# 博士論文

# Dynamic Traffic Resource Allocation Problems with Behavioral Choice in Networks

（ネットワーク上の行動選択を考慮した動的交通資源配分問題）

早川　敬一郎

THE UNIVERSITY OF TOKYO

DOCTORAL THESIS

---

# Dynamic Traffic Resource Allocation Problems with Behavioral Choice in Networks

---


*Author:*
Keiichiro HAYAKAWA

*Supervisor:*
Professor. Eiji HATO


*A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy*

*in the*


Behavior in Networks Studies Unit
Department of Civil Engineering

*"Go as far as you can see; when you get there, you'll be able to see further."*

Thomas Carlyle

The University of Tokyo

# *Abstract*

School of Engineering
Department of Civil Engineering

Doctor of Philosophy

**Dynamic Traffic Resource Allocation Problems with Behavioral Choice in Networks**

by Keiichiro HAYAKAWA

In the 20th Century, cities and mobility services have dramatically changed by the penetration of "automobiles", even though they actually are the "automobiles manipulated by human beings." In the coming age, penetrating of the true "automobiles", namely automated-vehicles (AVs), and mobility services using such vehicles as well-controllable tools, they will change again, dramatically. In this thesis, we try to show the direction of transportation research in such age and propose mechanisms, as well as algorithms, that make important roles in such a new type of mobility society. Specifically, we try to construct dynamic traffic allocation algorithms that effectively utilize the limited traffic resources while considering users' behavioral choice in the traffic network.

The penetration of AVs arises issues both in urban and suburban areas. In urban areas, the traffic volume of automobiles may increase with the improvement of convenience by AVs. Thus a traffic control is required which disperses the traffics concentrating at specific points. In Chapter 3, we propose a traffic control algorithm to prevent the occurrence of gridlock phenomena due to traffic concentration in urban over-saturated networks. Meanwhile, in the suburbs, efficient mobility services that aggregate the travel demands of multiple users are desired in order to realize the fruitful activities of users with limited traffic resources. In Chapter 4, we propose an algorithm that realizes efficient traffic resource allocation while considering space–time constraints of each user in a suburban area with sparse supply and demand. In both settings, the important point to note is the non-cooperative relationship between administrators of traffic controls or services and users actually using traffic resources. The "price of anarchy" defined as the gap between the system optimal and the user equilibrium state grows larger along with the improvement in convenience due to the spread of AVs. To compensate the gap, the pricing mechanism plays an important role. In Chapter 5, we propose a dynamic pricing algorithm to appropriately operate limited traffic resources on the premise of selfish decision making by users.

# *Acknowledgements*

First and foremost, I would like to thank my supervisor Prof. Eiji HATO whose enormous support and resourceful comments were invaluable not only in this theses but also in my whole research works. My horizons were broadened by a lot of substantive discussions with him. I would also like to my co-supervisors Prof. Noboru HARATA, Prof. Takamasa IRYO, Prof. Daisuke FUKUDA and Prof. Maddegedara LALITH who gave me invaluable comments and warm encouragements. I would thank Prof. Enrico GERDING and Prof. Sebastian STEIN who gave me the opportunity to have discussions about game theory and mechanism design. I am also grateful to Prof. Makoto CHIKARAISHI, Dr. Yusuke HARA, Dr. Junji URATA, Dr. Hideki YAGINUMA, and many other researchers for having fruitful discussions.

My research works are also greatly supported by TOYOTA CENTRAL R&D Labs., Inc.. Especially, I would like to thank Dr. Takahiro SHIGA from whom I received a lot of encouragement for engaging fundamental research works. I am grateful to all colleges for having both theoretical and practical discussions.

I would also like to express my gratitude to my family. I would like to thank my mother Reiko who has made extensive effort to support what I would like to do. I would also like to thank my father Masahiro who left me the core of the way of my life although he has passed away when I was 2 years old. I would also like to thank my brother Yuichiro for having discussions from the viewpoint of other domains. Finally, I am grateful to my wife Kuniko for her extensive support for all my research works and understanding for my pursuit of completion of this thesis. I greatly thank her contribution and appreciate her trust in me.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| **ABM** | **A**ctivity **B**ased **M**odeling |
| **ACC** | **A**daptive **C**ruise **C**ontrol |
| **AV** | **A**utomated **V**ehicle |
| **BB** | **B**udget **B**alanced |
| **BNIC** | **B**ayesian-**N**ash **I**ncentive **C**ompatible |
| **DARP** | **D**ial **A** **R**ide **P**roblem |
| **DSIC** | **D**ominant **S**trategy **I**ncentive **C**ompatible |
| **DSO** | **D**ynamic **S**ocial **O**ptimum |
| **DTC** | **D**ynamic **T**raffic **C**ontrol |
| **DUE** | **D**ynamic **U**ser **E**quilibrium |
| **FCFS** | **F**irst-**C**ome **F**irst-**S**erved |
| **FIFO** | **F**irst-**I**n **F**irst-**O**ut |
| **GA** | **G**enetic **A**lgorithm |
| **ILP** | **I**nteger **L**inear **P**rogramming |
| **IR** | **I**ndividually **R**ational |
| **KW** | **K**inematic **W**ave |
| **LKA** | **L**ane **K**eeping **A**ssist |
| **LP** | **L**inear **P**rogramming |
| **MaaS** | **M**obility **as** **a** **S**ervice |
| **MAPF** | **M**ulti-**A**gent **P**ath **F**inding Problem |
| **MDP** | **M**arkov **D**ecision **P**rocess |
| **MFD** | **M**acroscopic **F**undamental **D**iagram |
| **MIP** | **M**ixed **I**nteger **P**rogramming |
| **MPEC** | **M**athematical **P**rogramming with **E**quilibrium **C**onstraints |
| **PDPTW** | **P**ickup and **D**elivery **P**roblem with **T**ime **W**indow |
| **PI** | **P**erformance **I**ndex |
| **RC** | **R**esources-**C**ustomers |
| **RQB** | **R**elative **Q**ueue **B**alance |
| **SFM** | **S**tore-and-**F**orward **M**odel |
| **SO** | **S**ocial **O**ptimum |
| **STEN** | **S**pace-**T**ime **E**xpanded **N**etwork |
| **SUE** | **S**tochastic **U**ser **E**quilibrium |
| **TC** | **T**raffic **C**ontrol |
| **TTS** | **T**otal **T**ime **S**pent |
| **UE** | **U**ser **E**quilibrium |
| **VCG** | **V**ickrey-**C**larke-**G**roves |
| **WBB** | **W**eakly **B**udget **B**alanced |
| **ZDD** | **Z**ero-suppressed **B**inary **D**ecision **D**iagram |

# List of Symbols

| | | |
|---|---|---|
| $a_i$ | action of agent (or driver) $i$ | |
| $a_{-i}$ | actions of all agents (or drivers) except $i$ | |
| $\boldsymbol{a}$ | actions of all agents (or drivers) | |
| $A_i$ | set of actions for agent (or driver) $i$ | |
| $b_n$ | active time-window for node $n$ | |
| $\boldsymbol{c}$ | executed traffic control | |
| $D_i$ | cumulative departures from link $i$ | *veh.* |
| $E$ | set of links in the traffic network | |
| $\boldsymbol{E}[\cdot]$ | expectation function | |
| $f_i^s$ | saturated flow rate of link $i$ | *veh./s* |
| $g_i$ | fraction of the right of priority given to link $i$ | |
| $h^j(\cdot)$ | function of the $j$th constraint | |
| $I$ | set of agents (or drivers) | |
| $\bar{I}$ | number of agents (or drivers) | |
| $J_i$ | cumulative arrivals at link $i$ | *veh.* |
| $l_i$ | series of actions of agent $i$ | |
| $L_i$ | set of executable series of actions of agent $i$ | |
| $N$ | set of nodes in the traffic network | |
| $\bar{N}$ | number of nodes in the traffic network | |
| $n_i$ | number of vehicles in link $i$ | *veh.* |
| $n_i^{max}$ | maximum number of vehicles in link $i$ | *veh.* |
| $p_{ij}$ | ratio of the arrival at link $j$ from link $i$ to all departures from link $i$ | |
| $\boldsymbol{P}$ | transition matrix between links | |
| $q_i$ | demand at link $i$ | *veh./s* |
| $R_i(\cdot)$ | reward function of agent $i$ | |
| $S_i$ | set of states of agent $i$ | |
| $s_{i,t}$ | state of agent $i$ at time $t$ | |
| $\boldsymbol{s}_t$ | state of the traffic network at time $t$ | |
| $T_i$ | active time window of agent $i$ | |
| $\hat{T}_i$ | reported active time window of agent $i$ | |
| $t_i^B$ | beginning of the active time window of agent $i$ | |
| $\hat{t}_i^B$ | reported beginning of the active time window of agent $i$ | |
| $t_i^E$ | end of the active time window of agent $i$ | |
| $\hat{t}_i^E$ | reported end of the active time window of agent $i$ | |
| $V(\cdot)$ | value function | |
| $v_i^n$ | reward that agent $i$ receives by staying at node $n$ | |
| $\hat{v}_i^n$ | reported reward that agent $i$ receives by staying at node $n$ | |
| $w_i$ | velocity of the forward wave in link $i$ | *m/s* |
| $w_i'$ | velocity of the backward wave in link $i$ | *m/s* |
| $x_i$ | proposed state variable at link $i$ (Chapter 4) | *veh.* |
| $y_i$ | proposed state variable at link $i$ (Chapter 4) | *veh.* |

| | | |
|---|---|---|
| $U_i(\cdot)$ | utility function of agent (or driver) $i$ | |
| $DU_i(\cdot)$ | discounted utility function of agent (or driver) $i$ | |
| $SW(\cdot)$ | social welfare function | |
| $DSW(\cdot)$ | discounted social welfare function | |
| $\mathcal{A}$ | set of executable joint actions by all agents (or drivers) | |
| $\mathcal{C}$ | set of feasible traffic control | |
| $\tilde{\mathcal{D}}$ | stochastically predicted demand | |
| $\mathcal{G}_i$ | activity-state network of agent $i$ | |
| $\mathcal{G}$ | activity-state network of all agents | |
| $\mathcal{N}(\cdot)$ | normal distribution | |
| $\mathcal{R}$ | set of links including in a closed-loop | |
| $\boldsymbol{\mathcal{R}}$ | set of closed-loops in the traffic network | |
| $\mathcal{Z}$ | enumerated plans of all agents' actions | |
| $\alpha_i$ | length of link $i$ | $m$ |
| $\beta$ | time discount rate | |
| $\zeta_i(t)$ | ratio of departure flow inside the loop | |
| $\eta_i(t)$ | ratio of arrival flow inside the loop | |
| $\theta_i$ | type of agent $i$ | |
| $\theta_{-i}$ | type of all agents except $i$ | |
| $\hat{\theta}_i$ | reported type of agent $i$ | |
| $\lambda_i$ | arrival rate at link $i$ | $veh./s$ |
| $\lambda_i^{\langle j \rangle}$ | arrival rate at link $i$ from link $j$ | $veh./s$ |
| $\mu_i$ | departure rate from link $i$ | $veh./s$ |
| $\mu_i^{\langle j \rangle}$ | departure rate from link $i$ to link $j$ | $veh./s$ |
| $\boldsymbol{\pi}$ | decided joint action of all agents by the mechanism | |
| $\rho_i^{max}$ | saturated space density of link $i$ | |
| $\tau_i$ | transit time of the forward wave in link $i$ | $s$ |
| $\tau_i'$ | transit time of the backward wave in link $i$ | $s$ |

# Chapter 1

# Introduction

Advanced information technologies have recently led to the great change in the way of traffic services. In the past several decades, private owned and manually driven automobiles played a key role of traffic situations in many areas in the world. A great deal of effort has been devoted to developing better traffic control systems that control such automobiles, e.g., by traffic signals. On the contrary, two keywords, related to the automobile, that have attracted great attention recently are "automation" and "sharing." Automated vehicles can communicate to other vehicles and/or road infrastructures by themselves and thus the way of traffic control is greatly changing. The sharing system of automobiles relief the fixed pattern of usage of traffic resources by users. Users do not have to select the same traffic mode in both of outward and backward trip. This increases the importance of traffic services that treat multi-modal transportation resources, rather than traffic controls that only control the road traffics. Indeed, the concept of such mobility services, that is so-called "Mobility as a Service (MaaS)" has recently been emerging and growing. However, there exists no small example in which the newly introduced services end up in failure because of the reaction of the self-interested users to the policy. It implies that the behavior of users of systems should be well-considered in designing such systems.

In this chapter, we introduce the background of our study, present the scope of the thesis, and show the outline of the thesis.

## 1.1 Background

Automated driving technology (Burns, 2013) has received considerable attention from industry and academia as a decisive factor in solving the various problems of a motorized society, and is expected to have wide-ranging effects such as reducing traffic accidents, eliminating traffic jams, and improving the environment. However, pessimistic scenarios also exist. For instance, in the case of unexpected accidents or disasters, many automated vehicles (AVs) making emergency stops may decrease the road network capacity and cause traffic congestion. Additionally, in a society where automated driving systems have become common, persons who could not previously drive automobiles, such as children, the elderly, or those without drivers' licenses, will be able to travel freely, so it is possible that traffic will dramatically increase (Harper et al., 2016). Therefore, especially in cities, the construction of traffic control systems adapted to the presence of AVs is essential.

Fig. 1.1 shows examples of decision-making by an automobile driver. Decision-making related to driving is mainly divided into pre-trip and en-route decision-making. Pre-trip decision-making includes deciding whether to make the trip ("Trip Execution Decision") and deciding the destination of the trip ("Destination Choice"),

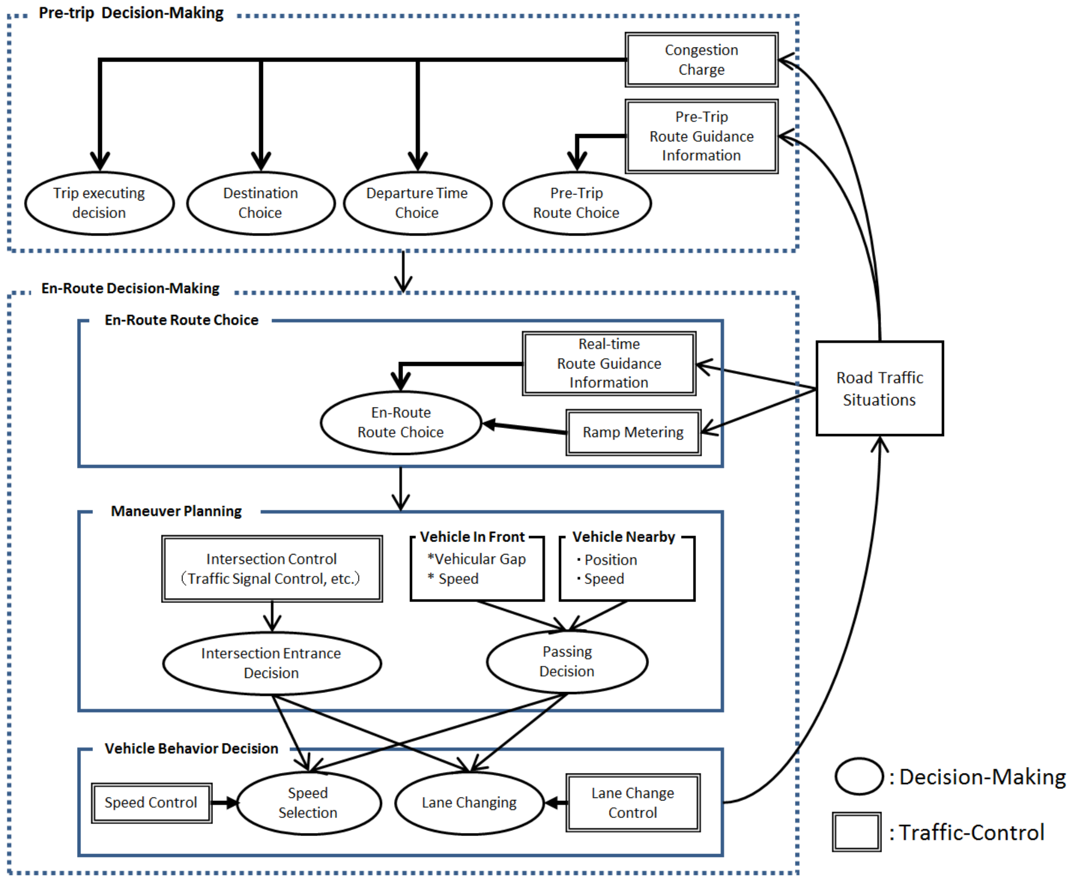



FIGURE 1.1: Decision-Making and Control Methods in Automobile Driving Behavior

as well as "Departure Time Choice" and "Pre-Trip Route Choice." En-route decision-making includes route changing based on information received during the trip ("En-Route Route Choice"), as well as Maneuver Planning, which includes items such as "Intersection Entrance Decision" and "Passing Decision," and vehicle behavior decision, which includes items such as "Speed Selection" and "Lane Changing." In a traffic system that includes AVs, some of the decision-making authority previously held by drivers is transferred to the system. For example, with current ACC (Adaptive Cruise Control) systems, the decision-making authority of "Speed Selection" is transferred to the system. Additionally, with the AVs under current development, it is assumed that decision-making authority related to all aspects of "Vehicle Behavior Decision," including "Speed Selection" and "Lane Changing," could be transferred to the system. Finally, in the future, decision-making related to "Route Choice" may also be transferred to an automated driving system. When a society implements automated driving, it is necessary to define which parts of the decision-making functions previously performed by drivers will become the responsibility of the automated driving system.

A traffic control system is a system that participates in the driver's decision-making to improve traffic conditions. Fig. 1.1 shows specific examples of traffic control methods related to decision-making. Existing traffic control methods include, for example, traffic control involved in pre-trip decision-making (such as congestion fees and presentation of pre-trip route guidance information), en-route route choice (real-time route guidance information and ramp metering), vehicle movement planning (intersection control), and vehicle behavior (speed control and lane change control). Among these traffic control methods, intersection control using traffic signals has traditionally played a large role in cities. At an intersection without traffic signals, the driver has all the decision-making authority; the driver must determine what the surrounding conditions are, and the intersection is crossed with caution. Conversely, at an intersection controlled by a traffic signal, the decision-making authority for "the decision to enter the intersection" is transferred to the system. The driver entrusts the decision-making to the signal, proceeding into the intersection on a green light, or stopping on a red light. The control system controls the traffic flow in the city by using the traffic signal parameters of "signal cycle," "offset," and "split" as control variables. Now, suppose that an intersection is controlled by an automated driving system. In this case, the driver transfers not only "the decision to enter the intersection," but also the "vehicle behavior decision" to the system. The system controls traffic flow in the city by directly using the speeds and locations of all the vehicles as control variables. With such control, the executable space of optimization problems is obviously larger than that of the traffic signal control, and better traffic conditions should therefore be realizable. Thus, there have been many reports regarding intersection control for automated driving systems (Chen and Cheng, 2010; Dresner and Stone, 2008).

However, there exist three problems when a traffic control method adapted to a single intersection is extended to the traffic control for an entire city. First, in a city with a high-density road network, describing traffic phenomena is extremely complicated. In such a complicated road network, the traffic conditions change dynamically, and vehicle queues lengthen with time. If the vehicle queues reach upstream intersections, the effect extends further to multiple upstream links, and chaotic traffic conditions occur. This phenomenon may occur habitually, because the demand exceeds the supply capacity of the road network, or it may unexpectedly occur owing to a major event, traffic accident, or disaster. In this paper, we call a road network beset with such congestion an "oversaturated network". Oversaturated networks are characterized by the "unsteady traffic flow" and "chaotic growth of vehicle queues" described above, so a traffic control system which can appropriately manage such situations is required.

The second problem is the interaction between traffic control and drivers' decision-making. As already mentioned, decision-making related to driving includes many activities, and it is difficult for a system to control all decision-making. For example, even if an automated driving system is optimized for "vehicle behavior decision," "route choice," and "departure time selection," it may still be the driver who chooses the "destination" and determines whether the trip is necessary in the first place. As shown in Fig. 1.1, the actual congestion of the roads reflects the results of decision-making by drivers. In general, because traffic control uses the (current or past) road congestion status as input information, a traffic control system that can appropriately process such information becomes a complicated system that includes a driver's decision-making in a feedback loop.

Even if the vehicle population consisted solely of AVs, similar problems would occur. Imagine two types of decision-making policies for an AV: a "system-optimized type" and a "user-optimized type." In the automated driving society of the future that many people vaguely imagine, the automobiles that travel around the city will be vehicles of the "system-optimized type" that are under central control, always providing smooth traffic flow. However, the current AVs are essentially "user-optimized" vehicles, performing decision-making that will increase the satisfaction of the user as much as possible within the perceived environment and traffic control framework. Such user-optimized decision-making processes may include, for example, shortest-time route choice and speed control that maintains a pleasant environment within the vehicle. Therefore, traffic control systems will include the behavior of "user-optimized" vehicles in a feedback loop, which will present the same problems as decision-making by drivers.

The third problem is the amount of computation. In a complicated road network with many vehicles, it is difficult to achieve traffic flow optimization control in which the behaviors of all the vehicles are used as parameters. Adding the two problems already mentioned, the amount of computation for a traffic control algorithm would be extremely large.

In this thesis, we show the direction of traffic controls and services with cities in which AVs are broadly introduced. We show that well-structured traffic controls/services are desired to achieve the ideal traffic situation in the coming age. Namely, the traffic controls by traffic administrators that aims to maximize social welfare and the mobility services by operators that aims to maximize the utility of their customers should be coordinated by using a well-controllable AVs. The algorithms related to control AVs are well-studied. With the penetration of AVs, the decision-making of driving such as acceleration and deceleration will be transferred from drivers to automated systems and thus traffic situations will become more and more controllable. However, the traffic controls/services considering higher-level decision-making of people such as route-choice or traffic-mode choice and so on, have not been well-studied. To address these problems, we specifically show the algorithms for traffic controls and services that appropriately considers users' behavioral choice in the traffic network.

## 1.2 Scope of the thesis

In this thesis, we focus on a traffic controls/services with limited traffic resources. One instance of such cases that we consider in this thesis is the traffic control in a large city with the densely distributed road network. In such a city, the limited road traffic capacities cause to the severe traffic conditions, such as heavy traffic jams and tragic traffic accidents. To avoid such a situation, a strand of works related to the traffic control of large cities has been studied. However, there are still many cities that are suffered by severe traffic jams in the city center. Especially, it is well-known that once the road-network falls into the over-saturated states, namely the number of vehicles located on the road networks excesses a specific level, the traffic situations become chaotic and difficult to control. A traffic control algorithm that can avoid this chaotic situation is needed. Another instance that we consider in this thesis is the ride share service in suburban areas. Unlike urban areas with a large number of users and drivers, the myopic and trip-based matching algorithms fail in such areas. To increase the sanctification of users with limited traffic resources, the service operator should focus more on the heterogeneity of users. The space–time constraint of each user plays an important role in the service operation.

To tackle these situations, we have to consider the behavior of users dynamically reacting to dynamic environments. In this thesis, we thus present dynamic capacity control of traffic resources focusing on the behavior of users that can be implemented to traffic controls/services.

## 1.3 Contributions of the thesis

To address the problems that are described in the previous part, we make the following contributions in this thesis.

- We organize a strand of works related to the traffic control systems considering the behavior of users reacted to the control, traffic control systems intended to apply to the oversaturated traffic situations, and traffic control systems that can treat advanced automated vehicles. We also organize the works related to activity-based travel analysis in which the travel demands are regarded as the derived demand of activities. Based on that, we present the way the traffic controls/services should be in a coming age.
- We present a novel traffic signal control algorithm that can be used in an oversaturated road network with limited capacity. We then evaluate some algorithms including our proposed one, considering the drivers react to the control and show the importance of the consideration of such behavior of drivers.
- We propose a novel traffic service algorithm that can appropriately allocate limited traffic resources to heterogeneous users that can be expressed as a utility maximizer and have individual private preferences and constraints. The proposed search-based algorithm can compute the efficient utilization of traffic resources considering the activity-based user behavior model.
- We propose dynamic pricing mechanisms for mobility service. While the proposed mechanisms are based on auction theory, they consider the specific features in the domain of mobility services, that is, the strong space–time constraints and dynamically changing environments.

## 1.4 Outline of the thesis

This thesis consists of following chapters.

In Chapter 2, we review the related work. We review conventional traffic control schemes that focus on the chaotic traffic situations in saturated-networks in large cities, and advanced traffic control schemes that directly control the motion of automated vehicles. We first show a strand of works that focus on the traffic control systems considering the behaviors of users. A traditional combined traffic control and traffic assignment problems are included in this part. We present dynamic interactions between the traffic control and users, focusing on the timing of traffic control and the reaction of users to the control. We formulate it in a game-theoretic form, considering that the users are self-interested utility-maximizer. Then we show the state-of-the-art of activity-based utility model of traffic users. In contrast with the traditional trip-based utility model by which the traffic controls are formulated to minimize travel costs, we can treat traffic control that aims to maximize utilities of users by considering activity-based utility model.

In Chapter 3, we present dynamic traffic signal controls in unsteady and oversaturated road networks, focusing on closed-loop structures in road networks and vehicle queue advancement in these loops. We propose a novel traffic control algorithm called Z-control, which prevents gridlock and minimizes the total time spent based on a model predictive scheme. Moreover, we evaluated some algorithms including our proposed Z-control, considering two extreme scenarios, “non-reactive” scenario where drivers do not react to traffic conditions in the network and "reactive" scenario where drivers are assumed to have perfect information about the traffic conditions in the network and react to it.

In Chapter 4, we present dynamic matching algorithms for the ridesharing service in suburb areas. Under the sparse ridesharing market where the common myopic and trip-based matching algorithms fail, we propose a mechanism that satisfies both users' space and time constraints and the capacity constraints of traffic resources at any time. We also propose a solution algorithm based on a graph-algorithms to solve this problem.

In Chapter 5, we present pricing algorithms for the integrated mobility services that treat not only automobiles but also various traffic mode in cities, as are often mentioned to MaaS (mobility as a service). We formalize *Activity-chain auction* that aims to achieve the socially optimal allocation of time-dependent traffic resources considering the self-interested behavior of users with private information. Especially, we focus on the situations where mechanism cannot know the type of users until they report their preferences and constraints to the booking system and propose a sequential mechanism that maximizes the discounted social welfare under the strategic behaviors of selfish users.

Finally, in Chapter 6, we conclude this series of studies and present the future works.

# Chapter 2

# Literature Review

In this chapter, we review the related work. We first organize the related works that focus on the traffic control systems considering the behaviors of users. We formalize the dynamic interaction between traffic controls and users in a game-theoretic form, assuming that the users are self-interested utility-maximizer. We then show the state-of-the-art of activity-based utility model of traffic users. In contrast with the traditional trip-based utility model by which the traffic controls are formulated to minimize travel costs, we can treat traffic control that aims to maximize utilities of users by considering activity-based utility model.

## 2.1 Traffic Control Methods for Oversaturated Networks

There have been many studies regarding traffic control methods for oversaturated networks. This section focuses on the relationship between traffic control and driver decision-making to summarize previous research, and then presents a basic plan for introducing an automated driving system into a city.

### 2.1.1 Traffic Signal Control

Traffic signal control (Papageorgiou et al., 2007, 2003) is the most important and basic method for controlling traffic in the complicated road network of a city, and aims to achieve smooth traffic flow by transferring the decision-making authority related to entering an intersection from the driver to the system. SCOOT (Hunt et al., 1982, 1981) present an example of typical traffic control, where the objective function is represented as a performance index (PI) that comprises items such as total travel time and the number of signal stops. The control variables are "cycle length," "split," and "offset," and items such as the law of conservation of traffic volume and the shortest green-light time are used in the constraint condition formulas. As shown in the top portion of Table 2.1, many traffic signal control algorithms have been implemented, but these are generally algorithms that assume the road network is not saturated, and are therefore unsuitable for the control of oversaturated networks.

As shown in the bottom portion of Table 2.1, there have also been various studies on signal processing for oversaturated networks. This research is mainly divided into a problem of minimizing total travel time, or a queue management problem. Next, we summarize each of these.

TABLE 2.1: Classification of Traffic Signal Control Algorithms

| | Network | Route Change | Traffic Flow Model | Control Variables | Optimization Target Variables | Solution Methods, Other Characteristics |
|---|---|---|---|---|---|---|
| Algorithms that Have Been Used | | | | | | |
| Fixed-time | | | | | | |
| TRANSYT (ROBERTSON, 1997) | Urban area | Yes | CFP | c,s,o | PI | |
| Traffic-responsive | | | | | | |
| Centralized Control | | | | | | |
| SCOOT (Hunt et al., 1982, 1981) | Urban area | Yes | CFP | c,s,o | PI | Detectors on the upstream ends of links |
| SCATS (Lowrie, 1982) | Urban area | Yes | - | c,s,o | TT, etc. | Detectors on the downstream ends of links |
| RHODES (Head et al., 1992) | Urban area | Yes | MM | c,s,o | TT, etc. | Detectors on the upstream ends of links |
| Distributed Control | | | | | | |
| UTOPIA (Donati et al., 1984) | Urban area | Yes | MM | c,s,o | TT | Detectors on the upstream ends of links |
| Research Targeting Oversaturated Conditions | | | | | | |
| Formalized as a Total Travel Time Minimization Problem | | | | | | |
| Formalized as a Store-and-Forward Model | | | | | | |
| Gazis (1964) | 2intersections | No | SFM | s | TT | Graphical solution method |
| D'ans and Gazis (1976) | 2 intersections | No | SFM | s | TT | LP |
| D'ans and Gazis (1976) | General | No | SFM | s | TT | Only formalized, no solution method |
| Michalopoulos and Stephanopoulos (1977) | 2 intersections | Yes | SFM | s | TT | Formalized as optimization control theory |
| Chang and Lin (2000) | 1 intersection | Yes | SFM | s | TT | successive optimization |
| Liu et al. (2008) | 1 intersection | Yes | SFM | s | TH | LP |
| Focus on the Detailed Dynamics of Traffic Flow | | | | | | |
| Lo (1999, 2001) | General | No | CT | c,s,o | TT | MIP |
| Lo et al. (2001) | General | Yes | CT | c,s,o | TT | GA |
| Formalized as a Queue Management Problem | | | | | | |
| Singh and Tamura (1974) | 3 intersections | Yes | SFM | s | QL | Formalized as a time-delay system |
| Rathi (1988) | General | Yes | SFM | s,o | PV | |
| Abu-Lebdeh and Benekohal (1997) | 2 intersections | No | SFM | s,o | WG | GA |
| Aboudolas et al. (2009, 2010) | General | | SFM | s | DQ | Model prediction control |

Traffic Flow Models: CFP: Cyclic Flow Profiles,MM: Micro-Macro Hybrid Model, SFM: Store-and-forward model, KW: Kinematic wave,CT: Cell transmission,VT: Variational theory
Control Variables: c: Cycle Length, s: Split, o: Offset
Optimization Target Variables: PI: Performance Index, TT: Travel time, TH; Throughput, QL: Vehicle queue length, WG: Wasted green time, DQ: Distribution of vehicle queue, PV: Prevention of interference with upstream intersections due to vehicle queues
Solution Methods: LP: linear program,MIP: mixed integer program GA: genetic algorithm

### Research that Formalizes Traffic Control as a Total Travel Time Minimization Problem

First, we present research examples that formalize traffic signal control as a total travel time minimization problem and attempt to derive a solution. Gazis (1964) proposed a store-and-forward model (SFM) that expresses the capacity of a link controlled by a traffic signal as the product of a saturation capacity and the signal green-time ratio (split ratio). Using this model, a graphical solution method is used to determine the signal split that minimizes the total travel time in a road network of two intersections without left or right turns. D'ans and Gazis (1976) formalized traffic control with vehicle queue length as the state variable in a similar two-intersection network without turns, and showed that the optimal split can be calculated as a linear programming (LP) problem. They also formalized traffic control with a total travel time minimization problem for general networks without left and right turns, but did not present a solution method. Michalopoulos and Stephanopoulos (1977) used optimal control theory (Kirk, 2012) for a two-intersection network with left and right turns, and presented the properties of solutions for optimal traffic control. They showed that if vehicle queue capacity constraints are introduced into multiple links, then the constraint condition for the total travel time minimization problem contains a delay term for the state variable. Therefore, the existence of a solution is not guaranteed, and optimal control becomes extremely complicated. Chang and Lin (2000) used discrete-time SFM for one oversaturated intersection and calculated an appropriate split by successive optimization. Chang and Sun (2004) applied this method to a control method that is envisioned to be applied to TRANSYT-7F (Wallace et al., 1984), and its effectiveness was evaluated by simulation. Liu et al. (2008) formalized a throughput maximization problem and an equivalent total travel time minimization problem as an LP problem, using one oversaturated intersection.

This line of research uses SFM to express traffic flow, and aims to find a solution for the total travel time minimization problem. In the case of one intersection, the goal has been achieved, but for multiple intersections, the optimization problem becomes complicated, requiring a delay term in the state variable, such that finding the optimal solution becomes difficult. In addition, SFM cannot express signal offset or cycle length, and thus for offset and cycle length control, a different algorithm is needed.

Lo (1999, 2001) modeled the dynamics of traffic flow using a cell-transmission model (Daganzo, 1994, 1995) for a general network without left and right turns, and formalized signal control that minimizes the total travel time as a mixed integer programming (MIP) problem. Lo et al. (2001) expanded this to networks with left and right turns, and showed a method for deriving appropriate signal parameters using a genetic algorithm (GA). In their research, the traffic flow dynamics are expressed in detail, but the required amount of computation makes it difficult to apply to a large-scale network.

### Research that Formalizes Traffic Control as a Queue Management Problem

As shown so far, it is generally difficult to find a solution to the total travel time minimization problem for an oversaturated network. Therefore, there has been research to treat traffic signal control as a queue management (Quinn, 1992) problem, specifying an objective function other than total travel time minimization.

Singh and Tamura (1974) formalized traffic control in an oversaturated network as a time-delay system (Richard, 2003) optimization problem that attempted to minimize vehicle queue length. They expressed the condition of an entire discrete-time network as a linear system with a delay term and used numerical calculation to determine the dynamically optimized control method for a network with three intersections. Because vehicle queue extension that interferes with the traffic flow of an upstream intersection (queue spillback) is the most significant cause of traffic congestion, Rathi (1988) proposed controls that reduced the frequency and effect of queue spillback. Specifically, by using a traffic flow model that introduced an adjustable parameter $\alpha$ to express the distribution of traffic during each cycle, on top of an SFM base, splits were calculated to minimize the effect of queue spillback. Optimal offset was separately considered by using kinematic wave (KW) logic, and was formalized for a network of two intersections. However, deriving the same type of solution for a general network would be difficult, thus this only qualitatively demonstrated the idea that, as the probability of occurrence of queue spillback increases, a "reverse" offset setting is desirable. In order to eliminate wasted green signal time (de facto red) caused by queue spillback, Abu-Lebdeh and Benekohal (1997) proposed a traffic signal control model that adaptively changed splits and offsets corresponding to vehicle queue length and traffic demand, and calculated a solution by using a GA. In order to suppress the risk of queue spillback, Aboudolas et al. (2009, 2010) proposed traffic control in which vehicle queue balance within a network, i.e., relative queue balance (RQB), was used as an evaluation reference, and used a model prediction control framework to predict future traffic conditions.

These methods consider unsteady traffic flow and the behavior of vehicle queues, and appear to be highly compatible with oversaturated networks. However, depending on the traffic conditions and the network, control that targets queue management itself may not be able to achieve appropriate results. Acceptable results may also not be obtained when higher-level decision-making such as destination choice and route choice are taken into account. Section 2.2 considers these points.

### 2.1.2 Ramp Metering

Ramp metering is a general traffic control method associated with traffic signal control. This is a method of proactively preventing the occurrence of a traffic jam on an expressway by controlling the amount of traffic that enters the expressway from entrance ramps. It aims to control expressway density and facilitate smooth traffic flow by transferring the driver's decision-making authority related to route choice to the system. Wattleworth (1965) divided an expressway into several intervals and expressed the amount of traffic $q_j$ in interval $j \in J$ by the formula

$$q_j = \sum_{i \in I} \alpha_{ij} r_j \tag{2.1}$$

where $r_j$ is the traffic volume that enters from entrance ramp $i \in I$ and $\alpha_{ij} \in [0,1]$ is the fraction of vehicles that pass through interval $j$ of the expressway considering all vehicles that entered from entrance ramp $i$. If $\alpha_{ij} \in [0,1]$ is known, such as from past statistical data, then by establishing the constraint condition

$$\forall j : q_j \leq q_{cap,j} \tag{2.2}$$

which involves the traffic capacity $q_{cap,j}$ for interval $j$, it is possible to perform entrance control that does not allow a traffic jam to occur on the expressway. The target variable for the control may be, for example, maximizing the amount of entering traffic, or maximizing the total travel distance, and such optimization problems can be formalized as LP problems.

This method hypothesizes steady traffic flow, which rules out application to conditions in which traffic quantity changes. With regard to this problem, Papageorgiou (1980) formalized the problem by considering the time for a vehicle that entered from entrance ramp $i$ to reach interval $j$ on the expressway. However, this research assumes that the demand up to a future point in time is known, and thus it cannot be applied to unexpected phenomena such as accidents.

Papageorgiou et al. (1991, 1997) proposed a feedback control named "ALINEA" that maintained the traffic density in downstream intervals at an optimal value for one entrance ramp, and demonstrated its effectiveness. Additionally, Benmohamed and Meerkov (1994) proposed a ramp metering strategy that considered fairness across the entrance ramps within a network, using a similar local feedback control formula.

Ramp metering aims to prevent an oversaturated condition from arising on a priority route. There is also research that considers the relationship between traffic control performed by a traffic manager and a driver's route choice behavior. However, while research is progressing on relatively simple networks, such as city expressways, there has not been progress with respect to complicated networks, such as the general roads of a city center.

### 2.1.3 Congestion Fees

In all the controls discussed so far, some type of decision-making during the trip was transferred to the system. By contrast, a congestion fee is an indirect traffic control that affects pre-trip decision-making, such as destination choice and trip execution decision. The purpose of this method is to prevent the road network of a city center from falling into an oversaturated condition by suppressing the amount of traffic that enters a certain area of the city. For example, Daganzo (2007) focused on the relationship

$$g = G(\rho) \tag{2.3}$$

between the traffic density $\rho$ of a city center area and the amount of traffic, g, flowing out of that area, and presented a method of controlling traffic density in order to maximize the amount of traffic flowing out. Formula 2.3 is called a macroscopic fundamental diagram (MFD) (Geroliminis and Daganzo, 2008) of the road network, and it has been shown both theoretically and experimentally that, if the traffic within the target area is homogeneous and in steady state, then the above formula becomes a convex-upward function that has a maximum value of $g_{max} = G(\rho_0)$ for some $\rho = \rho_0$.

A congestion fee is an effective traffic control method in cities that experience severe traffic jams, and demand suppression and congestion relief have been observed in many cities, such as London and Singapore. Additionally, these cities implement measures to adjust demand, such as dynamic pricing measures that change road travel fees according to conditions, travel prohibition measures on certain days based on vehicle license plate numbers, and requests for self-imposed restrictions on travel during special events. When demand adjustment is successful, the network maintains a state that is not oversaturated, and smooth traffic flow can be maintained. However, this type of method is not a control aimed at oversaturated networks, and if a network falls into an oversaturated condition, for example, due to an unexpected event, accident, or disaster, more aggressive traffic control is necessary to quickly eliminate the congestion. Furthermore, the methods to appropriately determine the area to be controlled and the specific methods of implementing entrance control are unclear.

### 2.1.4 Direct Control of Vehicle Behavior

Direct control of vehicle behavior refers to methods where a system directly controls decision-making related to vehicle behavior (such as acceleration, braking, and lane changing), and specifically includes items such as "speed control" and "lane change control." This has already been partially implemented in systems such as ACC and Lane Keeping Assist (LKA), but one can expect its importance to increase greatly with the introduction of automated driving.

#### Speed Control

It is thought that speed control will greatly affect other traffic control methods by transferring decision-making authority concerning speed selection from a human driver, whose probabilistic behavior is highly uncertain, to a system that can provide deterministic behavior (Varaiya, 1993). The following Gazis–Herman–Rothery model (Gazis et al., 1959) is a representative example of a vehicle tracking model (Brackstone and McDonald, 1999).

$$a_n(t) = cv_n^m(t)\frac{\Delta v(t-T)}{\Delta x^l(t-T)} \tag{2.4}$$

Here, $a_n(t)$ is the acceleration of vehicle $n$ at time $t$, and this equation shows its relationship to the speed $v_n(t)$ of vehicle $n$, and to the relative speed $\Delta v(t-T)$ and relative position $\Delta x(t-T)$ (inter-vehicle distance), relative to vehicle $n-1$ located in front at time $t-T$. $T$ is the driver's response time. It is reasonable to consider that an automated driving system can not only shorten the response time $T$, but also suppress variations in $T$. For example, Shladover et al. (1991), hypothesizing an expressway populated only by AVs, presented an algorithm for controlling groups of vehicles that can maintain a shorter inter-vehicle distance than when humans are driving. This method increases traffic density and traffic capacity using automated speed control.

By contrast, for cities, deceleration control and variable speed control have been proposed to avoid excessive congestion by maintaining the traffic density at an appropriate level. For example, Chien et al. (1997) hypothesized the Automated Highway System, in which a traffic manager can fully control the speed of an automobile, and presented an algorithm for maintaining the density of each link on a network at a desired value, whereas Hegyi et al. (2005) presented an algorithm that uses variable speed limit control to maintain traffic density at an appropriate level, thereby improving total travel time. These examples illustrate that when considering complicated city networks, it is not necessary to increase traffic density and traffic capacity, but rather control them to achieve appropriate levels.

#### Lane Change Control

It is thought that lane change control will effectively use road space and increase traffic network capacity by transferring the decision-making authority from the driver to the system. For example, Hall and Lotspeich (1996) demonstrated a lane use assignment method for optimizing traffic capacity when the traffic demand was known, and Ramaswamy et al. (1997) formalized lane assignment as a travel time minimization problem. Roncoli et al. (2015a,b)presented an optimization algorithm that segmented each lane into fixed distances and specified a desired traffic density for each.

**Integrated Vehicle Behavior Control**

There have also been many proposals for integrated vehicle behavior control systems that include both the speed control and lane change control methods discussed above. The purpose of this research is both to assign routes to vehicles and to determine vehicle movement planning (maneuvering) and vehicle behaviors for traversing those routes. That is, the system holds the decision-making authority not only for vehicle behavior decisions, but also for vehicle movement planning. Katrakazas et al. (2015) provide a detailed review of such research.

Whereas all of the research discussed so far has focused on individual vehicles, other studies have focused on traffic control that aims to eliminate congestion over an entire road network by using methods of information transmission, such as vehicle-to-vehicle communication and road-to-vehicle communication (Li et al., 2014). There are many examples of research into traffic control methods for one intersection, such as vehicle group control (for example, Jiang et al. (2006))), road-to-vehicle cooperation control (for example, Li and Wang (2006)), and vehicle-to-vehicle cooperation control (for example, Dresner and Stone (2008)). For trunk roads with continuous signals, there has been research (for example, Asadi and Vahidi (2011)) into minimizing signal stops by transmitting signal display timing to vehicles.

### 2.1.5 Summary of Traffic Control Methods for Oversaturated Networks

This section has discussed “traffic signal control", “ramp metering", “congestion fees" and “direct control of vehicle behavior" as traffic control methods for oversaturated networks, describing the decision-making relationship between the traffic control system and the driver. Each type of control aims at improving traffic conditions by transferring part of the decision-making related to the driving behaviors shown in Fig. 1.1 from the driver to the system. There has been much research into methods that increase traffic density and traffic capacity by introducing automated driving technology. For example, systems have been presented that optimize decision-making related to “maneuver planning" and “vehicle behavior decision" with the purpose of “maximizing traffic capacity." However, as has been indicated in this section, in city centers with complicated road networks, local maximization of traffic capacity does not necessarily lead to improvement of traffic conditions for the city as a whole. Traffic control must cover the entire network, such as by using “speed reduction control" to maintain appropriate traffic densities, and traffic signal control to appropriately control vehicle queues. Currently, the traffic control systems of cities mainly use decision-making related to “route choice" and “intersection control," and the driver owns the decision-making related to “vehicle behavior decision." However, in a traffic system that controls AVs, it is possible to implement traffic control systems that also hold decision-making authority related to “vehicle behavior decision." By exploiting this advantage, traffic control methods that react to dynamically changing traffic situations and quickly lead to appropriate traffic density levels in the complicated road networks of cities are possible.

## 2.2 Traffic Control Considering the Behaviors of User-Optimized Agents

There have been various reports on traffic control that treat the driver as a "user-optimized agent." This section will review this research, and present a basic idea for traffic control that includes user-optimized agents in a feedback loop.

### 2.2.1 Traffic control and traffic assignment

The interaction between traffic control and the decision-making of drivers will now be formalized to describe the form that traffic control should take.

We consider a set of drivers $I = \{1, \dots, \bar{I}\}$ in the traffic society to be controlled, and use $\boldsymbol{a} = \{a_1, \dots a_{\bar{I}}\}$ to denote their actions. Let $\boldsymbol{c}$ be the traffic control that is actually executed. The condition of the society is determined by the traffic control and the drivers' action, and thus the achieved total social welfare can be expressed as a function $SW(\boldsymbol{c}, \boldsymbol{a})$. Using this expression, the types of traffic control shown earlier, "direct control of vehicle behavior," "traffic signal control," and "ramp metering" can be expressed in the form of the traffic control $[TC]$ problem shown below.

$[TC]$

$$\max_{\boldsymbol{c} \in \mathcal{C}} SW(\boldsymbol{c}, \boldsymbol{a}) \tag{2.5}$$

$$s.t. \quad h^j(\boldsymbol{c}, \boldsymbol{a}) \leq 0, \; j = \{1, 2, \dots, k\}, \tag{2.6}$$

where $k$ denotes the number of constraints, $h^j(\cdot)$ denotes the function of $j$th constraint, and $\mathcal{C}$ denotes a set of feasible traffic control. This type of traffic control aims to maximize the value of the objective function $SW(\boldsymbol{c}, \boldsymbol{a})$ under multiple constraint conditions, such as traffic capacity constraints and the law of conservation of traffic volume, when the drivers' action $\boldsymbol{a}$ is steady and known. The optimal traffic control pattern $\boldsymbol{c}^*$ is obtained from the solution of the $[TC]$ problem. This type of problem does not consider the reactions of drivers to the traffic control that is executed.

On the other hand, "congestion fees" can be viewed as a traffic assignment problem in which the traffic control $\boldsymbol{c}$ is known. In this problem, the achieved traffic flow pattern is described by the Nash equilibrium condition in the form shown below.

$[UE]$ (Nash Equilibrium Condition )

$$\forall i, a_i \in A_i : U_i(\boldsymbol{c}, \boldsymbol{a}^*) \geq U_i(\boldsymbol{c}, a_i, a^*_{-i}) \tag{2.7}$$

Here, $A_i$ is a set of executable decision-making for driver $i$, $a_i \in A_i$ is the action of driver $i$ whereas $a_{-i}$ is the action of drivers other than $i$, and $U_i(\boldsymbol{c}, \boldsymbol{a})$ is the utility to user $i$ of $\boldsymbol{a}$ and $\boldsymbol{c}$. In formula (2.7) the Nash equilibrium condition $\boldsymbol{a}^*$ is the condition where no driver can increase its own utility by changing its own behavior. Apart from "congestion fees," another example of the user equilibrium $[UE]$ type of control is a "Route Guidance Information System (Papageorgiou et al., 2007, 2003)." This method is traffic control that affects the driver's decision-making related to route choice by providing information regarding the degree of congestion and the required time for each route.

In a steady network, the equilibrium condition that arises when the driver's behavior is deterministic is known as a Wardrop equilibrium, and can be expressed in the framework of "Congestion game (Rosenthal, 1973)." Or, in a steady network, the equilibrium condition that arises when the driver's behavior is probabilistic is called a probabilistic user equilibrium condition.

Examples of studies demonstrating route choice behavior with a probabilistic, discrete selection model include the Multi-Term Logit Model and the Nested Logit Model, which are based on the Generalized Extreme Value theory of McFadden (1978) and on the Mixed Logit Model proposed by McFadden (1989). On the other hand, Dial (1971) presented an algorithm that avoided the process of enumerating routes, and instead probabilistically distributed traffic volume over all paths that did not involve going back (efficient paths). It was later shown by Van Vliet (1981) that this model was equivalent to the Logit Model. Additionally, Akamatsu (1996) proposed a Logit-type probabilistic traffic assignment method in networks that have cyclic patterns, and Fosgerau et al. (2013) proposed the Recursive Logit Model, in which the ultimately obtained path is the result of successive forward-path selection for each node, and also proposed a traffic assignment method that did not list selection branches. In order to understand the type of society that can be achieved by the $[UE]$ type of traffic control, it is effective to use these methods.

### 2.2.2 Optimal control

Thus far, we have reviewed the $[TC]$ type of control and the $[UE]$ type of control. However, with the introduction of automated driving technology, the term "society-optimized" control has recently come into use. This concept can be formalized with the following type of society-optimized $[SO]$ problem.

$[SO]$

$$\max_{\boldsymbol{c}\in\mathcal{C},\boldsymbol{a}\in\mathcal{A}} SW(\boldsymbol{c},\boldsymbol{a}) \tag{2.8}$$

$$s.t. \quad h^j(\boldsymbol{c},\boldsymbol{a}) \leq 0,\ j = \{1,2,\ldots,k\}, \tag{2.9}$$

where $\mathcal{A}$ denotes a set of executable joint actions by all drivers.

This problem takes a form similar to the $[TC]$ problem, but the executable space of the optimization is different. In the $[TC]$ problem, the driver's action $\boldsymbol{a}$ is known and the optimal control pattern is found from within the executable space of controls $\mathcal{C}$. The $[SO]$ problem also considers changes in the driver's decision-making and seeks a combination of optimal traffic control $\boldsymbol{c}^*$ and drivers' action $\boldsymbol{a}^*$ within the ranges of the executable spaces $\mathcal{C}$ and $\mathcal{A}$. The executable space of the $[SO]$ problem includes the executable space of the $[TC]$ problem, and $[SO]$ will therefore always yield a condition that is equal to or better than that of $[TC]$; if one assumes that $\mathcal{A}$ includes all the action of all the players in the traffic society, then the condition that can be achieved with the $[SO]$ type of traffic control is a society-optimized condition.

As an example of a method that achieves this type of society-optimized condition by incorporating a centralized control in a future automated driving system, there is the game theoretic idea of "mechanism design." The Vickrey-Clarke-Groves (VCG) mechanism (Clarke, 1971; Groves, 1973; Vickrey, 1961), which is a basic method of this type, derives the decision-making that can achieve society-optimized conditions by assigning to each user a cost that corresponds to the externality that the user contributes to the society. Yang and Wang (2011), based on a similar idea, proposed a "tradable travel credits system " that achieves society-optimized conditions when "route choice" and "departure time selection" represent the driver's decision-making space. Additionally, assuming that the road network is always in an unsaturated state, Wada and Akamatsu (2013) proposed a transaction algorithm for implementing a similar system.

However, such approaches present two problems. First, there is the computational difficulty of applying these methods to the general networks in an actual city. Especially, if we apply it to an unsteady oversaturated network, it is necessary to solve the $[SO]$ problem dynamically. There have been many studies (Friesz et al., 1989; Merchant and Nemhauser, 1978) on the dynamically society-optimized control $[DSO]$ problem, which considers vehicle control and route choice as the driver's decision-making space. Lovell and Daganzo (2000) formalized DSO assignment by considering a FIFO (First-in First-out) constraint, and showed that an exact solution can be obtained for networks with a single origin or destination, or with a single bottleneck. But they also mentioned that model construction and its numerical analysis for a general network are difficult owing to a lack of convexity.

As for the second problem, the difficulty of allowing traffic control to control all of the driver's action remains. The action of a driver related to traffic behavior includes many activities. No matter how advanced the system, part of the decision-making of action $\boldsymbol{a}$ will remain with the driver, making it difficult to construct a system that implements an optimal condition across the entire action space $\mathcal{A}$.

TABLE 2.2: Traffic Control Problems with User Optimized Agents

| | Control Method | Target Variable | User Model | Solution Method | Obtained Solution |
|---|---|---|---|---|---|
| Allsop (1974) | SIG | Arbitrary | UE | Iterative Calculation | NC |
| Smith (1978, 1979) | SIG | OCAP | UE | Iterative Calculation | NC |
| Smith and Van Vuren (1993) | SIG | OCAP | UE | Iterative Calculation | NC |
| Sheffi and Powell (1983) | SIG | TT | UE | Gradient Method | ST |
| Fisk (1984) | SIG | TT | UE | Penalty Function Method | ST |
| Yang and Yagar (1995) | SIG | TT | UE | Sensitivity Analysis Method | ST |
| Ceylan and Bell (2004) | SIG | PI | SUE | GA and TRANSYT | ST |
| Sun et al. (2006) | SIG | TI | SUE | GA and CORSIM | ST |
| Braess's Paradox (Frank, 1981) | NET | TT | UE | - | - |
| Suwansirikul et al. (1987) | NET | TI | UE | Breakdown into Subproblems | ST |
| Yin (2000) | NET | TT | UE | GA | ST |
| Chiou (2005) | NET | TI | UE | Gradient Method, etc. | ST |
| Research Targeting Dynamic Networks | | | | | |
| Chen and Ben-Akiva (1998) | SIG | TT | SUO | - | ST |
| Gartner and Stamatiadis (1998) | SIG | TT | UO | - | ST |
| Ukkusuri et al. (2013) | SIG | TT | UE | Iterative Calculation | NC |
| Han et al. (2015) | SIG | TT | UE | Particle Swarm Optimization, etc. | ST |

Traffic Control Methods:
SIG:Traffic signal control, NET: Network design, RAMP:Ramp metering,
Control Objective Functions:
PI:Performance index calculated from total travel time or total stops, TT:Total travel time minimization,
TI:Total travel time improvement effect relative to total investment, OCAP:Outgoing traffic capacity maximization
Uses' Behavior Models:
UE:Nash equilibrium, SUE:Stochastic user equilibrium, SUO:Stochastic user-optimized
Solution Methods:
GA:Genetic algorithm, TRANSYT:Traffic signal control algorithm (Robertson, 1969; ROBERTSON, 1997),
CORSIM:A generic micro traffic flow simulator
Obtained Solutions:
ST: Stackelberg equilibrium solution, NC:Nash–Cournot equilibrium solution

### 2.2.3 Traffic control with a user equilibrium constraint

As shown in the previous section, it is difficult to implement society-optimized $[SO]$ control across a driver's entire action space. On the other hand, it is possible to implement conditions as close to $[SO]$ as possible by considering the driver's reaction to traffic control. Based on this idea, there are many reports on research into the "combined traffic control and assignment problem," which simultaneously considers the traffic control problem and the traffic assignment problem, and examples of this are shown in Table 2.2. Among the many studies into network design and traffic signal control, the first to mention this type of control was Allsop (1974). Allsop formalized the relationship between the user's route choice behavior, which changes day to day, and traffic signal control parameters as a bi-level problem. In detail, he introduced traffic signal control formalized by $[TC]$ as an upper-level problem and expressed users' reactions formalized by $[UE]$ as a lower-level problem, and obtained optimal traffic control parameters by iterated calculations of these upper and lower problems. In a similar manner, Smith (1978, 1979) focused on the convergence properties of the bi-level problem. He demonstrated that this problem does not generally converge if the objective of the traffic control is minimizing total travel time, or if Webster's method (Webster, 1958) is adapted as the control policy, but it generally converges if the objective is maximizing network-capacity. Smith and Van Vuren (1993) finally defined the conditions to be satisfied by the control objective function and by the link cost function that allow the iterative calculation to converge.

The iterative calculations performed in this line of research can be considered in the framework of "Cournot competition", and the equilibrium conditions that the solution $(\boldsymbol{c}^*, \boldsymbol{a}^*)$ should satisfy can be expressed as follows (Chen and Ben-Akiva, 1998):

[Nash–Cournot Equilibrium Conditions]

$$\forall \boldsymbol{c} \in \mathcal{C} : SW(\boldsymbol{c}^*, \boldsymbol{a}^*) \geq SW(\boldsymbol{c}, \boldsymbol{a}^*) \tag{2.10}$$

$$\forall i, a_i \in A_i : U_i(\boldsymbol{c}^*, \boldsymbol{a}^*) \geq U_i(\boldsymbol{c}^*, a_i, a^*_{-i}) \tag{2.11}$$

A solution $(\boldsymbol{c}^*, \boldsymbol{a}^*)$ that satisfies these conditions will be a fixed point that cannot be changed in either the upper-level $[TC]$ or lower-level $[UE]$ problem.

On the other hand, Fisk (1984) treated this as a problem in which a traffic manager incorporates the reaction of the driver to the traffic control system and then the optimal traffic control parameters are derived, as formalized by using the Stackelberg game shown below (Chen and Ben-Akiva, 1998).

[Stackelberg game]

$$\max_{\boldsymbol{c} \in \mathcal{C}} SW(\boldsymbol{c}, \boldsymbol{a}) \tag{2.12}$$

$$s.t. \quad h^j(\boldsymbol{c}, \boldsymbol{a}) \leq 0, \; j = \{1, 2, \ldots, k\} \tag{2.13}$$

$$\forall i, a_i \in \mathcal{A}_i : U_i(\boldsymbol{c}, \boldsymbol{a}) \geq U_i(\boldsymbol{c}, a_i, a_{-i}) \tag{2.14}$$

This is a mathematical programming problem with equilibrium constraints (MPEC), where combining the user equilibrium problem $[UE]$ with the constraint conditions for the optimal control problem $[TC]$, and the calculated solution $\boldsymbol{c}^*$, together with the drivers' action $\boldsymbol{a}^*$ at that time, is called the Stackelberg equilibrium solution. The Stackelberg equilibrium and the Nash-Cournot equilibrium use the same user equilibrium condition formulas (2.11 and 2.14), but in the Stackelberg equilibrium, the objective function is maximized by using formula (2.12). The following relationship therefore exists between the Stackelberg equilibrium solution $(\boldsymbol{c}^*_{ST}, \boldsymbol{a}^*_{ST})$ and the Nash–Cournot equilibrium solution $(\boldsymbol{c}^*_{NC}, \boldsymbol{a}^*_{NC})$:

$$SW(\boldsymbol{c}^*_{ST}, \boldsymbol{a}^*_{ST}) \geq SW(\boldsymbol{c}^*_{NC}, \boldsymbol{a}^*_{NC}) \tag{2.15}$$

In other words, the system can achieve the Stackelberg equilibrium condition, which is better than the Nash–Cournot equilibrium, by having a traffic manager first incorporate the reaction of the driver. The Stackelberg equilibrium solution is not a fixed point of the upper-level $[TC]$ problem; therefore, it cannot be found by iterative calculation of the upper-level and lower-level problems, and an algorithm for finding the solution is needed. The same type of problem has been considered not just for the traffic signal control problem, but also for network design problems and, as shown in Table 2.2, many heuristic algorithms have been proposed for finding the Stackelberg equilibrium solution. Some are summarized in the review of Mitsakis et al. (2011).

Considering the society with AVs, we may have to consider decision-making which is much higher than those shown in Fig. 1.1. For instance, Chen et al. (2017, 2016) formalized the relationship between the policy-maker which introduce exclusive AV lanes to the city and users considering replacing their conventional vehicles to AVs, as a Stackelberg game. As we can see from this work, many problems including higher decision-making can be treated by the framework discussed in this part.

### 2.2.4 Traffic Control Considering Dynamic Behaviors of User-Optimized Agents

In order to control an oversaturated network, the methods mentioned in Section 2.2.3 are extended to a dynamic framework that handles unsteady conditions. In this part, we consider the discrete time step $\{t_s, \ldots, t_e\}$, where $t_s$ and $t_e$ denotes the initial and the final stage of the control.

#### Dynamically User-Optimized Allocation and Dynamic User Equilibrium Allocation

First, we present a brief summary of dynamically user-optimized (DUO) allocation and dynamic user equilibrium (DUE) allocation. The time-varying traffic control $\boldsymbol{c}$ is known, and the problem is to find the driver's action $\boldsymbol{a} = \{a_1, \ldots, a_{\bar{I}}\}$, where the action of driver $i$ is denoted by $a_i = \{a_{i,t_s}, \ldots, a_{i,t_e}\}$. The set of executable strategies for driver $i$ at time $t$ is expressed as $A_{i,t}$, and the actually executed action is $a_{i,t} \in A_{i,t}$. The following formula expresses the network condition $s_t$ at time $t$, which depends on the traffic control and the decision-making for a time span $t' \leq t$.

$$s_t = \mathcal{S}(\boldsymbol{c}(t'), \boldsymbol{a}(t') | t' \leq t) \tag{2.16}$$

DUO allocation uses a strategy in which each user optimizes its own utility $U_i(a_{i,t}, s_t)$ based on the current network condition $s_t$. An example of a utility function $U_i(a_{i,t}, s_t)$ is the estimated travel time (under the current network condition $s_t$) from the current location to the destination. The strategy $a^*_{i,t}$ of driver $i$ at time $t$ is expressed as:

$[DUO]$

$$\forall i, t : a^*_{i,t} = \underset{a_{i,t} \in A_{i,t}}{\text{argmax}}\, U_i(a_{i,t}, s_t) \tag{2.17}$$

Because the network condition $s_t$ changes with time, each driver's strategy also changes, regardless of past decisions. That is, DUO allocation is a traffic allocation method that assumes "myopic" driver decision-making.

On the other hand, the DUE condition is expressed as follows:

$[DUE]$

$$\forall i, t, a_{i,t} \in A_{i,t} : U_i(\boldsymbol{c}, a^*_{i,t}, a^*_{-i,t}) \geq U_i(\boldsymbol{c}, a_{i,t}, a^*_{-i,t}) \tag{2.18}$$

The DUE condition is an ex post facto evaluation standard, in which anyone can not increase one's own ultimate utility by taking another strategy $a_{i,t} \neq a^*_i$ at any time. The optimal strategy for driver $i$ is to take $a_{i,t} = a^*_{i,t}$ which is the element of optimal action $a^*_i = \{a^*_{i,t_s}, \ldots, a^*_{i,t_e}\}$ at every time $t$. In other words, DUE is an equilibrium condition for the case that hypothesizes that all drivers completely and accurately predict future conditions in order to optimize their own behavior. DUE has been defined in various ways depending on the hypotheses concerning the details and timing of the driver's strategy. Akamatsu (1995) formalized DUE conditions for a problem in which, while traveling, the driver successively changes the route along the way. On the other hand, Friesz et al. (2011) formalized DUE conditions for a problem in which a driver who has a desired time of arrival at a destination simultaneously chooses a departure time and a route to use. However, the latter does not consider route changes after departure.

**Optimal control under the equilibrium constraints**

In the case of day-to-day control for habitually oversaturated conditions, it is possible to treat driver behavior as a problem that is repeated every day. Therefore, the $[DUE]$ equilibrium condition achieved by a "best response" strategy for each driver would be useful as a model that expresses the actual traffic conditions. Thus, the traffic control that should be sought is formalized in the following $[DTC - DUE]$ format.

$[DTC - DUE]$

$$\boldsymbol{c}^* = \underset{\boldsymbol{c} \in \mathcal{C}}{\text{argmax}} \int_{t_s}^{t_e} SW(\boldsymbol{c}, \boldsymbol{a}^*, t)dt \tag{2.19}$$

$$s.t. \quad h^j(\boldsymbol{c}, \boldsymbol{a}, t) \leq 0, j = \{1, 2, \ldots, k\} \tag{2.20}$$

$$\forall i, t, a_{i,t} \in A_{i,t} : U_i(\boldsymbol{c}, a^*_{i,t}, a^*_{-i,t}) \geq U_i(\boldsymbol{c}, a_{i,t}, a^*_{-i,t}) \tag{2.21}$$

Ukkusuri et al. (2013) and Han et al. (2015) formalized the [DTC-DUE] type of traffic control and proposed various heuristic solution methods. With regard to specific methods of DUE allocation originally presented by Horowitz (1984), there is a series of studies (Peeta and Ziliaskopoulos, 2001), such as Hu and Mahmassani (1997), who considered day-to-day reactions to information presentation, and Huang and Lam (2002), who considered FIFO conditions and vehicle queues. These studies are useful to implement $[DTC - DUE]$ type of control in the real world.

**Optimal control under the myopic behaviors of agents**

For unexpected oversaturated conditions that occur, such as due to an unexpected event, the driver's within-day behavior pattern is important. Considering that the driver's behavior will be even more myopic, and that traffic control and the driver's decision-making will be frequently updated while affecting each other, traffic assignment based on equilibrium conditions is unsuitable; rather, the $[DUO]$ type of allocation, which models the driver's myopic optimization reactions, will be useful. In this case, traffic control can be formalized with the following $[DTC - DUO]$ model.

$[DTC - DUO]$

$$\boldsymbol{c}^* = \underset{\boldsymbol{c} \in \mathcal{C}}{\operatorname{argmax}} \int_{t_s}^{t_e} SW(\boldsymbol{c}, \boldsymbol{a}^*, t) dt \tag{2.22}$$

$$s.t. \quad h^j(\boldsymbol{c}, \boldsymbol{a}, t) \leq 0, j = \{1, 2, \ldots, k\} \tag{2.23}$$

$$\forall i, t : a^*_{i,t} = \underset{a_{i,t} \in A_{i,t}}{\operatorname{argmax}} U_i(a_{i,t}, s_t) \tag{2.24}$$

Chen and Ben-Akiva (1998) formalized a $[DTC - DUO]$ type of control and demonstrated its effectiveness using a simple network. Gartner and Stamatiadis (1998) similarly formalized a $[DTC - DUO]$ type of control, and proposed looking for a solution by model prediction. In this type of traffic control, a model that expresses successive optimal reactions of a driver is very important. As an example of such a model, Hato et al. (1999) considered "travel time recognition methods," "information system accuracy," and "driver heterogeneity," and modeled driver route choice behavior based on information from multiple information sources.

When the future is uncertain, in order to know a user's expected utility, it is necessary to know the user's own perceived probability of all the conditions that may occur; however, it is difficult to know these probabilities. When a user predicts future conditions, various types of information services as well as the user's own experiences and the experiences of others are used. If the same road is repeatedly used, then the driver already has abundant knowledge, including the degree of congestion by day and by time period, methods of interpreting information obtained from information boards, etc. However, under unexpected conditions, that knowledge is not very useful. Most drivers who experience such uncertain circumstances try to obtain and use more information in order to reduce the degree of uncertainty, and select measures best-suited for themselves. For example, it is known from research into road selection that the road choices searched by a user crossing an unknown location are wider than for a user who knows the area well (Bonsall et al., 1997), and that a user under time constraints avoids the usual roads. However, there are more than a few users who react to uncertainty as a "game," and use aggressive behavior (e.g., frequent route changing) depending on the situation (Bonsall, 2004). Hara and Kuwahara (2015), using actual data from the time of the Great East Japan Earthquake, analyzed the behavior of drivers in severe congestion, and demonstrated that the route choice behavior of drivers in unexpected oversaturated conditions is totally different from that under normal circumstances. Additionally, Oyama et al. (2016) focused on this type of myopic behavior and presented a route-choice behavior mode using a time discount factor.

### 2.2.5 Summary of the Effect of User-Optimized Agents

This section has considered the effect of user-optimized agents on traffic control, and has summarized methods of traffic control for oversaturated networks. For habitually oversaturated conditions, the $[DTC - DUE]$ type of traffic control, which hypothesizes dynamic equilibrium conditions, is appropriate, and it appears that traffic control methods such as dynamic congestion fees are effective. On the other hand, for unexpected oversaturated conditions, the $[DTC - DUO]$ type of traffic control, which assumes myopic optimal reactions of drivers, is appropriate. To implement such traffic control, a system that observes traffic conditions in real time, immediately extracts the differences between what occurs habitually and what does not, and instantaneously studies the behavior of a user-optimized agent at a given time based on the obtained data is required.

This section has collectively summarized both human drivers and AVs that represent user-optimized agents. The behavior of human drivers has been observed to have limited rationality and a large degree of uncertainty, and is therefore difficult to model. However, the behavior of AVs is more rational, and therefore modeling it is relatively easy. Accordingly, when constructing a traffic control system, it is desirable to introduce traffic control methods that hypothesize the optimal reactions of a "user-optimized AV." In particular, considering cases in which both "user-optimized" and "society-optimized" AVs coexist, a traffic control method that does not consider the optimal reaction of a "user-optimized AV" could put "society-optimized AVs" at a severe disadvantage. This should be well-considered when designing a traffic control system with AVs.

## 2.3 Fog Computing

As shown in the previous sections, implementing a traffic control system for a city is an extremely complicated problem that must take into account an unsteady network and simultaneously consider the dynamic traffic control and dynamic traffic allocation problems. In particular, in the $[DTC - DUO]$ model that considers within-day behavior, it is necessary to observe the driver's behavior in real time, predict any myopic reactions, and provide quick feedback to the traffic control system. This requires cooperative control between a centrally-managed system with overall control, and local control systems that implement quick control in appropriately sized subareas. It is important to have not only this hierarchical centralized/localized interaction, but also cooperative local–local control. In particular, under oversaturated conditions, it is desirable to adaptively configure such a cooperative network according to the situation.

As a method of implementing such a system, attention has recently focused on the idea of "Fog computing (Bonomi et al., 2012)," as shown in Fig. 2.1. This system guarantees service with low delay by distributing computing resources close to users as cores for integrating multiple sensors and actuators. Additionally, this system aims at total optimization by performing cooperative network control among multiple cores, as well as with a higher-level cloud system. If such a system is assumed, then traffic control for an oversaturated network can appropriately implement the two problems of "dynamic traffic control" and "dynamic traffic allocation" in each of three hierarchical levels: a "cloud layer," a "fog layer," and an "edge layer." The cloud layer must understand the many conditions obtained from the lower-level fog and edge layers, divide the optimal traffic control into sub-problems of appropriate size, and present them to the corresponding cores. For example, in

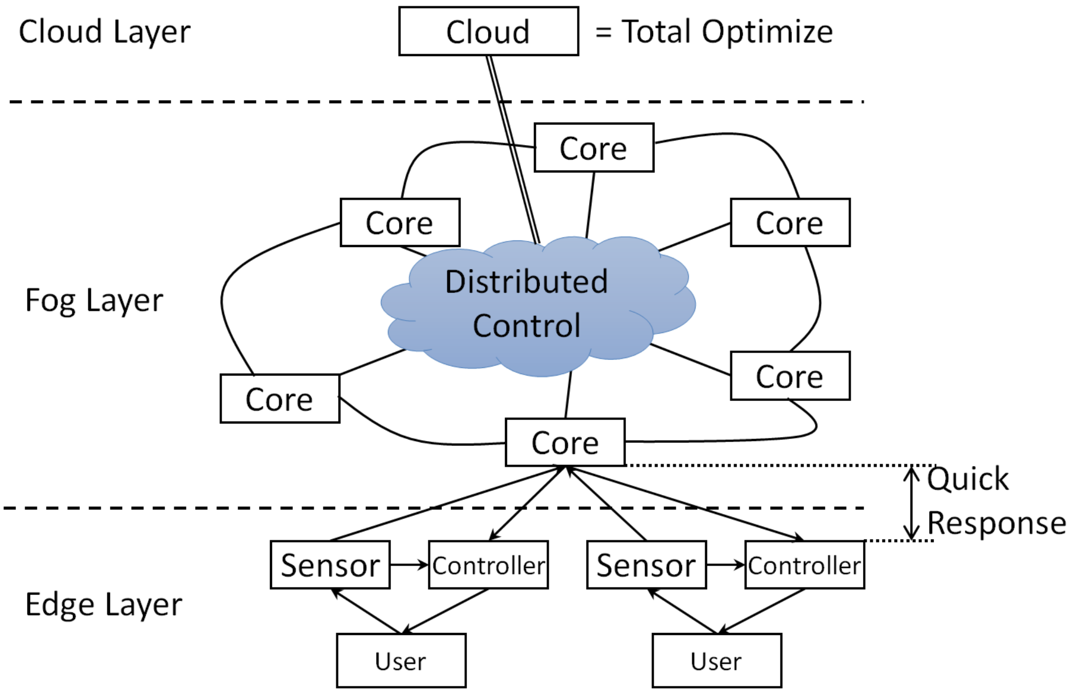



FIGURE 2.1: Fog Computing

the field of security games (Tambe, 2011) there have been many reports of research that formalizes a Stackelberg game, which includes uncertain human behavior, as MIP problems of appropriate sizes (Paruchuri et al., 2008; Pita et al., 2009), and this may be applicable to traffic control for a city. The fog layer requires area networking that is appropriate for congestion conditions, and encompasses control algorithms that can be implemented with low delay. It is also possible to apply the idea of self-adaptive systems (De Lemos et al., 2013; Kramer and Magee, 2007; Salehie and Tahvildari, 2009). Regarding such systems, research progress is being made on sensor networks that generate effective information by integrating the information of many sensors (Sohrabi et al., 2000; Yick et al., 2008), and in the field of software engineering. For the traffic control of a city, an example of a specific control variable that ties together a higher-level layer and a lower-level layer is "traffic density." In order to achieve the optimal condition in the city network at the higher-level layer, the fog layer is responsible for finding values for the appropriate traffic density in each area and transmitting them to the lower-level layer. The edge layer would both sense driver behavior and implement direct control of vehicle behavior. Research is needed in areas such as intersection control (Dresner and Stone, 2005) and platoon control (Shladover et al., 1991)that have anticipated automated driving.

In a city with a complicated network, rather than a traffic control system that maximizes local traffic capacity, a system that can quickly achieve the appropriate traffic density received from the higher-level layer is required.

## 2.4 Activity-based travel analysis

Some earlier studies of activity-based travel analysis have considered the problem of traffic assignment on a time–space expanded network. Here, we review these works, which contrast with traditional trip-based analysis in their framing of a trip as the derived demand of users' activities and try to model travelers' time usage within their daily activities. However, the choice set of such models becomes extensive, so the concept of a time–space prism (Hägerstrand, 1970) becomes vital. The earliest studies of activity-based travel analysis were reviewed extensively in Kitamura (1988).

### 2.4.1 Activity-based utility model (ABM)

Axhausen and Gärling (1992) classified activity analysis into two conceptual frameworks: *utility-maximization* and *electric*. The former assumes that people choose to spend their time in a way that maximizes utility within their time–space constraints, whereas the latter addresses the scheduling process explicitly. Although the problems discussed in these frameworks are indeed difficult to come to grips with, their solutions are essential for enabling policy makers to appropriately manage traffic demand generated from household activities. Kitamura et al. (1996) proposed an activity-based utility model that replicates adaptive time-of-day dynamics, and introduced a simulator that offers dynamic and integrated forecasting of elements such as transportation and land use. Recker (2001) discussed the relationship between trip-based and activity-based travel analysis and demonstrated that activity-based travel analysis can be formalized with mathematical programming based on traditional trip-based modeling methodologies with the addition of temporal and spatial constraints.

### 2.4.2 Space-time expanded network

The space-time expanded network is key to solving various problems formulated by the activity-based utility model. Such approaches were originally employed to solve dynamic traffic assignment (DTA) problems in many dynamic user equilibrium (DUE) models (Drissi-Kaitouni, 1993; Yang and Meng, 1998). This method is well-suited to activity-based travel analysis because it expressly considers time–space prism constraints. Indeed, Lam and Yin (2001) formulated the combined activity/route choice problem as the ideal DUE and provided a method for solving this problem using a space–time extended network. Moreover, Arentze and Timmermans (2004) introduced the multistate supernetwork, which can represent multimodal transport system with sequential activities; this model was inspired by the supernetwork concept introduced by Sheffi (1985), which aims to enrich network representations in order to model traffic mode choices. In the multistate supernetwork, each node expresses a combination of an activity state and a vehicle state, and each link expresses a transition between states. Thus, the choice of sequential activities is expressed as a trajectory in the supernetwork and complete trip chains that involve multiple transport modes can be obtained as a least-cost path. Liu et al. (2015) later formalized dynamic activity travel assignment as a discrete-time DUE problem on a multistate supernetwork. Oyama et al. (2016) proposed trajectory-oriented traffic management in which traveler activity is expressed as a trajectory in the time–space expanded network and travelers' behavior choices are modeled recursively with a discount rate. Hara and Hato (2017) considered optimal user and vehicle assignments for a car-sharing service by introducing the temporal and spatial connection of users with a vehicle with a time–space expanded network. Thus, activity-based analysis by the time–space expanded network is not only used to express multimodal and multistate behavior, but also to develop urban planning or traffic service policies that consider user heterogeneity.

## 2.5 Discussions

This chapter has summarized the problems and future directions of traffic control systems for cities that include AVs, considering that some parts of decision-making are transferred from users to systems.

First, we organize the existing works of traffic control for cities with densely distributed vehicles that do not consider AVs. In cities that have complicated road networks, maximization of local traffic capacity is not necessarily good for the traffic environment of the entire network. Rather, the control of traffic density, such as "speed control" and "vehicle-queue control" plays an important role in such road networks. Some works focus on the interaction between each driver and the traffic operator and formulate it as a bi-level optimization problem. By contrast, many existing works considering AVs propose algorithms that aim to increase the capacity of links or intersections, focusing on that the uncertainty of vehicle behavior due to human drivers will decrease with the penetration of AVs. However, most of these works only focus on the decision-making of drivers related to the vehicle behavior. Thus, they do not show how AVs will be introduced to cities with complicated road networks and how the system including AVs works in people with higher-level decision-making, such as route-choice, traffic-mode choice, and trip-making decision.

Nowadays, the route-choice algorithm that is conventionally implemented to the navigation system equipped with each vehicle is transferring to the centrally controlled servers in car OEMs. Also, new types of mobility services, such as ride-sharing or MaaS (Mobility as a Service) are widely introduced. Considering these trends, it can be said that the higher-level decision-making, such as route-choice and traffic-mode choice will be transferred gradually from users to systems.

Considering these situations, in this chapter, we clarified that traffic services in coming age should consider the "traffic control system with AVs" not only as the control that enlarge the capacity of links and intersections by the transfer of lower-level decision-making from drivers to the system, but also as the "Traffic services" that provide a pleasant mobility experience for people in cities by the transfer of higher-level decision-making. Such a traffic service will be implemented by traffic service operators that aim to maximize the utility of their customers, based on a traffic systems or equipment developed by the traffic administrator that aims to maximize social welfare, and AVs are included in such systems as a tool. Highly controllable AVs contribute to both of traffic service operators and the traffic administrator. The traffic administrator may be able to control traffic density in oversaturated road network by using AVs, while traffic operators may control AVs to provide appropriate services to heterogeneous customers that have heterogeneity to the origin, the destination, departure and arrival time, and the favor of the scenery around the road. Thus, the traffic phenomenon in cities with AVs can be expressed as the interaction between the traffic control by the traffic administrator that aims to maximize social welfare and traffic services by service operators that aims to maximize the utility of customers.

As shown in this chapter, the behavior of the users plays an important role in the scene of transportation. To consider the traffic services in coming age with the penetration of automated vehicles or the concept of MaaS (Mobility as a Service), we have to re-organize existing works in the related fields, as well as propose algorithms that can be implemented to the realistic situations in cities, focusing on the behavior of users. We try to tackle these problems in the following chapters of this theses.

# Chapter 3

# Dynamic Traffic Control in Unsteady Networks with Closed-loop Structures

In this chapter, we propose a dynamic traffic control algorithm in unsteady and oversaturated road network, as is the algorithm for traffic administrators. Specifically, we focus on closed-loop structures that are made by multiple links in the road network and propose an algorithm that aims to prevent gridlock phenomenon.

The content of this chapter has been presented in *Hayakawa and Hato (2018c), Evaluation of dynamic traffic control in unsteady networks with closed-loop structures, Transportation Research Board 97th Annual Meeting, Washington D.C.*. A part of the content is published as *Hayakawa and Hato (2017), Traffic control in oversaturated networks with closed-loop structures, Journal of JSCE D3 (in Japanese).*

We propose a novel traffic control algorithm that can be used in oversaturated road networks in large cities. For sophisticated traffic control, it is important to consider the dynamic behavior of vehicle queues. Such queues obstruct traffic flow in various directions, making the traffic situation chaotic. This results in severe traffic congestion, or so-called gridlock. In this study, we focus on closed-loop structures in road networks and vehicle queue advancement in these loops. We formalize the occurrence of gridlock in one-way road networks, and propose a traffic control algorithm called Z-control, which prevents gridlock and minimizes the total time spent based on a model predictive scheme. Moreover, we evaluated some algorithms for dynamic traffic control, including our proposed Z-control and some benchmarks. To do so, we conducted numerical experiments, considering two extreme scenarios pertaining to the behavior of drivers. In a "non-reactive" scenario, where drivers do not react to traffic conditions in the network, we showed that the proposed Z-control outperformed other benchmark algorithms. However, in a "reactive" scenario, where drivers are assumed to have perfect information about the traffic conditions in the network and react to it, no algorithm performed better than the situation without any control. Our research shows that to treat oversaturated traffic in cities suitably, it is important to control, or at least consider, the route-choice behavior of drivers, which may be achieved with automated vehicles.

## 3.1 Introduction

In a large city with a densely distributed road network, traffic jams at local spots can spread to wider areas, significantly influencing the economy(Hartgen et al., 2009). In particular, if the road network is oversaturated by many vehicles, the traffic flow is obstructed by vehicle queues, rapidly exacerbating the situation. We call this type of road network, in which there is an excess of demand over supply, an "oversaturated network."

In oversaturated networks, vehicle queues that are formed at intersections or merging sections interfere with the traffic flow in other directions, bringing vehicles to a standstill despite green traffic signals. As a result, vehicle queues increase in many directions, ultimately resulting in traffic paralysis, or so-called gridlock (Daganzo, 1996; Mahmassani et al., 2013). The definition of "gridlock" is unclear, but in this chapter, we refer to the state where widespread vehicle queues seriously congest traffic conditions as the "broad sense of the gridlock phenomenon," and the state where the traffic throughput falls to zero from congestion as the "narrow sense of the gridlock phenomenon," consistent with Mahmassani et al. (2013).

One basic idea for traffic control when addressing oversaturated networks is density control using a macroscopic fundamental diagram (MFD)(Daganzo, 2007). This idea is well known to be effective with uniform and steady traffic. However, to the best of our knowledge, no concrete traffic control algorithm that uses an MFD has been developed to achieve appropriate density control in non-uniform and unsteady traffic situations.

On the other hand, a considerable body of work has proposed traffic signal algorithms for oversaturated networks. For instance, Gazis (1964) proposed a basic idea to obtain a green split to minimize the total time spent in a network with two intersections and without (right and left) turning, using a graphical scheme. However, considering vehicle queues and turning in a general network with multiple intersections, the optimal problem with the objective function to minimize the total time spent is complicated by delay terms for state variables(Lovell and Daganzo, 2000), making it difficult to solve the optimal problem analytically. By contrast, Singh and Tamura (1974) treats traffic control in oversaturated networks as a problem of queue management(Quinn, 1992), and formalizes it as an optimal problem of time-delay systems(Richard, 2003) to minimize the total length of the vehicle queue. This approach considers unsteady traffic flow and the dynamic behavior of vehicle queues. However, the approach does not aim to minimize the total time spent, and consequently fails, depending on the shape or congested situation of the road network.

In addition, in oversaturated networks, especially in unexpected cases, drivers adopt reactive behavior to the traffic conditions of the network(Bonsall, 2004; Hara and Kuwahara, 2015). However, most existing work proposes traffic signal algorithms for oversaturated networks, assuming that traffic flow, including user route selection, is given. Thus, the effectiveness of such algorithms at dealing with reactive drivers should be discussed.

To address these shortcomings, we proposed an algorithm of traffic control in oversaturated networks with closed-loop structures. Based on the study, we make the following novel contributions.

- We focus on closed-loop structures in road networks and vehicle queues advancing in these loops, and formalize the occurrence condition of the gridlock phenomenon in one-way road networks.
- Considering this condition, we propose a traffic control algorithm, called Z-control, which minimizes the total time spent based on a model predictive scheme.
- In a numerical study, we evaluate dynamic traffic control algorithms, including the proposed Z-control, considering two scenarios: a scenario where drivers

do not react to the traffic conditions of the network, and a scenario where they do. Furthermore, we discuss the basic policy required for traffic control in unsteady networks.

## 3.2 Related Work

In this section, we discuss works relating to our research. The section is divided into three parts: existing works related to the gridlock phenomenon, traffic signal algorithms for oversaturated networks, and the combined traffic control and assignment problem.

### 3.2.1 Gridlock phenomenon

Regarding gridlock, Daganzo (1996) considers a highway network with multiple on and off ramps, showing the occurrence of traffic self-destruction from merging. Given a closed-loop road network, the half-life period of traffic volume along the main track can be expressed by the merging rate of an on ramp and the diversion rate of an off ramp. Limiting traffic flow on the main track is essential to prevent gridlock. However, this work considers a network with a single closed-loop, and cannot be applied to a more general network with multiple closed-loops, where the traffic flowing out from a closed-loop is at the same time the traffic flowing into another closed-loop.

By contrast, Daganzo (2007) considers gridlock from a macroscopic perspective, using an MFD. However, no concrete traffic control algorithm has been proposed based on this research.

In this chapter, we show the occurrence condition of the gridlock phenomenon in a general network with multiple closed-loops, with state variables that can be used for concrete traffic signal control. Considering this condition, we propose a traffic control algorithm to prevent gridlock.

### 3.2.2 Traffic signal algorithms for oversaturated networks

As explained in the previous section, traffic signal algorithms for oversaturated networks are divided into algorithms that minimize the total spent time(Chang and Lin, 2000; D'ans and Gazis, 1976; Gazis, 1964; Lo, 2001; Michalopoulos and Stephanopoulos, 1977) and algorithms based on queue management(Aboudolas et al., 2010; Abu-Lebdeh and Benekohal, 1997; Rathi, 1988; Singh and Tamura, 1974). Generally, the former are difficult to apply in complicated networks, because the calculation cost is too high. In such networks, the latter are more suitable. In particular, Aboudolas et al. (2010) proposes a model-predictive traffic control algorithm that aims at balancing vehicle queue occupancy among all links in the network, considering unsteady traffic flow and the chaotic behavior of a vehicle queue.

However, many existing works, including Aboudolas et al. (2010), consider the queue management with a link unit, and no work to our knowledge has proposed queue management strategies to prevent gridlock, while focusing on the geographic relations between links. In addition, in many algorithms for oversaturated networks, the route-choice behavior of users is given as an input, and the reactive behavior of users is not appropriately considered. In this chapter, we tackle these problems.

To treat a complicated oversaturated situation, all works shown in this part propose algorithms that control a green split for each link, instead of a green split for each signal phase that includes a combination of links at an intersection. Similar to

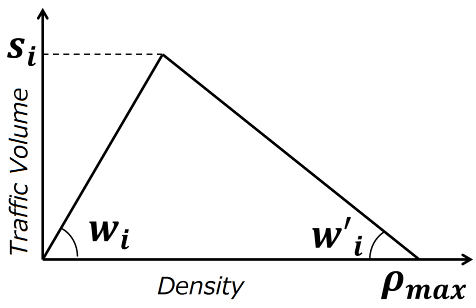



FIGURE 3.1: Fundamental diagllam

these works, we exclusively consider one-way networks in which a green split for each link can be independently controlled.

### 3.2.3 Combined traffic control and assignment problem

The importance of considering the reactive behavior of drivers, such as route choice, for traffic control algorithms is well-known(Smith, 1978), as confirmed by Braess's paradox(Frank, 1981). This type of problem is called as the combined traffic control and assignment problem, first considered in Allsop (1974) for traffic signal control. In that work, the problem is formalized as a bi-level optimization problem, and a method is proposed to obtain traffic control parameters with iterative calculations of these upper and lower problems. Similarly, Smith and Van Vuren (1993) summarizes the conditions for objective functions and link cost functions, on which the iterative calculations converge. If this type of iterative calculation is used to solve the bi-level problem, the solution coincides with the Cournot–Nash equilibrium, where the solution is a fixed point that does not update either the upper and lower problem.

On the other hand, Fisk (1984) formalizes this problem as a Stackelberg game, in which the traffic controller takes the reactive behavior of drivers into consideration in advance. In this case, because the solution is not a fixed point, many heuristic algorithms are proposed (Ceylan and Bell, 2004; Mitsakis et al., 2011; Ukkusuri et al., 2013; Yang and Yagar, 1995).

However, none of the research above focuses on oversaturated networks with multiple nodes and links. Thus, it remains unclear how robust these methods are to the reactive behavior of drivers in oversaturated networks. In the numerical study below, we tackle this problem.

## 3.3 Closed-loop Structures and the Gridlock Phenomenon

In this section, we first introduce the basic network and traffic flow model. A state variable in this model considers the property of time-delay, in order to express the development of a queue of vehicles in an over-saturated network. Then, we focus on the closed-loop structures in the network and describe the process of gridlock and a policy designed to prevent it.

### 3.3.1 Network and traffic flow model

We consider a network $(N, E)$ with a set $N$ of nodes and a set $E$ of links. The set $N$ of nodes has $k$ nodes. We assume a triangular fundamental diagram for the links

TABLE 3.1: Notations in Chapter 3

| Parameter | Notation | Unit |
|---|---|---|
| Link parameters | | |
| Length of link $i$ | $\alpha_i$ | $m$ |
| Saturated flow rate of link $i$ | $f_i^s$ | $veh./s$ |
| Saturated space density of link $i$ | $\rho_i^{max}$ | $veh./m$ |
| Maximum number of vehicles in link $i$ | $n_i^{max} = \rho_i^{max}\alpha_i$ | $veh.$ |
| Velocity of the forward wave in link $i$ | $w_i$ | $m/s$ |
| Velocity of the backward wave in link $i$ | $w_i'$ | $m/s$ |
| Transit time of the forward wave in link $i$ | $\tau_i = \alpha_i/w_i$ | $s$ |
| Transit time of the backward wave in link $i$ | $\tau_i' = \alpha_i/w_i'$ | $s$ |
| Traffic signal parameters at time $t$ | | |
| Fraction of the right of priority given to link $i$ | $g_i(t)$ | – |
| Link flow parameters at time $t$ | | |
| Cumulative arrivals at link $i$ | $J_i(t)$ | $veh.$ |
| Arrival rate at link $i$ | $\lambda_i(t) = \frac{d}{dt}J_i(t)$ | $veh./s$ |
| Arrival rate at link $i$ from link $j$ | $\lambda_i^{\langle j\rangle}(t)$ | $veh./s$ |
| Cumulative departures from link $i$ | $D_i(t)$ | $veh.$ |
| Departure rate from link $i$ | $\mu_i(t) = \frac{d}{dt}D_i(t)$ | $veh./s$ |
| Departure rate from link $i$ to link $j$ | $\mu_i^{\langle j\rangle}(t)$ | $veh./s$ |
| Demand at link $i$ | $q_i(t)$ | $veh./s$ |
| Number of vehicles in link $i$ | $n_i(t) = J_i(t) - D_i(t)$ | $veh.$ |
| Proposed state variable at link $i$ | $x_i(t) = J_i(t-\tau_i) - D_i(t)$ | $veh.$ |
| Proposed state variable at link $i$ | $y_i(t) = J_i(t) - D_i(t-\tau_i')$ | $veh.$ |

(See Figure 3.1) and denote the link parameters and link flow parameters as shown in Table 4.1.

Based on Kinematic Wave Theory (Newell, 1993), the following equations hold:

$$J_i(t-\tau_i) - D_i(t) \geq 0. \tag{3.1}$$

$$J_i(t) - D_i(t-\tau_i') \leq n_i^{max}. \tag{3.2}$$

We use $\boldsymbol{P}(t)$ to denote a transition matrix between links, where $\boldsymbol{P}(t)$ is a $K \times K$ matrix, and the element $p_{ij}(t) = \mu_i^{\langle j\rangle}(t)/\mu_i(t)$ denotes the ratio of the arrival at link $j$ from link $i$ to all departures from link $i$. The transition matrix $\boldsymbol{P}(t)$ satisfies the following equation:

$$\forall i : \sum_{j\in E} p_{ij} \leq 1. \tag{3.3}$$

The inequality only holds for nodes that are destinations for some traffic. Traffic volume $\mu_i(1-\sum_{j\in E} p_{ij})$ is absorbed in the node between links $i$ and $j$. The flow conservation at nodes is expressed as

$$\lambda_i(t) = \sum_{j\in E} p_{ji}(t)\cdot\mu_j(t) + q_i(t), \tag{3.4}$$

or also expressed as,

$$\boldsymbol{\lambda}(t) = \boldsymbol{\mu}(t)\cdot\boldsymbol{P}(t) + \boldsymbol{q}(t), \tag{3.5}$$

by using $\boldsymbol{\lambda}(t) = [\lambda_1(t),\ldots,\lambda_K(t)]$, $\boldsymbol{\mu}(t) = [\mu_1(t),\ldots,\mu_K(t)]$, $\boldsymbol{q}(t) = [q_1(t),\ldots,q_K(t)]$.

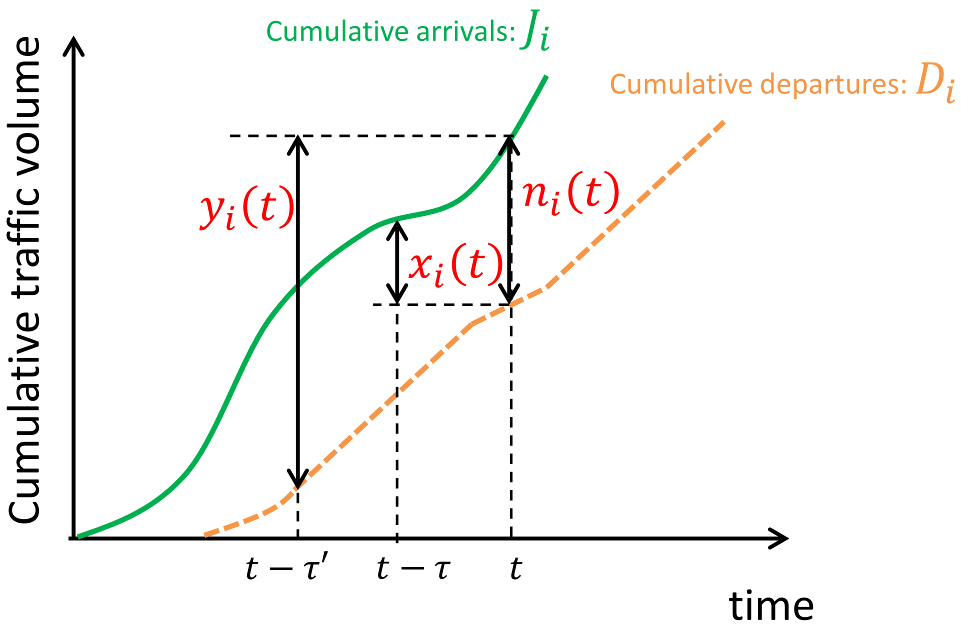



FIGURE 3.2: State variables on cumulative arrival and departure curves

### 3.3.2 State variables considering time-delay

Considering the time-delay of a vehicle queue within the link, we introduce the following state variables $n_i(t)$, $x_i(t)$, $y_i(t)$:

$$n_i(t) = J_i(t) - D_i(t), \tag{3.6}$$

$$x_i(t) = J_i(t-\tau_i) - D_i(t), \tag{3.7}$$

$$y_i(t) = J_i(t) - D_i(t-\tau_i'). \tag{3.8}$$

We show these state variables on cumulative arrival and departure curves in Fig. 3.2. From Eqs. (3.1) and (3.2), the following equations hold:

$$x_i(t) \geq 0, \tag{3.9}$$

$$y_i(t) \leq n_i^{max}. \tag{3.10}$$

If the equality in Eq. (3.9) holds, there is no vehicle queue downstream from link $i$. In this chapter, we call this state the "free-flow state." On the other hand, if the equality in Eq. (3.10) holds, the vehicle queue fills all over link $i$. In this chapter, we call this state a "saturated state."

The maximum departure rate $X_i(t)$ and the maximum arrival rate $Y_i(t)$ of link $i \in E$ are stated as follows(Kuwahara and Akamatsu, 2001):

$$X_i(t) = \begin{cases} g_i(t) \cdot f_i^s & \text{if } x_i(t) > 0 \text{ or } \lambda_i(t-\tau_i) \geq g_i(t) \cdot f_i^s \\ \lambda_i(t-\tau_i) & \text{otherwise} \end{cases} \tag{3.11}$$

$$Y_i(t) = \begin{cases} f_i^s & \text{if } y_i(t) < n_i^{max} \\ \mu_i(t-\tau_i') & \text{otherwise} \end{cases} \tag{3.12}$$

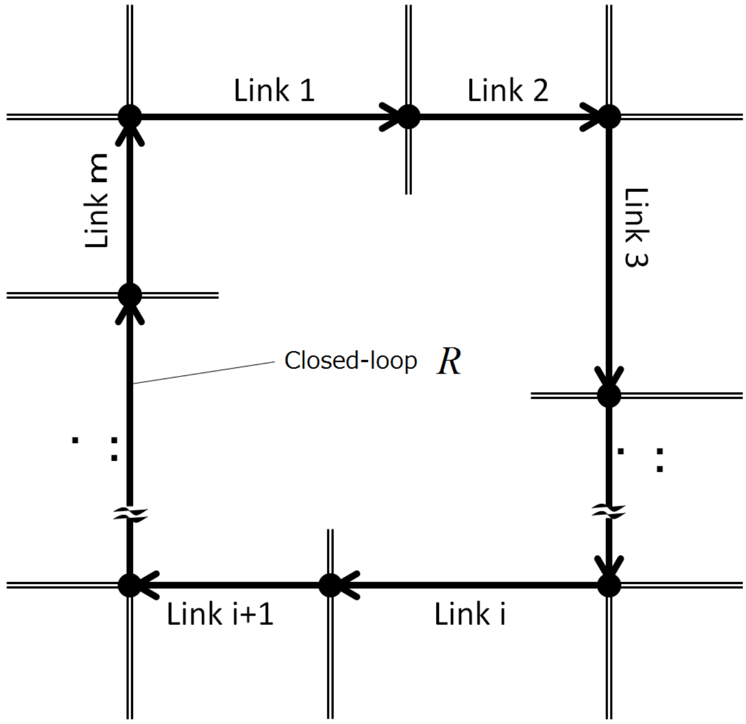



FIGURE 3.3: Closed-loop structures in a network

Here, $g_i(t)$ denotes the green split for link $i$. For $X_i(t)$ and $Y_i(t)$, the equations stated must always be satisfied.

$$\forall t, i \in E : \mu_i(t) \leq X_i(t), \tag{3.13}$$

$$\forall t, i \in E : \lambda_i(t) \leq Y_i(t). \tag{3.14}$$

### 3.3.3 Closed-loop structures in the network

We use $\mathcal{R} = \{1, 2, \cdots, m\} \subset E$ to denote a set of $m$ links that form a closed-loop, as shown in Fig. 3.3. In this chapter, we call this loop "closed-loop $\mathcal{R}$." In network $(N, E)$, there are many such closed-loops. We use $\boldsymbol{\mathcal{R}}$ to denote a set of these closed-loops.

Next, we focus on a specific closed-loop $\mathcal{R} = \{1, 2, \cdots, m\} \subset E$, in which the downstream of link $i \in \mathcal{R}$ is connected to the upstream of link $i+1 \in \mathcal{R}$, as shown in Fig. 3.3. In the equations stated in this section, we regard link $m+1$ as link 1, and we regard link $m+k$ as link $k$.

We introduce variables $\zeta_i(t)$ and $\eta_i(t)$, as follows:

$$\zeta_i(t) = \frac{\mu_i^{\langle i+1 \rangle}(t)}{\mu_i(t)}, \quad \eta_i(t) = \frac{\lambda_{i+1}^{\langle i \rangle}(t)}{\lambda_{i+1}(t)}, \tag{3.15}$$

where $\zeta_i(t)$ is the ratio of departure flow inside the loop, and $\eta_i(t)$ is the ratio of arrival flow inside the loop. Naturally, $0 \leq \zeta_i(t) \leq 1$ and $0 \leq \eta_i(t) \leq 1$ hold. In addition, by

$$\forall t : \zeta_i(t) \cdot \mu_i(t) = \mu_i^{\langle i+1 \rangle}(t) = \lambda_{i+1}^{\langle i \rangle}(t) = \eta_i(t) \cdot \lambda_{i+1}(t), \tag{3.16}$$

and Eq (3.14),

$$\mu_i(t) = \frac{\eta_i(t)}{\zeta_i(t)} \lambda_{i+1}(t) \leq \frac{\eta_i(t)}{\zeta_i(t)} Y_{i+1}(t), \tag{3.17}$$

holds.

### 3.3.4 Occurrence condition of the gridlock phenomenon

Assume that link $i+1$ is in a "saturated state" at time $t$. That is, $y_{i+1}(t) = n_{i+1}^{max}$. Then, by Eqs. (3.12) and (3.17),

$$\mu_i(t) \leq \frac{\eta_i(t)}{\zeta_i(t)} \cdot \mu_{i+1}(t - \tau'_{i+1}), \tag{3.18}$$

holds. Further assume that link $i+2$ is also in a "saturated state" at time $t - \tau'_{i+1}$. That is, $y_{i+2}(t - \tau'_{i+1}) = n_{i+2}^{max}$. Then,

$$\mu_i(t) \leq \frac{\eta_i(t)}{\zeta_i(t)} \cdot \frac{\eta_{i+1}(t - \tau'_{i+1})}{\zeta_{i+1}(t - \tau'_{i+1})} \cdot \mu_{i+2}(t - \tau'_{i+1} - \tau'_{i+2}), \tag{3.19}$$

holds. Now, we introduce variables $\tau'_{ij} = -\tau'_i + \sum_{k=i}^{j} \tau'_k$ and $\tau'_{\mathcal{R}} = \sum_{k \in \mathcal{R}} \tau'_k$, and adopt the similar assumption for all over the link in the closed-loop $\mathcal{R}$. Then, assume that $\forall k \in \mathcal{R} : y_{k+1}(t - \tau'_{ik}) = n_{k+1}^{max}$,

$$\mu_i(t) \leq \prod_{k \in \mathcal{R}} \frac{\eta_k(t - \tau'_{ik})}{\zeta_k(t - \tau'_{ik})} \cdot \mu_i(t - \tau'_{\mathcal{R}}) \tag{3.20}$$

holds. Here, we define variables $Z_{\mathcal{R},i}(t)$ and $B_{\mathcal{R},i}(t)$ as follows:

$$Z_{\mathcal{R},i}(t) = \frac{\sum_{k \in \mathcal{R}} y_{k+1}(t - \tau'_{ik})}{\sum_{k \in \mathcal{R}} n_{k+1}^{max}}, \tag{3.21}$$

$$B_{\mathcal{R},i}(t) = \prod_{k \in \mathcal{R}} \frac{\eta_k(t - \tau'_{ik})}{\zeta_k(t - \tau'_{ik})}. \tag{3.22}$$

Given these, under the assumption that $Z_{\mathcal{R},i}(t) = 1$,

$$\mu_i(t) \leq B_{\mathcal{R},i}(t) \mu_i(t - \tau'_{\mathcal{R}}) \tag{3.23}$$

holds, and, if $\forall t : B_{\mathcal{R},i}(t) < 1$, then $\lim_{t \to \infty} \mu_i(t) = 0$. That is, when both $\forall t : Z_{\mathcal{R},i}(t) = 1$ and $\forall t : B_{\mathcal{R},i}(t) < 1$ hold, the departure flows of all links in the closed-loop decrease to zero, and "the narrow sense of the gridlock phenomenon" is generated.

### 3.3.5 Strategies to prevent gridlock phenomenon

From the above, strategies to prevent gridlock in the closed-loop $\mathcal{R}$ can be classified into two categories: one aiming at $B_{\mathcal{R},i}(t) \geq 1$, and the other at $Z_{\mathcal{R},i}(t) < 1$.

Strategies designed to satisfy $B_{\mathcal{R},i}(t) \geq 1$ control $\zeta_i(t)$ and $\eta_i(t)$, which refer to the fraction of flow entering and leaving the closed-loop $\mathcal{R}$ [1]. More simply, this can be achieved by giving priority to the traffic flowing out from the closed-loop.

[1] If we assume that $\zeta_i(t)$ and $\eta_i(t)$ for all links in the closed-loop are uniform and steady, $B_{\mathcal{R},i}(t)$ coincides with the parameter $f_j$ introduced by Daganzo (1996).

This strategy is very common in roundabouts in many countries. However, it is not trivial to apply this strategy to more complex road networks in cities. For instance, in a city grid network, traffic flowing out from a closed-loop is also traffic flowing into another closed-loop. Thus, it is challenging to define the priority.

On the other hand, strategies designed to satisfy $Z_{\mathcal{R},i}(t) < 1$ manage the traffic queue by introducing a constraint, such that

$$\forall t, i \in \mathcal{R}; \quad Z_{\mathcal{R},i}(t) < 1, \tag{3.24}$$

for a loop $\mathcal{R}$. If this constraint is applied to a network with multiple closed-loop structures, the constraints are stated as

$$\forall t, \mathcal{R} \in \boldsymbol{\mathcal{R}}, i \in \mathcal{R}; \quad Z_{\mathcal{R},i}(t) < 1. \tag{3.25}$$

Here, $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M\}$ denotes the set of all closed-loops existing in the network. For every time step, each closed loop $\mathcal{R} \in \boldsymbol{\mathcal{R}}$ has as many constraints as the number of links included in $\mathcal{R}$. Consequently, there are too many constraints in the whole network. To decrease the number of constraints, we introduce

$$\tilde{Z}_{\mathcal{R}}(t) = \frac{\sum_{k \in \mathcal{R}} y_k(t)}{\sum_{k \in \mathcal{R}} n_k^{max}}, \tag{3.26}$$

which ignores the delay between links $\tau'_{ik}$ in Eq. (3.21). We then propose a set of constraints, such that

$$\forall t, \mathcal{R} \in \boldsymbol{\mathcal{R}}; \quad \tilde{Z}_{\mathcal{R}}(t) \leq 1 - \epsilon, \tag{3.27}$$

where $\epsilon > 0$ is the arbitrarily set by the controller. Note that, although we ignore the delay between links in the definition of $\tilde{Z}_{\mathcal{R}}(t)$, we still consider the delay within the link by the state variable $y_k(t)$. From the definition of $y_k(t)$, $\tilde{Z}_{\mathcal{R}}(t) = 1$ means that all links in the closed-loop are in a "saturated state." Thus, while Eq. (3.25) is a set of constraints designed to prevent the "narrow sense of gridlock," Eq. (3.27) is a set of constraints designed to prevent the "broad sense of gridlock." From Eq. (3.27), the number of constraints is reduced to the number of closed-loops in the network. In the next section, we propose a traffic control algorithm that can prevent gridlock, using Eq. (3.27).

## 3.4 Proposed Control Algorithm (Z-control)

In this section, we describe Z-control, a traffic control algorithm for oversaturated networks with closed-loop structures, using the parameter $\tilde{Z}_{\mathcal{R}}$ introduced in the previous section. Z-control is designed to minimize the total time spent by all drivers, with constraints stated by Eq. (3.27) to prevent gridlock. We discuss the concrete algorithm for Z-control in the following parts.

### 3.4.1 Formulation as a linear programming problem

In this part, we formalize Z-control as a linear programming problem. We consider a model with discrete time steps $T = \{t_0, \ldots t^f\}$ with time unit $\Delta t$. The control variables are the split $\boldsymbol{g} = [\boldsymbol{g}(t_0), \ldots, \boldsymbol{g}(t^f)]$ from the traffic signal with $\boldsymbol{g}(t) = [g_1(t), \ldots, g_K(t)]$. Here, $g_i(t)$ is the fraction of the right of priority given to link $i \in E$ at $t \in T$. Traffic control designed to minimize the total time spent by all users can be formalized as the following optimization problem [min-TT]:

$$\max_{\boldsymbol{g}} \quad \sum_{t\in T}\sum_{i\in E}(-n_i(t)) \tag{3.28}$$

$$\text{s.t.} \quad \boldsymbol{\lambda}(t) = \boldsymbol{\mu}(t)\boldsymbol{P}(t) + \boldsymbol{q}(t), \qquad \forall t \in T \tag{3.29}$$

$$\boldsymbol{n}(t+\Delta t) = \boldsymbol{n}(t) + \boldsymbol{\lambda}(t) - \boldsymbol{\mu}(t), \qquad \forall t \in T' \tag{3.30}$$

$$\boldsymbol{x}(t) = \boldsymbol{n}(t) - \boldsymbol{\lambda^{cum}}(t), \qquad \forall t \in T \tag{3.31}$$

$$\boldsymbol{y}(t) = \boldsymbol{n}(t) + \boldsymbol{\mu^{cum}}(t), \qquad \forall t \in T \tag{3.32}$$

$$\mu_i(t) = f_i^s \cdot g_i(t), \qquad \forall i \in E, \qquad \forall t \in T \tag{3.33}$$

$$\sum_{j\in I(k)} g_i(t) \leq 1, \qquad \forall k \in N, \qquad \forall t \in T \tag{3.34}$$

$$\tilde{Z}_{\mathcal{R}}(t) \leq 1-\epsilon, \qquad \forall \mathcal{R} \in \boldsymbol{\mathcal{R}}, \qquad \forall t \in T \tag{3.35}$$

$$x_i(t) \geq 0, \qquad \forall i \in E, \qquad \forall t \in T \tag{3.36}$$

$$y_i(t) \leq n_i^{max}, \qquad \forall i \in E, \qquad \forall t \in T \tag{3.37}$$

$$0 \leq \mu_i(t) \leq f_i^s, \qquad \forall i \in E, \qquad \forall t \in T \tag{3.38}$$

$$0 \leq \lambda_i(t) \leq f_i^s, \qquad \forall i \in E, \qquad \forall t \in T \tag{3.39}$$

$$n_i(0) = 0 \qquad \forall i \in E \tag{3.40}$$

where $\boldsymbol{n}(t) = [n_1(t),\ldots,n_K(t)]$, $\boldsymbol{x}(t) = [x_1(t),\ldots,x_K(t)]$, $\boldsymbol{y}(t) = [y_1(t),\ldots,y_K(t)]$. Eq. (3.29) expresses flow conservation. Eq. (3.30) is the state equation expressed in the discrete time model, where $T' = \{t_0,\ldots t^f - \Delta t)\}$. Eqs. (3.31) and (3.32) express the constraints introduced by the definition of $\boldsymbol{x}$ and $\boldsymbol{y}$. The elements $\lambda_i^{cum}(t)$ and $\mu_i^{cum}(t)$ of $\boldsymbol{\lambda^{cum}}(t)$ and $\boldsymbol{\mu^{cum}}(t)$ are defined as

$$\lambda_i^{cum}(t) = \sum_{t'=t-\tau_i+1}^{t} \lambda_i(t'), \tag{3.41}$$

and

$$\mu_i^{cum}(t) = \sum_{t'=t-\tau_i'+1}^{t} \mu_i(t'). \tag{3.42}$$

Eq. (3.33) expresses the relation between the signal split and the departure flow, and Eq. (3.34) expresses the capacity constraints on the nodes, where $I(k) \subset E$ is the set of all links flowing into node $k$. Here, we define the constraint (3.33) by an equality, and we define constraint (3.34) by an inequality. This means that the obtained solution is a minimum green split without any de-facto red. The total inflow volume to each link can be limited by setting left side of Eq. (3.34) strictly smaller than 1. Finally, Eq. (3.35) expresses the constraints designed to prevent gridlock, given by Eq. (3.27).

Because the objective function and all constraints are linear, this optimization problem can be solved as a linear programming problem, given the demands $\boldsymbol{q} = [\boldsymbol{q}(t_0),\ldots,\boldsymbol{q}(t^f)]$ and the transition matrix $\boldsymbol{P} = [\boldsymbol{P}(t_0),\ldots,\boldsymbol{P}(t^f)]$ at all times, the number of vehicles $\boldsymbol{n}(t_0)$, the cumulative arrivals and departures $\boldsymbol{\lambda^{cum}}(t_0)$ and $\boldsymbol{\mu^{cum}}(t_0)$ at time $t_0$, and the link property $\boldsymbol{s} = [s_1,\ldots,s_K]$ and $\boldsymbol{n^{max}} = [n_1^{max},\ldots,n_K^{max}]$.

### 3.4.2 Model predictive framework

Z-control is embedded in a rolling-horizon scheme, with time unit $t_u$ and control time length $t_h$. At time $t_0$, the algorithm solves the optimization problem [min-TT], based on the transition matrix $\boldsymbol{P}(t_0)$, initial conditions $\boldsymbol{n}(t_0)$, $\boldsymbol{\lambda^{cum}}(t_0)$ and $\boldsymbol{\mu^{cum}}(t_0)$, and control time $T = \{t_0, \ldots, t_0 + t_h\}$. From the obtained $\boldsymbol{g}$, the values $\{\boldsymbol{g}(t_0), \ldots \boldsymbol{g}(t_0 + t_u)\}$ are applied to the system. The values $\{\boldsymbol{g}(t_0 + t_u + 1), \ldots \boldsymbol{g}(t_0 + t_h)\}$ are dismissed, and the optimization problem is solved at time $t_0 + t_u$ based on renewed transition matrix $\boldsymbol{P}(t_0 + t_u)$, initial conditions $\boldsymbol{n}(t_0 + t_u)$, $\boldsymbol{\lambda}^{cum}(t_0 + t_u)$ and $\boldsymbol{\mu}^{cum}(t_0 + t_u)$, and control time $T = \{t_0 + t_u, \ldots, t_0 + t_u + t_h\}$.

## 3.5 Numerical Experiments

In this section, we evaluate our proposed Z-control with numerical experiments. In many cases, when traffic signal control algorithms are evaluated by generic microscopic traffic simulators, the settings of cycle-lengths and offsets of traffic signals considerably influence the results. Oversaturated networks are especially sensitive to these parameters, making it difficult to evaluate algorithms with generic simulators. Thus, we used a simple traffic simulation that does not consider cycle-lengths and offsets, in order to evaluate Z-control that only controls the splits of traffic signals. Consequently, the evaluation considers only control splits for traffic signals.

In addition, because the drivers' choice of routes significantly influences traffic in oversaturated networks, we considered two scenarios pertaining to route choice for our evaluation.

### 3.5.1 Traffic simulation model

In our traffic simulation model, we considered discrete time and continuous traffic volume. For traffic flow, we considered non-linear constraints, such as physical-queue conditions and First-In First-Out (FIFO) conditions, in addition to the basic model stated in the previous section. The physical-queue conditions were as follows, using $X(t)$ and $Y(t)$ defined by Eqs. (3.11) and (3.12).

$$\forall i \in E : \mu_i(t) \leq X_i(t), \tag{3.43}$$

$$\forall i \in E : \lambda_i(t) \leq Y_i(t), \tag{3.44}$$

$$\forall i \in E : (\mu_i(t) - X_i(t)) \cdot \prod_{k \in O(i)} (\lambda_k(t) - Y_k(t)) = 0, \tag{3.45}$$

where $O(i) \subset E$ denotes the set of all links flowing out from link $i$. The FIFO condition for continuous time systems is expressed as follows(Kuwahara and Akamatsu, 2001):

$$\frac{\mu_i^d(t)}{\mu_i(t)} = \frac{\lambda_i^d(t - T_i(t))}{\lambda_i(t - T_i(t))}, \tag{3.46}$$

where $T_i(t)$ is the transit time of link $i$ for the traffic leaving from link $i$ at time $t$, and index $d$ expresses the traffic destination. This condition is stated as follows, when it is applied to discrete time systems:

$$\begin{aligned} &\forall t' \leq t \text{ such that } A(t' - \Delta t) < D(t) + \mu(t + \Delta t) \leq A(t') : \\ &\mu_i^d(t + \Delta t) = \max\left(J_i^d(t' - \Delta t) - D_i^d(t), 0\right) + \frac{\lambda_i^d(t')}{\lambda_i(t')} \mu_i(t + \Delta t). \end{aligned} \tag{3.47}$$

We used this condition in this numerical experiment. In addition, we assumed that the merging ratio from multiple links to a saturated link was equal. Given these

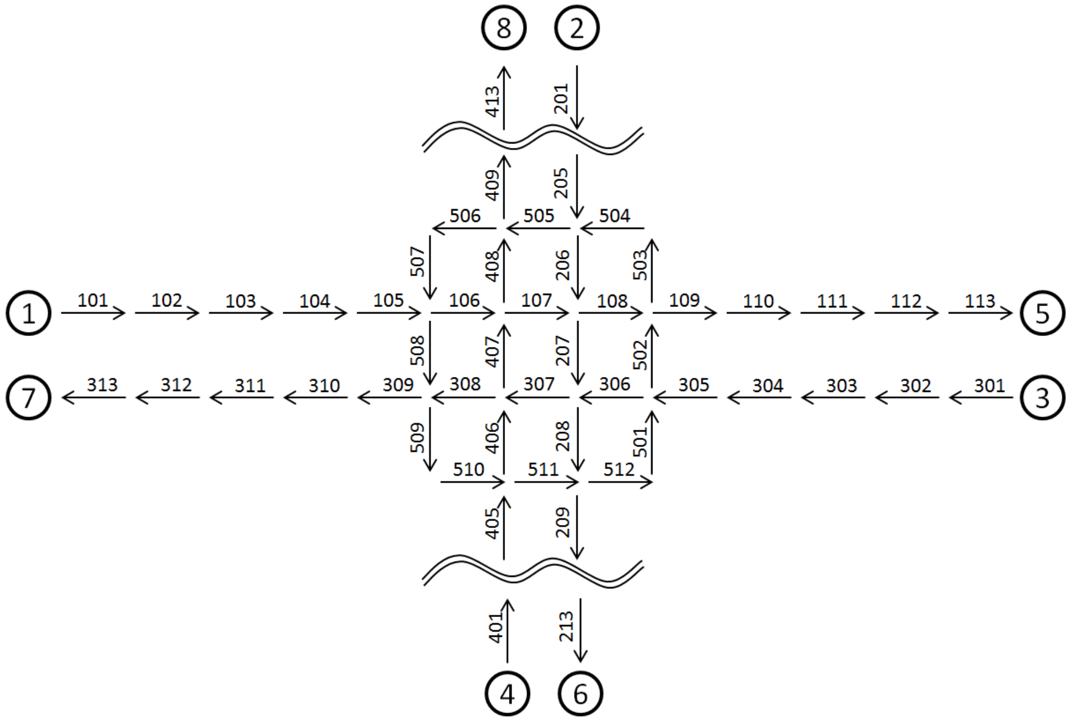



FIGURE 3.4: Sample network

TABLE 3.2: OD table, [veh./h]

| | | Destination | | | |
|---|---|---|---|---|---|
| | | (5) | (6) | (7) | (8) |
| Origin | (1) | 200 | 200 | 0 | 200 |
| | (2) | 100 | 100 | 400 | 0 |
| | (3) | 0 | 100 | 400 | 100 |
| | (4) | 100 | 0 | 400 | 100 |

conditions, $\mu_i(t)$ and $\lambda_k(t)$ can be uniquely determined. In the numerical study, we set the time step $\Delta t$ at 1 second.

### 3.5.2 Network and demand

A sample network for this numerical study is shown in Fig. 3.4. The network is symmetric, with 64 links and 56 nodes, including origin nodes 1 to 4 and destination nodes 5 to 8. The set of directed links 101 to 113 connects the origin node 1 to destination node 5, and links 201 to 213, 301 to 313, and 401 to 413, with connections between the origin node and destination node as well. In addition, the set of directed links 501 to 512 forms an one-way ring. All nodes except origin nodes and destination nodes are controlled by traffic signals. All links have the same parameters: length $l = 100\,(m)$, saturated flow rate $f^s = 1800\,(veh./h)$, and space capacity $n_{max} = 15\,(veh.)$. The transit time of the forward and backward waves is $\tau = 10\,(s)$ and $\tau' = 20\,(s)$, respectively. The demands at $t = 0$ are shown in Table 3.2. The traffic volume entering the network from each origin node 1 to 4 is equal, but their destinations are uneven, and more traffic is absorbed in node 7 than in other destination nodes. At $0 \leq t < 600$ and $1200 \leq t < 3600$, the demands are fixed, as shown in Table 3.2, and at $600 \le t < 1200$, there is 4.5 times as much demand as that shown in Table 3.2 loaded on the network.

### 3.5.3 Route choice model and scenarios

When drivers face extremely heavy traffic, their choice set of routes becomes more diverse (Bonsall, 2004). Some research (Hara and Kuwahara, 2015; Oyama et al., 2016) has modeled this kind of route choice behavior. In this chapter, we omit details from such models. Rather, we consider two extreme scenarios: a non-reactive scenario, where drivers do not react to the traffic in the network at all; and a reactive scenario, where drivers have perfect information regarding the traffic and react to it. Route choice behavior is classified into two types: pre-trip and en-route. In over-saturated networks, however, en-route route choices have more influence on the networks. In this study, then, we adopt a recursive logit model (Fosgerau et al., 2013), and calculate the transition probability from link $i \in E$ to link $j \in E$ for a driver with destination $d \in N$ at time $t$ as follows:

$$P^d(j|i) = \frac{e^{\frac{1}{\theta}\{v(j)+V^d(j)\}}}{\sum_{j' \in O(i)} e^{\frac{1}{\theta}\{v(j')+V^d(j')\}}}, \tag{3.48}$$

where $v(j)$ is the link cost of link $j$, and $V^d(j)$ is the expected utility of selecting link $j$ for the driver with destination $d$, which is defined as

$$V^d(j) = \boldsymbol{E}\big[\max_{j' \in O(j)} (v(j') + V^d(j') + \theta\epsilon(j'))\big], \tag{3.49}$$

where $\boldsymbol{E}[\cdot]$ is an expected value function, $\epsilon$ is distributed under a Gumbel distribution, and $\theta \ge 0$ is a scale parameter of the Gumbel distribution. In the non-reactive scenario, we define the link cost function as the normalized link length, i.e.,$v(i) = \alpha_i/\alpha_{norm}$, where $\alpha_{norm} = 1000(m)$. In the reactive scenario, by contrast, we define the link cost function as the travel time of the link, i.e., $v(i,t) = T_i(t - \Delta t)/T_{norm}$, where $T_i(t - \Delta t)$ is the observed travel time of link $i$ at time $t - \Delta t$, and $T_{norm} = 100(s)$. In both scenarios, we changed the scale parameter within the range $0.1 \le \theta \le 0.24$ to study the sensitivity of the algorithms to $\theta$.

### 3.5.4 Algorithm evaluation and benchmark

Our proposed Z-control requires as inputs the demand $\boldsymbol{q}(t)$ and link transition probability $\boldsymbol{P}(t)$ for $t_0 \le t \le t^f (= t_0 + t_h)$. Therefore, these must be estimated. We assumed that the demand is precisely known beforehand to the systems, i.e., $\forall t \in T$ : $\hat{\boldsymbol{q}}(t) = \boldsymbol{q}(t)$, where $\hat{\boldsymbol{q}}$(t) denotes the estimated value of $\boldsymbol{q}(t)$. For the link transition probability, we used the probability observed immediately before as the estimated value:

$$\forall t \in T : \hat{\boldsymbol{P}}(t) = \boldsymbol{P}(t_0 - \Delta t), \tag{3.50}$$

where $\hat{\boldsymbol{P}}$(t) denotes the estimated value of $\boldsymbol{P}(t)$. In the study, we set the time unit for model-predictive control to $t_u = 10(s)$. To avoid an extreme situation, we set a minimum green split $g_{min} = 0.1$. That is, if the value $g_i$ for link $i$ is smaller than $g_{min}$, we apply $g_{min}$ to the link.

As a benchmark, we adopted the QPC-B algorithm, referred to in Aboudolas et al. (2010), which also uses a model-predictive framework. The objective function of QPC-B is set to balance queue occupancy among all links in the network. In addition to this, we evaluated a variant of QPC-B with an objective function to minimize the total time spent. We refer to the former as relative queue balance (RQB)-control , and the latter as total time spent (TTS)-control. The objective function for RQB-control is stated as follows:

$$\min_{g} \sum_{t \in T} \sum_{i \in E} \frac{n_i^2(t)}{n_i^{max}}, \tag{3.51}$$

where $n_i(t)$ is the number of vehicles in link $i$ defined by Eq. (3.6). The objective function of TTS-control coincides with Z-control, and is defined by Eq. (3.28). The constraints for Z-control are defined by Eqs. (3.29)–(3.39). For RQB- and TTS-control, however, we removed the constraint (3.35) insofar as it focuses on closed-loops. Furthermore, rather than the link capacity constraints (3.31), (3.32), (3.36), and (3.37) which consider the time delay within links, we applied simple link capacity constraints expressed as

$$\forall i \in E, t \in T; 0 \le n_i(t) \le n_i^{max}. \tag{3.52}$$

Other settings were kept the same as Z-control, namely, the estimated demand, the link transition probability, the time unit for the model predictive system, and the minimum green split.

In addition to mechanism stated above, we considered a "no-control" situation, where all splits for all signals were fixed at 0.5.

Z-control and TTS-control were formalized as linear programming problems, and RQB-control was formalized as a quadratic programming problem. To solve these problems, we used Gurobi, a generic solver [2].

### 3.5.5 Evaluation criterion

As the criterion for the evaluation, we measured the total time spent by all users in the network during the evaluation time $0 \le t \le T_e$. That is,

$$\sum_{0 \le t \le T_e} \sum_{i \in E} (-n_i(t)). \tag{3.53}$$

In the numerical study, we set $T_e = 3600(s)$.

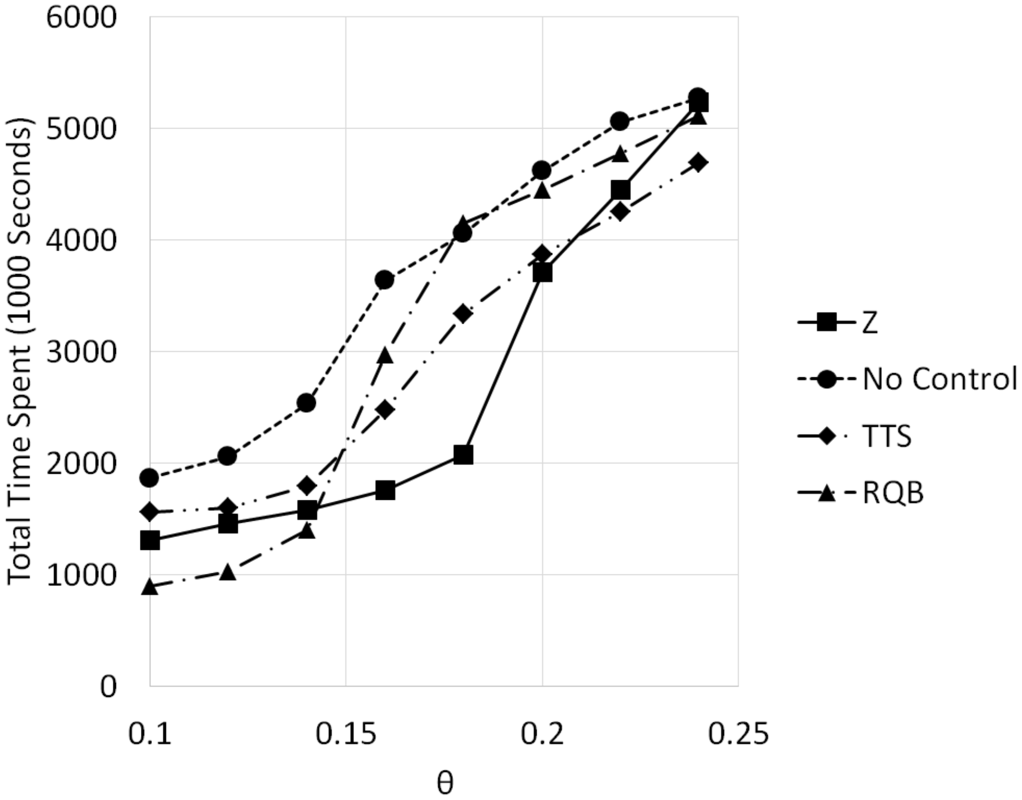
### 3.5.6 Results

In this section, we present the results for the non-reactive and reactive scenarios. The non-reactive scenario focuses on the basic behavior of the parameter $\tilde{Z}_{\mathcal{R}}$ against the congested situation of the network, in order to evaluate the basic idea of Z-control. Our evaluation of the reactive scenario was intended to observe the influence of drivers who react to the congested situation in a network generated with traffic control, in order to reveal any outstanding issues with Z-control.

#### Non-reactive scenario

First, we evaluated the algorithms in a non-reactive scenario. The total time spent is shown in Fig. 3.5. When $\theta \le 0.14$, RQB-control achieved the best performance, but its performance decreased as $\theta$ increased. On the other hand, Z-control performed well when $\theta \le 0.18$. Although Z-control had the same objective function as TTS-control, Z-control performed better than TTS-control in most cases.

[2] http://gurobi.com

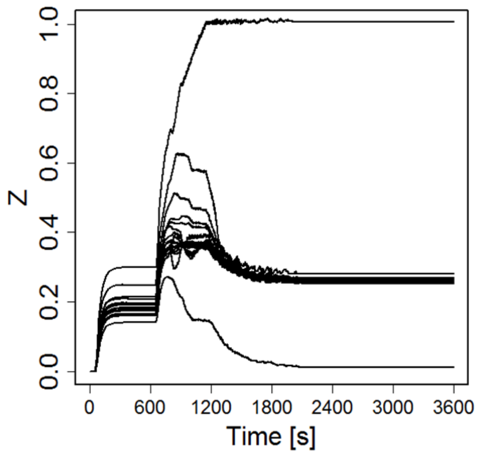



Z-control: Proposed control algorithm to minimize total time spent, considering time delay within links and constraints to prevent gridlock
TTS-control: Control algorithm to minimize total time spent
RQB-control: Control algorithm to balance the queue occupancy of links
No control: Fixed splits of all signals at 0.5

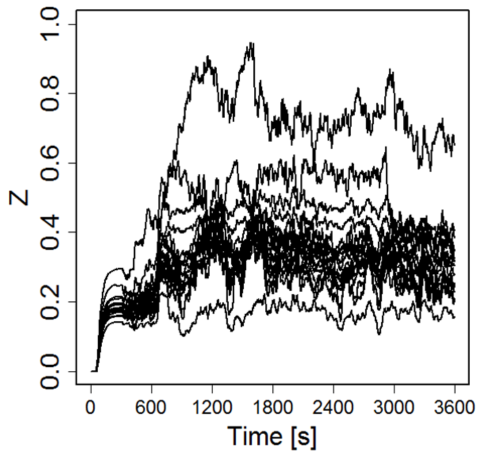
FIGURE 3.5: Total time spent (non-reactive scenario)

To see this in more detail, in Fig. 3.6, we show the value $\tilde{Z}_{\mathcal{R}}$ calculated in each algorithm when $\theta = 0.16$. Under this condition, "the narrow sense of gridlock phenomenon" is generated exclusively in the case of "no control." There were a total of 16 closed-loops in our sample network, and $\tilde{Z}_{\mathcal{R}}$ was calculated for each loop $\mathcal{R}$. In Fig. 3.6, all of these 16 values are superimposed. In the case of "no control," after time $t = 600$, when demand increased, $\tilde{Z}_{\mathcal{R}_0}$ for loop $\mathcal{R}_0$ increased rapidly and soon became $\tilde{Z}_{\mathcal{R}_0} = 1$. This loop $\mathcal{R}_0$ is formed by links $\{107, 207, 307, 407\}$ located at the center of the network. As $\tilde{Z}_{\mathcal{R}_0}$ approached 1, $\tilde{Z}_{\mathcal{R}}$ for other loops decreased. We can interpret this to mean that the traffic entering the network decreased as a result of the narrow sense of gridlock generated at loop $\mathcal{R}_0$. In the case of TTS-control and RQB-control, the narrow sense of gridlock was avoided, but $\tilde{Z}_{\mathcal{R}}$ remained high. In the case of RQB-control, $\tilde{Z}_{\mathcal{R}_0} = 1.0$ held, generating the broad sense of gridlock. In the case of Z-control, $\tilde{Z}_{\mathcal{R}}$ for all loops retained low values, contributing to a decrease in the total time spent.

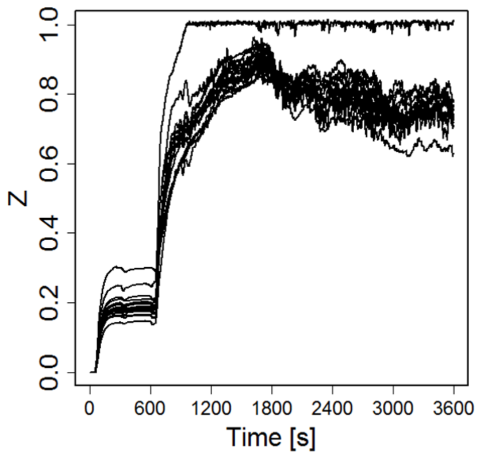
As shown above, the value of $\tilde{Z}_{\mathcal{R}}$ is closely connected to the gridlock phenomenon in the network. It can therefore be said that a control algorithm that adopts $\tilde{Z}_{\mathcal{R}}$ can prevent gridlock.



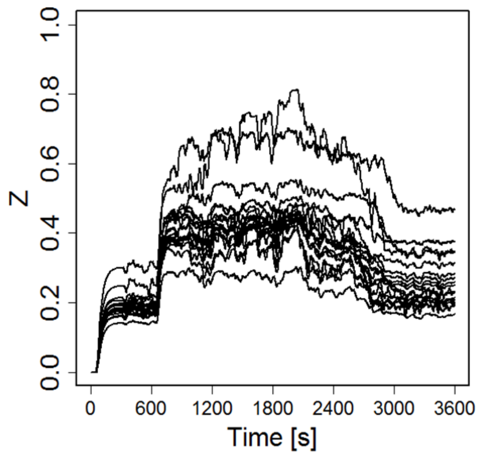

(a) No control




(b) TTS-control




(c) RQB-control




(d) Z-control

FIGURE 3.6: Value of $\tilde{Z}_{\mathcal{R}}$, when $\theta = 0.16$

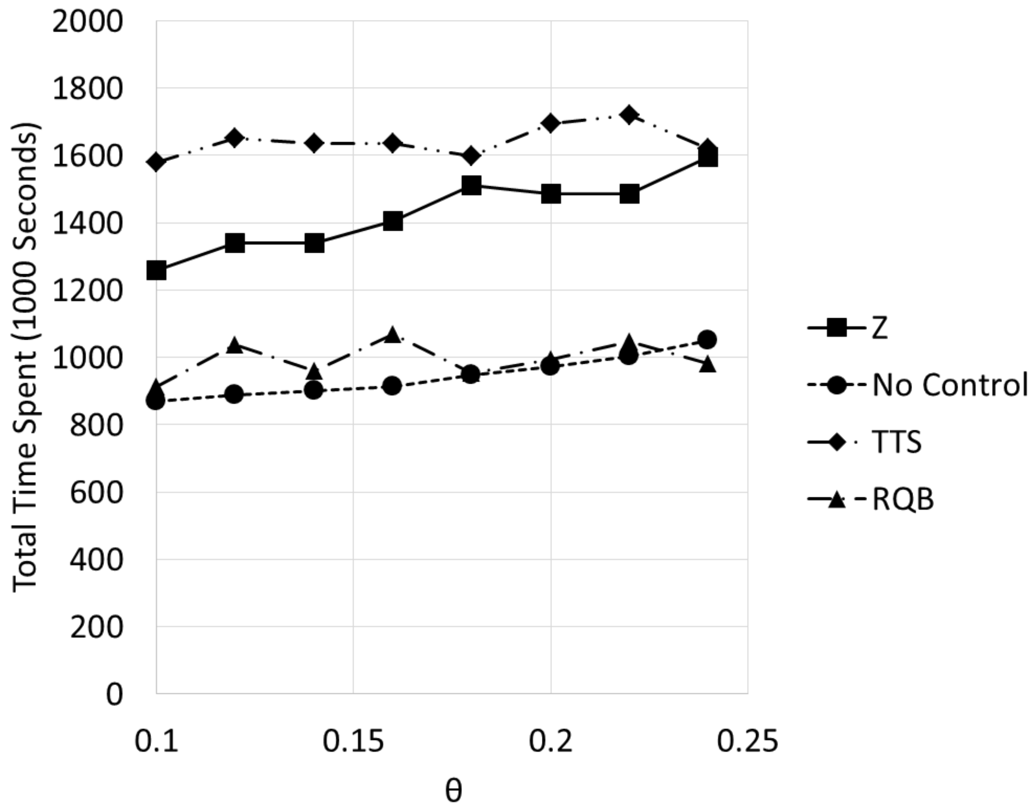



FIGURE 3.7: Total time spent (reactive scenario)

**Reactive scenario**

Next, we evaluated the algorithms in a reactive scenario. The total time spent is shown in Fig. 3.7. Unlike in the non-reactive scenario, "no control" performed best among all algorithms. The proposed Z-control performed better than TTS-control, but both algorithms performed much worse than "no control." Even RQB-control, the best of the three algorithms, performed worse than "no control." To better understand these results, we focused on the route choice probability of traffic flow from node 3 to node 7. When flowing out from link 305, vehicles in this traffic flow decided between going straight and flowing into link 306, or turning right and flowing into link 502. The probability of selecting link 306 is shown in Fig. 3.8. In this figure, the results for the non-reactive scenario are superimposed onto the simulated results for the case of Z-control and "no control" in the reactive scenario. After $t = 600$, when demand increased, the probability perturbed over the short term with "no control." However, this probability perturbs over the long term with Z-control, and to a greater extent. We next focused on two routes: Route A, comprising links $\{306, 307, 308\}$; and Route B, comprising links $\{502, 503, 504, 505, 506, 507, 508\}$. These routes are the main routes for vehicles selecting links 306 and 502, respectively. The travel time for both routes is shown in Fig. 3.9. In the case of "no control," the travel time of both routes maintained an equilibrium, as a result of the choice of the users. In the case of Z-control, by contrast, there was a large gap between the travel time of each route. In this study, we used Eq. (3.50) as the estimated link transition probability $\hat{\boldsymbol{P}}$. However, in oversaturated networks, the link transition probability changes quickly and significantly, owing to congested traffic. The gap between the

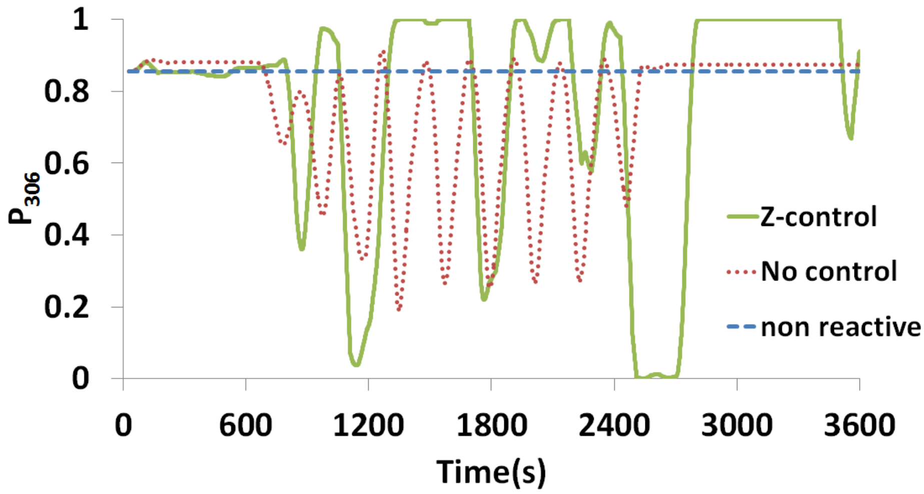
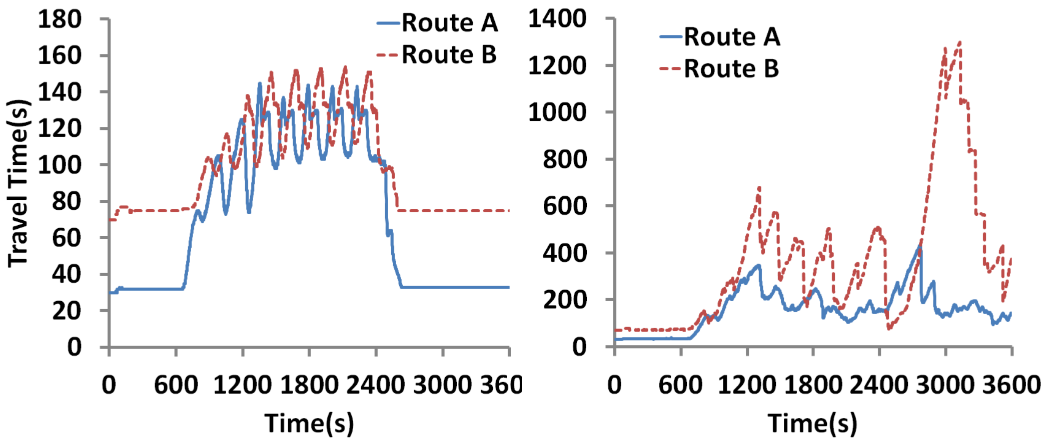



FIGURE 3.8: Transition probability from link 305 to 306




(A) No control

(B) Z-control

FIGURE 3.9: Travel time of Route A and Route B

actual transition probability $\boldsymbol{P}$ and its estimated value $\hat{\boldsymbol{P}}$ amplifies the perturbation, exacerbating the congestion. Comparing Figs. 3.5 and 3.7, the total time spent in the reactive scenario was much shorter than that in the non-reactive scenario under most conditions. The decision to find a detour away from the congestion considerably influences the network. Thus, traffic control algorithms for oversaturated networks should carefully consider this behavior.

### 3.5.7 Discussion of user behavior

As explained in the previous section, when users react to congestion, simple traffic control to prevent gridlock is ineffective. Assuming that the behavior of users is influenced by the state of the network, the link transition probability $\boldsymbol{P}$ is expressed as

$$\boldsymbol{P}(t + \Delta t) = \mathcal{P}(\boldsymbol{s}(t)), \tag{3.54}$$

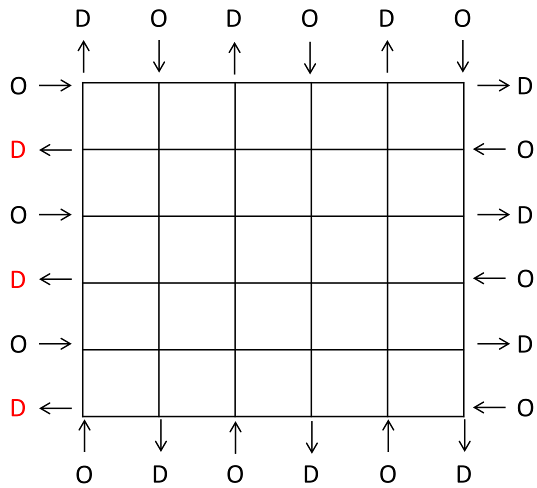



FIGURE 3.10: Grid network

where $\boldsymbol{s}(t)$ is a state variable for the network. For instance, assuming that users can obtain information regarding link occupancy and link travel time after time $t - t_p$, $\boldsymbol{s}(t)$ is a function of $\{ \boldsymbol{n}(t - t_p), \ldots, \boldsymbol{n}(t) \}$ and $\{ \boldsymbol{T}(t - t_p), \ldots, \boldsymbol{T}(t) \}$. Therefore, $\boldsymbol{P}$ can be expressed as

$$\boldsymbol{P}(t + \Delta t) = \mathcal{P}(\boldsymbol{n}(t - t_p), \ldots, \boldsymbol{n}(t), \boldsymbol{T}(t - t_p), \ldots, \boldsymbol{T}(t)), \tag{3.55}$$

where $\boldsymbol{T}(t) = [T_1(t), \ldots, T_K(t)]$.

Given the above, optimal traffic control considering user behavior can be formalized as

$$\max_{g} \quad \sum_{t \in T} \sum_{i \in E} (-n_i(t)), \tag{3.56}$$

$$\text{s.t.} \quad Eqs.(3.29) \text{ to } (3.39), \text{ and } (3.55).$$

The problem of obtaining the function $\mathcal{P}$ is a well-known traffic assignment problem. If users are assumed to have perfect information, this problem can be interpreted as the problem of obtaining the state of dynamic user equilibrium. Given this, the traffic control stated above can be interpreted as a Stackelberg game. This assumption is suitable for habitually oversaturated networks. Commonly, the function $\mathcal{P}$ is non-linear. Thus, heuristic algorithms are often adopted to solve this kind of problem(Ceylan and Bell, 2004; Mitsakis et al., 2011; Ukkusuri et al., 2013; Yang and Yagar, 1995). On the other hand, when treating oversaturated networks caused by unexpected situations such as accidents or disasters, it is natural to assume that the information users have is limited and imperfect. In this case, obtaining the function $\mathcal{P}$ is considered a traffic assignment problem that assumes that users react myopically to the traffic situation they face. For instance, Hato et al. (1999) and Oyama et al. (2016) analyze the route choice model of drivers in such situations.

## 3.6 Evaluation in a grid network

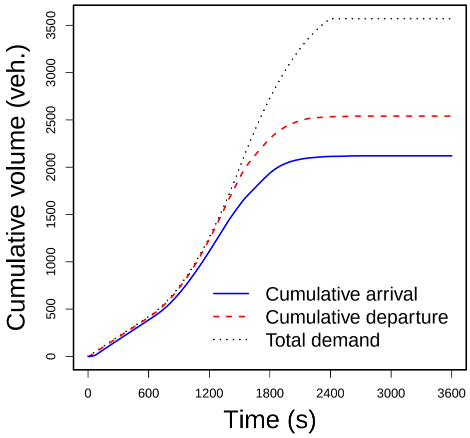
In this section, we evaluate our proposed Z-control in a grid network to analysis the condition where many closed-loops interfere with each other. The network is shown in Fig. 3.10. The link parameter is the same as that in Section 3.5. The network has

12 origin and 12 destination nodes, thus has 144 origin-destination (OD) pairs. We consider a scenario in which the demand heading to the left side of this network is heavier than that heading other directions. Specifically, we set the demand of $3.8[veh./h]$ to all OD pair except the three destination nodes in the left side, shown in red in Fig. 3.10. For these three destination nodes, we set four times much demand, that is $15.2[veh./h]$, from all origin nodes. We set the demands at $0 \leq t < 3600$, and at $600 \leq t < 1200$, we set twice much demand as stated above. We use the same traffic simulator stated in Section 3.5, and consider non-reactive and reactive scenarios as is similar to that section.

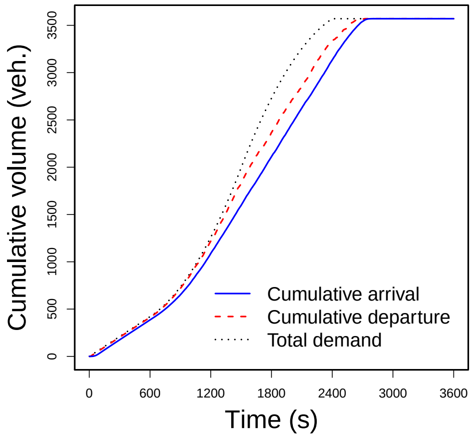
### 3.6.1 Non-reactive scenario

We first evaluate Z-control in the non-reactive scenario where users do not react to the congestion. The cumulative flow with and without Z-control is shown in Fig. 3.11. As is shown in Fig. 3.11(a), the flow volume falls into zero in the case without any control, which means that the gridlock phenomenon occurs. In contrast, as is shown in Fig. 3.11(b), the congestion are released in the case with Z-control. Fig. 3.12 shows the congestion in the network in these cases. The color in each link shows the density as is illustrated in Fig. 3.13

The value $\tilde{Z}_R$ for all loops in the network in these cases are shown in Fig. 3.14. Fig. 3.15 shows the distribution of $\tilde{Z}_R$ values. In the figures, solid lines shows the average of $\tilde{Z}_R$ values of all loops, dotted lines shows the standard deviations, and dashed lines shows the maximum and minimum values. As is shown in Fig. 3.14(a), in the case without any control, once the $\tilde{Z}_R$ value of a specific loop reaches to one, the loop falls into the gridlock. The $\tilde{Z}_R$ of other loops are slightly decreased after that and reaches a steady state where no traffic flows. The loop fallen in to the gridlock is observed to be a left bottom grid in Fig. 3.12(a-ii), and the steady state are shown in Fig. 3.12(a-iii). In contrast in the case with Z-control, as is shown in Fig. 3.15(b), $\tilde{Z}_R$ values of all links do not reach to one. The traffic is well-distributed as is shown in Fig. 3.12(b-ii). Here, we can see that the Z-control works well in the non-reactive scenario, as is similar to the results shown in Section 3.5.

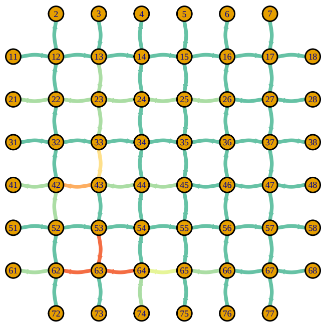



(a) No control

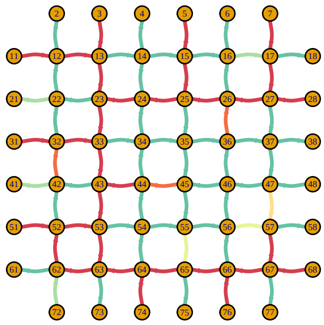



(b) Z-control

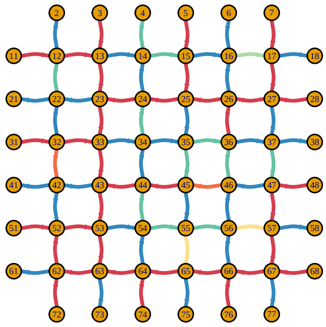
FIGURE 3.11: Cumulative flow in non-reactive scenario



(a-i) $t = 1200$

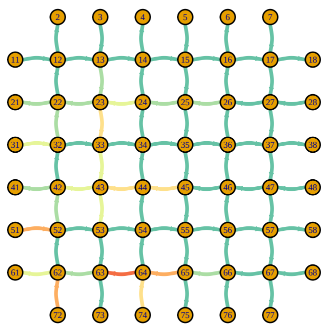


(a-ii) $t = 1800$

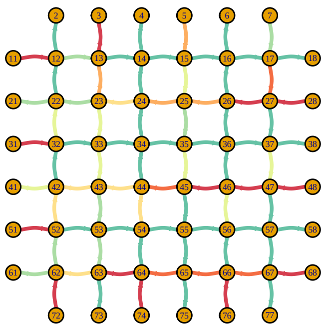


(a-iii) $t = 2700$

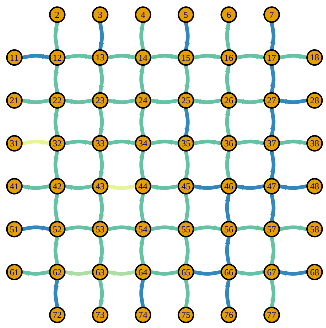
(a) No control



(b-i) $t = 1200$



(b-ii) $t = 1800$



(b-iii) $t = 2700$

(b) Z-control

FIGURE 3.12: Congestion in non-reactive scenario

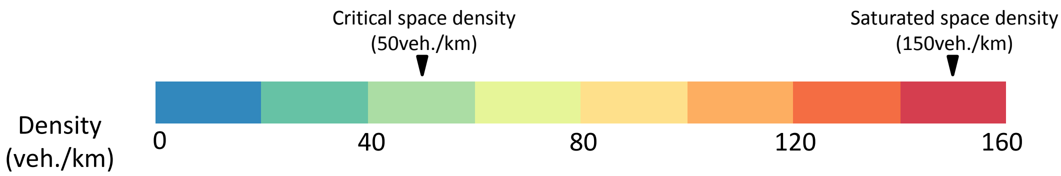
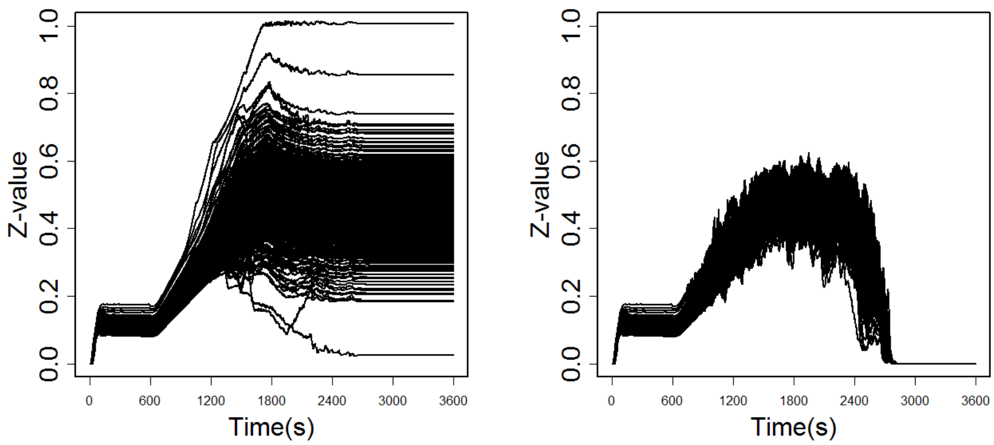
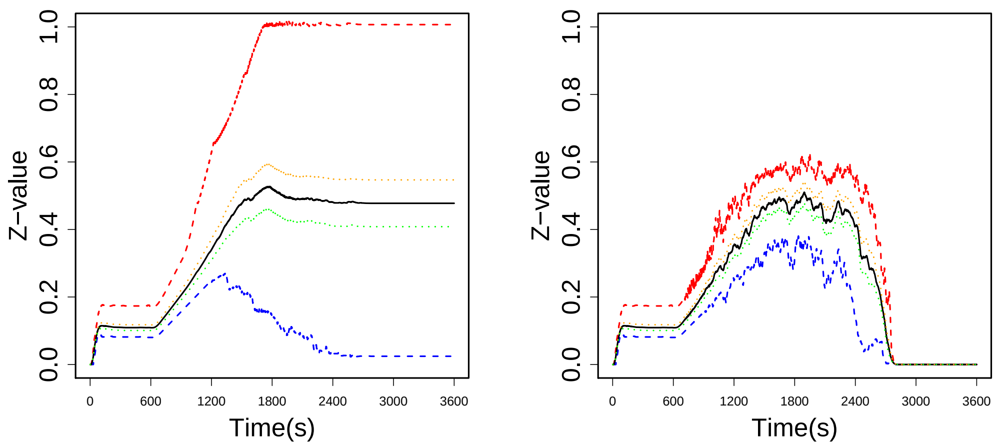



FIGURE 3.13: Color sample




(a) No control (b) Z-control

FIGURE 3.14: Z-values in non-reactive scenario




(a) No control (b) Z-control

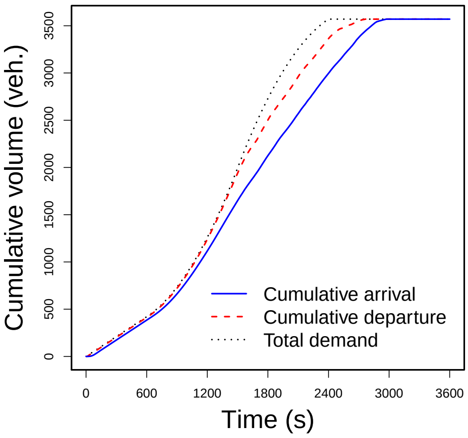
FIGURE 3.15: Average and distribution of Z-values in non-reactive scenario

### 3.6.2 Reactive scenario

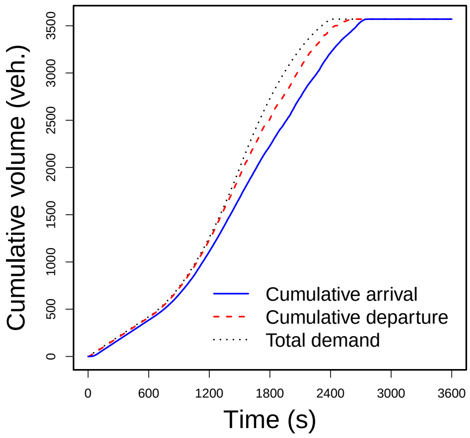
We then evaluate Z-control in the reactive scenario where users react to the congestion. Here we consider two cases with respect to the time-discounted rate $\beta$ of users, which represents the extent of the myopic decision of users. In irregular conditions, such as under disasters, it is known that the users' route choice decisions become more myopic and the time-discounted rate becomes lower than in a regular conditions (Oyama and Hato, 2017). We evaluate two cases, $\beta = 0.8$ which represents the regular conditions and $\beta = 0.2$ which represents the irregular conditions.

#### Regular conditions

The cumulative flow in the regular condition where $\beta = 0.8$ is shown in Fig. 3.16. Unlike in the non-reactive scenario, gridlock phenomenon does not occur both with and without Z-control. Even so, it can be seen that the congestion is slightly mitigated by the Z-control.

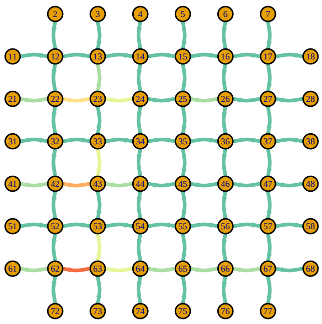



(a) No control

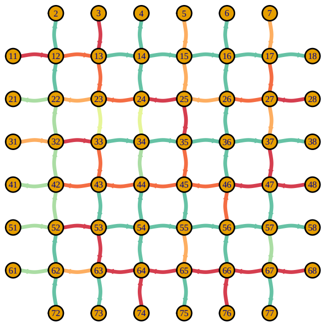



(b) Z-control

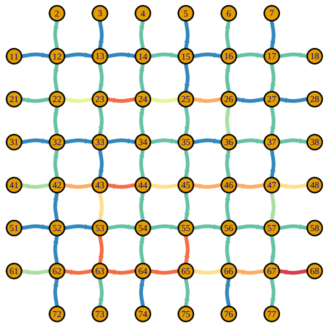
FIGURE 3.16: Cumulative flow in reactive scenario ($\beta = 0.8$)



(a-i) $t = 1200$

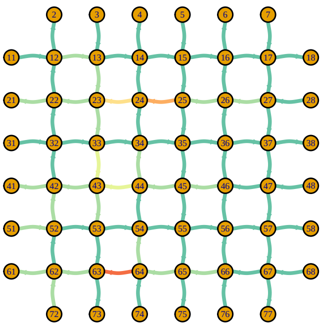


(a-ii) $t = 1800$

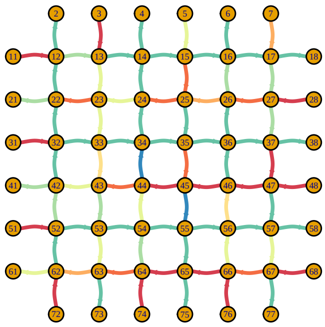


(a-iii) $t = 2700$

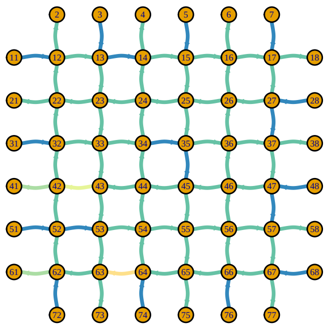
(a) No control



(b-i) $t = 1200$



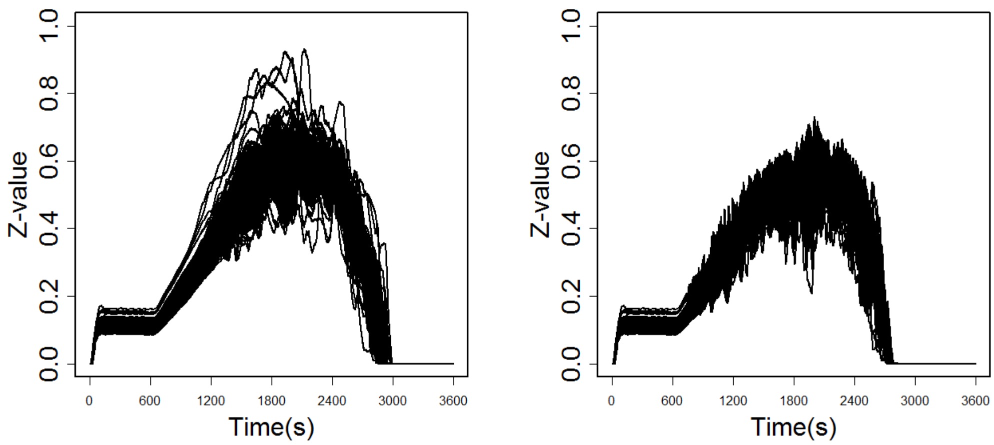
(b-ii) $t = 1800$



(b-iii) $t = 2700$

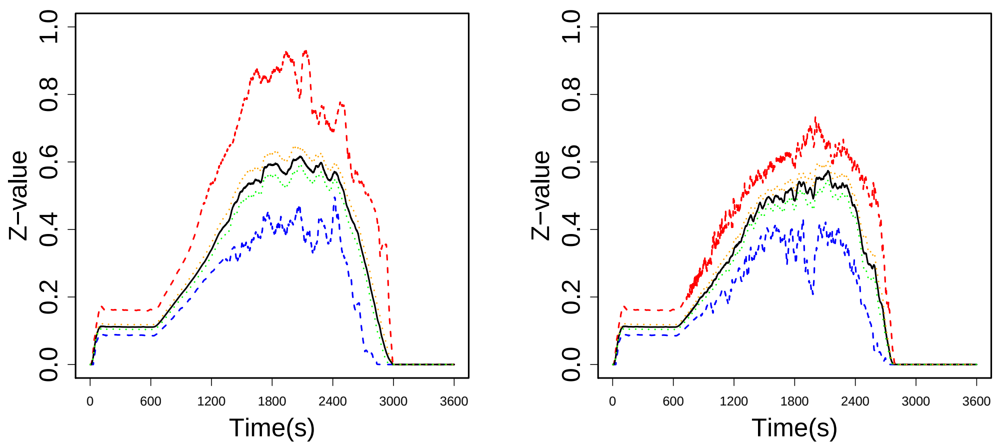
(b) Z-control

FIGURE 3.17: Congestion in reactive scenario ($\beta = 0.8$)




(a) No control (b) Z-control

FIGURE 3.18: Z-values in reactive scenario ($\beta = 0.8$)



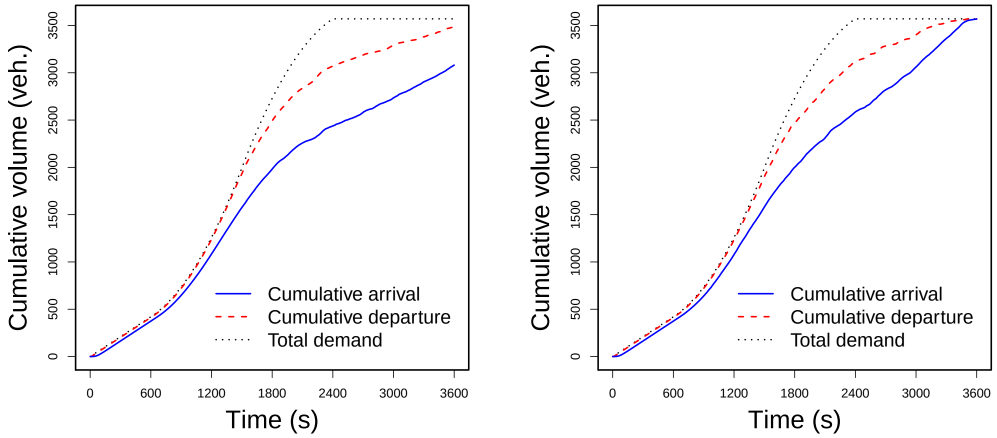

(a) No control (b) Z-control

FIGURE 3.19: Average and distribution of Z-values in reactive scenario ($\beta = 0.8$)

The congestion and the value $\tilde{Z}_R$ for all loops in the network in these cases are shown in Fig. 3.17 and Fig. 3.18. In addition, Fig. 3.19 shows the distribution of $\tilde{Z}_R$ values. In the figures, solid lines shows the average of $\tilde{Z}_R$ values of all loops, dotted lines shows the standard deviations, and dashed lines shows the maximum and minimum values. From Fig. 3.17(a-ii) and (b-ii), it can be seen that the congestion in the left side where the demands concentrate are well-distributed by the Z-control. It results in an early mitigation of the congestion, as is shown in Fig. 3.17(a-iii) and (b-iii). Fig. 3.18 and Fig. 3.19 also shows that the congestion is well-distributed to all loops in the network by the Z-control.

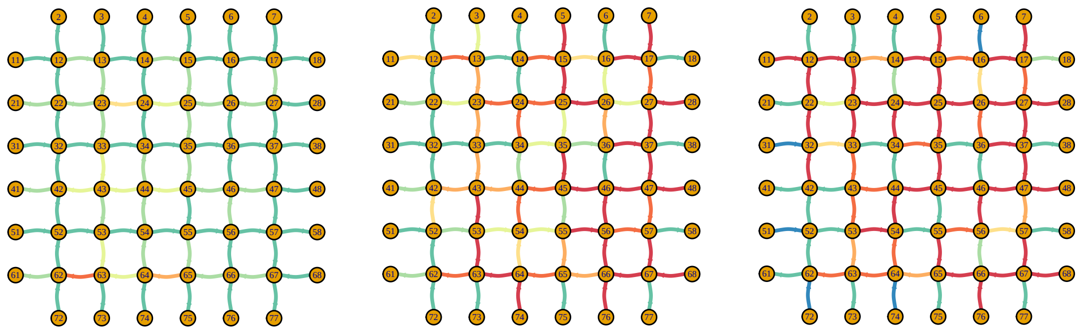
**Irregular conditions**

The cumulative flow in the irregular condition where $\beta = 0.2$ is shown in Fig. 3.20. Comparing to the regular condition, the congestion is heaver because of the users' myopic behavior. In this case, the Z-control works much better than the regular case.



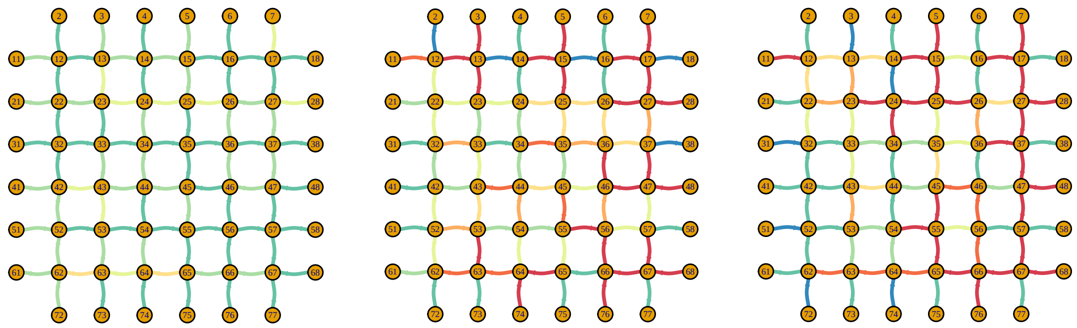

(a) No control (b) Z-control

FIGURE 3.20: Cumulative flow in reactive scenario ($\beta = 0.2$)



(a-i) $t = 1200$ (a-ii) $t = 1800$ (a-iii) $t = 2700$

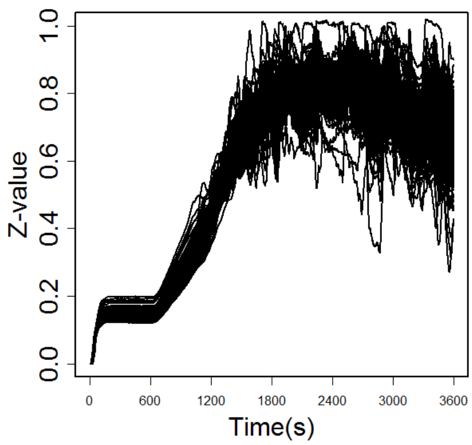
(a) No control



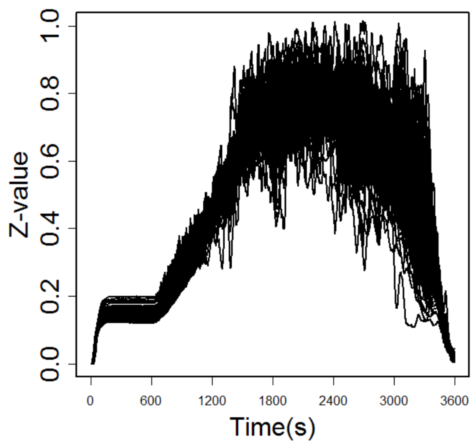
(b-i) $t = 1200$ (b-ii) $t = 1800$ (b-iii) $t = 2700$

(b) Z-control

FIGURE 3.21: Congestion in reactive scenario ($\beta = 0.2$)

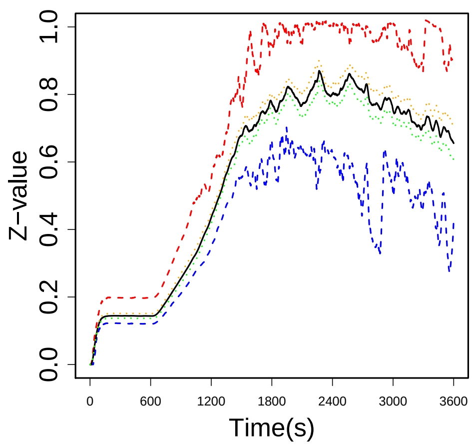



(a) No control

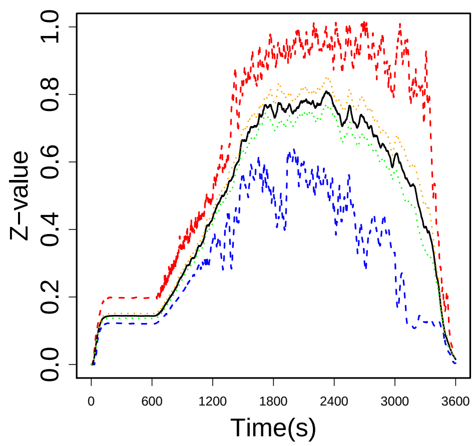



(b) Z-control

FIGURE 3.22: Z-values in reactive scenario ($\beta = 0.2$)




(a) No control




(b) Z-control

FIGURE 3.23: Average and distribution of Z-values in reactive scenario ($\beta = 0.2$)

The congestion and the value $\tilde{Z}_R$ for all loops in the network in these cases are shown in Fig. 3.21 and Fig. 3.22. In addition, Fig. 3.23 shows the distribution of $\tilde{Z}_R$ values. In the figures, solid lines shows the average of $\tilde{Z}_R$ values of all loops, dotted lines shows the standard deviations, and dashed lines shows the maximum and minimum values. Because of the myopic behavior of users, the congestion are distributed over the whole network in both cases with and without Z-control. However, as is shown in Fig. 3.20(b), the Z-control can well-distribute the congestion while keeping the total flow volume in the network. Thus, the Z-control works well in irregular conditions.

## 3.7 Conclusions

In this chapter, we proposed a traffic control algorithm for oversaturated networks with closed-loop structures. We introduced state variables that consider the time-delay property to address the dynamic behavior of vehicle queues. Using these variables, we described the process of gridlock, using the parameters $B_{\mathcal{R},i}$ and $Z_{\mathcal{R},i}$ to characterize the phenomenon. Then, we introduced the parameter $\tilde{Z_{\mathcal{R}}}$, which is an approximation of $Z_{\mathcal{R},i}$, and proposed a traffic control algorithm called Z-control that uses this parameter. The algorithm was formalized as a linear programming problem, such that it can be applied to large networks, provided that the demand $\boldsymbol{q}$ and link transition probability $\boldsymbol{P}$ are set properly.

We evaluated the algorithm with numerical experiments, considering two scenarios: a non-reactive scenario, where drivers do not react to traffic conditions in the network; and a reactive scenario where drivers are assumed to have perfect information about the traffic conditions in the network and react to it. In the non-reactive scenario, we showed that our proposed parameter $\tilde{Z_{\mathcal{R}}}$ is effective at preventing gridlock. Indeed, Z-control outperformed other benchmark algorithms. We also showed that the Z-control works well in the reactive scenario with a complicated network where many closed-loops interfere with each other. Especially, it works much better in irregular conditions where the users' behavior become myopic.

However, it still has many points to be improved. In this study, we adopted model-predictive control, using the observed link transition probability $\boldsymbol{P}(t_0 - \Delta t)$ as a estimate value $\hat{\boldsymbol{P}}(t)$. The difference between $\hat{\boldsymbol{P}}(t)$ and the actual link transition probability $\boldsymbol{P}(t)$ resulted in the algorithm's poor performance. To improve this situation, the drivers' reaction to the congested network should be considered in future estimations of the link transition probability. This can be formalized as a bi-level optimal problem, also known as the combined traffic control and traffic assignment problem. When treating habitually oversaturated networks, this bi-level problem can be formalized as a Stackelberg game. When treating unexpected oversaturated networks, moreover, myopic driving behavior should be considered in the bi-level problem. In both cases, the problem becomes very complicated, suggesting the need for heuristic algorithms or traffic simulations. The basic idea for our proposed Z-control can be easily implemented in such systems, and in future research, we will evaluate the performance of such traffic control systems.

Furthermore, our results can inform policy pertaining to the introduction of automated vehicles (AVs) in large cities. A discussion of AV decision-making regarding route-choice has not been well developed. Our results suggest that the central control of an AVs' route choice is important to avoid situations such as those observed in our "reactive scenario," where drivers were assumed to have perfect information about the traffic conditions in the network and react to it. Indeed, this scenario appropriately describes a world with "selfish" AV behavior. Even if the authority to choose the route remains with the AV, the route-choice model for AVs will probably be much more simple and rational than that of human drivers. As such, traffic control that considers the route choice of AVs can be established with rational models to ease traffic congestion.

# Chapter 4

# Search-based optimization of activity-based dynamic trip matching in sparse ridesharing markets

In this chapter, we present an algorithm for the ridesharing service operated in suburb areas. Specifically, we formalize a dynamic trip matching problem between customers and drivers and propose a search-based solution algorithm for the problem.

The content of this chapter is prepared to be submitted to *Transportation Research Part C: Emerging Technologies*.

We explore the dynamic trip matching problem between customers and drivers in sparse ride sharing markets. In sparse ride sharing markets like in suburbs, common trip-based myopic matching algorithm may force customers to wait an extremely long time. To avoid this, we take an approach of activity-based travel analysis that can describe a series of trips and activities of each customer and thus can consider the connectivity of multiple trips. We formulate the matching problem as a dynamic optimization problem that aims to minimize time-discounted total travel cost under capacity constraints of vehicles and space–time constraints of users and drivers. We then focus on the settings where the service operator knows appearance probabilities of users. In such cases, the optimal policy cannot be realized by common mixed integer programming (MIP) solvers. We propose a search-based solution algorithm that uses the data structure represented by the zero-suppressed binary decision diagram (ZDD). In numerical studies, we show that our proposed algorithm performs better than both static and dynamic benchmarks that can be solved by common MIP solvers, and obtains near-optimal solutions.

## 4.1 Introduction

With the widespread of on-demand ridesharing services, it becomes a significant challenge for the service operators to obtain a dynamic algorithm that sequentially and efficiently operates their traffic resources. For on-demand bus services, there exist many works (Horn, 2002; Sayarshad and Chow, 2015) that extend the static framework of Dial-a-ride problem (DARP)(Cordeau and Laporte, 2007; Ho et al., 2018) or Pickup and Delivery Problem with Time Windows (PDPTW) (Dumas et al., 1991) to dynamic conditions. For ride-sharing services, other works (Alonso-Mora et al., 2017; Kamar and Horvitz, 2009; Pelzer et al., 2015; Santi et al., 2014) present algorithms that myopically match drivers and customers to minimize the sum of waiting and traveling time. However, these works consider trip-based demand and thus do not consider the correlation of a series of trips made by customers.

Such trip-based algorithms cause fatal problems in a sparse market like suburbs where there are not many drivers and customers at the same time and location. For example, users who success booking their outbound trips from their home may fail to book their return trips because of the shortage of drivers. To avoid such situations, a framework is required that guarantees complete each customer's itinerary with multiple sub-trips within the customer's space–time constraints. A trivial solution to this problem is an advance reservation system by which customers can book and fix all sub-trips in advance. However, the customers often change minds during their series of itineraries, for example, they may find another place to visit or may go back their home earlier. The advance reservation systems cannot support such demands and cause opportunity losses of customers, especially in sparse markets. A flexible operation is desired that dynamically adapt to the flexible demands while guaranteeing the completion of all customers' trips even in the worst case.

To consider a series of trip by customers, the framework of activity-based model (Axhausen and Gärling, 1992; Kitamura et al., 1996) are well-studied. In this framework, individuals are considered as utility-maximizer within a space–time prism constraints (Hägerstrand, 1970) derived from their home locations and mandatory activities. The trip is considered as a derived demand of customers' activities. Thus the quantity of trips made by all customers depends on provided transportation services. Kang et al. (2013) formulate this as activity-based network design problems that consist of network design problem and household activity pattern problem (HAPP) (Recker, 1995). Liu et al. (2018) formulate it as an optimization problem with considering the HAPP as constraints, and present an efficient solution algorithm. However, these works consider static conditions in which all information about household members are fixed and given in advance, and do not consider dynamic conditions.

In this chapter, we propose a dynamic matching algorithm to operate a ridesharing service efficiently in sparse ridesharing markets in suburb areas. Our proposed algorithm adaptively minimizes time-discounted total travel cost of all drivers and users with guaranteeing the completion of all accepted users' trips.

To aim this, we make the following contributions:

- We first formulate the dynamic trip assignment problem in sparse ridesharing markets that aims to minimize time-discounted total travel cost with considering space–time constraints of both users and drivers.
- We propose a search-based algorithm to solve this problem. The algorithm provides *anytime* feasible solution owing to the full-search using a data structure called zero-suppressed binary decision diagram (ZDD) (Minato, 1993, 2001).
- We numerically show that our proposed algorithm realizes efficient matching between drivers and users. The algorithm makes sequential but non-myopic decisions and thus makes appropriate judges whether it should accept or reject users.

The remainder of this chapter is organized as follows. In Section 4.2, we state the ridesharing services for members only that we consider in this chapter, showing an example that illustrates the necessity of flexible booking systems. In Section 4.3, we state the mathematical model for the flexible booking systems that aims to minimize time-discounted total travel cost. In Section 4.4, we propose a solution algorithm

TABLE 4.1: Notations

| | |
|---|---|
| Time horizon | |
| $T = \{1, \ldots, \bar{T}\}$ | Discretized time horizon |
| $\beta$ | Time-discounted rate |
| Network | |
| $\mathcal{N}$ | Set of nodes in the road network |
| $\mathcal{E}$ | Set of links in the road network |
| $\mathrm{I}(n) \subset \mathcal{E}$ | Set of inflow links to node $n \in \mathcal{N}$ |
| $\mathrm{O}(n) \subset \mathcal{E}$ | Set of outflow links from node $n \in \mathcal{N}$ |
| $\tau_e \in \mathbb{N}$ | travel time of edge $e \in \mathcal{E}$ |
| Individuals | |
| $I = I^D \cup I^U$ | Set of individuals |
| $I^D$ | Set of drivers |
| $I^U$ | Set of users |
| $I^U_{=t} \subseteq I^U$ | Set of users whose active time-window starts at time $t \in T$ |
| $\bar{I}^U \subseteq I^U$ | Set of users who do not require trips |
| Parameters of drivers and users | |
| Travel costs | |
| $C_i^M \in \mathbb{R}$ | Travel cost of individual $i \in I$ per a unit time |
| $C_i^A \in \mathbb{R}_{<0}$ | Value (i.e., negative travel cost) of user $i \in I^U$ completing all mandatory activities in $A_i$ |
| $C_{i,n}^S \in \mathbb{R}$ | Travel cost of individual $i \in I$ staying at node $n \in \mathcal{N}$ |
| $\tilde{C}_{i,n}^S \in \mathbb{R}$ | Estimated travel cost of individual $i \in I$ staying at node $n \in \mathcal{N}$ in advance |
| $\boldsymbol{C}_i$ | Set of travel costs of individual $i$, including $C_i^M$, $C_{i,n}^S$ for all $n \in \mathcal{N}$, and $C_i^A$ |
| $\rho$ | Correlation coefficients between the true travel cost $C_{i,n}^S$ and the estimated travel cost $\tilde{C}_{i,n}^S$ |
| Space–time constraints | |
| $H_i \in \mathcal{N}$ | Home location, i.e., origin and destination node, of individual $i \in I$ |
| $T_i = [t_i^b, t_i^e]$ | Active time duration of individual $i \in I$ |
| $L_i$ | Set of feasible trip-paths for individual $i \in I$ |
| Parameters of drivers | |
| $Q_i \in \mathbb{N}$ | The capacity, i.e., the number of users, that driver $i \in I^D$ can serve |
| Parameters of users | |
| $J_i$ | Number of mandatory activities for user $i \in I^U$ |
| $A_i = \{a_{i,1}, \ldots, a_{i,J_i}\}$ | Set of mandatory activities for user $i \in I^U$ |
| $n_{a_{i,j}} \in \mathcal{N}$ | The location of the mandatory activity $a_{i,j} \in A_i$ |
| $T_{a_{i,j}} = [t^b_{a_{i,j}}, t^e_{a_{i,j}}]$ | The time-window of the mandatory activity $a_{i,j} \in A_i$ |
| $p_i$ | The appearance probability of user $i \in I^U$ |
| Decision variables | |
| $\delta_i \in \{0,1\}$ | Binary indicator variable whether the request of individual $i \in I$ makes her trip |
| $\eta_{i,e,t} \in \{0,1\}$ | Binary indicator variable whether individual $i \in I$ flows into link $e \in \mathcal{E}$ at time $t$ |
| $\theta_{i,n,t} \in \{0,1\}$ | Binary indicator variable whether individual $i \in I$ stays at node $n \in \mathcal{N}$ at time $t$ |

that can efficiently solve the problem. In Section 4.5, we numerically demonstrate the performance of our proposed algorithm. Finally, in Section 4.6, we conclude this chapter and discuss the potential direction of the extension of this work.

## 4.2 Problem statement

In this section, we show an overview of the ridesharing service in suburbs that we discuss in this chapter. The notations introduced in this chapter is shown in Table 4.1. The service is offered by a service operator which match users with drivers. Both users and drivers have to register as a member of the service in advance. Users can require a series of trips, among which some mandatory activities are included, at any time. For each users' request, the operator can decide whether it accepts or rejects the request. From the operator's point of view, it is desirable that multiple users share their trips to reduce the travel cost. Thus, the operator tries to bundle their trips, as far as it satisfies their constraints. It can also reject users' request if the operation cost is too much.

To consider the heterogeneity of users and drivers, we introduce three types of travel costs. The first is the travel cost per unit time during moving denoted by $C_i^M \in$

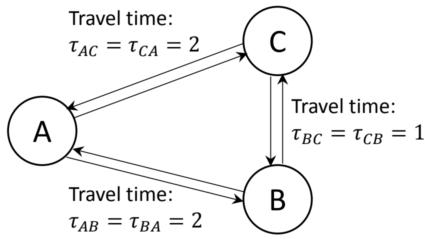



FIGURE 4.1: Sample network and the link travel time

$\mathbb{R}$, which represents the value of time of each individual. The second is the value, i.e., negative cost, of completing all mandatory activities denoted by $C_i^A \in \mathbb{R}_{<0}$, which is considered only for users, not for drivers. The third is the cost of staying at each node denoted by $C_{i,n}^S \in \mathbb{R}$. For drivers, it represents the cost of waiting. For users, it represents the value of optional activities executed on each node. We call the negative cost for users staying at each node as VoS (*Value of staying*). We assume that the true VoS is not revealed before the user actually visits the location.

The operator has perfect knowledge of drivers. It also knows a set of registered users $I^U$ and the home locations $H_i$ and active time duration $T_i$ of each user $i \in I^U$. It has perfect knowledge of mandatory activities $A_i$, for example going to the school or office, but does not know about their optional activities, for example going shopping. Through the usage history, it knows the appearance probability $p_i$ of each user but does not know whether a certain user actually appears, until the beginning time $t_i^b$ of the active time duration $T_i$. It also has information about the travel cost of each user. However, it does not know the user's true VoS $C_{i,n}^S$ until the user $i$ visits the node $n$. Instead, the estimated VoS denoted by $\tilde{C}_{i,n}^S$ is given in advance.

In the following part, we show an example that illustrates the case we focus on this chapter.

### 4.2.1 Motivating example

#### Network

A road network $(\mathcal{N}, \mathcal{E})$ with three nodes and six edges is given as shown in Fig. 4.1. The travel time $\tau_e$ of each edge $e \in \mathcal{E}$ is given in the figure. The travel time of each edge is assumed to be constant.

#### Space–time constraints of drivers and users

Assume a service with one driver $X$ and two users 1 and 2. The parameters are given in Table 4.2 and 4.3. Both drivers and users have their own home location and active time-window. Each user $i$ has $J_i$ mandatory activities, and the location and time-window of each mandatory activities $a_{i,j}$ are given. The time–space paths of drivers and users are spatially and temporarily constrained (Hägerstrand, 1970) by the home location, active time-window and mandatory activities. This setting is consistent with existed works (Kang et al., 2013; Liu et al., 2018; Recker, 1995).

#### Appearance probability and travel costs

The cost of travel $C_i^M$ and cost of waiting $C_{i,n}^S$ for driver $X$ is shown in Table 4.2. As is shown in the table, it cost 100 per unit time while the Driver X moves and costs 50

TABLE 4.2: Parameters of the Driver X

| | Driver X | |
|---|---|---|
| Home location | Node B | $H_X = B$ |
| Active time-window | [0,12] | $T_X = [0, 12]$ |
| Service capacity | 2 | $Q_X = 2$ |
| Cost of travel | 100 | $C_X^M = 100$ |
| Cost of waiting | 50 | $C_{X,A}^S = 50, C_{X,B}^S = 0, C_{X,C}^S = 50$ |

TABLE 4.3: Parameters of the User 1 and 2

| | | User 1 | User2 | |
|---|---|---|---|---|
| Home location | | Node A | Node A | $H_1 = H_2 = A$ |
| Active time-window | | [0,10] | [1,10] | $T_1 = [0, 10], T_2 = [1, 10]$ |
| Appearance probability | | 0.9 | 0.5 | $p_1 = 0.9, p_2 = 0.5$ |
| Number of mandatory activities | | 1 | 1 | $J_1 = J_2 = 1$ |
| Mandatory activity | Location | Node B | Node C | $n_{a_{1,1}} = B, n_{a_{2,1}} = C$ |
| | Time-window | [5,6] | [4,5] | $[t^b_{a_{1,1}}, t^e_{a_{1,1}}] = [5, 6], [t^b_{a_{2,1}}, t^e_{a_{2,1}}] = [4, 5]$ |
| | Value | 400 | 600 | $C_1^A = -400, C_2^A = -600$ |
| Cost of travel | | 5 | 5 | $C_1^M = 20, C_2^M = 20$ |
| Value of staying (VoS) at | Node A | 0 | 0 | $C_{1,A}^S = 0, C_{2,A}^S = 0$ |
| *Priori information* | | | | |
| Estimated VoS at | Node B | 50 | 40 | $\tilde{C}_{1,B}^S = -50, \tilde{C}_{2,B}^S = -40$ |
| Estimated VoS at | Node C | 40 | 50 | $\tilde{C}_{1,C}^S = -40, \tilde{C}_{2,C}^S = -50$ |
| *True values revealing after the visit* | | | | |
| True VoS at | Node B | 26 | 51 | $C_{1,B}^S = -26, C_{2,B}^S = -51$ |
| True VoS at | Node C | 62 | 48 | $C_{1,C}^S = -62, C_{2,C}^S = -48$ |

per unit time if the Driver X waits at Node A or C. Without loss of generality, we set the cost is zero if drivers spend time on their own home location.

The appearance probability and travel costs of users are shown in Table 4.3. We set the VoS at the users' home location is zero without loss of generality. Because the true VoS $C_{i,n}^S$ is not revealed before the user actually visit the location, the estimated VoS $\tilde{C}_{i,n}^S$ is given as shown in Table 4.3.

### 4.2.2 Matching between users and drivers

Assuming that both User 1 and 2 appear, the optimal matching is shown in Fig. 4.2(a). The driver X firstly takes both users to Node C, and then takes User 1 to Node B. After a time-unit, the driver takes User 1 back to Node C and takes both users to back to Node A after the staying at Node C for a time-unit. By these itineraries, the travel costs of User 1, 2 and Driver X are $-458$, $-828$, and 1050 and thus the total travel cost is $-236$. However, this result is not trivially obtained by common matching algorithms.

For example, the first-come first-serve (FCFS) algorithm is often adopted, by which each user's itinerary is fixed greedily to minimize her cost in order of the appearance. By this algorithm, the itinerary of User 1 with $T_1 = [0, 10]$ is fixed at $t = 0$ based on her estimated VoS. Driver X is assigned for the round trip of User 1, as shown in Fig. 4.2(b). So, the user 2 with $T_2 = [1, 10]$ appearing at $t = 1$ is rejected because no drivers can serve her. The travel costs of User 1 and Driver X are $-484$ and 800, and the total travel cost is 316.

Another common algorithm is that tries to calculate optimal operation using estimated VoS known in advance, assuming that all users will appear. We call this

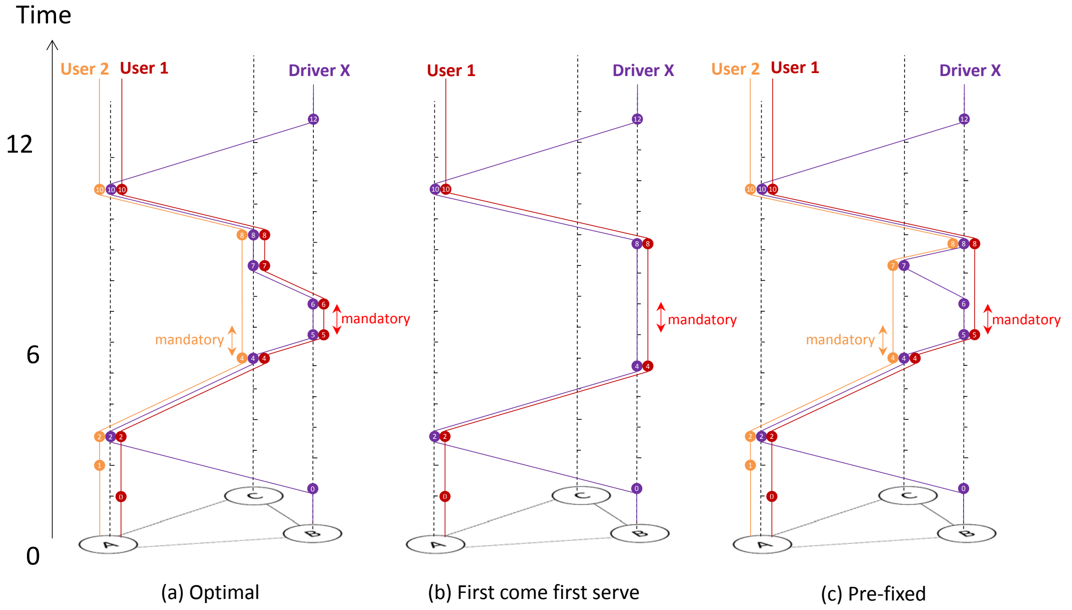



FIGURE 4.2: Trip plans for the case study

algorithm as *pre-fixed*. The itineraries calculated by this algorithm are shown in Fig. 4.2(c). The travel costs of User 1, 2 and Driver X are $-453$, $-761$, and 1100, and the total travel cost is $-114$. All of them are higher than the optimal allocation case shown in Fig. 4.2(a), which is caused by the gap between the estimated and true VoS. It can be seen that this algorithm may not achieve optimal matching even if the information about the appearance of users is exactly given.

Now we consider an algorithm that can achieve the optimal matching in this case. The algorithm should make decisions sequentially considering newly revealed information about users' true VoS at each time. Unlike the FCFS algorithm, it should have some redundancy to accept later users. For example in this case, the operator can accept User 1's request at $t = 0$ without fixing the itinerary but with guaranteeing that User 1 can complete her itinerary within her space–time constraints. The operator present the itinerary shown in Fig. 4.2(b) if User 2 does not appears at $t = 1$ and present the itinerary shown in Fig. 4.2(b) based on the estimated VoS if User 2 appears. Note that both of them satisfies the space–time constraints of User 1. Now assume that the User 2 appears and the operator adopt the itineraries show in Fig. 4.2(c). In this case, User 1 is firstly taken to Node C and the true VoS at Node C is revealed. Then, the User 1 visits Node B and the true VoS at Node B is also revealed. As shown in Table 4.3, for User 1, the true VoS at Node C is higher than the estimated one while the true VoS at Node B is lower. At this point, User 1 notices that it is better to spend time at Node C instead staying at Node B. If the algorithm can re-calculate the itineraries at this point, the operator can offer a better services. In this case, indeed, the itineraries shown in Fig. 4.2(a) is better not only for User 1 but also for User 2 and Driver X.

The case stated above illustrates two essential problems observed in sparse ridesharing markets. The first is the uncertainty of the appearance of users that often causes the inefficient traffic operation. The second is users' mind-changing in the middle of their trips, which should be considered to improve the quality of the service. To address these problems, we propose an algorithm by which, the itineraries are re-calculated at each time with guaranteeing all accepted users' space–time constraints *at any time*. We call booking systems using such algorithms as *floating booking systems* (Hayakawa and Hato, 2018b).

## 4.3 Model of the floating booking system

In this section, we show the dynamic optimization model that achieves the concept of the floating booking system mentioned in the previous section.

### 4.3.1 Benchmark offline model

We first show the benchmark offline model that minimizes the total travel cost (TTC), given the perfect knowledge about both drivers and users.

**Offline optimal**

$$\min_{\boldsymbol{\delta},\boldsymbol{\eta},\boldsymbol{\theta}} TTC = \sum_{i\in I}(\delta_i(C_i^A + \sum_{t\in T_i}(\sum_{e\in\mathcal{E}} \eta_{i,e,t}\tau_e C_i^M + \sum_{n\in\mathcal{N}} \theta_{i,n,t}C_{i,n}^S))) \tag{4.1}$$

s.t:

$$\forall e \in \mathcal{E}, \forall t \in T : \sum_{i\in I^U} \eta_{i,e,t} \leq \sum_{i\in I^D} Q_i \eta_{i,e,t}, \tag{4.2}$$

$$\forall n \in \mathcal{N}, \forall i \in I, \forall t \in T : (\theta_{i,n,t} + \sum_{e\in \mathrm{O}(n)} \eta_{i,e,t}) - (\theta_{i,n,t-1} + \sum_{e\in \mathrm{I}(n)} \eta_{i,e,t-\tau_e}) = \begin{cases} \delta_i & \text{if } t = t_i^b \\ -\delta_i & \text{if } t = t_i^e \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

$$\forall n \in \mathcal{N}, \forall i \in I : \theta_{i,n,\bar{T}} = 0, \tag{4.4}$$

$$\forall e \in \mathcal{E}, \forall i \in I : \eta_{i,e,\bar{T}} = 0, \tag{4.5}$$

$$\forall i \in I^U, \forall a_{i,j} \in A_i : \sum_{t=[t^b_{a_{i,j}}, t^e_{a_{i,j}}-1]} \theta_{i,n_{a_{i,j}},t} \geq \delta_i \tag{4.6}$$

$$\forall i \in I^D, \delta_i = 1 \tag{4.7}$$

$$\forall i \in \bar{I}^U, \delta_i = 0 \tag{4.8}$$

The decision variables $\delta_i \in (0,1)$, $\eta_{i,e,t} \in (0,1)$ and $\theta_{i,n,t} \in (0,1)$ represent the acceptance of individual $i$'s request, flowing into link $e$ at time $t$, and staying at node $n$ at time $t$, respectively. For all drivers $i \in I^D$, $\delta_i = 1$. Eq. 4.2 expresses the capacity constraints. Eq. 4.3 expresses the flow conservation. Eq. 4.6 expresses the constraints such that users have to spend at least one time step to complete their mandatory activities. Eq. 4.7 shows that all drivers appears and Eq. 4.8 expresses the user who do not require the trip. The travel costs of users who does not appear and are rejected are zero because the cost of staying their own home-location is set to zero, as stated in Section 4.2.1.

### 4.3.2 Dynamic model for floating booking systems

To establish algorithms that re-calculate users' itineraries at each time, we extend the static model to dynamic settings. The objective function at any time $t$ is given as:

$$\min_{\boldsymbol{\delta}_{\geq t},\boldsymbol{\eta}_{\geq t},\boldsymbol{\theta}_{\geq t}} TTC_t = \mathbb{E}\left[\sum_{t'=t}^{\bar{T}_i-1} \beta^{t-t'}(\sum_{i\in I_{=t}} \delta_i C_i^A + \sum_{i\in I}\delta_i(\sum_{e\in\mathcal{E}} \eta_{i,e,t'}\tau_e C_i^M + \sum_{n\in\mathcal{N}} \theta_{i,n,t'}C_{i,n}^S))\right], \tag{4.9}$$

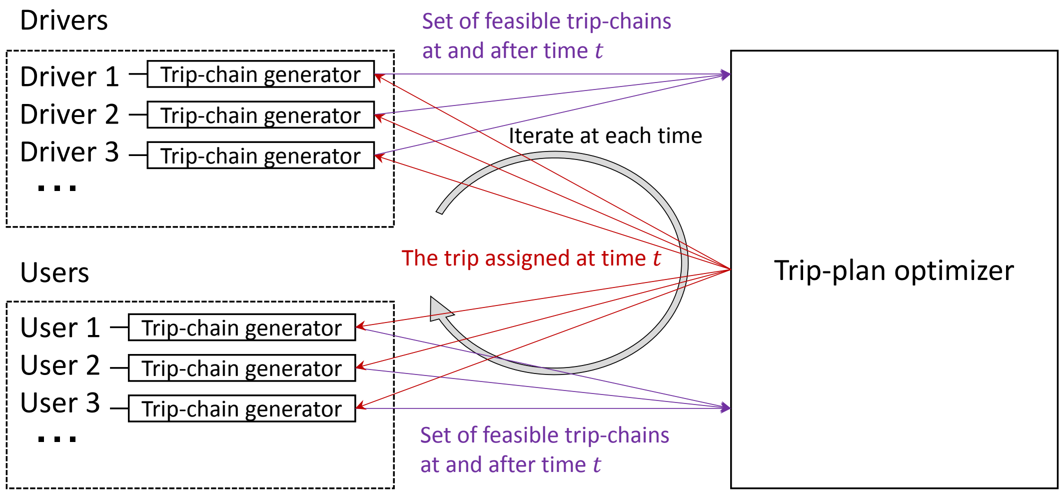



FIGURE 4.3: Framework of the proposed algorithms

where $\delta_{\geq t}$, $\boldsymbol{\eta}_{\geq t}$ and $\boldsymbol{\theta}_{\geq t}$ represents the decisions at and after the current time $t$. $\mathbb{E}[\cdot]$ is the expectation with respect to the given future demand model, specifically the appearance probability of users. The time-discount rate $\beta$ is assumed to be common over all users and drivers.

By Bellman's principle of optimality (Bellman, 1957), this objective function is rewritten as a recursive form, as follows.

**Dynamic model**

$$\min_{\boldsymbol{\delta}_t,\boldsymbol{\eta}_t,\boldsymbol{\theta}_t} TTC_t = \sum_{i\in I}\delta_i\Big(\sum_{e\in\mathcal{E}}\eta_{i,e,t'}\tau_e C_i^M + \sum_{n\in\mathcal{N}}\theta_{i,n,t'}C_{i,n}^S\Big) + \sum_{i\in I_{=t}}\delta_i C_i^A + \min_{\boldsymbol{\delta}_{>t},\boldsymbol{\eta}_{>t},\boldsymbol{\theta}_{>t}} TTC_{t+1}, \tag{4.10}$$

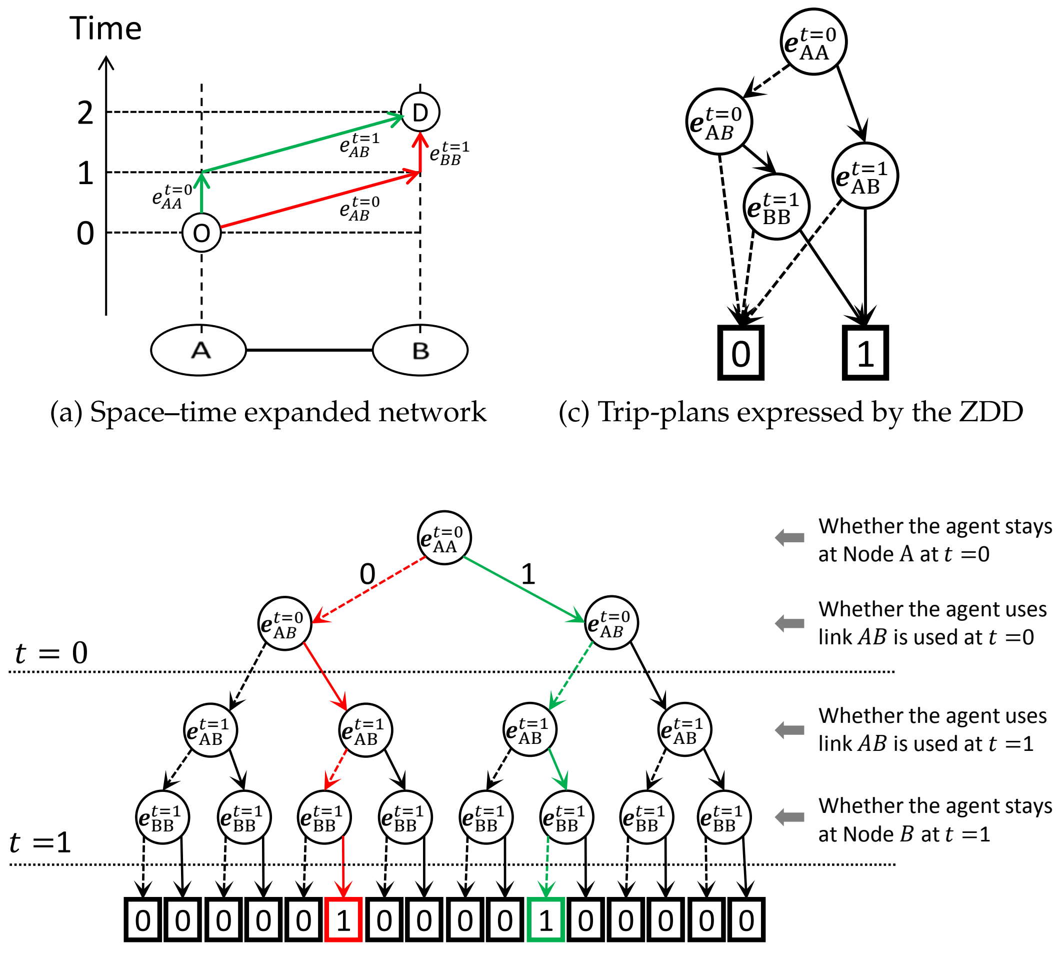
s.t: Eqs. 4.2 to 4.8.

While the objective function is time-decomposed, the constraints are not time-decomposed. Thus the capacity constraints and space–time constraints in future time $t' > t$ should be taken consideration into the decisions at each time $t$.

Here, we focus on the constraints about the users' appearance given by Eq. 4.8. If the information about users' appearance is given exactly, the model is a common mixed integer programming (MIP) model and thus the solution can be obtained by common MIP solvers. However, if there is uncertainty on the users' appearance, the model cannot be solved trivially. We propose a solution algorithm for this problem in the following section.

## 4.4 Solution algorithms

In this section, we show search-based algorithms that obtains *anytime* feasible solution of the dynamic model shown in the previous section. Our proposed algorithm consists of two parts, *trip-chain generator* and *trip plan optimizer*, as shown in Fig. 4.3. *Trip-chain generator* is set to all users and drivers. At each time, it enumerates a set of itineraries that satisfies the space–time constraints of each individual. *Trip plan optimizer* decides the optimal policy at each time, collecting the set of itineraries of all individuals enumerated by their *trip-chain generators*. The combinations of itineraries




(a) Space–time expanded network

(c) Trip-plans expressed by the ZDD

(b) Trip-plans expressed by the binary decision tree

FIGURE 4.4: Trip-plan enumeration using the ZDD

of all individuals are presented with considering capacity constraints. In the following parts, we show the algorithm in detail.

### 4.4.1 Trip-chain generator

*Trip-chain generator* set at each individual outputs a set of feasible itineraries $L_i$ that satisfies the space–time constraints. It considers the home-location and active time-window of individuals, and also considers mandatory activities for users. The feasible itineraries are greatly lessen by the space–time prism constraints (Hägerstrand, 1970) derived from this information and are expressed as the sparse graph in an space–time expanded network.

Let me show an example. We consider a simple network with only 2 nodes A and B, and a link with $\tau_{AB} = 1$. Given an individual that start from Node A at $t = 0$ and arrive at Node B at $t = 2$, the space–time expanded network is shown as in Fig. 4.4(a). In this figure, actions of the individual are displayed. For example, the actions $e^{t=0}_{AA}$ and $e^{t=0}_{AB}$ represents that the staying at Node A and the moving along the Link AB at $t = 0$, respectively. A series of actions for the individual is shown in Fig. 4.4(b), using a binary decision diagram. The arrows express the binary-decisions, specifically, the solid arrow represents that the action is taken while the

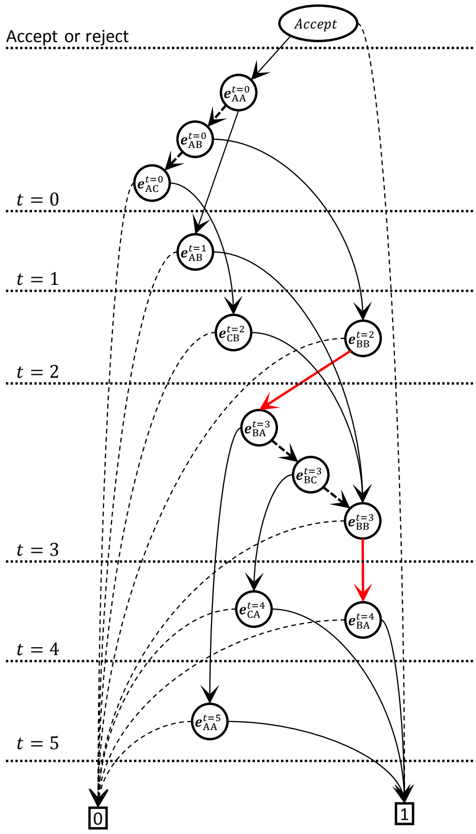


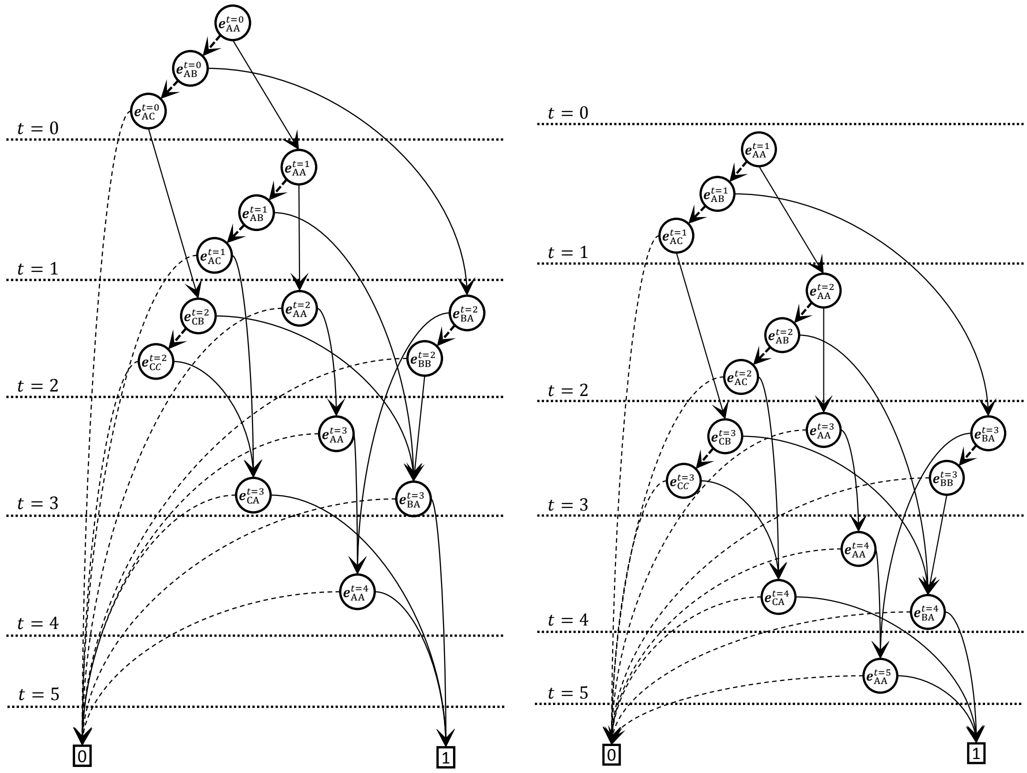

FIGURE 4.5: A set of trip-chains for user 3

dashed arrow represents that the action is not taken. The numbers in $\{0, 1\}$ shown in the bottom express whether the plan is feasible or not. For example, an itinerary displayed in red in the figure means that the individual moves from Node A to Node B at $t = 0$ and stays at Node B at $t = 1$. In this case, two itineraries shown in red and green in Fig. 4.4(a) and (b) are feasible. As is shown in Fig. 4.4(b), feasible paths with spatial and temporal connection are very sparse in the binary-decision tree. By using the ZDD data structure (Minato, 1993), such a sparse graph is represented in a compact form as shown in Fig. 4.4(c).

Let me show an another example. Assume the road network shown in Fig. 4.1 and an User 3 shown in Table 4.4. Using the ZDD data structure, the feasible itineraries of this user is expressed as shown in Fig. 4.5. The top node in the figure represents the decision whether this user is accepted or rejected. Two red arrows in the figure represents the staying at Node B during time $[2, 4]$, which is the time-window of the mandatory activity. All paths connected '1' at the bottom pass at least one red arrow, which means that the mandatory activity is completed. The itineraries of Driver Y

TABLE 4.4: Parameters of user 3, driver X and driver Y

| | Home location | Active time-window | Mandatory activities | |
|---|---|---|---|---|
| | | | Location | Time-window |
| User 3 | Node A | [0,6] | Node B | [2,4] |
| Driver Y | Node A | [0,5] | | |
| Driver Z | Node A | [1,6] | | |




(b) Driver Z

(a) Driver Y

FIGURE 4.6: A set of trip-chains for Drivers Y and Z

and Z, whose parameters are given in Table 4.4, are similarly enumerated as shown in Fig. 4.6.

Thus, the *trip-chain generator* of each individual enumerates all feasible paths that satisfies space–time constraints given by Eq. 4.3 to 4.6. Using the ZDD data structure, the enumerated paths are kept in a compact form.

### 4.4.2 Trip-plan optimizer

At each time, *Trip-plan optimizer* calculates the optimal action for all individuals based on the model shown in Section 4.3.2. For each individual, all itineraries satisfying space–time constraints given by Eqs. 4.3 to 4.6 are enumerated by the *trip-chain generator*. *Trip-plan optimizer* select the combination of these itineraries that minimizes time-discounted travel cost given by Eq. 4.10 with satisfying capacity constraints given by Eq. 4.2 and appearance constraints given by Eqs. 4.7 and 4.8. The user appearance constraints Eq. 4.8 are considered by taking multiple scenario with respect to future users. The overall algorithm is shown in Algorithm 1.

The algorithm has a variable $\mathcal{Z}$ that represents the combination of feasible itineraries by all individuals. All combination of feasible itineraries enumerated by all drivers are saved using ZDD data structure (Line 1). Then the itineraries of each user are combined to this with checking the capacity constraints (Line 2 to 5). Thus all feasible combination of itineraries are enumerated that satisfy both space–time

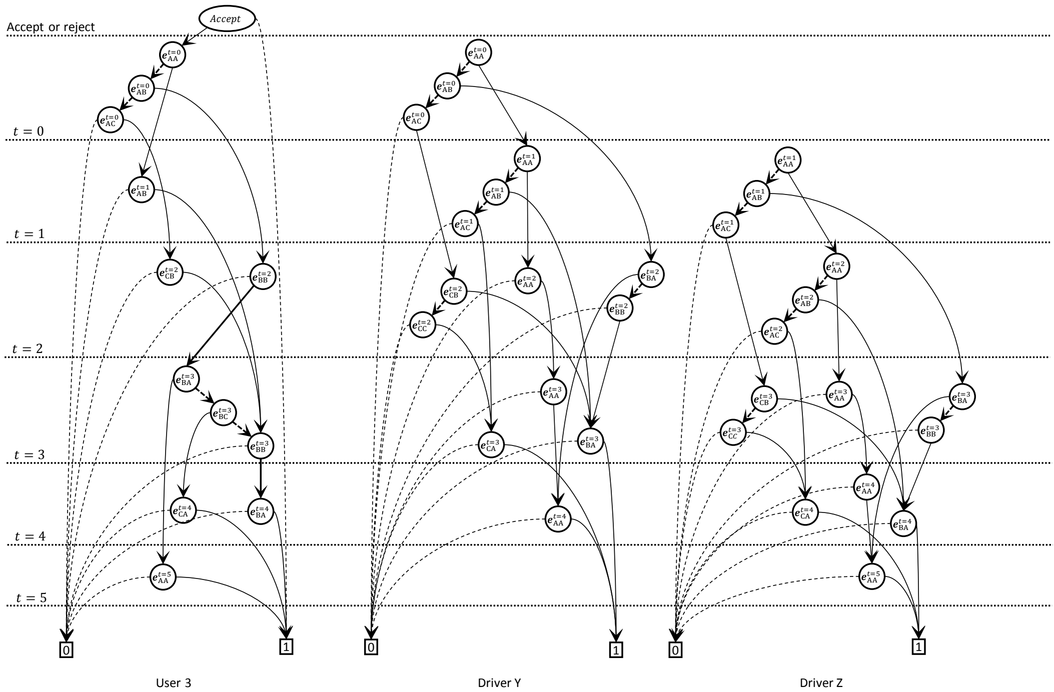
**Algorithm 1** Algorithm

1: $\mathcal{Z} \leftarrow \Pi_{i \in I^D} L_i$
2: **for all** $i \in I^U$ **do**
3: $\quad \mathcal{Z} \leftarrow \mathcal{Z} \times L_i$
4: $\quad \mathcal{Z} \leftarrow \mathcal{Z} \setminus \textsc{CapacityViolationSet}(\mathcal{Z})$ {Capacity constraints}
5: **end for**
6: **for** $t = 1$ to $\bar{T} - 1$ **do**
7: $\quad \mathcal{Z}^w \leftarrow \text{WeightCosts}(\mathcal{Z}, \boldsymbol{C}_t)$
8: $\quad \mathcal{Z}_t \leftarrow \text{FeasibleActions}(\mathcal{Z})$ {Feasible combination of actions at the current time}
9: $\quad \forall \{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\} \in \mathcal{Z}_t : Score(\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\}) \leftarrow 0$
10: $\quad$ **for** $m = 1$ to $M$ **do**
11: $\quad\quad \bar{I}^m \leftarrow \text{SampleAbsentUsers}(I^U_{>t})$ {Sample a set of absent users}
12: $\quad\quad$ **for** $\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\} \in \{\Delta_t, H_t, \Theta_t\}$ **do**
13: $\quad\quad\quad Score(\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\}) \leftarrow Score(\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\}) + TTC_t(\mathcal{Z}^w, \{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\}, \bar{I}^m)$
14: $\quad\quad$ **end for**
15: $\quad$ **end for**
16: $\quad \{\boldsymbol{\delta}_t, \boldsymbol{\eta}_t, \boldsymbol{\theta}_t\} = \text{argmin}_{\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\} \in \mathcal{Z}_t} Score(\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\})$ {Optimal policy}
17: $\quad \mathcal{Z} \leftarrow \textsc{Update}(\mathcal{Z}, \{\boldsymbol{\delta}_t, \boldsymbol{\eta}_t, \boldsymbol{\theta}_t\})$
18: **end for**

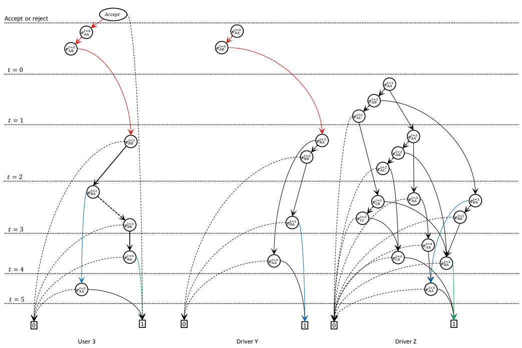
and capacity constraints. The set operations using this part can be computed efficiently by using ZDD data structure [1]. In the instance with User 3, Driver Y and Driver Z shown in Table 4.4, which is introduced in the previous section, the feasible itineraries can be enumerated as shown in Fig. 4.7(a). From this, the algorithm select feasible combination of itineraries that satisfy capacity constraints.

At each time, the algorithm weight the cost $C_t$ revealed at that time to $\mathcal{Z}$ (Line 7). This process can be simply executed by weighting the cost to all edge shown in Fig. 4.7(a). Then, the algorithm extracts the combinatorial action $\mathcal{Z}_t$ of all individuals existing at that time (Line 8) and calculate the score of each action $\{\boldsymbol{\delta}'_t, \boldsymbol{\eta}'_t, \boldsymbol{\theta}'_t\} \in \mathcal{Z}_t$ (Line 9 to 15). We take multi-scenario approach (Chang et al., 2000), that is, take the total sum of travel cost calculated in multiple scenario with respect to the appearance of future users. For example in the instance of Fig. 4.7(a), if the trip request of User 3 is accepted and the trip from Node A to Node B is presented at $t = 0$, the feasible itineraries for all individuals are shown in Fig. 4.7(b). The feasible itineraries of Driver Y is largely decreased because she has to carry User 3 from Node A to Node B at $t = 0$. The return trip of User 3 shown in blue and green arrows in Fig. 4.7(b) is also flexibly reserved in the enumerated combination of itineraries although this is not shown in the figure. By contrast, if the trip from Node A to Node C is presented at $t = 0$, the feasible itineraries for all individuals are shown in Fig. 4.7(c). In this case, the following itinerary of User 3 is fixed as is shown in the figure. The return trip from Node B to Node A at $t = 4$ should be served by Driver Z. Thus the feasible itineraries of Driver Z are largely decreased. Thus, the feasible combination of future itineraries are enumerated for each action. Using this, the minimum total travel cost for each scenario can be easily obtained as a shortest path problem. We use the total sum of the cost over all $M$ scenarios as the score of the action. The action with the minimum score is executed (Line 16).

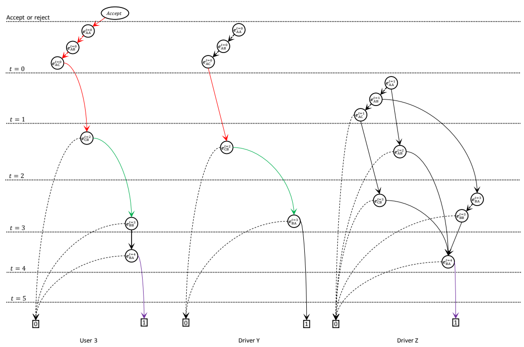
[1] A software package 'Graphillion' for these set operations are provided at https://github.com/takemaru/graphillion/wiki




(a) Feasible itineraries of User 3, Driver Y and Driver Z



(b) The case where User 3 is taken from Node A to Node B at $t = 0$



(c) The case where User 3 is taken from Node A to Node C at $t = 0$

FIGURE 4.7: A set of combinations of trip-chains among agents

TABLE 4.5: Parameters of the Drivers

| | Driver 1 | Driver 2 | Driver 3 | Driver 4 |
|---|---|---|---|---|
| Home location | Node A | Node B | Node C | Node C |
| Active time-window | [0,9] | [0,9] | [0,5] | [4,9] |
| Service capacity | 3 | 6 | 3 | 3 |
| Cost of travel | 5 | 7 | 5 | 5 |
| Cost of waiting | 2 | 2 | 2 | 2 |

TABLE 4.6: User categories

| | | Category 1 | Category 2 | Category 3 |
|---|---|---|---|---|
| Home location | | Node B | Node B | Node C |
| Start time of active time-window | | 0 to 2 | 0 to 1 | 0 to 2 |
| End time of active time-window | | 5 to 7 | 3 to 4 | 5 to 7 |
| Appearance probability | | $\mathcal{U}(0.5, 1.0)$ | $\mathcal{U}(0.5, 1.0)$ | $\mathcal{U}(0.5, 1.0)$ |
| Mandatory activity | Location | Node A | Node C | Node A |
| | Time-window | [4,5] | [2,7] | [4,5] |
| | Value | 20 | 20 | 20 |
| Cost of travel | | 1 | 1 | 1 |
| True value of staying at | Node A | $\mathcal{U}(-10, 10)$ | 0 | $\mathcal{U}(-10, 10)$ |
| | Node B | 0 | 0 | $\mathcal{U}(-10, 10)$ |
| | Node C | $\mathcal{U}(-10, 10)$ | $\mathcal{U}(-10, 10)$ | 0 |

## 4.5 Numerical Examples

### 4.5.1 Experimental setup

Assume a network shown in Fig. 4.1. There are four drivers shown in Table 4.5. There are three categories of users, which are shown in Table 4.6. We vary the number of users of each category in this numerical study. The mandatory activity of users in each category is given as shown in Table 4.6. However, the optional location where each user desires to visit depends on the VoS at each node and thus is not known to the operator. Each user's start and end time of the active time-window is set randomly within the range shown in Table 4.6. The user's true VoS at each node is drawn from a uniform distribution on $[-10, 10]$, such that $C^S_{i,n} \sim \mathcal{U}(-10, 10)$. The estimated VoS $\tilde{C}^S_{i,n}$ is given by $\tilde{C}^S_{i,n} = \rho C^S_{i,n} + \sqrt{1-\rho^2} \cdot x$, where $\rho$ is a correlation coefficient and $x$ is drawn from a uniform distribution on $[-10, 10]$. The appearance probability of each user is drawn from a uniform distribution on $[0.5, 1.0]$, such that $p_i \sim \mathcal{U}(0.5, 1.0)$.

### 4.5.2 Algorithms

We compare the performance of the algorithm from Section 4.4, called **Proposed**, with benchmark algorithms shown in Table 4.7. In the table, we show three static and three dynamic algorithms. Offline optimal (**OPT**) algorithm is the benchmark assuming that the operator has perfect knowledge of users, including both the appearance and true VoS. Optimal under the estimated VoS (**OPT-est**) algorithm assumes that the operator has perfect knowledge of users' appearance but does not know the true VoS. These algorithms statically optimizes the traffic assignment by solving *Offline optimal* optimization problem shown in Section 4.3.1 based on the true

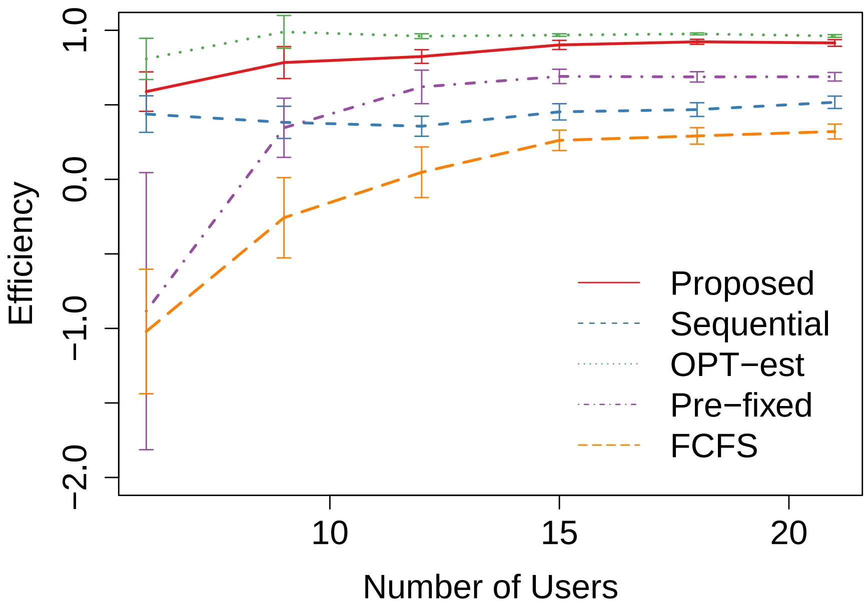



FIGURE 4.8: Efficiency of the algorithms over number of users

VoS in **OPT** and the estimated VoS in **OPT-est**. Pre-fixed (**Pre-fixed**) algorithm assumes no knowledge of either users' appearance and the true VoS. They solve the *Offline optimal* based on the estimated VoS, assuming that all users appear. The trivial dynamic benchmark algorithm is First-come first-serve (**FCFS**) that decides and fixes each user's itineraries to minimize the user's travel at the start time of each user's active time-window. The algorithm only users estimated VoS of each user. **Sequential** algorithm myopically solves the dynamic optimization problem shown in Section 4.3.2 at each time, only considering users who have already started their trips. In addition to the estimated VoS of users, the true VoS can be used for the optimization after the VoS is revealed. This algorithm can be solved by any MIP solver. Our proposed algorithm (**Proposed**) uses the information of prior probability of users' appearance. The service is expected to be more efficient because of this information, although it causes a difficulty to be solved by common MIP solvers.

### 4.5.3 Results

#### Efficiency

We evaluate the *efficiency* of our proposed algorithms. Here, we define *efficiency* as the negative total travel cost, i.e., social welfare, achieved by each mechanism as a

TABLE 4.7: Evaluated algorithms

| | Appearance | Value of staying at nodes (VoS) | MIP solvers |
|---|---|---|---|
| **Static algorithms** | | | |
| Offline optimal (**OPT**) | perfect knowledge | true VoS | ✓ |
| Optimal under the estimated VoS (**OPT-est**) | perfect knowledge | estimated VoS | ✓ |
| Pre-fixed (**Pre-fixed**) | all users | estimated VoS | ✓ |
| **Dynamic algorithms** | | | |
| First come first serve (**FCFS**) | - | estimated VoS | ✓ |
| Sequential (**Sequential**) | - | estimated and revealed VoS | ✓ |
| Proposed (**Proposed**) | prior probability | estimated and revealed VoS | |

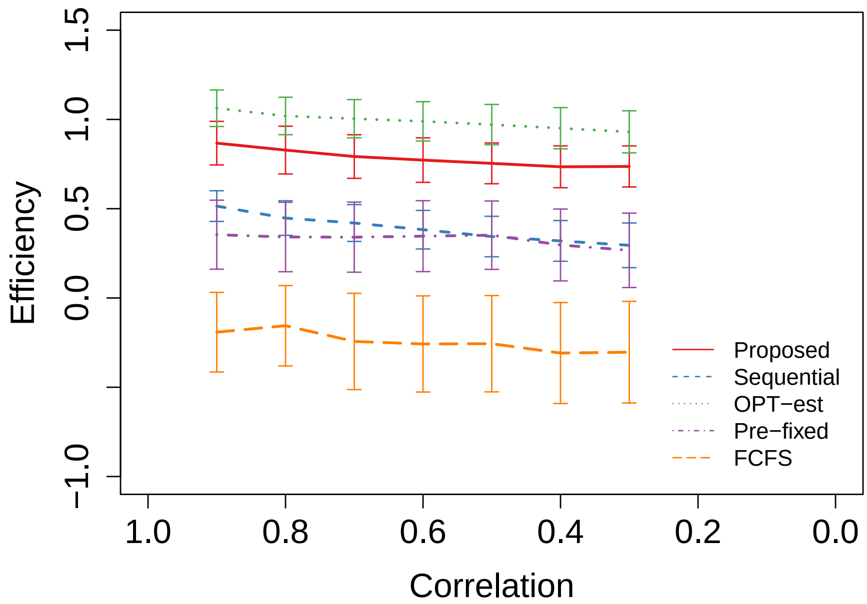



FIGURE 4.9: Efficiency of the algorithms over correlation coefficient between the true and the estimated VoS

proportion of the offline-optimal (**OPT**). We consider a case where $\rho = 0.6$, and vary the number of users from 3 to 21. The number of users of each category is set to be the same. The experiment was repeated 100 trials for each setting. The results are shown in Fig. 4.8. The plots in these figures show the mean value of the trials and the error bars express 95% confidence.

The gap between **OPT-est** and **OPT** (i.e., 1.0) shows the loss derived from the gap between estimated and true VoS, which is decreasing with the number of users. The gap between **Pre-fixed** and **OPT-est** shows the loss derived from the information of the appearance of users. It seems much larger than the gap derived from the inaccurate VoS. The efficiency of **Pre-fixed** is negative when the number of users is small, which means that the algorithm makes a deficit because of the non-appearance of users. A trivial dynamic algorithm, **FCFS**, performs much worse because it does not have any information about future users. **Sequential** algorithm performs better because it sequentially recalculate the optimization problem using information about revealed VoS and newly appearing users. This myopic approach performs better than static **Pre-fixed** algorithm if the number of users is small, but performs worse if the number of users is large. Our **Proposed** algorithm performs better by considering the appearance probability of future users. Fig. 4.9 shows the efficiency with varying correlation $\rho$ between the estimated and the true VoS, where the number of users is set to nine. As is seen in this figure, the trend is consistent with the accuracy of the estimated VoS.

### Reject rate

Fig. 4.10 shows the reject rate of each algorithm. Here, the reject rate represents the ratio of rejected users per all appeared users. When the number of users is small, the mechanism can improve its efficiency by rejecting some users to assure the cost. By contrast, when the number of users is large, it may reject some users because of the shortage of drivers. As is shown in Fig. 4.10, the rejection rate by the optimal

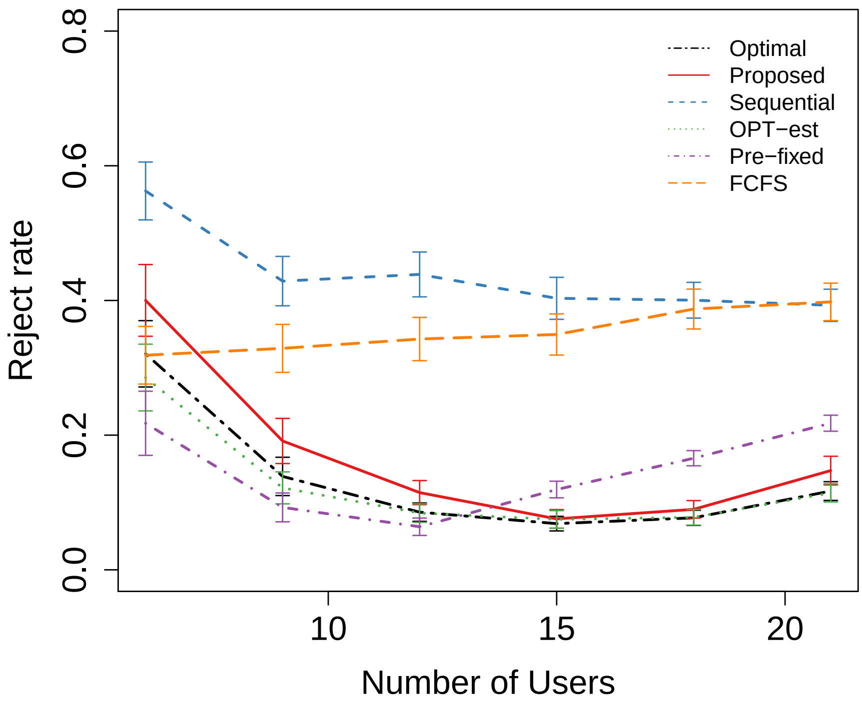



FIGURE 4.10: Rejection rate of the algorithms

solution (**OPT**) is concave with respect to the number of users, that is the adequate number of users exists to the supply condition. Similarly, the rejection rates by other static algorithms, **OPT-est** and **Pre-fixed** are also concave. However, in the case of the **Pre-fixed** algorithm, the number of users that achieve the least rejection rate is fewer than the optimal solution. It can be seen that the **Pre-fixed** algorithm can treat less users than the optimal algorithm. The rejection rate by common myopic dynamic algorithms, **FCFS** and **Sequential**, are extensively high because of the myopic feature. However, our **Proposed** dynamic algorithm achieve much lower and near-optimal rejection rate because of the non-myopic feature. The rejection rate is concave and the adequate number of users are consistent with the optimal solution.

## 4.6 Conclusions and discussions

In this study, we proposed a dynamic matching algorithm for sparse ridesharing markets. In contrast to existing trip-based algorithms, we took the approach of active-based travel analysis to consider the connectivity of trips of each user and driver. We formulate the dynamic optimization problem for ridesharing services that realizes the *floating booking system* which aims to minimize time-discounted total travel cost by all users and drivers under the dynamic environment. We showed a search-based algorithm that solves the optimization model using the ZDD data structure. In numerical studies, we showed that our proposed algorithm performs better than both static and dynamic benchmarks that can be solved by common MIP solvers, and obtains near-optimal solutions.

Our proposed framework brings up various directions for discussion. The first is the non-cooperative settings among users, drivers, and the service operator. Although we assume the cooperative settings in this chapter, users can be better-off by strategically reporting their demand. Thus the pricing algorithm that promotes users' truthful reports should be discussed. To this end, Bergemann and Välimäki (2010) proposed the pivot mechanism by which strategy-proofness of users is guaranteed, in that customers can never be better-off by misreporting. Our proposed framework that aims to minimize expected time-discounted total travel costs is consistent with this mechanism. Given that, we can design a mobility system that achieves social optimal states by the best response strategy of self-interested customers (Hayakawa and Hato, 2018a). In considering the pricing algorithm, it is natural to focus on the profit of the operator. The operator of mobility services may not hold a fixed number of traffic resources but procure them flexibly. In such a setting, the combined procurement, pricing, and allocation problems should be considered, which bring up a trade-off between the profit of the operator and the benefit of its customers (Hayakawa et al., 2015, 2018).

The second is the discussion on the applications. Our proposed framework presents the possibilities of many emerging applications. One instance is the electric vehicle (EV) dispatch problems for car-sharing or ride-sharing services. EVs have a limited cruising distance depending on their battery capacities and thus, they have to return to a charging station before the battery runs out. Thus, the executable trajectory of vehicles is expressed in space–time prism constraints based on a series of customers and charging stations. The problem is formulated as a combination of the matching between vehicles and customers, routing, and selecting charging stations problems. This setting provides a lot of non-linear constraints and thus, our proposed search-based algorithm is effective.

The third is the discussion on the various types of space–time prism constraints. Hägerstrand (1970) introduced three types of space–time prism constraints, namely, "capability constraints", "coupling constraints" and "authority constraints." In this chapter, we consider the "capability constraints" of vehicles and "coupling constraints" between users and drivers. However, by introducing other types of constraints, our proposed framework becomes more resourceful. Considering "coupling constraints" between multiple users, we can express the activities with families and friends. If we introduce "authority constraints," we can discuss far wider problems. For example, it can be used as a tool to design premium charge for limited priority members. It may also be useful to discuss the diversity of mobility services aiming to realize services for all people, including the elderly, children, handicapped persons, and so on. Emerging technologies in the coming age, including automated vehicles, can provide mobility services for people who cannot have their own driver's licenses, and our proposed framework can be extended to evaluate the possibility of such achievements by new services.

However, our study still has many limitations. The first and the most important problem to be discussed is the method to measure the utility of activities. In the earlier study, Kitamura and Supernak (1997) presented the concept of "temporal utility profiles" of activities and travel, and empirically discussed it using data obtained in the San Diego Zoo, and after that, many studies approached this problem (Chikaraishi, 2018). With recent progress in information technologies, a data driven approach for this problem is promising. Another limitation to be discussed is solution algorithms to solve the problems more efficiently. Depending on applications, the search-based algorithms should be refined. Heuristic approach such as $A^*$-based algorithms can potentially be expanded to our settings. In the future, we plan to explore these problems.

# Chapter 5

# Mechanism Design of Mobility Services Incorporating Behavioral Time Preference

In this chapter, we present a conceptual framework for the dynamic traffic resources allocation problem in the mobility as a service (MaaS) setting. We discuss it using activity-based model (Kitamura et al., 1996) by which the heterogeneity among users and the relation between a series of transfers are represented. We also introduce the framework of game theory to capture the elasticity of demands influenced by the price of services.

A part of the content of this chapter has been presented in *Hayakawa and Hato (2018a), Auction-based implementation of traffic services to maximize activity-based social welfare, The 7th Symposium of the European Association for Research in Transportation (hEART2018), Athens*, and the preprint is published as *Hayakawa and Hato (2018b), Dynamic traffic resources allocation under elastic demand of users with space-time prism constraints.*

We present a conceptual framework for the combined problem of traffic resource allocation and pricing for dynamic on-demand traffic services, often referred to as Mobility as a Service (MaaS), considering users with heterogeneous preferences and space–time constraints. We express the users' successive actions and transfers by using activity-based travel analysis and formulate the problem as trip-chain auctions. Using the formulation, we seek the mechanisms that reasonably give priority to late-coming high-value users by providing incentives to the early-coming low-valued users, unlike the common first-come first-served mechanism that always gives priority to the early-coming users. We characterize the optimal and truthful mechanisms in the MaaS settings that achieve system optimal state spontaneously under the capacity constraints of traffic resources and space–time constraints of users by the actions of the selfish user agents. We then introduce two optimal and truthful mechanisms, one of which guarantees non-negative *ex-post* revenue and the other guarantees non-negative *ex-post* utility of agents. We subsequently numerically evaluate them and discuss the properties required by the MaaS applications.

## 5.1 Introduction

Recently, transportation services under which multiple traffic mode services are provided by a service operator through mobile apps are often called Mobility as a Service (MaaS). For the operators of such systems, it is a significant challenge to dynamically allocate traffic resources with limited capacity to users having heterogeneous preferences and constraints. To this end, there exist numerous studies such as those on the dial-a-ride problem (Cordeau and Laporte, 2007; Ho et al., 2018) or dynamic ridesharing (Agatz et al., 2012). However, these works assume users are homogeneous; for example, a fixed admissible travel delay is set for all users. They also assume that trip-based demand, such as origin-destination tables, is given deterministically or stochastically and does not depend on the quality of services. However, real-world users are heterogeneous and decide to whether use the service or not depending on its quality. Specifically, users often make this decision by considering an entire trip-chain, which consists of successive transfers and activities. In other words, the users decide to use a traffic service if it provides an overall valuable experience. For example, a user books a taxi to a dinner with friends at a restaurant but would not use the taxi if the dinner is canceled. Considering that the trip is derived demand, the MaaS is required to provide value from its usage. To address this problem, we propose a conceptual framework by using activity-based travel analysis (Kitamura et al., 1996), in which heterogeneous users have their own space–time prism constraints (Hägerstrand, 1970). Specifically, we characterize mechanisms that satisfy the capacity constraints of traffic resources and the space–time constraints of users *at any time*. If the MaaS system is operated by such mechanisms, users can know in advance whether their trip-chain, for example, going shopping on the way to going to see movie and returning home before dinner, is executable or not under the capacity constraints of traffic resources and can thus decide whether to use the service.

The another important element that influences decision making is the price of the service. Users may accept a less preferable trip-plan if costs are lower, thus a well-defined pricing mechanism might lead to the efficient use of limited traffic resources. Akamatsu and Wada (2017) proposed a demand response scheme based on the Vickrey–Clarke–Groves (VCG) mechanism (Clarke, 1971; Groves, 1973; Vickrey, 1961) in a static setting. The VCG is a mechanism that optimally allocates resources based on reported user types and charges the externality that each user gives to the society. The VCG mechanism is known to be dominant strategy incentive compatible, meaning users can maximize their utility by the truthful report of their types. Using this mechanism, the system optimal (SO) state is achieved as an user equilibrium (UE) derived by the selfish behavior of each agent. The dynamic pivot (Bergemann and Välimäki, 2010) and online VCG (Parkes and Singh, 2004) mechanisms extend the VCG mechanism to dynamic settings. Both mechanisms are Bayesian–Nash incentive compatible (BNIC), meaning the dynamic SO state is achieved by the selfish strategy of agents. The properties of these mechanisms are explained in detail by Cavallo et al. (2009). However, no existing studies indicate the effectiveness of introducing these mechanisms into transportation systems such as MaaS, in which heterogeneous users require a series of trips within their space–time prism constraints. In this chapter, we thus consider an on-demand mobility service that requires the sequential decision-making of the operator, and discuss the allocation and pricing mechanisms that achieve the dynamic SO state under the selfish behavior of users. Specifically, we apply the *RC-optimal* mechanism combined with the pricing algorithms that are adopted in the dynamic pivot (Bergemann and Välimäki, 2010) and online VCG (Parkes and Singh, 2004) mechanisms, and discuss the conceptual property of the system.

Overall, this chapter makes the following contributions:

- We characterize, for the first time, a class of dynamic activity-based traffic resources allocation mechanisms that guarantee the satisfaction of both, space–time prism constraints of customers and capacity constraints of traffic resources at any time, in its sequential decision-making.
- We propose a framework of activity-based trip-chain auctions. The proposed concept supports the pricing scheme introduced under MaaS, in which users have flexibility in their trips and may give way to other users depending on price. We characterize a class of BNIC mechanisms in the settings of activity-based trip-chain auctions. We provide solution algorithms that allocates traffic resources efficiently by rationally reallocating previously assigned traffic resources to late-coming high-value customers.
- We introduce two pricing algorithms, both satisfying the property stated above. We evaluate them numerically and discuss their natures from the viewpoint of traffic services providers.
- We numerically show that our proposed solution algorithm can keep more than trillions of combinatorial trip options of current and future agents in rational computational time. We also numerically explore the trade-off between allocation efficiency and computational costs of our proposed wide range algorithms.

## 5.2 Literature review

There are many studies on transportation research that design incentives through which a socially optimal state is achieved voluntarily by self-interested users (Pigou, 1920). A representative study of this approach is the tradable credit scheme proposed by Yang and Wang (2011), wherein tradable credits are initially distributed to users, and optimal traffic condition can thus always be achieved if credits are optimally distributed. And this approach is extended by many works, for example, He et al. (2013); Nie and Yin (2013); Wu et al. (2012). By contrast, Akamatsu and Wada (2017) proposed a demand management scheme employing tradable bottleneck permits based on the VCG mechanism, in which traffic conditions are optimized without prior knowledge of travel demands. Further, Hara and Hato (2017) had a similar approach to car-sharing. However, these studies consider static settings, that is, those in which the traffic resources and the demand reported by users are statically given to the operator. The pricing scheme for traffic services in which the operator dynamically allocate traffic resources to users based on the sequentially revealed demand has not been well studied.

There are also several studies that proposed auction-based pricing algorithms for transportation services, mainly considering ridesharing. Kamar and Horvitz (2009) introduced the VCG-based payments to optimize shared plans for ridesharing. Kleiner et al. (2011) proposed the modified second-prize scheme to achieve efficient allocation in dynamic ridesharing. Further, Asghari et al. (2016) proposed an auction-based pricing mechanism for ridesharing services that maximizes the revenue of the platform provider. However, these studies assume trip-based demand is given and, thus, the connectivity of successive trips made by heterogeneous users with space–time constraints is not considered. Ma et al. (2018) proposed a spatio-temporal pricing for ridesharing that is incentive compatible for drivers with unique

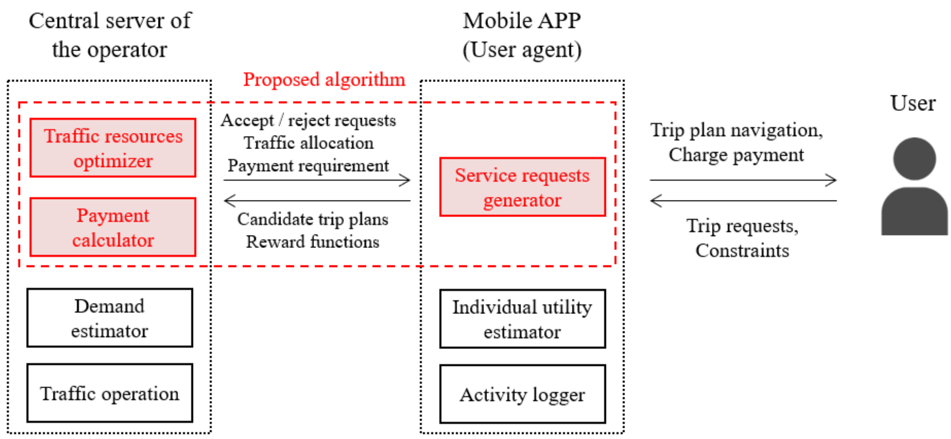



FIGURE 5.1: System overview

time preferences, considering successive trips. However, it also assumes the fixed trip-based demands are given.

In this chapter, we discuss the mechanism design of mobility services that consider the activity-based utilities of heterogeneous users with time–space prism constraints, extending the basic idea of Hayakawa and Hato (2018a).

## 5.3 System overview

We consider a MaaS system, which is implemented by a traffic service operator, whose overview is shown in Fig. 5.1. As shown in this figure, the mobile app for the service is installed on the mobile phones of users, and also equips the activity logger and individual utility estimator. As a user books a trip, the mobile app generates a set of candidate trip-plans based on the *preference* and *constraints* that are generated at both users' direct requests and the estimated preference of users. Then, the mobile app reports the set of trip-plans and the reward function to the central server of the operator. As it receives the request, the operator determines whether to accept or reject the request. If the request is accepted, the operator sequentially determines traffic allocation. The operator grants to all accepted agents one trip from the reported candidate trip plans, regardless of the behavior of other users that report subsequently. Additionally, the operator decides the payment the user pays for the service. In what follows, we explain the system model of our MaaS settings.

### 5.3.1 Traffic network model

We assume that decisions are made at discrete time steps $t \in T = \{0, 1, \dots, \bar{T}\}$, where $T$ is the set of all time steps. The traffic network is expressed by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. It can represent multiple traffic modes by a concept of supernetwork (Arentze and Timmermans, 2004; Sheffi, 1985). The service operator, having a limited capacity of traffic resources for each edge and each time in the network, allocates the resource for each user. Since the traffic capacity constraints are strictly kept in the allocation, we do not consider congestion and assume that the travel time of each edge remains constant. We use $\tau_e$ to denote the travel time of edge $e \in \mathcal{E}$. Some facilities on node $n \in \mathcal{N}$ may have an active time duration, such as for example, the

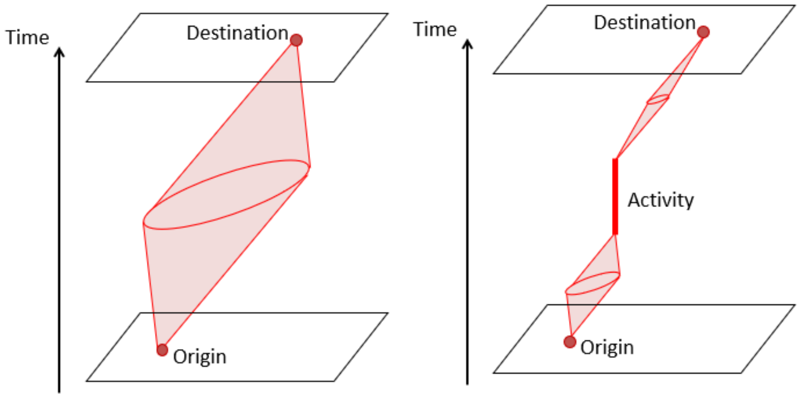



FIGURE 5.2: Space–time prism constraints

opening time of a restaurant. We use $b_n \subset T$ to denote the active time duration of node $n$.

### 5.3.2 User agent model

We consider a set of user agents denoted by $I$. Each user has its space–time prism constraints(Hägerstrand, 1970), that is, constraints related to the location and time of the trip. We show examples of space–time constraint in Fig. 5.2 as follows. Fig. 5.2(a) shows the space–time constraint derived from the set of origin, destination, and active time duration, and Fig. 5.2(b) the constraints with given additional activity during the active time duration. As such, the space–time prism of users in which users can execute some activities is tightened by taking a specific activity. Therefore, the users of transportation services are considered to be utility-maximizers within the given space–time prism constraints (Lam and Yin, 2001). We use $T_i = \{t_i^B, \dots, t_i^E\} \subset T$ to denote the active time duration of agent $i \in I$, where $t_i^B$ and $t_i^E$ are the beginning and the end of the active time duration, respectively. The origin and destination of agent $i$ are denoted by $O_i$ and $D_i$, respectively.

#### State transition

We introduce $s_{i,t} \in \mathcal{S}_i$ to denote the state of agent $i \in I$ at time $t \in T$, where the state is a multi-dimensional index including for example, location $l$, traffic mode $m$, and so on. $\mathcal{S}_i$ denotes a set of states that agent $i$ can take. We introduce the location function $\lambda(\cdot)$, namely the location $l$ of an agent with state $s_{i,t}$ is obtained by $l = \lambda(s_{i,t})$. Note that $l$ can be the middle of an edge when $\tau_e \geq 2$. The action that agent $i$ with state $s_{i,t}$ takes at time $t$ is denoted by $a_{i,t} \in \Gamma(s_{i,t}) \subset \mathcal{A}_i$, where $\Gamma(s_{i,t})$ is a set of actions that the agent can take at that time and $\mathcal{A}_i : \mathcal{S}_i \to \mathcal{S}_i$ denotes a set of actions that agent $i$ can take under all possible states. Taking action $a_{i,t} \in \Gamma(s_{i,t})$, the state of the agent transits from state $s_{i,t}$ to state $s_{i,t+1} \in \mathcal{S}_i$. We use $\mathcal{S} = \bigcup_{i \in I} \mathcal{S}_i$ to denote a set of states for all agents and $\mathcal{A} = \bigcup_{i \in I} \mathcal{A}_i$ to denote a set of actions taken by all agents.

As actions of agents, we consider *Moving* and *Staying*. We use $\mathcal{A}_i^M \subset \mathcal{A}_i$ to denote a set of *Moving* actions of agent $i$ while $\mathcal{A}_i^S \subset \mathcal{A}_i$ to denote a set of *Staying* actions, where $\mathcal{A}_i = \mathcal{A}_i^M \cup \mathcal{A}_i^S$. A set of staying actions $\mathcal{A}_i^S$ expresses a stay of agent $i$ on any node $n \in \mathcal{N}$, potentially, doing some activity. We assume that the staying actions are taken only on any node $n \in \mathcal{N}$ in the traffic network $\mathcal{G}$ and are not taken anywhere else such as, for example, in the middle of a edge. In contrast, moving action $a_{i,t} \in \mathcal{A}_i^M$ is taken along any edge $e \in \mathcal{E}$ in the traffic network $\mathcal{G}$. We introduce the edge function $\eta(\cdot)$. The edge $e \in \mathcal{E}$, related to action $a_{i,t} \in \mathcal{A}_i^M$ of an agent, is obtained by $e = \eta(a_{i,t})$. For any staying action $a_{i,t} \in \mathcal{A}_i^S$, $\eta(a_{i,t}) = \emptyset$.

Given that, we introduce a function $\delta : \mathcal{S} \times \mathcal{A} \times \mathcal{E} \to \{0, 1\}$ to express the relationship between actions and edges, as follows:

$$\delta(s_{i,t}, a_{i,t}, e) = \begin{cases} 1 & \text{if} \quad \lambda(s_{i,t}) \in \mathcal{N}, \text{ and } \eta(a_{i,t}) = e \\ 0 & \text{otherwise.} \end{cases} \tag{5.1}$$

Specifically, $\delta(s_{i,t}, a_{i,t}, e) = 1$ if an agent $i$ that is located on any node in the traffic network at time $t$ take a moving action related to edge $e$ at that time, and otherwise $\delta(s_{i,t}, a_{i,t}, e) = 0$. Given that, the traffic volume $F_{e,t}$ that flows into edge $e$ at time $t$ is expressed as:

$$F_{e,t} = \sum_{i \in I} \delta(s_{i,t}, a_{i,t}, e). \tag{5.2}$$

We assume that an agent starting the moving action at time $t \in T$ along a edge $e$ continues the action until it reaches the end of the edge at time $t + \tau_e$ and does not change the action in the middle of the edge.

Given these notations, the activity-based reward function of agent $i$ is denoted by $R_i : \mathcal{S}_i \times \mathcal{A}_i \times T \to \mathbb{R}$.

### Trip plan

A series of state-action transitions $l_i$ during active time duration $T_i = [t_i^B, t_i^E]$ is denoted by;

$$l_i = \{s_{i,t_i^B}, a_{i,t_i^B}, \ldots, s_{i,t_i^E-1}, a_{i,t_i^E-1}, s_{i,t_i^E}\} \in L_i, \tag{5.3}$$

where $L_i$ is the set of all state-action transitions that the agent can take within its space–time prism constraints bonded by the origin $O_i$, the destination $D_i$, the active time duration $T_i$, and other constraints for example derived by some mandatory activities. The series of state-action transitions $l_i$ expresses a trip-chain during the active time duration, which we subsequently call a trip-plan. Considering time $t$ such that $t_i^B \leq t < t_i^E$, a trip-plan after time $t$ is denoted by $l_i^{\langle t \rangle}$, so that:

$$l_i^{\langle t \rangle} = \{s_{i,t}, a_{i,t}, \ldots, s_{i,t_i^E-1}, a_{i,t_i^E-1}, s_{i,t_i^E}\} \in L_i^{\langle t \rangle}, \tag{5.4}$$

where $L_i^{\langle t \rangle}$ denotes a set of executable trip-plans after time $t$, while keeping the space–time prism constraints. Consequently, agent type $i$ is denoted by $\theta_i = \{R_i, L_i\}$.

Each agent reports its type to the operator at an arbitrary timing after the beginning of its active time duration, $t_i^B$. We consider agents to be strategic and able to misreport their type. We use $\hat{\theta}_i = \{\hat{R}_i, \hat{L}_i\}$ to denote the report of an agent. Here, we assume a limited misreport (Parkes, 2007) for the set of trip plans, that is, the agent can only report a subset of the true set of trip-plans, formally:

$$\hat{L}_i, \subset L_i. \tag{5.5}$$

The above is a natural assumption in this setting, because the agent does not usually report the undesirable trip-plan but report a limited set of trip-plans, typically reporting only the most desirable trip-plan, which leads to increasing its utility in existing booking systems operated on the first-come first-served (FCFS) basis.

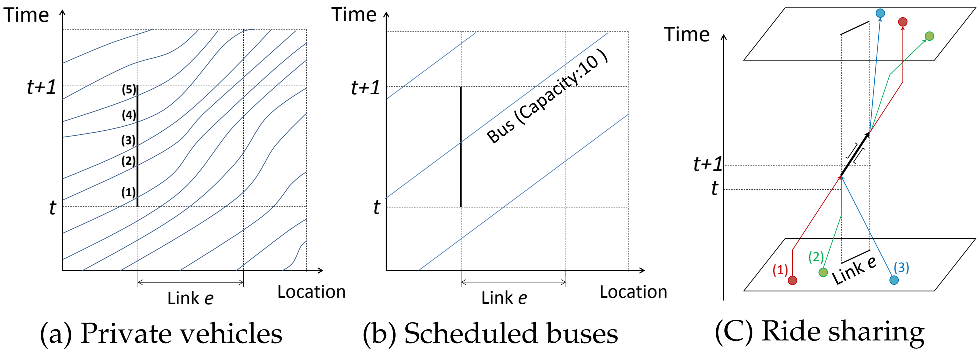



FIGURE 5.3: Traffic resource capacity of various traffic mode

### 5.3.3 Operator model

The service operator sequentially receives the agents' report and is required to allocate traffic resources for each user adequately at each time. In this part, we focus on the operator and introduce the model of the traffic resources capacity and its allocation.

#### Capacity of traffic resources

In this part, we explain the *capacity* of traffic resources. We use $C_{e,t}$ to denote the traffic capacity of edge $e \in \mathcal{E}$ at time $t \in T$, namely at most $C_{e,t}$ users can flow into edge $e$ at time $t$. Here, the capacity of traffic resources is a generalized concept of the road capacity for private vehicles, transportation capacity of scheduled bus services, acceptable volume of users of ride-share services, and so on. The traffic volume $F_{e,t}$ on edge $e$ at time $t$ is defined as $F_{e,t} = \sum_{i \in I} \delta(s_{i,t}, a_{i,t}, e)$ by Eq. 5.2. For example, given the space-time trajectories of private vehicles as shown in Fig.5.3(a), five vehicles flow into link $e$ at time $t$, and thus the traffic volume is obtained as $F_{e,t} = 5$, assuming that each vehicle is occupied by only one person. In this case, the traffic capacity $C_{e,t}$ coincides with the link capacity, that is, the maximum number of vehicles that can flow into the link within a unit time, and is constant over time. In the case of scheduled buses, the space–time trajectories of which are shown in Fig.5.3(b), the traffic volume is obtained as $F_{e,t} = 10$, since one bus with the capacity of 10 flows in to the link at time $t$. In this case, the traffic capacity $C_{e,t}$ is controlled by the service operator by changing the frequency and size of buses, and can be time-dependent. Finally, in the example of ride-share services, the space–time trajectories of which are shown in Fig.5.3(c), the traffic volume is obtained as $F_{e,t} = 3$ since three users share the vehicles that starts moving along the link $e$ at time $t$. In this case, the traffic capacity $C_{e,t}$ is also time-dependent and is determined by the combinatorial matching of trips by users. The method of bundling the trips of users are studied in many works (Alonso-Mora et al., 2017; Hara and Hato, 2017; Santi et al., 2014).

The assignment of the spatial road capacity for each traffic mode is also an important problem to be studied. For example, traffic operators can assign road capacity to private vehicles, scheduled buses or ride share services, by setting bus priority lanes or high-occupancy vehicle (HOV) lanes. With the widened reach of automated vehicles (AVs) as a result of which road capacity can possibly be increased, the assignment of priority lanes for AVs are also considered (Chen et al., 2016). To make the discussion simple, in this chapter, we assume that the traffic capacity of each traffic mode is given and represented by the capacity of edge $C_{e,t}$, note that an edge represents the transfer made by a traffic mode. Specifically, our proposed framework discusses the allocation of limited traffic resources to users with heterogeneous space–time prism constraints. However, this framework can be generalized to the settings where the capacity $C_{e,t}$ of each traffic mode is not given and only the constraints, such as road capacities, the number of vehicles, or total budgets, are given, which we discuss in Section 5.6

**Allocation of traffic resources**

Under capacity constraints, while collecting the agents' reports, the operator dynamically decides whether it accepts or rejects the requests. For all agents whose reports are accepted, the operator allocates traffic resources and determines the payments. Once the operator accepts the request of agent $i \in I$, it guarantees the agent is assigned any trip within the set of candidate trip-plans $L_i$, meaning the agent can finish its trip within its time–space prism constraints.

We assume the operator has a predicted demand model $\tilde{\mathcal{D}}_t$ that represents stochastic information about agents reporting after time $t$. However, the operator does not know the accurate agent types until their report. We use $I_t^{reported} = \{1, 2, \ldots, \bar{I}_t\}$ to denote the set of all agents that report to the operator by time $t$, and $\hat{\boldsymbol{\theta}}_t = \{\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{\bar{I}_t}\}$ to denote all reports the operator has received from agents until time $t$. Then, we use $I_t$ for the set of agents existing at time $t$, that is, all agents with $t_i^B \leq t \leq t_i^E - 1$, except agents that are rejected before time $t$. We further assume the operator knows the state $s_{i,t}$ of agent $i \in I_t$ at time $t$ that has already reported, which is natural under the assumption that the operator provides traffic services for all agents. We use $\boldsymbol{S}_t$ to denote the joint state of all existing agents $i \in I_t$. The operator then decides the joint action $\boldsymbol{\pi}_t$ for all existing agents $i \in I_t$ as:

$$\boldsymbol{\pi}_t = \boldsymbol{\pi}(\hat{\boldsymbol{\theta}}_t, \boldsymbol{S}_t, \tilde{\mathcal{D}}_t), \tag{5.6}$$

where $\pi_{i,t} \in \Gamma(s_{i,t})$, the elements of $\boldsymbol{\pi}_t$, are the action allocated to agent $i$ at time $t$.

To guarantee service quality for users even in the worst case scenario, the decision has to satisfy the space–time prism constraints of users and capacity constraints of traffic resources at any time. Formally, the space–time prism constraints are given by;

$$\forall i \in I_t, \exists l_i^{\langle t \rangle} \in \{L_i^{\langle t \rangle} \cup l_{i,Reject}^{\langle t \rangle}\}, \pi_{i,t} \in l_i^{\langle t \rangle}, \tag{5.7}$$

, where $l_{i,Reject}^{\langle t \rangle}$ denotes that the report of agent $i$ at time $t$ is rejected. This constraint means that there exists at least one executable trip plan for all existing agents, except for the agents rejected at time $t$. By contrast, the capacity constraints of traffic resources are given by;

$$\exists \boldsymbol{l}_{I_t}^{\langle t \rangle} \in \boldsymbol{L}_{I_t}^{\langle t \rangle}, \forall e \in \mathcal{E}, \forall t' \geq t : \sum_{i \in I_t} \sum_{(s_{i,t'}, \pi_{i,t'}) \in l_i^{\langle t \rangle}} \delta(s_{i,t'}, \pi_{i,t'}, e) \leq C_{e,t'}, \tag{5.8}$$

, where $\delta : \mathcal{S} \times \mathcal{A} \times \mathcal{E} \rightarrow \{0, 1\}$ expresses the spatial relationship between the actions and edges. This constraint means that there are joint trip plans across all accepted agents, which do not violate the capacity of traffic resources, not only at the current time but also at any time in the future.

In addition to determining the allocation of traffic resources, the operator also determines the payment $x_{i,t}$, that is, the payment of agent $i$ at time $t$, so that

$$x_{i,t} = x_i(\hat{\boldsymbol{\theta}}_t, \mathbf{S}_t, \tilde{\mathcal{D}}_t), \tag{5.9}$$

and charges it to the agent at each time.

### 5.3.4 Definitions of utility and social welfare

We assume the agents are self-interested and rational utility maximizers. The *ex-post* utility $U_i$ of agent $i$, which is allocated the trip plan $l_i = \{s_{i,t_i^B}, \pi_{i,t_i^B}, \dots, s_{i,t_i^E-1}, \pi_{i,t_i^E-1}, s_{i,t_i^E}\}$ and charged $x_i = \{x_{i,t_i^B}, \dots, x_{i,t_i^E-1}\}$, is expressed as

$$U_i = \sum_{t=t_i^B}^{t_i^E-1} \{R_i(s_{i,t}, \pi_{i,t}, t) - x_{i,t}\}. \tag{5.10}$$

Given the time-discount rate $\beta$ (Rust, 1994), the discount utility $DU_{i,t}$ of agent $i$ at time $t$ is expressed as

$$DU_{i,t} = \sum_{t'=t}^{t_i^E-1} \beta^{t'-t}\{R_i(s_{i,t'}, \pi_{i,t'}, t') - x_{i,t'}\}, \tag{5.11}$$

where $\beta$ is the time discount rate. We assume that $\beta$ is common for all users and the operator.

Hence, we define social welfare as the summation of utilities achieved by all agents and the operator. Namely, the *ex-post* total social welfare $SW$ achieved by allocation $\boldsymbol{\pi}$ is stated as:

$$SW(\boldsymbol{\pi}|\boldsymbol{s}_0) = \sum_{i\in I} U_i + \sum_{t\in T}\sum_{i\in I_t} x_{i,t} = \sum_{t\in T}\sum_{i\in I_t} R_i(s_{i,t}, \pi_{i,t}, t) = \sum_{t\in T} R(\boldsymbol{s}_t, \boldsymbol{\pi}_t), \tag{5.12}$$

where $\boldsymbol{\pi} = \{\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_{\bar{T}-1}\}$ denotes the allocations to all agents at all times and $R(\boldsymbol{s}_t, \boldsymbol{\pi}_t) = \sum_{i\in I_t} R_i(\boldsymbol{s}_t, \pi_{i,t}, t)$ denotes the sum of rewards of all agents given allocation $\boldsymbol{\pi}_t$ under state $\boldsymbol{s}_t$. Here, $\boldsymbol{s}_0$ denotes the given initial state. As shown in this equation, payment $x_{i,t}$ is canceled out between agents and the operator, and social welfare is expressed by the summation of the rewards received by all agents. As the operator makes sequential decisions at each time, we consider the discounted social welfare $DSW_t$ as follows:

$$DSW_t = DSW(\boldsymbol{\pi_t}, \boldsymbol{\pi_{t+1}}, \dots, \boldsymbol{\pi_{\bar{T}}}|\boldsymbol{s}_t) = \sum_{t'=t}^{\bar{T}-1} \beta^{t'-t} R(\boldsymbol{s}_{t'}, \boldsymbol{\pi}_{t'}). \tag{5.13}$$

### 5.3.5 Strategic behavior

As previously stated, we assume the agent can strategically misreport its type, $\hat{\theta}_i \neq \theta_i$. We use $\theta_{-i}$ and $\hat{\theta}_{-i}$ to denote the type and reports of all agents except agent $i$. The reports of all agents are expressed as $\{\hat{\theta}_i, \hat{\theta}_{-i}\}$ and the *ex-post* utility of agent $i$, which misreports $\hat{\theta}_i$, is expressed as:

$$U_i(\boldsymbol{\pi}(\{\hat{\theta}_i, \hat{\theta}_{-i}\}), \theta_i) = \sum_{t\in T_i} \{R_i(\pi_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\}), \theta_i) - x_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\})\}, \tag{5.14}$$

where $\pi_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\})$ and $x_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\})$ are the the allocation and the payment given the reports of all agents. Similarly, the discounted utility of agent $i$ that misreports $\hat{\theta}_i$ at time $t$ is expressed as:

$$DU_i(\boldsymbol{\pi}(\{\hat{\theta}_i, \hat{\theta}_{-i}\}), \theta_i) = \sum_{t'=t}^{t_i^E - 1} \beta^{t'-t} \{R_i(\pi_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\}), \theta_i) - x_{i,t}(\{\hat{\theta}_i, \hat{\theta}_{-i}\})\}. \quad (5.15)$$

We aim to design a mechanism that is BNIC, meaning agents can maximize their expected discounted utility by truthfully reporting their own types if all other agents truthfully report $\hat{\theta}_{-i} = \theta_{-i}$ (Bergemann and Välimäki, 2010; Parkes and Singh, 2004). Formally, this is written as

$$\forall \theta_i, \hat{\theta}_i, \theta_{-i} : \mathbb{E}\left[DU_i(\boldsymbol{\pi}(\{\theta_i, \theta_{-i}\}), \theta_i)\right] \geq \mathbb{E}\left[DU_i(\boldsymbol{\pi}(\{\hat{\theta}_i, \theta_{-i}\}), \theta_i)\right], \quad (5.16)$$

where $\mathbb{E}[\cdot]$ expresses expectation with respect to unrevealed future demands.

We also aim to establish individual rationality (IR), that is, an agent's utility is guaranteed to be non-negative when it reports truthfully. We introduce two IR concepts: *ex-post* and *ex-ante* IR. The former guarantees the *ex-post* utility is non-negative regardless of the reports of other agents, whereas for the latter the expected discount utility is non-negative assuming the truthful report of all other agents. Formally, *ex-post* IR is expressed as

$$\forall \theta_i, \hat{\theta}_{-i} : U_i(\boldsymbol{\pi}(\{\theta_i, \hat{\theta}_{-i}\}), \theta_i) \geq 0, \quad (5.17)$$

whereas *ex-ante* IR is written as

$$\forall \theta_i, \theta_{-i} : \mathbb{E}\left[DU_i(\boldsymbol{\pi}(\{\theta_i, \theta_{-i}\}), \theta_i, )\right] \geq 0. \quad (5.18)$$

The IR property is important for penetrating the system because people may hesitate to participate if the system may result in negative utility.

### 5.3.6 Optimal and truthful mechanisms in the MaaS settings

We now define the optimal and truthful mechanisms in the MaaS settings, using the definition of the BNIC introduced in the previous part and the *RC-optimal* allocation introduced in Hayakawa and Hato (2018b).

**Definition 1** (Optimal and truthful mechanism in the MaaS settings)**.** *We say that a mechanism is optimal and truthful in the MaaS settings, if the mechanism achieves the RC-optimal allocation and is BNIC.*

If a mechanism is *RC-optimal* and BNIC, the Bayesian–Nash equilibrium is spontaneously achieved by truthful reports from all agents and the achieved equilibrium state coincides with the SO states under the time–space constraints of users and capacity constraints of traffic resources.

## 5.4 Mechanisms

Here, we introduce optimal and truthful mechanisms in the MaaS settings defined in the previous section. Specifically, we adopt the dynamic pivot (Bergemann and Välimäki, 2010) and online VCG (Parkes and Singh, 2004) mechanisms, which extend the VCG mechanism (Clarke, 1971; Groves, 1973; Vickrey, 1961) that is incentive-compatible and maximizes social welfare in static settings.

### 5.4.1 Efficient allocation that sequentially maximizes social welfare

Both the dynamic pivot and online VCG mechanisms adopt an *efficient allocation*, which is defined as maximizing the expected discounted social welfare at any time. Formally, this is described as follows:

$$\boldsymbol{\pi}_t = \underset{\boldsymbol{\pi'}_t \in \Gamma(\boldsymbol{S}_t)}{\text{argmax}} \ \mathbb{E}\left[DSW_t\right], \tag{5.19}$$

where $\Gamma(\boldsymbol{S}_t)$ is a set of executable joint actions of all agents at time $t$. From Eq. 5.13, the efficient allocation in our setting is expressed as:

$$\boldsymbol{\pi}_t = \underset{\boldsymbol{\pi'}_t \in \Gamma(\boldsymbol{s_t})}{\text{argmax}} \ \mathbb{E}\left[\sum_{t'=t}^{\bar{T}-1} \beta^{t'-t} R(\boldsymbol{s}_{t'}, \boldsymbol{\pi}_{t'})\right]. \tag{5.20}$$

By Bellman's principle of optimality, it can be reformulated as a recursive form:

$$\boldsymbol{\pi}_t = \underset{\boldsymbol{\pi'}_t \in \Gamma(\boldsymbol{s_t})}{\text{argmax}} \ \left[R(\boldsymbol{s}_t, \boldsymbol{\pi'}_t) + \beta \cdot V(\mathcal{T}(\boldsymbol{s_t}, \boldsymbol{\pi'}_t))\right], \tag{5.21}$$

where $\mathcal{T}(\boldsymbol{s_t}, \boldsymbol{\pi}_t)$ denotes the state at time $t+1$, given the state $\boldsymbol{s_t}$ and action $\boldsymbol{\pi}_t$ at time $t$, and $V(\boldsymbol{s}_t)$ is a value function given by:

$$V(\boldsymbol{s}_t) = \mathbb{E}\left[\sum_{t'=t}^{\bar{T}} \beta^{t'-t} R(\boldsymbol{s}_{t'}, \boldsymbol{\pi}_{t'})\right]. \tag{5.22}$$

In the MaaS setting, the efficient allocation maximizes the expected discounted social welfare given by Eq. 5.21, while satisfying the space–time constraints of all users given by Eq. 5.7 and capacity constraints of traffic resources given by Eq. 5.8 at any time. Such an allocation mechanism is defined as the *RC-optimal* mechanism in Hayakawa and Hato (2018b).

### 5.4.2 Incentive compatible pricing schemes

Here, we show the pricing schemes adopted under the dynamic pivot and online VCG mechanisms, both achieving BNIC under *RC-optimal* allocation (Cavallo et al., 2009). We subsequently interpret these schemes from the viewpoint of applications related to transportation systems.

#### Dynamic pivot mechanism

First, we introduce the dynamic pivot mechanism (Bergemann and Välimäki, 2010). The pricing function of this mechanism is:

$$x_{i,t} = -R_{-i}(\boldsymbol{\pi_t}) - \beta \cdot V_{-i}(\mathcal{T}(\boldsymbol{s_t}, \boldsymbol{\pi_t})) + V_{-i}(\boldsymbol{s_t}), \tag{5.23}$$

where $R_{-i}$ denotes the total rewards by all agents, except $i$. Further, $V_{-i}(\boldsymbol{s_t})$ denotes the value function of the virtual market assuming that agent $i$ does not exist under the state $\boldsymbol{s_t}$. The first two terms in this equation show the expected discounted rewards for all agents, except $i$, given the joint action $\boldsymbol{\pi_t}$ at time $t$. Specifically, the first term shows the rewards of the current time step and the second term shows the expected future rewards. By contrast, the third term shows the expected discounted rewards of all agents, except $i$, at the current time, assuming that agent $i$

does not exist. Therefore, the payment given by Eq. 5.23 shows the marginal discounted externalities that agent $i$ gives to the market. The final payment $x_i$ of agent $i$ is determined by $x_i = \sum_{t\in T_i} x_{i,t}$.

The dynamic pivot mechanism that makes efficient allocations and charges payments as per Eq. 5.23 is BNIC. Additionally, under common settings, where agents can decide whether they participate on the market or not at each time, the mechanism is *ex-ante* IR, meaning that the expected utility of agents is guaranteed to be non-negative (Bergemann and Välimäki, 2010). However, in the transportation system, agents cannot cancel their trips at an arbitrarily timing. They can only cancel when reaching specific places, for example, their home or some hotels where to stay or continue traveling otherwise. The *RC-optimal* allocation mechanism considers such situations and allocates traffic resources so that all agents can finish their trips within their initially reported set of trip-plans. However, because each agent is charged marginal discounted externalities each time, the charged amount may exceed the discounted expected utility if a large number of high-value agents appears after starting the trip. Therefore, the dynamic pivot mechanism is *ex-ante* IR, meaning that the expected utility at the time of report is non-negative but the achieved *ex-post* utility can become negative. By contrast, because the payment of each agent at each time is non-negative, the *ex-post* revenue of the service operator is guaranteed to be non-negative. Further, this mechanism can treat the *dynamic type* of agents, meaning the agents can report their types (i.e., the reward functions and space–time constraints) at any time.

### Online VCG mechanism

Here, we introduce the online VCG mechanism (Parkes and Singh, 2004). The pricing function of this mechanism is:

$$x_{i,t} = \begin{cases} R_i(\pi_{i,t}) - V(\boldsymbol{s_t}) + V_{-i}(\boldsymbol{s_t}) & (t = t_i^B) \\ R_i(\pi_{i,t}) & (otherwise). \end{cases} \tag{5.24}$$

Namely, agents pay the expected externalities at point $t_i^B$ when reporting, and then pay as much as the obtained rewards in the following time step. The final payment $x_i$ of agent $i$ is determined by $x_i = \sum_{t\in T_i} x_{i,t}$.

The online VCG mechanism that makes efficient allocations and charges payments as per Eq. 5.24 is BNIC, same as for the dynamic pivot mechanism. However, in contrast to the dynamic pivot mechanism, it is *ex-post* IR, meaning the *ex-post* utility of agents is guaranteed to be non-negative if the agents report truthfully. Therefore, the online VCG mechanism with the *RC-optimal* allocation in the transportation system guarantees agents can finish their trip with (weakly) positive utility, if once accepted at the beginning. Practically, this can be implemented by the partly refund system. The service operator collects the maximum payment $x_i^{max}$ at the beginning given by:

$$x_i^{max} = \max_{l_i\in L_i} \sum_{\pi_{i,t}\in l_i} R_i(\pi_{i,t}) - V(\boldsymbol{s}_{t_i^B}) + V_{-i}(\boldsymbol{s}_{t_i^B}), \tag{5.25}$$

which is achieved by the best allocation for agent $i$, and refund $x_i^{max} - x_i$ depending on the inconvenience it has as a results. In any case, the utility of agents is fixed as $V(\boldsymbol{s}_{t_i^B}) - V_{-i}(\boldsymbol{s}_{t_i^B})$, being non-negative. This is similar to the collecting volunteer at the airport when the seats on an airplane are overbooked: part of the payment are refunded to the passengers who accept an alternative flight to compensate for the inconvenience. In contrast to the dynamic pivot mechanism, the *ex-post* revenue of

TABLE 5.1: Summary of the proposed framework of the solution algorithms

| Myopic/ Non-Myopic | Reports at a time-step | Maximum options to keep | Computational complexity | Efficiency of allocation | |
|---|---|---|---|---|---|
| Myopic | Sequential | 1<br>N<br>$\infty$ | small<br>↕ | low<br>↕ | FCFS |
| | Simultaneous | 1<br>N<br>$\infty$ | large | high | ↕ |
| Non-Myopic | Sequential | 1<br>N<br>$\infty$ | middle<br>↕ | middle<br>↕ | |
| | Simultaneous | 1<br>N<br>$\infty$ | extra large | extra high | Optimal |

the service operator could be negative, although the discounted expected revenue is non-negative at any time. This mechanism can only treat the *static type* of agents, meaning the agents can report their types only at the beginning at their trip and cannot modify them during the trip.

### 5.4.3 Solution algorithm

As previously shown, the *RC-optimal* allocation is adopted both in the dynamic pivot and online VCG mechanisms. The *RC-optimal* allocation can be obtained by a solution algorithm using the ZDD data structure, as is similar to algorithms shown in Chapter 4. Although payments under the dynamic pivot and online VCG mechanisms stated in the previous sections include several optimization processes, they can be calculated efficiently by a search-based algorithm using the ZDD data structure.

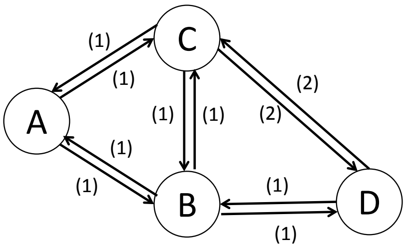
In addition to the *RC-optimal* allocation, some approximation algorithms can be considered (Hayakawa and Hato, 2018b). *Per agent* algorithm makes decisions for each agent separately to avoid the combinatorial explosion derived from the joint actions $\Gamma(\boldsymbol{s}_t)$ of all agents existing at time $t$. *Branch cutting* algorithm introduces the limitation $N^{max}_{branch}$ on the number of options. Namely, the number of options that the enumerated plan $\mathcal{Z}$ always keeps;

$$|\mathcal{Z}| \leq N^{max}_{branch}. \tag{5.26}$$

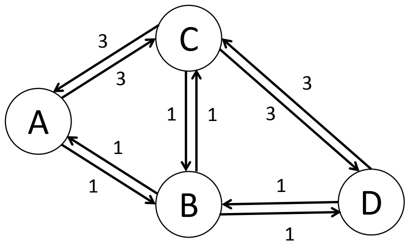
If the number of options exceeds $N^{max}_{branch}$, the only $N^{max}_{branch}$ options are left and the rest are cut off. There are various ways to select $N^{max}_{branch}$ options, for example, to select the highest $N^{max}_{branch}$ options in terms of rewards, to select $N^{max}_{branch}$ options randomly, or to use a combination of these two methods.

### 5.4.4 Property of proposed solution algorithms

The proposed framework of the solution algorithms, including the approximation algorithms, are summarized in Table 5.1. Our proposed framework provides myopic/non-myopic algorithms that process the information reported from agents at the same




(a) Travel time of each edge




(b) Traffic capacity of each edge

FIGURE 5.4: Sample network

time-step sequentially or simultaneously and keep the pre-defined maximum options. The framework provides a wide range of trade-offs between the computational complexity and the efficiency of allocation. At the simplest, the myopic algorithm that sequentially processes reports at a time-step and keeps only one option coincides with the FCFS algorithm that requires a low computational complexity, but is not efficient. In contrast, a non-myopic algorithm that simultaneously processes reports at a time-step and does not limit the number of options, achieves the optimal allocation that maximizes the discounted social welfare but requires extremely high computational efforts. As shown in Table 5.1, we provide wide-range algorithms between these two extreme cases. All of these algorithms are feasible and can be interrelatedly implemented by a search-based algorithm.

## 5.5 Numerical analysis

In a numerical analysis, we consider a simple network representing the setting where the trip demands of tourists interferes with the background trip demand of commuters. In the study, we use Graphillion[1], a Python software package on search, optimization, and enumeration using ZDD.

### 5.5.1 Experimental setup

Fist, we introduce a simple experimental setting to evaluate the basic performance of the proposed algorithm. In this setting, we set $\bar{T} = 8$. Specifically, we consider discrete time $T = \{0, 1, \ldots, 8\}$ in this analysis. We also assume that the time discount rate is $\beta = 1$. We show the experimental setup for the *static agent-type setting* in Table 5.2, which we state in detail in the following parts.

#### Network

The considered sample network is shown in Fig. 5.4. The numbers in brackets in Fig. 5.4(a) show the required travel time on each edge. The capacity of traffic resources, that is, the possible number of agents using the edge simultaneously at one time step is shown in Fig. 5.4(b). We assume that the capacity is constant over time. In this numerical study, we consider Nodes A and D to be residential and office areas, and Nodes B and C to be amusement areas. We set the active time for the facility on Node B as $b_B = \{3, 4, 5\}$, while the facilities on other nodes are set to be active for the entire time duration in $T$.

[1] https://github.com/takemaru/graphillion

TABLE 5.2: Experimental setup in the *static agent-type* setting

| | | |
|---|---|---|
| Network | Set of nodes | $\{A, B, C, D\} \in \mathcal{N}$ |
| | Set of edges | $\{AB, BA, AC, CA, BC, CB, BD, DB, CD, DC\} \in \mathcal{E}$ |
| | Travel time | $\tau_{AB} = \tau_{BA} = \tau_{AC} = \tau_{CA} = \tau_{BC} = \tau_{CB} = \tau_{BD} = \tau_{DB} = 1$, $\tau_{CD} = \tau_{DC} = 2$, as shown in Fig. 5.4(a). |
| | Traffic resource capacity | $\forall t : C_{AB,t} = C_{BA,t} = C_{BC,t} = C_{CB,t} = C_{BC,t} = C_{CB,t} = 1$, $\forall t : C_{AC,t} = C_{CA,t} = C_{CD,t} = C_{DC,t} = 3$, as shown in Fig. 5.4(b). |
| | Active time for nodes | $b(B) = \{3, 4, 5\}$ |
| Agents | | |
| Passing agents $I_P^{AD}$ | Origin | Node A |
| | Destination | Node D |
| | Value of time and location | $v_i^A = 0,\ v_i^B = 0,\ v_i^C = 0,\ v_i^D \sim N(250, 10000)$ |
| | Departure time | $t_i^B$ is distributed uniformly in $\{0, 1, 2\}$ |
| | Deadline | $t_i^E = t_i^B + 4$ |
| Passing agents $I_P^{DA}$ | Origin | Node D |
| | Destination | Node A |
| | Value of time and location | $v_i^A \sim N(250, 10000),\ v_i^B = 0,\ v_i^C = 0,\ v_i^D = 0$ |
| | Departure time | $t_i^B$ is distributed uniformly in $\{2, 3, 4\}$ |
| | Deadline | $t_i^E = t_i^B + 4$ |
| Cruising agents $I_C$ | Origin | Node A |
| | Destination | Node A |
| | Value of time and location | $v_i^A \sim N(50, 400),\ v_i^B \sim N(200, 6400),\ v_i^C \sim N(100, 1600),\ v_i^D \sim N(50, 400)$ |
| | Departure time | $t_i^B$ is distributed uniformly in $\{0, 1, 2\}$ |
| | Deadline | $t_i^E = t_i^B + 6$ |
| | Additional constraints | Spend at least one time step at Node B during active time $b(B)$ |

### Passing and cruising agents

We consider two types of agents, passing and cruising agents, denoted by $I_P$ and $I_C$, respectively, on the condition that $I_P \cup I_C = I$. The main objective of passing agents is to move between places. The origin and destination of passing agents are different. In contrast, the main objective of cruising agents is spending time at some places, enriching their experience by cruising around. The origin and destination of cruising agents are the same. A key motivating example for this setting is the demand response services in tourist sites. Passing agents express the background traffic demand that is represented by morning or evening commuters, and the cruising agents express tourists cruising multiple areas in the network. In the following section, we describe the agents in this analysis in detail.

- Passing Agents

  In our analysis, we introduce passing agents as background traffic. We consider two types of passing agents, namely, traveling from Node A to Node D and vice versa, representing morning and evening commuters. Sets of these agents are denoted by $I_P^{AD}$ and $I_P^{DA}$, respectively, on the condition that $I_P^{AD} \cup I_P^{DA} = I_P$. For each agent $i \in I_P^{AD}$ traveling from Node A to Node D with the value $\boldsymbol{v_i} = \{v_i^A, v_i^B, v_i^C, v_i^D, \}$, we set the values as follows:

  $$v_i^A = 0,\ v_i^B = 0,\ v_i^C = 0,\ v_i^D \sim N(250, 10000), \tag{5.27}$$

  where $N(\mu, \sigma^2)$ means the normal distribution with the mean of $\mu$ and the variance of $\sigma^2$. Similarly, for each agent $i \in I_P^{DA}$, we set the values as follows:

  $$v_i^A \sim N(250, 10000),\ v_i^B = 0,\ v_i^C = 0,\ v_i^D = 0. \tag{5.28}$$

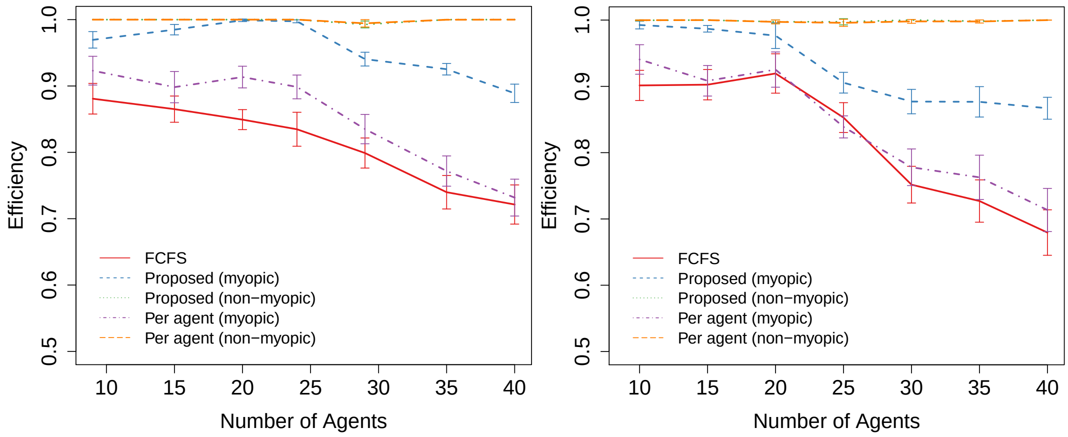
  We assume that the departure time $t_i^B$ of each agent $i \in I_P^{AD}$ is distributed uniformly in $\{0, 1, 2\}$ whereas the departure time $t_i^B$ of each agent $i \in I_P^{DA}$ is distributed uniformly in $\{2, 3, 4\}$. The agents cannot stay at the origin, in that they have to start immediately after they request the trip. We further assume that the deadline for the trip $t_i^E$ is set by $t_i^E = t_i^B + 4$ for all passing agents $i \in I_P$, meaning that passing agents have an upper-limit travel time of 4. However, they have to spend the final time-step to stay at the destination.

- Cruising Agents

  Unlike passing agents, we introduce cruising agents as visitors that have a large amount of flexibility on their trips. The efficiency of traffic services can be improved by using this flexibility effectively. We assume that the origin and destination of all cruising agents is Node A, representing the visitor staying at Node A and intending to cruise the area nearby. The main purpose of the cruising agents is visiting the facility on Node B. Thus, we set the constraints so that all the cruising agents must spend at least one time step at Node B during active time $b(B)$, except when the agents cannot obtain a permit and thus, cancel their trips. We set the value of these agents as follows:

  $$v_i^A \sim N(50, 400),\ v_i^B \sim N(200, 6400),\ v_i^C \sim N(100, 1600),\ v_i^D \sim N(50, 400). \tag{5.29}$$

  We assume that the departure time $t_i^B$ of each agent $i \in I_C$ is distributed uniformly in $\{0, 1, 2\}$ and the deadline for trip $t_i^E$ is set by $t_i^E = t_i^B + 6$, meaning that the cruising agents have an upper-limit travel time of 6.




(a) Fraction of cruising agents = 0.1 (b) Fraction of cruising agents = 0.2

FIGURE 5.5: Efficiency in *static agent-type* setting

In this setting, passing agents traveling from Node A to Node D interfere with the cruising agents when they depart from Node A; passing agents traveling from Node D to Node A interfere with cruising agents when they arrive at Node D. Thus, appropriate control of traffic resources is desired.

### 5.5.2 Results

In this part, we present the results of the numerical analysis and discuss the advantage of our proposed mechanism over the benchmark mechanism.

#### Efficiency

First, we evaluate the *efficiency* of our proposed algorithms in *static agent-type setting*. Here, we define *efficiency* as the total social welfare achieved by each mechanism as a proportion of the offline-optimal welfare. We consider two cases, with the ratio of 10% or 20% of cruising agents over all agents, and vary the number of agents from 10 to 40. The experiment was repeated 10 trials for each setting. The results of *Proposed* and *Per agent* algorithms in myopic and non-myopic settings are shown in Fig. 5.5. The results in settings where the fraction of cruising agents is 0.1 and 0.2 are shown in Fig. 5.5(a) and Fig. 5.5(b), respectively. The efficiency of the FCFS mechanism is also shown in the same figure. The plots in these figures show the mean value of the trials and the error bars express 95% confidence.

As seen in Fig. 5.5, the efficiency of *FCFS* decreases with the interference of agents. On the other hand, *Proposed (myopic)* achieves better efficiency than *FCFS* does, and *Proposed (non-myopic)* retains almost the same quality as the optimal allocation does. While the proposed mechanism guarantees the minimum service quality to the early-coming agents with space–time prism constraints, it takes later-coming high-valued agents into consideration in its decision-making and thus, achieves high efficiency.

As for the *Per agent* approximation, *Per agent (myopic)* results in losses in the efficiency that is almost near the level of *FCFS*, while *Per agent (non-myopic)* does not result in losses in efficiency. Although the decision-making process in *Per agent* algorithm is divided for each agent, the information of all agents reporting at the

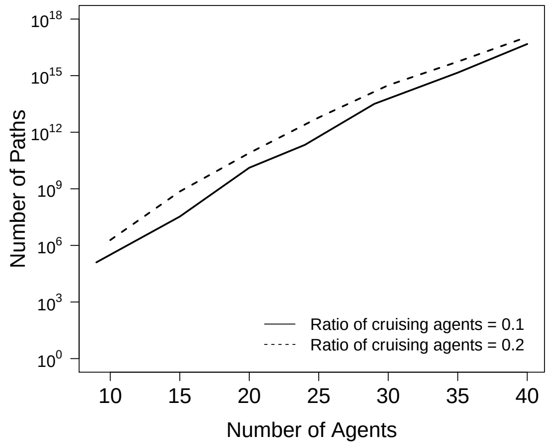
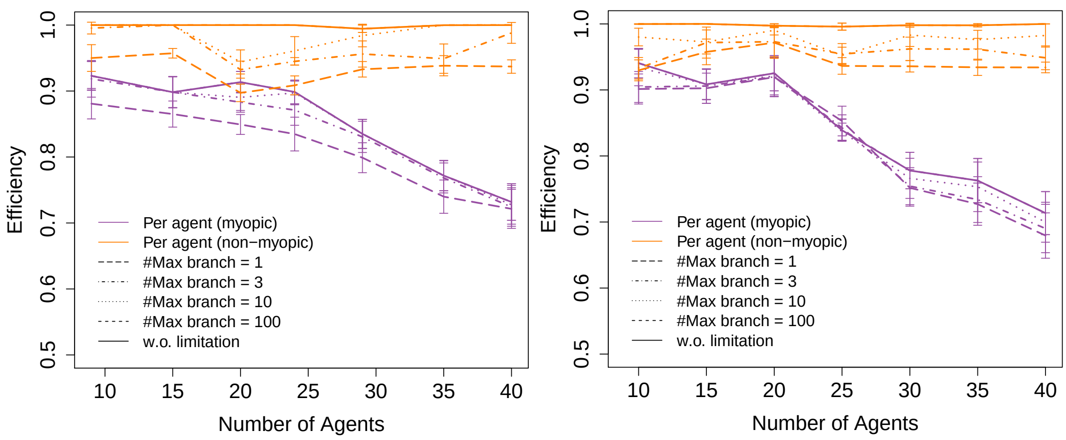



FIGURE 5.6: The total number of executable plans for each setting



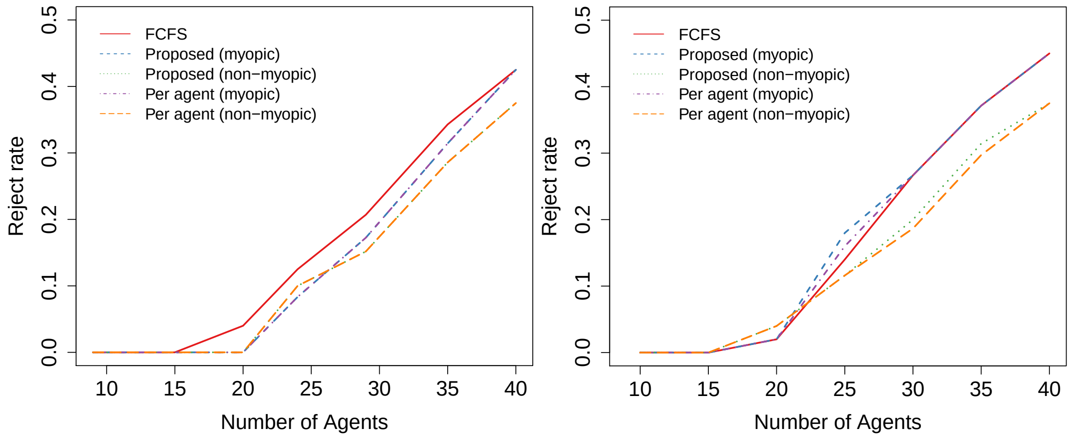

(a) Fraction of cruising agents = 0.1 (b) Fraction of cruising agents = 0.2

FIGURE 5.7: Efficiency of *Branch cutting* algorithms

same time-step is taken into consideration in the non-myopic algorithm and thus, the efficiency loss is suppressed.

In Fig. 5.6, we show the number of enumerated executable plans that keep space–time constraints and capacity constraints considered in the numerical analysis stated above. As the number of agents increases, the number of executable plans also increases extensively, owing to the combinatorial nature of the problem. It reaches $1.13 \times 10^{17}$ when the number of agents is 40 and the fraction of cruising agents is 0.2. *Proposed (non-myopic)* offers a full-search of all these plans for making decisions and thus achieves the highest efficiency. On the other hand, *Proposed (myopic)* makes decisions myopically, but keeps all executable plans in the future, under the given myopic states with all agents reported until that time. This results in higher efficiency when compared to the FCFS.

Then, we evaluate the performance of *Branch cutting* approximation as introduced into *Per agent (myopic)* and *Per agent (non-myopic)* algorithms, running a numerical experiment in the same condition as in Fig. 5.5. For each algorithm, we vary the maximum branch $N_{branch}^{max}$ from 1 to 100, and evaluate the performance. In this analysis, we choose the highest $N_{branch}^{max}$ branches in terms of the expected discounted social welfare at each step of the mechanism. The results are shown in Fig. 5.7.




(a) Fraction of cruising agents = 0.1 (b) Fraction of cruising agents = 0.2

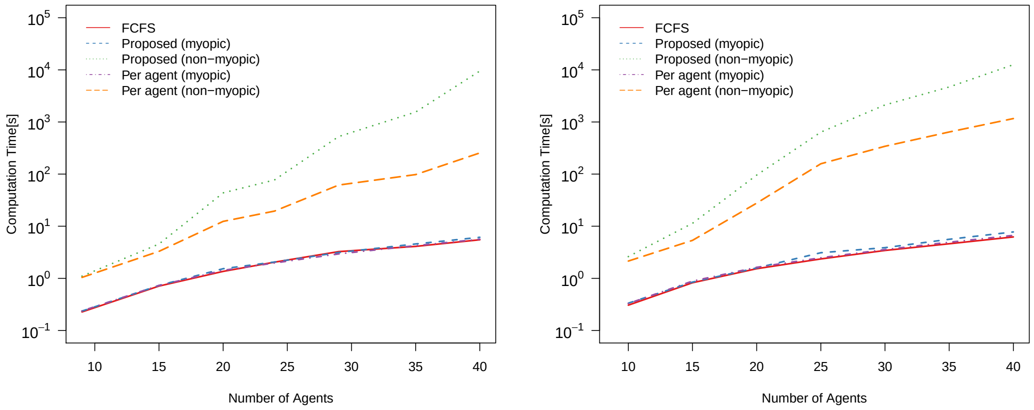
FIGURE 5.8: Rejection rate

Note that, in this figure, the *Per agent (myopic)* algorithm with maximum branch $N^{max}_{branch} = 1$ coincides with the FCFS mechanism. As we can see from this figure, the efficiency decreases as the maximum branch decreases. In these settings, the performance of the *Branch cutting* algorithm with maximum branch $N^{max}_{branch} = 10$ is fairly close to the performance, without the limitation of the maximum branch in myopic settings, but it still has a gap in non-myopic settings. In the setting where the fraction of cruising agents is 0.1 (Fig. 5.7(a)), the results with maximum branch $N^{max}_{branch} = 3$ outperform the results of FCFS in myopic settings. It implies that our proposed algorithms may achieve considerably higher efficiency than FCFS, by keeping just a few options, depending on the situation.

### Rejection rate

In Fig. 5.8, we show the rejection rate as the fraction of the number of rejected agents over the number of all agents, by each algorithm. In Fig. 5.8(a), where the fraction of cruising agents is 0.1, the rejection rates of our proposed algorithms are lower than the FCFS. In contrast, in Fig. 5.8(b), where the fraction of cruising agents is 0.2, the rejection rates of our proposed non-myopic algorithms are lower, but those of our proposed myopic algorithms are higher than FCFS. In both cases, the rejection rate of our proposed non-myopic algorithms are higher than myopic algorithms when the number of agents is small, in the range from 20 to 25 in Fig. 5.8(a) or from 15 to 20 in Fig. 5.8(b). We can see from this, that the non-myopic mechanism makes decisions to reject early-coming agents to keep traffic resources for later-coming high-value agents in order to achieve high efficiency.

As we can seen from Fig. 5.8, in our experimental settings, the rejection rate is considerably high and exceeds 0.4 in the largest settings. It seems to be too high if the discussion is based on an emerged demand, for example, the actual records of taxi services in the real world. However, in this chapter, we aim to consider hidden demands that have not emerged because users give up on traveling before they request the service, making judgments that the trip would possibly suffer their space–time constraints. Considering that such potential users may become customers if the mobility services are improved, the rejection rates in Fig. 5.8 are rational. In such settings with high rejection rates, it is difficult to solve the combined problems of finding a combination of feasible agents and allocating traffic resources for those




(a) Fraction of cruising agents = 0.1 (b) Fraction of cruising agents = 0.2

FIGURE 5.9: Calculation Time

agents using ILP. Graph-based algorithms, including our proposed algorithms using the ZDD, are appropriate in these settings.

### Calculation time

We show the calculation time of the proposed mechanism in Fig. 5.9. All the analysis was performed on a machine with an Intel(R)Xeon(R)E5-2690v4 CPU@2.60GHz, using a single core, and 56GB of RAM. As we can see from these figures, the calculation time of the proposed non-myopic algorithm becomes large as the number of agents increases, while that of the proposed myopic algorithm stays small. However, considering that the non-myopic algorithm treats more than a trillion options (as shown in Fig. 5.6) of the future combinatorial behavior of agents and repeats the optimization process for each action plan and each sample scenario, the calculation time is still small, owing to the nature of the ZDD.

As we have shown, our proposed non-myopic algorithm takes a large amount of computation time, but has high performance, while the basic FCFS mechanism can compute faster, but results in poor performance. Within the trade-off between the performance and computation time, our proposed framework offers a wide range of options. We can achieve better performance by adopting a myopic mechanism with an appropriate maximum branch. Depending on the situation, such as for example, a car-sharing service for limited pre-described customers in which we have enough time to compute, we can introduce a non-myopic mechanism to achieve better performance.

### Payment

First, we demonstrate the numerical analysis in a setting with five cruising and two passing agents, and examine the reward, payment, and utility of all agents under the dynamic pivot and online VCG mechanisms. The results are shown in Table 5.3. Because both mechanisms execute the same *RC-optimal* allocation, the rewards the agents obtain are the same. The dynamic pivot mechanism charges the externality that each agent gives each time and, thus, the payment of agents is (weakly) positive.

TABLE 5.3: Example result for the two pricing algorithms

| Agent ID | Reward | Dynamic Pivot Payment | Dynamic Pivot Utility | Online VCG Payment | Online VCG Utility |
|---|---|---|---|---|---|
| 0 | 288 | 172 | 116 | 148 | 140 |
| 1 | 948 | 464 | 484 | 440 | 508 |
| 2 | 293 | 207 | 86 | 207 | 86 |
| 3 | 469 | 218 | 251 | 218 | 251 |
| 4 | 207 | 6 | 201 | -319 | 526 |
| 5 | 328 | 17 | 311 | 0 | 328 |
| 6 | 548 | 63 | 485 | 112 | 436 |
| 7 | 204 | 0 | 204 | -14 | 218 |
| Summation | 3285 | 1147 | 2138 | 792 | 2493 |

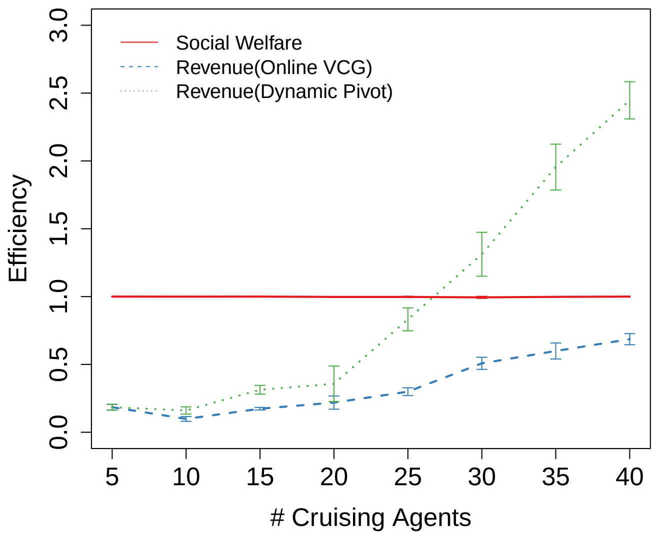



FIGURE 5.10: Revenue achieved under the two pricing algorithms

By contrast, the online VCG mechanism fixes the maximum payment at the beginning, when agents request trips, and compensates dis-utility if their plan has to be changed due to high-value later-coming agents. Therefore, the payment of agents may become negative, as per Table 5.3.

Second, we show the achieved revenue by the dynamic pivot and online VCG mechanisms, varying the number of agents from five to 40 in Fig. 5.10. The fraction of cruising agents is set to 10% of all agents. The achieved social welfare by the *RC-optimal* allocation is shown by the red line in the figure, and the revenues achieved by the two algorithms are shown together. As per Eq. 5.12, social welfare is defined as the summation of the utilities of all agents and the revenue of the operator. Therefore, the total utilities of all agents can also be identified from this figure as the difference between the social welfare and revenue. Because the achieved social welfare is the same for both algorithms, there is a trade-off between the utility of agents and revenue of the operator. As per Fig. 5.10, revenue under the online VCG mechanism is lower than that under the dynamic pivot mechanism. When the number of cruising agents increases, the revenue of the dynamic pivot sometimes exceeds social welfare, meaning that the summation of total utility for all agents becomes negative. On the other hand, the revenue for online VCG may become negative when the number of cruising agents is small. These results express the nature of these two algorithms well, as stated in Section 5.4.

## 5.6 Conclusions and discussion

In this study, we addressed dynamic traffic resource allocation problems with strict capacity constraints and elastic demand of users with space–time prism constraints who require the guarantee of service quality in the worst cases. We characterized this problem by using activity-based user model, in which the relationship between the successive transfers generated from activities was considered. In many settings, this problem includes many non-linear constraints. Thus we take an approach that does not use the ILP, but uses graph algorithms.

In such a setting, we characterize mechanisms that strictly keep both, space–time prism constraints of customers and capacity constraints of traffic resources. We also showed the RC-optimal mechanism that maximizes the discounted social welfare by keeping these constraints.

In several studies, we showed that our proposed model can keep more than trillions of combinatorial trip options of current and future agents in rational computational time, and can thus be used effectively in settings with high rejection rates, meaning that this mechanism can be used for focusing on the behavior of latent customers who have not used mobility services so far. Specifically, our proposed framework is effective as a demand prediction tool that can consider induced demands depending on the service quality.

We also showed that our proposed *Per agent* approximation algorithm when introduced to the non-myopic algorithm, is effective. Commonly, the operator can achieve high efficiency by treating agents reporting in certain durations simultaneously in its decision-making, but it requires high computational costs. Our proposed mechanism solves this problem by making decisions at each agent using information from multiple agents. This approach is effective in designing mobility services with a fixed small number of customers, such as car-sharing systems with specified residents. In such settings, the space–time prism constraints of customers can be highly predictable by observing the daily active patterns of specified customers and thus, the non-myopic algorithm is suitable.

Moreover, we showed that the *Per agent* approximation algorithm introduced to the myopic algorithm performs much better than the common FCFS mechanism does, in settings where the type of agents changes dynamically. This approach is effective in designing mobility services for unspecified many customers, such as shared taxi systems in large cities. In such settings, our proposed floating booking system can provide a lot of flexibility to customers. Customers can arbitrarily change the booking as far as it does not bother other users, and in case the change is rejected, the space–time constraints that the customer originally registered are still guaranteed.

We further discuss the pricing schemes of transportation services such as MaaS. We focus on the connectivity of trips by heterogeneous users and identify a framework of activity-based trip-chain auctions. Using the proposed framework, the service operator realizes a *floating* type booking system that gives priority flexibly to late-coming high-value users by providing incentives to the early-coming low-value users. Specifically, we show a class of mechanisms under which the socially optimal states are achieved by the Bayesian–Nash equilibrium under the best-response strategy of all agents, while satisfying the strict space–time constraints of users and capacity constraints of traffic resources at any time. It is noteworthy that users are guaranteed to increase their utility by reporting more options that can be acceptable within their own space–time constraints. This leads the operator to allocate traffic resources more efficiently. In the numerical analysis, we examine two mechanisms—the dynamic pivot and online VCG—and discuss the difference between them. The dynamic pivot mechanism guarantees that the *ex-post* revenue of the operator is non-negative, while the *ex-post* total utility of all agents may become negative. By contrast, the online VCG mechanism guarantees that the *ex-post* utility of each agent is non-negative, while the *ex-post* the revenue of the operator may become negative. Considering transportation services, the decision of users of whether they execute a trip or not is important, because once users start their trips they cannot arbitrarily leave from the networks. From this viewpoint, the online VCG mechanism, which provides important information on the maximum payment, is suitable for transportation services. However, the online VCG mechanism cannot treat the *dynamic* type of agents, as opposed to the dynamic pivot mechanism. It is also important in transportation services that the user has flexibility on trips. In this respect, the dynamic pivot is superior to the online VCG mechanism. In future work, we plan to explore the pricing scheme that satisfies both of these properties. The price-based mechanism proposed by Hayakawa et al. (2018) is a possible direction to achieve this scope. Further, we only consider the mechanism design of the demand side and ignore the supply side in this study, but the latter should also be considered in future studies.

# Chapter 6

# Conclusions and Future Works

Through the whole part of this thesis, we have discussed the dynamic capacity control of traffic resources, focusing on the behavior of users. In a coming age with the penetrations of automated and/or shared vehicles, the way of mobility services may change drastically. The administrator or operator of the service will have to focus much on the behavior of users that react to the services. In Chapter 3, we introduced a time-delayed control variables to consider the congestion arises one intersection after another along the time axis. Using the variables, we proposed a traffic control algorithm that prevents the gridlock phenomena and efficiently utilizes the limited road resources for a large quantity of demand in a dense city. In Chapter 4, we proposed a traffic service algorithm that efficiently utilizes limited traffic resources while considering the heterogeneity of decision making by users, which develop along the time axis. In Chapter 5, we proposed a dynamic pricing algorithm to induce the user's dynamic decision making to the system optimal state. All of these problems are difficult to solve by existed works targeting aggregate traffic volume and trip-based demands. In this thesis, we thus tried to understand the dynamic characteristics of traffic phenomena developed along the time axis. We formulated the desired traffic controls and transportation services and proposed solution algorithms for them.

There remained plenty of challenges for establishing a basic policy of future traffic controls/services. About the traffic control algorithm for unsteady over-saturated road network mentioned in Chapter 3, a series of future works should be done, which consider the sequential route-choice behavior of drivers, self-organized implementation of each closed loop structures, and so on. About the ridesharing service algorithm presented in Chapter 4, future studies are desired, which focus on a various type of space–time constraints. About the multi-modal traffic service algorithm mentioned in Chapter 5, a future work should be studied in detail, which shed light on the reword function of users by exploring the data taken by the real world is desired. Moreover, the computational burdensomeness of the algorithm.

Conventionally, the transportation systems have tried to transport as many people in as short time as possible. The performance of the traffic control algorithms have been evaluated based on the total travel time. The system that can minimize the total travel time of all users has been regarded as a good system. The heterogeneity of users is not taken into consideration in the process, and the care for vulnerable users such as elderly people or children is not enough. The mobility services in coming ages uses a plenty of computing resources and a large amount of data to provide customized experience for each user. It may be different between urban and suburban areas or vulnerable and non-handicapped users. To this end, it is necessary to develop a technology to estimate the value of each user's activity, using accumulated data of users who repeatedly use the service. It is also desirable to develop a recommendation system that presents customized trip plans for each user. Such recommendation systems provide an alternative set of activities that users have not been aware of, which may increase not only the utility of the use but also the utility of the whole society.

Furthermore, a work that attempts to implement such basic concept to the real world is also needed. To do so, in addition to deepening the basic research, many steady works is desired, e.g., collecting the various needs in the community that the service will be introduced in. We would like to contribute our effort to solve such problems, in order to realize a good mobility scene for everybody, including elderly people and young children, in a coming age.

# Bibliography

Aboudolas, K., Papageorgiou, M., Kosmatopoulos, E., 2009. Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. Transportation Research Part C: Emerging Technologies 17, 163–174.

Aboudolas, K., Papageorgiou, M., Kouvelas, A., Kosmatopoulos, E., 2010. A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks. Transportation Research Part C: Emerging Technologies 18, 680–694.

Abu-Lebdeh, G., Benekohal, R., 1997. Development of traffic control and queue management procedures for oversaturated arterials. Transportation Research Record: Journal of the Transportation Research Board , 119–127.

Agatz, N., Erera, A., Savelsbergh, M., Wang, X., 2012. Optimization for dynamic ride-sharing: A review. European Journal of Operational Research 223, 295–303.

Akamatsu, T., 1995. Forecasting and control of dynamic transportation network flows. Journal of Japan Society of Civil Engineers 13, 23–48.

Akamatsu, T., 1996. Cyclic flows, markov process and stochastic traffic assignment. Transportation Research Part B: Methodological 30, 369–386.

Akamatsu, T., Wada, K., 2017. Tradable network permits: A new scheme for the most efficient use of network capacity. Transportation Research Part C: Emerging Technologies 79, 178–195.

Allsop, R.E., 1974. Some possibilities for using traffic control to influence trip distribution and route choice, in: Transportation and traffic theory, proceedings.

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., Rus, D., 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proceedings of the National Academy of Sciences 114, 462–467.

Arentze, T., Timmermans, H., 2004. Multistate supernetwork approach to modelling multi-activity, multimodal trip chains. International Journal of Geographical Information Science 18, 631–651.

Asadi, B., Vahidi, A., 2011. Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time. IEEE transactions on control systems technology 19, 707–714.

Asghari, M., Deng, D., Shahabi, C., Demiryurek, U., Li, Y., 2016. Price-aware real-time ride-sharing at scale: an auction-based approach, in: Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM. p. 3.

Axhausen, K.W., Gärling, T., 1992. Activity-based approaches to travel analysis: conceptual frameworks, models, and research problems. Transport reviews 12, 323–341.

Bellman, R., 1957. Dynamic programming. Princeton UniversityPress.

Benmohamed, L., Meerkov, S.M., 1994. Feedback control of highway congestion by a fair on-ramp metering, in: Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on, IEEE. pp. 2437–2442.

Bergemann, D., Välimäki, J., 2010. The dynamic pivot mechanism. Econometrica 78, 771–789.

Bonomi, F., Milito, R., Zhu, J., Addepalli, S., 2012. Fog computing and its role in the internet of things, in: Proceedings of the first edition of the MCC workshop on Mobile cloud computing, ACM. pp. 13–16.

Bonsall, P., 2004. Traveller behavior: Decision-making in an unpredictable world, in: Intelligent Transportation Systems, Taylor & Francis. pp. 45–60.

Bonsall, P., Firmin, P., Anderson, M., Palmer, I., Balmforth, P., 1997. Validating the results of a route choice simulator. Transportation Research Part C: Emerging Technologies 5, 371–387.

Brackstone, M., McDonald, M., 1999. Car-following: a historical review. Transportation Research Part F: Traffic Psychology and Behaviour 2, 181–196.

Burns, L.D., 2013. Sustainable mobility: a vision of our transport future. Nature 497, 181–182.

Cavallo, R., Parkes, D.C., Singh, S., 2009. Efficient mechanisms with dynamic populations and dynamic types. Unpublished manuscript, Harvard University .

Ceylan, H., Bell, M.G., 2004. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. Transportation Research Part B: Methodological 38, 329–342.

Chang, H.S., Givan, R., Chong, E.K., 2000. On-line scheduling via sampling, in: Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems, AAAI Press. pp. 62–71.

Chang, T.H., Lin, J.T., 2000. Optimal signal timing for an oversaturated intersection. Transportation Research Part B: Methodological 34, 471–491.

Chang, T.H., Sun, G.Y., 2004. Modeling and optimization of an oversaturated signalized network. Transportation Research Part B: Methodological 38, 687–707.

Chen, B., Cheng, H.H., 2010. A review of the applications of agent technology in traffic and transportation systems. IEEE Transactions on Intelligent Transportation Systems 11, 485–497.

Chen, O., Ben-Akiva, M., 1998. Game-theoretic formulations of interaction between dynamic traffic control and dynamic traffic assignment. Transportation Research Record: Journal of the Transportation Research Board , 179–188.

Chen, Z., He, F., Yin, Y., Du, Y., 2017. Optimal design of autonomous vehicle zones in transportation networks. Transportation Research Part B: Methodological 99, 44–61.

Chen, Z., He, F., Zhang, L., Yin, Y., 2016. Optimal deployment of autonomous vehicle lanes with endogenous market penetration. Transportation Research Part C: Emerging Technologies 72, 143–156.

Chien, C.C., Zhang, Y., Ioannou, P.A., 1997. Traffic density control for automated highway systems. Automatica 33, 1273–1285.

Chikaraishi, M., 2018. Empirical estimation of temporal utility profiles under time-space prism constraints, in: Proceedings of the 15th International Conference on Travel Behaviour Research (ICTBR).

Chiou, S.W., 2005. Bilevel programming for the continuous transport network design problem. Transportation Research Part B: Methodological 39, 361–383.

Clarke, E.H., 1971. Multipart pricing of public goods. Public choice 11, 17–33.

Cordeau, J.F., Laporte, G., 2007. The dial-a-ride problem: models and algorithms. Annals of operations research 153, 29–46.

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

Daganzo, C.F., 1995. The cell transmission model, part II: network traffic. Transportation Research Part B: Methodological 29, 79–93.

Daganzo, C.F., 1996. The nature of freeway gridlock and how to prevent it, in: International Symposium on Transportation and Traffic Theory, pp. 629–646.

Daganzo, C.F., 2007. Urban gridlock: macroscopic modeling and mitigation approaches. Transportation Research Part B: Methodological 41, 49–62.

D'ans, G., Gazis, D., 1976. Optimal control of oversaturated store-and-forward transportation networks. Transportation Science 10, 1–19.

De Lemos, R., Giese, H., Müller, H.A., Shaw, M., Andersson, J., Litoiu, M., Schmerl, B., Tamura, G., Villegas, N.M., Vogel, T., et al., 2013. Software engineering for self-adaptive systems: A second research roadmap, in: Software Engineering for Self-Adaptive Systems II. Springer, pp. 1–32.

Dial, R.B., 1971. A probabilistic multipath traffic assignment model which obviates path enumeration. Transportation research 5, 83–111.

Donati, F., Mauro, V., Roncolini, G., Vallauri, M., 1984. A hierarchical decentralized traffic light control system. the first realisation" progetto torino", in: Proceedings of the 9th World Congress of the International Federation of Automotive Control, pp. 2853–2858.

Dresner, K., Stone, P., 2005. Multiagent traffic management: An improved intersection control mechanism, in: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, ACM. pp. 471–477.

Dresner, K., Stone, P., 2008. A multiagent approach to autonomous intersection management. Journal of artificial intelligence research 31, 591–656.

Drissi-Kaitouni, O., 1993. A variational inequality formulation of the dynamic traffic assignment problem. European journal of operational research 71, 188–204.

Dumas, Y., Desrosiers, J., Soumis, F., 1991. The pickup and delivery problem with time windows. European journal of operational research 54, 7–22.

Fisk, C., 1984. Game theory and transportation systems modelling. Transportation Research Part B: Methodological 18, 301–313.

Fosgerau, M., Frejinger, E., Karlstrom, A., 2013. A link based network route choice model with unrestricted choice set. Transportation Research Part B: Methodological 56, 70–80.

Frank, M., 1981. The braess paradox. Mathematical Programming 20, 283–302.

Friesz, T.L., Kim, T., Kwon, C., Rigdon, M.A., 2011. Approximate network loading and dual-time-scale dynamic user equilibrium. Transportation Research Part B: Methodological 45, 176–207.

Friesz, T.L., Luque, J., Tobin, R.L., Wie, B.W., 1989. Dynamic network traffic assignment considered as a continuous time optimal control problem. Operations Research 37, 893–901.

Gartner, N., Stamatiadis, C., 1998. Integration of dynamic traffic assignment with real-time traffic adaptive control system. Transportation Research Record: Journal of the Transportation Research Board , 150–156.

Gazis, D.C., 1964. Optimum control of a system of oversaturated intersections. Operations Research 12, 815–831.

Gazis, D.C., Herman, R., Potts, R.B., 1959. Car-following theory of steady-state traffic flow. Operations research 7, 499–505.

Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. Transportation Research Part B: Methodological 42, 759–770.

Groves, T., 1973. Incentives in teams. Econometrica: Journal of the Econometric Society , 617–631.

Hägerstrand, T., 1970. What about people in regional science? Papers in regional science 24, 7–24.

Hall, R.W., Lotspeich, D., 1996. Optimized lane assignment on an automated highway. Transportation Research Part C: Emerging Technologies 4, 211–229.

Han, K., Sun, Y., Liu, H., Friesz, T.L., Yao, T., 2015. A bi-level model of dynamic traffic signal control with continuum approximation. Transportation Research Part C: Emerging Technologies 55, 409–431.

Hara, Y., Hato, E., 2017. A car sharing auction with temporal-spatial od connection conditions. Transportation Research Part B: Methodological .

Hara, Y., Kuwahara, M., 2015. Traffic monitoring immediately after a major natural disaster as revealed by probe data - a Case in Ishinomaki after the Great East Japan Earthquake. Transportation Research Part A 75, 1–15.

Harper, C.D., Hendrickson, C.T., Mangones, S., Samaras, C., 2016. Estimating potential increases in travel with autonomous vehicles for the non-driving, elderly and people with travel-restrictive medical conditions. Transportation research part C: emerging technologies 72, 1–9.

Hartgen, D.T., Fields, M.G., Moore, A.T., 2009. Gridlock and growth: the effect of traffic congestion on regional economic performance. Technical Report. Reason Foundation.

Hato, E., Taniguchi, M., Sugie, Y., Kuwahara, M., Morita, H., 1999. Incorporating an information acquisition process into a route choice model with multiple information sources. Transportation Research Part C: Emerging Technologies 7, 109–129.

Hayakawa, K., Gerding, E.H., Stein, S., Shiga, T., 2015. Online mechanisms for charging electric vehicles in settings with varying marginal electricity costs, in: Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), pp. 2610–2616.

Hayakawa, K., Gerding, E.H., Stein, S., Shiga, T., 2018. Price-based online mechanisms for settings with uncertain future procurement costs and multi-unit demand, in: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 309–317.

Hayakawa, K., Hato, E., 2017. Traffic control in oversaturated networks with closed-loop structures. Journal of JSCE D3 (in Japanese) 73, 1159–1172.

Hayakawa, K., Hato, E., 2018a. Auction-based implementation of traffic services to maximize activity-based social welfare, in: Proceedings of the 7th Symposium of the European Association for Research in Transportation conference (hEART).

Hayakawa, K., Hato, E., 2018b. Dynamic traffic resources allocation under elastic demand of users with space-time prism constraints. arXiv preprint arXiv:1806.10719 .

Hayakawa, K., Hato, E., 2018c. Evaluation of dynamic traffic control in unsteady networks with closed-loop structures, in: Transportation Research Board 97th Annual Meeting, Washington D.C.

He, F., Yin, Y., Shirmohammadi, N., Nie, Y.M., 2013. Tradable credit schemes on networks with mixed equilibrium behaviors. Transportation Research Part B: Methodological 57, 47–65.

Head, K.L., Mirchandani, P.B., Sheppard, D., 1992. Hierarchical framework for real-time traffic control. Transportation research record , 82–88.

Hegyi, A., De Schutter, B., Hellendoorn, J., 2005. Optimal coordination of variable speed limits to suppress shock waves. IEEE Transactions on intelligent transportation systems 6, 102–112.

Ho, S.C., Szeto, W., Kuo, Y.H., Leung, J.M., Petering, M., Tou, T.W., 2018. A survey of dial-a-ride problems: Literature review and recent developments. Transportation Research Part B: Methodological .

Horn, M.E., 2002. Fleet scheduling and dispatching for demand-responsive passenger services. Transportation Research Part C: Emerging Technologies 10, 35–63.

Horowitz, J.L., 1984. The stability of stochastic equilibrium in a two-link transportation network. Transportation Research Part B: Methodological 18, 13–28.

Hu, T.Y., Mahmassani, H.S., 1997. Day-to-day evolution of network flows under real-time information and reactive signal control. Transportation Research Part C: Emerging Technologies 5, 51–69.

Huang, H.J., Lam, W.H., 2002. Modeling and solving the dynamic user equilibrium route and departure time choice problem in network with queues. Transportation Research Part B: Methodological 36, 253–273.

Hunt, P., Robertson, D., Bretherton, R., Royle, M.C., 1982. The SCOOT on-line traffic signal optimisation technique. Traffic Engineering & Control 23.

Hunt, P., Robertson, D., Bretherton, R., Winton, R., 1981. SCOOT-a traffic responsive method of coordinating signals. Technical Report.

Jiang, Y., Li, S., Shamo, D.E., 2006. A platoon-based traffic signal timing algorithm for major–minor intersection types. Transportation Research Part B: Methodological 40, 543–562.

Kamar, E., Horvitz, E., 2009. Collaboration and shared plans in the open world: Studies of ridesharing., in: IJCAI, p. 187.

Kang, J.E., Chow, J.Y., Recker, W.W., 2013. On activity-based network design problems. Transportation Research Part B: Methodological 57, 398 – 418. URL: http://www.sciencedirect.com/science/article/pii/S019126151300091X, doi:https://doi.org/10.1016/j.trb.2013.05.009.

Katrakazas, C., Quddus, M., Chen, W.H., Deka, L., 2015. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. Transportation Research Part C: Emerging Technologies 60, 416–442.

Kirk, D.E., 2012. Optimal control theory: an introduction. Courier Corporation.

Kitamura, R., 1988. An evaluation of activity-based travel analysis. Transportation 15, 9–34.

Kitamura, R., Pas, E.I., Lula, C.V., Lawton, T.K., Benson, P.E., 1996. The sequenced activity mobility simulator (sams): an integrated approach to modeling transportation, land use and air quality. Transportation 23, 267–291.

Kitamura, R., Supernak, J., 1997. Temporal utility profiles of activities and travel: some empirical evidence, in: Stopher, P., and Lee-Gosselin, M. (Eds.) Understanding Travel Behavior in an Era of Change., Elsevier. pp. 339–350.

Kleiner, A., Nebel, B., Ziparo, V.A., 2011. A mechanism for dynamic ride sharing based on parallel auctions, in: IJCAI, pp. 266–272.

Kramer, J., Magee, J., 2007. Self-managed systems: an architectural challenge, in: 2007 Future of Software Engineering, IEEE Computer Society. pp. 259–268.

Kuwahara, M., Akamatsu, T., 2001. Dynamic user optimal assignment with physical queues for a many-to-many OD pattern. Transportation Research Part B: Methodological 35, 461–479.

Lam, W.H., Yin, Y., 2001. An activity-based time-dependent traffic assignment model. Transportation Research Part B: Methodological 35, 549–574.

Li, L., Wang, F.Y., 2006. Cooperative driving at blind crossings using intervehicle communication. IEEE Transactions on Vehicular technology 55, 1712–1724.

Li, L., Wen, D., Yao, D., 2014. A survey of traffic control with vehicular communications. IEEE Transactions on Intelligent Transportation Systems 15, 425–432.

Liu, H., Balke, K.N., Lin, W.H., 2008. A reverse causal-effect modeling approach for signal control of an oversaturated intersection. Transportation Research Part C: Emerging Technologies 16, 742–754.

Liu, J., Kang, J.E., Zhou, X., Pendyala, R., 2018. Network-oriented household activity pattern problem for system optimization. Transportation Research Part C: Emerging Technologies 94, 250 – 269. URL: http://www.sciencedirect.com/science/article/pii/S0968090X17302413, doi:https://doi.org/10.1016/j.trc.2017.09.006. iSTTT22.

Liu, P., Liao, F., Huang, H.J., Timmermans, H., et al., 2015. Dynamic activity-travel assignment in multi-state supernetworks. Transportation Research Part B: Methodological 81, 656–671.

Lo, H.K., 1999. A novel traffic signal control formulation. Transportation Research Part A: Policy and Practice 33, 433–448.

Lo, H.K., 2001. A cell-based traffic control formulation: strategies and benefits of dynamic timing plans. Transportation Science 35, 148–164.

Lo, H.K., Chang, E., Chan, Y.C., 2001. Dynamic network traffic control. Transportation Research Part A: Policy and Practice 35, 721–744.

Lovell, D.J., Daganzo, C.F., 2000. Access control on networks with unique origin–destination paths. Transportation Research Part B: Methodological 34, 185–202.

Lowrie, P., 1982. The sydney coordinated adaptive traffic system-principles, methodology, algorithms, in: International Conference on Road Traffic Signalling, 1982, London, United Kingdom.

Ma, H., Fang, F., Parkes, D.C., 2018. Spatio-temporal pricing for ridesharing platforms. arXiv preprint arXiv:1801.04015 .

Mahmassani, H.S., Saberi, M., Zockaie, A., 2013. Urban network gridlock: Theory, characteristics, and dynamics. Transportation Research Part C: Emerging Technologies 36, 480–497.

McFadden, D., 1978. Modelling the choice of residential location, in: Karlqvist, A., Lundqvist, L., Snickars, F., Weibull, J. (Eds.), Spatial Interaction Theory and Planning Models, North Holland, Amesterdam. pp. 75–96.

McFadden, D., 1989. A method of simulated moments for estimation of discrete response models without numerical integration. Econometrica: Journal of the Econometric Society , 995–1026.

Merchant, D.K., Nemhauser, G.L., 1978. A model and an algorithm for the dynamic traffic assignment problems. Transportation Science 12, 183–199.

Michalopoulos, P.G., Stephanopoulos, G., 1977. Oversaturated signal systems with queue length constraints – II: Systems of intersections. Transportation Research 11, 423–428.

Minato, S.i., 1993. Zero-suppressed BDDs for set manipulation in combinatorial problems, in: Proceedings of the 30th international Design Automation Conference, ACM. pp. 272–277.

Minato, S.i., 2001. Zero-suppressed bdds and their applications. International Journal on Software Tools for Technology Transfer 3, 156–170.

Mitsakis, E., Salanova, J.M., Giannopoulos, G., 2011. Combined dynamic traffic assignment and urban traffic control models. Procedia-Social and Behavioral Sciences 20, 427–436.

Newell, G.F., 1993. A simplified theory of kinematic waves in highway traffic, part I: General theory. Transportation Research Part B: Methodological 27, 281–287.

Nie, Y.M., Yin, Y., 2013. Managing rush hour travel choices with tradable credit scheme. Transportation Research Part B: Methodological 50, 1–19.

Oyama, Y., Chikamatsu, K., Shoji, Y., Hato, E., Koga, M., 2016. Trajectory-oriented traffic management using sequential discount rate: a case study of the Great East Japan Earthquake, in: New transport services from satellite technologies in ITS European Congress 2016.

Oyama, Y., Hato, E., 2017. A discounted recursive logit model for dynamic gridlock network analysis. Transportation Research Part C: Emerging Technologies 85, 509–527.

Papageorgiou, M., 1980. A new approach to time-of-day control based on a dynamic freeway traffic model. Transportation Research Part B: Methodological 14, 349–360.

Papageorgiou, M., Ben-Akiva, M., Bottom, J., Bovy, P.H., Hoogendoorn, S., Hounsell, N.B., Kotsialos, A., McDonald, M., 2007. ITS and Traffic Management. Handbooks in Operations Research and Management Science 14, 715–774.

Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A., Wang, Y., 2003. Review of road traffic control strategies. Proceedings of the IEEE 91, 2043–2067.

Papageorgiou, M., Hadj-Salem, H., Blosseville, J.M., 1991. Alinea: A local feedback control law for on-ramp metering. Transportation Research Record , 58–64.

Papageorgiou, M., Hadj-Salem, H., Middelham, F., 1997. Alinea local ramp metering: Summary of field results. Transportation Research Record: Journal of the Transportation Research Board , 90–98.

Parkes, D.C., 2007. Online mechanisms, in: Nisan, N., Roughgarden, T., Tardos, E., and Vazirani, V. (Eds.) Algorithmic Game Theory, pp. 411–439.

Parkes, D.C., Singh, S.P., 2004. An mdp-based approach to online mechanism design, in: Advances in Neural Information Processing Systems 16 (NIPS), pp. 791–798.

Paruchuri, P., Pearce, J.P., Marecki, J., Tambe, M., Ordonez, F., Kraus, S., 2008. Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games, in: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2, International Foundation for Autonomous Agents and Multiagent Systems. pp. 895–902.

Peeta, S., Ziliaskopoulos, A.K., 2001. Foundations of dynamic traffic assignment: The past, the present and the future. Networks and Spatial Economics 1, 233–265.

Pelzer, D., Xiao, J., Zehe, D., Lees, M.H., Knoll, A.C., Aydt, H., 2015. A partition-based match making algorithm for dynamic ridesharing. IEEE Transactions on Intelligent Transportation Systems 16, 2587–2598.

Pigou, A.C., 1920. The economics of welfare. McMillan&Co., London .

Pita, J., Jain, M., Ordóñez, F., Tambe, M., Kraus, S., Magori-Cohen, R., 2009. Effective solutions for real-world stackelberg games: When agents must deal with human uncertainties, in: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1, International Foundation for Autonomous Agents and Multiagent Systems. pp. 369–376.

Quinn, D., 1992. A review of queue management strategies. Traffic Engineering+ Control 33, 600–605.

Ramaswamy, D., Medanic, J.V., Perkins, W.R., Benekohal, R.F., 1997. Lane assignment on automated highway systems. IEEE Transactions on Vehicular Technology 46, 755–769.

Rathi, A.K., 1988. A control scheme for high traffic density sectors. Transportation Research Part B: Methodological 22, 81–101.

Recker, W.W., 1995. The household activity pattern problem: general formulation and solution. Transportation Research Part B: Methodological 29, 61–77.

Recker, W.W., 2001. A bridge between travel demand modeling and activity-based travel analysis. Transportation Research Part B: Methodological 35, 481–506.

Richard, J.P., 2003. Time-delay systems: an overview of some recent advances and open problems. Automatica 39, 1667–1694.

Robertson, D.I., 1969. 'TANSYT' method for area traffic control. Traffic Engineering & Control 10, 276–281.

ROBERTSON, D.I., 1997. The transyt method of co-ordinating traffic signals. Traffic engineering & control 38, 76–77.

Roncoli, C., Papageorgiou, M., Papamichail, I., 2015a. Traffic flow optimisation in presence of vehicle automation and communication systems–part i: A first-order multi-lane model for motorway traffic. Transportation Research Part C: Emerging Technologies 57, 241–259.

Roncoli, C., Papageorgiou, M., Papamichail, I., 2015b. Traffic flow optimisation in presence of vehicle automation and communication systems–part II: Optimal control for multi-lane motorways. Transportation Research Part C: Emerging Technologies 57, 260–275.

Rosenthal, R.W., 1973. A class of games possessing pure-strategy nash equilibria. International Journal of Game Theory 2, 65–67.

Rust, J., 1994. Structural estimation of markov decision processes. Handbook of econometrics 4, 3081–3143.

Salehie, M., Tahvildari, L., 2009. Self-adaptive software: Landscape and research challenges. ACM transactions on autonomous and adaptive systems (TAAS) 4, 14.

Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S.H., Ratti, C., 2014. Quantifying the benefits of vehicle pooling with shareability networks. Proceedings of the National Academy of Sciences 111, 13290–13294.

Sayarshad, H.R., Chow, J.Y., 2015. A scalable non-myopic dynamic dial-a-ride and pricing problem. Transportation Research Part B: Methodological 81, 539–554.

Sheffi, Y., 1985. Urban transportation networks: equilibrium analysis with mathematical programming methods. Traffic engineering control. Prentice-Hall, ISBN 0-13-93-972 .

Sheffi, Y., Powell, W.B., 1983. Optimal signal settings over transportation networks. Journal of Transportation Engineering 109, 824–839.

Shladover, S.E., Desoer, C.A., Hedrick, J.K., Tomizuka, M., Walrand, J., Zhang, W.B., McMahon, D.H., Peng, H., Sheikholeslam, S., McKeown, N., 1991. Automated vehicle control developments in the path program. IEEE Transactions on vehicular technology 40, 114–130.

Singh, M., Tamura, H., 1974. Modelling and hierarchical optimization for oversaturated urban road traffic networks. International Journal of Control 20, 913–934.

Smith, M., 1978. In a road network, increasing delay locally can reduce delay globally. Transportation Research 12, 419–422.

Smith, M., 1979. Traffic control and route-choice; a simple example. Transportation Research Part B: Methodological 13, 289–294.

Smith, M., Van Vuren, T., 1993. Traffic equilibrium with responsive traffic control. Transportation Science 27, 118–132.

Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G.J., 2000. Protocols for self-organization of a wireless sensor network. IEEE personal communications 7, 16–27.

Sun, D., Benekohal, R.F., Waller, S.T., 2006. Bi-level programming formulation and heuristic solution approach for dynamic traffic signal optimization. Computer-Aided Civil and Infrastructure Engineering 21, 321–333.

Suwansirikul, C., Friesz, T.L., Tobin, R.L., 1987. Equilibrium decomposed optimization: a heuristic for the continuous equilibrium network design problem. Transportation science 21, 254–263.

Tambe, M., 2011. Security and game theory: algorithms, deployed systems, lessons learned. Cambridge University Press.

Ukkusuri, S., Doan, K., Aziz, H.A., 2013. A bi-level formulation for the combined dynamic equilibrium based traffic signal control. Procedia-Social and Behavioral Sciences 80, 729–752.

Van Vliet, D., 1981. Selected node-pair analysis in dial's assignment algorithm. Transportation Research Part B: Methodological 15, 65–68.

Varaiya, P., 1993. Smart cars on smart roads: problems of control. IEEE Transactions on automatic control 38, 195–207.

Vickrey, W., 1961. Counterspeculation, auctions, and competitive sealed tenders. The Journal of finance 16, 8–37.

Wada, K., Akamatsu, T., 2013. A hybrid implementation mechanism of tradable network permits system which obviates path enumeration: An auction mechanism with day-to-day capacity control. Transportation Research Part E: Logistics and Transportation Review 60, 94–112.

Wallace, C.E., Courage, K., Reaves, D., Schoene, G., Euler, G., 1984. TRANSYT-7F user's manual. Technical Report.

Wattleworth, J.A., 1965. Peak period analysis and control of a freeway system. Highway Research Record , 1–21.

Webster, F.V., 1958. Traffic signal settings. Technical Report.

Wu, D., Yin, Y., Lawphongpanich, S., Yang, H., 2012. Design of more equitable congestion pricing and tradable credit schemes for multimodal transportation networks. Transportation Research Part B: Methodological 46, 1273–1287.

Yang, H., Meng, Q., 1998. Departure time, route choice and congestion toll in a queuing network with elastic demand. Transportation Research Part B: Methodological 32, 247–260.

Yang, H., Wang, X., 2011. Managing network mobility with tradable credits. Transportation Research Part B: Methodological 45, 580–594.

Yang, H., Yagar, S., 1995. Traffic assignment and signal control in saturated road networks. Transportation Research Part A: Policy and Practice 29, 125–139.

Yick, J., Mukherjee, B., Ghosal, D., 2008. Wireless sensor network survey. Computer networks 52, 2292–2330.

Yin, Y., 2000. Genetic-algorithms-based approach for bilevel programming models. Journal of Transportation Engineering 126, 115–120.