The Dissertation Committee for Shoupeng Tang
certifies that this is the approved version of the following dissertation:

Network Routing and Equilibrium Models for Urban Parking Search

Committee:

Stephen D. Boyles, Supervisor

C. Michael Walton

Randy B. Machemehl

John Hasenbein

Avinash Unnikrishnan

Network Routing and Equilibrium Models for Urban Parking Search

by

Shoupeng Tang, B.E., M.E.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2014

Dedicated to my wife Nan and My son Jayden

Acknowledgments

I wish to first thank my supervisor Dr. Stephen Boyles who is always there providing ideas, suggestions, and any other help I need. He has led me to a deep level of understanding the parking problems and the methods to model it and solve it.

I would also thank my committee members, Dr. C. Michael Walton, Dr. Randy B. Machemehl, Dr. John Hasenbein and Dr. Avinash Unnikrishnan. They gave me many very helpful suggestions and comments which greatly improved my dissertation, and have always been willing to meet with me when needed. This research reported in this dissertation was supported by the National Science Foundation (grant nos. 1069141/1157294 and 1254921) as well as the Data-Supported Transportation Planning and Operations University Transportation Center, and I am grateful for their support.

Thank my parents, Lingju Kong and Dianqin Tang, for their support for my learning abroad. They are always there when ever and where ever I need help or advise.

Thank Tarun and Reese who helped me a lot with the literature reviews and algorithm codings.

Thank all my friends and colleagues for their help with my research and the happiness they brought to my life.

Network Routing and Equilibrium Models for Urban Parking Search

by

Shoupeng Tang, Ph.D.
The University of Texas at Austin, 2014

Supervisor: Stephen D. Boyles

This dissertation focuses on modeling parking search behavior in traffic assignment models. Parking contributes greatly to urban traffic congestion. When the parking supply is scarce, it is very common for a vehicle to circle around for a considerable period just for an open parking spot. This circling or "cruising" add additional traffic flow onto the network. However, traditional traffic assignment models either ignore parking completely or simply treat it in limited ways. Most traffic assignment models simply assume travelers just directly drive from their origin to their destination without considering the parking search behavior. This would result in a systematic underestimation of road traffic flows and congestion which may mislead traffic managers to give inappropriate planning or control strategies. Models which do incorporate parking effects either constrain their implementation in limited small networks or ignore the stochasticity of parking choice by drivers.

This dissertation improves upon previous research into network parking modeling, explicitly capturing drivers' behavior and stochasticity in the parking search process, and is applicable to general networks. This dissertation constructs three types of parking search models. The first one is to model a single driver's parking search process, taking into account the likelihood of finding parking in different locations from past experience as well as observations gained during the search itself. This model uses the *a priori* probability of finding parking on a link, which reflects the average possibility of finding a parking space based on past experience. This probability is then adjusted based on observations during the current search. With these concepts, the parking search behavior is modeled as a Markov decision process (MDP). The primary contribution of the proposed model is its ability to reflect history dependence which combines the advantages of assuming “full reset” and “no reset” . “Full reset” assumes the probability of finding a parking space on a link is independent of any observations in the current search, while “no reset” assumes the state of parking availability is completely determined by past observations, never changing once observed. For instance, assume that the *a priori* probability of finding parking on a link is 30%. “Full reset” implies that if a driver drives on this link and sees no parking available, if he or she immediately turns around and drives on the link again, the probability of finding parking is again 30% independent of the past observation. By contrast, “no reset” implies that if a parking space is available on a link, it will always be available to return to in the future at any point. This dissertation develops an “asymptotic reset” principle which generalizes these principles and allows past observations to affect the probability of finding parking on a link and this impact weakens as time goes by. Both full reset and no reset are shown to be special cases of asymptotic reset.

The second problem is modeling multiple drivers through a parking search equilibrium on a static network. Similar to the first type of problem, drivers aim to minimize their total travel costs. Their driving and parking search behaviors depend on the probabilities of finding parkings at particular locations in the network. On the other side, these probabilities depend on drivers' route and parking choices. This mutual dependency can be modeled as an equilibrium problem. At the equilibrium condition no driver can improve his or her expected travel cost by unilaterally changing his or her routing and parking search strategy. To accomplish this, a network transformation is introduced to distinguish between drivers searching for parking on a link and drivers merely passing through. The dependence of parking probability on flow rates results in a set of nonlinear flow conservation equations. Nevertheless, under relatively weak assumptions the existence and uniqueness of the network loading can be shown, and an intuitive "flow-pushing" algorithm can be used to solve for the solution of this nonlinear system. Built on this network loading algorithm, travel times can be computed. The equilibrium is formulated as a variational inequality, and a heuristic algorithm is presented to solve it. An extensive set of numerical tests shows how parking availability and traffic congestion (flows and delays) vary with the input data.

The third problem aims at developing a dynamic equivalent for the network parking search equilibrium problem. This problem attempts to model a similar set of features as the static model, but aims to reflect changes in input demand, congestion, and parking space availability over time. The approach described in the dissertation is complementary to the static approach, taking on the flavor of simulation more than mathematical formulation. The dynamic model augments the cell transmission model with additional state variables to reflect parking availability, and integrates this network loading with an MDP-based parking search behavior model.

Finally, case studies and sensitivity analysis are taken for each of the three models. These analyses demonstrate the models' validity and feasibility for practice use. Specifically, all the models show excess travel time and flow on the transportation networks because of taking into account the "parking search cruising" and can show the individual links so affected. They all reflect the scattered parking distribution on links while traditional traffic assignment models only assign vehicles onto specified destination nodes.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Transportation plays a critical role in our daily life. It not only transports people to their destinations for work, entertainment, education, and shopping purposes but also helps deliver critical goods such as food and fuel. Despite the importance of the transportation system, it is also associated with congestion, pollution, noise, and other negative impacts. Congestion is a root cause for many of these negative impacts: for instance, excess congestion causes worse pollution as vehicles idle in stopped traffic.

The direct reason that the congestion is becoming more common and longer-lasting in most cities is that the increase in vehicle demand is not associated with a corresponding increase in infrastructure. This is due to several reasons, including financial constraints, environmental impacts, and availability of space in crowded cities. Another reason for congestion is that current transportation system is not running efficiently or in an optimum condition. For instance, signal timing plans may need to be improved, or the travelers' information system can do much more than it does now.

Parking issues play a very important role in traffic congestions, especially at locations in downtown areas, near universities, or near popular entertainment destinations. In urban centers where space is limited and cars are prevalent, parking shortages cause drivers to “cruise” for parking, a frustrating endeavor both for drivers and city planners. Clearly, there is a need for city planners to better manage parking in their jurisdictions. While recent programs in San Francisco, Boston, Seattle, Washington D.C., and elsewhere can provide much-needed data, sophisticated and accurate parking models are needed to help forecast the effects of changes in capacity, pricing, and strategies aimed at reducing the “cruising traffic”. Similarly, more accurate parking guidance systems would would benefit from the models. With current models for parking, it is difficult to quantify the impact of cruising on delay and congestion even in the present state, let alone quantifying the benefits of proposed policies or technologies.

These policy and technological strategies for improving parking-related congestion can be classified into two types: supply-side and demand-side. Supply causes mainly refer to the capability of providing transportation infrastructure and services by the transportation systems, while demand causes refer to the number of trips or other things that will use the transportation systems. Supply-side strategies could include providing additional parking, adjusting the price of parking (even in real-time based on availability), adjusting the duration of allowable parking. Demand-side strategies would attempt to reduce the travel demand or shift part of the demand from a mode which requires parking search to another mode. For instance, encourage more people to work at home will help reduce travel demand while provide better transit service and attract people from driving a car to the transit can be also very helpful for reducing parking-related congestion. There are also measures we can take to both improve the supply-side and the demand-side. For instance, one can monitor parking availability in real time and disseminate this information to drivers via roadside signs or mobile apps. This thesis focuses primarily on the modeling required to support supply-side strategies. Nevertheless, assessing demand-side strategies would require a similar modeling framework, and the models presented here can form a useful foundation for demand-side modeling as well. Some of the numerical experiments involve predicting impacts of changes in demand without explicitly modeling the policies that would result in these demand reductions.

1.1 Background

Since the introduction of the automobile, cities have increasingly oriented themselves towards personal vehicles as the dominant mode of transportation. Emerging from World War II, car ownership became cemented as part of the American cultural fabric, acting not just as a symbol of independence and prosperity, but physically liberating middle-class families to travel when and where they pleased. Over a span of four decades, vehicle ownership rates soared in the U.S., from less than 250 vehicles per 1,000 people in 1950, to over 700 vehicles per 1,000 people by 1990 (Shoup and Association, 2011). To accommodate this rise in personal automobiles, cities and states expanded their roads, highway networks, and supply of parking spaces.

This trend clearly has had negative consequences, which are becoming more and more pronounced as populations rise. In urban centers where space is limited and cars are prevalent, parking shortages cause drivers to cruise for parking. In the hopes of finding cheap, available parking, drivers circle indefinitely. This practice significantly worsens congestion, increases carbon emissions, and makes the roadways less safe for other drivers, pedestrians, and cyclists (Feeney, 1989). Averaging the results of ten different international studies, it is estimated that during typical peak hours, approximately 34% of congestion in urban areas is made up of people "cruising" for parking (Shoup and Association, 2011). Analyzing the problem from a drivers perspective, a study in Frankfurt showed that searching for a parking spot during peak hours accounted for as much as 40% of the total travel time for journeys to central urban areas (Axhausen et al., 1994).

The other major parking dilemma stems from an abundance of off-street parking. In order to ensure that businesses and developers allocate enough parking to satisfy the "demand" of the citizens, most cities place minimum requirements on the amount of free parking that each type of business must provide, based on land use and square footage. When deciding what these minimums ought to be, the vast majority of cities derive their policies — either directly or indirectly — from the Institute of Transportation Engineers (ITE) report *Parking Generation*. Shoup and Association (2011) argues that the parking generation rates they produce are often overestimates, derived from sparse amounts of data drawn from suburban areas with no transit ridership. The practice of setting inflated parking minimum requirements creates two problems: it hides the true cost of parking while worsening urban sprawl. Parking lots use valuable land and cost a substantial amount of money to construct, maintain and police. According to a Massachusetts developer, the construction of a parking structure in Boston can cost $30,000 to $50,000 per space (Ross, 2013). Rather than passing on these costs only to the individuals who choose to travel by car, the costs are often hidden, with business and developers instead passing on the costs to everyone in the form of higher rents, lower wages, and higher costs for goods and services. Moreover, the expansive free surface parking lots mandated by city codes only serve to spread out and weaken the city as a whole. They destroy a citys aesthetic, lower property values, limit density, and weaken the tax base. They also create a positive feedback loop, where not only is there a lack of incentives for people to adopt alternative transportation, but the sprawling suburban blueprint necessitates the use of personal vehicles, which in turn drives up the demand for more parking (Shoup and Association, 2011).

Due to the huge impact that parking management has on urban development and transportation networks, focus on parking policy has intensified over the past several years. Numerous studies have analyzed the effects of changes in parking capacity, pricing, and strategies aimed at reducing the number of single person trips by car. While politically challenging, setting appropriate pricing for parking can serve as the most effective tool for shifting peoples transportation choices and mitigating urban congestion (Hensher and King, 2001). Of the cities who have instituted pricing reforms, San Franciscos demand-based parking program has been arguably the most innovative and comprehensive. Termed SFpark, the program launched in April 2011, funded primarily by a $20 million federal grant. Beginning in 2010 the city installed enough sensors to cover 7,000 metered parking spots (one fourth of all metered spots in the city) as well as 12,250 spots in 15 of the 20 city-maintained garages. The city then collected and published the occupancy data, lowering and raising parking rates block by block. The goal of the project is to provide real-time price and availability data to the public while using straightforward and transparent pricing adjustments to control turnover, availability, and to influence peoples travel decisions. Rates are set as low as possible while still being high enough to provide at least 1 to 2 free spots on each block at any given time, thus negating the need to cruise for parking. As of May 2012, some meters had reached as high as $6 per hour, but in aggregate, the average meter price fell 1% over the first year. The program also achieved its primary goal, as congestion has decreased measurably, allowing public transit to run quicker and more reliably. There is still room for improvement — pricing could become more predictive than reactionary — but the program has been hailed as a success thus far, with in-depth analysis to be conducted in the coming months (Pierce and Shoup, 2013). Similar programs have been implemented in Seattle and Washington D.C. at a much smaller scale. Despite relying on human observations since sensors were prohibitively expensive, the projects yielded comparable results to the SFpark program. Other successful strategies employed by urban planners include parking taxes in central business districts, providing incentives for travel during non-peak hours, extending parking meter hours, and installing electronic guidance systems to direct drivers to available lots (Greenberg, 2012).

Using parking policy to combat urban sprawl as well as congestion, several cities are reducing or even eliminating minimum parking requirements. The city of Austin recently launched a pilot program to reduce parking requirements for expanding businesses who commit to implementing measures designed to reduce individual motor vehicle trips. Some of the potential strategies include providing services (cafeteria, daycare, etc.) for tenants so they dont have to go offsite, installing showers for tenants who choose to bike or walk to work, establishing preferential parking for carpoolers, providing cash incentives for employees who dont use a personal vehicle to get to work, guaranteeing rides home, and unbundling parking, whereby parking spaces are offered at a fee, rather than provided for free.

Some cities, including San Francisco, Boston, Milwaukee, Seattle, and Portland, are going a step further by completely reversing policy and setting caps on the number of parking spaces developers can provide (Ross, 2013). In San Francisco, large new residential projects are not only governed by parking caps, but developers must also unbundle parking and provide car share spaces. A follow-up study found that buildings which combined unbundled parking with car share availability had significantly fewer occupants owning personal vehicles than did buildings which had only one or neither of the measures in place (Greenberg, 2012). Policies which limit parking capacity have received some backlash, with citizens arguing that the plans ignore the present reality in the pursuit of an idealistic future (Ross, 2013). However, several studies show a discernible decade-long trend away from personal vehicles. Young Americans (aged 16-34) are driving 23% fewer miles than they were in 2004 and are more committed to alternative forms of transit, even those with the means to purchase a vehicle (Davis et al., 2012). Also, urban growth has just outpaced suburban growth for the first time since the 1920s, fueled by a growing number of people opting to live near their jobs, entertainment, and public transit (Yen and Wyatt, 2012).

1.2 Motivation

As discussed in the background section, parking has a significant impact on urban transportation. Transportation planners, traffic engineers, and others in the private sector are beginning to realize the importance of parking management. New sensor and information provision technologies are being used as the basis for innovative parking management strategies. For instance, parking garages can be equipped with sensors to monitor the location and number of available spaces. Cities such as Dallas and San Francisco are also equipping on-street parking spaces with sensors. This information can be disseminated to drivers through variable message signs, smartphone applications, or used to adjust the price of parking in real time based on availability.

However, network models that can accurately assess the impact of parking on urban congestion, and the effectiveness of these control strategies, have yet to be developed. In particular, providing more information does not always improve transportation conditions, partially due to latency effects but also due to the conflict between drivers' individual objectives and system objectives.

Existing network models either ignore parking search altogether, or treat it in limited ways which fail to capture its full impacts. Most network models simply route travelers from their origin node to the destination node, ignoring the parking search process. Ignoring this will systematically underestimate congestion, potentially by a significant amount. Current models which do incorporate parking effects are limited in several ways. One class of parking models is simulation based, which has the advantage of rich traffic flow modeling and modeling uncertainty in parking space location, but fails to capture the “gaming” behavior in parking choices — drivers choose a searching strategy based on the likelihood of finding parking, but this likelihood depends on the searching strategies used by other drivers. Another class of models adds parking links to network models and imposes a flow-dependent impedance function to reflect parking delay. While it is easy to incorporate an equilibrium principle into this framework, this approach neglects stochasticity altogether and cannot predict which roadway links will experience higher congestion due to parking search. Yet another class of models derives exact mathematical results in stylized networks. While these approaches can provide insights into specific policies and derive elegant results, they are highly restricted by assumptions on the network topology (e.g., that the city is a single link, or a homogeneous circle).

This dissertation develops network models for parking search which simultaneously (1) account explicitly for driver behavior in response to network conditions, rather than simply simulating network performance with exogenous behavior; (2) are stochastic, reflecting uncertainty in the locations of available parking spaces and identifying the impacts of parking search on each network link; and (3) are scalable to large networks without restrictive assumptions on the network topology. To the author’s knowledge, this is the first time all three features have been combined in a single modeling framework.

1.3 Problem Statements

This dissertation addresses three parking-related network problems. The first is to model an individual driver’s parking search behavior, taking into account the likelihood of finding parking in different locations from past experience as well as observations gained during the search itself. In this problem, the *a priori* probabilities of finding parking are assumed fixed and known since the focus is on an individual's behavior, and a single driver has a negligible impact on the probability of finding parking at a location. This problem is formulated as a stochastic shortest path problem with recourse, and described using the framework of Markov decision processes (MDPs). In this MDP model, the state space consists of the most recently visited nodes and the parking availability at each of these nodes. The decision variable is at a link the driver can park and walk to the destination or continue to drive to one of the adjacent nodes. The transition probability specifies the probability from one state to another. Since the probability is calculated depending upon the time last since last visit of the link and the current state, it can reflect the traveling history impacts on the transition. The detailed calculation and explanation are given in Chapter 2.

The primary contribution of the proposed model is its ability to reflect history dependence. Other recourse shortest path problems in the literature either assume "full reset," in which the state of a link is independent of its state in any past observation, or "no reset" in which the state of a link never changes once observed, completely determined by past observation. For instance, assume that the *a priori* probability of finding parking on a link is 30%. "Full reset" implies that if a driver drives on this link and sees no parking available, if he or she immediately turns around and drives on the link again, the probability of finding parking is again 30% independent of the past observation. By contrast, "no reset" implies that if a parking space is available on a link, it will always be available to return to in the future at any point. This dissertation develops an “asymptotic reset” principle which generalizes these principles and allows past observations to affect the probability of finding parking on a link and this impact weakens as time goes by. Both full reset and no reset are shown to be special cases of asymptotic reset.

The second problem is modeling multiple drivers through a parking search equilibrium on a static network. Similar to the first type of problem, drivers aim to minimize their total travel costs. Their driving and parking search behaviors depend on the probabilities of finding parkings at particular locations in the network. On the other side, these probabilities depend on drivers’ route and parking choices. This mutual dependency can be modeled as an equilibrium problem. At the equilibrium condition no driver can improve his or her expected travel cost by unilaterally changing his or her routing and parking search strategy. To accomplish this, a network transformation is introduced to distinguish between drivers searching for parking on a link and drivers merely passing through. The dependence of parking probability on flow rates results in a set of nonlinear flow conservation equations. Nevertheless, under relatively weak assumptions the existence and uniqueness of the network loading can be shown, and an intuitive “flow-pushing” algorithm can be used to solve for the solution of this nonlinear system. Built on this network loading algorithm, travel times can be computed. The equilibrium is formulated as a variational inequality, and a heuristic algorithm is presented to solve it. An extensive set of numerical tests shows how parking availability and traffic congestion (flows and delays) vary with the input data.

The third problem aims at developing a dynamic equivalent for the network parking search equilibrium problem. This problem attempts to model a similar set of features as the static model, but aims to reflect changes in input demand, congestion, and parking space availability over time. The approach described in the dissertation is complementary to the static approach, taking on the flavor of simulation more than mathematical formulation. The dynamic model augments the cell transmission model with additional state variables to reflect parking availability, and integrates this network loading with an MDP-based parking search behavior model.

As stated above, all of these problems develop parking models which explicitly consider driver behavior, stochasticity in the parking search process, and are applicable to general networks.

1.4 Organization

As mentioned above, this dissertation mainly incorporates stochastic parking search behavior into traffic assignment problems. This work is divided into three parts in this dissertation. The first part is to develop models of single vehicle's stochastic parking search process on a network with history dependence. The second part constructs models for multi vehicles' parking search process on a network and calculate the static traffic equilibrium for them. The third part formulates dynamic traffic equilibrium models for multi vehicle's parking search behavior. A more detailed outline of the remainder of the dissertation is as follows:

Chapter 2 Parking Search with History Dependence

Drivers have memory of parking availability on recently-traversed links and may use this information to reevaluate their route and parking choice every time they get to a new network link. In our model drivers seek to minimize their total expected travel time, driving plus walking, though this disutility function could easily be adapted to incorporate fees, search time, or a host of other characteristics. We name our approach an asymptotic reset model, as it generalizes the “reset” and “no reset” formulations identified by Provan (2003). This model is actually an application of MDPs the parking search process on a network. Finally a value iteration method is designed to solve the MDP.

Chapter 3 Static Parking Search Equilibrium

A network transformation is introduced to differentiate common links that for passing through traffic from dummy links that account for vehicles searching for parking. Traffic flows will be loaded onto a network based on the parking space availability probability and traffic equilibrium principles. The probability of parking space availability and link flows are mutually dependent, which results in a set of nonlinear flow conservation equations. The equilibrium is formulated as a variational inequality and a heuristic algorithm is developed to solve it.

Chapter 4 Dynamic Traffic Assignment and Parking Search

The goal of this part is particularly aimed at the “cruising” phenomenon when drivers are searching for parking. Therefore an equilibrium formulation accounting for stochastic and dynamic parking search by routing drivers is developed based on *policies* rather than *paths*, using the language and framework of Markov decision processes. The stochastic and dynamic nature of the parking search process is incorporated in the traffic flow model which is built on the cell transmission model (CTM) with added state variables to represent the number of available parking spaces on links.

Chapter 5 Conclusion

A summary of the three models mentioned in the previous chapters and the contributions of this dissertation are detailed in this chapter. Based on current research results, the future possbile research directions and topics are listed.

Chapter 2

Parking Search with History Dependence

2.1 Introduction

Existing parking search models usually make extreme simplifying assumptions and may not be able to capture real parking behavior. In reality, since drivers are unaware of the exact likelihood of finding parking near their destination and circling wastes time, they will typically pay attention to availability as they approach the area. Their parking choice will then be a memory-influenced decision, where they circle back to a previously seen spot if necessary. Furthermore, the realized probability of parking availability depends on the memory. This chapter characterizes this memory-influenced parking search behavior with the help of a Markov decision process model.

2.2 Literature Review

As mentioned in Chapter 1, parking search models can be classified into three groups: discrete choice based approaches, simulation based approaches, and network assignment based approaches.

Both discrete choice and simulation approaches examine parking choice explicitly. Discrete choice models work at the macro level, using random utility theory to understand parking choice as a function of various driver and parking facility attributes. Such models differ in their complexity, with some utilizing the multinomial logit model (Van der Goot, 1982; Axhausen and Polak, 1991; Lambe, 1996) and others using mixed multinomial logit (Hess and Polak, 2004), or nested logit models (Hensher and King, 2001). However, by neglecting the network structure, discrete choice models are unable to model the stochastic and adaptive nature of the parking search process, as drivers sequentially traverse roadway links which may or may not have available parking.

In contrast with discrete choice models, simulation models try to capture the parking search at the micro level. Thompson and Richardson (1998) developed an analytical model to mimic the search process where the disutility of a car park location was defined as a function of in-vehicle travel time, in-car park search time, waiting time, fees, fines, and walk time. Other researchers (Thompson and Richardson, 1998; Benenson et al., 2008; Martens and Benenson, 2008; Dieussaert et al., 2009) have adopted agent-based approaches where the behavioral and parking decision making rules were assigned to the drivers. An issue with micro-simulation models however, is that their size must be restrained due to computational complexity and to date none fully address the dynamic effects that congestion and parking choices have on one another (Waraich and Axhausen, 2012).

Also working at the macroscopic level, network approaches based in equilibrium assignments are regarded for their ability to successfully model the interaction between road traffic and parking choices (Waraich and Axhausen, 2012). Like non-network models, network approaches are predicated on individuals choosing parking locations which maximize their utility or minimize travel costs, though they try to simulate the parking choice implicitly (DellOrco et al., 2003). Hall (1986) developed a recursive algorithm to solve shortest paths in networks where arc costs were random and time-dependent. Introducing the concept of recourse, Polychronopoulos and Tsitsiklis (1996) proposed a formulation where arc costs are learned progressively as an end-node of an arc is visited, enabling policies to include cycling and corrective actions where information gathering is beneficial.Waller and Ziliaskopoulos (2002) analyzed networks with spatial dependence and temporal dependence of arc costs in further detail, showing that online optimum paths outperform offline shortest paths by up to 40% under certain conditions. Provan (2003) provided a polynomial-time algorithm for solving shortest paths with recourse where arc lengths are determined by a Markov process and reset upon each traversal. Even these models (Arnott and Inci, 2010; Leurent and Boujnah, 2012; Provan, 2003; Hall, 1986; Polychronopoulos and Tsitsiklis, 1996; Waller and Ziliaskopoulos, 2002) which capture the unknown and stochastic nature of arc costs do not incorporate any kind of memory-based decisions, a key feature of the individual parking search, and typically assume either full reset or no reset conditions, to use the language of Provan (2003). Neither assumption well-characterizes the parking search problem.

Therefore, the contribution of the model is to include a memory for the traveler, in which the probability of finding parking after traversing a link gradually resets to an a priori probability as the time since traversal increases, a formulation we term asymptotic reset. This formulation generalizes the full reset and no reset formulations, which can be obtained as special cases. Incorporating the concepts discussed above, our model treats the individual parking search as a Markov decision process

(MDP), examining whether a node has been visited previously, and then using that information to influence the probability of finding parking at that link in the near future.

2.3 Methodology

2.3.1 Notation and problem description

Let $G^o = (N^o, A^o)$ be an undirected graph/network, where N^0 represents the set of nodes and A^o consists of arcs/links. The set N^o is defined as $N_r^o \cup N_d^o$ where N_r^o is the set of actual intersections in the network and N_d^o denotes a set of dummy nodes. Similarly, A^o is comprised of actual roadway links (A_r^o) and a set of dummy links (A_d^o). The construction of dummy nodes and links, and their function will become evident in the example discussed later. Assume $G = (N, A)$ represents the dual graph of G^o, i.e., $N = A^o$ and a link $a = \{i, j\} \in A$ where $i, j \in N = A^o \Leftrightarrow i$ and j have a common end-point in G^o. In other words, the nodes in G represent arcs in G^o and the arcs in G represent turn movements in G^o. We also refer to G^o as the original graph/network and refer to the dual graph simply as graph/network. Usage of the dual graph to model the parking search process is solely due to its relative ease in demonstrating the proposed methods. Further, the formulation on the dual graph is consistent with the widely used notion of making decisions at a node rather than on a link, in various transportation network models.

The cost of an arc $i \in A_r^o$, is denoted by c_i^o and is static and deterministic. The corresponding node in the dual network is assumed to be equidistant from its end-points. With the exception of the dual nodes and arcs created from the dummy nodes and links of the original network, the cost of an arc $a = \{i, j\} \in A$ is defined as $c_a = (c_i^o + c_j^o)/2$. The choice of the dummy link costs will be explained later using an example. Let the walking travel time from node $i \in N$ to the destination be w_i. The prior probabilities of finding parking at each node, i , in the network is also assumed to be known and is denoted by ρ_i.

As drivers travel through the network, they are assumed to remember if a previously traversed node had parking (P) or not (NP). However, this ability to retain information is also assumed to be limited to the m nodes that were most recently visited (excluding the current node of the traveler). The value of m, also called the *memory limit*, is a parameter of the model. Let the set $\Omega = \{P, NP\}$ represent the parking conditions at a node. The state of a driver s is then defined using an $(m+1)$-dimensional vector of ordered pairs $((i_1, p_1), (i_2, p_2), ..., (i_{m+1}, p_{m+1}))$, where $i_1, i_2, ..., i_{m+1} \in N, p_1, p_2, ..., p_{m+1} \in \Omega$ and $i_1 - i_2 - ... - i_{m+1}$ represents a path (with repetition and cycles allowed) of the most recent m-nodes visited. The current node at which the traveler is present, i_1, is also represented by $\eta(s)$; and the current parking availability, p_1, is denoted by $\pi(s)$. The set of all states or the state space is denoted by S. At each state the driver may choose to park (only if $\pi(s) = P$) or continue to drive. For each decision at s, the driver may find himself/herself at a subset of states with some known probability. The parking search problem can thus be formulated as an MDP in which drivers use an adaptive strategy that is conditional on their current state. The objective of the problem is to find such an adaptive strategy that minimizes the expected cost of reaching the destination. The list of symbols used is shown in Table 2.1.

Table 2.1: List of Symbols

Symbol	*Description*
N	Set of nodes in the dual network
A	Set of links in the dual network
S	State space
c_a	Cost of travel on link a, where $a \in A$
ρ_i	Prior probability of finding parking at node i, where $i \in N$
w_i	Walk time to destination from node i, where $i \in N$
$\eta(s)$	The current node at which the traveler is present in state $s \in S$
$\pi(s)$	Parking availability at node $\eta(s)$, where $s \in S$
$N(i)$	Adjacency list of node i, where $i \in N$
$X(s)$	Decision space at state s, where $s \in S$
λ_i	Reset rate at node i, where $i \in N$
$V(s)$	Value or reward of state s, where $s \in S$

2.3.2 Assumptions

In modeling the parking search process as an MDP the following assumptions are made:

- The network is undirected and ρ_i represents the probability of finding parking on either sides of a link in the original network. This assumption is not restrictive and extending the proposed model to directed networks is straightforward.

- The traveler is experienced enough to have a knowledge of the arc costs and prior probabilities of finding parking.

- The model does not explicitly capture the effect of parking costs. However, this can be incorporated in a straightforward way by introducing node costs and minimizing the expected generalized cost of travel.

- Travelers are willing to walk to the destination from any node in the network. In practice, we may select a small sub-network around the destination, only considering nodes for which walking is feasible, or set the walking costs to a sufficiently large value.

- The walking distance from a link in the original network to the destination does not depend on the location at which the traveler parks. This simplification is reasonable unless links in the original network are extremely long. If the links are long, we can split a long link into shorter ones and model the search process on the resulting network.

- The transition probabilities depend on whether a node that a driver considers to visit features in his/her current state vector, the time elapsed since it was last visited and a reset rate parameter (λ_i). The reset rate is a measure of how quickly parking probabilities reset to their priors. In practice, this rate could depend on factors such as land use of the neighborhood, parking meter rates and occurrence of special events. This assumption is a key characteristic in this chapter and is a generalization of the full reset and no-reset versions of stochastic shortest path problems (SSP). A more detailed description of the transition probabilities will follow later.

2.3.3 An illustration of parking search process

The following example illustrates the MDP formulation of the parking search problem. Consider the network shown in Figure 2.1. The left panel contains the original network. Assume that the traveler departs from node 1. The dummy nodes

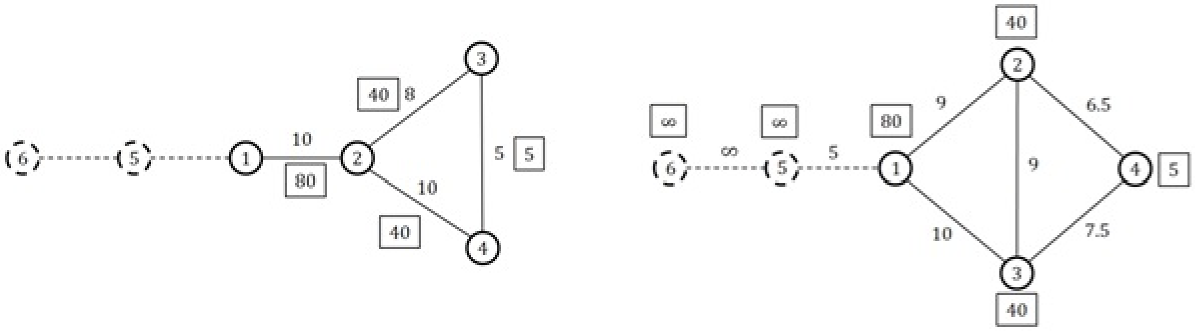


Figure 2.1: An illustration of the parking search process

and arcs are shown in dotted lines. The travel cost on each arc and the walking time to the destination (within boxes) are shown on the links. We create $m+1$ dummy nodes and arcs (m is assumed to be 1 in this example) in series and connect it to the source node, i.e., node 1. These dummy nodes and links are constructed because when a traveler starts at the origin, he/she has no memory of links traversed. In order to define the state of the traveler in such situations we assume that he/she had traversed these dummy links before arriving at node 1.

The resulting dual network is shown in the right panel of Figure 2.1. Arc $\{1, 2\}$ in original network is node 1 in the dual network, arc $\{2, 3\}$ is node 2, etc. The origin node of the traveler in the dual network is node 5. The cost of arc $\{5, 1\}$ in the dual network is set to $10/2 = 5$, as dual node 1 is assumed to be located midway of arc $\{1, 2\}$ in the original network. Since the dummy nodes are non-existent in reality, the walking distances to the destination and the parking probabilities are set to ∞ and 0 respectively. Assume that the probability of finding parking at nodes $1, 2$ and 4 in the dual network is 0.5 and the probability of finding parking at node 3 is 0.8.

In the full reset case, the travelers state is defined using his/her current node and the availability of parking at it. The optimal solution under the full reset version prescribes a traveler to first choose path $5-1-2-4$ and park at node 4 and walk to the destination if possible. However, if parking is unavailable at node 4, the strategy suggests to cycle between nodes 4 and 2 until parking is found at node 4. The optimal expected cost of the adaptive strategy was found to be 38.5 units. Although, this variant of the parking search problem is easy to solve, the probabilities of finding parking reset to the prior probabilities each time the driver revisits a node. This is unrealistic as travelers would update their beliefs of finding parking based on their previous experiences.

Hence, we propose an asymptotic reset version which assumes that the probability of finding parking depends on the state vector. For instance, if a driver cannot park at node 4; when at node 2, the probability of finding parking at node 4 is updated to a value that is less than 0.5 since he/she could not find parking at an earlier point in time. This value is modeled to be dependent on the time taken to travel back and forth between node 4, i.e., 13 units and the reset rate of node 4. However if node 4 does not appear in the current state of the traveler, even if it was previously visited (i.e., the traveler forgets having visited node 4), the probability of finding parking at node 4 is reset to its prior value. The optimal expected cost for the asymptotic reset version was found to be 46.78 units and can be obtained from the value/reward of any one of the states $((5, p_1), (6, p_2))$, where $p_1, p_2 \in \Omega$. Unlike the full-reset case the optimal strategy is harder to describe as it explores node 3 and suggests to park on nodes 2 and 3 if parking was unavailable at node 4.

2.3.4 Dynamic programming formulation of the parking search process

In this section, we discuss the components of the Markov decision process used to model the parking search problem: the state space, decision space, transition probabilities and the value functions. The value iteration algorithm used to compute the optimal strategy is also explained.

State Space

As discussed earlier, the state space consists of the most recently visited nodes and the parking availability at each of these nodes. Populating the state space requires enumeration of all paths of size m. An efficient way to enumerate such paths using repeated breadth first search (BFS) is outlined in the following pseudo code. Let $\Gamma_i(j)$ represent the set of nodes which can be reached from node i by traversing j arcs or less. Suppose the set of all paths of size m is denoted by Γ. The state space can then be written as $S = \Gamma \times \Omega^{m+1}$.

Algorithm 1 State Space Contruction

```
for all $i \in N$ do
    Perform BFS with $i$ as the origin
    Store the BFS distance labels (shortest number of arc required ro reach each node)
    for $j = 0$ to $m$ do
        Populate $\Gamma_i(j)$ using BFS labels
    end for
    $\Gamma_i = \times_{j=0}^{m} \Gamma_i(j)$
    Scan each element of $\Gamma_i$ and discard infeasible paths
end for
$\Gamma = \cup_i \Gamma_i$
```

Decision Space

The set of decisions available at state s, denoted by $X(s)$, can be defined by classifying the states into the following types:

$$X(s) = \begin{cases} N(\eta(s)) \cup \{Destination\} & \text{if } \pi(s) = P \\ N(\eta(s)) & \text{otherwise} \end{cases}$$

In the first case, the driver can park and walk to the destination or continue to drive to one of the adjacent nodes. However, if parking is unavailable at the current state (second case), the driver has no option but to drive to a node in $N(\eta(s))$.

Transition Probabilities

Given a state $s = ((i_1, p_1), (i_2, p_2), ..., (i_{m+1}, p_{m+1}))$, and a decision $i \in N(\eta(s))$, the transition probabilities, denoted by $\Pr(s' \mid s, i)$, specifies the probability of reaching state $s' = ((i, P), (i_1, p_1), ..., (i_m, p_m))$. Notice that the probability of reaching state $s'' = ((i, NP), (i_1, p_1), ..., (i_m, p_m))$, $\Pr(s'' \mid s, i)$, is simply $1 - \Pr(s' \mid s, i)$. $\Pr(s' \mid s, i)$depends on whether node i appears in s and if it does; it is a function of the time elapsed between recent revisits to node i which is equal to the cost of path $i - i_1 - i_2 - ... - i_k = i$, where k in $\{1, ..., m+1\}$ and $i_l \neq i \forall l \in \{2, ..., k-1\}$. Let $t = c_{i,i_1} + c_{i_1,i_2} + + c_{i_{k-1},i_k}$ represent the cost of this path. The transition probabilities are assumed to be governed by the following equations:

$$\Pr(s' \mid s, i) = \begin{cases} \rho_i(1 - \exp^{-\lambda_i t}) & \text{if } p_k = NP \\ \rho_i + (1 - \rho_i)\exp^{-\lambda_i t} & \text{if } p_k = P \end{cases}$$

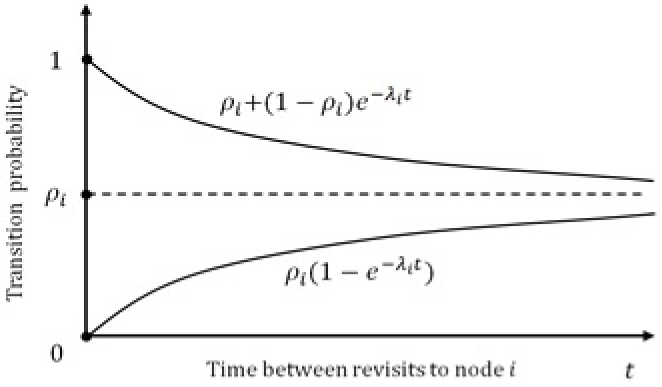


Figure 2.2: Asymptotic reset of transition probability

Figure 2.2 shows the variation of the transition probability with time. The probability of finding parking is reset to the prior probabilities in an asymptotic manner. As mentioned earlier, these equations help us formulate an intermediate version of the full and no-reset SSP models. If node i does not appear in state s, the conditional probability of finding parking $\Pr(s' \mid s, i)$ is simply assumed to be ρ_i.

Value Functions

The value or reward of a state $V(s)$ is the expected cost of reaching the destination from state s. Using the notation defined in the previous section, the optimality criteria for the dynamic program can be expressed as follows:

$$V(s) = \begin{cases} \min\limits_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V(s') + \Pr(s'' \mid s, i)V(s'') \right\} & \text{if } \pi(s) = NP \\ \min\limits_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V(s') + \Pr(s'' \mid s, i)V(s''), w_{\eta(s)} \right\} & \text{if } \pi(s) = P \end{cases}$$

$$\text{If } \pi(s) = NP, V(s) = \min_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V(s') + \Pr(s'' \mid s, i)V(s'') \right\}$$

$$\text{If } \pi(s) = P, V(s) = \min_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V(s') + \Pr(s'' \mid s, i)V(s''), w_{\eta(s)} \right\}$$

The optimal values of the states can be computed using the value iteration method, a pseudo code for which is presented below. The algorithm iteratively updates the values of each state using the above optimality criteria and terminates if the change in values across successive iterations is less than a given threshold ϵ.

Algorithm 2 VALUE ITERATION

```
Step 0: Initialization
$V^0(s) = w_{\eta(s)} \forall s : \pi(s) = P$
$V^0(s) = 0 \ \forall \ s : \pi(s) = NP$
$k = 1$
Step 1:
for all $s \in S$ do
  if $\pi(s) = NP$ then
    $V^k(s) = \min_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V^{k-1}(s') + \Pr(s'' \mid s, i)V^{k-1}(s'') \right\}$
  else
    $V^k(s) = \min_{i \in N(\eta(s))} \left\{ c_{\eta(s),i} + \Pr(s' \mid s, i)V^{k-1}(s') + \Pr(s'' \mid s, i)V^{k-1}(s''), w_{\eta(s)} \right\}$
  end if
end for
Step 2:
if $|V^k(s) - V^{k-1}(s)| < \epsilon \forall s \in S$ then
  Terminate and $V^k(s)$ is the optimal value of $s$
else
  $k = k + 1$ and go to Step 1
end if
```

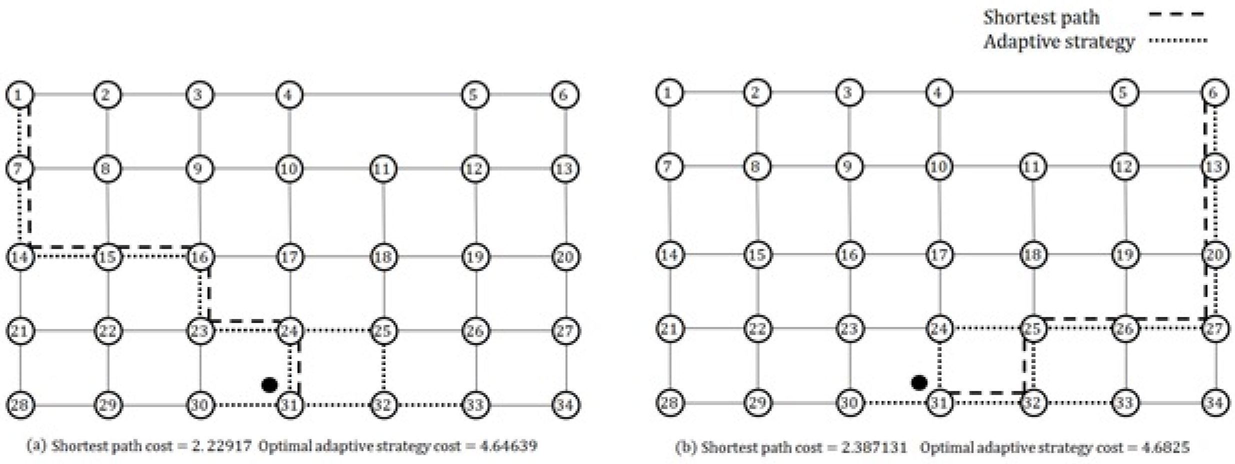
2.4 Case Study

This section contains a case study of the parking search model. A network representing the main campus of the University of Wyoming (UW), Laramie, WY, was used for this demonstration. The network consists of 34 nodes and 56 arcs (see Figure 2.3), and the destination is closest to node 31(represented by a black circle in Figure 2.3). All results are discussed using the original network and not its dual. The probabilities of finding parking were found from previous parking studies. A graduate student estimated the prior parking probabilities by riding his bicycle on these streets for 10 days, which suffices for the purpose of this demonstration. A constant reset rate λ was used for all links in the network. The implementation was carried out in C++ (using the g++ compiler with -O3 optimization flags) on a Linux machine with a 4 core Intel Xeon processor (3.47 GHz) and 12 MB Cache.

2.4.1 Excess cost of parking

Most transportation models assume that trips begin and end at nodes and parking is not explicitly modeled. However, in reality one is likely to drive around the destination until a suitable parking spot is found, resulting in an increase in trip duration. In this section, we quantify the expected increase in trip duration by comparing the shortest path cost of reaching the destination and the optimal expected cost of parking using an adaptive strategy. The values of m and λ were set to 4 and 1 respectively.

Figure 2.3 shows the links used in the shortest path and the optimal adaptive strategy for two origins 1 and 6. As expected the optimal adaptive strategy explores



(a) Shortest path cost = 2. 22917 Optimal adaptive strategy cost = 4.64639

(b) Shortest path cost = 2.387131 Optimal adaptive strategy cost = 4.6825

Figure 2.3: Impact of Parking on Trip Costs

more links either because of the lack of parking or due to the anticipation of finding a better parking spot. The expected cost of the adaptive strategy was found to be approximately twice as much as the shortest path cost. Notice that most of the arcs that are revisited are centered around the destination.

2.4.2 Effect of memory

From a theoretical standpoint, it would be ideal to compute an adaptive strategy based on an infinite memory. One could assume that the driver is assisted by a navigation system which keeps track of parking conditions on all traversed links. However, as the memory limit is increased, the size of the state space grows exponentially and the problem ends up being computationally intractable. For instance, as can be seen from Table 2.2, the size of the state space for a memory limit of 5 was found to be nearly 6.5 million and the wall clock time for computing the optimal

Table 2.2: Results of the MDP for Different Memory Limits

Memory limit	Optimal solution	MC estimate of the optimal strategy	Estimate of standard deviation	95% CI	Computation time (in second)	State space size
1	4.0698	7.212	6.008	(7.045, 7.378)	0.0076	556
2	4.55252	5.584	3.676	(5.482, 5.686)	0.0853	5640
3	4.55252	5.591	3.644	(5.490, 5.692)	1.2840	58352
4	4.64639	5.223	3.095	(5.137, 5.309)	17.6006	611424
5	4.72688	5.061	2.516	(4.991, 5.130)	231.4460	6457664

solution was found to be approximately 4 minutes.

We explored the performance of the optimal strategy for different memory limits in an infinite memory setting using Monte Carlo (MC) simulations. Specifically, at each state, the probabilities of finding parking were drawn from a distribution that is a function of the infinite memory (which comprises of the parking conditions on nodes visited since the start of the trip), but the adaptive strategy used prescribes a decision only based on the state of the traveler. The following table shows the expected cost of the optimal policy for memory limits 1 through 5 and an estimate of expected cost of the policy under the infinite memory setting. A sample size of 5000 was used for the MC simulations. The confidence intervals for the estimated expected costs are reported. The computational time for the MDP in seconds and the sizes of the state space are also shown. As the memory limit increases, the gap between the optimal solution and the MC estimate of the optimal strategy reduces as expected. This gap decreases with increase in the memory limit and captures the trade-off between the optimal solution and its computational cost.

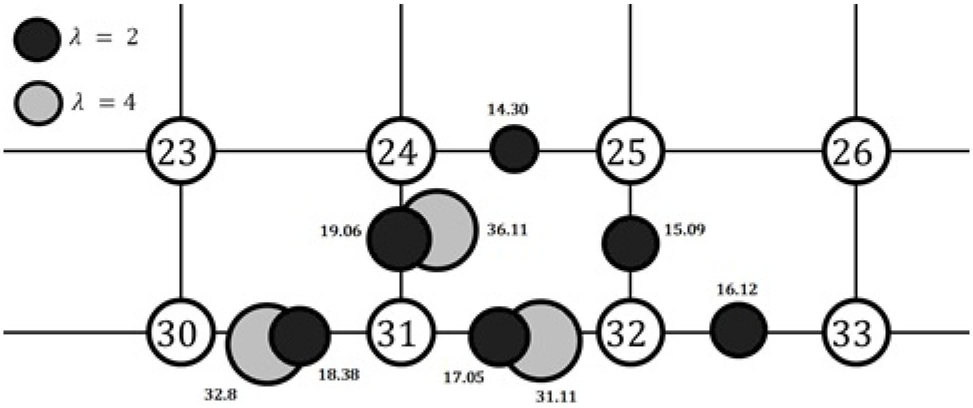


Figure 2.4: Percentage Distribution of Trip Ends

2.4.3 Location of trip ends

Lack of parking at a desired parking spot that is close to the destination forces drivers to park on nearby links and walk to the destination. Figure 2.4 shows the percentage of trips that end of links in the network for two values of λ and was computed by simulating the adaptive strategy in an infinite memory setting. It is interesting to note that as the value of λ increases, travelers are more likely to park closer to the destination. This is because for higher λ (which could be the case in the presence of special events) the probabilities reset to their prior values faster and mimics the full-reset version. Hence, it is advantageous to revisit links that are closer to the destination, even if parking has not been available on past visits.

2.5 Conclusions

This chapter formulates the parking search process as an online shortest path problem, developing the asymptotic reset model to incorporate memory of the parking status of links visited before. This online shortest path problem identifies a routing policy specifying whether drivers will choose to park at an available space or continue to search. Unlike previous research in this area, this approach simultaneously recognizes the stochasticity inherent in the parking search process, and represents the spatiotemporal characteristics of the underlying network structure. The case study demonstrates this model in a network representing a neighborhood near the University of Wyoming, analyzing the sensitivity of the solution to memory size and the value of the reset rate parameter.

This chapter lays the foundation for future research in several directions. First, spatial correlations can be accounted for, in that parking availability or lack of availability on particular links provides partial information on the likely parking availability on other links. More sophisticated cost functions could account for parking fees, consecutive parking time limitations, and other factors. More broadly, this stochastic shortest path formulation may be usable as the basis for an equilibrium algorithm involving many drivers, in which the probability of finding parking on a link depends on the search patterns used by all drivers.

Chapter 3

Static Parking Search Equilibrium

3.1 Introduction

The previous chapter explored the parking problem from the perspective of an individual driver. Because the impact of a single driver on the parking availability probabilities is small, we could assume those values constant and exogenous. However, planners are more concerned with aggregate travel patterns. When there are many drivers simultaneously searching for parking, however, this assumption is no longer reasonable.

Assuming that drivers aim to minimize the time spent traveling (including both driving and walking from the parking space to the destination), drivers' route and parking search behaviors depend on the probabilities of finding parking at particular locations in the network; however, these probabilities depend on the route and search strategies employed by drivers in the network. A natural model for this mutual dependency is an equilibrium framework in which no driver can improve his or her expected travel time by adjusting their strategy.

As discussed in the literature review, the model presented in this chapter builds on existing network parking models in the following ways. First, it is explicitly stochastic and reflects the dependence of parking probability on searching rates.

Second, it applies to general networks of any topology, and directly allows planners to identify which specific links and regions are particularly affected by increases in volume due to parking search. Third, the concepts of route choice and parking search are unified in a natural way which does not require assumptions such as drivers "transitioning" from driving towards the destination to searching for parking. Fourth, the introduction of an equilibrium concept captures the dependency between searching strategies and parking availability.

To accomplish this, a network transformation is introduced to distinguish between drivers searching for parking on a link and drivers merely passing through. The dependence of parking probability on flow rates results in a set of nonlinear flow conservation equations. Nevertheless, as shown below, under relatively weak assumptions the existence and uniqueness of the network loading can be shown, and an intuitive "flow-pushing" algorithm can be used to solve for the solution of this nonlinear system. Built on this network loading algorithm, travel times can be computed. The equilibrium is formulated as a variational inequality, and a heuristic algorithm is presented to solve it.

Unlike the models in Chapter 2, in this chapter, we do not consider history dependence in the formulation process. This is because including history dependence makes the equilibrium model would complicate the presentation considerably. This would distract from the main focus of the chapter, which is developing equilibrium for the parking search behavior. Future research will investigate this direction to refine the static assignment and parking search model.

The remainder of the chapter is organized as follows. Section 3.2 reviews relevant literature on the impact of parking in urban areas, along with network modeling approaches which have been proposed. Section 3.3 introduces the network transformation used to represent the stochastic nature of the parking search and notation which will be used throughout. Network loading and flow conservation are described in Section 3.4, along with the flow-pushing algorithm; this section describes the impact of travel choices on parking availability and link flows. Next, Section 3.5 introduces the complementary perspective of the impact of parking availability on travel choices, leading to an equilibrium definition to reconcile both perspectives. A solution heuristic is presented in this section as well. Section 3.6 demonstrates the algorithm's performance numerically and conducts sensitivity analyses, while Section 3.8 concludes and discusses future directions.

3.2 Literature Review

Network models that incorporate parking can be broadly classified into simulation-based approaches and analytic approaches. Readers may refer to Section 2.2 in Chapter 2 to find some finished research on simulation-based approaches. While simulation has the advantage of explicitly modeling parking dynamics and accommodating behavioral heterogeneity, they are limited in their ability to model large networks and are generally not amenable to exact results regarding the network loading and the equilibrium state. A further limitation is that in the absence of field data, there is an arbitrary element to the behavior rules, such as assuming that drivers will not cruise for parking if vacant spaces are available (Arnott et al., 1991), or that drivers will route deterministically to a preferred parking location; if that choice is unavailable, they will proceed to a second choice, third choice, and so on (Leurent and Boujnah, 2012). While some degree of arbitrariness is inevitable without field data, we prefer to build a model on a more fundamental principle. As described below, in our model the route choice and choice whether to take an available space if one exists are both governed by the principle of expected cost minimization, without the need to introduce a distinction between "driving toward the destination" and "searching for a parking space."

Analytical approaches, by contrast, are based on traffic assignment concepts and transform the network by adding new links to represent parking options. Typically these links are equipped with an impedance function to reflect delay due to parking search as more drivers attempt to park on that link. These approaches include Nour Eldin et al. (1981), Lam (2002), Lam et al. (2006), and Li et al. (2007a), and incorporate features such as endogenous mode choice accounting for parking, bilevel models for parking price. The main advantage of these approaches is their tractability, and ability to incorporate well-known results from the traffic assignment literature. However, by assuming a deterministic impedance for parking, the models are unable to reflect additional delay or volume on specific network links as drivers search for parking (possibly traversing a link multiple times as they cycle). Discrete choice concepts have also been used to study parking choice (Hunt and Teply, 1993), without explicit reference to a network, but using a nested logit model to account for similarities in on-street and off-street alternatives.

By contrast, the model described in this chapter is explicitly stochastic and can be used to identify specific links with increased volume due to parking search.

Furthermore, the network loading can be described analytically, and the equilibrium principle can be formulated mathematically. Furthermore, the behavior model relies on a fairly simple principle (travel time minimization) and does not require exogenous assumptions on when drivers begin searching for parking (which are common in agent-based simulation models). In this way it is similar to the model of Arnott and Rowse (1999), but allows for general network topologies, as opposed to assuming the network is a homogeneous circle, and does not require introducing a "cruising distance" from the destination, where drivers accept any vacant space within this threshold and reject any vacant space outside it. However, in contrast to Arnott and Rowse (1999) this chapter does not attempt to quantify the value of parking information systems or other parking-related policies, which seems challenging in more general network topologies.

Similarly, other researchers have studied parking-related policies, such as optimal pricing and control of parking spaces (Qian and Rajagopal, 2014), more general pricing problems with a single bottleneck for congestion (Zhang et al., 2008; Qian et al., 2012; Yang et al., 2013), distribution of permits for parking (Zhang et al., 2011). In order to focus on these policies, these researchers studied the parking problem in relatively stylized settings, such as a single bottleneck, a linear homogeneous city, or with two parallel alternatives. This chapter presents a complementary approach by focusing on the issues associated with a generic network topology, and aims to provide a foundation for extending the results of these earlier papers to more general networks.

An analogy can be drawn between the parking model developed in this chapter and the user equilibrium with recourse model (Unnikrishnan and Waller, 2009), in which travelers are routed stochastically based on realized network states. Indeed, both models have as a subproblem the online shortest path problem (Waller and Ziliaskopoulos, 2002; Provan, 2003), as recognized in Tang et al. (2014). The primary distinction is that the routing probabilities in user equilibrium with recourse are flow-independent, whereas in the parking model routing probabilities depend on flow to reflect the dependence of parking availability on searching intensity. This results in nonlinear flow conservation equations, which require more finesse. There is also a passing similarity to the equilibrium model developed in Nie (2011), in which the probability distributions of link travel times are flow-dependent; the primary distinction is that in this chapter the flow-dependent stochasticity affects the routing of vehicles, not just the time experienced. The following sections describe the model and modifications in more detail.

3.3 Network Structure

Consider a transportation network $G = (\mathcal{N}, \mathcal{A})$ with node and arc sets $\mathcal{N}$ and $\mathcal{A}$, respectively. The parking process is modeled using a network transformation (Figure 3.1) to represent potential parking availability. To represent this transformation, the node and arc sets are partitioned: the nodes $\mathcal{N}$ are divided into disjoint subsets N_R, N_P, N_D, and N_T: N_R represents the "regular" intersection nodes (the traditional nodes in most transportation network models), N_P represents the parking nodes (one for each link), N_D the destination nodes, and N_T the transition nodes connecting parked flows to the respective destinations. Notice that the destination nodes are

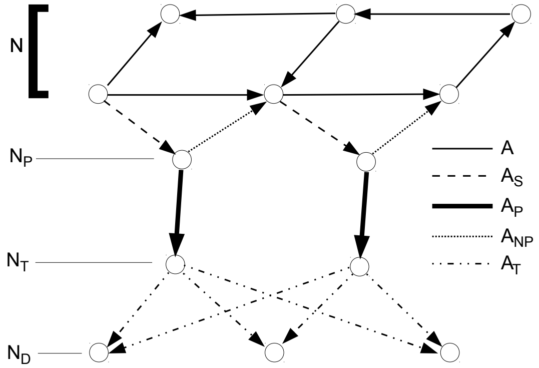


Figure 3.1: Linkages and layers between nodes and arcs for modeling parking.

connected to the intersection nodes only through parking and transfer nodes.

Similarly, the arcs $\mathcal{A}$ are divided into disjoint subsets A_R, A_S, A_P, A_{NP}, and A_T. These arcs can represent both on-street parking and off-street lots or garages. A_R represents "regular" or thru links, the traditional arcs in network models (the physical roadway infrastructure), and flow on this arc represents drivers who are *not* looking for parking. Each arc $(i,j)_R \in A_R$ is associated with one arc in each of the remaining subsets: flow on the corresponding arc $(i,j)_S \in A_S$ represents drivers who are searching for parking on link (i,j) and will park there if a space is available. The arc $(i,j)_P \in A_P$ is used to represent drivers who are actually able to find a space, and $(i,j)_{NP} \in A_{NP}$ drivers who are unable to find a space. Finally the links $(i,j)_T^d$ connect flow parked on link (i,j) to destination d. As a notational convention, we use $(i,j) \in \mathcal{A}$ and the subscript ij in equations whenever the type of arc (regular, parking, etc.) is irrelevant or when multiple types of arcs are meant. When only a single type of arc is intended, the links $(i,j)_R$, $(i,j)_S$, $(i,j)_P$, and $(i,j)_T^d$ are referred to with the subscripts ij, R; ij, S; ij, P; and ij, T, d, respectively.

Notice that each node in N may have incoming links from A_R and A_{NP} and outgoing links from A_R and A_S; each node in N_P has a single incoming link from A_S and two outgoing links, one from A_P and one from A_{NP}; each node in N_D has incoming links from A_T and no outgoing links; and each node in N_T has a single incoming link from A_P and outgoing links from A_T. We further assume that an arc exists between each transfer node and every destination. The interpretation is that it is always possible to walk from any link to the destination, although perhaps with high disutility.

Each link is associated with the steady-state flow rate of vehicles on this link: this value is denoted x_{ij}. Each node is associated with a flow conservation constraint relating the flow values on the incoming and outgoing links. These flow conservation constraints for our model differ from standard flow conservation constraints in two ways: (1) flow is split at parking nodes N_P based on the (flow-dependent) availability of parking spaces; and (2) flow is proportionately split at intersection nodes N to reflect the parking search process. The first component is discussed in Section 3.4, the second in Section 3.5.

Each link is also equipped with a generalized cost t_{ij} intended to reflect the total disutility of travel on that link (including time, cost, and other factors). For instance, additional disutility may be added to searching links to represent slower travel or more mental exertion and stress compared to thru driving. To simplify the notation these values are assumed constant and independent of flow. This is not restrictive and all of the algorithms and results in this chapter immediately transfer to the case of flow-dependent costs, even when the mapping from flow to cost is not separable (since congestion on searching and the corresponding regular link should be identical), as long as it is continuous.

Flow conservation equations have slightly different form for different types of nodes. Flow conservation for nodes in N_P is described in Section 3.4.1 along with the model for parking availability probabilities. Flow conservation for nodes in N_R is described in Section 3.4.2 along with the description of user behavior. Flow conservation for N_T and N_D nodes are straightforward and found in that subsection as well.

Our behavior model assumes that drivers make all parking-related choices to minimize the expected total cost of travel. In this chapter we assume that drivers can be aggregated by destination, which is consistent with this assumption. By linearity, any travel which has occurred between the origin and a driver's current location is a "sunk cost" which should not affect the decisions between this current location and the destination; therefore at any location, under the steady-state assumption, all drivers with the same destination face the same set of choices with the same set of costs.

Furthermore, the only place in our transformed network where drivers exercise "choice" is at the intersection nodes N_R. At parking nodes, the split of drivers is determined entirely by the parking probabilities, and at transfer nodes, all drivers head to the appropriate destination, as specified below. We model behavior through the use of splitting proportions, in which the fraction of drivers arriving at a node and heading to a common destination, and departing on a link (either a parking search or thru link), is specified.

These have a similar interpretation to the flow proportions in Bar-Gera (2002) and Gentile (2009), but differ because these works identify acyclic bushes in which these proportions identify path flows directly. The parking search process described here may involve cycles, because parking availability on any link is stochastic and cannot be predicted in advance. This issue is discussed in more detail in the following sections. As a preview, Section 3.4 introduces the notion of *strong feasibility*, which ensures that a unique solution exists to the nonlinear flow conservation equations. Section 3.5 then introduces a new feasibility notion which proves more practical for formulating and solving the equilibrium version of the problem, although the behavioral interpretation is slightly less elegant.

3.4 Network Loading

This section has three major goals: (1) formally provide the set of flow conservation equations (including representations of the probability of successfully finding parking on a link and user choice at nodes); (2) define the notion of strong feasibility and show existence and uniqueness of solutions to the flow conservation equations, which is nontrivial since they are nonlinear; and (3) provide a simple "flow-pushing" algorithm for solving this nonlinear system and demonstrate its correctness under nonrestrictive assumptions.

3.4.1 Parking availability probability

Associated with each searching link $(i, j)_S \in A_S$ is a function $p_{ij} : \mathbb{R}^+ \to [0, 1]$ representing the probability that a vehicle seeking to park on this link will find an available space. Particularly, $p_{ij}(x_{ij}^S)$ reflects the proportion of vehicles seeking to park on (i,j) that are able to, and $x_{ij}^S p_{ij}(x_{ij}^S)$ reflects the rate at which vehicles are parking on this link. (If parking is not allowed on this link $p_{ij}(x) \equiv 0$.) This is expressed through the flow conservation constraints which apply to each parking node:

$$x_{ij}^P = x_{ij}^S p_{ij}(x_{ij}^S) \qquad \forall (i,j)_P \in A_P \tag{3.1}$$

$$x_{ij}^{NP} = x_{ij}^S - x_{ij}^P \qquad \forall (i,j)_{NP} \in A_{NP} \tag{3.2}$$

The formulation is general and given in terms of an arbitrary function p_{ij}. However, it is reasonable to assume that each function p_{ij} is nonincreasing (the more vehicles searching for parking, the lower the chance each will find an available space), but that $xp_{ij}(x)$ is strictly increasing (the more vehicles searching for parking, the higher the total rate of parking vehicles will be even as the probability any specific vehicle can find a space decreases). Furthermore, these assumptions are used to establish uniqueness of the solution to the flow conservation equations and convergence of the network loading algorithm. If $p_{ij}(x)$ is differentiable, then $0 < \frac{d}{dx}xp_{ij}(x) \leq 1$ is sufficient for these assumptions to be satisfied. While we believe this assumption to be reasonable, the specific results given below will use weaker conditions when possible.

For each parking link $(i,j)_P$, define its *parking capacity* to be

$$C_{ij,P} = \sup_x \{xp_{ij}(x)\} \tag{3.3}$$

reflecting an upper bound on the rate at which vehicles can leave this link.

In practice, this function is likely related to the number of available parking spaces on a link and the duration vehicles remain parked there. For instance, if a link has S parking spaces, and the parking duration is distributed exponentially with mean μ, then the probability a given vehicle will find an available space when vehicles are searching for parking on that link at rate x is

$$p(x) = \frac{\sum_{k=0}^{S-1} (\mu x)^k / k!}{\sum_{k=0}^{S} (\mu x)^k / k!} \tag{3.4}$$

as derived in the Appendix, which also includes an approximation to facilitate computation when S is large.

3.4.2 Routing and parking search strategies

As introduced above, both route choice and the choice of parking in a vacant space (if available) are described with splitting fractions at each node. In particular, let α_{ij}^d reflect the fraction of travelers arriving at node i en route to destination d who choose to exit on link (i, j). If this link is a regular link, the driver will not park on this link even if an available space is found (presumably in hopes of parking on a link with a shorter walking distance), and this driver will not affect the probability of a parking space being available. On the other hand, if (i, j) is a searching link, then the driver will park on this link should a space become available.

Define the set of *weakly feasible* splitting proportions

$$\Omega = \left(\boldsymbol{\alpha} \in \mathbb{R}_+^{|N_D|(|A_R|+|A_S|)} : \sum_{(i,j) \in \mathcal{A}} \alpha_{ij}^d = 1 \quad \forall i \in N_R, d \in N_D \right) \tag{3.5}$$

As the name suggests, there are additional feasibility concerns beyond these obvious requirements on $\boldsymbol{\alpha}$, discussed in the next section.

Let q_{id} denote the demand for travel to destination $d \in N_D$ from node $i \in N_R$, expressed as a rate in the same units as x. Let the superscript d on a flow variable index the destination-disaggregated flow; for instance x_{ij}^d is the flow on link (i,j) destined for node $d \in N_D$. Clearly

$$x_{ij} = \sum_{d \in N_D} x_{ij}^d \tag{3.6}$$

for all $(i,j) \in \mathcal{A}$. The flow proportions are given by α_{ij}^d, representing the fraction of drivers arriving at node i and heading for destination d which will choose link $(i,j) \in \mathcal{A}$. This gives the disaggregate flow conservation equations for intersection nodes:

$$x_{ij}^d = \alpha_{ij}^d \left(q_{id} + \sum_{(h,i) \in \mathcal{A}} x_{hi}^d \right) \qquad \forall i \in N_R, (i,j) \in \mathcal{A}, d \in N_D \tag{3.7}$$

Flow conservation equations for transfer nodes and destination nodes follow trivially:

$$x_{ij,T,d}^d = x_{ij,P}^d \qquad \forall (i,j) \in A_T, d \in N_D \tag{3.8}$$

$$x_{ij,T,e}^d = 0 \qquad \forall (i,j) \in A_T; d, e \in N_D; d \neq e \tag{3.9}$$

$$\sum_{(i,j)_T \in A_T} x_{ij,T,d}^d = \sum_{i \in N_R} q_{id} \qquad \forall d \in N_D \tag{3.10}$$

In what follows, references to "the flow conservation equations" are to the collection of equations (3.1), (3.2), (3.7), (3.8), (3.9), and (3.10).

As an example of these concepts, refer to panels (a) and (b) of Figure 3.2. Panel (a) shows a network with two parking options from the origin i to destination d: parking along link (i,j) is closer to the origin, but vehicles can only park successfully with probability of 1/2 (for illustrative purposes, assumed constant and independent of searching intensity). Link (k,m) is further away, but parking is assured. Panel (b)

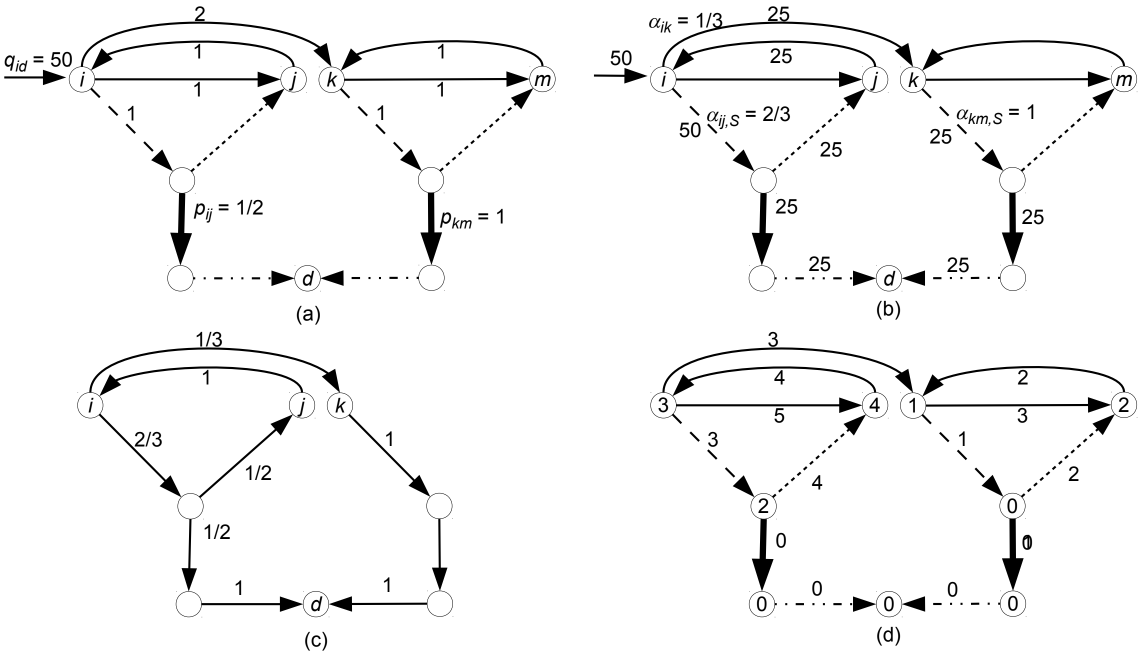


Figure 3.2: Example network for demonstration purposes. (a) Problem data, with link travel times and parking probabilities indicated. (b) Network loading ($\boldsymbol{\alpha}$ and $\mathbf{x}$). (c) The induced graph $\hat{G}$. (d) Travel time labels $\boldsymbol{T}$. Non-labeled links have zero values.

shows the network loading corresponding to $\alpha_{ij,S} = 2/3$, $\alpha_{ik} = 1/3$, $\alpha_{ji} = \alpha_{km,S} = 1$, and all other components of $\boldsymbol{\alpha}$ zero. The reader may verify that the flow conservation equations are satisfied for each link in this network, and note that half of the vehicles park in on link (i, j), and the other half on (k, m).

3.4.3 Feasibility

As the name *weak feasibility* suggests, existence of a solution to the flow conservation equations requires a subtler approach because the p_{ij} functions involve make the system of equations nonlinear. As an example, consider the network in Figure 3.3, and assume that there is a single destination, the inflow q_i is 1, $\alpha_{ij,S} = \alpha_{ji} = \alpha_{km,S} = \alpha_{mk} = 1$ and all other components of $\boldsymbol{\alpha}$ are zero. The probability of finding parking on link (i, j) is half the reciprocal of the amount of searching flow; this means that the rate at which vehicles park on this link is $x_{ij,S} p_{ij}(x_{ij,S}) = 1/2$ if $x_{ij} \geq 1/2$. By contrast, link (k, m) has unlimited parking capacity. The numbers on links indicate costs; links with no adjacent number have zero cost.

Clearly, the given solution is weakly feasible. However, the flow conservation equations for this solution include $x_{ij,S} = 1 + x_{ji}$ and $x_{ji} = x_{ij,NP} = x_{ij,S} - 1/2$; by substitution this requires $x_{ij,S} = x_{ij,S} + 1/2$ which is a contradiction. Intuitively, vehicles are attempting to park on (i, j) at a rate exceeding its capacity and not considering any other options; thus there is no steady-state solution. (There is no issue if the splitting proportions are modified to $\alpha_{ij,S} = \alpha_{ik} = 1/2$, and all vehicles successfully park.) It is clear that cycling is at the root of possible nonexistence of solutions; in an acyclic network a solution can always be constructed by proceeding

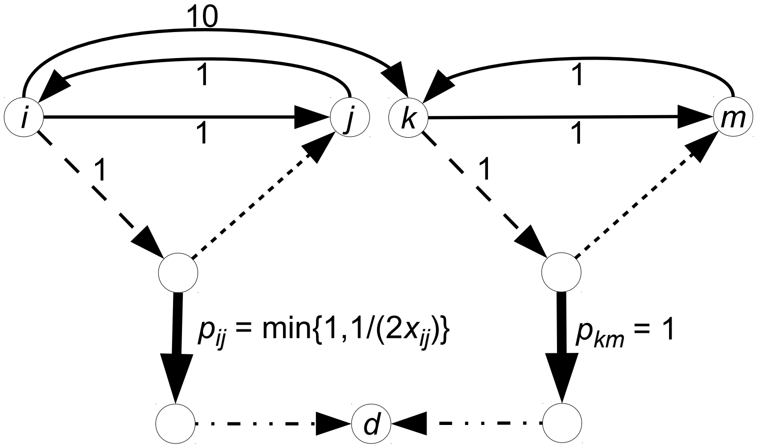


Figure 3.3: Not all weakly feasible solutions are strongly feasible.

in topological order.

Behaviorally, cycles represent the case where a driver returns to a node visited earlier. Traditional equilibrium models exclude this possibility, but modeling parking requires it for several reasons. First, it is common knowledge that drivers searching for on-street parking may "circle the block" several times looking for a convenient space. Second, because parking availability is not deterministic, in general there is no *a priori* route which ensures an available parking space is found, and nodes may need to be revisited. Third, even when such routes do exist, it is likely that routing policies involving cycles may lead to a lower expected travel times than deterministic ones. If the example in Figure 3.3 is modified so that $p_{ij} \equiv 1/2$ and the original splitting proportions $\alpha_{ij,S} = 1$ is used, drivers experience a faster travel time by cycling on (i, j) and (j, i) until a space is found, compared to driving a longer distance to the link (k, m) with guaranteed parking.

The use of constant splitting proportions can be thought of as each driver choosing the next link (and whether to accept an open parking space or not) independent of any such choices in the past, including the current location. This is also consistent with the "sunk cost" interpretation above, but does imply the reset assumption (cf. Provan, 2003; Tang et al., 2014).

Hence, we define a further condition on $\boldsymbol{\alpha}$ called *strong feasibility*: the vector $\boldsymbol{\alpha}$ is strongly feasible if it is weakly feasible, and if the flow conservation equations have a finite solution in $\mathbf{x}$. Let Ω_S be the set of strongly feasible splitting proportions. Below, we characterize this set further and show that under a weak assumption (the functions $xp_{ij}(x)$ are strictly increasing and differentiable) the solution to the flow conservation equations is unique when $\boldsymbol{\alpha} \in \Omega_S$.

If the parking probabilities p_{ij} were constants, the flow conservation equations would form a linear system and strong feasibility would follow in a simple way from a nonsingularity condition. However, in the more interesting variants of the problem the parking probabilities do depend on the intensity of drivers searching for parking, and in such cases the flow conservation equations form a nonlinear system which is harder to characterize. Particularly, strong feasibility depends not only on the $\boldsymbol{\alpha}$ values themselves but also the parking probability functions. Nevertheless, in this section we provide a few results on the existence of strongly feasible solutions, including a necessary condition and a sufficient condition. The issue of strong feasibility will be discussed further in Section 3.5.3 on solution algorithms.

Recall that $C_{ij,P}$ was previously defined as the parking capacity of a link. Further define the network capacity $C = \sum_{(i,j)_P \in A_S} C_{ij,P}$ and the total demand to be $D = \sum_{i \in N} \sum_{d \in N_D} q_{id}$. The section concludes with four results concerning Ω_S: a necessary condition for the existence of a strongly feasible solution, a sufficient condition for the same, a condition for uniqueness of flows corresponding to a strongly feasible solution, and an analytic result which will serve as a lemma for later results.

Proposition 1. *(Necessary condition.)* If the problem instance (defined by G, the p functions, and $\mathbf{q}$) is such that $D > C$, then no strongly feasible solution exists.

Proof. Assume that such an $\boldsymbol{\alpha}$ exists, and let $\mathbf{x}$ be the corresponding feasible solution to the flow conservation equations. By the definition of $C_{ij,S}$, we have $x_{ij,P} \leq C_{ij,S}$ for all $(i,j)_P$; summing over all $(i,j)_P$ yields

$$\sum_{(i,j)_P \in A_P} x_{ij,P} \leq C \tag{3.11}$$

Substituting (3.6) and (3.8), we have

$$\sum_{d \in N_D} \sum_{(i,j) \in A_T} x^d_{ij,T,d} \leq C \tag{3.12}$$

However, by (3.9) and (3.10) we have

$$\sum_{d \in N_D} \sum_{(i,j) \in A_T} x^d_{ij,T,d} = \sum_{d \in N_D} \sum_{i \in N_R} q_{id} = D \tag{3.13}$$

contradicting $D > C$. □

Proposition 2. *(Sufficient condition.)* If the problem instance satisfies $D < C$, the function $xp_{ij}(x)$ is strictly increasing for all $(i,j) \in A$, and the network is strongly connected in the sense that a path exists between any two regular nodes, then at least one strongly feasible solution exists.

Proof. We create such a strongly feasible solution by construction, first identifying $\mathbf{x}$ which satisfy flow conservation and then constructing $\boldsymbol{\alpha}$. Under the stated assumptions $xp_{ij}(x)$ is an invertible function on $[0, C_{ij,P})$; denote this inverse by χ_{ij}. Choose values y_{ij}^{hd} such that $\sum_{(i,j)\in A} y_{ij}^{hd} = q_{hd}$ and $\sum_{h\in N} \sum_{d\in N_D} y_{ij}^{hd} \equiv X_{ij} < C_{ij,P}$ (such a choice is always possible when $D < C$; say by solving an assignment problem). Further define $\xi_{ij} = \chi_{ij}(X_{ij})/X_{ij}$. Now, for each positive y_{ij}^{hd}, identify some path Π from h to i and some cycle Γ from i to itself, only using links in A_R (such a path and cycle exist from the assumption of strong connectivity) and define

$$z_{k\ell}^{hd} = q_{hd}\left([(k,\ell) \in \Pi] + \xi_{ij}\,[(k,\ell) \in \Gamma]\right) \tag{3.14}$$

where the square brackets denote indicator functions using the Iverson notation (equal to one if the quantity in brackets is true, and zero otherwise) Now generate $x_{ij}^d = \sum_{h\in N} z_{ij}^{hd}$ for all $d \in N_D$, and x_{ij} from (3.6). This solution satisfies each of the flow conservation equations and is thus strongly feasibile. □

If $D = C$, then solutions may or may not exist depending on the problem instance.

Proposition 3. If $\boldsymbol{\alpha}$ is strongly feasible and $xp_{ij}(x)$ is a strictly increasing function for all links (i,j), then there is exactly one solution $\mathbf{x}$ to the flow conservation equations.

Proof. Assume not, and let $\mathbf{x}$ and $\mathbf{y}$ be two distinct solutions to the flow conservation equations for a given strongly feasible $\boldsymbol{\alpha}$. Without loss of generality choose some link (i,j) and destination d where $x_{ij}^d > y_{ij}^d$. We can always find such an (i,j) which is a regular link since $x_{ij,T,d} > y_{ij,T,d}$ implies $x_{ij,P}^d > y_{ij,P}^d$, which (by the assumption that $xp_{ij}(x)$ is strictly increasing) implies $x^d_{ij,S} > y^d_{ij,S}$; equation (3.7) thus implies $x^d_{ij} > y^d_{ij}$; a similar argument holds if $x^d_{ij,NP} > y^d_{ij,NP}$.

Equation (3.7) thus implies that $x^d_{hi} > y^d_{hi}$ for some link $(h,i) \in \mathcal{A}$. This argument can be iterated until a complete, finite set of links $\mathcal{A}^+(i) \subseteq \mathcal{A}$ has been obtained for which $(i',j') \in \mathcal{A}^+(i)$ implies $x^d_{i'j'} > y^d_{i'j'}$ and for which a path exists from i' to i in G with strictly postiive $\boldsymbol{\alpha^d}$ components for each link in the path. Sum up the flow conservation equations for each link in $\mathcal{A}^+(i)$; the resulting equations show that the inflows and outflows must balance. Since $xp_{ij}(x)$ is increasing, the difference in outflows $(\mathbf{x^d} - \mathbf{y^d})$ on all links (completed trips plus trips routed to other regular or searching links outside of $\mathcal{A}^+(i)$) must be strictly positive. However, by the completeness of $\mathcal{A}^+(i)$, the corresponding terms of $\mathbf{x} - \mathbf{y}$ for incoming links must be nonpositive, which is a contradiction. □

If $\boldsymbol{\alpha_0}$ is strongly feasible and $xp_{ij}(x)$ is strictly increasing for all links, then the corresponding flows $\mathbf{x}$ are such that $x_{ij,P}$ is strictly less than $C_{ij,P}$ for all $(i,j)_P$.

Proposition 4. If $\boldsymbol{\alpha_0}$ is strongly feasible, $xp(x)$ is differentiable, and $\frac{d}{dx}xp(x) > 0$ for all links, then there exists some neighborhood $\mathcal{B}$ around $\boldsymbol{\alpha_0}$ such that for all $\boldsymbol{\alpha} \in \mathcal{B} \cap \Omega$, $\boldsymbol{\alpha}$ is strongly feasible.

Proof. Under the hypotheses of the proposition, the Jacobian matrix of the flow conservation equations is continuous and irreducibly diagonally dominant (thus nonsingular) in some neighborhood $\mathcal{B}$ around $\boldsymbol{\alpha_0}$. Then, by the implicit function theorem the function $\mathbf{x}(\boldsymbol{\alpha})$ exists and is continuous in $\mathcal{B}$. Thus, any weakly feasible $\boldsymbol{\alpha}$ in this neighborhood is strongly feasible. □

3.4.4 A network loading algorithm

Consider the "flow-pushing" network loading algorithm, depicted in Algorithm 3, which attempts to find a solution to the flow conservation equations. It uses variables η_i^d to reflect the flow conservation "imbalance" (inflow minus outflow) at each node at the time of processing. In the following algorithm η_i is used as a shorthand for $\sum_{d \in N_D} \eta_i^d$. Likewise x_{ij} is always understood in the sense of (3.6) as a shorthand for the sum of the current destination-specific x_{ij}^d values.

Proposition 5. When line 9 of LoadNetwork is first executed (and then for the remainder of the algorithm), $\eta_i \geq 0$ for all i if all $xp_{ij}(x)$ are nondecreasing.

Proof. The result is clearly true when line 9 is executed for the first time. By induction, assume that it is true at some point when line 9 is executed and consider the next steps of the algorithm until line 9 is executed again. The only η values which change until line 9 is executed the next time are those selected in lines 14, 23, and 25. In line 14, η_j is increased by $\alpha_{ij}^d \eta_i^d$ which is nonnegative; in line 23, $x_{ij,NP}^d \geq y_{ij,NP}^d$ if $xp_{ij}(x)$ is nondecreasing; and line 25 simply sets η_i to zero. In all cases the induction hypothesis holds. □

Proposition 6. Algorithm LoadNetwork always terminates in finitely many iterations if $\boldsymbol{\alpha}$ is strongly feasible, $xp(x)$ is differentiable, and $0 < \frac{d}{dx}(xp(x)) \leq 1$ for all links.

Proof. To simply notation, assume there is a single destination; the logic of the proof generalizes naturally to the case of multiple destinations. Define the potential function

Algorithm 3 LoadNetwork$(G, \boldsymbol{\alpha}, \epsilon)$

```
{Arguments are a graph $G$ (as defined in Section 3.3 and equipped with $p_{ij}(x)$ functions), an exogenous vector of splitting proportions $\boldsymbol{\alpha}$ and a convergence tolerance $\epsilon > 0$.}
{Initialization}
for all $(i,j) \in \mathcal{A}, d \in N_D$ do
    $x_{ij}^d \leftarrow 0$
end for
for all $i \in N_R$ do
    $\eta_i^d \leftarrow q_{id}$
end for
while $\max_i \{\eta_i\} > \epsilon$ do
    Choose $i$ such that $\eta_i$ is maximal.
    {Push flow not searching for parking}
    for all $(i,j)_R \in A_R, d \in N_D$ do
        $x_{ij,R} \leftarrow x_{ij,R} + \alpha_{ij,R}^d \eta_i^d$
        $\eta_j \leftarrow \eta_j + \alpha_{ij,R}^d \eta_i^d$
    end for
    {Push flow searching for parking}
    for all $(i,j)_S \in A_S$ do
        $y_{ij,NP}^d \leftarrow x_{ij,NP}^d$ {Temporary value for updating imbalance at $j$}
        $x_{ij,S}^d \leftarrow x_{ij_S}^d + \alpha_{ij,S}^d \eta_i^d \qquad \forall d \in N_D$
        $x_{ij,P}^d \leftarrow x_{ij}^S p_{ij}(x_{ij}^S) \left( x_{ij,S}^d / \sum_{d' \in N_D} x_{ij,S}^{d'} \right) \qquad \forall d \in N_D$
        $x_{ij,T,d}^d \leftarrow x_{ij,P}^d$
        $x_{ij,NP}^d \leftarrow x_{ij,S}^d - x_{ij,P}^d \qquad \forall d \in N_D$
        $\eta_j^d \leftarrow \eta_j^d + x_{ij,NP}^d - y_{ij,NP}^d \qquad \forall d \in N_D$
    end for
    $\eta_i \leftarrow 0$
end while
return $\mathbf{x}$
```

$U = \sum_i \eta_i$. Since the η_i are nonnegative at each step of the algorithm, showing $U \to 0$ implies $\eta_i \to 0$ for all nodes and that flow conservation equations (3.7) are satisfied. Equations (3.1), (3.2), (3.8), and (3.9) are obviously satisfied during the updates "push flow searching for parking." Furthermore, it is not difficult to show that $\sum_{i \in N_R} q_{id} - \sum_{(i,j)_T \in A_T} d^d_{ij,T,d} = \sum_i \eta^d_i$ is an invariant after line 23 is performed, and thus $U = 0$ implies (3.10) is satisfied as well. Since $\boldsymbol{\alpha}$ is strongly feasible and $xp_{ij}(x)$ is strictly increasing, at the unique solution x^* to the flow conservation equations we have $0 < \delta_{ij} \equiv \frac{d}{dx}(x_{ij}p_{ij}(x_{ij}))\big|_{x^*_{ij}}$

Whenever node i is selected for "pushing," flow potentially increases on all regular and searching links emanating from i. Flow pushed onto a regular link (i,j) does not affect U (the imbalance is simply shifted from i to j), whereas some of the flow pushed onto searching links may end up parking and reaching the destination. Consider one such searching link $(i,j)_S$; after performing line 18 the flow $x^d_{ij,S}$ increases by $\alpha_{ij,S}\eta^d_i$. The change in parking flow on $(i,j)_P$ is thus

$$(x^S_{ij} + \alpha^S_{ij}\eta^d_i)p_{ij}(x^S_{ij} + \alpha^S_{ij}\eta^d_i) - x_{ij,S}p_{ij}(x^S_{ij}) \tag{3.15}$$

which is positive since $xp_{ij}(x)$ is strictly increasing. Likewise, the change in flow in $(i,j)_{NP}$ is

$$x_{ij,NP} - y_{ij} = \alpha^S_{ij}\eta^d_i - \left[(x^S_{ij} + \alpha^S_{ij}\eta^d_i)p_{ij}(x^S_{ij} + \alpha^S_{ij}\eta^d_i) - x^S_{ij}p_{ij}(x^S_{ij})\right] \tag{3.16}$$

and it is this amount which is added to η_j. Since $\delta_{ij} < \frac{d}{dx}(xp(x)) \leq 1$ we have $0 \leq x_{ij,NP} - y_{ij} < \alpha^S_{ij}\eta_i(1 - \delta_{ij})$ by the mean value theorem. Therefore, the increase in U from performing lines 16–23 of the algorithm is no greater than $\eta_i \sum_{(i,j)^S \in \mathcal{A}} \alpha^S_{ij}(1-\delta_{ij})$.

When step 24 is performed, U decreases by η_i; therefore the total change ΔU in U from performing steps 10–24 is given by

$$\Delta U \leq \eta_i \left(\sum_{(i,j) \in A} \alpha_{ij} + \sum_{(i,j)^S \in A^S} \alpha_{ij}^S (1 - \delta_{ij}) - 1 \right) \tag{3.17}$$

Since $\sum_{(i,j) \in A} \alpha_{ij} + \sum_{(i,j) \in A^S} \alpha_{ij}^S = 1$ this simplifies to

$$\Delta U \leq -\eta_i \left(\sum_{(i,j)^S \in A^S} \alpha_{ij}^S \delta_{ij} \right) \tag{3.18}$$

For any node i, let $\bar{\alpha}_i = \sum_{(i,j) \in \mathcal{A}} \alpha_{ij}$ denote the total fraction of flow searching for parking on a link leaving node i. (In particular, if $\bar{\alpha}_i = 0$, all vehicles passing through node i are not immediately searching for parking.) Further define $P = \{i \in N_D : \bar{\alpha}_i > 0\}$, $\bar{\alpha} = \min_{i \in P} \bar{\alpha}_i$, and $\delta = \min_{(i,j) \in A} \delta_{ij}$. Let $i(k)$ denote the node selected during the k-th execution of step 10 of the algorithm, and let U_k be the value of U at this point. Whenever $i(k) \notin P$, we have $U_{k+1} = U_k$; otherwise

$$\begin{aligned} U_{k+1} &= U_k + \Delta U \\ &\leq U_k - \eta_{i(k)} \left(\sum_{(i(k),j) \in A^S} \alpha_{i(k),j}^S \delta_{i(k),j} \right) \\ &\leq U_k - \frac{U_k}{|N|} \left(\sum_{(i(k),j) \in A^S} \alpha_{i(k),j}^S \delta_{i(k),j} \right) \\ &= U_k \left(1 - \frac{\bar{\alpha}\delta}{|N|} \right) \end{aligned}$$

where the second inequality follows because $i(k)$ is a node with maximal η value. Thus $\lim_{k \to \infty} U_k = 0$ unless $i(k) \in P$ only finitely many times. However, in that case, there is a cycle of nodes which does not send any flow onto searching links and the flow conservation equations cannot be satisfied, contradicting the assumption of strong feasibility. □

At termination, the algorithm clearly produces a flow $\mathbf{x}$ which deviates from the flow conservation equations by no more than ϵ for each link. A trivial corollary is that if the potential function U does not convege linearly to zero, then the given solution is not strongly feasible. This can be used as a strong feasibility test, much in the same way that nonconvergence of label-correcting shortest path algorithms can be used to test for existence of negative-cost cycles.

3.4.5 Interpretation from a driver's perspective

The parking model described in this chapter is formulated in terms of aggregate flows $\mathbf{x}$. However, this model can be viewed through the perspective of an individual driver as well. This section formulates this as a Markov chain, and shows that if the aggregate flows $\mathbf{x}$ are induced by a strongly feasible $\boldsymbol{\alpha}$, then the probability that each driver reaches his or her destination asymptotically approaches 1 after a sufficient amount of driving and searching time.

In this subsection, consider an individual driver who departs origin node $o \in N_R$ destined for node $d \in N_D$ (thus $q_{od} > 0$). Also assume that the splitting proportions $\boldsymbol{\alpha}$ are strongly feasible, so the aggregate flow vector $\mathbf{x}$ is well-defined and uniquely determined. The progression of this driver through the network can be expressed as a Markov chain, where the states are the set of all nodes $\mathcal{N}$, the initial state is o, and the transition probabilities are given as follows: for each regular node

$i \in N_R$ and outgoing link $(i, j) \in \mathcal{A}$, the probability of transitioning from i to j is given by α_{ij}. For each parking node ij_P corresponding to link (i, j), the probability of transitioning from ij_P to the corresponding transfer node ij_T is p_{ij} and to the regular node j is $1 - p_{ij}$. For each transfer node, the transition probability to the destination d is 1, and 0 for all other nodes. The destination node d is an absorbing state.

We aim to show that if $\boldsymbol{\alpha}$ is strongly feasible, the Markov chain reaches d with probability asymptotically reaching 1. To assist with this, define the set $\hat{\mathcal{N}}$ denoting all states of the Markov chain reached with positive probability when starting from o. More formally, $\hat{\mathcal{N}}$ is defined as the smallest set satisfying $o \in \hat{\mathcal{N}}$ and the condition $i \in \hat{\mathcal{N}} \Rightarrow j \in \hat{\mathcal{N}}$ for all $(i, j) \in \mathcal{A}$ with $x^d_{ij} > 0$. The asymptotic result is obtained in two steps, first showing that strong feasibility of $\boldsymbol{\alpha}$ implies that the destination d is part of the reachable set $\hat{\mathcal{N}}$, then showing that this probability approaches 1 as the driver spends longer searching.

Proposition 7. If $\boldsymbol{\alpha}$ is strongly feasible, then $d \in \hat{\mathcal{N}}$.

Proof. The flow conservation equations above are given in terms of links; however together with weak feasibility they imply the following relations for nodes in N_R, N_P,

N_T, and N_D, respectively:

$$\begin{aligned}
q_{id} + \sum_{(h,i)\in\mathcal{A}} x^d_{hi} &= \sum_{(i,j)\in\mathcal{A}} x^d_{ij} && \forall i \in N_R, d \in N_D \\
x^d_{ij,S} &= x^d_{ij,P} + x^d_{ij,NP} && \forall (i,j)_S \in A_S, d \in N_D \\
x^d_{ij,P} &= x^d_{ij,T,d} && \forall (i,j)_P \in A_P, d \in N_D \\
\sum_{(i,j)^d_T \in A_T} x^d_{ij,T,d} &= \sum_{k\in N_R} q_{kd} && \forall d \in N_D
\end{aligned}$$

The left-hand side of each equation represents the "inflow" to a node, while the right-hand side represents the "outflow." Summing these equations over a subset of nodes thus equates the total inflow to the subset to the total outflow. The flow x^d_{ij} on links connecting two nodes of such a subset appears on both sides, and thus can be canceled. By definition of $\hat{\mathcal{N}}$, in such a sum, no positive x^d_{ij} appears on the right-hand side without also appearing on the left-hand side. We thus have

$$\sum_{i\in N_R\cap\hat{\mathcal{N}}} q_{id} + \sum_{(i,j)\in\mathcal{A}: i\notin\hat{\mathcal{N}}} x^d_{ij} = \left[d \in \hat{\mathcal{N}}\right] \sum_{i\in N_R} q_{id} \tag{3.19}$$

again using the bracket notation for the indicator function. Since $q_{od} > 0$, the left-hand side of the equation is positive. Strong feasibility implies that this equation is consistent and thus the right-hand side must be positive as well; this is only possible if $d \in \hat{\mathcal{N}}$. □

Proposition 8. Let $P(t)$ be the probability that the (absorbing) destination state d has been reached after t transitions from the initial state o, with transition probabilities based on a given $\boldsymbol{\alpha}$. If $\boldsymbol{\alpha}$ is strongly feasible, then $\lim_{t\to\infty} P(t) = 1$.

Proof. Let $\hat{G}$ be the subgraph of G induced by the node set $\hat{\mathcal{N}}$. (See Figure 3.2(c), with transition probabilities for the Markov chain indicated on the links.) By Proposition 7 and the definition of $\hat{\mathcal{N}}$ there is at least one simple path in $\hat{G}$ from each node $i \in \hat{\mathcal{N}}$ to the destination d, and the probability of successivly transitioning from state i to d by traversing this path is given by the product of the transition probabilities along successive arcs (the relevant α value for regular or searching links, and p or $1-p$ for parking and no parking links, respectively). Again by definition of $\hat{\mathcal{N}}$ at least one such path has strictly positive traversal probability. Let K be the maximum length of such a simple path, and let $\hat{p} > 0$ be the least traversal probability of any such path. Regardless of the initial state, after K transitions we have $P(K) \geq \hat{p}$ since the destination may be reachable via multiple paths in $\hat{G}$. After nK transitions, we have $P(nK) = 1 - (1 - P(nK)) \geq 1 - (1 - \hat{p})^n$ with the inequality following because every K steps the probability of reaching d is at least $\hat{p}$. Since d is absorbing, $P(t)$ is nondecreasing and $P(t) \geq P(K\lfloor t/K \rfloor) \geq 1 - (1-\hat{p})^{\lfloor t/K \rfloor}$ which asymptotically approaches 1 when t is large. □

3.5 Equilibium

This section introduces the equilibrium framework built upon the network loading from Section 3.4. Its major goals are (1) expressing the expected travel time to the destination as a function of the choices made at each node; (2) developing a preliminary variational inequality formulation of the equilibrium problem; (3) establishing a more practical approach to feasibility than strong feasibility, and its suitability for equilibrium; and finally (4) a convex combinations heuristic for solving the equilibrium problem.

3.5.1 Cost labels

As a first step toward development of the equilibrium model, we calculate the costs associated with each choice travelers can make. These calculations are made separately from the network loading, so we can assume that a flow vector $\mathbf{x}$ satisfying the flow conservation equations is given and held constant. For each destination d and regular link $(i,j)_R$, let $T^d_{ij,R}$ represent the expected remaining cost among travelers from node i to the destination d, including the cost on link $(i,j)_R$ and remaining costs to the destination:

$$T^d_{ij,R} = t_{ij,R} + \sum_{(j,k)\in A_R \cup A_S} \alpha^d_{jk} T^d_{jk} \tag{3.20}$$

where T^d_{jk} reflect downstream costs from d. For searching links, the corresponding labels are defined as:

$$T^d_{ij,S} = t_{ij,S} + p_{ij}\left(t_{ij,P} + t_{ij,T,d}\right) + (1 - p_{ij}) \sum_{(j,k)\in A_R \cup A_S} \alpha^d_{jk} T^d_{jk} \tag{3.21}$$

Note that p_{ij} can be treated as constant in (3.21) since $\mathbf{x}$ is given and fixed. Hence (3.20) and (3.21) form a linear system in T^d_{ij} and $T^d_{ij,S}$ with $|N_D|(|A_R| + |A_S|)$ equations and variables. While this linear system can be solved directly and without excessive difficulty (since transportation networks are relatively sparse), a network algorithm similar to LoadNetwork performs even better. This algorithm, CalculateCostLabels is presented as Algorithm 4. In this algorithm, $\boldsymbol{\alpha}$ and $\boldsymbol{p}$ are to be calculated from the flow vector $\mathbf{x}$ provided as an argument. Given a vector $\mathbf{T}$ of cost labels which do not necessarily solve (3.20) and (3.21), the "imbalance" ζ_i of node i is calculated by summing the absolute difference between the left and right hand sides of (3.20) and (3.21) across all outgoing links and destinations:

$$\zeta_i = \sum_d \left(\sum_{(i,j)_R \in A_R} |\text{RHS(3.20) - LHS(3.20)}| + \sum_{(i,j)_S \in A_S} |\text{RHS(3.21) - LHS(3.21)}| \right) \tag{3.22}$$

At each iteration a node i with maximum ζ_i is chosen, and the labels T_{ij}^d of outgoing links are calculated using (3.20) and (3.21). Convergence of this algorithm is not difficult to show, since each iteration reduces the distance between the current $\mathbf{T}$ and the solution $\mathbf{T}^*$ of the linear system by as least as much as the Gauss-Seidel method, which is sure to converge since the linear system is irreducibly diagonally dominant (as follows from strong feasibility).

Algorithm 4 CALCULATETIMELABELS$(G, \mathbf{x}, \epsilon)$

```
{Arguments are a graph $G$ (as defined in Section 3.3), an exogenous vector of link flows $\mathbf{x}$ satisfying flow conservation, and a convergence tolerance $\epsilon > 0$.}
{Initialization}
$T_{ij}^d \leftarrow 0 \quad \forall (i,j) \in A_R \cup A_C, d \in D$
$T_{ij,S}^d \leftarrow 0 \quad \forall (i,j)_S \in A_S, d \in D$
$\zeta_i \leftarrow (3.22) \quad \forall i \in N_R$
while $\max_i \{\zeta_i\} > \epsilon$ do
    Choose $i$ such that $\zeta_i$ is maximal.
    {Update labels for $i$}
    $T_{ij,R}^d \leftarrow$ RHS(3.20) $\quad \forall (i,j)_R \in A_R, d \in N_D$
    $T_{ij,S}^d \leftarrow$ RHS(3.21) $\quad \forall (i,j)_S \in A_S, d \in N_D$
    $\zeta_j \leftarrow (3.22) \quad \forall \{j \in N : (i,j)_R \in A_R\}$
    $\zeta_i \leftarrow 0$
end while
return $\mathbf{T}$
```

3.5.2 Equilibrium formulation

The cost labels defined in the previous subsection lead to formulating an equilibrium principle, that a given alternative will be used only if its cost is minimal among all alternatives (implying equality of cost when multiple alternatives are used). Let $\boldsymbol{\alpha}$ be strongly feasible; then there is a unique network loading $\mathbf{x}(\boldsymbol{\alpha})$ based on the flow conservation equations, and a unique set of travel time labels $\mathbf{T}(\mathbf{x}(\boldsymbol{\alpha}))$ based on (3.20) and (3.21); for brevity, this latter relation is abbreviated $\mathbf{T}(\boldsymbol{\alpha})$.

A strictly positive α_{ij}^d value suggests that some travelers heading to d and passing through node i are leaving via node j; at equilibrium this is only possible if T_{ij}^d is no greater than $T_{ij'}^d$ for any other outgoing link (i, j'). That is, a strictly feasible $\boldsymbol{\alpha}$ is defined to be an equilibrium if

$$\alpha_{ij}^d > 0 \Rightarrow T_{ij}^d = \min_{(i,j') \in A_R \cup A_S} \{T_{ij'}^d\} \tag{3.23}$$

is satisfied for all (i, j) and d.

Proposition 9. If $xp(x)$ is differentiable and $\frac{d}{dx}xp(x) > 0$, then a strongly feasible vector of splitting proportions $\boldsymbol{\alpha}^*$ is an equilibrium if $\mathbf{T}(\boldsymbol{\alpha}^*) \cdot (\boldsymbol{\alpha}^* - \boldsymbol{\alpha}) \leq 0$ for all $\boldsymbol{\alpha} \in \Omega_S$.

Proof. Assume that $\boldsymbol{\alpha}^*$ is not an equilibrium. Then there exist two distinct links $(i, j) \in \mathcal{A}$ and $(i', j') \in \mathcal{A}$ and some $d \in N_D$ such that $\alpha_{ij}^{*d} > 0$ and $T_{ij}^d < T_{i'j'}^d$. By Proposition 4, the vector $\boldsymbol{\alpha} = \boldsymbol{\alpha}^* + \delta\boldsymbol{\Delta}$ is strongly feasible for sufficiently small positive δ, where Δ has two nonzero components: +1 for (i', j') and d, and -1 for (i, j) and d; in this case $\mathbf{T}(\boldsymbol{\alpha}^*) \cdot \boldsymbol{\alpha}^* > \mathbf{T}(\boldsymbol{\alpha}^*) \cdot \boldsymbol{\alpha}$ and the variational inequality is not satisfied. □

Unfortunately, the utility of this formulation is limited by the irregular structure of Ω_S: Proposition 4 suggests that Ω_S is not a closed set, unless it includes the entire boundary of Ω; and in any case, Ω_S need not be convex. This is demonstrated in Figure 3.4, where the left and right parking links have capacity 5, the center parking link has capacity 1, and the total demand is 4. Panel (a) shows the network with relevant parameters (link styles are the same as in Figure 3.3), and panels (b) and (c) respectively illustrate two strongly feasible solutions in which all vehicles park either at the left or right links. Panel (d) illustrates the average of these two strongly feasible solutions; in this solution, 1 vehicle attempts to park at the left link, 1 vehicle attempts to park at the right link, and 2 vehicles attempt to park at the middle link. The number of vehicles attempting to park at the middle link exceeds its capacity, and the network structure forces drivers to continue searching on the link they initially chose, so this solution is not strongly feasible.

These properties of Ω_S are not favorable. The vast majority of variational inequality results and algorithms, including fixed-point theorems to establish equilibrium existence, require the feasible set to be closed and convex (cf. Facchinei and Pang, 2003). To address this, the following subsection describes a modification to the network structure and feasible set which are more useful.

3.5.3 A practical approach to feasibility

As shown in the previous section, Ω_S need not be a closed or convex set, which poses difficulties both for demonstrating existence of an equilibrium solution and for finding such a solution if it exists. Since Ω is closed convex, the difficulty

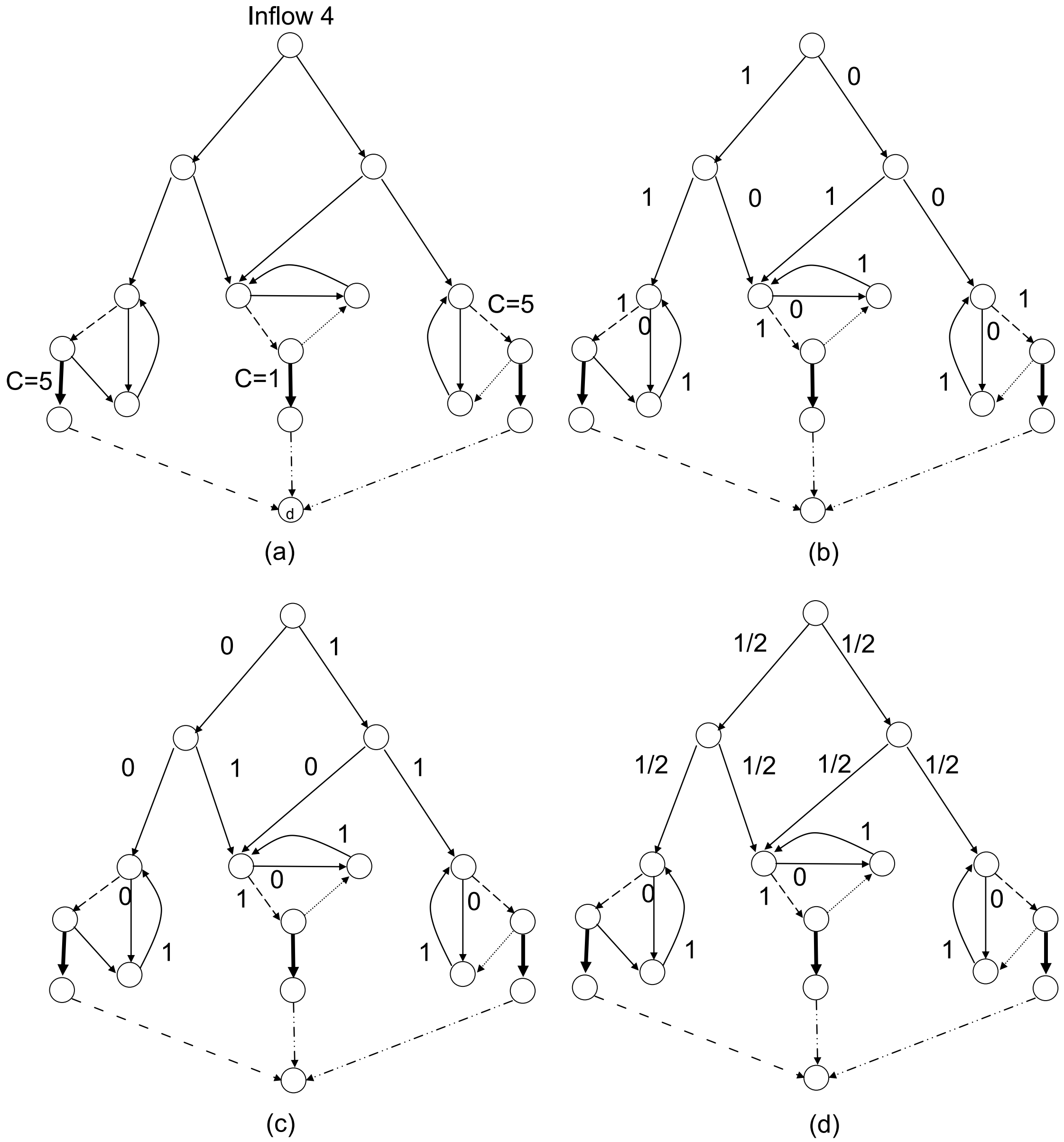


Figure 3.4: The set Ω_S is not convex.

is introduced by requiring a finite solution to the flow conservation equations. An alternative method is to transform the network further by adding a direct link from each node i to each destination d; let each such link have a high, but finite travel time $\bar{t}$. Denote the collection of these links $\bar{A}$, and the new network $\bar{G}$. Given a small positive parameter ϵ, define the set $\Omega_\epsilon = \left\{\alpha \in \Omega : \alpha_{ij} \geq \epsilon \quad \forall (i,j) \in \bar{A}\right\}$.

Intuitively, the new links in $\bar{A}$ create direct connections to the destination with high cost. One possible behavioral interpretation is that these links represent "failed trips" where travelers give up searching for parking; the connection to the destination is only enforced to maintain flow conservation. Whenever travelers pass through a node, a fraction ϵ of the drivers will give up searching and terminate their trips. This explanation is not perfect (since the vehicles disappear from the network at this point in time), but facilitates the solution process — as shown below, any set Ω_ϵ is closed convex and admits a unique solution to the flow conservation equations. Furthermore, link flows $\mathbf{x}$ and travel time labels $\mathbf{T}$ corresponding to $\boldsymbol{\alpha} \in \Omega_S$ in G can be approximated arbitrary closely by splitting proportions in Ω_ϵ and $\bar{G}$ as $\epsilon \to 0$.

Proposition 10. Given any $\epsilon > 0$ and $\boldsymbol{\alpha} \in \Omega_\epsilon$, at least one solution exists to the flow conservation equations in $\bar{G}$.

Proof. Apply algorithm LoadPolicy. When processing node i, at least $\epsilon\eta_i$ units of flow reach the destination; since i is the node with maximal η_i value, the reduction in the potential function U is at least $\epsilon U/|N|$ (even neglecting any flow which may find parking), and the algorithm converges linearly to a solution of the flow conservation equations. □

Proposition 11. Given any $\epsilon > 0$ and $\boldsymbol{\alpha} \in \Omega_\epsilon$, the solution to the flow conservation equations in $\bar{G}$ is unique.

Proof. Identical to Proposition 3. □

Proposition 12. Let $\mathbf{T_0}$ and $\mathbf{x_0}$ correspond to $\boldsymbol{\alpha_0} \in \Omega_S$ in G. For any $\delta > 0$, there exists an $\epsilon > 0$ and $\mathbf{T}_\epsilon$ and $\mathbf{x}_\epsilon$ corresponding to some $\boldsymbol{\alpha_\epsilon} \in \Omega_\epsilon$ in $\bar{G}$ such that $||\mathbf{T}_\epsilon - \mathbf{T_0}|| < \delta$ and $||\mathbf{x}_\epsilon - \mathbf{x_0}|| < \delta$.

Proof. Define the projection operator $\Pi(\mathbf{z}, K)$ returning the point $\mathbf{z}^*$ in K minimizing $||\mathbf{z}^* - \mathbf{z}||$; this point is unique if K is convex. Consider some sequence $\{\epsilon_k\}$ converging to zero, and let $\boldsymbol{\alpha_k} = \Pi(\boldsymbol{\alpha_0}, \Omega_{\epsilon_k})$. Clearly as $k \to \infty$, $||\boldsymbol{\alpha_k} - \boldsymbol{\alpha_0}|| \to 0$. By the implicit function theorem, the function $\mathbf{x}(\boldsymbol{\alpha})$ is continuous when $\boldsymbol{\alpha}$ is strongly feasible, so as $\boldsymbol{\alpha}_\mathbf{k} \to \boldsymbol{\alpha_0}$ we have $\mathbf{x_\epsilon} \equiv \mathbf{X}(\boldsymbol{\alpha_k}) \to \mathbf{x_0}$. Finally, since $\mathbf{T}$ is obtained from $\mathbf{x}$ by solving a nonsingular linear system, the mapping $\mathbf{T}(\mathbf{x})$ is continuous and $\mathbf{T}_\epsilon \equiv \mathbf{T}(\boldsymbol{\alpha}_\mathbf{k}) \to \mathbf{T_0}$. □

Proposition 13. For any $\epsilon > 0$, there exists at least one equilibrium solution in Ω_ϵ.

Proof. The variational inequality $\mathbf{T}(\boldsymbol{\alpha}^*) \cdot (\boldsymbol{\alpha}^* - \boldsymbol{\alpha}) \leq 0$ for all $\boldsymbol{\alpha} \in \Omega_\epsilon$ is defined on a closed, convex set. Furthermore the mapping $\mathbf{T}$ is continuous in α as shown in the proof of Proposition 12. Theorem 2.1.1 of Facchinei and Pang (2003) thus applies and the variational inequality has at least one solution. □

3.5.4 An equilibrium heuristic

Although $\mathbf{T}$ is a continuous function of $\boldsymbol{\alpha}$ on Ω_ϵ, ensuring existence of an equilibrium solution, the function lacks other favorable properties (such as monotonicity). Nevertheless, a convex combinations heuristic seems to work well; this is shown in Algorithm 5. This algorithm involves iteratively performing a network loading, calculating travel time labels, identifying a "target" flow proportions vector $\boldsymbol{\alpha}^*$ placing maximal weight on minimum-cost choices for each node and destination, and updating the flow proportions by taking a convex combination of the current and target vectors.

Convergence is determined with a gap function measuring deviation from the equilibrium principle. One such gap function is the *average excess cost*

$$AEC = \frac{\sum_{i \in N} \sum_{d \in N_D} \sum_{(i,j) \in A \cup A_S} \left(T_{ij}^d - \min_{(i,j') \in A_R \cup A_S} T_{ij'}^d \right)}{D} \tag{3.24}$$

As shown in the demonstrations below, the simple choice of $\lambda = 1/k$ appears to work satisfactorally, where k is the number of times algorithm LoadNetwork has been performed when step 8 is executed.

3.6 Demonstration

The model is demonstrated using the well-known Sioux Falls network (Bar-Gera, 2014) with 24 nodes, 76 links, and 24 origins and destinations. A schematic of this network is shown in Figure 3.5. An instance of the parking search equilibrium problem was generated from this network using the following procedure. First, a destination was created for each zone centroid; the walking time from any link to

Algorithm 5 $\textsc{ConvexCombinations}(G_\epsilon, \epsilon_x, \epsilon_T, \epsilon_{AEC})$

```
{Arguments are a graph $G_\epsilon$ (as defined in Section 3.5.3), and tolerances $\epsilon_x$, $\epsilon_T$, and $\epsilon_{AEC}$ for the network loading, travel time calculation, and master algorithm respectively.}
{Initialization}
Choose some $\boldsymbol{\alpha} \in \Omega_\epsilon$
$\mathbf{x} \leftarrow \textsc{LoadNetwork}(G_\epsilon, \boldsymbol{\alpha}, \epsilon_x)$
$\mathbf{T} \leftarrow \textsc{CalculateTimeLabels}(G_\epsilon, \mathbf{x}, \epsilon_T)$
while $AEC > \epsilon_{AEC}$ do
    Choose some $\boldsymbol{\alpha}^* \in \arg\min_{\boldsymbol{\alpha} \in \Omega_\epsilon} \{\alpha \cdot \boldsymbol{T}\}$
    $\boldsymbol{\alpha} \leftarrow \lambda \boldsymbol{\alpha}^* + (1 - \lambda)\boldsymbol{\alpha}$ for some $\lambda \in [0, 1]$.
    $\mathbf{x} \leftarrow \textsc{LoadNetwork}(G_\epsilon, \boldsymbol{\alpha}, \epsilon_x)$
    $\mathbf{T} \leftarrow \textsc{CalculateTimeLabels}(G_\epsilon, \mathbf{x}, \epsilon_T)$
end while
return x
```

each destination is proportional to the Euclidean distance between the origin and destination nodes, using the node coordinates associated with the Sioux Falls network. We assume parking is allowed on each link in the network, with the number of spaces proportional to the physical length of the link. The mean duration of parking and parking cost are uniform on all links.

The algorithms were implemented in C and run on a 2.60 GHz Intel machine with 4 GB memory running Windows 7. Figure 3.6 shows the convergence rate of the algorithm, reporting the average excess cost obtained after a given amount of computation time has elapsed. The convex combinations method with step size inversely proportional to iteration count seems to function acceptably, at least in this small network, and the gap function decreases steadily as the solution stabilizes.

Figure 3.7 shows the importance of considering parking in the assignment and

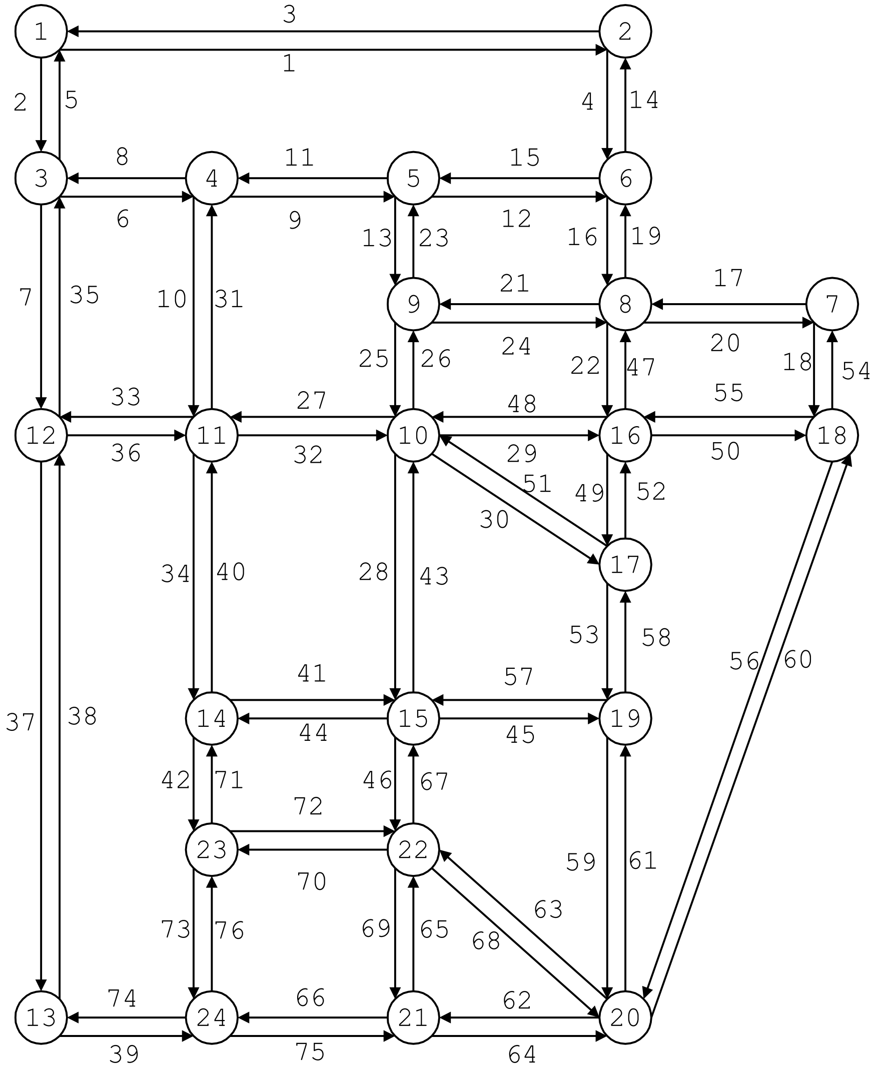


Figure 3.5: The Sioux-Falls Network

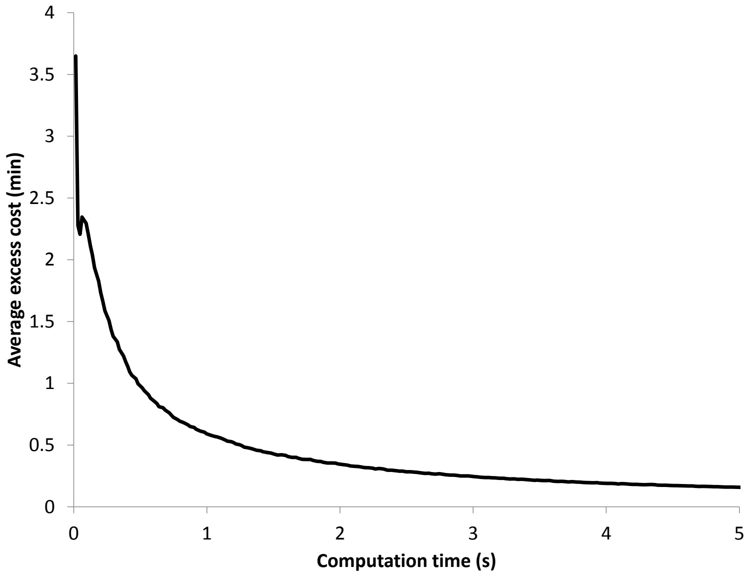


Figure 3.6: Progress of the convex combinations heuristic

equilibrium process. The left panel of the figure shows the link volumes when parking search is neglected[1] and trips can park at any location with probability 1 (setting $p_{ij}(x) \equiv 1$ in the algorithms). The right panel shows link volumes after accounting for parking search; there is a significant increase in these volumes. Network-wide, the total vehicle-hours traveled (VHT) increased from 24310 to 29059, an increase of 19.5%. Figure 3.8 shows the fraction of flow on each link searching for parking (as opposed to driving through) and probability of finding parking at different links. In particular, notice how available parking is scarcer in the city center, and more plentiful around the network perimeter. This model thus provides quantitative estimates of

[1] All links in the Sioux Falls network have a "mirror" with the tail and head node reversed; for clarity in the figures, the higher value of the two mirror links is shown.

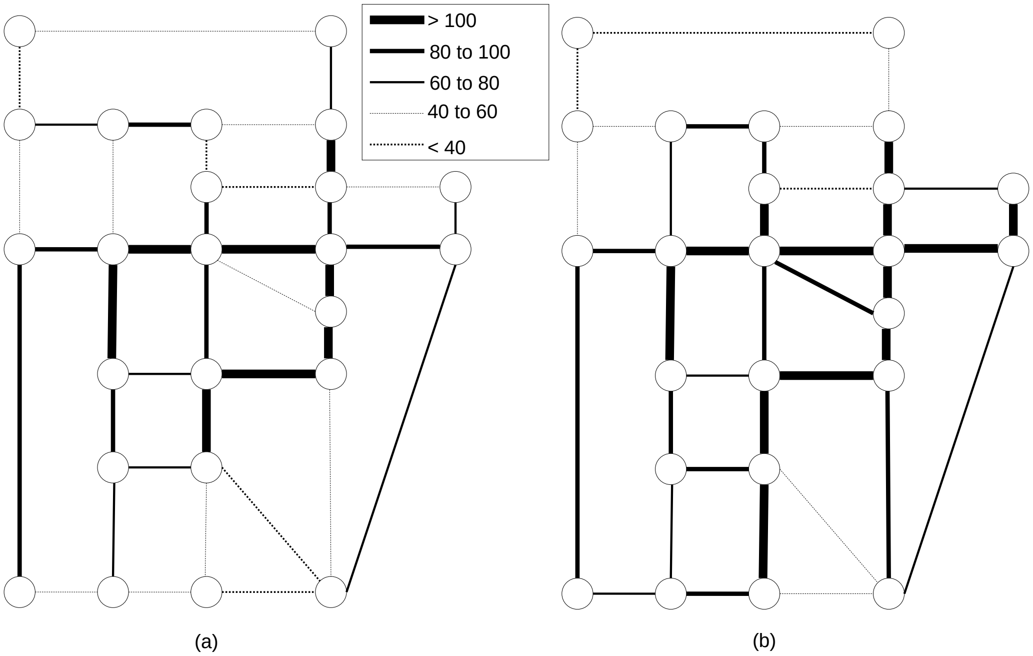


Figure 3.7: Traffic volumes (a) without considering parking search and (b) considering search

parking availability in different regions of the network.

Two sensitivity analyses were conducted to the input parameters. In the first, the mean parking duration is varied, perhaps reflecting changes in parking time limits. As shown in Figure 3.9, increasing the duration of parking results in increases in both average driving time (as vehicles search longer before finding an available space) and average walking time (as some drivers accept parking spaces further from their destination). These trends are roughly convex, indicating greater sensitivity to parking duration when parking is scarcer than when it is more plentiful. The second sensitivity analysis varies the relative cost of walking time to driving time, perhaps

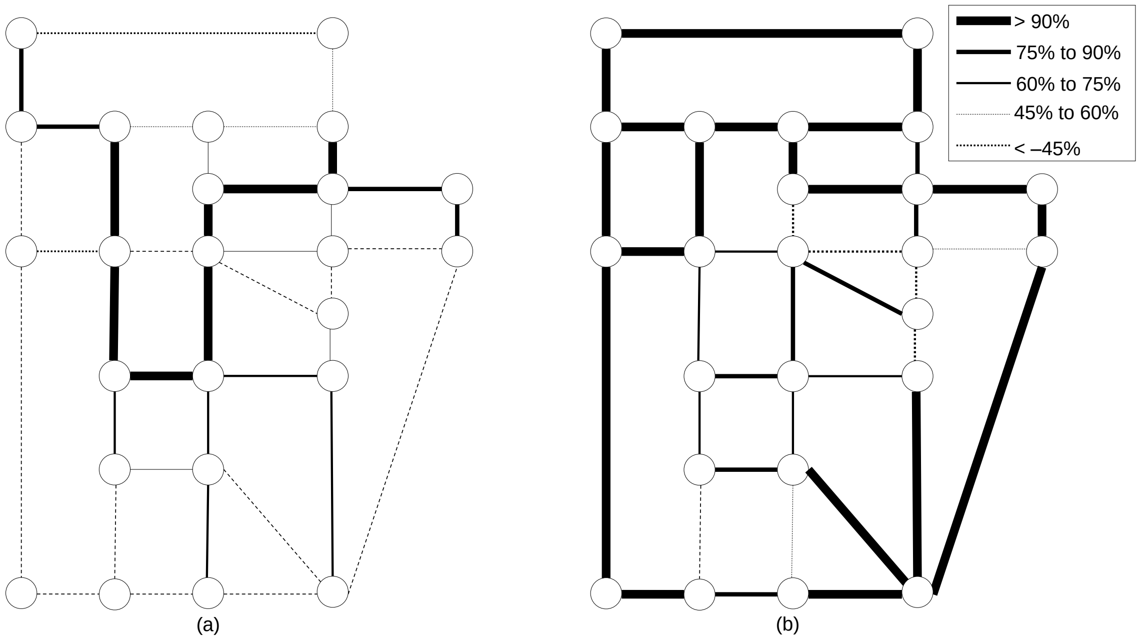


Figure 3.8: (a) Fraction of link volume searching for parking and (b) probability of successfully finding parking

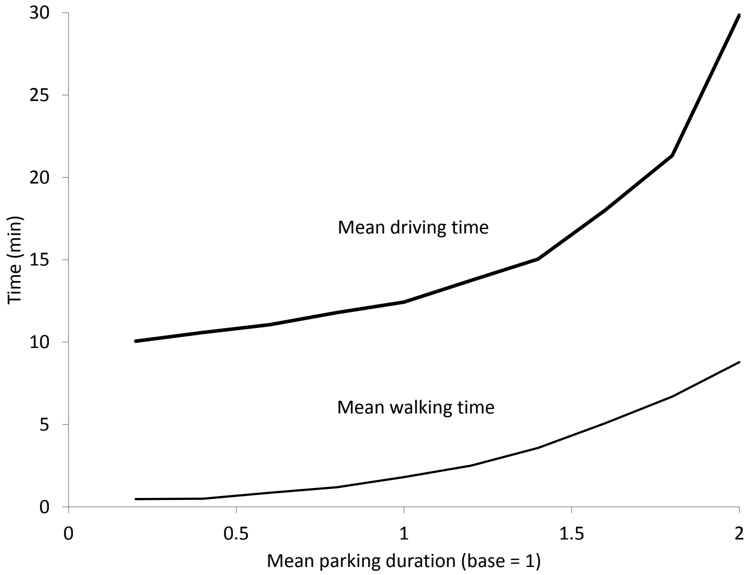


Figure 3.9: Sensitivity analysis of travel times to mean parking duration

reflecting changes due to weather (in poor weather, drivers may prefer to spend more time searching inside the vehicle and less time walking). (Figure 3.10.) As walking becomes onerous, the amount of driving time increases and the average amount of walking time decreases, but only slightly; this suggests that in the base conditions, drivers are already parking relatively close to their destinations.

3.7 Model Validity and Sensitivity Analysis

Introducing parking behavior into traffic assignment models, we expect this new type of traffic assignment model can reflect much more realistic flow distribution and parking behaviors. Compared to those without considering parking search, we expect the following conditions and behaviors for the new model: First, because of

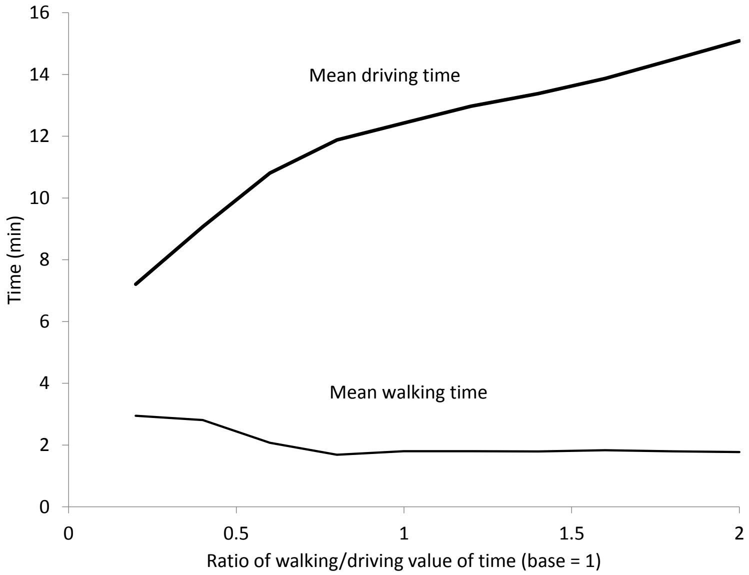


Figure 3.10: Sensitivity analysis of travel times to value of walking time

cycling for parking spaces, the total system travel time should be higher, and most of the links should have some "extra" flow. Second, vehicles should park relatively close to their final destination. Third, if the supply of parking on some links is reduced, some of the vehicles parking on these links will likely to park on nearby links, while increasing parking supply may cause an increased parking rate on those links.

The sensitivity analysis focuses on how the assignment results will change if travel demand, parking price, parking space capacity and link status change. As with classic traffic assignment models, when we change these parameters, the link flow patterns may change. For instance, if we increase the parking rate of a link, we would expect less vehicles parking on this link. If the total travel demand becomes less, we expect the total vehicles that are searching for parking would also be less. However, for some special links, the parking searching flows may increase because the increased flows may be transfered from other links. Another issue this section deals with is what the impact range and quantity that a parameter change has. For instance, if we increase the parking rate on a link which usually has many vacant parking spaces, the impact on flow patterns and parking distributions could be very little. If the increase happens on a very crowed link, the impact may be large on nearby links, but still very little on links that are far away.

All the tests that are used to check if our model can successfully reflect the parking search behavior are taken on the Sioux-Falls network as in Figure 3.5 and listed in Table 3.1. In the Sioux-Falls network each node and link has been labeled.

3.7.1 Demand Change Test

Several concepts need to be clarified before we discuss the demand impacts. By "demand impacts" we mean the impacts that are caused by the change to the origin-destination matrix. The trip demand for an origin and destination pair is the number of trips generated from the origin to the destination. In our model, trip demands are constants. The total parking flow is equal to the total trip demand because all the trips need to end with parking at some location. The parking demand on a link equal the searching flow on that link and it is a variable. The parked vehicles on a link are part of the searching-parking flow which successfully find parking spaces and parked on that link. The concept of parking distribution describes on which link a vehicle parks and it also shows the number of vehicles that park on a link.

As described in Table 3.1, we want to see how changes to the travel demand

Table 3.1: Sensitivity Analysis of the Static Parking Model on Sioux-Falls Network

Test Type	**Test Goal**	**Test Detail**
Demand change test	Test how the changes to demand would impact the parking distribution, link flow pattern and the probability of successfully finding parking	Reduce the demand to destination 10, 15, 16, 17 and 19 by 10, 30, 50 and 80 percent. The reason of choosing these destinations is that the parked vehicles and passing flows are concentrated on the links near to these destinations. Notice that the destinations are not the nodes, but they are close to the nodes that are with the same numbers
Price test	Check how the parking price would impact parking distribution, link flow pattern and the probability of successfully finding parking	only increase the cost on links connecting nodes 10 and 16 by 10, 30, 50, 80, and 100 percent.
Parking space capacity test	Check if increase the parking capacity on some specific links, how it will impact the flow pattern and parking distribution	Increase the number of parking spaces on link 1 and 3 by 10, 30, 50, 80 and 100 percent; Increase the number of parking spaces on link 29 and 48 by 10, 30, 50, 80 and 100 percent.

will impact the flow pattern and the parking distribution. The links chosen in this test have high flows and parking demands with the original travel demand. We expect that if the parking demand on these links are reduced, both the parked vehicles and the passing through flows should be decreased or at least not be increased. First we check how the flows and parking demands change on the whole network, then we choose several different types of links to compare the impacts.

Figure 3.11 shows the link flow change distribution with the impact of demand decrease. The sub-figures correspond to 10%, 30%, 50% and 80% decrease in demand to destinations 10, 15, 16, 17, 19, respectively. The horizontal axis indicates the link flow change range in terms of percentage while the vertical axis means how many links fall into that range. This figure shows that when the travel demand to these destinations is decreased, most of the links on this network have lower link flows. However, there are still some links even have higher flows, but with a mild increase. These four figures also tell that when the demand reduction is low, say 10 percent, the distributions of the links are more concentrated with a negatively skewed distribution, while when the reduction is very large the distribution is scattered more widely. This means if we reduce the demand to some destination a little, most of the links have little change in link flows, but if we reduce it very much, the impact will be big and very different among links. With further check we found that the flows on the links near to these destinations are decreased a lot. On the other hand, the flows on the links that are far from these destinations do not change much. Figure 3.12 demonstrates this effect more clearly. The left sub figure shows that flows on links (1,3),(3,12) and (12,13) do not change much even those destinations have a 80 percent demand reduction because these links are far from the demand-reduced destinations. The right sub figure just shows the opposite, since the links in this figure are near to the demand-reduced destinations, their link flow highly depend on their travel demand.

Generally speaking, reducing demand to a destination causes less flows on links near to that destination. However, there are still several links having more flows than the original case. For instance, link (11,10) and link (11,12) have more flows after reducing travel demand. The reason is that the demand change reduces congestion level on the links that are near to destination 10, 15, 16 and 17 which attracts more passing through traffic onto the two links.

Reducing demand can also increase the probability of successfully find a parking space. Figure 3.13 shows that reducing demand to destination 10, 15, 16, 17 and 19 causes increasing the probability of successfully finding a parking spot on nearby links (10,16), (16,17), (17,19), (19,15) and (15,10). The horizontal axis indicates the reduction percentage while the vertical axis means the probability of finding a parking space.

Intuitively, reducing the flow searching for parking on a link will increase the probability of successfully parking on that link. Another expected result of reducing some demand to some specific destination is that some vehicles which originally park on links that are far from that destination may have chance to park nearer to their destination. As a result, the reduction rate of parked vehicles on the links near to the destination may be less than the reduction rate of searching parking vehicles on those links. We are more interested in the area that near by the destinations with

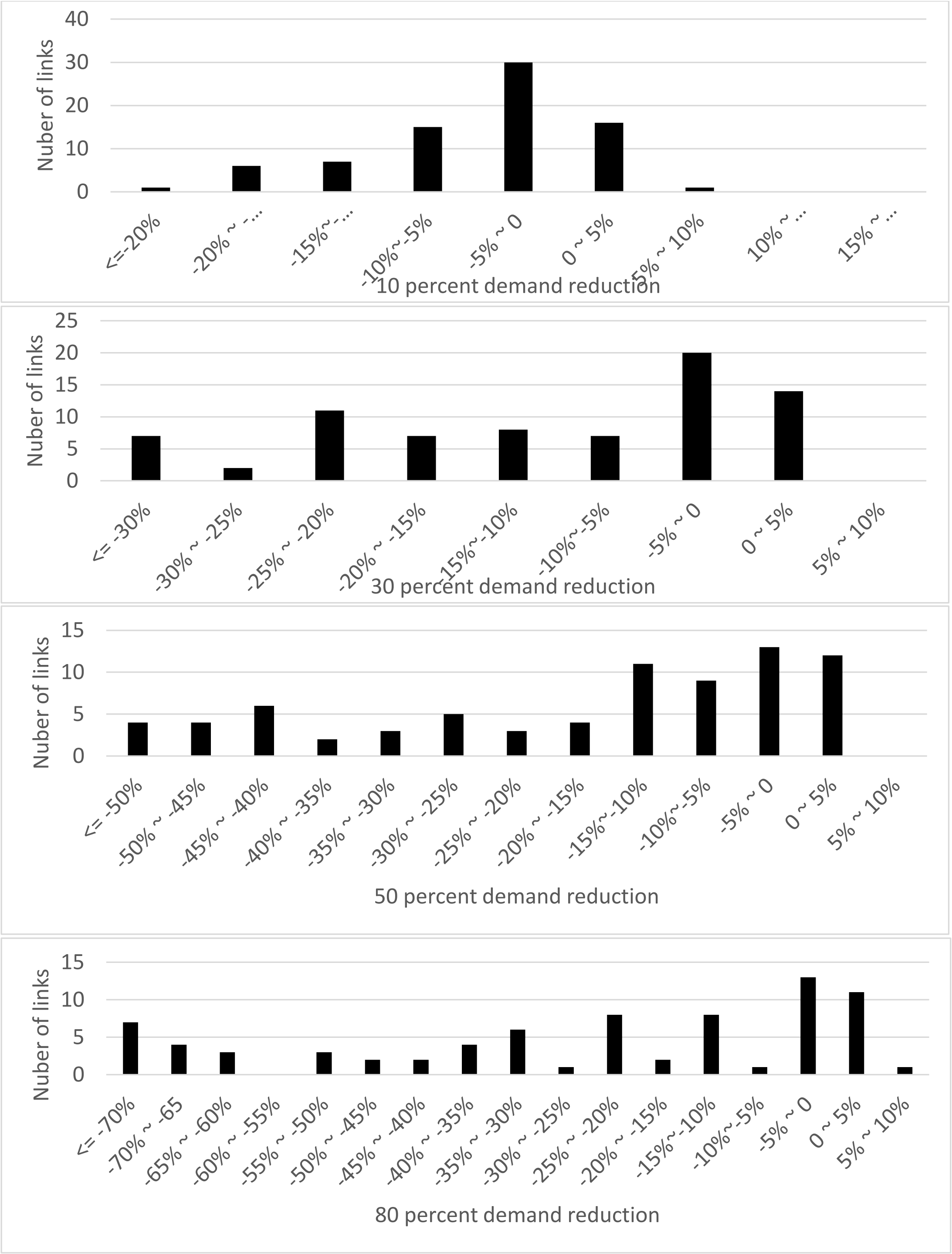
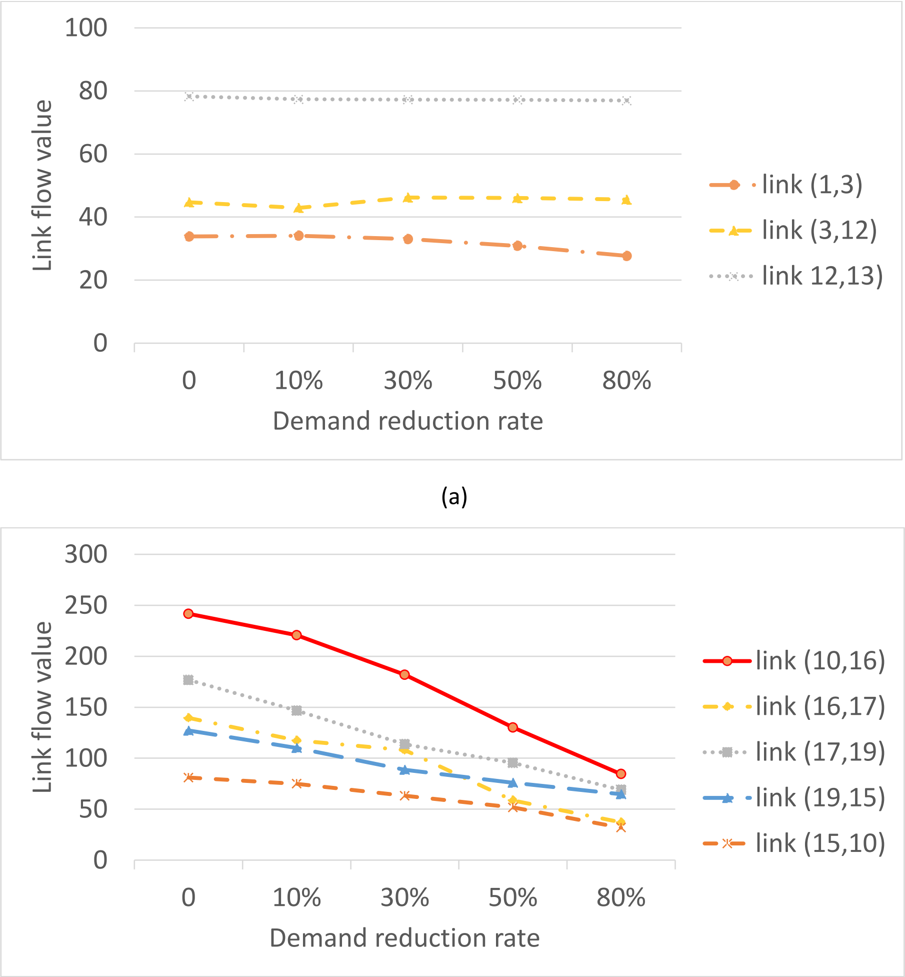


Figure 3.11: Link traffic flow change distributions in response to travel demand reduction



(a)

(b)

Figure 3.12: Flow percentage change with the demand decrease (a) links far from the demand change destinations, (b) links near to the demand change destinations

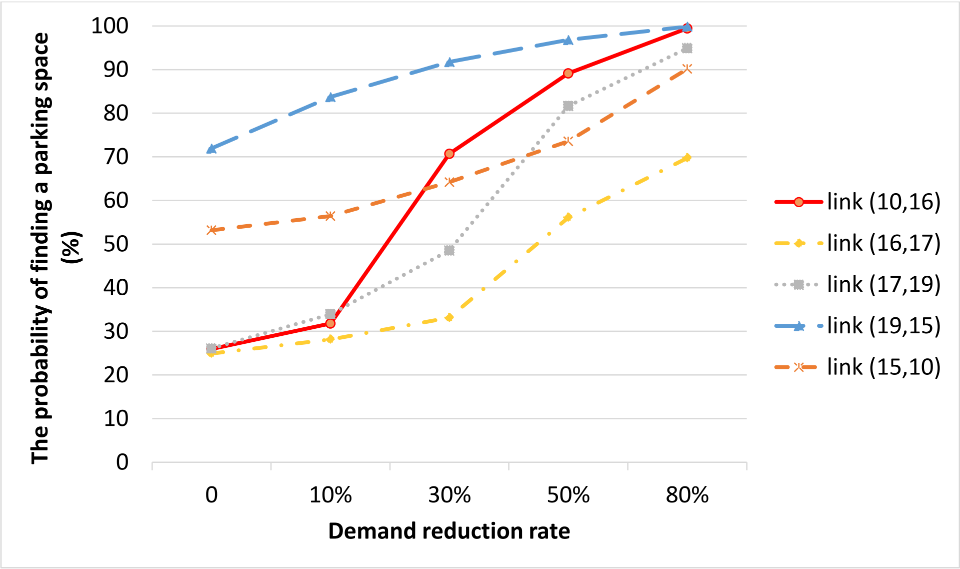
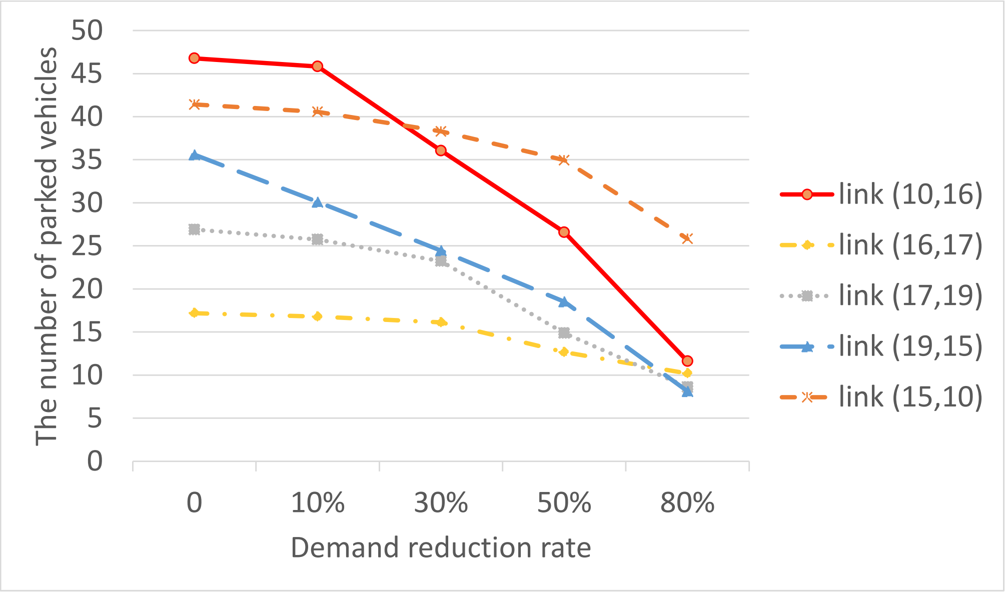


Figure 3.13: The probability of finding a parking space with different demand values

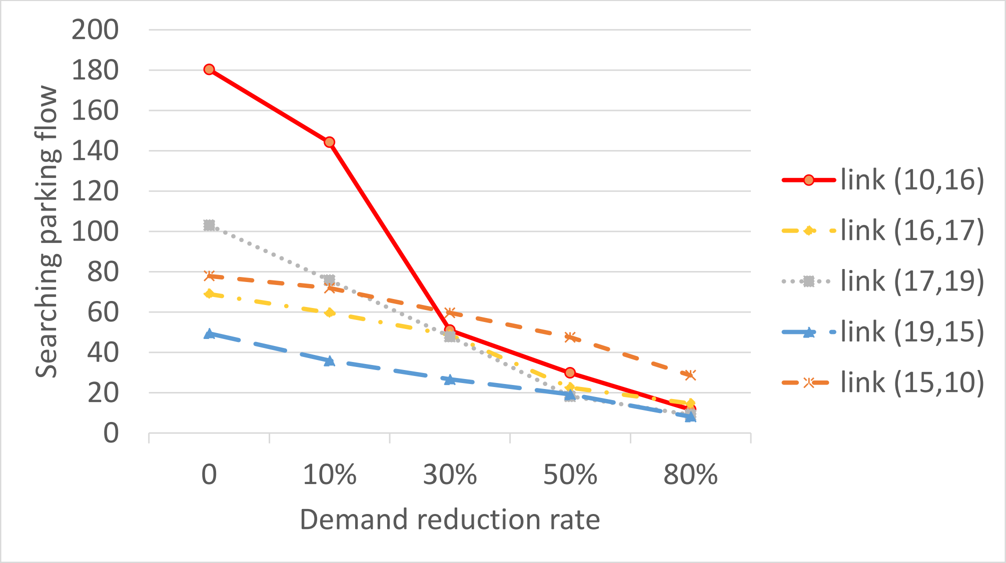
reduced demand. Thus, we check the how the searching parking flow and parked vehicles change on these links.

In Figure 3.14, sub-figure (a) shows how the number of parked vehicles change and(b) shows how the searching-parking flow change. This figure clearly tells us that the number of parked vehicles does not change as much as the searching-parking flow. The reason is when we reduce the demand, less vehicles will search around, so the searching-parking drops much, but because it is easier to find a parking space, the ratio between the parked vehicles and the total searching-parking flow is actually higher. This effect is more clearly seen in Figure 3.15, which shows how the link flows, searching-parking flows and the parked vehicles change in response to a 30 percent demand reduction to destinations 10, 15, 16, and 17. In this figure, the horizontal axis identifies the three types of flow, and the vertical axis gives the change rate value. It is obvious that if the demand changes, the searching-parking flow will change most, while the total link flow and the parked vehicles do not change much. This is especially true for links that usually bear a large amount of parking flow and cannot accommodate the searching flow at all times.

Travel demand changes affect travel time sa well as the flow pattern and parking distribution. Generally, with lower travel demand, lower travel time is expected. This is not only because lower demand reduces congestion level, but also because lower demand can improve the parking condition so vehicles need less time to find available parking spaces. However, much of the impact depends on the destination's location. If it is far from the destinations which have reduced demand, the travel times of trips toward this destination may not be impacted very much. On the other hand, if a



(a)



(b)

Figure 3.14: Parking flow change in response to different demand reduction rate

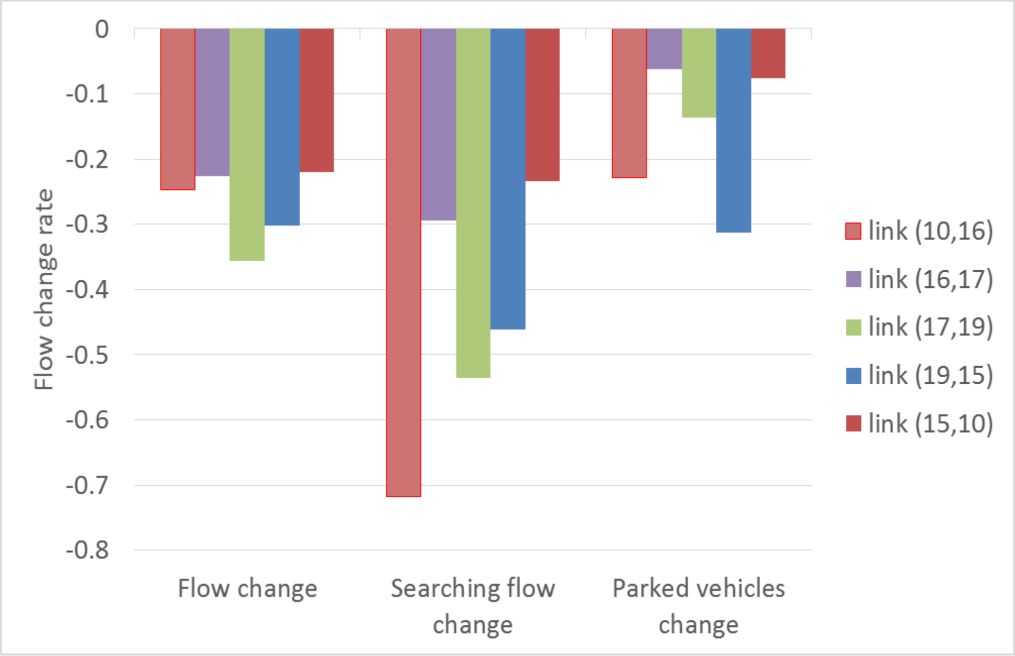


Figure 3.15: Comparison of the impact of demand change on link flow, searching-parking flow and parked vehicles

destination is very near to demand-reduced destinations, the trips to this destination may have much less travel time than the conditions before demand reduction.

Figure 3.16 shows how the link travel times change in response to reducing the travel demand to destination 10 by 10, 30, 50 and 80 percent, respectively. For the sake of presenting clearly, only three typical links aer chosen in this figure. Link (1,2) represents links far from demand-reduced destinations, link (10,16) represents links that are very near to these destinations, while link (5,6) represents those with middle distance to the destination. Apparently, for all types of links, the lower the travel demand, the less travel times for trips to destination 10. Nevertheless, for some destinations, reducing the demand to them may not have much impact on the travel times. For instance, Figure 3.17 shows if we reduce the demand to destination 1 by 10, 30, 50 and 80 percent, respectively, there is almost no impact on travel times for any type of the links. (In this case, link (1,2) represents the nearby ones, while link (10,16) represents the further links.)

3.7.2 Pricing Test

Common sense suggests that increasing the parking price would always reduce driver's willingness to park. The truth is more complex. For links where the probability of finding parking is very small due to congestion, slightly increasing the parking price may not have a significant impacts on parking occupancy. Or, if the probability of finding parking a link is close to 1always has many opening parking spaces, it may be not a good place to park. For such link, increasing pricing rate will also have little impact on parking occupancy. Increasing parking rate can also cause flow shifting.

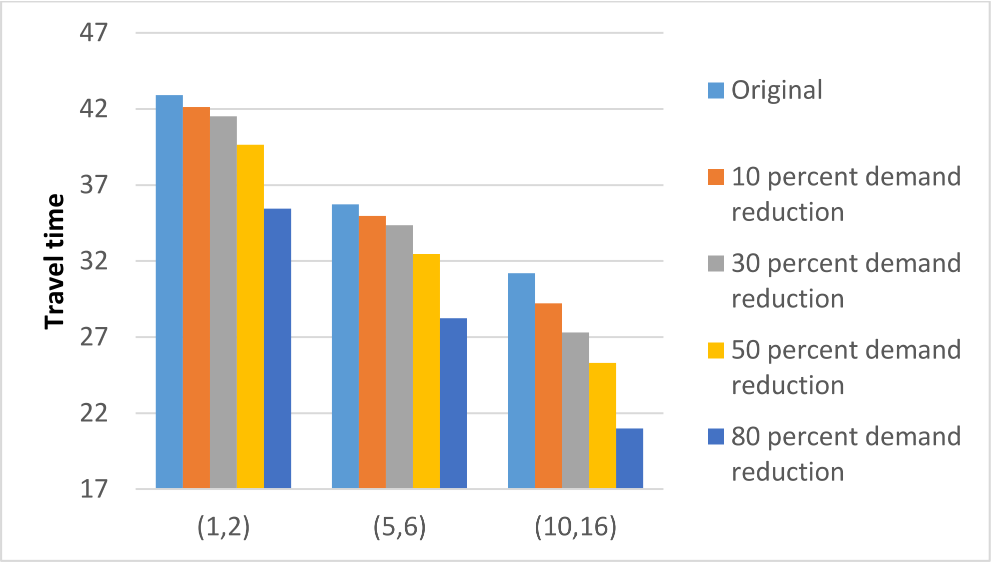


Figure 3.16: Travel times to destination 10 with different demand values

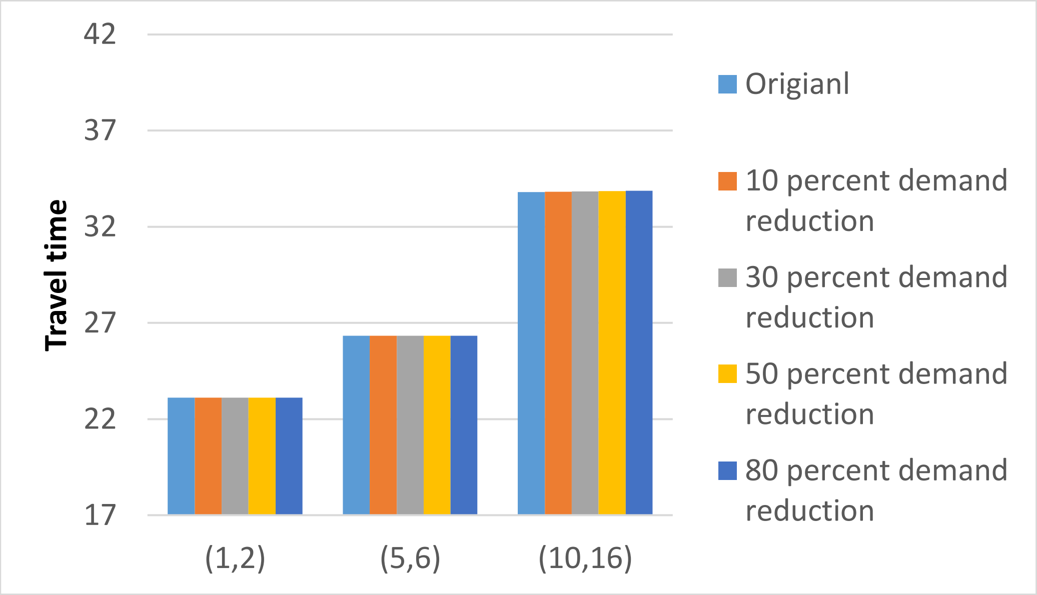


Figure 3.17: Travel times to destination 1 with different demand values

Traffic flows that are usually searching parking on a link may transfer to other links because of a parking rate increase on this link.

Since we are mostly interested in the pricing change impacts on links that have relatively high parking demand, we chose links (10,16) and (16,10) to increase their parking rate by 10, 30, 50, 80 and 100 percent to see how it will influence the flow pattern and the probability of successfully find a parking space. Figure 3.18 shows how the probability of finding a parking space would change as parking price increases. It is obvious that it is getting easier to find a parking space on links (10,16) and (16,10) as increasing the parking rate. One interesting thing is that we did not increase the parking rate on link (16,18), but as a neighbor link it also has higher probabilities of finding a parking space as parking rate increases on links (10,16) and (16,10). Figure 3.19 shows that compare to the increased probability of finding a parking space, the parking searching flow decreases on the two links. This makes sense as people want to reduce their total trip cost by avoiding paying extra parking fee. However, for other links, such (4,11), the parking searching flow increased because of the "flow shift."

3.7.3 Parking Capacity Test

Increasing the number of parking spaces on a link would allow vehicles to find parking spaces more easily. The question is how much this capacity change can increase the probability of finding a parking space. If the parking availability probability on a link is high, increasing the number of parking spaces would have little impact. However, if a link always have a high demand of parking, the increasing

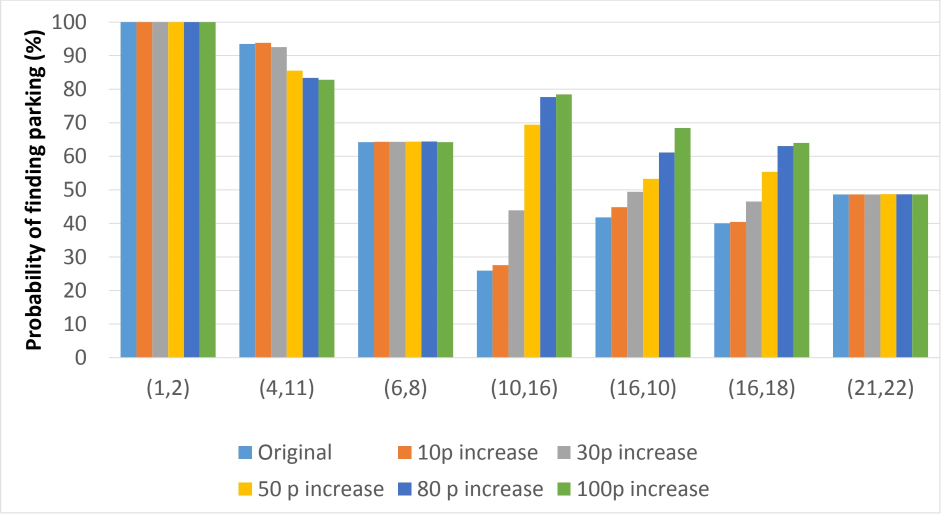


Figure 3.18: The probability of finding a parking space as parking rate increases

might change searching and parking flows on this link and on nearby links. For links far from the capacity changed link, we would not expect much change in the probability of finding a parking space.

Figure 3.20 and Figure 3.21 show how the increase in parking capacity would impact the probability of successfully find a parking space on links. Figure 3.20 is for link (1,3) and (3,1) which originally has nearly 100% chance of an available parking space. After increasing the number of parking spaces on these two links, the probability of finding a parking space on any link changes little. This is because, even increasing the number of parking spaces, these two links cannot attract vehicles originally parking on other links to park on them, so there is very little impact on the parking distribution and trip time pattern. Contrarily, Figure 3.21 demonstrate how

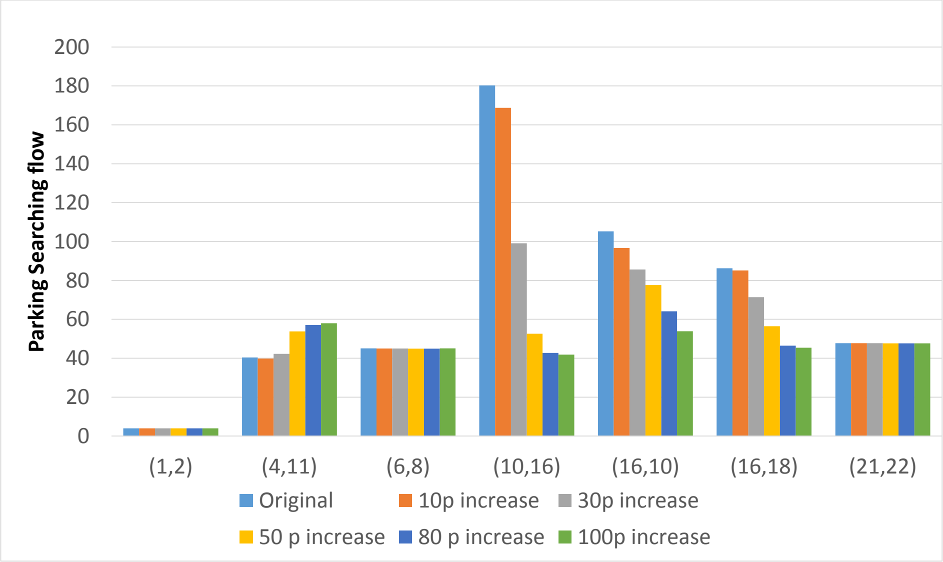


Figure 3.19: Parking searching flow on links as parking rate increases

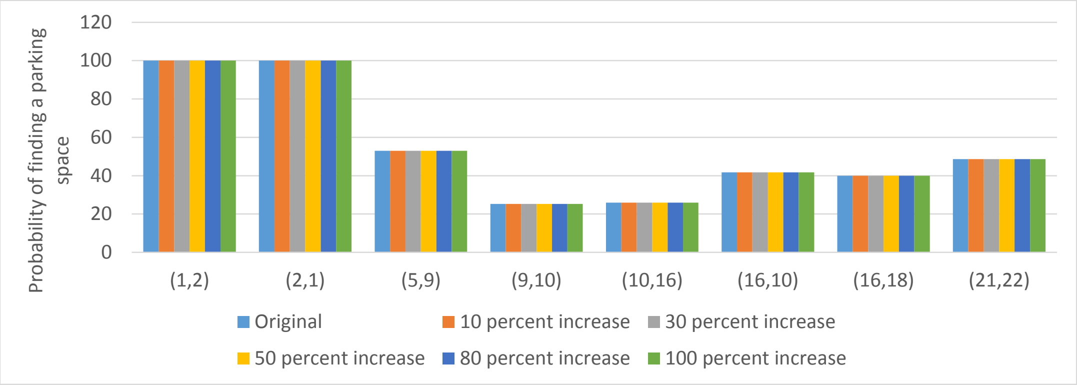


Figure 3.20: The probability of finding parking spaces with parking capacity increase on links (1,3) and (3,1)

the parking capacity change on link (10,16) and (16,10) would impact the probabilities of finding a parking space on the links. Apparently, the parking capacity change on these two links does not only increase the chance of finding a parking space on themselves, it also increases the chances on other link. For instance, after the capacity increase, the first two links which have the highest probability increase are link (10,16) and link (16,18) which is immediately connected to link (10,16). Figure 3.22 shows that if the parking capacity is increased, the probability of finding a parking space also increased, roughly in proportion to the increase in capacity.

Since parking capacity increased, a link which used to have a high parking demand should attract more vehicles to park on it. This should result in a flow increase on neighboring links. The reason is that increasing parking capacity on a link will attract more vehicles to park on it which result in more flow passing through some of the neighbor links, while it also decreases the searching flow on the other

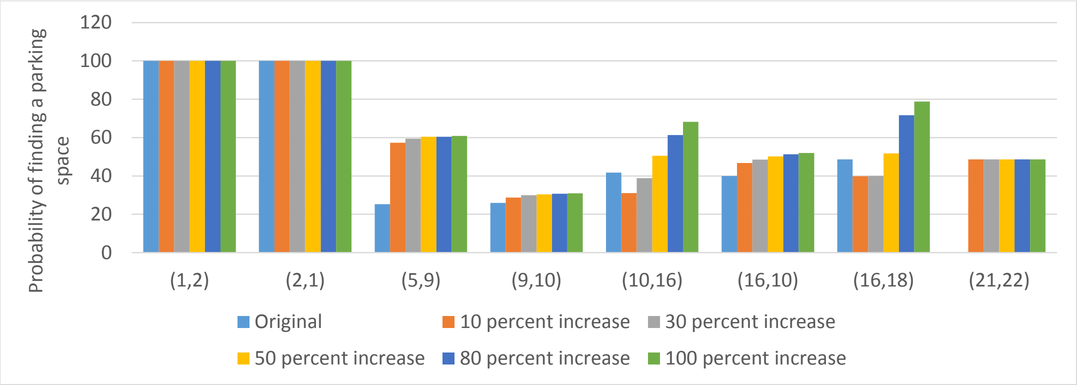


Figure 3.21: The probability of finding parking spaces with parking capacity increase on links (10,16) and (16,10)

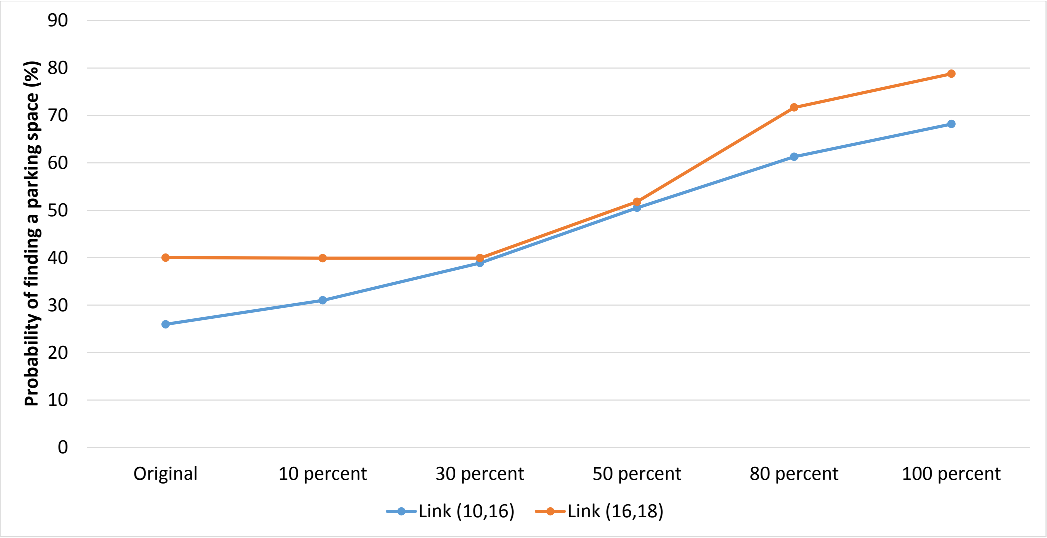


Figure 3.22: The impacts of parking capacity increase on link (10,16) and link(16,18)

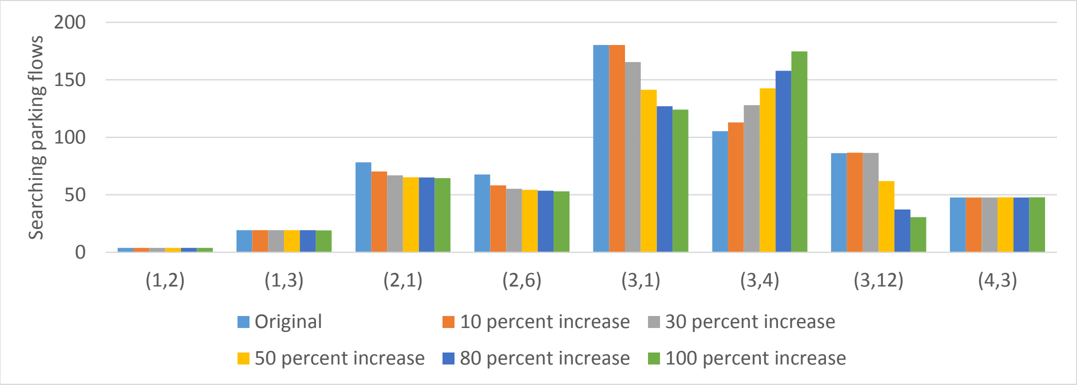


Figure 3.23: Searching parking flow change in response to parking capacity increase on links (10,16) and (16,10)

links. Figure 3.23 shows almost all the links have lower searching parking flows with the increasing of parking capacity. Among those which have higher searching flows, link (16,10) has a very obvious jump in searching flow when its parking capacity is increased. This actually mean more vehicles wish to park on this link. unexpectedly, link (10,16) has lower searching flow even when its capacity has been increased.

Figure 3.24 can explain this in one aspect. It shows that with the capacity increasing, the probability of finding a parking space on link (16,10) does not change much. Considering the searching parking flow increased a lot, this means more vehicles parked on this link compared to the original case. ON the other hand, the probability of finding a parking space on link (10,16) changes dramatically when increasing the capacity. Even the searching parking flow decreased, the actually parked vehicles does not decrease. Figure 3.25 shows the number of parked vehicles on the two links. We can see that even they have very different searching flow and total link

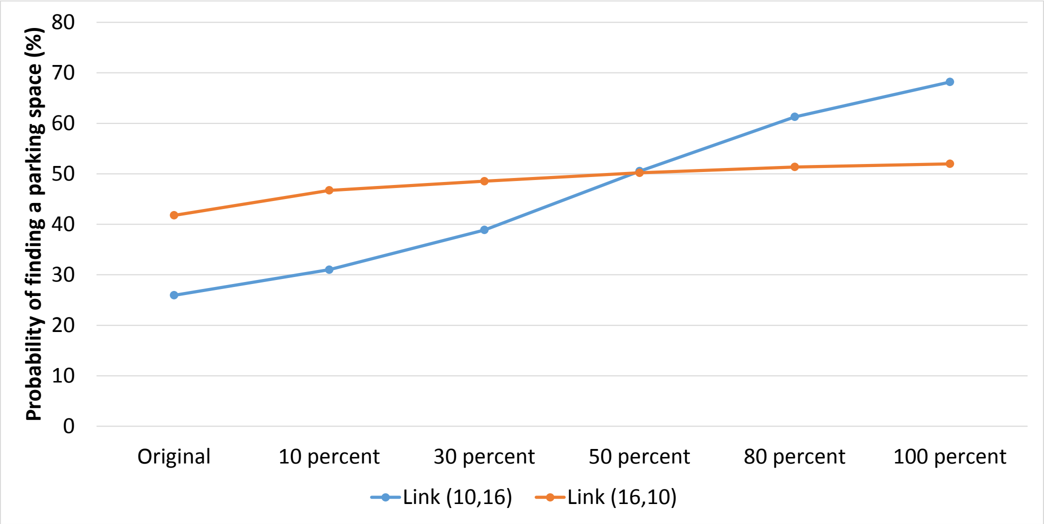


Figure 3.24: Changes to the probability of parking availability in response to parking capacity increase on links (10,16) and (16,10)

flow changing direction, both of them have similar changing direction and quantity in the number of parked vehicles.

We are also interested in how the travel time would change after increasing the parking capacity. Figure 3.26 shows that trips from all types of links (as aforementioned,the types are classified in terms of its distance to the capacity-increasing link) to destination 10 have lower values if increasing the parking capacity on links (10,16) and(16,10). It is because after parking capacity increased, on average vehicles are more easily to find a parking space which in turn reduce the searching time and the congestion level.

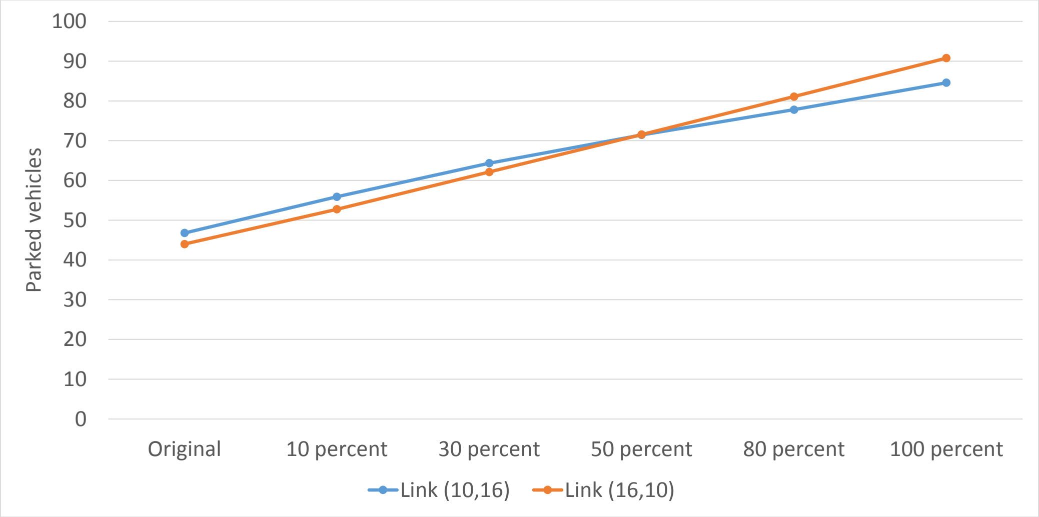


Figure 3.25: Parked vehicles change in response to parking capacity increase on links (10,16) and (16,10)

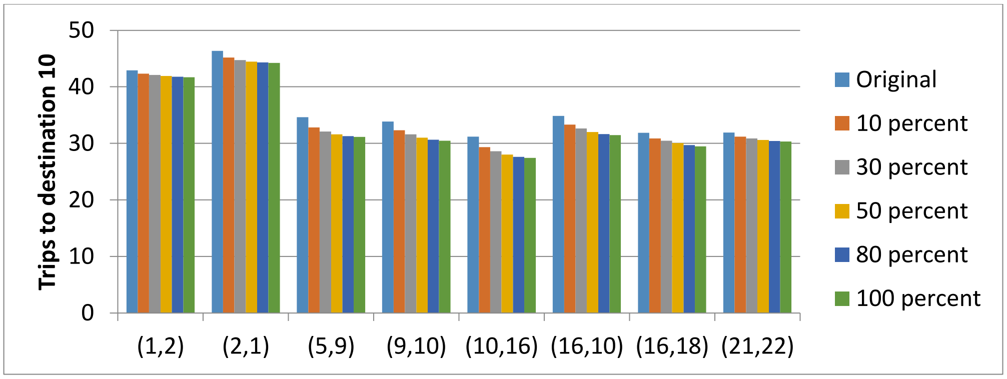


Figure 3.26: travel time to destination 10 with parking capacity increase on links (10,16) and (16,10)

3.8 Conclusions

This chapter presented an equilibrium model for the parking search process. Notable features of the model are its ability to model the probability of parking availability as a function of searching intensity; the introduction of an equilibrium framework to account for the dependence of searching intensity on parking availability probabilities; and a formulation admitting general networks, allowing the increases in flow on specific links due to parking search to be seen.

There are many directions for future research. As the focus of this chapter was on formulating the basic model, it would be fruitful to search for more efficient algorithms or algorithms which provably converge to equilibrium, or to more fully develop practical case studies using field data on parking availability and destinations. The model itself can also be extended to account for congestion in link travel times, dynamic evolution in demand and congestion, or destination choice or demand elasticity depending on parking availability.

Chapter 4

Dynamic Traffic Assignment and Parking Search

4.1 Introduction

The model in this chapter is particularly aimed at the "cruising" phenomenon when drivers are searching for parking. Although as mentioned in Chapter 1 parking search traffic account for a considerable amount of traffic volume, typical traffic assignment models based on equilibrium and shortest path concepts do not consider additional delay or stochasticity due to parking at the destination; a preliminary investigation in Tang et al. (2014) show that for short trips, this can underestimate travel times by up to 50% when parking delays are included.

As discussed in more detail in the literature review section, most methodological approaches for modeling parking delay on networks are generally based on discrete choice concepts, or the introduction of artificial parking links. In our opinion, both of these approaches have serious shortcomings: neither approach explicitly models the additional congestion caused by cruising behavior, and do not reflect the stochastic and adaptive choices drivers make as they pass multiple parking options *en route* to a destination. Simulation-based approaches have also been proposed, but these generally lack the behavioral foundations expected of modern planning models (e.g., appropriate generalizations of the equilibrium concept).

By contrast, this chapter develops an equilibrium formulation accounting for stochastic and dynamic parking search by routing drivers based on *policies* rather than *paths*, using the language and framework of Markov decision processes. The mean features of this dynamic traffic assignment model are (i) incorporation of a stochastic parking model into the cell transmission model to represent traffic flow and (ii) a policy-based stochastic routing model to reflect driver behavior and adaptive choices regarding available parking spaces (for instance, if a driver sees an available space, should he or she take it or continue driving in hope of finding a more convenient space further downstream). In this way, the stochastic and dynamic nature of the parking search process can be explicitly modeled while still building on behavioral assumptions common to planning models.

These concepts are united in an equilibrium framework. While the notions of equilibrium and stochastic networks may seem mutually exclusive, this chapter adopts a similar framework as the recourse equilibrium formulations of Unnikrishnan and Waller (2009) and Boyles (2009), in which the equilibrium is formulated in terms of policies (containing a set of contingent plans based on observed network conditions) rather than paths. However, in contrast to these two earlier works, the state probabilities depend on flows in addition to the state travel times, as explained in more detail below. In this way, the model developed in this chapter is a generalization of these recourse equilibrium models. This model is amenable to implementation in the form of agent-based simulation, although specific details of such an implementation are not described in this chapter.

Similar to the static model in Chapter 3, the model in this chapter also pursue traffic assignment equilibrium while incorporating parking search behavior. The difference is that this chapter uses simulation based approach which can reflect the changes to the input data, demand, congestion level and parking space capacities over time. This dynamic approach can give intuitive results about how the network performance will change over time with time-dependent variable inputs.

The remainder of this chapter is organized as follows. Section 4.2 presents prior research in the areas of parking search modeling and policy-based routing and equilibrium. Section 4.3 provides an overview of the proposed parking model, with additional details on its specific components in the following three sections: Section 4.4 discusses the extensions to the cell transmission model and the parking dynamics, Section 4.5 discusses the routing policies used by drivers, and Section 4.6 explains the equilibrium concept which ties them together. Finally, Section 4.9 concludes this chapter and provides discussion of some practical considerations.

4.2 Literature Review

Parking plays a surprisingly large role in traffic operations in dense urban areas, near universities or other demand centers, and in managing special events. Shoup (2006) reviewed parking search studies from 1927 to 2001 and found that between 8% and 74% of the traffic in congested downtowns were drivers cruising in search of parking locations. Recognizing the impact of parking searches on urban congestion, several studies have been conducted using economic, statistical, and optimization frameworks on various aspects of parking such as parking choice and pricing. Parking choice models can be classified into network assignment-based approaches, discrete choice-based approaches and simulation-based approaches.

Network assignment-based approaches model the parking choice in conjunction with the route choice. Nour Eldin et al. (1981) used an incremental assignment approach to solve the traffic assignment problem which models the interaction between route choice, resulting vehicular flows, and parking choices. Bifulco (1993) developed a network level stochastic user equilibrium model to model parking search cost as a function of parking level occupancy as well as the route choice. Li et al. (2007a) consider mode choice between auto and transit and the simultaneous route and parking choice for automobile users using the user equilibrium framework. Lam et al. (2006) developed a variational inequality formulation for a multi-class network assignment model which considers departure time, route, parking location choice with drivers classified based on parking durations. Li et al. (2008) develop a fixed point based assignment model to study the impact of time dependent and normally distributed uncertain travel times and parking search time on network level reliability. Gallo et al. (2011) developed a stochastic user equilibrium based fixed point formulation which modeled car trip, cruising for parking, and walking to final destination in multiple network layers. Several network assignment based models were used to evaluate pricing based parking policies. Lam (2002) and Li et al. (2007b) adopted a bilevel programming framework where the upper level determines the optimal tolls and parking charges and the lower level models the equilibrium assignment in response to the tolls and parking charges. D'Acierno et al. (2006) developed several optimization models to determine the parking prices taking into account the transit connectivity between origin-destination pairs with a network using a multi-modal network assignment model. However, in the these models parking is generally modeled as a deterministic phenomenon imposing a known cost to drivers, and not contributing to congestion or delay for other drivers not searching for parking.

Discrete choice models neglect the network structure and use random utility theory to understand parking choice as a function of various driver and parking location attributes. Van der Goot (1982), Axhausen and Polak (1991), and Lambe (1996) use multinomial logit model to model parking location choice. Other discrete choice model forms considered include the mixed multinomial logit model (Hess and Polak, 2004; Hess and John, 2004) and the nested logit model (Hensher and King, 2001). However, these models do not directly incorporate parking costs into the network loading and assignment.

The third category of parking choice studies has adopted an agent-based approach to model parking search. Readers may refer to Section 2.2 in Chapter 2 to find previous works on this.

In contrast to all of the above, the model presented here is based on the concept of stochastic shortest paths with recourse, an approach pioneered for parking search in (Tang et al., 2014). Also known as the online shortest path problem (Cheung, 1998; Waller and Ziliaskopoulos, 2002; Provan, 2003), in this problem drivers progressively learn the realizations of stochastic network costs and adapt their chosen path *en route*. Although not explicitly noted in these papers, the online shortest path problem with reset takes the form of a classical Markov decision process (Bertsekas, 2012). This was adapted to parking modeling by using the recourse concept to model driver choices upon learning the state of a link (whether parking is available or not); these choices can include parking if a spot is available, or a choice of successor link to follow if no spot is available.

4.3 Modeling Overview

This section presents a general overview of the modeling components and motivating concepts using a simple example, before defining them in more general mathematical terms in the sections that follow. For illustrative purposes, the simple network in Figure 4.1 will be referred to throughout this section. In this network, drivers are ultimately attempting to reach the destination D. Unlike traditional transportation planning models, D is not a node in the transportation network (which represents the transportation infrastructure itself). Rather, drivers must park on a network link and then walk to D, incurring a walking cost. Each of the three links A, B, and C has a uniform travel time of 1 unit, but represents a different parking situation. Links A and B represent free on-street parking, while link C represents a paid lot with cost c. Link B is closest to the destination, while links A and C are further away and have a higher walking time $w > 1$. Assuming a uniform value of time and measuring costs in time units, we can incorporate any monetary cost into the walking time, and simply say that the walking times from A, B, and C are w, 0, and $w + c$, respectively.

Drivers clearly prefer to park on link B. Assuming that the arrival rate of vehicles is smaller than the departure rate of parked vehicles on link B, all drivers' desires can be accommodated. However, if this arrival rate increases, not all drivers will be able to park on this link. Assume that drivers have no information on the locations of available parking spaces until they traverse a link, but from experience

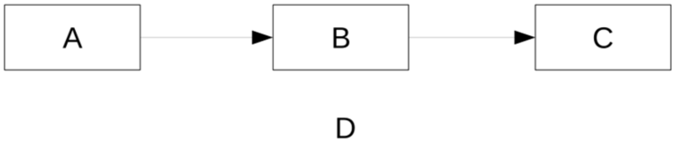


Figure 4.1: Small cell network to demonstrate model concepts

know that the probability of finding parking on link B is p. The question is, upon seeing an available parking space on link A, will the driver choose to park there or continue in hopes of being able to park at B. Assuming that drivers wish to minimize expected travel time, they will park on link A whenever $w < 1 + (1 - p)(1 + w + c)$ (the expected additional time for continuing to link B, including driving and walking time), or equivalently $p < (2+c)/(1+w+c)$. Likewise, whenever $p > (2+c)/(1+w+c)$ expected travel time is minimized by continuing to link B, and then parking at link C if no space is available at B.

However, the probability p depends on these choices drivers make, and the only stable solution occurs when $p = (2 + c)/(1 + w + c)$, and the fraction of drivers choosing to park on link A is exactly the right amount for this value of p to occur. In this case, the expected travel times are equal from parking at A or continuing on to B, and drivers are indifferent between these two options. (Figure 4.2). For any other p value, drivers would switch their behavior at link A, with more drivers seeking to park at A if p falls below this threshold and fewer seeking to park at A if p exceeds this value, and these behavioral switches would move p closer to $(2 + c)/(1 + w + c)$. In this way, the stable states correspond to equilibria in terms of policies, where the policy concerns the choice drivers make when an available space is found on link A.

To specify this model in a way that applies to general networks, we need to represent both (i) the traffic flow and parking dynamics, given driver policies; (ii) a policy selection model representing drivers' desire to minimize expected travel times; and (iii) the equilibrium framework uniting the first two. Each is respectively described in the following three sections.

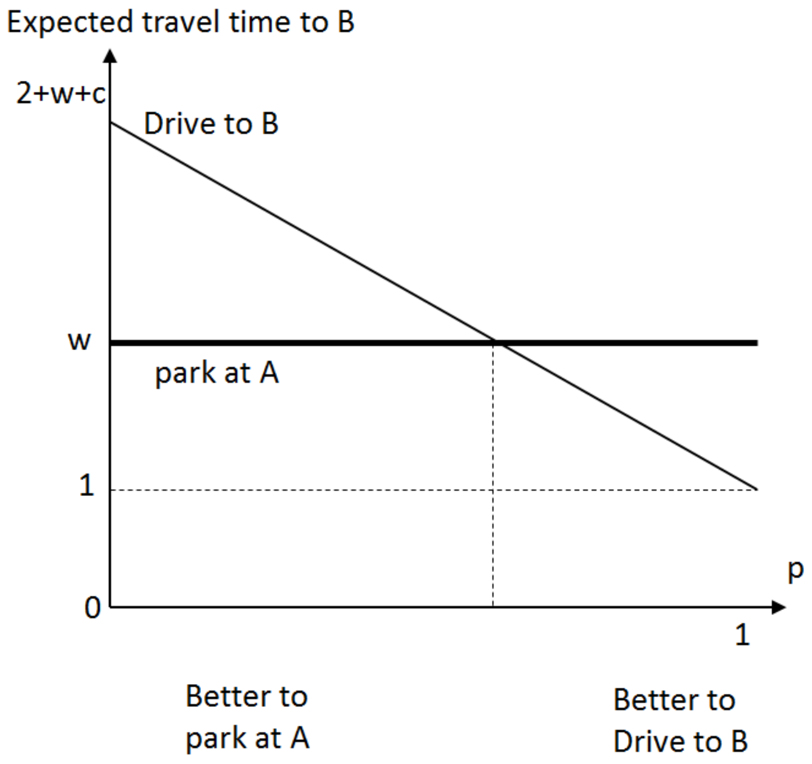


Figure 4.2: Optimal decisions at A as a function of parking availability at B

4.4 Supply-side: Networks and Traffic flow

The flow model is built upon the cell transmission model (CTM), developed by Daganzo (1994, 1995) as a discrete solution method for the LWR hydrodynamic model (Lighthill and Whitham, 1955; Richards, 1956) based on a Godunov scheme. To implement CTM, each link must be divided into a finite number of cells whose length is the distance a vehicle travels at free-flow in one simulation tick of length Δt. Briefly recapitulating, each cell c is associated with parameters representing its capacity Q_c, the maximum number of vehicles which can fit into the cell N_c, and the ratio of backward-to-forward wave speeds δ_c, and a state variable $x_c(t)$ denoting the number of vehicles in cell c during the t-th time interval. $\tau_c(t)$ is used to indicate the time necessary for a vehicle to traverse cell c when entering during time interval $t \in \{0, 1, \ldots, T\}$.

To propagate flow, at each time interval define the sending flow $S_c(t) = \min\{x_c(t), Q_c\}$ and receiving flow $R_c(t) = \min\{\delta_c(N_c - x_c(t)), Q_c\}$. A variety of intersection models exist mapping the sending flows of incoming links and receiving flows of outgoing links to transition flows between cells. For instance, if cells c and d are in series, the number of vehicles moving between these cells during the t-th time interval is the lesser of $S_c(t)$ and $R_d(t)$. Multiple diverge and merge models have been formulated in the literature (Daganzo, 1995; Nie et al., 2008; Yperman, 2007), along with models for more general intersection types (Yperman, 2007; Tampre et al., 2011; Corthout et al., 2012). These are not discussed here for brevity, and any of these can be used as the basis for the parking model presented below, in which each cell is equipped with additional state variables used to represent the number of available parking spaces.

Consider a time-dependent network with node and arc sets N and A, respectively, and let C denote the set of cells; $\Gamma(c)$ is the set of cells immediately downstream of cell c. Let S denote the set of destinations. Unlike most network equilibrium models, the destinations S are distinct from the network nodes N, and are not directly connected to the links. Instead, for each cell c and destination s, define w_{cs} to be the walking time between cell c and destination z. Monetary and other generalized costs can easily be incorporated into this term. If parking is not permitted on cell c, by convention define $w_{cs} = \infty$ for all s. Let the parameters P_c and μ_c respectively denote the number of parking spaces associated with cell c and the mean duration vehicles park on this cell, and let the state variable $a_c(t)$ the number of available parking spaces on this cell at the start of interval t ($0 \leq a_c(t) \leq P_c$). Demand is specified by the parameters $d_{cs}(t)$, denoting the number of vehicles beginning trips at cell c during the t-th time interval, traveling towards destination s.

Let $x_c(t)$ denote the number of vehicles in cell c at time t; these vehicles are distinguished by whether they would park at this cell if a space is available, or whether they would choose to keep driving to find a parking space closer to the destination. Denote these numbers of vehicles as $x_c^P(t)$ and $x_c^{NP}(t)$, respectively, so that $x_c(t) = x_c^P(t) + x_c^{NP}(t)$. These values are determined by the policies drivers choose, as described in the next section.

After propagating flow already on the network in the t-th time interval, any vehicles originating at cell c are loaded onto that cell if space permits (otherwise, they are held back until the next time interval), along with the vehicles vacating parking spaces (denoted $d_c(t)$). Modeling parking departures as a Poisson process, the probability that any occupied space will be vacated in a simulation tick is approximately $\Delta t/\mu_c$, assuming Δt is sufficiently small. Following this, the number of parking vehicles $e_c(t)$ is calculated as the lesser of $x_c^P(t)$ and $a_c(t) + d_c(t)$; the vehicles which can park are randomly sampled from $x_c^P(t)$ and the probability of finding parking on this cell at time t is $p_c(t) = e_c(t)/[a_c(t) + d_c(t)]$.[1] Note that this ordering implies that vehicles already on the network have priority over vehicles attempting to enter the network or leave parking spaces, and that vehicles seeking to park are willing to wait to allow vehicles vacating parking spaces to do so. This process is shown in Figure 4.3.

4.5 Demand-side: Parking Policies and Choices

Each vehicle is assigned a parking policy which determines its route and actions whenever an available parking space is found. Formally, define the state space $\mathcal{S} = C \times T \times \{P, NP\}$, whose elements $\sigma = (c, t, \rho) \in \mathcal{S}$ indicate the current cell c, time interval t, and parking status ρ (P for parking, NP for no parking) for a vehicle. The notation $c(\sigma)$, $t(\sigma)$, or $\rho(\sigma)$ is used to refer to these elements of state σ. A policy is a mapping $\pi : \mathcal{S} \to C \cup \{P\}$ associating with each state s a corresponding action to take — either a downstream cell to move towards, or the parking action P. A policy is feasible if for all σ (i) $\pi(\sigma) \in C$ implies $\pi(\sigma) \in \Gamma(c)$ and (ii) $\pi(\sigma) = P$ only if $\rho(\sigma) = 1$.

[1] This intentionally violates first-in/first-out ordering; a parking space opening just upstream of a lead vehicle may be taken by a following vehicle.

1. Initialize all cell occupancies $x_c \leftarrow 0$, $t \leftarrow 0$, and parking availability $a_c = P_c$.

2. Propagate flow at time t using cell transmission model.

3. For each cell c and each occupied parking space, generate a random real number ξ by uniformly sampling the interval $[0, 1]$; if $\xi < \Delta t / \mu_c$, increment a_c.

4. For each cell c identify the number of vehicles searching for parking x_c^P based on policies associated with each vehicle.

5. For each cell c randomly select $e_c = \min\{x_c^P, a_c\}$ vehicles among the x_c^P searching for parking, move them to parking spaces and $a_c \leftarrow a_c - e_c$

6. Update cell travel times and parking probabilities.

7. If $t < T$ increment t and return to step 2.

Figure 4.3: Cell transmission model algorithm with parking added.

That is, each driver in a cell chooses whether or not to park on that cell if a space is available; if no space is available or if the driver chooses not to park, then the driver must choose which downstream cell to traverse next. If the driver chooses action P for a feasible policy, their trip is complete because a parking space is available. If the driver chooses a downstream cell, so $\pi(\sigma) = d \in \Gamma(c)$ when $\sigma = (c, t, \rho)$, then they enter cell d at time $t + \tau_c(t)$. With probability $p_d(t + \tau_c(t))$, the driver is next in state $(d, t + \tau_c(t), P)$, and with probability $1 - p_d(t + \tau_c(t))$ they are in state $(d, t + \tau_c(t), NP)$.

Given fixed values of travel times τ and parking probabilities p, the expected travel time corresponding to a feasible policy π can be calculated as follows. Let $L_\pi(\sigma, s)$ denote the expected remaining travel time from the current state $\sigma = (c, s, \rho)$ to destination s when using policy π. These labels satisfy the recursion $L_\pi(\sigma, s) = w_{cs}$ if $\pi(\sigma) = P$ and $L_\pi(\sigma, s) = \tau_c(t) + p_d(t + \tau_c(t))L(d, t + \tau_c(t), P) + (1 - p_d(t + \tau_c(t))L(d, t + \tau_c(t), NP)$ otherwise, where $d = \pi(\sigma)$ for brevity. The first case corresponds to the parking action, while the second corresponds to driving to the next cell.

Finding an *optimal* policy π^* for a given destination s and values of p and τ is not difficult, and is accomplished by a standard label-correcting algorithm such as that in Figure 4.4. In our behavior model, we assume that each driver chooses a policy so as to minimize his or her expected travel time.

4.6 Equilibrium

The models presented in the previous two sections exhibit the mutual dependency which is typical of transportation network models: travel times and parking

```
1. Initialize all labels $L(\sigma) = \infty$, create new policy $\pi^*$, set $t \leftarrow T$.
2. For each cell $c$:
    (a) Update $L(c,t,NP) \leftarrow \min_{d \in \Gamma(c)} \{p_d(t + \tau_c(t)) L(d, t + \tau_c(t), P) + (1 - p_d(t + \tau_c(t))) L(d, t + \tau_c(t), NP)\}$
    (b) Update $\pi^*(c,t,NP) \leftarrow \arg\min_{d \in \Gamma(c)} \{p_d(t + \tau_c(t)) L(d, t + \tau_c(t), P) + (1 - p_d(t + \tau_c(t))) L(d, t + \tau_c(t), NP)\}$
    (c) If $w_{cs} \leq L(c,t,NP)$ then update $L(c,t,P) \leftarrow w_{cs}$ and $\pi^*(c,t,P) = P$; else $L(c,t,P) \leftarrow L(c,t,NP)$ and $\pi^*(c,t,P) \leftarrow \pi^*(c,t,NP)$.
3. If $t > 0$ decrement $t$ and return to step 2.
```

Figure 4.4: Label correcting algorithm for finding optimal policies given destination s.

probabilities depend on the policies used by drivers, but the optimal policies chosen by drivers depend on the travel times and parking probabilities obtained from the flow model. As suggested in Section 4.3, these perspectives are harmonized by the introduction of an equilibrium principle. In particular, *an equilibrium solution associates a policy with each vehicle such that each vehicle's assigned policy is optimal for its destination, given cell travel times and parking probabilities consistent with this policy assignment.*

The model, as formulated above, assumes that vehicles are discrete; this is necessary because the parking spaces are modeled as discrete entities. Therefore, an exact equilibrium solution may not exist. Furthermore, equilibrium existence arguments generally require assumptions on the cost mapping, such as continuity,

1. Initialize travel times to free-flow and parking probabilities to 1, iteration counter $k \leftarrow 1$.
2. Identify optimal policies $\pi^*(s)$ for each destination s, and assign to all vehicles. (Figure 4.4)
3. Increment k
4. Simulate cell transmission model for given policies. (Figure 4.3)
5. *(Optional.)* Repeat previous step multiple times to generate empirical distribution of t and p based on repeated simulation.
6. Identify optimal policies $\pi^*(s)$ for each destination s. (Figure 4.4)
7. For each vehicle heading to destination s, switch its policy to $\pi^*(s)$ with probability $1/k$
8. Unless gap sufficiently small, return to step 3.

Figure 4.5: Method of successive averages equilibrium heuristic.

which may not be satisfied with the model presented in the previous sections, which is both discrete and stochastic — owing to the random departures from parking spaces, multiple samples may be needed to obtain reliable estimates for the parking probabilities p. For these reasons, we present only a heuristic which aims to produce a near-equilibrium solution with a small gap (as defined by the difference between the labels of the chosen policies and optimal policies for those travelers). This heuristic is based on the well-known method of successive averages, and is presented in Figure 4.5.

4.7 Demonstration

This section shows a simple example of the parking search simulation based on CTM. Figure 4.6 presents the network structure and its attributes of this example. In this network, all travelers are attempting to park to attend a special event at the location of the star. The network consists of two ring roads connected by small local streets. Travelers may park either at the small lot (most convenient, but least capacity), the large lot (largest capacity, but a far walk away), or along the small local streets (intermediate in capacity and distance). This network represents an extreme case of time dependency, when parked vehicles never clear — in other words, the likelihood of finding parking at different locations will vary significantly depending on when someone begins searching, and again during the search itself. Therefore, this is a suitable demonstration for the features of the dynamic model.

The algorithm were implemented in C and runs on a 2.60 GHz Intel machine with 4 GB memory running Windows 7. Figure 4.7 shows the convergence rate of the algorithm, reporting the average excess cost obtained after a given amount of computation time has elapsed. The figure shows that the relative gap is decreasing as time elapses thought it may have temporary increase at some point. The convergence is notably less smooth and direct than the static model presented in the previous chapter, which is logical because the dynamic model is a stochastic simulation based on a discrete model.

Figure 4.8 shows how the number of parked vehicles change with running iteration by iteration in the process of finding equilibrium policies. In the early iterations, very few vehicles park, because initially all drivers are assigned to the

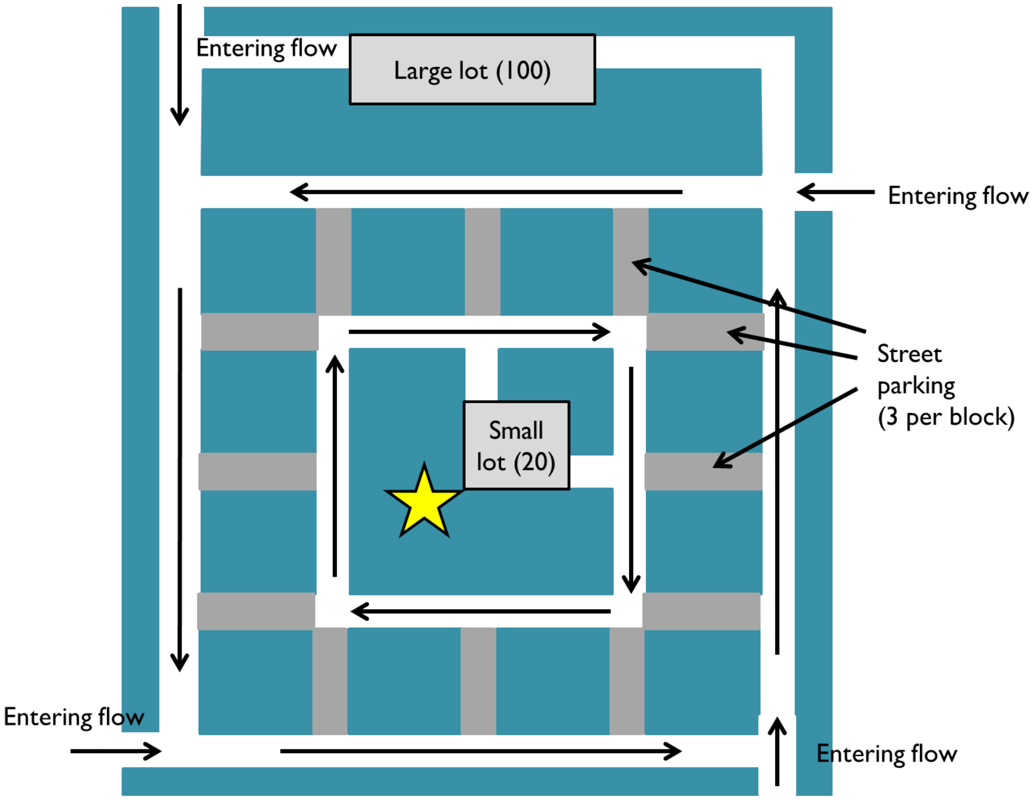


Figure 4.6: The demonstration network for the dynamic parking assignment

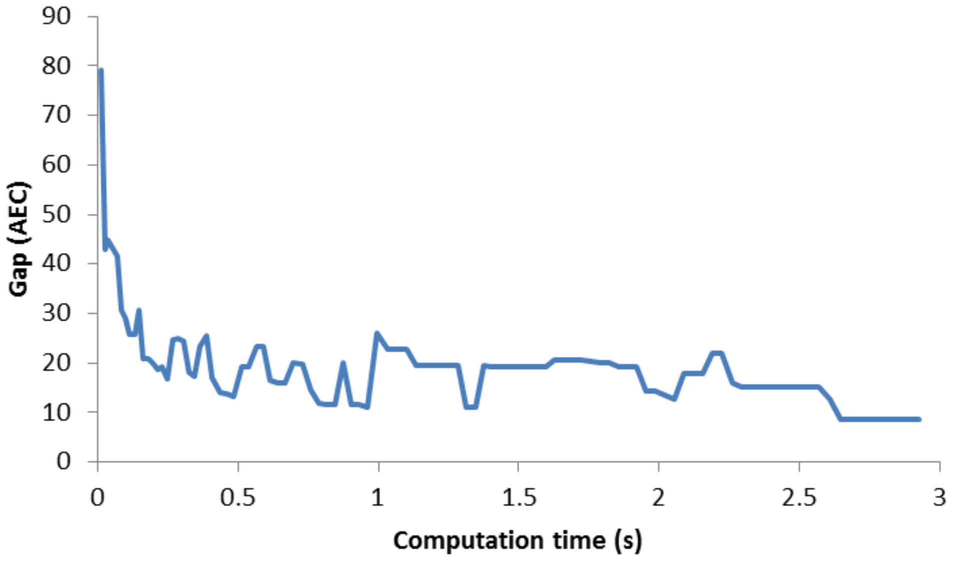


Figure 4.7: The DTA model convergence rate

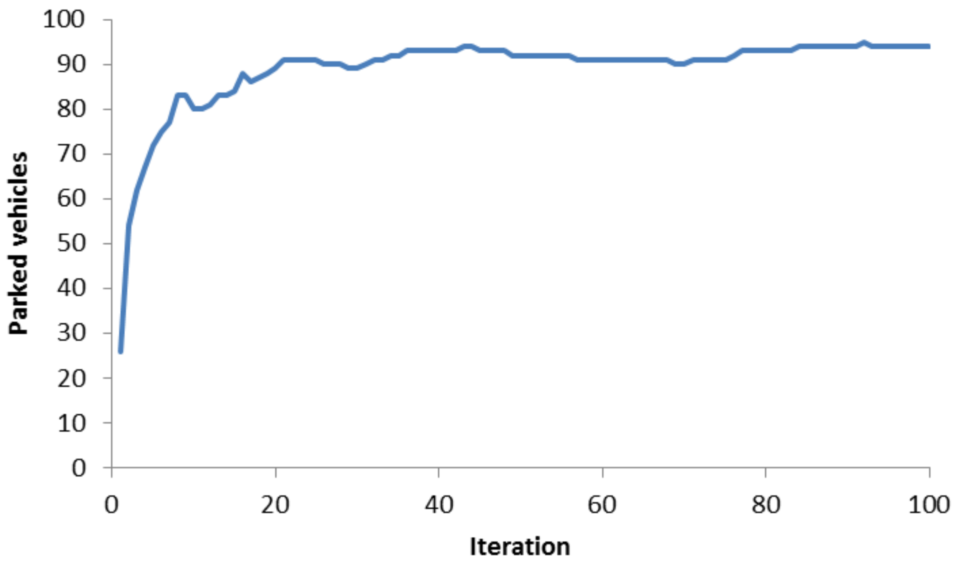


Figure 4.8: The number of parked vehicles in each iteration

best policy with no congestion, which is to aim to park at the small lot. This causes significant congestion, and in fact most drivers will circle endlessly in the first iteration since the capacity of the small lot is less than the number of vehicles. (This example is a challenging instance for the dynamic algorithm, since parked vehicles never clear and this type of infinite cycling can occur.) As iterations progress, more and more vehicles are shifted to alternate policies which involve searching for street parking or driving to the large lot.

Figure 4.9 shows how the parking distribution after 100 iterations. The size and number of the circle indicate how many vehicles parked at that location. Notice that even though the small parking lot is the nearest location to the final destination, there are still many vehicles parked at other locations because the small parking lot does not have enough space for all the parking demand. Also notice that the total number of parked vehicles is 96 which is not equal to the demand 100. This is because we only run our model 100 iterations. Based on the 100 iterations, the chosen policies only allow 96 vehicles to park but the other 4 vehicles are set to continue driving for a desirable location. We did not run more iterations because after 100 iteration, the result only changes slowly. This is largely due to our naive implementation of the Method of Successive Averages, since step 7 chooses the vehicles to update randomly. A more intelligent version would shift vehicles with higher-cost policies with greater probability, rather than treating all vehicles uniformly in this regard, but developing this version is beyond the scope of the dissertation. For many practical applications, vehicles will vacate parking spots as well, which would also eliminate the problem of vehicles cycling endlessly. We are not only interested in parking distribution, since it is dynamic traffic assignment we also care about how the parking spaces occupied as time goes by.

Figure 4.10 shows the number of parked vehicles in each location over time. Notice that some of the first vehicles will first choose on-street parking, correctly anticipating that the small lot would be filled by the time they arrive, *even though at the present time it is completely empty.*. This shows the strength of the equilibrium modeling concept. The small lot then fills, with drivers eventually choosing the large lot. If we adjust the walking cost and free flow speed to some point, these profiles and results may change.

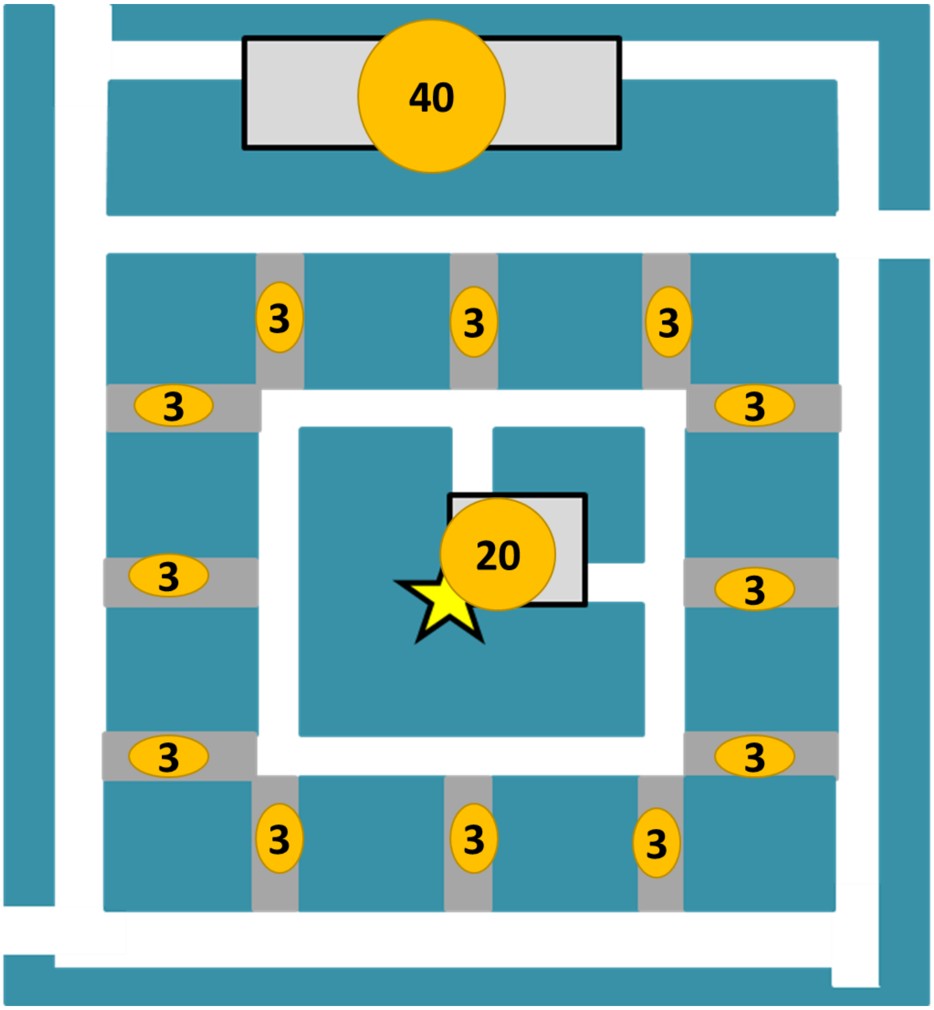


Figure 4.9: The number of parked vehicles at each parking location

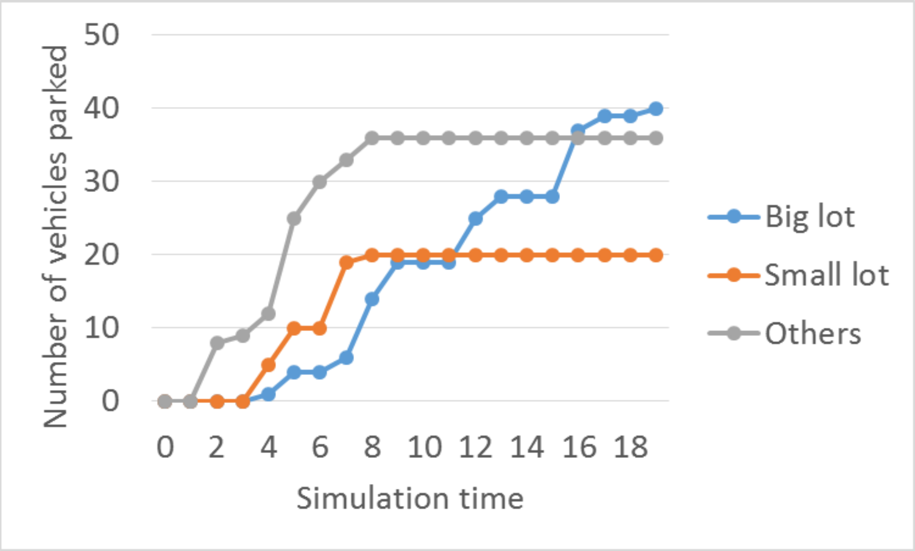


Figure 4.10: The number of parked vehicles at different locations

4.8 Sensitivity Analysis

A natural question may come out when we deal with traffic assignment models: what would happen if change some parameters. For instance, if the traffic or parking demand changes how the parking distribution would change. Doing such analysis is very useful since it illustrates how the model may be used in practice. Predicting the possible parking distribution change is very helpful to set up parking guidance or redesign parking services.

As for this analysis for the example in Section 4.7, we reduced the demand by 10, 30, 50 and 80 percent, respectively. Figure 4.11 shows that if we decrease the demand, the number of parked vehicles in the big parking lot will reduce dramatically, while it in the small lot does not change much.

Since we know for this special example, drivers mostly like to park in the small lot or nearby links, we want to see if we increase the parking capacity of the small parking lot, how the parking distribution will change. Figure 4.12 shows if we increase the parking capacity of the small lot by 10, 30, 50, 80 and 100 percent, respectively, how the number of parked vehicles will change at different locations. Clearly, if there are more parking spaces at the small lot, many people do not need to get to the far-away big lot to park, so its number is decreasing. Notice, all the parking spaces in the small lot are always occupied.

Another reason that would impact parking distribution is the parking price rate. Figure 4.13 shows how the price changing in the small lot would impact the parking lot occupancy and the profit made from the ticket sale. As expected, increase

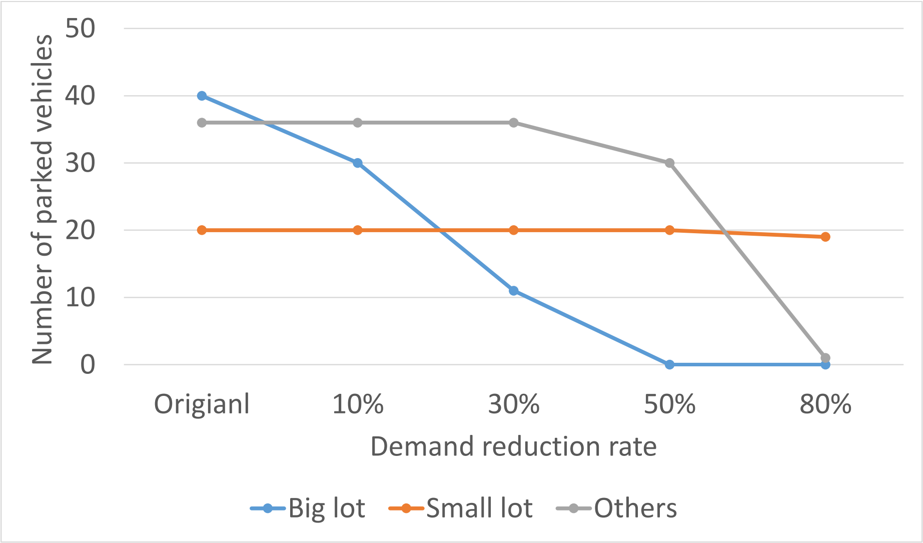


Figure 4.11: The number of parked vehicles at different demand levels

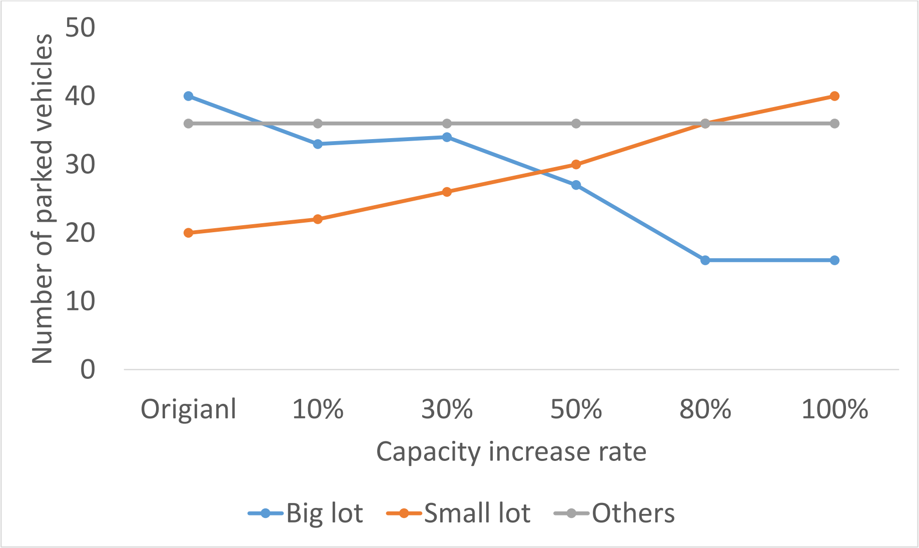


Figure 4.12: The number of parked vehicles with different parking capacity of the small parking lot

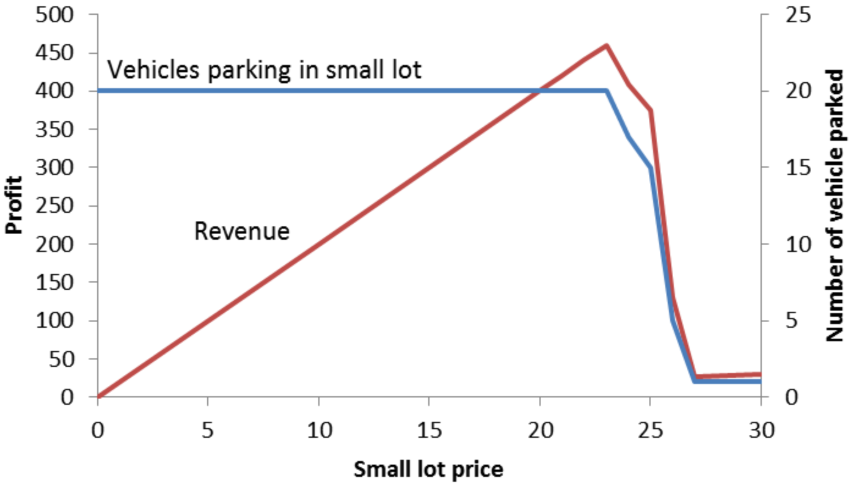


Figure 4.13: The number of parked vehicles in the small lot with different pricing strategies

the price will reduce the number of vehicles parking in the lot, but the profit will first increase until getting to some point where the total profit starts dropping. This analysis is very useful for evaluating price strategies in garage or other kind of parking lot management.

4.9 Conclusion and Practical Considerations

The model described in this chapter represents a dynamic traffic assignment model which has been extended to include delays and congestion resulting from the parking search process. In contrast to previous network-based approaches, these delays and congestion effects are modeled explicitly, using an online shortest path approach. The resulting equilibrium state is a natural generalization of the Wardrop condition traditionally used in transportation planning. Furthermore, the algorithms presented above are easily amenable to implementation in agent-based simulation. For large-scale instances, each component algorithm can be parallelized to decrease computation time. This model can be used as is to evaluate parking policies related to pricing and duration, both for routine conditions and special events. Its general principles can also serve as the basis for more involved investigations concerning real-time parking information or dynamic pricing policies; both of these are valuable topics for future research. Other future research topics include extending the demand model to handle trip chains (a vehicle departing from one parking space may head to several other destinations, including parking at each one, before returning to the origin), or developing alternative algorithms for reaching near-equilibrium solutions.

From the standpoint of practical implementation, the additional data requirements for this model, relative to existing dynamic traffic assignment models, are the number of available parking spaces on each cell, the mean parking duration on each cell, and the walking distances between each cell and each destination. The latter data is relatively easy to estimate, given the network topology and an assumed walking speed. It may be easier to estimate the total number of available parking spaces on a link, and divide them evenly among the cells, or to estimate the mean parking duration with a proxy (such as a fraction of the maximum allowable time.)

Chapter 5

Conclusion

5.1 Summary and Contribution

This dissertation developed three parking search models. To our knowledge, these models were the first to incorporate parking search behavior into traffic assignment models while considering the stochasticity in the parking availability and drivers' parking choice behavior. The models can be used to evaluate proposed parking planning strategies or help provide more predicted parking information to travelers. This would improve the parking services and reduce congestion levels because it would work better than present traditional traffic assignment models which systematically underestimate the parking and traveling demand by neglecting the parking search process. As mentioned in Chapter 1, parking traffic may account for a considerable portion of total flow, so it is necessary to incorporate parking into traffic assignment models to obtain accurate forecasts of traffic and congestion levels.

To do this, we must consider the reason that causes parking searching traffic. The first reason is uncertainty: before drivers get to their destinations, they do not know the locations of available parking space. The result is ruising traffic and cycling as drivers seek a parking space close to the destination. Commonly, drivers' decisions on where to park depend on their experience and perceived information. If there is real time parking availability information available to drivers, they will can make improved decisions whether or not she to park at an available space or continue searching for a closer space. The three models aim to capture this stochasticity both in parking supply side and in driver's choice behavior side.

Chapter 2 focuses on modeling a single drivers parking search behavior with history dependence. The term "history dependence" refer to the driver's memory of parking space availability during passed visits. In this model, a so called "asymptotic reset" probability function is built. This function can reflect impacts of the driver's past observations of parking availability on his/her future route and parking choice decisions. This is the most important contribution of this model as it generalized the concepts of "full reset", in which the probability of finding a parking space is independent of any past observations, and the "no reset", in which parking availability is completely determined by past observation. In asymptotic reset, there is large dependence by past observations which are recent, which decay exponentially to the *a priori* probabilities used by full reset. Finally the single driver's parking search behavior is modeled as a Markov decision process and solved with a value iteration method.

Chapter 3 deals with multiple drivers' parking search behavior. Similar to the model for a single driver, this model also captures uncertainty in parking space availability and drivers' parking choice. The difference, this models currently does not consider memory impacts, but take into account the mutual dependence between route-parking choice and the probability of parking availability. The first contribution of this model is that the network is transformed with dummy links and nodes which can account for splitting flows into two types: passing through flow and searching parking flow. Vehicles of searching parking flows can either park through an added dumy link or continue driving back to regular link and searching parking again. The second contribution is that an traffic equilibrium solution algorithm is developed. This can help this model to be used in practice or embedded into currently static traffic assignment models.

Chapter 4 expanded the theories and principles in the static parking search model into a cell transmission (CTM) based dynamic traffic assignment (DTA) framework. Compare to the static model, this DTA model not only can describe the stochasticity as in the static model, but also can reflect changes to demand, flows, parking choice and occupancy over time. The policy-based equilibrium of this model can be achieved by running the solution algorithm which can be used to evaluate parking planning alternatives or parking controls for events. It can also be used to predict real time traffic and parking conditions.

Overall, to our knowledge the innovative models developed in this dissertation are the first models that combine traffic assignment and stochastic parking search process, and have big potentials to be used in practice and benefit the transportation system.

5.2 Future Work

Future work includes improving the models and generalizing them into more conditions. The first consideration is expand the history dependence into multiple vehicles models. This makes it much more interesting as not only the memory impacts should considered, but also the interaction between drivers or even between their memories should also be considered.

Another possible work is to make the models reflect flow-congestion interactions. Now, one of the biggest assumptions for the first two models is that the link travel times are constant but not flow dependent. this is because the time we started modeling it, we try to simplify it and focus on the memory impacts but not other things. However, in the future, developing flow dependent link cost functions and incorporate it into the models can make them more realistic.

Incorporate more information into present models. While currently we only consider memory impacts on parking choice behaviors, in the future, we may consider include spatial correlations into the model. For instance, if a link has no opening parking spot, it is highly possible that its neighbor link has no spot either. This requires bigger state spaces for modeling it as a MDP.

Run the models even on bigger networks. This dissertation focuses on model construction and algorithm design, so it does not test them on very large network. one of the reason is that running the models on a large network usually needs a very long time which seems not worthy of in current stage. The other reason is that it is hard to get real data, especially parking information for large networks. Making assumptions on the these data for large networks may causes big errors.

Comparing the DTA model with the static model and other common DTA models may also give surprises. As we mentioned before, without considering parking, there will be a big underestimation of traffic flows in some area. After comparison it may help understand why some traffic management or control strategies that have been tested with simulations and proved to be feasible do not work in reality. Another difference is that common DTA or even static assignment models assume each vehicle ends at its final destination point, usually the centroid point. This requires some vehicles with the same destination centroid concentrate at a point at some time which may cause short term extreme local congestion. While the DTA in this dissertation let vehicles park on the link or to some parking lot with a “scattered” way. there will probably no short term extreme congestion at all.

Appendix

Appendix 1

Probability of Parking Availability

In this section we develop one potential specification for the p_{ij} functions. For each link, assume that the total number of parking spaces is given by the positive integer S_{ij}. (If no parking is available, p_{ij} is uniformly zero.) Further assume that the mean parking time for any vehicle on this link is μ_{ij}, that the parking duration for any vehicle is exponentially distributed with this mean independent of any other, and that the headways between vehicles arriving to park are also exponentially distributed with mean headway $1/x_{ij,S}$. Omitting subscripts for brevity, we now derive p_{ij} as follows.

Define the state variable $P \in \{0, \cdots, S\}$ denoting the number of occupied parking spaces. The evolution of P can be modeled as a Markov chain with transition matrix

$$\begin{bmatrix} 1-x & x & 0 & \cdots & 0 & 0 \\ 1/\mu & 1-x-1/\mu & x & \cdots & 0 & 0 \\ 0 & 2/\mu & 1-x-1/\mu & \cdots & 0 & 0 \\ 0 & 0 & 3/\mu & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1-(S-1)/\mu - x & x \\ 0 & 0 & 0 & \cdots & S/\mu & 1-S/\mu \end{bmatrix} \tag{1.1}$$

whose eigenvector gives the steady-state probabilities π_k for the number of occupied parking spaces. Exploiting the matrix structure, we can calculate this eigenvector recursively:

$$\frac{\pi_1}{\pi_0} = \mu x \tag{1.2}$$
$$\frac{\pi_2}{\pi_1} = \frac{\mu x}{2} \tag{1.3}$$
$$\vdots \tag{1.4}$$
$$\frac{\pi_n}{\pi_{n-1}} = \frac{\mu x}{n} \tag{1.5}$$
$$\vdots \tag{1.6}$$
$$\frac{\pi_S}{\pi_{S-1}} = \frac{\mu x}{S} \tag{1.7}$$

so $\pi_k \propto (\mu x)^k/k!$ and

$$\pi_k = \frac{(\mu x)^k/k!}{\sum_{k'=0}^{S} (\mu x)^{k'}/(k')!} \tag{1.8}$$

Thus we have

$$p(x) = 1 - \pi_S = \frac{\sum_{k'=0}^{S-1} (\mu x)^{k'}/(k')!}{\sum_{k'=0}^{S} (\mu x)^{k'}/(k')!} \tag{1.9}$$

With this specification of $p(x)$, $xp(x)$ is strictly increasing as shown below. Application of l'Hospital's rule shows that $C = \lim_{x\to\infty} xp(x) = S/\mu$ in accordance with intuition. (The maximum rate new vehicles can park is the number of available spaces multiplied by the average rate parked vehicles depart.)

Proposition 14. If parking probabilities are given by (1.9) then $xp(x)$ is strictly increasing for $x > 0$.

Proof. Let $f(x) = xp(x)$. Both the numerator and denominator of f are strictly positive whenever x is; therefore to show that $f'(x) > 0$ it is enough to show that the numerator of its derivative is strictly positive, that is,

$$\left(\sum_{k=0}^{S}\frac{\mu^k}{k!}x^k\right)\left(\sum_{k=0}^{S-1}\frac{\mu^k}{(k+1)!}x^k\right)-\left(\sum_{k=0}^{S-1}\frac{\mu^k}{k!}x^k\right)\left(\sum_{k=0}^{S}\frac{\mu^k}{(k+1)!}x^k\right)>0 \tag{1.10}$$

Noting that the first term can be replaced by

$$\left(\sum_{k=0}^{S}\frac{\mu^k}{k!}x^k\right)\left(\sum_{k=0}^{S}\frac{\mu^k}{(k+1)!}x^k\right)-\frac{\mu^S x^S}{(S+1)!}\left(\sum_{k=0}^{S}\frac{\mu^k}{(k)!}x^k\right) \tag{1.11}$$

and the second by

$$\left(\sum_{k=0}^{S}\frac{\mu^k}{k!}x^k\right)\left(\sum_{k=0}^{S}\frac{\mu^k}{(k+1)!}x^k\right)-\frac{\mu^S x^S}{S!}\left(\sum_{k=0}^{S}\frac{\mu^k}{(k+1)!}x^k\right) \tag{1.12}$$

it is enough to show that

$$\frac{\mu^S x^S}{S!}\left(\sum_{k=0}^{S}\frac{\mu^k}{(k+1)!}x^k\right)>\frac{\mu^S x^S}{(S+1)!}\left(\sum_{k=0}^{S}\frac{\mu^k}{k!}x^k\right) \tag{1.13}$$

Comparing term by term,

$$\frac{\mu^S x^S}{S!}\frac{\mu^k}{(k+1)!}x^k=\frac{\mu^S x^S}{(S+1)!}\frac{S+1}{k+1}\frac{\mu^k}{k!}x^k \tag{1.14}$$

Since $S \geq k$ for all terms in the sum (and $S > k$ for all but the last) this establishes the result. □

When S is large, the formula (1.9) may be difficult to evaluate numerically, so an approximation is given here. Define $B_i = (\mu x)^i/i!$ so (1.9) is simply

$$p(x) = 1 - \frac{B_S}{\sum_{k=0}^{S} B_k} \tag{1.15}$$

Now $B_S = e^{\log B_S}$ and

$$\log B_k \approx k\log(\mu x) - k\log k + k - \frac{1}{2}\log(2\pi k) \tag{1.16}$$

using Stirling's approximation to $k!$. Furthermore, using the Taylor expansion of e^x we have

$$B_k \approx 1 + k\log(\mu x) - k\log k + k - \frac{1}{2}\log(2\pi k) \tag{1.17}$$

which can be directly substituted in the numerator of (1.15). In the denominator, using the asymptotic expressions

$$\sum_{k=1}^{S} \log k \approx (S + \frac{1}{2})\log S - S + \frac{1}{2}\log(2\pi) \tag{1.18}$$

and

$$\sum_{k=1}^{S} k\log k \approx K - \frac{S^2}{4} + \frac{S(S+1)}{2}\log S + \frac{\log S}{12} \tag{1.19}$$

with $K = \frac{1}{12} - \zeta'(-1)$ where ζ is the Riemann zeta function, we finally obtain

$$p(x) \approx 1 - \frac{1 + S\left(\log\frac{\mu x}{S} + 1\right) - \frac{1}{2}\log(2\pi S)}{1 - K + S - \frac{1}{4}\log(8\pi S) + \frac{S(S+1)}{2}\left(\log\frac{\mu x}{S} + 1\right) + \frac{S^2}{4} + \frac{S}{2}\left(1 - \log(2\pi S)\right)} \tag{1.20}$$

after some algebra. While an approximation, this function is easier to evaluate for large S (say, if a parking link represents a large lot.)

Bibliography

Richard Arnott and Eren Inci. The stability of downtown parking and traffic congestion. *Journal of Urban Economics*, 68(3):260–276, 2010. ISSN 0094-1190.

Richard Arnott and John Rowse. Modeling parking. *Journal of Urban Economics*, 45(1):97–124, 1999. ISSN 0094-1190. doi: http://dx.doi.org/10.1006/juec.1998.2084. URL http://www.sciencedirect.com/science/article/pii/S0094119098920848.

Richard Arnott, Andre de Palma, and Robin Lindsey. Does providing information to drivers reduce traffic congestion? *Transportation Research Part A: General*, 25(5):309–318, 1991. ISSN 0191-2607. doi: http://dx.doi.org/10.1016/0191-2607(91)90146-H. URL http://www.sciencedirect.com/science/article/pii/019126079190146H.

Kay W Axhausen and John W Polak. Choice of parking: stated preference approach. *Transportation*, 18(1):59–81, 1991. ISSN 0049-4488.

Kay W Axhausen, John W Polak, M. Boltze, and J. Puzicha. Effectiveness of the parking gudance system in frankfurt am main. *Traffic Engineering and Control*, 35 (5):304–309, 1994.

Hillel Bar-Gera. Origin-based algorithm for the traffic assignment problem. *Transportation Science*, 36(4):398–417, 2002. ISSN 0041-1655. doi: 10.1287/trsc.36.4.

398.549. URL http://pubsonline.informs.org/doi/abs/10.1287/trsc.36.4.398.549.

Hillel Bar-Gera. Transportation network test problems, 2014. URL http://www.bgu.ac.il/~bargera/tntp/. Accessed June 25, 2014.

Itzhak Benenson, Karel Martens, and Slava Birfir. Parkagent: An agent-based model of parking in the city. *Computers, Environment and Urban Systems*, 32(6):431–439, 2008. ISSN 0198-9715. doi: http://dx.doi.org/10.1016/j.compenvurbsys.2008.09.011. URL http://www.sciencedirect.com/science/article/pii/S0198971508000689.

Dimitri P. Bertsekas. *Dynamic programming and optimal control*, volume 2. Athena Sientific, Belmont, MA, 4 edition, 2012.

Gennaro N. Bifulco. A stochastic user equilibrium assignment model for the evaluation of parking policies. *European Journal of Operational Research*, 71(2):269–287, 1993. ISSN 0377-2217.

Stephen D. Boyles. *Operational, supply-side uncertainty in transportation networks: causes, effects, and mitigation strategies*. Thesis, The University of Texas at Austin, 2009.

Raymond K. Cheung. Iterative methods for dynamic stochastic shortest path problems. *Naval Research Logistics (NRL)*, 45(8):769–789, 1998. ISSN 1520-6750.

Ruben Corthout, Gunnar Fltterd, Francesco Viti, and Chris M. J. Tampre. Non-unique flows in macroscopic first-order intersection models. *Transportation Research Part B: Methodological*, 46(3):343–359, 2012. ISSN 0191-2615.

Luca D'Acierno, Mariano Gallo, and Bruno Montella. Optimisation models for the urban parking pricing problem. *Transport Policy*, 13(1):34–48, 2006. ISSN 0967-070X.

Carlos F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994. ISSN 0191-2615.

Carlos F. Daganzo. The cell transmission model, part ii: network traffic. *Transportation Research Part B: Methodological*, 29(2):79–93, 1995. ISSN 0191-2615.

Benjamin Davis, T. Dutzik, and P. Baxandall. Transportation and the new generation: Why young people are driving less and what it means for transportation policy. Report, Frontier Group, 2012. URL `http://www.frontiergroup.org/reports/fg/transportation-and-new-generation`.

Mauro DellOrco, Michele Ottomanelli, and Domenico Sassanelli. Modelling uncertainty in parking choice behaviour. In *82nd Annual Meeting of the Transportation Research Board*, 2003.

Karel Dieussaert, Koen Aerts, Thrse Steenberghen, Sven Maerivoet, and Karel Spitaels. Sustapark: an agent-based model for simulating parking search. In *AGILE International Conference on Geographic Information Science, Hannover*, 2009.

Francisco Facchinei and Jong-Shi Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems.* Finite-dimensional Variational Inequalities and Complementarity Problems. Springer, 2003. ISBN 9780387218151. URL http://books.google.com/books?id=lX_7Rce3_Q0C. v. 2.

Bernard P. Feeney. A review of the impact of parking policy measures on travel demand. *Transportation Planning and Technology*, 13(4):229–244, 1989.

Mariano Gallo, Luca D'Acierno, and Bruno Montella. A multilayer model to simulate cruising for parking in urban areas. *Transport Policy*, 18(5):735–744, 2011. ISSN 0967-070X. doi: http://dx.doi.org/10.1016/j.tranpol.2011.01.009. URL http://www.sciencedirect.com/science/article/pii/S0967070X11000217.

Guido Gentile. Linear user cost equilibrium. 2009. Working paper.

Allen Greenberg. Contemporary approaches to parking pricing. 2012. URL http://www.parkingspot.parking.org/media/129582/fhwa%20parking%20pricing%20primer.pdf.

Randolph W. Hall. The fastest path through a network with random time-dependent travel times. *Transportation Science*, 20(3):182–188, 1986. ISSN 00411655. doi: 10.2307/25768228. URL http://www.jstor.org/stable/25768228.

David A. Hensher and Jenny King. Parking demand and responsiveness to supply, pricing and location in the sydney central business district. *Transportation Research Part A: Policy and Practice*, 35(3):177–196, 2001. ISSN 0965-8564. doi: http:

//dx.doi.org/10.1016/S0965-8564(99)00054-3. URL http://www.sciencedirect.com/science/article/pii/S0965856499000543.

Stephane Hess and W. Polak John. An analysis of parking behaviour using discrete choice models calibrated on sp datasets. Report, European Regional Science Association, Aug 2004. URL http://ideas.repec.org/p/wiw/wiwrsa/ersa04p60.html. ERSA conference papers.

Stephane Hess and John W Polak. Mixed logit estimation of parking type choice. In *83rd Annual Meeting of the Transportation Research Board, Washington, DC*, 2004.

John D. Hunt and S. Teply. A nested logit model of parking location choice. *Transportation Research Part B: Methodological*, 27(4):253–265, 1993. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/0191-2615(93)90035-9. URL http://www.sciencedirect.com/science/article/pii/0191261593900359.

William H. K. Lam, Zhi-Chun Li, Hai-Jun Huang, and S. C. Wong. Modeling time-dependent travel choice problems in road networks with multiple user classes and multiple parking facilities. *Transportation Research Part B: Methodological*, 40(5):368–395, 2006. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2005.05.003. URL http://www.sciencedirect.com/science/article/pii/S0191261505000718.

William H. K. Tam Mei-Lam Bell M. G. H. Lam. Optimal road tolls and parking charges for balancing the demand and supply of road transport facilities.

Transportation and traffic theory in the 21st century., pages 561–582, 2002. ISSN 0080439268.

Thomas A. Lambe. Driver choice of parking in the city. *Socio-Economic Planning Sciences*, 30(3):207–219, 1996. ISSN 0038-0121. doi: http://dx.doi.org/10.1016/0038-0121(96)00008-0. URL `http://www.sciencedirect.com/science/article/pii/0038012196000080`.

Fabien Leurent and Houda Boujnah. Traffic equilibrium in a network model of parking and route choice, with search circuits and cruising flows. *Procedia - Social and Behavioral Sciences*, 54(0):808–821, 2012. ISSN 1877-0428. doi: http://dx.doi.org/10.1016/j.sbspro.2012.09.797. URL `http://www.sciencedirect.com/science/article/pii/S1877042812042590`.

Zhi-Chun Li, Hai-Jun Huang, WilliamH K. Lam, and S. C. Wong. A model for evaluation of transport policies in multimodal networks with road and parking capacity constraints. *Journal of Mathematical Modelling and Algorithms*, 6(2): 239–257, 2007a. ISSN 1570-1166. doi: 10.1007/s10852-006-9040-7. URL `http://dx.doi.org/10.1007/s10852-006-9040-7`.

Zhi-Chun Li, Hai-Jun Huang, WilliamH K. Lam, and S. C. Wong. Time-differential pricing of road tolls and parking charges in a transport network with elastic demand. In *Proceedings of the 17th international symposium on transportation and traffic theory*, pages 55–85, 2007b.

Zhi-Chun Li, WilliamH K. Lam, S. C. Wong, Hai-Jun Huang, and Dao-Li Zhu. Reliability evaluation for stochastic and time-dependent networks with multiple parking facilities. *Networks and Spatial Economics*, 8(4):355–381, 2008. ISSN 1566-113X. doi: 10.1007/s11067-007-9018-7. URL http://dx.doi.org/10.1007/s11067-007-9018-7.

Michael J. Lighthill and Gerald Beresford Whitham. On kinematic waves. ii. a theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 229(1178):317–345, 1955. ISSN 1364-5021.

Karel Martens and Itzhak Benenson. Evaluating urban parking policies with agent-based model of driver parking behavior. *Transportation Research Record: Journal of the Transportation Research Board*, 2046(1):37–44, 2008. doi: 10.3141/2046-05. URL http://dx.doi.org/10.3141/2046-05.

Yu Nie. Multi-class percentile user equilibrium with flow-dependent stochasticity. *Transportation Research Part B: Methodological*, 45(10):1641–1659, 2011. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2011.06.001. URL http://www.sciencedirect.com/science/article/pii/S0191261511000889. (Marco).

Yu Nie, Jingtao Ma, and H. Michael Zhang. A polymorphic dynamic network loading model. *Computer-Aided Civil and Infrastructure Engineering*, 23(2):86–103, 2008. ISSN 1467-8667. doi: 10.1111/j.1467-8667.2007.00525.x. URL http://dx.doi.org/10.1111/j.1467-8667.2007.00525.x.

M. S. Nour Eldin, T. Y. El-Reedy, and H. K. Ismail. A combined parking and traffic assignment model. *Traffic engineering and control*, 22:524–530, 1981.

Gregory Pierce and Donald Shoup. Getting the prices right: an evaluation of pricing parking by demand in san francisco. *Journal of the American Planning Association*, 79(1):67–81, 2013. ISSN 0194-4363.

George H. Polychronopoulos and John N. Tsitsiklis. Stochastic shortest path problems with recourse. *Networks*, 27(2):133–143, 1996. ISSN 1097-0037. doi: 10.1002/(SICI)1097-0037(199603)27:2⟨133::AID-NET5⟩3.0.CO;2-L. URL `http://dx.doi.org/10.1002/(SICI)1097-0037(199603)27:2<133::AID-NET5>3.0.CO`.

J Scott Provan. A polynomialtime algorithm to find shortest paths with recourse. *Networks*, 41(2):115–125, 2003. ISSN 1097-0037.

Zhen Qian and Ram Rajagopal. Optimal occupancy-driven parking pricing under demand uncertainties and traveler heterogeneity: A stochastic control approach. *Transportation Research Part B: Methodological*, 67(0):144–165, 2014. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2014.03.002. URL `http://www.sciencedirect.com/science/article/pii/S0191261514000368`. (Sean).

Zhen Qian, Feng Xiao, and H. M. Zhang. Managing morning commute traffic with parking. *Transportation Research Part B: Methodological*, 46(7):894–916, 2012. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2012.01.011. URL `http://www.sciencedirect.com/science/article/pii/S019126151200015X`. (Sean) (Evan).

Paul I. Richards. Shock waves on the highway. *Operations research*, 4(1):42–51, 1956. ISSN 0030-364X.

Casey Ross. Boston limiting new parking as number of residences soars, 2013. URL http://www.boston.com/cars/news-and-reviews/2013/07/04/boston-limiting-new-parking-number-residences-soars/u3NDVOWQE2ze2RNn6lMcSP/story.html.

Donald C. Shoup. Cruising for parking. *Transport Policy*, 13(6):479–486, 2006. ISSN 0967-070X. doi: http://dx.doi.org/10.1016/j.tranpol.2006.05.005. URL http://www.sciencedirect.com/science/article/pii/S0967070X06000448.

Donald C. Shoup and American Planning Association. *The high cost of free parking.* Planners Press, American Planning Association, Chicago, updated. edition, 2011. ISBN 9781932364965 193236496X. URL Contributorbiographicalinformationhttp://catdir.loc.gov/catdir/enhancements/fy1208/2012359417-b.html.

Chris M. J. Tampre, Ruben Corthout, Dirk Cattrysse, and Lambertus H. Immers. A generic class of first order node models for dynamic macroscopic simulation of traffic flows. *Transportation Research Part B: Methodological*, 45(1):289–309, 2011. ISSN 0191-2615.

Shoupeng Tang, Tarun Rambha, Reese Hatridge, Stephen D. Boyles, and Avinash Unnikrishnan. Modeling parking search on a network using stochastic shortest paths with history dependence. *Transportation Research Record*, 2014. Accepted for publication.

Russell G. Thompson and Anthony J. Richardson. A parking search model. *Transportation Research Part A: Policy and Practice*, 32(3):159–170, 1998. ISSN

0965-8564. doi: http://dx.doi.org/10.1016/S0965-8564(97)00005-0. URL http://www.sciencedirect.com/science/article/pii/S0965856497000050.

Avinash Unnikrishnan and StevenTravis Waller. User equilibrium with recourse. *Networks and Spatial Economics*, 9(4):575–593, 2009. ISSN 1566-113X. doi: 10.1007/s11067-009-9114-y. URL http://dx.doi.org/10.1007/s11067-009-9114-y.

D Van der Goot. A model to describe the choice of parking places. *Transportation Research Part A: General*, 16(2):109–115, 1982. ISSN 0191-2607.

S. Travis Waller and Athanasios K. Ziliaskopoulos. On the online shortest path problem with limited arc cost dependencies. *Networks*, 40(4):216–227, 2002. ISSN 1097-0037. doi: 10.1002/net.10049. URL http://dx.doi.org/10.1002/net.10049.

Rashid Waraich and Kay Axhausen. Agent-based parking choice model. *Transportation Research Record: Journal of the Transportation Research Board*, 2319(-1):39–46, 2012. doi: 10.3141/2319-05. URL http://dx.doi.org/10.3141/2319-05.

Hai Yang, Wei Liu, Xiaolei Wang, and Xiaoning Zhang. On the morning commute problem with bottleneck congestion and parking space constraints. *Transportation Research Part B: Methodological*, 58(0):106–118, 2013. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2013.10.003. URL http://www.sciencedirect.com/science/article/pii/S0191261513001768.

Hope Yen and kristen Wyatt. Cities grow more than suburbs, 2012. URL http://www.today.com/id/47992439/ns/today-money/t/cities-grow-more-suburbs-first-time-years/#.UfNMLG3ODW6.

Isaak Yperman. *The link transmission model for dynamic network loading.* Thesis, Katholieke Universiteit Leuven, 2007.

Xiaoning Zhang, Hai-Jun Huang, and H. M. Zhang. Integrated daily commuting patterns and optimal road tolls and parking fees in a linear city. *Transportation Research Part B: Methodological*, 42(1):38–56, 2008. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2007.06.001. URL `http://www.sciencedirect.com/science/article/pii/S0191261507000653`.

Xiaoning Zhang, Hai Yang, and Hai-Jun Huang. Improving travel efficiency by parking permits distribution and trading. *Transportation Research Part B: Methodological*, 45(7):1018–1034, 2011. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/j.trb.2011.05.003. URL `http://www.sciencedirect.com/science/article/pii/S0191261511000531`.

Vita

Shoupeng Tang received the Bachelor of Engineering degree from Jilin University, China in 2006 and the Master of Engineering from Tongji University, China in 2009. He enrolled in the department of civil architectural and environmental engineering at the University of Texas at Austin in Fall 2010. After more than two year's research foundation construction, he started working on parking search problems and made it the dissertation topic. With the help from his advisor Stephen Boyles, he successfully solved the problem of how to incorporate the parking search behavior into traffic assignment models and finished the draft of this dissertation in October. Now, after one month's editing, he submitted this final version of the dissertation for assessment and approval by the dissertation committee.

Email address: shouptang@gmail.com

This dissertation was typeset with LaTeX[†] by Shoupeng Tang.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.