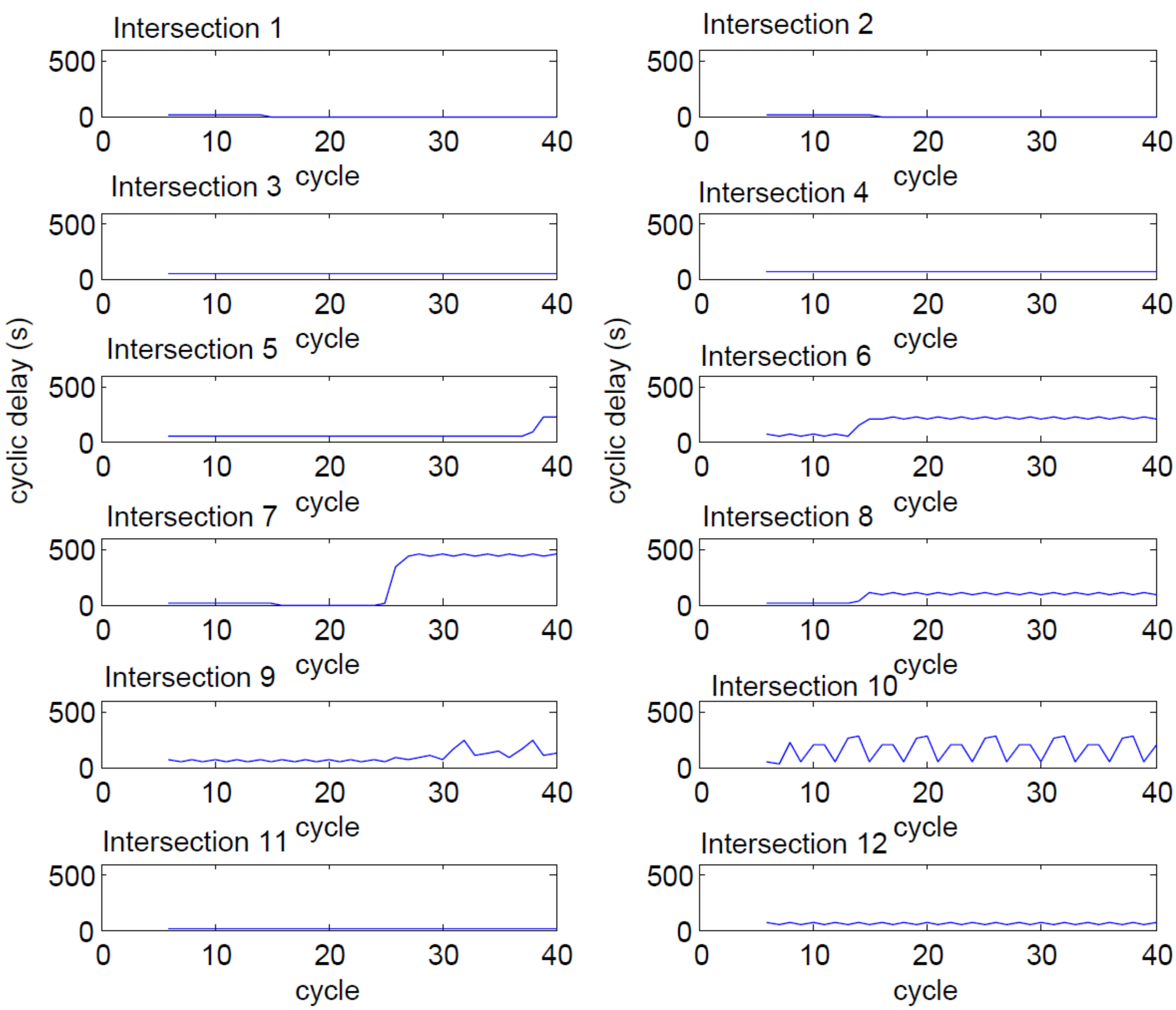


Figure 7.3-9 Initial Intersection Cyclic Delays (Intersections 1-12)

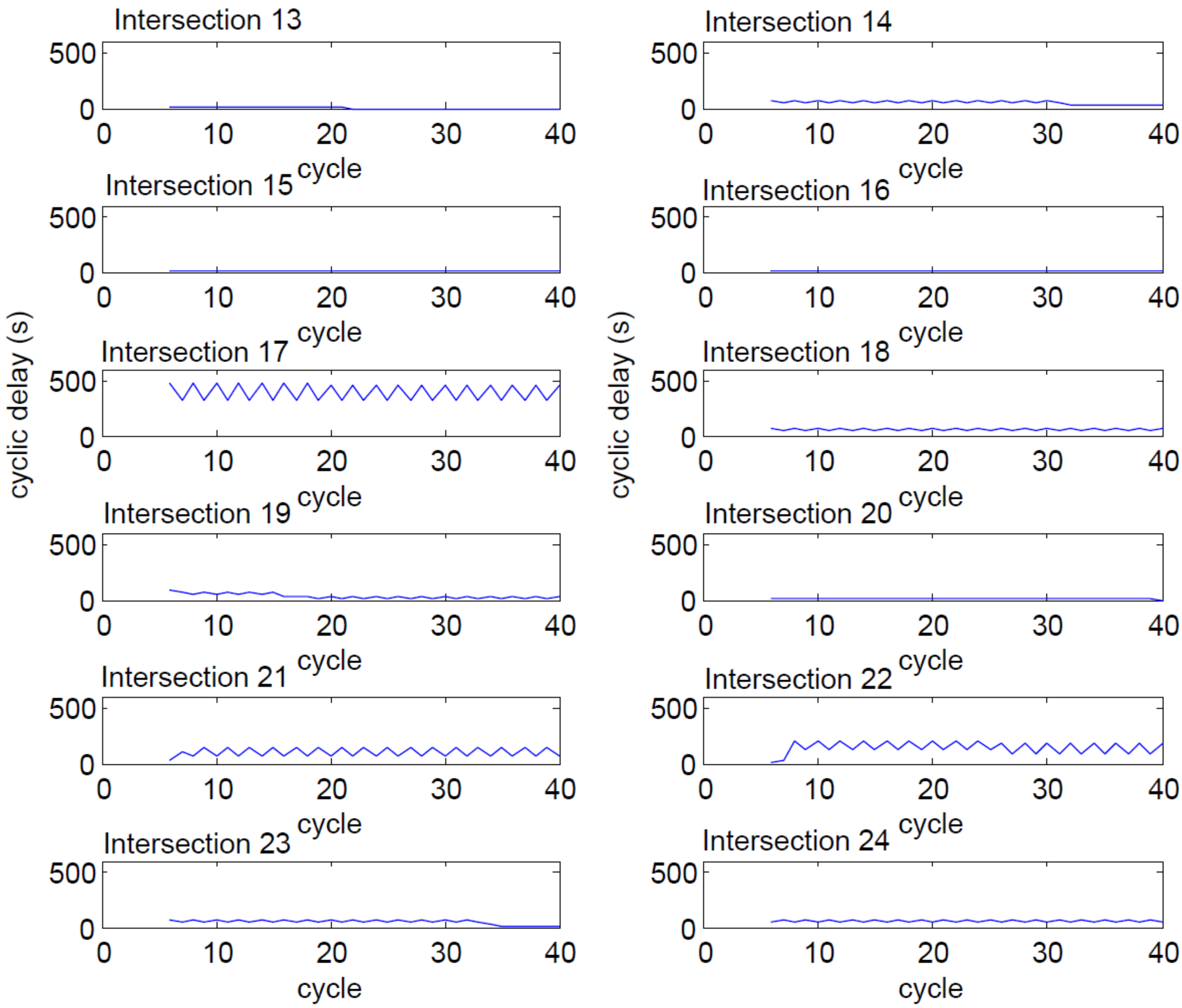


Figure 7.3-10 Initial Intersection Cyclic Delays (Intersections 13-24)

Similar to previous analysis, projects are selected as pair of links between two nodes which means improvements are implemented on both directions. Figure 7.3-11 displays the average density of all links over 40 cycles.

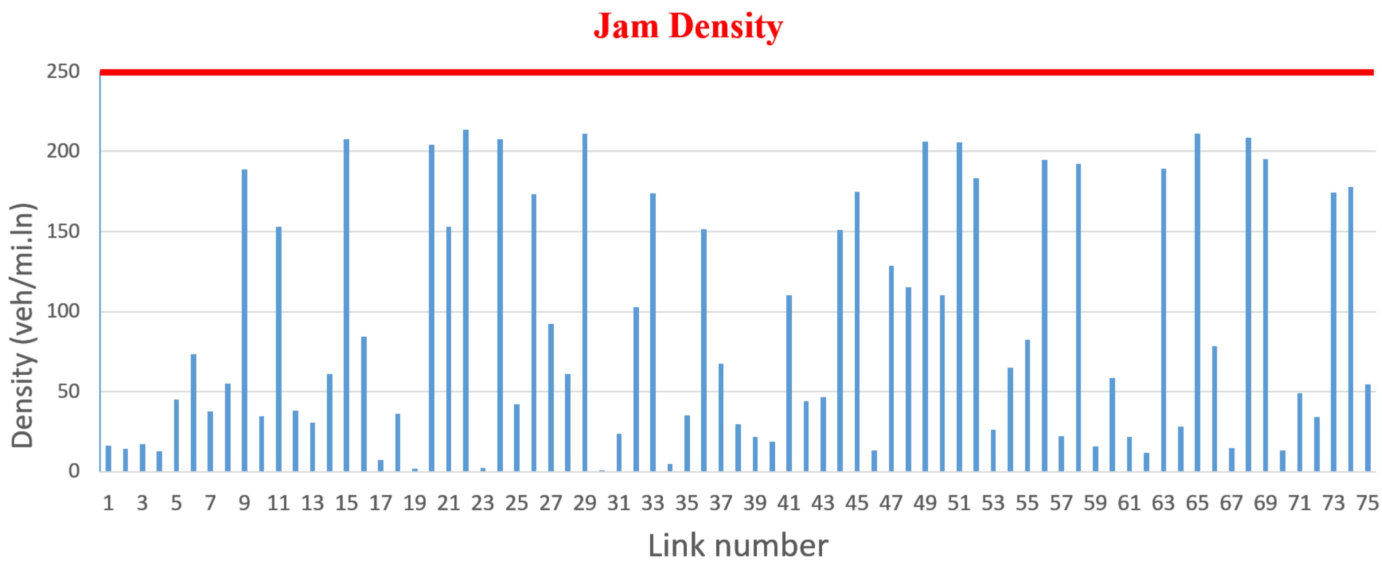


Figure 7.3-11 Initial Link Densities Variations among All Links

Accordingly, the links with highest densities (i.e. closest to jam density) are selected as candidate link improvements. The list is provided in Table 7.3-1. Additionally, the intersections at both ends of candidate links, are also selected for capacity improvements. Table 7.3-2 presents the list of selected intersections.

Table 7.3-1 Candidate Links and Initial Link Attributes

Project #	*link 1*	*link 2*	*length (mi)*	*Link 1 Density (veh/mi)*	*Link 2 Density (veh/mi)*
1	22	15	0.9	213.8	207.7
2	29	24	0.4	211.2	207.8
3	65	68	0.6	211.1	208.8
4	51	58	0.6	205.7	192.1
5	20	17	1.1	204.4	193.1
6	69	45	1.2	195.1	174.9
7	56	44	1.1	194.7	150.8

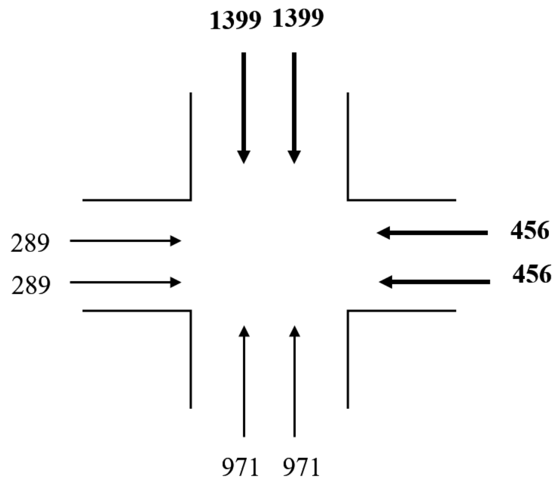
8	63	74	1	189.2	177.7
9	9	11	1	188.6	152.8

Table 7.3-2 Intersection Candidates and Initial Cyclic Delays

Intersection candidate	***Cyclic delay (veh.s/cycle)***	***Cycle time (s)***
4	68.3	169
5	59.0	156
6	175.4	166
8	69.9	119
9	89.6	73
10	163.0	156
15	0.1	180
17	391.5	177
19	29.9	29
21	104.0	33
22	147.8	161
24	61.1	169

As mentioned earlier, cyclic delay at each intersection is defined as the total time all vehicle are stopped in queue while waiting to pass through the intersection at each cycle (veh.s/cycle).

In Table 7.3-2, cycle time is calculated using equation 34. The example below demonstrates how the critical lane volume and the cycle length is estimated for intersection 8. Figure 7.3-12 illustrates the critical lanes for intersection 8.



V_c = 1399+456=1855 (veh/h)

Figure 7.3-12 Critical lanes for intersection 8

$$C_{min} = \frac{N * t_L}{1 - (\frac{V_c}{3600/h})} = \frac{2 * 4}{1 - (\frac{1855}{3600/1.81})} \sim 119\ (s)$$

Figure 7.3-13 displays the location of candidate projects on Sioux-Falls map.

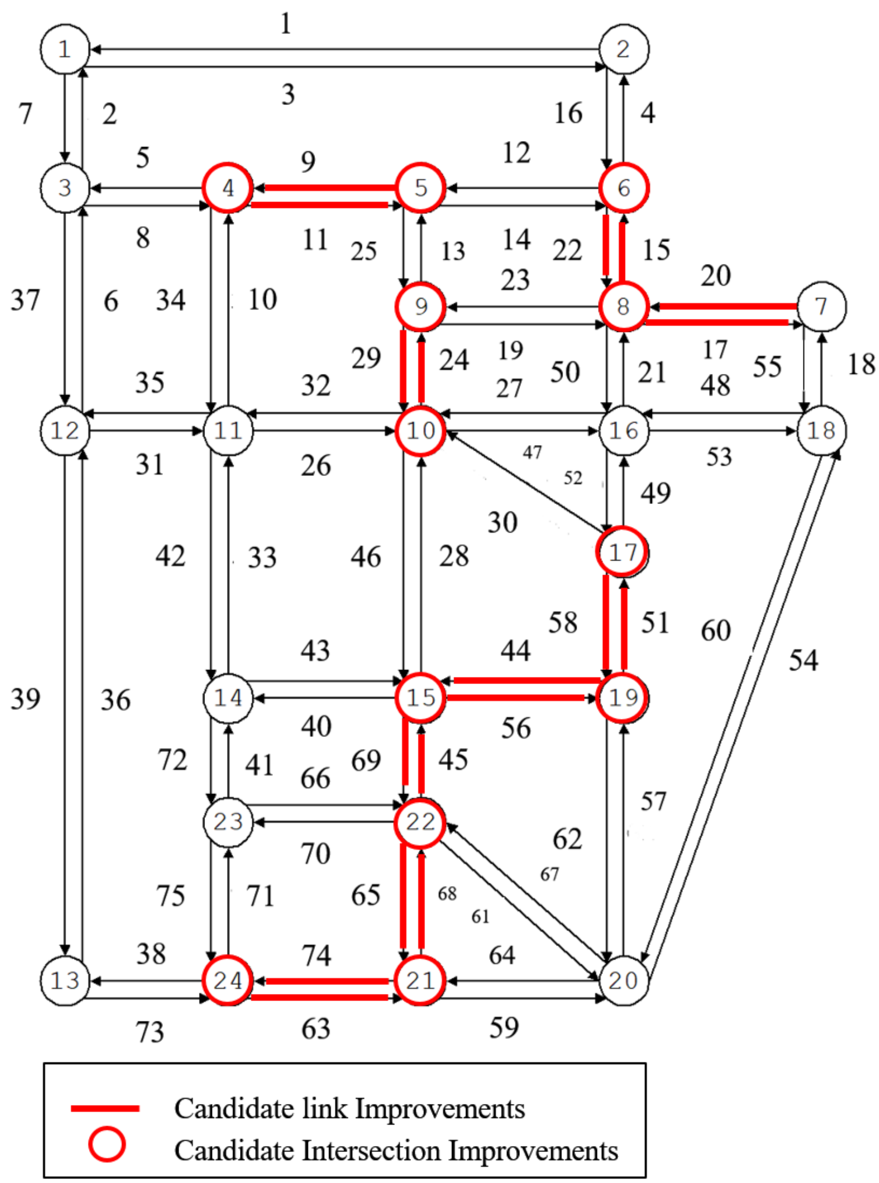


Figure 7.3-13 Location of candidate link and intersection improvement projects

7.4 Problem Formulation

7.4.1 Objective function

The objective function is similar to the one described in section 3.2. However, an additional term $\sum_{i=1}^{n_I} del_{ij}$ is added to account for intersection delays. n_I is the number of intersections while del_{ij}

denotes delay at intersection *i* at time period *j*. The goal is to minimize the PV of total cost including travel time, vehicle operating and safety costs.

$$min\, Z = \sum_{s=1}^{S} P_s \left\{ \sum_{j=1}^{T} \left\{ \frac{1}{(1+r)^j} \left(\sum_{i=1}^{n_l} w_{ijs} * v_t + \sum_{i=1}^{n_I} del_{ij} * v_t + \sum_{i=1}^{n_l} \{C_{vops(ij)} * VMT_{ijs}\} + N_{crs(j)} * \right. \right. \right. \\ \left. + \sum_{i=1}^{n_p} \frac{c_i x_i(t)}{(1+r)^t} \right. \tag{35}$$

As done earlier in section 4.1, the demand increases exponentially as a function of time over the planning horizon as follows:

$$d_{ij}^t = d_{ij}^0 * (1+r)^t \tag{21}$$

Three plausible demand scenarios are considered: (i) low demand growth, (ii) med (medium) demand growth, and (iii) high demand growth. Under the three demand growth scenarios we assume the growth rate per year r = 0.005, 0.01, 0.015 for the low, med and high scenarios, respectively (in Equation 35: S = {low, med, high}). Figure 7.3-1 displays demand change under three demand growth scenarios over 20 years of analysis.

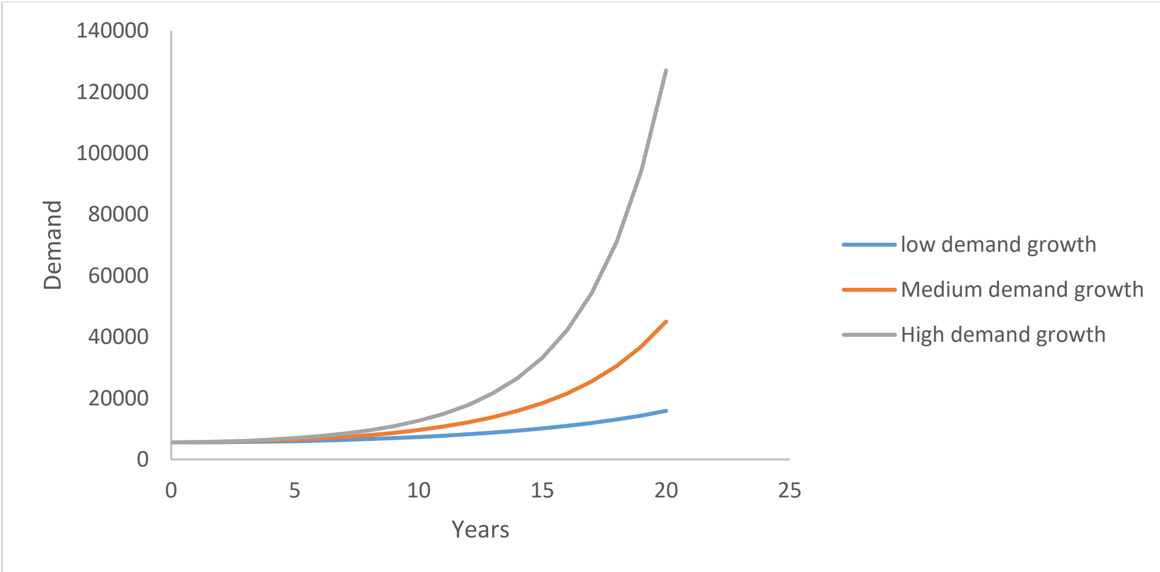


Figure 7.4-1 Demand growth scenarios

7.4.2 Constraints

Budget Constraint

The budget constraint is described earlier in section 3.2.3. As a reminder, assuming that n_p is the number of candidate projects, for $0 \leq t \leq T$ the budget flow constraint is formulated as:

$$\sum_{i=1}^{n_p} c_i x_i(t) \leq \int_0^t b(t)_{external} + b(t)_{internal}\, dt \tag{10}$$

$$b(t_i)_{internal} = VMT(t_{i-1}) * f_r * f_c * f_t \tag{9}$$

Precedence and Concurrent Relation Constraint

In addition to the budget constraint, the precedence and concurrent relation constraints among projects are considered in this study. Generally, due to political or geographical considerations, it may be necessary to implement some projects before others or some projects might have to be implemented concurrently with other projects. (Wang and Schonfeld, 2012).

In this study we consider both link and intersection improvements. Logically, it is preferred to concurrently improve links and their upstream and downstream intersections or have the end-point intersections improved prior to the link improvements. Therefore, this constraint is enforced to the link and intersection projects.

For a given sequence S, let's assume that h_i for $i=1 \ldots n_p$ is the location of project *i* in the sequence. If the precedence relation requires that project i precedes project j, then this constraint could be demonstrated as $h_i < h_j$. A scheduling problem may have multiple precedence relation constraints, and a sequence that violates any of them is infeasible.

In order to maintain the diversity of solutions in GA (the GA and its operators are explained further in the text in section 3-5), it is not desirable to discard solutions that violate the precedence constraints. Instead, a high penalty for solutions that violate the precedence constraint cab be considered. This means that if any solution in the population violates these constraints, instead of evaluating it, a very large number (10^{20}) is assigned to its fitness value. This will significantly reduce the chance of the infeasible solution to be selected for mutation and cross over for the next generation.

7.4.3 Benefits-Costs

Conventional methods for prioritizing projects are based on the benefit-cost ratio. This section provides the details on estimating the benefits and costs of each project. However, benefit-cost ratio does not yield the optimal sequence of projects as is proven at the end of this study. In this section, the benefits and costs of each individual project is introduced. Then the benefit-cost ratio of each projects is estimated.

Project Costs

It is assumed that the cost of link improvements is $4,500,000 per lane. mile. The cost of intersection project *i* can be estimated as:

$$C_i = A_{I_i} * C_{c_i} + A_i * C_p = A_{I_i} \cdot 51.6 + A_i \cdot 20 \quad (36)$$

where:

C_{c_i} – capital cost of improvement of intersection *i ($/ft^2)*,

C_p – unit cost of pavement maintenance *($/ft^2)*,

A_{I_i} – area of the land needed to improve intersection *i (ft^2),*

A_i – overall area of the intersection *i (ft^2)*

Project Benefits

Although there are different approaches to calculate the benefits of projects, in this study the benefit of each project is the cost saving form implementing a project compared to the no project scenario. Accordingly, the benefits, costs and the benefit-cost ratio of each project are summarized in the following table:

Table 7.4-1 Project Benefits and Costs

Project #	***links***	***Intersections***	***Length (mi)***	***cost ($million)***	***Benefits ($million)***	***B/C ratio***
1	22,15	6,8	0.9	3.6	24.8	6.9
2	29,24	9,10	0.4	2.1	2.3	1.1
3	65,68	22,21	0.6	2.7	3.6	1.3
4	51,58	19,17	0.6	2.2	4.7	2.1
5	20,17	8	1.1	3.3	12.9	3.9

6	69,45	15,22	1.2	4.0	16.3	4.0
7	56,44	15,19	1.1	3.3	11.9	3.6
8	63,74	21,24	1	3.4	22.0	6.4
9	9,11	4,5	1	3.9	6.2	1.6

7.5 Optimization Process

Figure 7.5-1 illustrates the optimization process. Each population is comprised of *I* sequences, and each sequence *i* is a string of *J* numbers which represents the location of the candidate project.

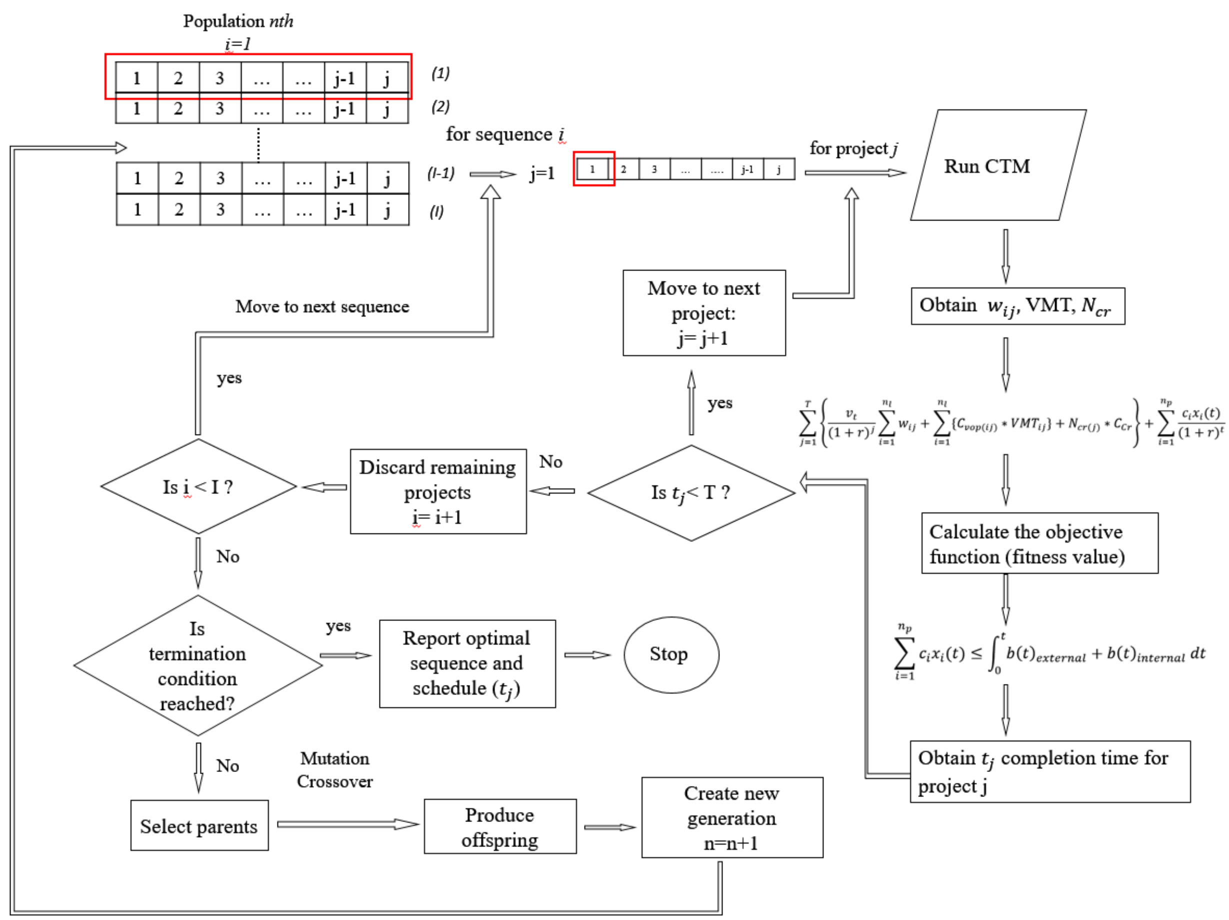


Figure 7.5-1 Optimization Process

The analysis begins by selecting the greedy-order solution (ranked based on vehicle density) and bottleneck-order solution (ranked based on benefit-cost ratio) presented in Table 7.5-1 as initial population. For each sequence *i*, the algorithm selects projects 1 to j one-by-one and runs the CTM after each one is implemented. In this sense, each project implementation requires a specific change in the network e.g. widening lanes and increasing the intersection capacity. At each step, the CTM outputs the cell occupancies which can be translated into link densities. Using the fundamental diagram in Figure 7.2-1 travel time, and speed are obtained which are plugged to the objective function to calculate the fitness value (i.e. objective function). Next, the budget constraint, uniquely determines the completion time t_j of project *j*. That is, project *j* is completed as soon as the available budget equalizes the cost of project. This process is performed for all projects in the sequence until the completion time exceeds the planning horizon *T*. Then the algorithm moves to the next sequence until all sequences in the population are evaluated. At this step, the best sequences i.e. the ones with lowest fitness values, are given higher probabilities to be selected as parents and produce offspring. Through several crossover and mutation operators the selected parents produce the next generation. The algorithm begins to evaluate the new generation and continues to do so until the termination criterion is met. In this case, the algorithm stops if the optimal sequence does not change after 10 generations.

Table 7.5-1 Bottleneck and Greedy-order Sequences

Bottleneck-order	***Density (veh/mi)***	***Greedy-order***	***B/C ratio***
1	213.7	1	6.9
2	211.2	8	6.4
3	211.0	6	4.0
4	205.6	5	3.9

5	204.3	7	3.6
6	195.0	4	2.1
7	194.7	9	1.6
8	189.2	3	1.3
9	188.5	2	1.1

7.6 Results

For this test network, 9 link improvements and 12 intersection improvements are considered over 20 years planning horizon. As stated before precedence and concurrent implementation constraints are imposed to implement end-point intersections prior or simultaneous with each link improvement. The proposed GA is used to search for the sub-optimal sequence, and the proposed CTM is applied to evaluate each sequence. Table 7.6-1 optimized sequence and schedule for the candidate link and intersection improvements.

Table 7.6-1 Optimized Sequence and Schedule from GA

Optimal sequence	*Links improved*	*Intersections improved*	*Schedule (completion year)*	*Total cost (PV)*
2	links 29, 24	9,10	1.4	$2,942,134,572
9	links 9, 11	4,5	4.7	$4,597,458,784
3	links 65, 68	22,21	6.2	$5,805,421,657
1	links 22,15	6,8	8.6	$7,310,551,145
5	links 20, 17	-	10.8	$8,522,919,492

7	links 56, 44	15,19	13.4	$9,693,185,295
4	links 51, 58	17	14.9	$10,459,070,244
8	links 63, 74	24	17.3	$11,386,405,776
6	links 69, 45	22	18.8	$12,049,723,799

Figure 7.6-1 illustrates the evolution of GA process. It indicates the fitness value for each individual in each generation. This optimization process is completed after the genetic search has stopped improving for 10 generations. In this case the algorithm reaches convergence in generation 40 at stops at the 50th generation.

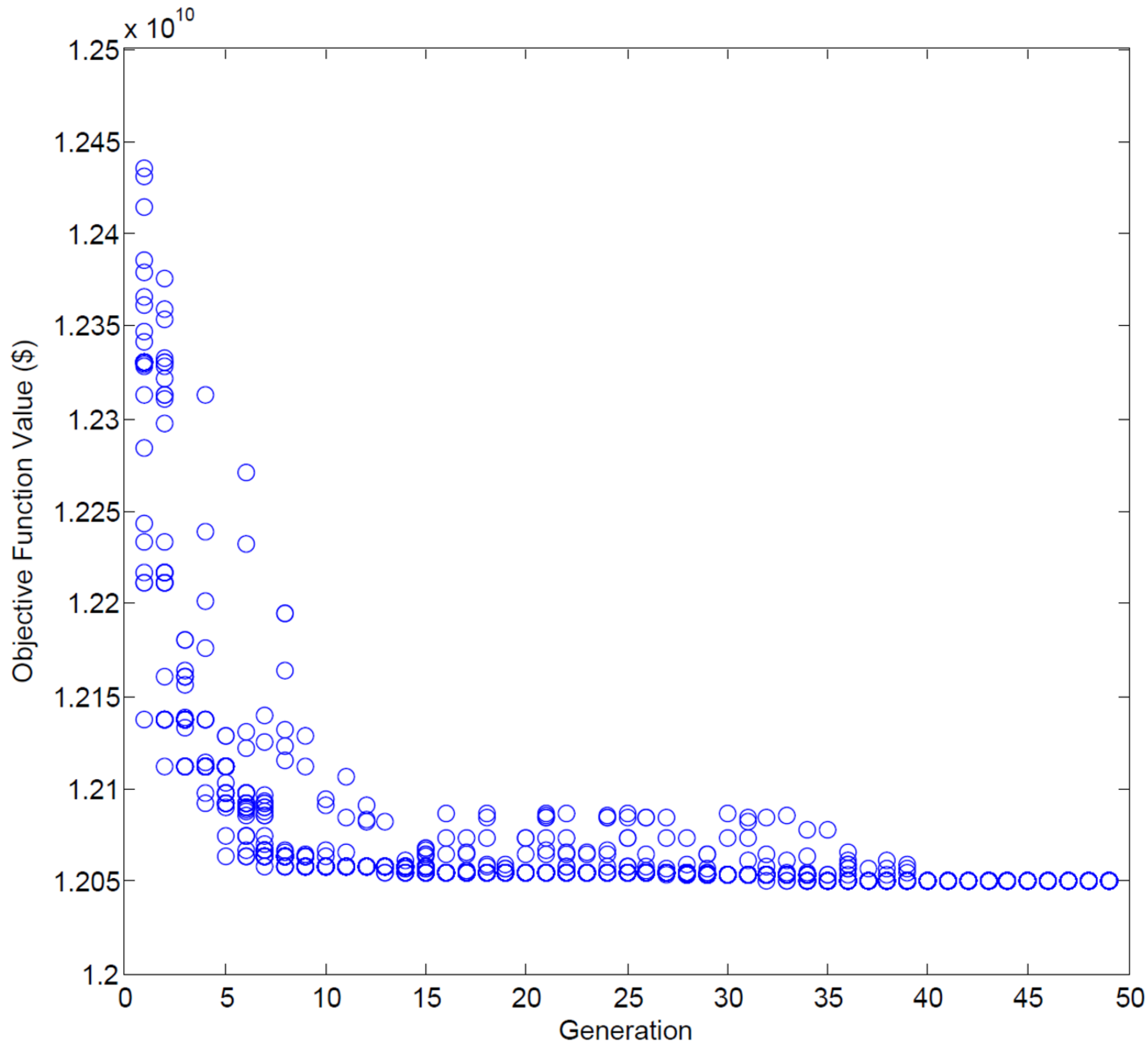


Figure 7.6-1 Genetic Search Progress

Intuitively, if projects are considered individually, the construction projects would be implemented according to the rank of their congestion severities (Bottleneck-order), or their benefit-cost ratio would be their proritization criteria (Greedy-order). Table 7.6-2 presents the sequence and schedule and total accumulated cost from these two ranks. which is different from the optimazed GA results in Table 7.6-1. It can be seen that the bottleneck and greedy-order solutions result in total costs of 1.247×10^9 and 1.259×10^9 respectively. However, the optimal sequence found in Table 7.6-1 yields a lower total cost of 12.049×10^9 .

Table 7.6-2 Greedy and Bottleneck Order Solutions

Bottle-neck solution			**Greedy solution**		
sequence	**schedule**	**Total cost(PV)**	**sequence**	**schedule**	**Total cost (PV)**
1	2.4	$2,653,761,866	**1**	2.4	$2,653,761,866
2	3.8	$4,053,972,412	**8**	4.9	$4,801,308,166
3	5.3	$5,383,842,359	**6**	6.4	$6,059,315,892
4	6.8	$6,588,495,541	**5**	8.6	$7,561,467,852
5	9.0	$8,029,698,012	**7**	11.2	$9,006,384,789
6	10.5	$9,016,009,862	**4**	12.7	$9,919,951,558
7	13.1	$10,290,858,238	**9**	16.0	$11,242,543,974
8	15.5	$11,368,803,475	**3**	17.5	$11,946,111,153
9	18.8	$12,478,192,927	**2**	18.8	$12,591,957,683

Figure 7.6-2, Figure 7.6-3, and Figure 7.6-4 indicate the accumulated cost for the GA optimized sequence, bottleneck-order and greedy-order sequences respectively. The cost is broken down into travel time, vehicle operating, and crash cost as formulated in the objective function.

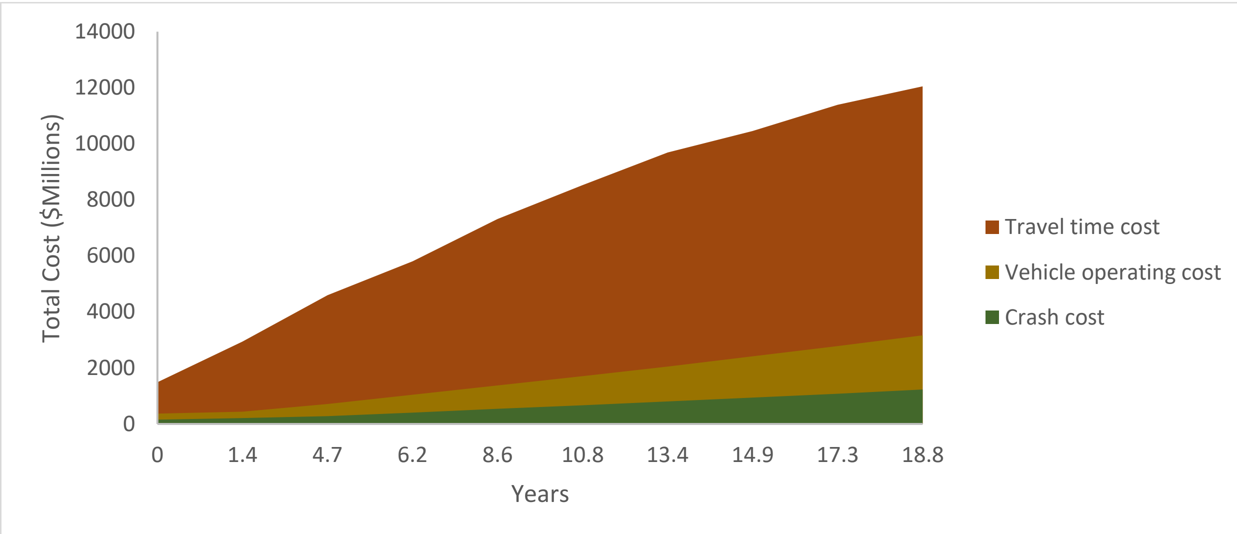


Figure 7.6-2 Cost Change Over Analysis Period for GA solution

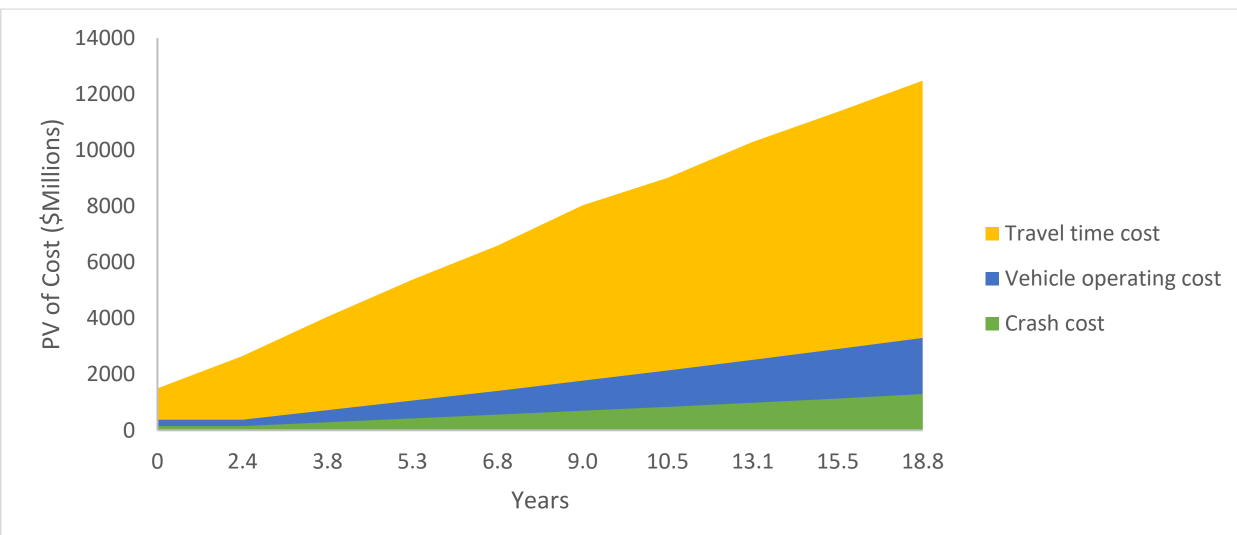


Figure 7.6-3 Cost Change Over Analysis Period for Bottleneck-order solution

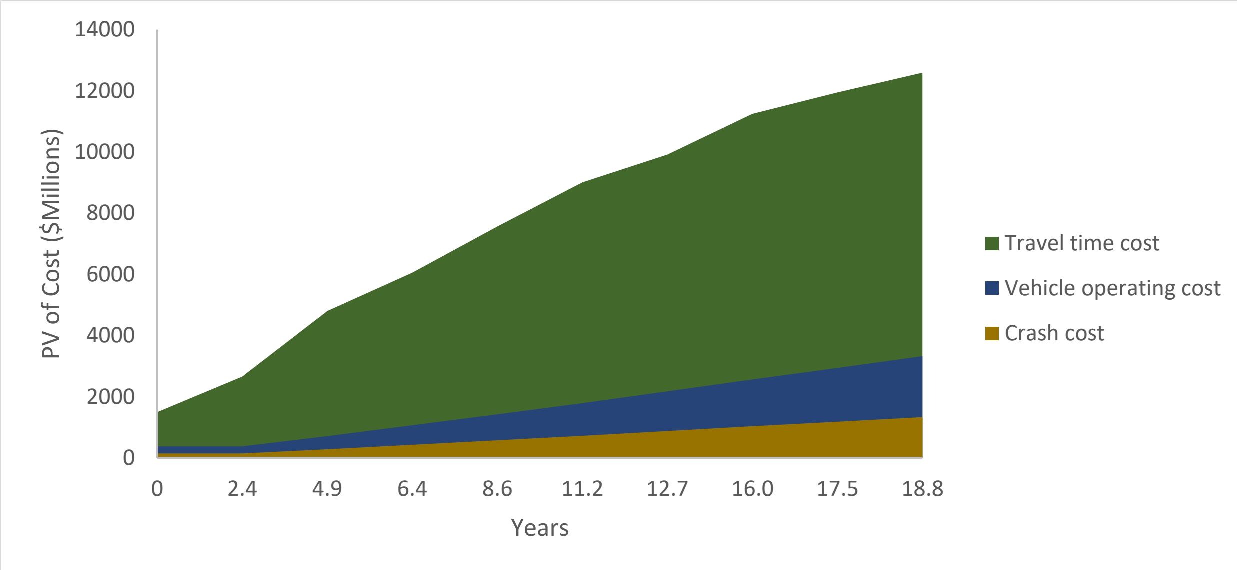


Figure 7.6-4 Cost Change Over Analysis Period for Greedy-order solution

Table 7.6-3 shows to what extent the GA improves the outcome of the project sequencing and schedule. More specifically, the table shows the percentage of cost savings resulting from employing the GA compared to bottleneck and greedy ranks. It can be seen that the GA improves costs for all cost categories. It improves the total cost by 3.6% from bottleneck-order and 4.5% from greedy-order solution. Note that these values should be interpreted in comparison with the "No Project" scenario. Figure 7.6-5 compares the "No project" scenario where no improvement projects are implemented, versus the GA solution. The results indicate a 6.9% improvement in total cost. The improvement seems low because in this case we are dealing with a relatively small and low congested network. The improvements are expected to be much higher for a larger network with higher level of congestion. As such, 3.6% and 4.5% improvements from the GA solution compared to the bottle-neck and greedy-order solutions are significant in this case.

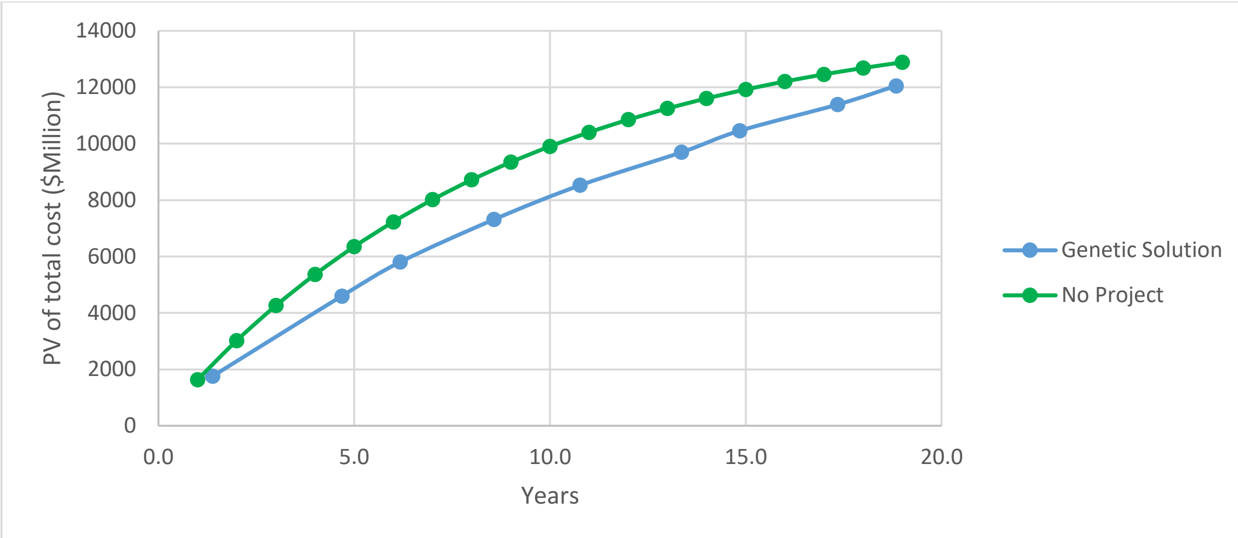


Figure 7.6-5 GA Solution versus "No Project"

Table 7.6-3 Cost Saving from GA Compared to Bottleneck and Greedy Order Solutions

	Bottle-neck solution	***Greedy solution***
Crash cost	5.2%	8.3%
Vehicle operating cost	3.5%	3.0%
Travel time cost	3.3%	4.3%
Total Cost	3.6%	4.5%

Figure 7.6-6 and Figure 7.6-7 illustrate total cost change over the analysis period for GA solution, greedy and bottleneck-order solutions. It is evident that at each period the genetic solution yields a lower objective value.

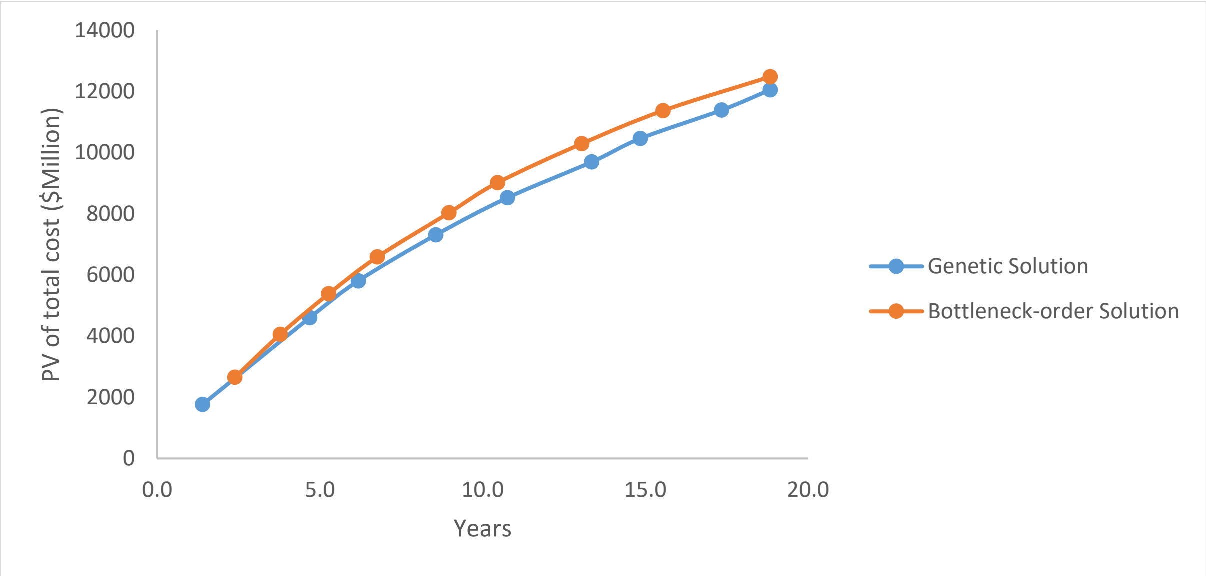


Figure 7.6-6 GA versus Bottleneck solution

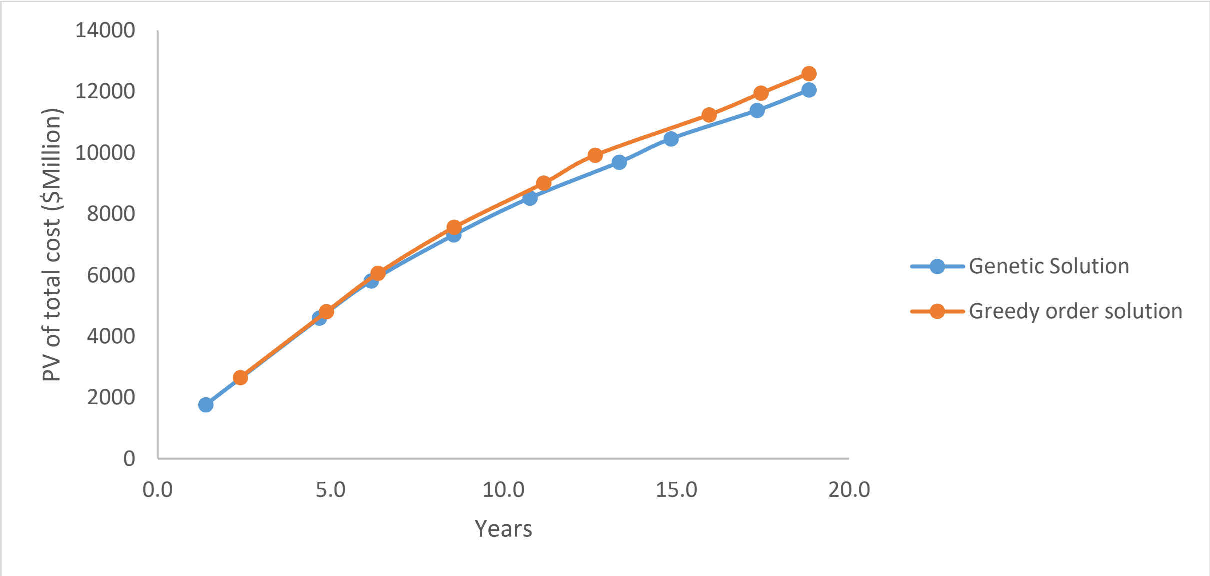


Figure 7.6-7 versus Greedy solution

Figure 7.6-8 and Figure 7.6-9 display how the implemented projects impact the cyclic delays at each intersection. It can be seen that in some cases the delay has increased, however, the intersections with the highest cyclic delays such as 6,7,10,17 and 22 are significantly improved and experience much less delay compared to the "before" scenario.

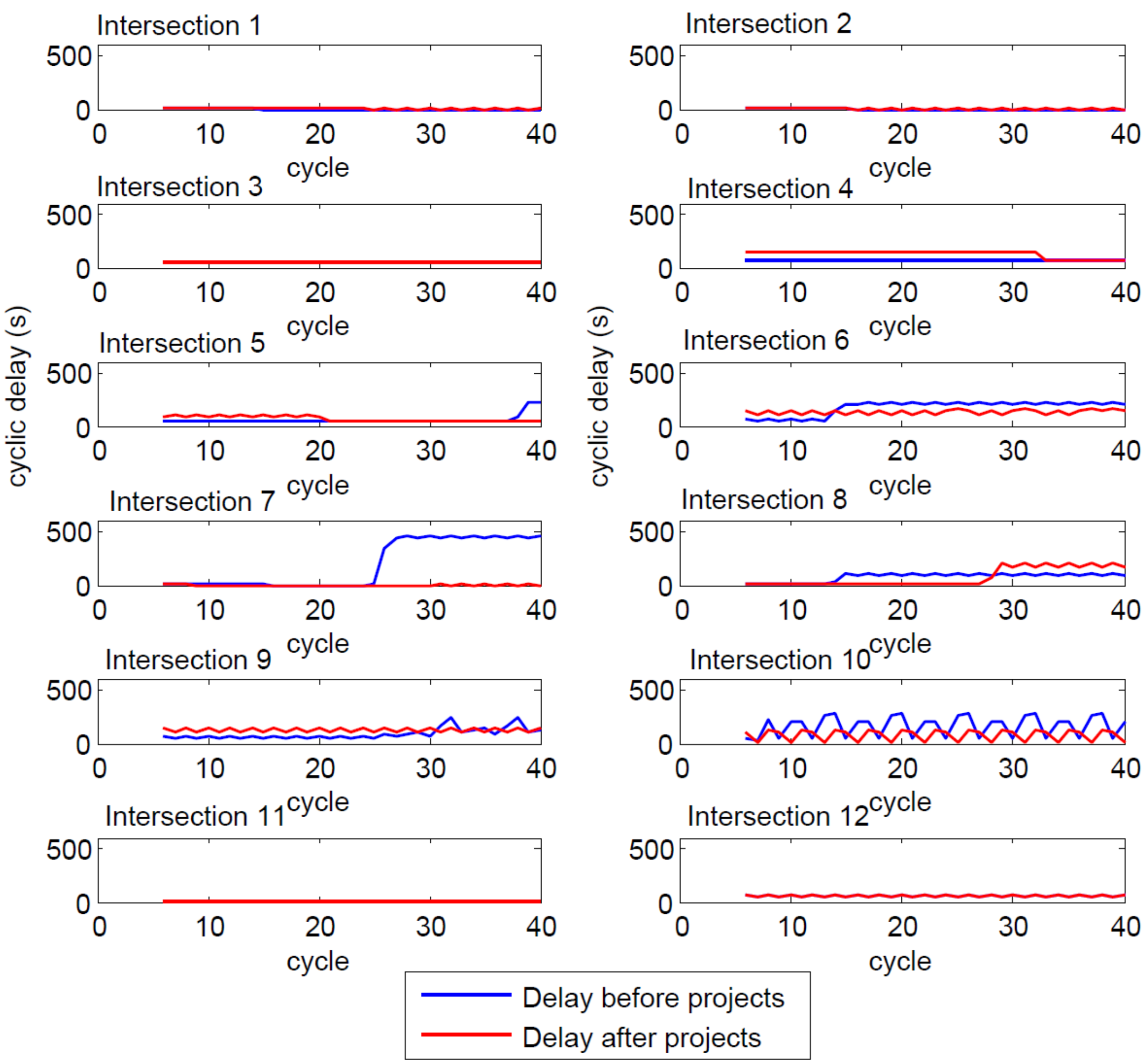


Figure 7.6-8 Cyclic Delays Before and After Projects (intersections (1-12)

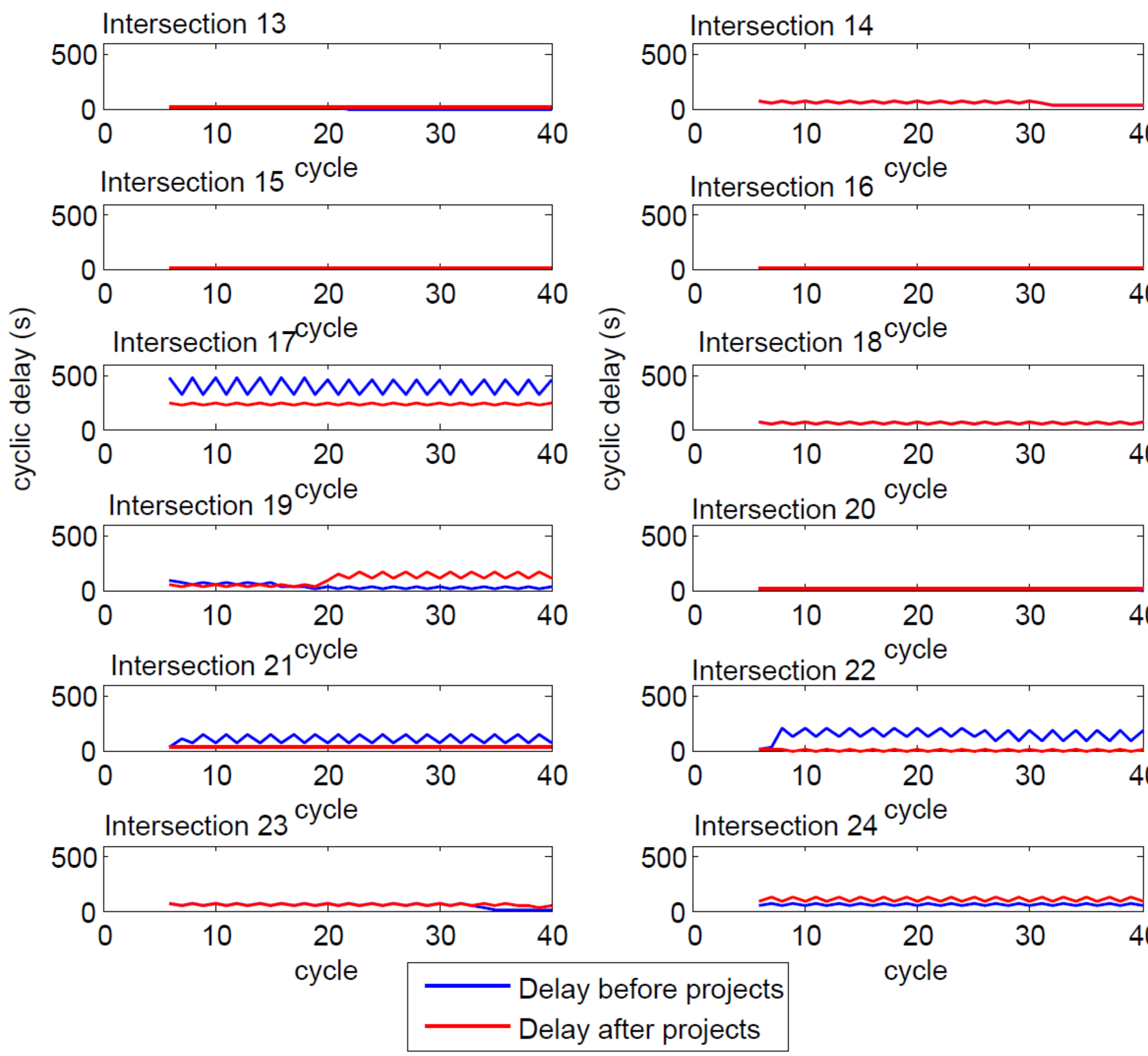


Figure 7.6-9 Cyclic Delays Before and After Projects (intersections (13-24)

The following figures illustrate the density improvements on the most congested links. It can be seen that the density is significantly reduced on most links.

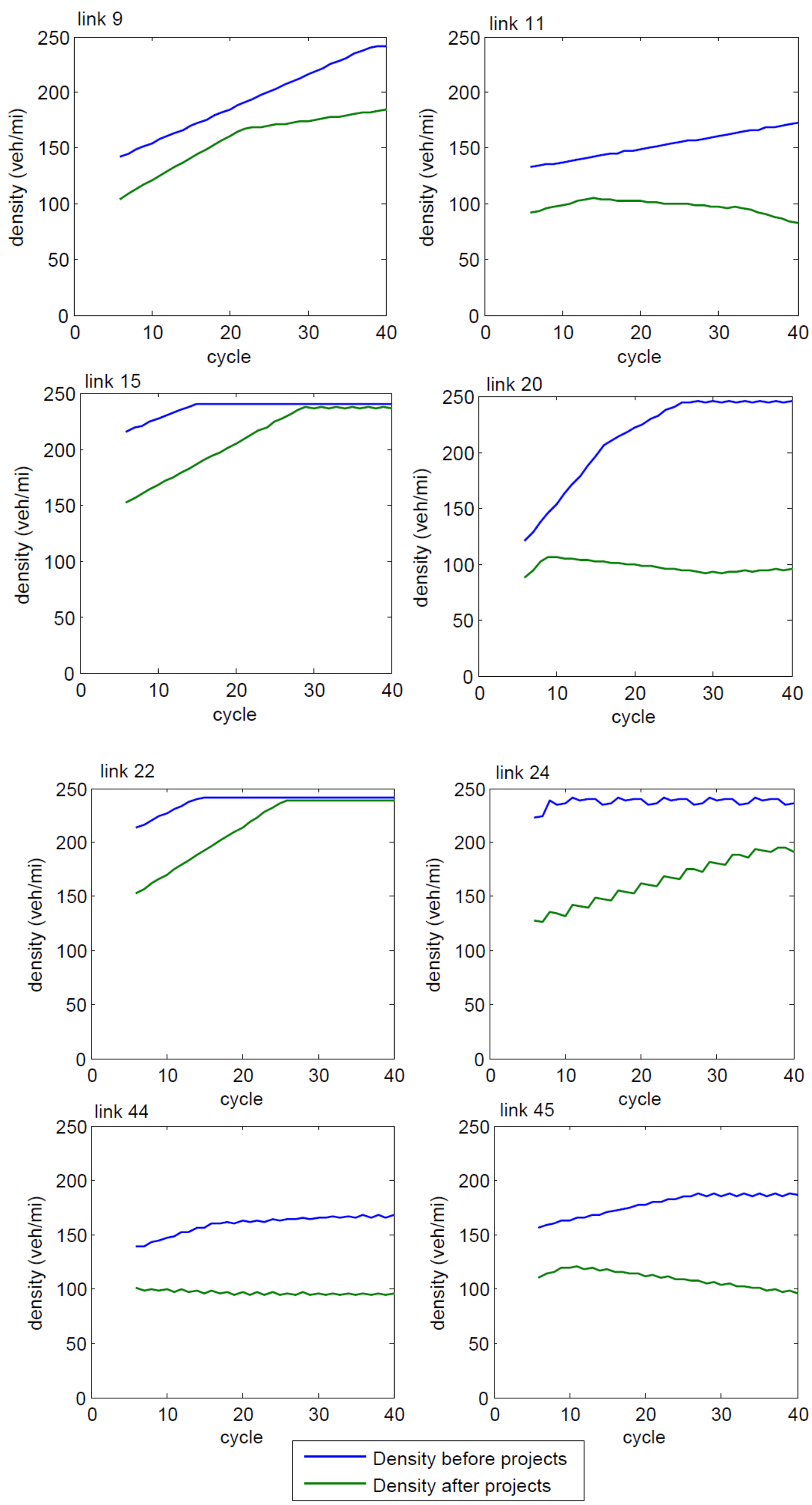


Figure 7.6-10 Link Densities Before and After Projects (1)

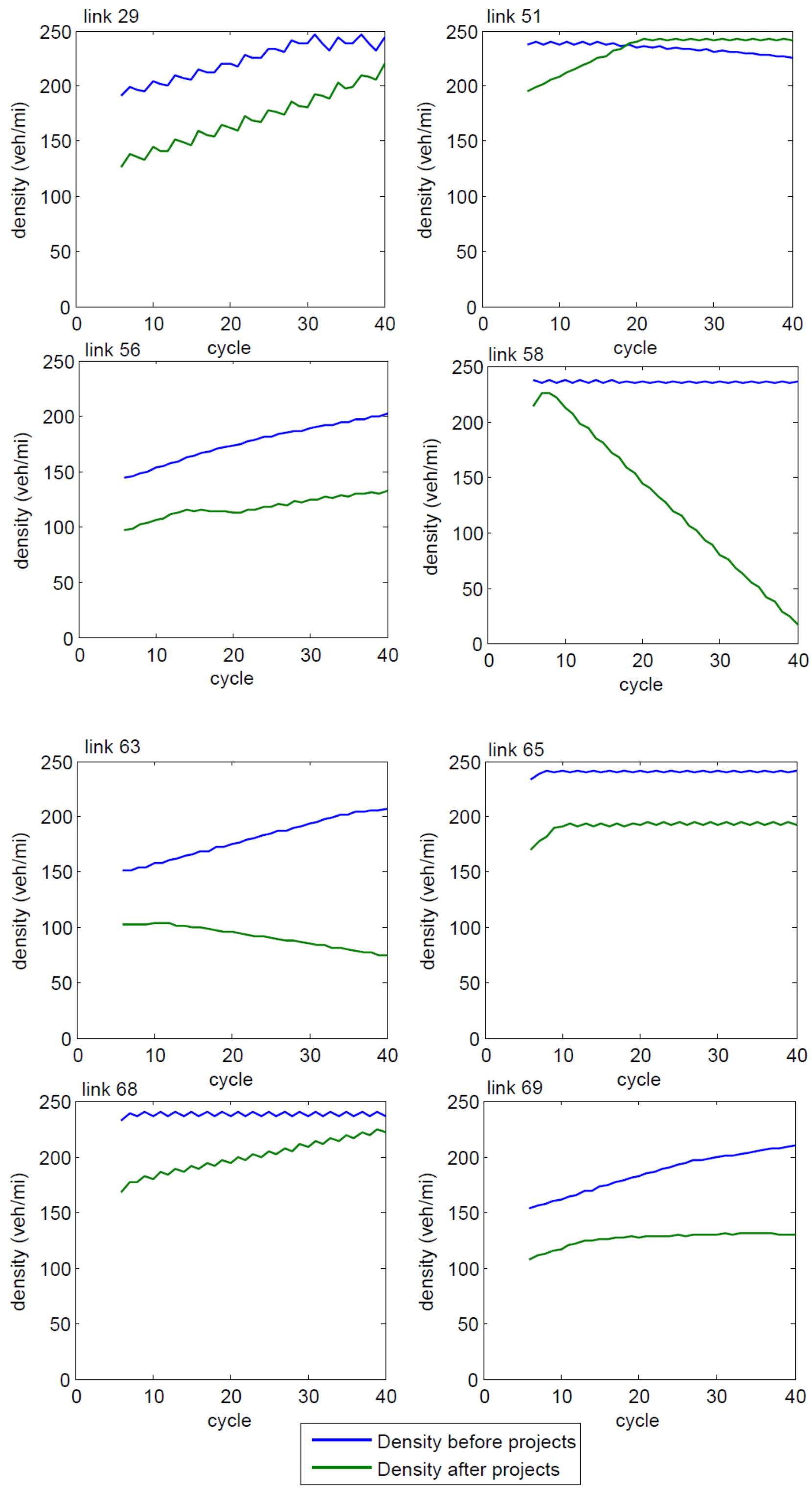


Figure 7.6-11 Link Densities Before and After Projects (2)

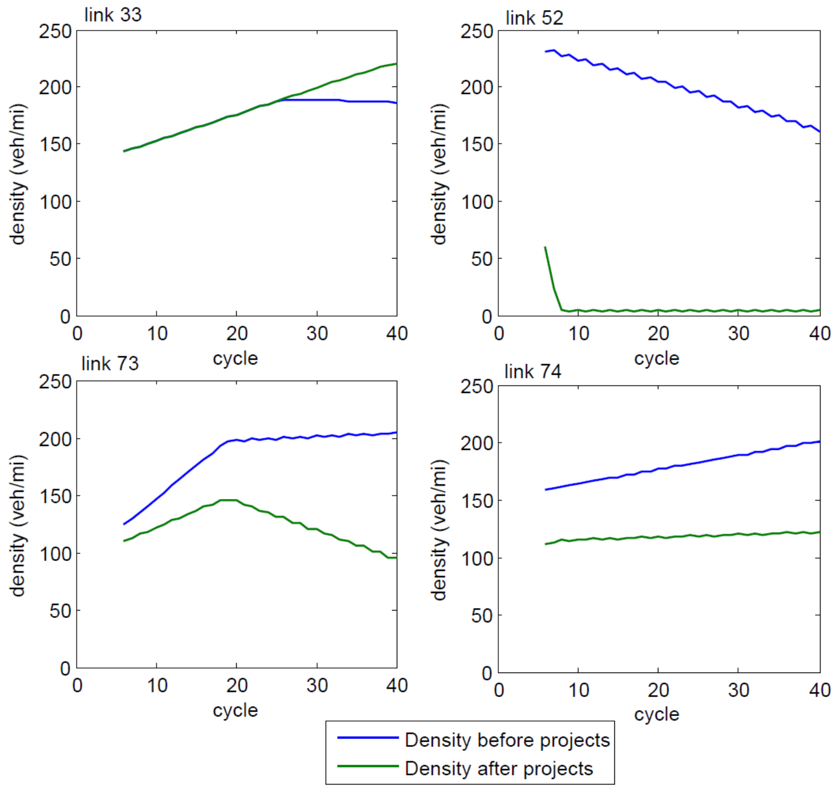


Figure 7.6-12 Link Density Before and After Projects (3)

It can be concluded from this case study that for a complex system, even though the improvements are simply adding capacity to links and intersections, no general project sequencing and scheduling pattern can be discerned from the resulting optimal sequence due to the interrelations and diverse geometric configurations. Therefore, an optimization process such as the one proposed in this study, including efficient and accurate evaluation and search algorithms can effectively solve the problem of planning and scheduling of interrelated systems.

7.7 Sensitivity Analysis

7.7.1 Optimal Sequence and Schedule

This section explores how the uncertainty in the output of the optimization model (sequence and schedule of projects) can be apportioned among different sources of uncertainty in inputs. This is useful in understanding model behavior and enhancing the efficiency of the proposed methodology. For this purpose, sensitivity analysis is conducted to investigate the effects based on project cost, available budget and demand growth rate. In real life systems, such factors could significantly impact the output of the optimization model.

Project cost

Table 7.7-1 demonstrates the sensitivity of the optimized sequence, schedule, and the objective function value (total cost) to changes in project cost. The cost of each project is changed by the same percentage of their original value specified in Table 7.4-1. It is evident that as cost increases the optimal sequence slightly changes, and the completion time and total cost increases.

Table 7.7-1 Sensitivity of Optimal Sequence and Schedule to Project Cost

70% Cost		100% Cost		150% Cost	
Sequence	Schedule	Sequence	Schedule	Sequence	Schedule
3	1.0	2	1.4	9	4.9
9	3.4	9	4.7	2	7.0
2	4.3	3	6.2	1	10.6
1	6.0	1	8.6	5	13.9
5	7.5	5	10.8	3	16.2
4	8.6	7	13.4	7	20.0
7	10.4	4	14.9	4	22.3
8	12.1	8	17.3	8	26.0
6	13.2	6	18.8	6	28.3
PV of Total cost	$10,805,066,641	Total cost	$12,049,723,799	Total cost	$13,198,762,022

Figure 7.7-1 demonstrates how the objective function value (in terms of PV of total cost) increases as the available budget grows.

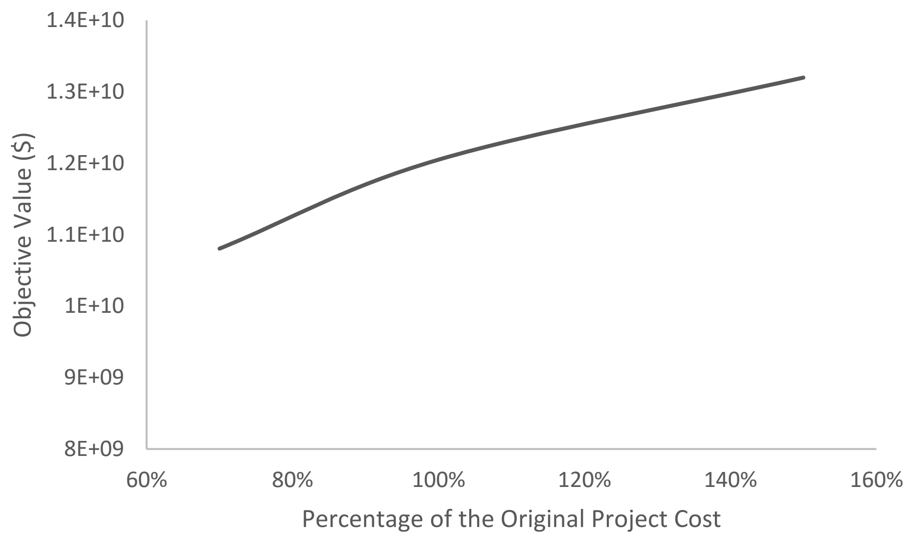


Figure 7.7-1 Objective Value for Different Levels of Project Cost

Available Budget

Similarly, Table 7.7-2 presents the sensitivity of results to changes in the available budget. The variation in budget is specified as different percentages of the original value, which was set to $1.5 million/year. It can be seen that the available budget affects the optimal sequence, schedule and total cost.

Table 7.7-2 Sensitivity of Optimal Sequence and Schedule to Available Budget

50% Budget		100% Budget		150% Budget	
Sequence	Schedule	Sequence	Schedule	Sequence	Schedule
3	2.8	2	1.4	9	0.9
9	9.4	9	4.7	2	3.1
2	14.1	3	6.2	1	4.8
1	18.5	1	8.6	5	6.3
5	21.5	5	10.8	3	8.0
4	24.5	7	13.4	7	9.0
7	29.7	4	14.9	4	10.0
8	34.7	8	17.3	8	11.0
6	37.7	6	18.8	6	12.6
Total cost	$13,579,074,339	Total cost	$12,049,723,799	Total cost	$10,694,073,881

Figure 7.7-2 demonstrates how the objective function value (in terms of PV of total cost) decreases as the available budget grows.

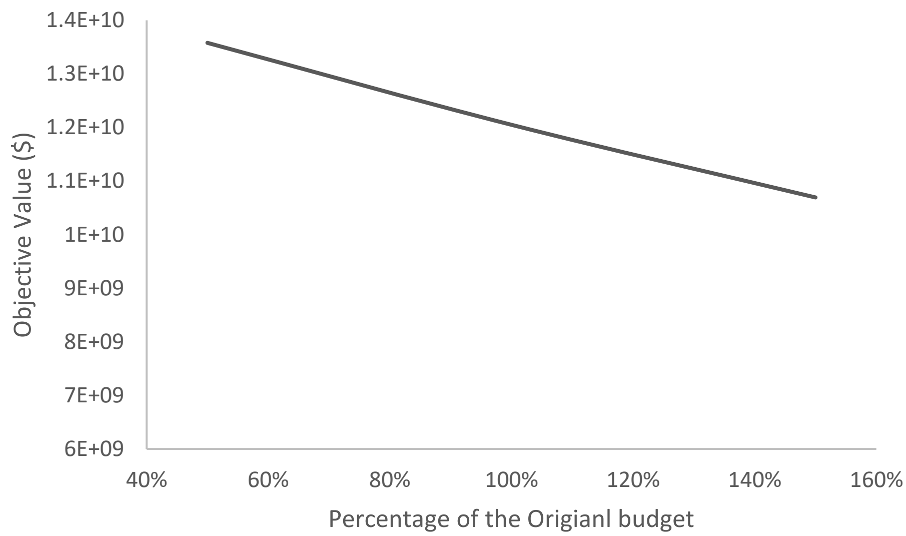


Figure 7.7-2 Objective Value for Different Budget Levels

Demand Growth

Table 7.7-3 shows the optimized sequence and total cost under three demand growth scenarios discussed in section 7.2.1. If decisions are made only based on the average scenario, some high costs are expected if the high demand scenario occurs. In this case, the results indicate that the total cost under the average demand growth scenario (solving only a deterministic program) is 2.1% above that using the proposed stochastic program. This shows that solving the stochastic model yields a better solution with a lower objective function, i.e. yields a solution with a lower cost. Furthermore, the difference between the objective function value (PV of total cost) of the deterministic and stochastic program, which is called the Value of Stochastic Solution (VSS), is

$252 million. This value shows the possible gain from using the stochastic model rather than using the expected value and solving the deterministic model.

Table 7.7-3 Optimal sequence and total cost for different demand scenarios

	Growth rate	*Optimal Sequence*	*Total cost*
Low demand growth	0.005	2-9-3-1-5-4-7-6-8	11,682,443,810
Medium demand growth	0.01	2-9-3-5-1-7-8-4-6	12,302,399,183
High demand growth	0.015	9-2-1-5-3-7-4-8-6	12,886,405,776
Stochastic solution	-	2-9-3-1-5-7-4-8-6	12,049,723,799

Interest Rate

Table 7.7-4 demonstrates the sensitivity of the optimized sequence, schedule, and the objective function value (total cost) to changes in interest rate. For this purpose, the interest rate is given values from 3% to 12%. From Table 7.7-4, it is evident that the objective value is higher for lower interest rates; however, the sequence and the schedule appear not to change for different values of interest rate.

Table 7.7-4 Sensitivity of Optimal Sequence and Schedule to Interest Rate

3% Interest Rate		7% Interest Rate		12% Interest Rate	
Sequence	Schedule	Sequence	Schedule	Sequence	Schedule
2	1.4	2	1.4	2	1.4
9	4.7	9	4.7	9	4.7
3	6.2	3	6.2	3	6.2
1	8.6	1	8.6	1	8.6
5	10.8	5	10.8	5	10.8
7	13.4	7	13.4	7	13.4
4	14.9	4	14.9	4	14.9
8	17.3	8	17.3	8	17.3
6	18.8	6	18.8	6	18.8
Total cost	$18,338,800,606	Total cost	$12,049,723,799	Total cost	$10,995,199,180

Figure 7.7-3 shows how the objective function value (in terms of PV of total cost) decreases as the interest rate rises.

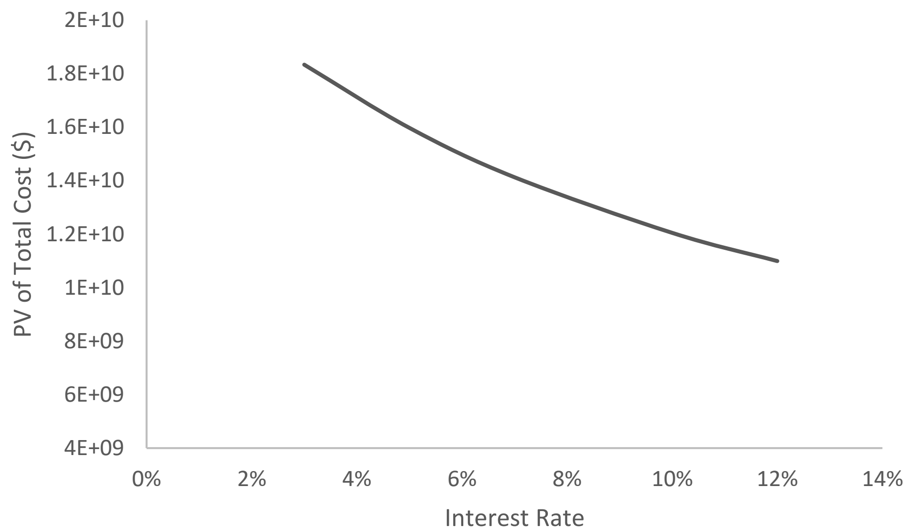


Figure 7.7-3 Objective Value for Different Interest Rates

7.7.2 Computation Time

The computation time, is a major factor in determining the feasibility and efficiency of any optimization algorithm. In general, for a large scheduling problem, a huge computation time is required if a mesoscopic model such as CTM is used model for project evaluations. Therefore, this section extensively investigates the sensitivity of computation time to different network configurations, CTM parameters, and problem size.

Problem size

Computation time is highly correlated with problem size and is of great concern. In general, computation time grows more than linearly as the problem size increases, thus important for investigation. In selection and scheduling projects, problem size is defined as the number of candidate projects for implementation. While increasing the problem size, the population size should also increase to guarantee sufficient exploration of the solution space. It should be noted that due to project interdependency, the computation time is also related to network configuration. More specifically, if the problem size is the same for different network sizes (i.e. number of nodes and links), the computation time needed to solve the problem will be different. However, in this section, a network with the same characteristics is tested in this section, and the only variable that changes is the problem size.

Table 7.7-5 and Figure 7.7-4 show the optimization results and the computation time for each problem size. It can be seen that the computation time increases exponentially as problem size grows.

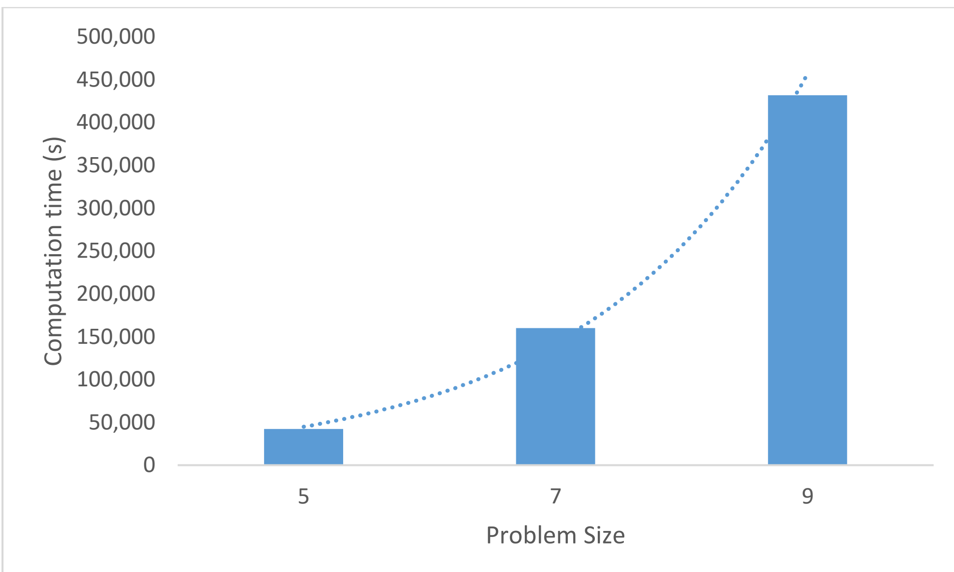


Figure 7.7-4 Computation time versus problem size

Table 7.7-5 Computation time

Problem size	***5***	***7***	***9***
# of generations	20	30	50
Population size	15	20	25
Computation time (s)	42,383	160,059	431,617

CTM time interval

The time interval (dt) is the time between two click ticks which is an important parameter that impacts the overall computation time. Each clock tick requires a specific set of calculations to be done for every cell and every link. For large networks, this compares well with the number of calculations for an iteration of the static equilibrium traffic assignment model, which is proportional to the number of the destinations but grows supralinearly with the number of nodes in network. Figure 7.7-6 shows how the computation time changes when the time interval is increased. Basically, for a fixed number of simulation cycles (40 cycles), the set of calculations decreases as the interval time increases. The figure shows the computation time for 40 intersection cycles. Note that the optimization algorithm runs CTM numerous times for different sequences throughout different generations. Therefore, the time interval greatly effects the optimization computation time as well. In the case study example, the time interval was set to 5 seconds.

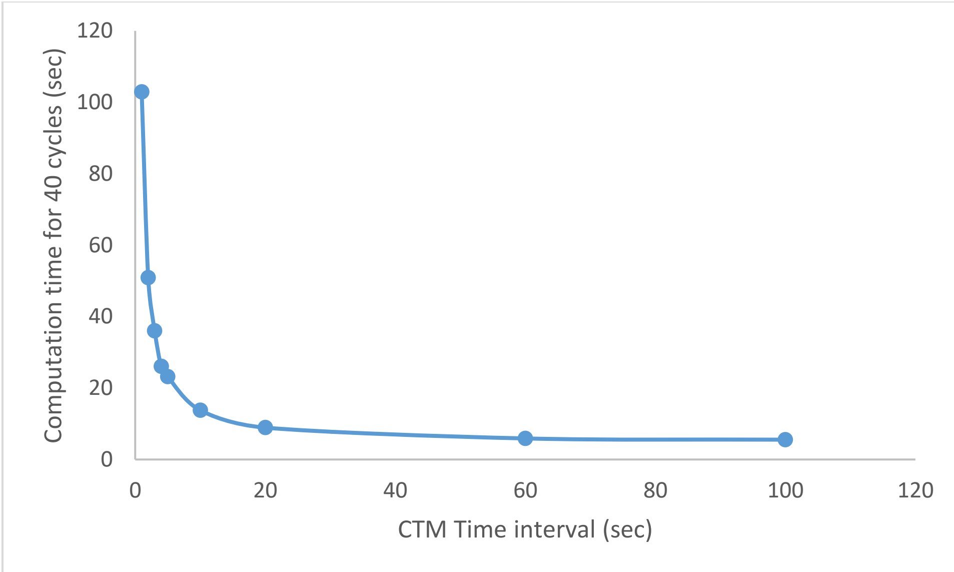


Figure 7.7-5 CTM Computation Time for Different Time Intervals

Cell length

In CTM introduced before, each link is divided into homogenous cells. The number of cells simply depends on the cell length. Intuitively if the cell length is short there are more cells in each link to consider at each clock tick. Therefore, it is expected that longer cell lengths would shorten the computation time. To demonstrate this, Figure 7.7-6 plots the CTM computation time (for 40 intersection cycles) subject to different cell lengths.

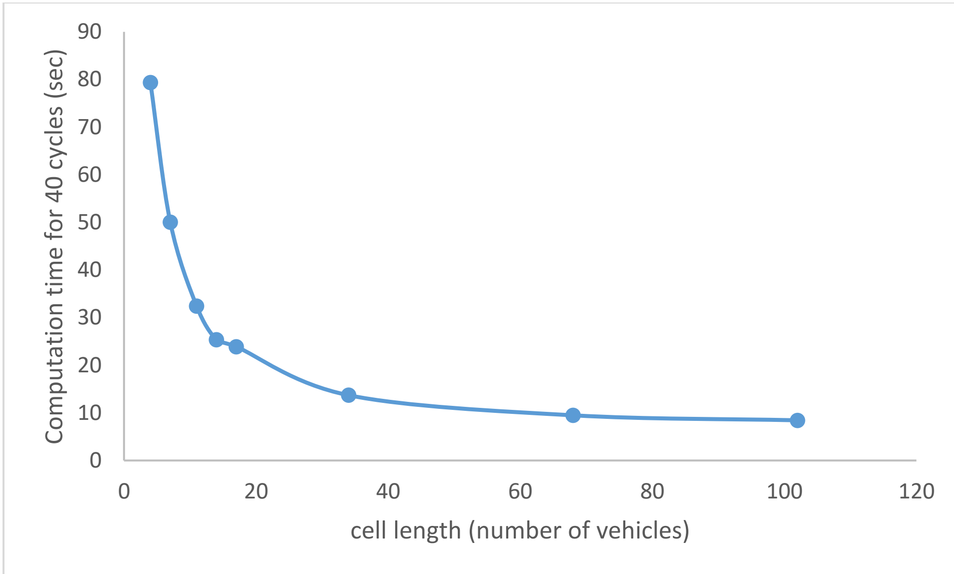


Figure 7.7-6 CTM Computation Time for Different Cell Lengths

7.8 Statistical Test

A statistical test as described in section 5.5 is applied to the genetic search. However, because CTM requires much more computation time than the traffic assignment model, a smaller sample of 3,000 random solutions is created and evaluated. After testing different distribution functions, the Normal (mu=1.242×10^{10}, sigma= 1.492×10^{8}) distribution is found to best fit the sample. The probability of finding a solution better than the GA solution (1.204×10^{10} from Table 7.6-1) can be calculated using the Cumulative Distribution Function (CDF) of the Normal Distribution: $p = F(x|\,\mu,\sigma) = F(1.204\times10^{10}|1.242\times10^{10}, 1.492\times10^{8}) = 6.2\times10^{-3}$.

This result implies that the probability of finding a solution better than the GA solution is extremely low, i.e. 6.2×10^{-3} . Therefore, the likelihood of finding a significantly better solution is negligible.

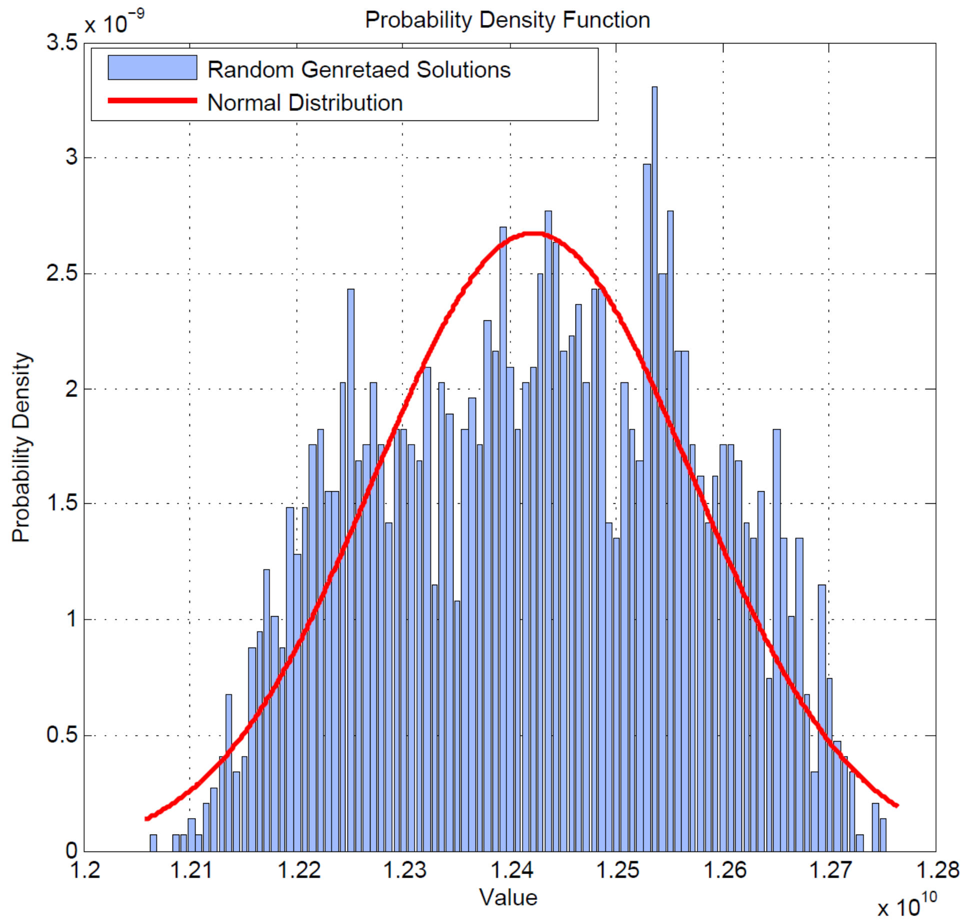


Figure 7.8-1 Objective function distribution of randomly generated solutions

Chapter 8 Conclusions and Recommendations

8.1 Conclusion

The goal of this study is to develop a general framework for selecting and scheduling interrelated alternatives. For this purpose, first, a simple traffic assignment and later a detailed Cell Transmission Model is developed to evaluate link and intersection improvements in a road network. With this model's detailed evaluation of project interrelations, capital investment planning can be properly conducted. Further, a reliable and efficient search approach using a genetic algorithm to optimize the selection and sequence of interrelated projects is presented. The model development and study results are summarized below:

8.1.1 Traffic assignment evaluation model

In the first part of this study a simple traffic assignment model (Frank-Wolf) is developed to evaluate improvement alternatives. Although traffic assignment is a basic evaluation model, it is a simple and quick way to test different features the algorithm. First, a stochastic objective function is introduced to solve the problem under three demand growth scenarios. The study then investigates the merit of the stochastic optimization compared to the deterministic one. It is shown that by considering demand uncertainties we can substantially improve the resulting objective function. Second, the model considers multiple improvements at each location. In fact, the algorithm selects potential locations for improvement and considers multiple improvement alternatives based on link conditions. For this purpose, a probabilistic procedure is introduced to select the optimal improvement at each location. Third, the algorithm is further developed to consider the possibility of adding new links in addition to expanding the capacities of existing links. Fourth, the model is further developed to account for vehicle operation and safety costs in addition to travel time costs. For this purpose, vehicle operating cost and safety cost functions are incorporated in the combined model. Their resulting fuel, tire, maintenance and repair costs, along with the cost of crashes in the system, are included in the objective function.

8.1.2 Cell Transmission Model

In the second part of this dissertation (chapters 6 and 7) a detailed Cell Transmission Model (CTM) is introduced and replaced the traffic assignment model. The goal of employing CTM is to effectively model traffic intersections and capture important traffic phenomena, such as queue build-up and dissipation, and backward propagation of congestion waves. The optimization model is further developed by considering intersection improvement alternatives in addition to link improvements and intersection delays in the objective function. Also, concurrent and precedence relation constraints are introduced in this section to restrict the relative implementation schedules between link and intersection improvements (i.e. intersection improvements should be scheduled simultaneously or prior to corresponding link improvements). The improved model is applied to a problem similar to that in the chapter 4. The optimized sequence and schedule, system performance, and sensitivity of results to important factors is provided in the rest of the study.

8.1.3 Genetic Algorithm

A genetic procedure performs numerous evaluations for the objective functions while exploring each generation. Especially with the CTM-based genetic search, problem size significantly affects the solution space and required evaluation time. Thus, establishing the termination rule is as an important factor. However, either specifying the number of generations or tracing the value of the objective function for a fixed number of generations cannot guarantee achieving global optimum. Due to limited computational resources, it is impossible to employ the genetic search on the entire feasible region even by starting with a sizable population or an enormous number of generations. Therefore, setting the best genetic parameters is essential to avoid local optimums and approaching the global optimum as closely as possible. Thus, a major limitation of genetic search is that global optimality is almost never guaranteed, and it is challenging to assess the goodness of solutions obtained by evolutionary methods. Therefore, for large problems where no globally optimal solution can be guaranteed, the goodness of the search can be verified with statistical tests which draws distribution of sample solutions and estimates probabilities of finding better solutions.

8.2 Applications

Although road and intersection expansion projects are the focus of this study, the proposed methodology should be applicable to general cases involving more complex systems. In fact, GAs can be effectively combined with any appropriate evaluation tool (e.g. microscopic simulation, simulation approximates, queuing or neural networks) specific to the problem, to solve the planning and scheduling problem for a variety of interrelated alternatives.

With a well-developed evaluation model, users can investigate the system according to their interests. For example, in case of congested urban networks, such as one presented in this study, CTM can be a good evaluation tool in assessing the system performance. The proposed CTM provides a good representation of the urban traffic with congested links and intersections. Any analysis of urban network improvements and the related sensitivity analysis could be easily conducted through the CTM-based genetic process. Unlike simulation models which are expensive and time consuming, CTM assesses the overall system performance while considering realistic traffic conditions such as queuing forming and dissipation. Furthermore, the model can be modified in terms of objective function and constraints to fit different application preferences.

8.3 Future Research

This study develops an approach for solving a combinatorial problem of interrelated project sequencing and scheduling through CTM and genetic search. However, considerable scope is left for future research.

- In addition to demand uncertainties, it would be interesting to explore supply uncertainties. In general, CTM, has a common assumption of a steady-state speed–density relationship which does not allow fluctuations around the equilibrium (nominal) fundamental flow–density diagram, and adopts a number of deterministic parameters (e.g. free-flow speed, jam density, capacity, etc.). However, research and empirical studies on the fundamental flow–density diagram have revealed that the fundamental flow–density diagram admits large variations (Figure 8.3-1). Recent microscopic and mesoscopic modeling approaches such as the Stochastic Cell Transmission Model (SCTM) (Sumalee et.al, 2011), which is an extension to CTM, model and interpret variations in the fundamental diagram. Future studies can incorporate such models into the selecting and scheduling problem to consider both demand and supply uncertainties.

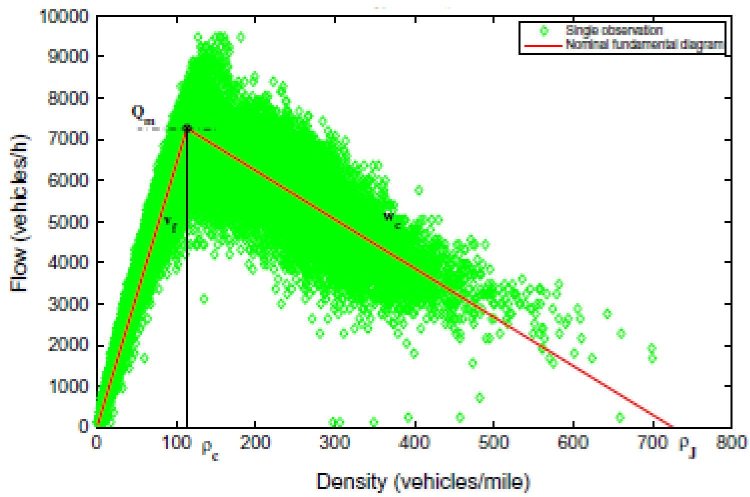


Figure 8.3-1 Variation in Fundamental Flow–Density Diagram

- As discussed in the previous section, the proposed optimization approach based on CTM-based genetic search requires burdensome computations of multiple generations. One way to tackle this problem is parallel processing. The main goal of parallel processing is to reduce the search time by using multiple processors concurrently. For a genetic algorithm, the fitness evaluation for individuals in the population are independent. Therefore, these tasks could be modeled as subtasks of the whole problem and different processors can be used concurrently. Integrating methods of parallel computing with the genetic algorithms and applying them in the project selection and scheduling problem remains a future task.
- Problem specific genetic operators may significantly enhance the performance of GAs. GA parameters for project sequencing actually depend on network characteristics. A general rule for determining GA parameters cannot be conclusively determined from a single network analysis. More test examples are needed to investigate the characteristics of the solution algorithms.

- Incorporate a hybrid optimization process to reduce the computation time. In general, for assessing the overall system performance a quick and computationally efficient model such as a traffic assignment can be used rather than a detailed and cumbersome model. After identifying critical locations, a more sophisticated model such as CTM or simulation can be used to understand the detailed traffic behaviors.
- Consider some more realistic characteristics, such changes over time at the same location, and traffic delays during construction.
- Instead of assuming some pre-specified improvement projects, it is useful to optimize and design improvement projects at each location. In other words, in addition to project sequencing and scheduling at multiple locations, the model also optimizes and designs improvement projects at each location.
- It is also desired to consider broader impacts such as travel time reliability, environmental effects and economic impacts in addition to the conventional direct impacts in the objective function. Future research may attempt to quantify such impacts and incorporate them in the optimization process.

Appendix

This Appendix provides more detailed information about the Cell Transmission Model components applied to Sioux Falls network. The information and examples provided in this section could be helpful for application on other networks.

A.1 CTM Link Table

The following table shows the detailed cell information for each link. Each link is divided to homogenous cells which are numbered consecutively from the *first* to *last* cell. Each link also includes an *input* and *output* cell which accommodates the exogenous flow going in and out the network from. These cells are numbered after the last cell in the link.

Table A.1-1 Cell Information for All Links

Link number	***sat_rate***	***input cell***	***output cell***	***first cell***	***last cell***	***number of cells***	***out_link***
1	1.2	52	51	1	50	50	1
2	1.2	88	87	53	86	34	50
3	1.2	140	139	89	138	50	83
4	1.2	173	172	141	171	31	132
5	1.2	201	200	174	199	26	162
6	1.2	238	237	202	236	35	187
7	1.2	274	273	239	272	34	221
8	1.2	302	301	275	300	26	254
9	1.2	319	318	303	317	15	279
10	1.2	343	342	320	341	22	293
11	1.2	360	359	344	358	15	314
12	1.2	384	383	361	382	22	328
13	1.2	402	401	385	400	16	349
14	1.2	426	425	403	424	22	364
15	1.2	441	440	427	439	13	385
16	1.2	474	473	442	472	31	397
17	1.2	492	491	475	490	16	427
18	1.2	512	511	493	510	18	442
19	1.2	542	541	513	540	28	459

20	1.2	560	559	543	558	16	486
21	1.2	580	579	561	578	18	501
22	1.2	595	594	581	593	13	518
23	1.2	625	624	596	623	28	530
24	1.2	633	632	626	631	6	557
25	1.2	651	650	634	649	16	562
26	1.2	665	664	652	663	12	577
27	1.2	683	682	666	681	16	588
28	1.2	701	700	684	699	16	603
29	1.2	709	708	702	707	6	618
30	1.2	735	734	710	733	24	623
31	1.2	766	765	736	764	29	646
32	1.2	780	779	767	778	12	674
33	1.2	795	794	781	793	13	685
34	1.2	819	818	796	817	22	697
35	1.2	850	849	820	848	29	718
36	1.2	881	880	851	879	29	746
37	1.2	918	917	882	916	35	774
38	1.2	942	941	919	940	22	808
39	1.2	973	972	943	971	29	829
40	1.2	991	990	974	989	16	857
41	1.2	1006	1005	992	1004	13	872
42	1.2	1021	1020	1007	1019	13	884
43	1.2	1039	1038	1022	1037	16	896
44	1.2	1057	1056	1040	1055	16	911
45	1.2	1077	1076	1058	1075	18	926
46	1.2	1095	1094	1078	1093	16	943
47	1.2	1113	1112	1096	1111	16	958
48	1.2	1128	1127	1114	1126	13	973
49	1.2	1136	1135	1129	1134	6	985
50	1.2	1156	1155	1137	1154	18	990
51	1.2	1167	1166	1157	1165	9	1007
52	1.2	1175	1174	1168	1173	6	1015
53	1.2	1190	1189	1176	1188	13	1020
54	1.2	1229	1228	1191	1227	37	1032
55	1.2	1249	1248	1230	1247	18	1068
56	1.2	1267	1266	1250	1265	16	1085
57	1.2	1285	1284	1268	1283	16	1100
58	1.2	1296	1295	1286	1294	9	1115
59	1.2	1317	1316	1297	1315	19	1123
60	1.2	1356	1355	1318	1354	37	1141
61	1.2	1379	1378	1357	1377	21	1177
62	1.2	1397	1396	1380	1395	16	1197
63	1.2	1414	1413	1398	1412	15	1212
64	1.2	1435	1434	1415	1433	19	1226

65	1.2	1446	1445	1436	1444	9	1244
66	1.2	1463	1462	1447	1461	15	1252
67	1.2	1486	1485	1464	1484	21	1266
68	1.2	1497	1496	1487	1495	9	1286
69	1.2	1517	1516	1498	1515	18	1294
70	1.2	1534	1533	1518	1532	15	1311
71	1.2	1543	1542	1535	1541	7	1325
72	1.2	1558	1557	1544	1556	13	1331
73	1.2	1582	1581	1559	1580	22	1343
74	1.2	1599	1598	1583	1597	15	1364
75	1.2	1608	1607	1600	1606	7	1378

A.2 Intersection CTM design

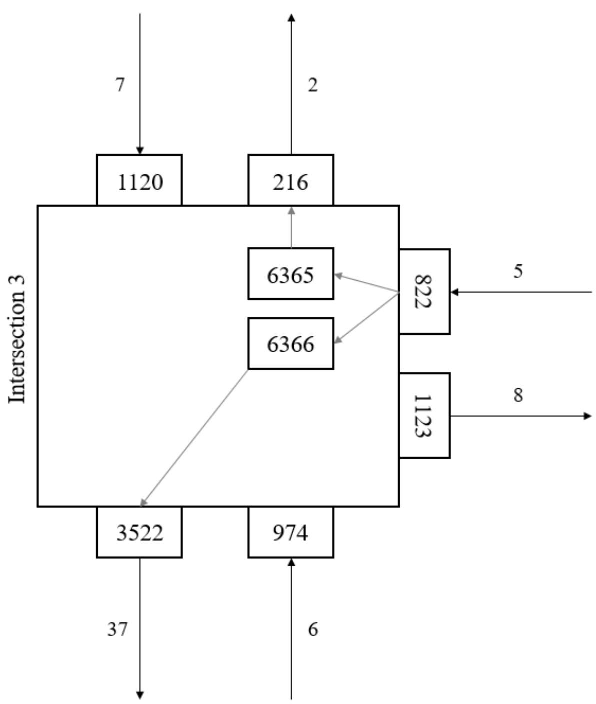
In this section an example of a three-leg and four-leg intersection is provided. First, Table A.2-1 shows the cell information including input, output and inner cell numbers for each intersection. Then an example of two-phase intersection is displayed. It is shown how links are set for East-West and North-South phases.

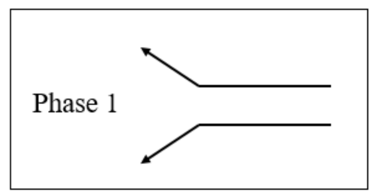
Table A.2-1 Cell Information for each Intersection

Intersection Number	Input-cells	Output-cells	Inner-cells
1	[213,359]	[362,977]	[6361,6362]
2	[574,707]	[1,1785]	[6363,6364]
3	[822,974,1120]	[1123,216,3522]	[6365,6366,6367,6368]
4	[1235,1300,1396]	[1399,710,3172]	[6369,6370,6371,6372]
5	[1461,1557,1628]	[1631,1238,2541]	[6373,6374,6375,6376]
6	[1724,1782,1915]	[1464,577,2335]	[6377,6378,6379,6380]
7	[1986,2063]	[2187,4890]	[6381,6382]
8	[2184,2255,2332,2390]	[1918,2393,1727,4530]	[6383,6384,6385,6386,6387,6388]
9	[2511,2538,2609]	[2066,1560,2806]	[6389,6390,6391,6392]
10	[2661,2732,2803,2830,2932]	[4374,3062,2514,4303]	[6393,6394,6395,6396,6397,6398]
11	[3059,3111,3169,3265]	[2612,3268,1303,4026]	[6399,6400,6401,6402,6403,6404]
12	[3392,3519,3671]	[2935,825,3770]	[6405,6406,6407,6408]
13	[3767,3894]	[6167,3395]	[6409,6410]
14	[3965,4023,4081]	[4084,3114,6109]	[6411,6412,6413,6414]
15	[4152,4223,4300,4371]	[4967,3897,2735,5934]	[6415,6416,6417,6418,6419,6420]
16	[4442,4500,4527,4604]	[4674,2664,2258,4647]	[6421,6422,6423,6424,6425,6426]

17	[4644,4671]	[4503,5109,2833]	[6427,6428]
18	[4729,4887,4964]	[4445,1989,5232]	[6429,6430,6431,6432]
19	[5035,5106,5146]	[4155,4607,5480]	[6433,6434,6435,6436]
20	[5229,5387,5477,5548]	[4732,5616,5038,5804]	[6437,6438,6439,6440,6441,6442]
21	[5613,5696,5736]	[5149,6263,5894]	[6443,6444,6445,6446]
22	[5801,5891,5931,6008]	[5390,6011,4226,5699]	[6447,6448,6449,6450,6451,6452]
23	[6073,6106,6164]	[5739,3968,6328]	[6453,6454,6455,6456]
24	[6260,6325,6358]	[5551,3674,6076]	[6457,6458,6459,6460]

Example of a three-leg intersection (intersection at node 3) is provided below. In phase 1, there are 2 inner cells, and 3 links (1 diverge + 2 normal links) which provides left-turn and right-turn movements. In phase 2, there are 4 inner cells, and 5 links (2 diverge + 1 merge + 2 normal links) which provides left-turn through and right-turn movements.

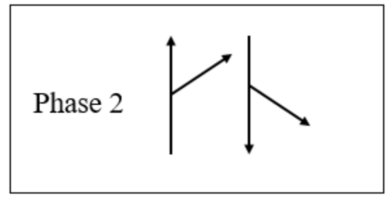


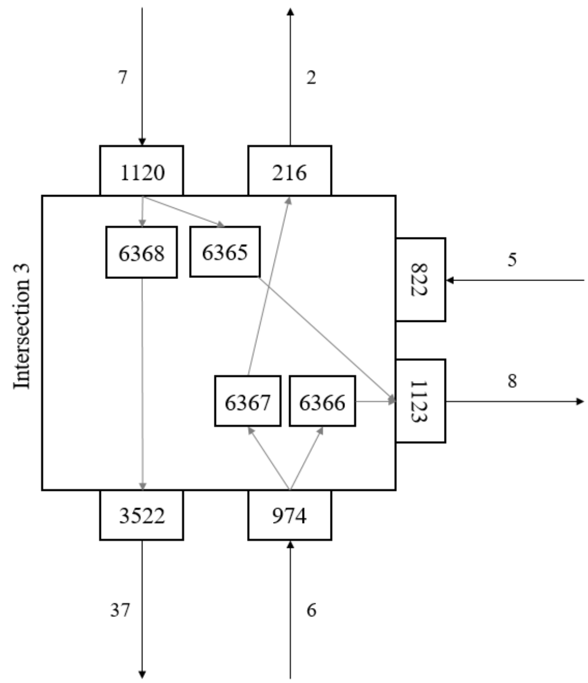


	1	2	3	4
Input (1)	822	974	1120	-
Output(2)	1123	216	3522	-
Inner(0)	6365	6366	6367	6368

Link type	1st cell type	Cell #	2nd cell type	Cell #	3rd cell type	Cell #
2	1	1	0	2	0	1
0	0	1	2	2	0	0
0	0	2	2	3	0	0

Figure A.2-1 Example of a Three-leg Intersection Design (Phase 1)



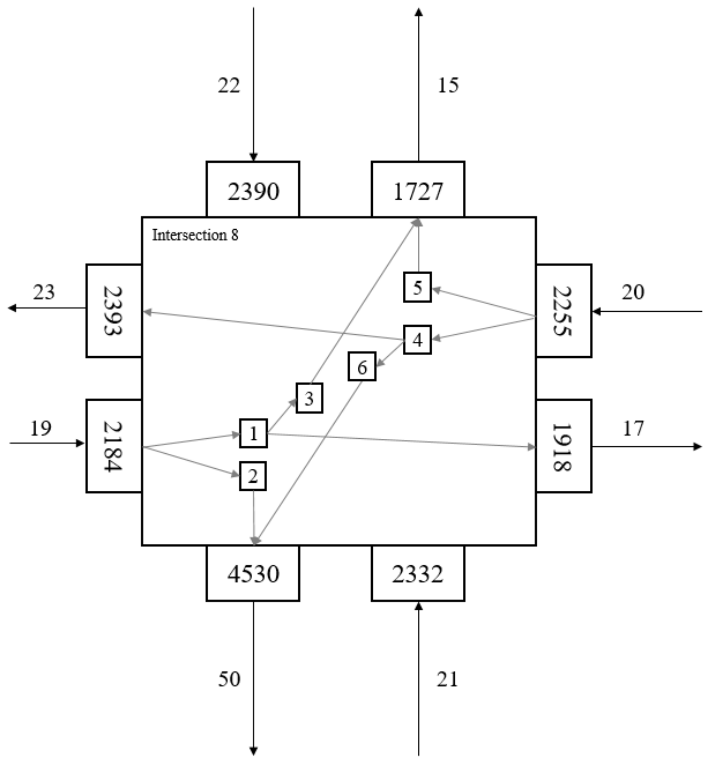


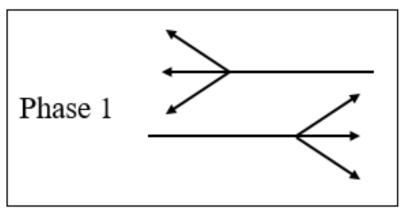
	1	2	3	4
Input (1)	822	974	1120	-
Output(2)	1123	216	3522	-
Inner(0)	6365	6366	6367	6368

Link type	1st cell type	Cell #	2nd cell type	Cell #	3rd cell type	Cell #
2	1	3	0	1	0	4
2	1	2	0	3	0	2
1	0	2	0	1	2	1
0	0	3	2	2	0	0
0	0	4	2	3	0	0

Figure A.2-2 Example of a Three-leg Intersection Design (Phase 2)

Example of a four-leg intersection (intersection at node 8) is provided below. In both phases, there are 6 inner cells, and 6 links (4 diverge + 2 merge links) which provides which provides left-turn through and right-turn movements.

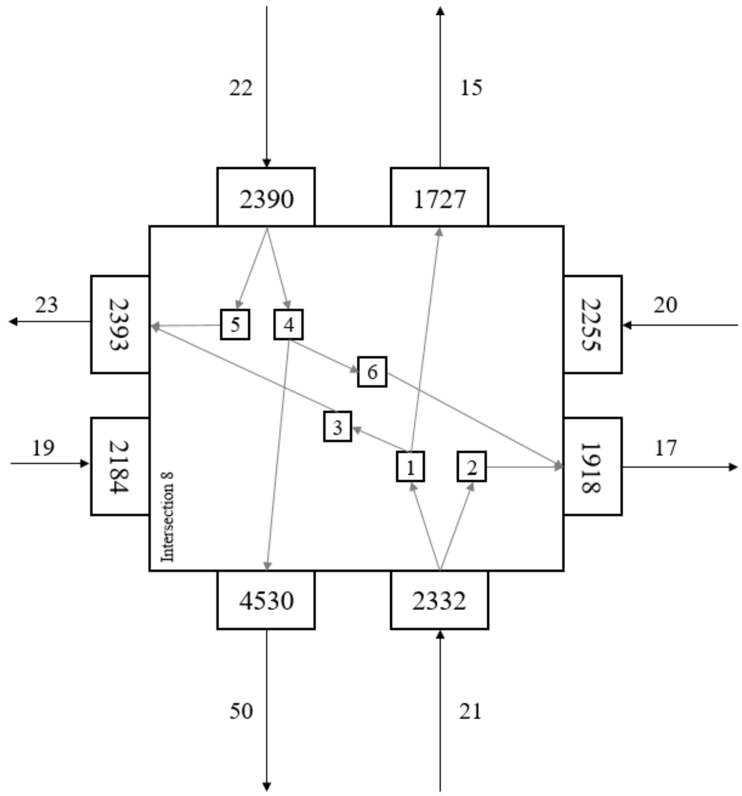


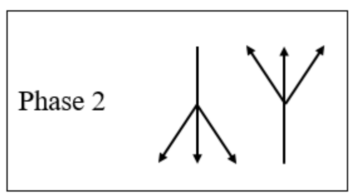


	1	2	3	4	5	6
Input (1)	2184	2255	2332	2390	-	-
Output(2)	1918	2393	1727	4530	-	-
Inner(0)	6383	6384	6385	6386	6386	6386

Link type	1st cell type	Cell #	2nd cell type	Cell #	3rd cell type	Cell #
2	1	1	0	1	0	2
2	0	1	2	1	0	3
2	1	2	0	4	0	5
2	0	4	2	2	0	6
1	0	2	0	6	2	4
1	0	5	0	3	2	3

Figure A.2-3 Example of a Four-leg Intersection Design (phase 1)





	1	2	3	4	5	6
Input (1)	2184	2255	2332	2390	-	-
Output(2)	1918	2393	1727	4530	-	-
Inner(0)	6383	6384	6385	6386	6386	6386

Link type	1st cell type	Cell #	2nd cell type	Cell #	3rd cell type	Cell #
2	1	3	0	1	0	2
2	0	1	2	3	0	3
2	1	4	0	4	0	5
2	0	4	2	4	0	6
1	0	2	0	6	2	1
1	0	5	0	3	2	2

Figure A.2-4 Example of a Four-leg Intersection Design (phase 2)

A.3 Intersection delay subject to demand change

Figure A.3-1 and Figure A.3-2 demonstrate how the intersection delay increases as demand grows. In these figures demand change is expressed as percentages of the original demand.

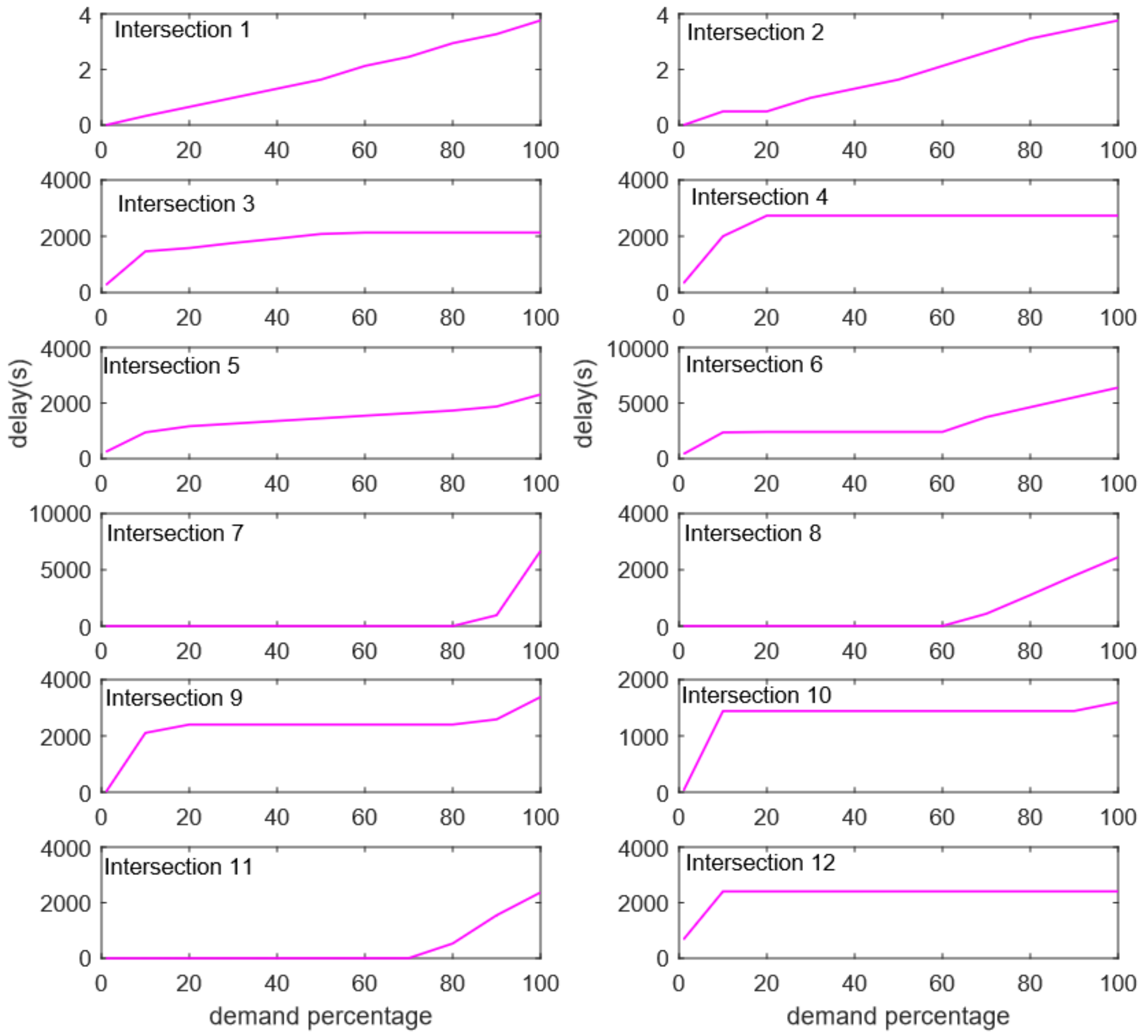


Figure A.3-1 Intersection Delay Change Based on Different Demand Levels (1-12)

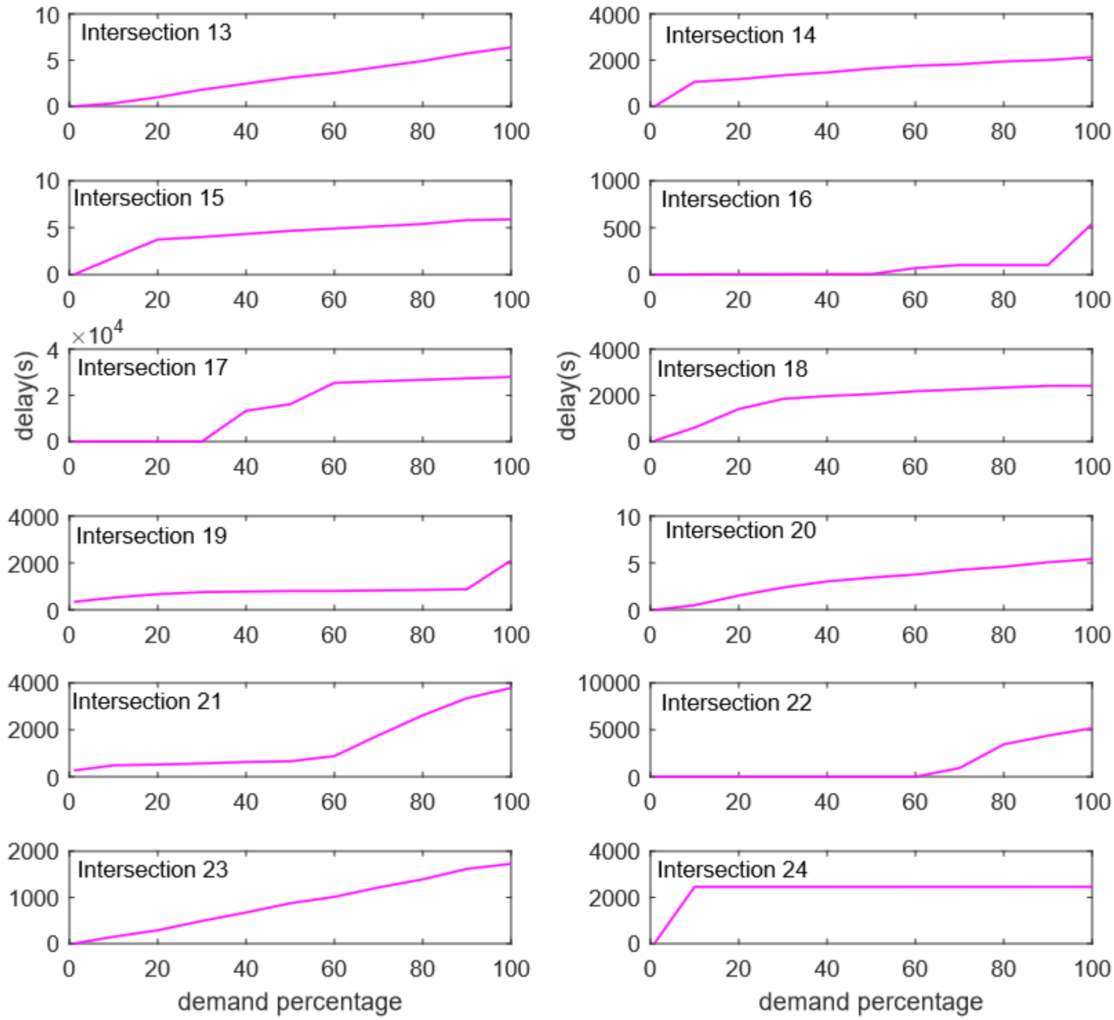


Figure A.3-2 Intersection Delay Change Based on Different Demand Levels (13-24)

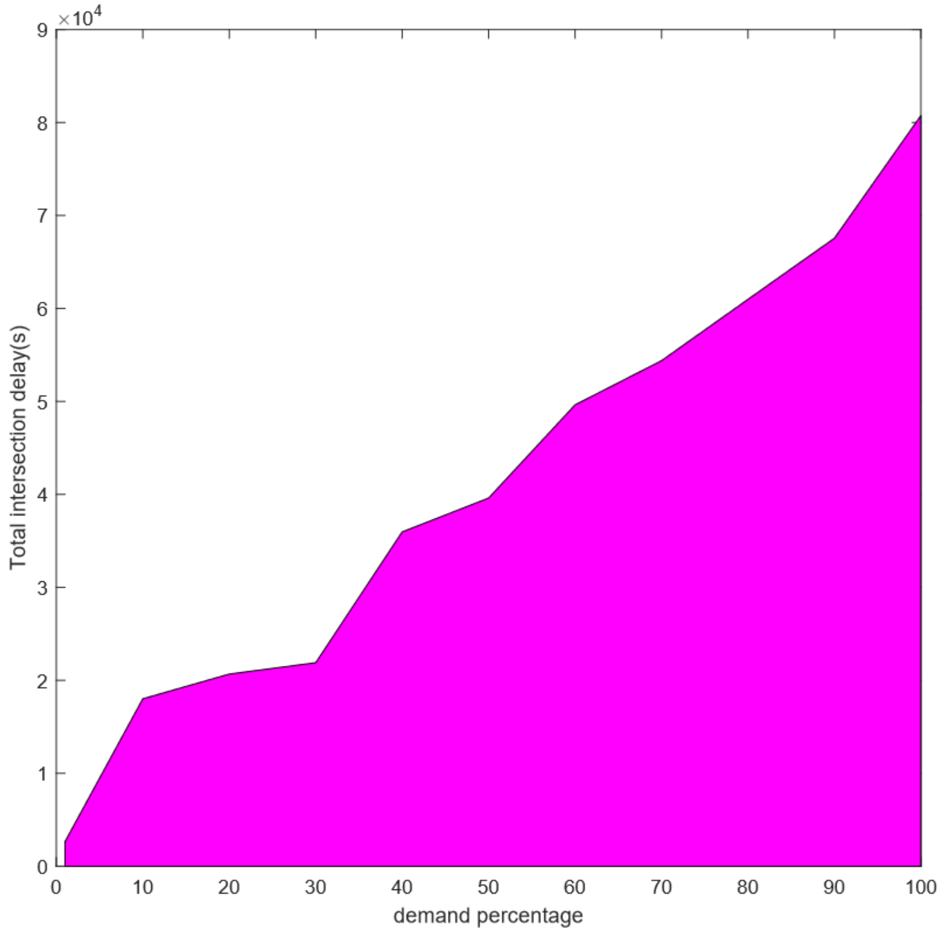


Figure A.3-3 Total Intersection Delay for Different Demand Levels

A.4 Link Density subject to demand change

Figure A.3-1 and Figure A.3-2 demonstrate how the intersection delay increases as demand grows.

In these figures demand changes are expressed as percentages of the original demand.

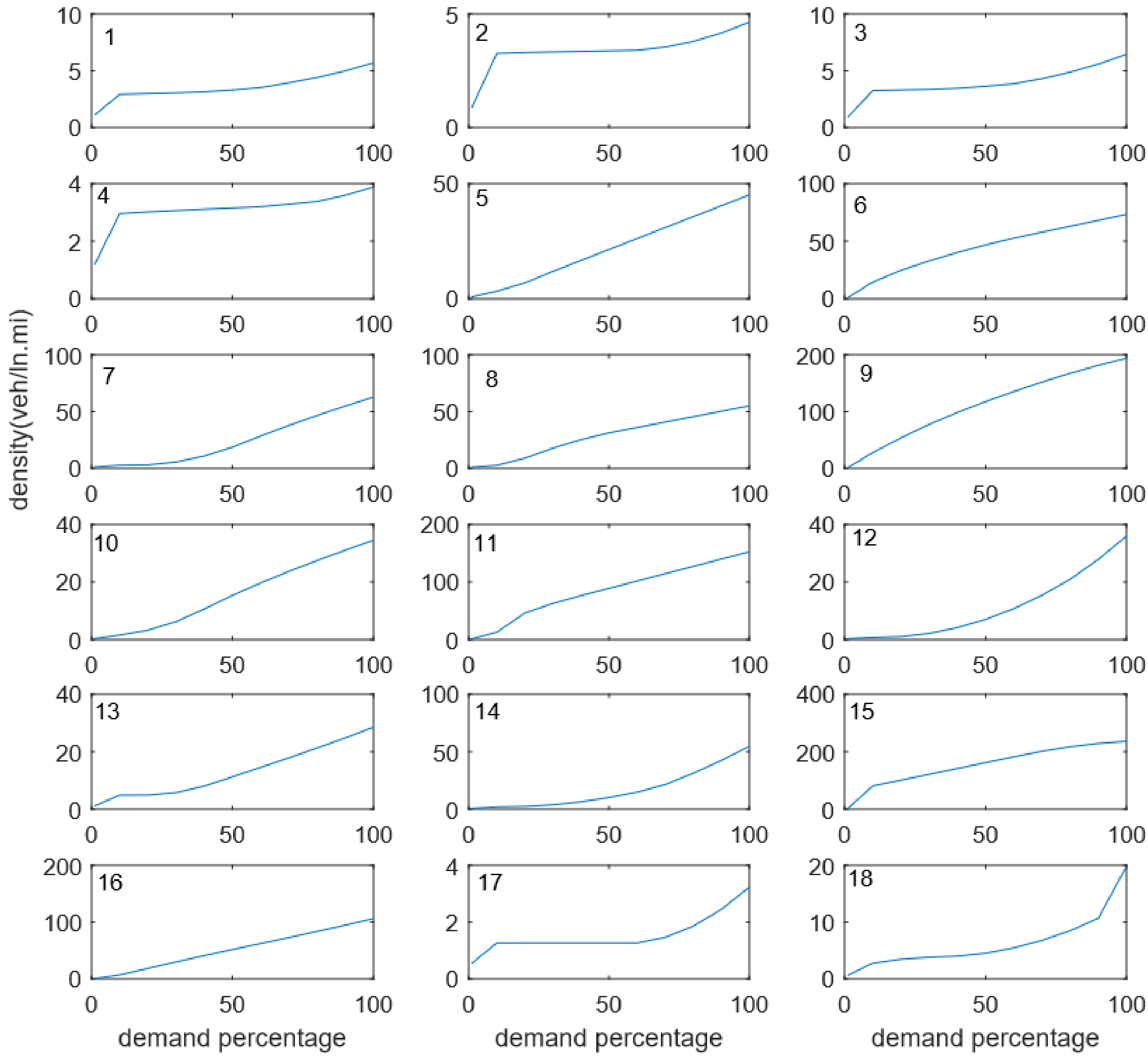


Figure A.4-1 Link Density Change Based on Different Demand Levels (1-18)

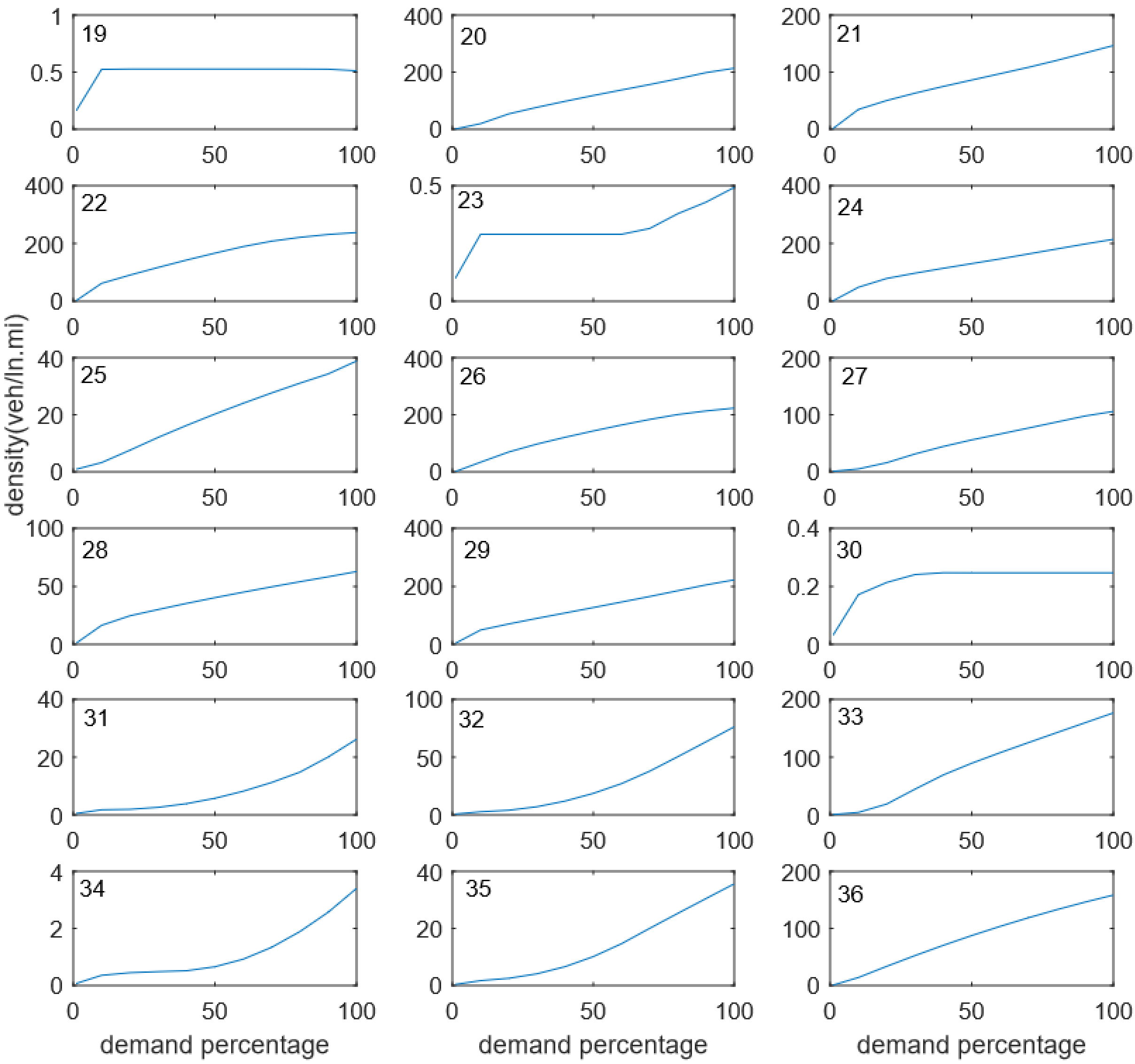


Figure A.4-2 Link Density Change Based on Different Demand Levels (19-36)

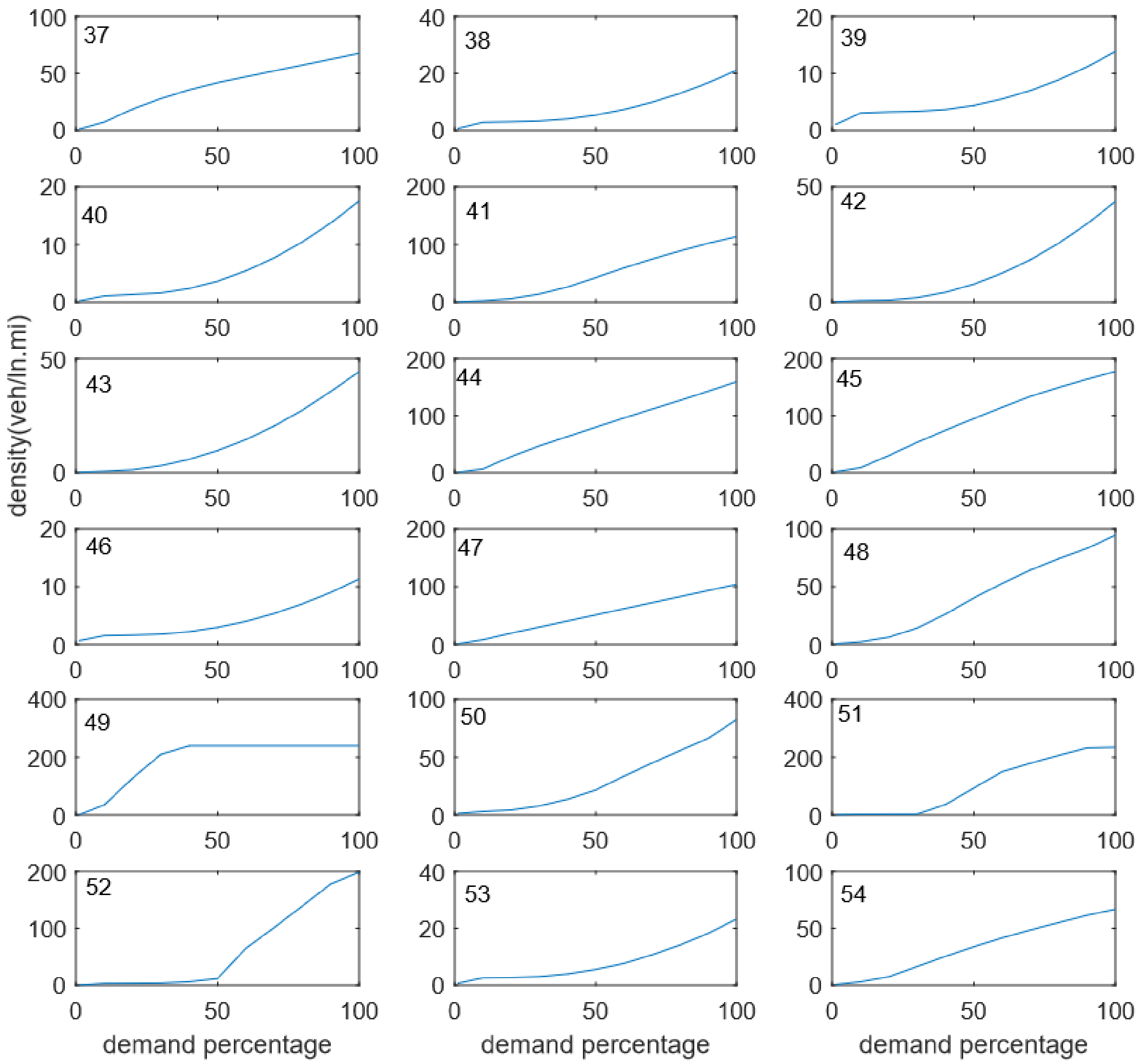


Figure A.4-3 Link Density Change Based on Different Demand Levels (37-54)

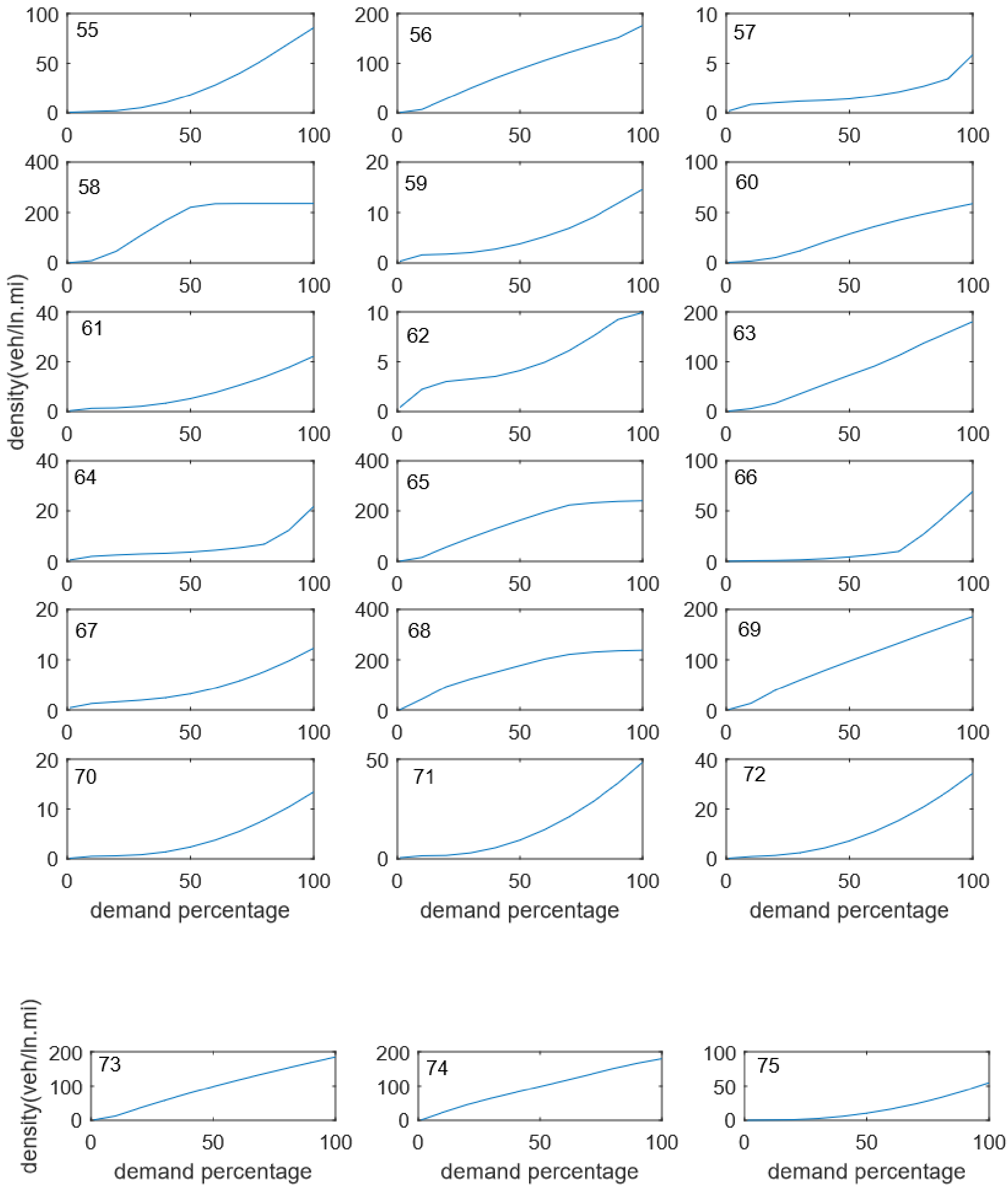


Figure A.4-4 Link Density Change Based on Different Demand Levels (55-75)

References

1. Bagloee, S. A., and Tavana, M. (2012) ‘An efficient hybrid heuristic method for prioritising large transportation projects with interdependent activities’, *International Journal of Logistics Systems and Management*, Vol.11 No. 1, pp.114-142.
2. Bagloee, S. A., and Asadi, M. (2015) ‘Prioritizing road extension projects with interdependent benefits under time constraint’, *Transportation Research Part A: Policy and Practice,* Vol.75, pp.196-216.
3. Balamurugan, K., Selladurai, V., and Ilamathi, B. (2006) ‘Solving unequal area facility layout problems using genetic algorithm’, *International Journal of Logistics Systems and Management*, Vol.2 No.3, pp.281-301.
4. Bar-Gera H. (2011). Transportation Network Test Problems.[online] http://www.bgu.ac.il/~bargera/tntp/ (Accessed 10 October 2016).
5. Bhattacharyya, R., Kumar, P., and Kar, S. (2011) ‘Fuzzy R&D Portfolio Selection of Interdependent Projects’, *Computers & Mathematics with Applications*, Vol. 62 No. 10, pp. 3857-3870.
6. Bouleimen, K., and Lecocq, H. (2003) ‘A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version’, *European Journal of Operational Research*, Vol.149 No.2, pp. 268-281.
7. Bureau of Public Roads (1964) *Traffic Assignment Manual*. Urban Planning Division, US Department of Commerce, Washington, DC.
8. Chen, X., Zhu, Z., He, X., and Zhang, L. (2015) ‘Surrogate-based optimization for solving mixed integer network design problem’, *Transportation Research Record: Journal of the Transportation Research Board,* Vol. 2497, pp.124-134.

9. Clark, J. J., Hindelang, T. J., and Pritchard, R. E. (1989). "Capital budgeting: planning and control of capital expenditures." Prentice Hall.

10. Cochran, M. A., Pyle, E. B., Greene, L. C., Clymer, H. A., and Bender, A. D. (1971). "Investment model for R and D project evaluation and selection." *IEEE T. Eng. Manage.*, (3), 89.

11. Crowder, H. P., Johnson, E. L., and Padberg, M. W. (1983). Solving Large-Scale Zero-One Linear Programming Problems. *Operations Research*, Vol. 31, 1983, pp. 803–834.

12. Cruz, L., Fernandez, E., Gomez, C., Rivera, G., and Perez, F. (2014) 'Many-objective portfolio optimization of interdependent projects with 'a priori' incorporation of decision-maker preferences', *Applied Mathematics & Information Sciences*, Vol. 8 No. 4, pp.1517-1531.

13. Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, *28*(4), 269-287.

14. Daganzo, C. F. (1995). The cell transmission model, part II: network traffic. *Transportation Research Part B: Methodological*, *29*(2), 79-93.

15. Dai, D.M., and Schonfeld, P. (1998) 'Metamodels for estimating delays through series of waterway queues', *Transportation Research Part B: Methodological*, Vol. 32 No.1, pp. 1-19.

16. Dickinson, M. W., Thornton, A. C., and Graves, S. (2001) 'Technology portfolio management: optimizing interdependent projects over multiple time periods', *IEEE Transactions on Engineering Management*, Vol. 48 No.4, pp.518-527.

17. Disatnik, D. J., and Benninga, S. (2007) ‘Shrinking the covariance matrix’, *The Journal of Portfolio Management*, Vol. 33, No. 4, pp. 55-63.

18. Dueñas-Osorio, L., Craig, J. I., Goodno, B. J., and Bostrom, A. (2007) ‘Interdependent response of networked systems’, *Journal of Infrastructure Systems*, Vol.13, No. 3, pp. 185-194.

19. Durango-Cohen, P. L., and Sarutipand, P. (2007) ‘Capturing interdependencies and heterogeneity in the management of multifacility transportation infrastructure systems’, *Journal of Infrastructure Systems*, Vol.13, No. 2, pp. 115-123.

20. Davis, L. (1985), “Applying Adaptive Algorithms to Domains”, Proceedings of International Joint Conference on Artificial Intelligence, pp.162-164.

21. De Neufville, R., Scholtes, S. (2011). Flexibility in Engineering Design. MIT Press.

22. Erlenkotter, D. (1973). “Sequencing of interdependent hydroelectric projects.” *Water Resour. Res.*, *9*(1), 21-27.

23. Frank, M., and Wolfe, P. (1956) ‘An algorithm for quadratic programming’, *Naval Research Logistics Quarterly*, Vol. 3, No. 1, pp. 95-110.

24. Goldberg, D. and R. Lingle (1985), “Alleles, Loci and the Traveling Salesman Problem”, Proceedings of the First International Conference on Genetic Algorithms, Lawrnece Erlbaum Associates, Hillsdale, NJ, pp.154-159.

25. Haq, A. N., and Kannan, G. (2006) ‘Two-echelon distribution-inventory supply chain model for the bread industry using genetic algorithm’, *International Journal of Logistics Systems and Management*, Vol. 2 No. 2, pp.177-193.

26. Hartmann, S., & Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research, 207(1), 1-14.

27. Highway Capacity Manual (2010). Washington, D.C., Transportation Research Board.

28. Highway Safety Manual (2010). AASHTO, Washington, D.C.

29. Highway Economic Requirements System- State Version. (2005), Technical report, U.S.

30. Jong, J. C., and Schonfeld, P. (2001) 'Genetic algorithm for selecting and scheduling interdependent projects', *Journal of Waterway, Port, Coastal, and Ocean Engineering*, Vol.127 No. 1, pp. 45-52.

31. Koziel, S. & Yang, X.-S. (Eds.) (2011). Computing Optimization, Methods and Algorithms, SCI 356, pp. 33-59.

32. 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, pp. 309-318.

33. Lee, J. W., and Kim, S. H. (2000). Using analytic network process and goal programming for interdependent information system project selection. Computers & Operations Research, 27(4), 367-382.

34. Li, Z., Roshandeh, A. M., Zhou, B., and Lee, S. H. (2013) 'Optimal decision making of interdependent tollway capital investments incorporating risk and uncertainty', *Journal of Transportation Engineering*, Vol. 139 No.7, pp. 686-696.

35. Li, X., Fang, S. C., Guo, X., Deng, Z., and Qi, J. (2016) 'An extended model for project portfolio selection with project divisibility and interdependency' *Journal of Systems Science and Systems Engineering*, Vol. 25 No. 1, pp. 119-138.

36. Lombardi, M., & Milano, M. (2012). Optimal methods for resource allocation and scheduling: a cross-disciplinary survey. Constraints, 17(1), 51-85.

37. Lorie, J. H., and Savage, L. J. (1955) Three Problems in Rationing Capital. *Journal of Business*, pp. 679–694.

38. Machado-León, J. L., de Oña, R., Baouni, T., and de Oña, J. (2017) 'Railway transit services in Algiers: priority improvement actions based on user's perceptions', *Transport Policy*, 53, pp. 175-185.

39. Markowitz, H. (1952) 'Portfolio selection', *The Journal of Finance*, Vol.7 No.1, pp. 77-91.

40. Martello, S., and Toth, P. (1990) *Knapsack Problems: Algorithms and Computer Implementations.* John Wiley and Sons, Chichester, England, pp. 189–220.

41. Martinelli, D. (1993). Investment planning of interdependent waterway improvement projects, Ph.D. Dissertation, University of Maryland, College Park.

42. Martinelli, D., & Schonfeld, P. (1995). Approximating delays at interdependent locks. Journal of waterway, port, coastal, and ocean engineering, 121(6), 300-307.

43. Michalewicz, Z., and Janikow, C. Z. (1991) 'Genetic algorithms for numerical optimization', *Statistics and Computing*, Vol. 1 No. 2, pp. 75-91.

44. Michalewicz, Z. (1995) *Genetic algorithms + data Structure = evolution programs*, Springer.

45. Mika, M., Waligóra, G., and Węglarz, J. (2005). "Simulated annealing and tabu search for multi-mode resource-constrained project scheduling with positive discounted cash flows and different payment models." *Eur. J. Oper. Res.*, 164(3), 639-668.

46. Mollanejad, M., and Zhang, L. (2014). “Incorporating Equity into Interurban Road Network Design.” *Transport Geography*, 39: 156-164

47. Morin, T. L., and Esogbue, A. M. (1971)."Some efficient dynamic programing algorithms for the optimal sequencing and scheduling of water supply projects." *Water Resour. Res.* 7(3) 479-48

48. Morin, T. L., and Esogbue, A. M. (1974). “A useful theorem in the dynamic programing solution of sequencing and scheduling problems occurring in capital expenditure planning.” *Water Resour. Res.*, 10(1), 49-50.

49. Nemhauser, G. L., and Ullman, Z. (1969) Discrete Dynamic Programming and Capital Allocation. *Management Science,* Vol. 15, No. 9, pp. 494–505.

50. Quadros, S. G. R., and Nassi, C. D. (2015) ‘An evaluation on the criteria to prioritize transportation infrastructure investments in Brazil’, *Transport Policy,* Vol. 40, pp. 8-16.

51. Rebiasz, B., Gaweł, B., and Skalna, I. (2014) ‘Capital budgeting of interdependent projects with fuzziness and randomness’, *Information systems architecture and technology, Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej*, pp. 125-135.

52. Roess, R. P., Prassas, E. S., & McShane, W. R. (2011). *Traffic engineering*.

53. Sandhu, M. (2006) ‘Project logistics with the dependency structure matrix approach–an analysis of a power plant delivery’ *International Journal of Logistics Systems and Management,* Vol.2 No.4, pp.387-403.

54. Sadeghi, A., and Moghaddam, A. M. (2016) ‘Uncertainty-based prioritization of road safety projects: An application of data envelopment analysis’, *Transport policy,* Vol. 52, pp. 28-36.

55. Savage, S. L., and Markowitz, H. M. (2009). *The flaw of averages: Why we underestimate risk in the face of uncertainty*. John Wiley & Sons.

56. Shayanfar, E., Abianeh, A. S., Schonfeld, P., and Zhang, L. (2016) ‘Prioritizing interrelated road projects using meta-heuristics’, *Journal of Infrastructure Systems*, Vol.22 No.2, 04016004.

57. Sumalee, A., Zhong, R. X., Pan, T. L., & Szeto, W. Y. (2011). Stochastic cell transmission model (SCTM): A stochastic dynamic traffic model for traffic state surveillance and assignment. Transportation Research Part B: Methodological, 45(3), 507-533.

58. Syswerda, G. (1991), “Scheduling Optimization Using Genetic Algorithms”, Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York, pp.824-829.

59. Szimba, E., and Rothengatter, W. (2012) ‘Spending scarce funds more efficiently—including the pattern of interdependence in cost-benefit analysis’, *Journal of Infrastructure Systems*, Vol.18 No. 4, pp. 242-251.

60. Tao, X., and Schonfeld, P. (2005) ‘Lagrangian relaxation heuristic for selecting interdependent transportation projects under cost uncertainty’, *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1931, pp.74-80.

61. Tao, X., and Schonfeld, P. (2006) ‘Selection and scheduling of interdependent transportation projects with island models’, *Transportation Research Record: Journal of the Transportation Research Board*, Vol.1981, pp.133-141.

62. Tao, X., and Schonfeld, P. (2007) ‘Island models for a stochastic problem of transportation project selection and scheduling’, *Transportation Research Record: Journal of the Transportation Research Board,* Vol. 2039, pp.16-23.

63. Ting, C. J., and Schonfeld, P. (1998). Optimization through simulation of waterway transportation investments. Transportation Research Record: Journal of the Transportation Research Board, 1620(1), 11-16.

64. Tofighian, A. A., and Naderi, B. (2015) ‘Modeling and solving the project selection and scheduling’, *Computers & Industrial Engineering*, Vol.83, pp.30-38.

65. Ukkusuri, S. V., & Waller, S. T. (2008). Linear programming models for the user and system optimal dynamic network design problem: formulations, comparisons and extensions. *Networks and Spatial Economics*, *8*(4), 383-406.

66. Vilkkumaa, E., Liesiö, J., & Salo, A. (2014). Optimal strategies for selecting project portfolios using uncertain value estimates. European Journal of Operational Research, 233(3), 772-783.

67. Wang, S. L. (2001). “Simulation and optimization of interdependent waterway improvement projects.” PhD dissertation, Univ. of Maryland, College Park, Md.

68. Wang, S. L., and Schonfeld, P. (2008) ‘Scheduling of waterway projects with complex interrelations’, *Transportation Research Record: Journal of the Transportation Research Board,* Vol. 2062, pp.59-65.

69. Wang, S. L., and Schonfeld, P. (2005) ‘Scheduling interdependent waterway projects through simulation and genetic optimization’, *Journal of Waterway, Port, Coastal, and Ocean Engineering*, Vol.131, No. 3, pp. 89-97.

70. Wang, S. L., & Schonfeld, P. (2012). Simulation-based scheduling of mutually exclusive projects with precedence and regional budget constraints. Transportation Research Record: Journal of the Transportation Research Board, 2273(1), 1-9.

71. Wei, C.H., and Schonfeld, P. (1994) ‘Multi-period network improvement model’, *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1443, pp.110-118.9

72. Weingartner, H. M. (1966). “Capital budgeting of interrelated projects: survey and synthesis.” *Manage. Sci.*, 12(7), 485-516.

73. Whitley, D., Starkweather, T., and Fuquay, D’A. (1989) , “Scheduling Problems and Traveling Salesman: The Genetic Edge Recombination Operator”, Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmanm Publishers, San Mateo, CA, pp. 133-140.

74. Xu, W., Long, Y., and Zhang W. (2017) ‘Prioritizing future funding and construction of the planned high-speed rail corridors of China–According to regional structure and urban land development potential indices, *Transport Policy*.

75. Yang, N., Schonfeld, P., & Kang, M. W. (2009). A hybrid methodology for freeway work-zone optimization with time constraints. Public Works Management & Policy, 13(3), 253-264.

76. Zhang, L., Ji, M., and Ferrari, N. (2013) *Comprehensive Highway Corridor Planning with Sustainability Indicators* (Final Report), Maryland State Highway Administration.

77. Zhu, L., Schonfeld, P., Kim, Y. M., Flood, I., and Ting, C. J. (1999). Queuing network analysis for waterways with artificial neural networks. AI EDAM, 13(5), 365-375.

78. Ziliaskopoulos, A. K. (2000). A linear programming model for the single destination system optimum dynamic traffic assignment problem. *Transportation science*, *34*(1), 37-49.