Two Applications of Intelligent Transportation System

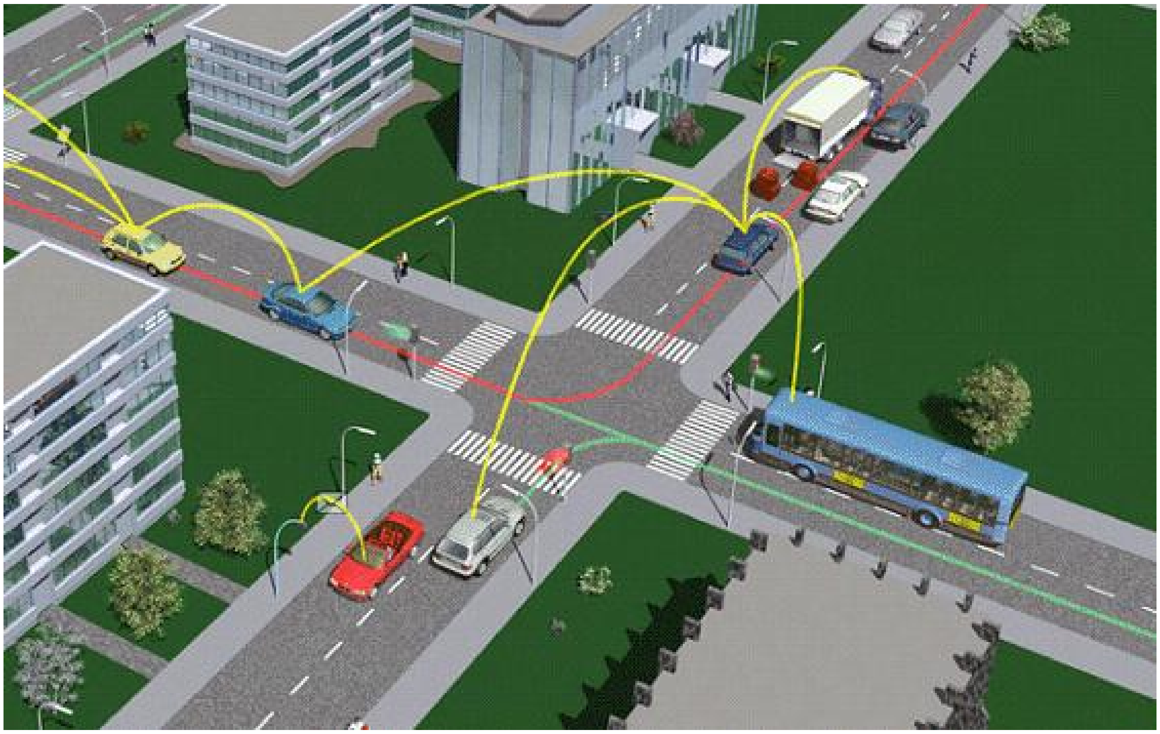
by

Hao Zhou

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Industrial and Operations Engineering)
in The University of Michigan
2013

Doctoral Committee:

Professor Romesh Saigal, Chair
Professor Mingyan Liu
Assistant Professor Gábor Orosz
Assistant Professor Siqian May Shen



To my parents.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my advisor Professor Romesh Saigal. This dissertation would not have been done without the tremendous help and encouragement from him. His insightful guidance and continuous support during the past four years not only makes me a qualified researcher, but also inspires me to be a better person.

I would like to also thank my committee members: Professor Siqian Shen, Professor Mingyan Liu, and Professor Gábor Orosz, for their great efforts and help in my dissertation and defense. My special thanks goes to Dr. Francois Dion, the co-author of my first paper, who gave me a lot of suggestion and advices for the paper.

Also my gratitude goes to all the professors in University of Michigan who have taught me in the past few years. Among others, special thanks to Professor Amy Cohn and Professor Robert Smith, who taught excellent Linear Programming and Dynamic Programming courses when I first come into the university; to Professor Katta Murty, who taught Network Flow class in my first year of study; and Professor Demosthenis Teneketzis from EECS department, whose mechanism design course gave me a lot of ideas and inspiration on my research. It is their excellent teaching and thoughtful help that lay down the foundation of this dissertation, and open the door for my research career.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Two Applications of Intelligent Transportation System

by

Hao Zhou

Chair: Romesh Saigal

We consider here two essential technologies of Intelligent Transportation System (ITS): Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication. In Chapter II, we present a method to automatically control a platoon of vehicles equipped with V2V devices. One of the major issues with platoon control is latency in wireless communications. Latency has a negative impact on safety and disrupts the stability of platoons. We propose a decentralized longitudinal platoon-controlling mechanism that uses a Model Predictive Control (MPC) approach to ensure vehicles safety, even in high-latency communications environments. The sensitivity of this method is analyzed to derive the conditions for the safety of the vehicles in the platoon. A simulation test bed for this control method is implemented to test its effectiveness and safety under two communications latency settings. The results show that the model predictive control method can safely control the platoon even in high-latency communications environments.

In Chapter III, we propose a combinatorial auction implemented via a V2I system to toll and allocate traffic to eliminate congestion on a sub-network of links. We de-

sign a Vickrey-Clarke-Groove (VCG) type auction mechanism, which enables vehicles to bid for paths through V2I devices before entering the network. Using the individual vehicle bids, an optimization problem is formulated and solved, to generate the assignment of vehicles to paths and the corresponding tolls. The underlying model is analyzed for its special properties. We prove that this auction mechanism guarantees truthful reporting and maximizes the social utility. We then test this auction mechanism in two numerical experiments: first with a network of 6 links and 5100 vehicles, and then in a network with 98 links and 12000 vehicles. We prove that in a multiple origin-destination network, it is necessary to add an additional free path for each origin destination pair, in order to guarantee that the toll is always no greater than the bid made by the vehicle. We also discuss various implementation issues of this model, including use of a rolling horizon for multiple-round auctions, and the potential of this auction system as a toll setting mechanism for High-Occupancy Vehicle (HOV) or (High-Occupancy Tolled) HOT lanes.

CHAPTER I

Introduction - Intelligent Transportation System

An Intelligent Transportation System (ITS) incorporates various advanced technologies, such as wireless communication, navigation, sensing, and computing technologies, to make the transportation system safer, more efficient, more 'intelligent', enables more convenient, and informed travel for users. Examples of ITS applications range from basic car navigation and traffic signal control, to more advanced emergency notification and collision avoidance system.

Wireless communication is an essential part of the intelligent transportation system. It enables Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication, which allow the exchange of important traffic information, such as speed, road condition, electronic tolling, etc., among different vehicles, as well as between vehicles and the infrastructure. These wireless communication are implemented on Dedicated Short-Range Communications (DSRC) devices. DSRC provides one-way or two-way short- to medium-range wireless communication channels specifically designed for automotive use and a corresponding set of protocols and standards. Protocols such as IEEE 802.11p, has been developed to enable DSRC communication. A spectrum has been allocated for DSRC use, for example, the 5.850-5.925 GHz band in USA, and the 5.855 MHz to 5.925 GHz band in Europe. In this dissertation, we study two applications of ITS, respectively using V2V and V2I communication.

V2V devices enable the exchange of information among vehicles. Based on these information, in Chapter II, we develop a mechanism that can automatically control a platoon of vehicles. Vehicle platooning is a method of increasing road capacity by grouping vehicles in a tightly-spaced formation. In order to form a safe and efficient platoon, wireless communication has to be used to exchange the accelerating and braking information among vehicles. One of the major issues with platoon control is latency in wireless communications. Latency has a negative impact on safety and disrupts the stability of platoons. We propose a decentralized longitudinal platoon-controlling mechanism that uses a Model Predictive Control (MPC) approach to control vehicles safely, even in high-latency communication environments. The sensitivity of this method is analyzed to derive the conditions for this method to work safely. A simulation test bed for this control method is implemented to test effectiveness and safety under two communication latency settings. The results show that the model predictive control method can safely control the platoon even in high-latency communications environments.

On the other hand, V2I devices can be used to implement an effective tolling mechanism to resolve traffic congestion. In Chapter III, we develop a combinatorial auction system implemented via V2I devices to toll and allocate traffic and reduce congestion. We design a Vickrey-Clarke-Groove (VCG) type auction, which enables vehicles to bid for paths through V2I devices before entering the network. After collecting bids, an optimization problem is solved, and the system assigns vehicles to paths and computes the corresponding toll. A mathematical model of this auction is presented and analyzed. We prove that this auction mechanism guarantees truthful reporting and maximizes the social utility. Then we test the auction in two experiments: first in a network with 6 links and 5100 vehicles, then in a network with 98 links and 12000 vehicles. We prove that in a multiple origin-destination network, an additional free path can guarantee that the toll is always no greater than the bid of a

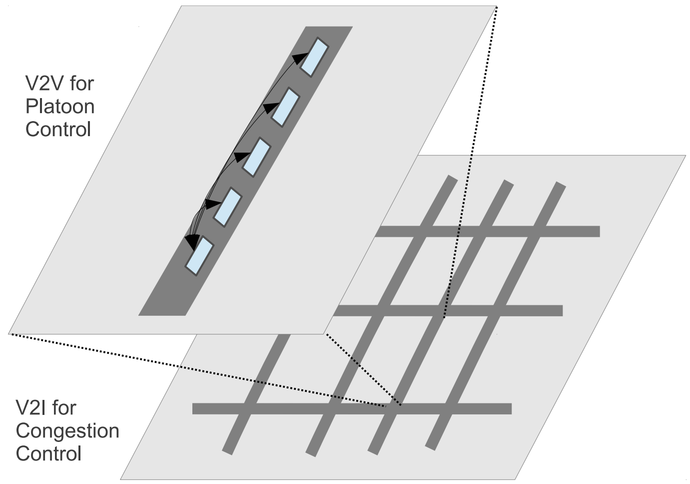


Figure 1.1: Framework of Combined V2V and V2I Applications

vehicle. We also discuss various implementation issues of this model, including using rolling horizon for multiple-round auction, and the potential of this auction system as a toll setting mechanism for HOV or HOT lanes.

These two applications of V2V and V2I can be implemented separately, or combined as an integrated framework for a more efficient and safer transportation system. When these two applications are used together, V2V and V2I applications can be implemented on microscopic and macroscopic level, respectively. Traffic in the microscopic level is optimized through vehicle platooning, using V2V technology, which reduces fuel consumption and improves safety. The macroscopic level traffic is optimized through an auction system and implemented via V2I devices, reducing congestion and travel time of vehicles. More specifically, vehicles would be efficiently assigned to paths (and tolled correspondingly) by the auction mechanism described in Chapter III, and during their trip, vehicles sharing the same path can be grouped into platoons, using the method presented in Chapter II. This framework is illustrated in Figure 1.1.

CHAPTER II

Vehicle Platoon Control in High-Latency Wireless Communications Environment

2.1 Introduction

Due to the ever increasing transportation demand throughout the world, traffic congestion and safety become more and more important issues. One way to reduce the impact of congestion and improve safety is to use Intelligent Transportation System (ITS) (*Horowitz and Varaiya*, 2000; *Varaiya and Shladover*, 1991). The idea is to increase the capacity of highway by automatically coordinating and controlling vehicles to form vehicle platoons, in which vehicles are kept at a small spacing from each other. (*Varaiya*, 1993). To facilitate the exchange of control information, vehicles are equipped with wireless communication devices, also known as Dedicated Short-Range Communication (DSRC) devices. Protocols such as IEEE 802.11p, have been developed to enable vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication (AST, 2003).

The benefit of using ITS includes increased highway capacity, improved safety and increased fuel efficiency. It has been shown that by using accurate sensors and appropriate vehicle control algorithm, one can significantly improve the highway capacity (*Rajamani and Shladover*, 2001). Meanwhile, highway safety can also be improved by broadcasting emergency messages to the entire platoon so that vehicles can brake in advance to avoid collision (*Biswas et al.*, 2001). ITS also has the potential to reduce fuel consumption because vehicle driving are better coordinated through wireless communication, thus reduce the amount of unnecessary acceleration or deceleration for each vehicle (*Baskar et al.*, 2011).

Early attempts to implement the idea of automatic vehicle control was conducted by PATH program at University of California at Berkeley since 1990s. (*Shladover et al.*, 1991) uses the concept of vehicle-follower control (trying to maintain certain spacing with other vehicles) rather than point-follower control (trying to follow markers along the road), to operate vehicles in very close-formation platoons. In this chapter, a hierarchical control scheme was introduced to accommodate the nonlinear dynamics of vehicle mechanical system (engine, transmission and drive-train). This chapter also has a thorough discussion of communication methods and channel capacity requirements.

A safe control system requires sophisticated methods to handle the latency or delays brought by both vehicle's mechanical system and communication systems. Besides mechanical latency, a challenge of developing safe intelligent vehicle control system is to adequately information delay. The relatively narrow radio spectrum and competing nature of wireless communication limits the data rates on the wireless channel (*Gupta and Kumar*, 2000). Moreover, the channel may be noisy and unreliable, due to the reflections and attenuation of the wireless signal being transmitted. These effects inevitably introduce some random delay and packet losses (*Liu et al.*, 2001). Experiments have for instance shown that latency is much higher in urban highway than in open field (*Bai and Krishnan*, 2006), as a result of the signal distortions caused by the structure of buildings and highways.

A high number of vehicles using DSRC devices to exchange information may also eventually cause channel congestion, and thus higher packet loss ratios. According to the study in (*Huang et al.*, 2010), latency also depends on the protocols used to implement the wireless data transmissions. The chapter notably pointed out that a trade-off exists between message transmission rate and packet-loss ratio: if we try to increase the transmission rate, channel congestion will more likely happen, thus increasing the packet-loss ratio, and vice versa. Therefore, a sophisticated wireless channel control is needed to maintain a desirable latency level.

Communication delay may have two negative impacts on the automatic vehicle control system: increased risk of collision and violation of string stability. It has been shown in (*Liu et al.*, 2006) that information delay more than 0.5 seconds would increase probability of collision significantly.

Another issue of controlling a platoon of vehicles is string stability. String stability of a platoon refers to a property that guarantees the spacing error does not amplify as it propagates along a string of vehicles (*Swaroop and Hedrick*, 1996). Control methods were proposed to handle constant information delays using leading and preceding car information (*Rajamani and Zhu*, 2002). However, as is shown in (*Liu et al.*, 2001; *Middleton and Braslavsky*, 2010), such systems do not necessarily create string stable platoons when they only consider information delay with the leading or preceding car. Other method to control a platoon of vehicles includes parallel estimation (*Smith and Hadaegh*, 2007), where each vehicle first estimate the state of entire platoon, and then update its estimates by communicating with other vehicles, and receding horizon control algorithms (*Dunbar and Caveney*, 2012) In this case, a special communication network topology need to be considered to achieve stability.

This chapter is organized as follows. Section II describes the goals and assumptions of the project. Section III describes the model based predictive control (MPC) method we use to control vehicle platoons. Section IV presents an analysis of the effectiveness and robustness of the MPC method, while Section V presents the simulation results of MPC control model. Conclusion and future research are discussed in Section VI.

2.2 Goals and Assumptions

2.2.1 Goals

We want to develop a higher-lever longitudinal control algorithm (details discuss in the next section) for a vehicle platoon that can work under unreliable wireless communication environment and achieve the following goals:

1. Improve highway safety
2. Increase highway capacity
3. Improve energy efficiency

2.2.2 Hierarchy of Control

Two types of control that are crucial to control intelligent vehicles: longitudinal control (throttle and brake) and lateral control (steering). In this chapter, we assume that lateral control can be readily established by a separate controller, and focus primarily on the longitudinal control aspect.

When making longitudinal control decisions, we further assume use of a hierarchical control system implementing an upper and lower control level similar to the one described in (*Rajamani et al.*, 2000). At each time step, it is assumed that the upper level controller determines the desired acceleration for each vehicle based on the following two objectives:

1) maintain appropriate spacing between cars

2) ensure string stability of the platoon

The acceleration decision is made based on perceived state of other vehicles. Because of communication latency, outdated information regarding the position and velocity of surrounding vehicles may be used in the decisions. Once determined, the acceleration or deceleration decision is passed to the lower level controller to be executed.

The lower level controller is responsible for applying the throttle and brake actuator to ensure that the desired acceleration is achieved. The design of lower level controller is a complex problem because we not only need to make many assumptions on the mechanic system of engine transmission and drive-train, but also need to consider the effect on tire and road condition. We also need to be aware of the mechanical latency between the upper and lower level controllers. There exist a lot of literature about the design and analysis of lower level controllers (*Rajamani et al.*, 2000). Here we mainly focuses on the upper-level control, so we assume that a lower level controller is readily built and usable with constant mechanical latency.

2.2.3 Assumptions

In order to solve the vehicle platoon control problem, the following assumptions are made.

1. Full automation.

   In terms of level of automation, there are three major types of systems:

   (a) Emergency warning system that alerts the driver when incidents happen upstream.

   (b) Semi-autonomous cruise control system that can take over certain parts or all of vehicle control, but does not coordinate with other vehicles.

   (c) Fully automatic control that can fully control the vehicles when in the highway. This system will coordinate with other vehicle to maintain a safe distance, and provide steering control to stay within a lane.

   It was argued in (*Varaiya*, 1993) that although a partially automated system may improve safety, only full automation can achieve significant capacities increases. Therefore, we assume that all vehicles are fully controlled by computers.

2. Identical Vehicles.

   To simplify the description of model, we assume that all vehicles within a platoon are identical. However, as is shown in the following section, this assumption can be relaxed by simply replacing a constraint (2.7) in the optimization problem to allow the modeling of different types of cars in the platoon.

3. Decentralized Control.

   Each vehicle has its own controller. In each time step, each car made its own decision on acceleration and steering control. There is no central controller telling each vehicle what to do. However, while making individual decisions, each vehicle still tries to coordinate its actions with neighboring cars.

   The benefit of decentralized control is two-fold: first it requires less communication capacity than centralized control, thus reducing the likelihood of channel congestion. Secondly, the decentralized system is more robust than a centralized system because the overall safety of a platoon is not compromised if one or more controllers fail.

4. Vehicle Spacing Policy

   Each vehicle is required to keep a safe distance from its preceding vehicle. There are many spacing policies we can choose from (*Zhang et al.*, 1999). Among them the constant spacing and constant time headway spacing are frequently used for platoon control. Constant spacing refers to the policy of keeping a constant distance between consecutive vehicles no matter how fast they are traveling. While this policy can achieve very high highway capacity, it may also lead to higher risks of collision when emergency braking occurs.

The spacing policy used in this project is constant time headway policy, which tries to keep the ratio of vehicles spacing and velocity a constant. This policy has been shown in (*Liu et al.*, 2006) to providing a high level of safety.

5. Leading Car Control.

   The platoon control scheme does not include leading car control. We assume that the motion of leading car is exogenous to the model, either controlled by a human driver or an automatic guidance system.

6. Wireless Communication.

   Each vehicle is equipped with IEEE 802.11p (DSRC) transceiver and sends out a message containing its position, speed and acceleration (often known as "Here I am" message). We further assume the following for the communication system:

   (a) For each car, messages are sent every K_s time steps from time 0.

   (b) After each message is sent, it takes τ_0 seconds for encoding and decoding the message.

   (c) When one message is sent, it is received by its designated receiver independently with probability $1-\rho$, where ρ is the message loss rate. We assume this rate ρ is a constant for all sender-receiver pairs and for all time.

2.3 Description of Control Method

2.3.1 Problem Analysis

2.3.1.1 Centralized Control With No Latency

Firstly if we have a platoon of N vehicles, and a centralized controller that has perfect real-time information about every vehicle, we will have the following basic state-space model for (global) control:

$$\dot{X} = AX + BU$$

where $X = [x_1, \ldots, x_N, \dot{x_1}, \ldots, \dot{x_N}]^T$, $U = [\ddot{x_1}, \ldots, \ddot{x_N}]^T$, and x_i is the position of the ith car, A and B are matrix of appropriate size.

This model will work properly if we assume every vehicle can send its information (position, velocity, etc.) to the central controller without any delay. However, in a real world scenario, there is a delay in sending and receiving messages through wireless channel. And usually the delay is a random variable. It has been shown in (*Liu et al.*, 2001) that if the delay is not a constant, then the system is not guaranteed to be stable. Moreover, as is discussed in the previous section, the centralized control makes the whole platoon vulnerable to disruption of wireless communication or failure of central controller. We now establish a decentralized control model to handle these issues.

2.3.1.2 Decentralized Controller With Latency

We assume that every vehicle in the platoon has its own controller. In order to make a control decision, each vehicle's controller needs to know how other vehicles are moving. So we assume every vehicle broadcasts its information to all other vehicles (thus every vehicle also receives information from all other vehicles).

Because of the communication latency, each vehicle may not have the current information about other vehicles, but most likely, has the information sent by other vehicles a fraction of seconds ago. In order to properly handle these out-dated information, we will use a prediction model together with an optimization algorithm, also known as Model-Predictive Control (MPC) method in the next sub-section.

Figure (2.1) illustrates the overview of communication and control scheme of a

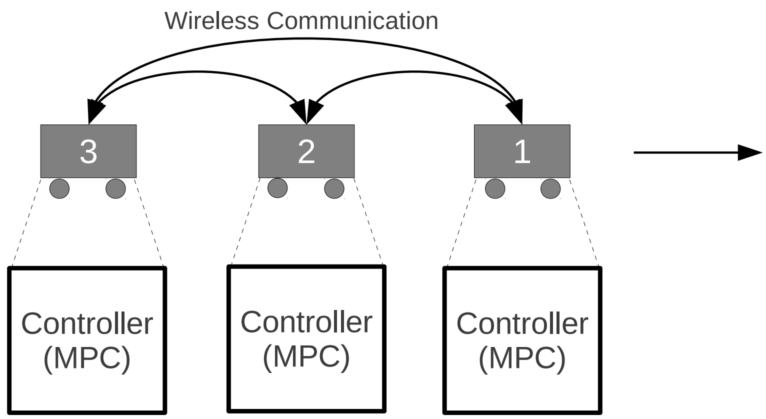


Figure 2.1: Illustration of Communication and Control Scheme

three-vehicle platoon.

2.3.2 MPC Method

The method to be used in controlling vehicles is Model Predictive Control (MPC). MPC has been applied in process industry and other many other control applications (*Maciejowski*, 2002) (*Camacho and Bordons*, 1995), and was implemented for controlling lane allocation of intelligent vehicles (*Baskar et al.*, 2008).

MPC is based on a prediction model and an online optimization to obtain an optimal control actions for the system. Firstly we discretize time horizon and set the sampling period to T. At each time step k, the controller measures the current state of the system, and use a predictive model to predict the system states in the future, i.e., from time step $k+1$ to $k+K_p$, where K_p is the prediction horizon. Then the predictive future states are used as parameters of an optimization problem that minimizes some objective function $J(k)$ over the decision variables $u(k), \ldots, u(k+K_p)$, where $u(k), \ldots, u(k+K_p)$ are control variables. The general process of MPC is shown in Figure 2.2 (also in paper (*Baskar et al.*, 2008)).

In the following sub-sections, we will discuss these components of MPC methods in details.

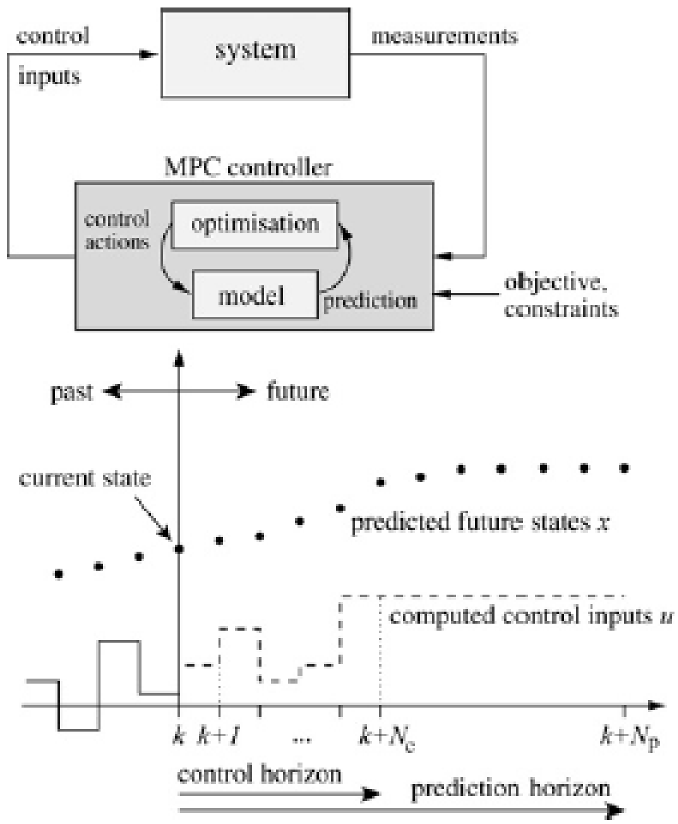


Figure 2.2: Illustration of MPC Control Scheme

2.3.3 Discretization of Time

Usually platoon controlling models are continuous in time (*Rajamani et al.*, 2000). However, we chose the discretized-time model over continuous-time model for the following reasons: (1) discretized model is easier to fit in prediction and optimization algorithm (2) to coincide with the discretized nature of computerized automated control.

Now we discretize the time horizon into time intervals of length T seconds (i.e., sampling period is T), and apply the control action $u(k)$ when time $t = kT, \quad k = 0, 1, \ldots, K_p$, and hold constant within time period $[k, k+1)$. This can be seen in the lower part of Figure (2.2).

Similarly, we assume that messages are sent and received only at time $t = kT, \quad k = 0, 1, \ldots$. This assumptions holds true in practical applications: normally communication devices have “buffers” that can hold messages sent or received during time $((k-1)T, kT)$, and deliver it to the sender or receiver at time kT.

Note that sampling period T can be adjusted according to the needs of different applications. For instance, in the simulation experiments in the following sections, we have chosen $T = 0.05$ after considering the trade-off between accuracy and simulation speed.

2.3.4 Prediction Model to Handle Latency

We now define a set of variables which define the vehicle movement in discrete time: let $x_i(k)$, $v_i(k)$ and $a_i(k)$ be respectively the position, speed and acceleration of the ith vehicle at time kT .

As is discussed before, each car i at time kT sends $x_i(k)$, $v_i(k)$ and $a_i(k)$ to all the other cars in the platoon. But because of the stochastic nature of communication latency, these information arrives at destination car j at time kT with delay $\tau_{i,j}(k)$ time steps, where $j = 1, \ldots, N$. Therefore, at any given time kT, car j has information from different cars with different “ages”. Figure 2.3 demonstrates this asynchronous information transmission: white boxes represent information that is received by car 2, whereas grey boxes represent information not known to car 2. In this case, $\tau_{1,2}(k) = 3$, $\tau_{3,2}(k) = 2$ and $\tau_{4,2}(k) = 3$.

To handle this asynchronous information transmission, we assume every vehicle has a buffer that can hold received information up to $\tau_{\max} + 3$ time steps ago, where $\tau_{\max}$ is the maximum delay counted in time steps. Now we can use this historic information about other cars and a statistical model to “fill in the gaps” created by latency. An effective statistical prediction model can predict how the other cars are moving at the current time step, based on the historical information stored in the buffer.

The statistical model we use to predict the movement of cars in this chapter is ARMAX(3,2,1) (Auto Regressive and Moving Average with eXogenous) model. ARMAX model is a regression model that incorporate past observation of data as well as estimation errors to predict future data series. For example if a vehicle j at time step k wants to know what vehicle i's speed is at the current time step, but it only has speed of car i from period $k - \tau_{\max} - 3$ to period $k - \tau_{i,j}(k)$, we can use the following equation to estimate the speed of vehicles i during period $[k - \tau_{i,j}(k) + 1, k]$:

$$
\begin{aligned}
\hat{v}_{i,j}(\kappa) = & \phi_1 \hat{v}_{i,j}(\kappa - 1) + \phi_2 \hat{v}_{i,j}(\kappa - 2) + \phi_3 \hat{v}_{i,j}(\kappa - 3) \\
& + \epsilon_i(\kappa) - \theta_1 \epsilon_i(\kappa - 1) - \theta_2 \epsilon_i(\kappa - 2) \\
& + \eta_1 \hat{a}_{i,j}(\kappa - 1) \qquad \kappa = k - \tau_{i,j}(k) + 1, \ldots, k
\end{aligned}
\tag{2.1}
$$

where $\hat{v}_{i,j}(\kappa)$ is the estimated speed of car i using car j's information at time κ, and $\hat{a}_{i,j}$ is the estimated acceleration of car i at time κ. $\epsilon_i(k)$ is the estimation error of car i at time t, and the coefficients ϕ, ϵ and η are estimated from data in the buffer: $v_i(k - \tau_{\max} - 3), \ldots, v_i(k - \tau_{i,j}(k))$, using least-square estimate.

Using estimated speed, we can get the estimated positions of all the other cars. These estimation are fed into the optimization problem $P(j, k)$ described below, which is then solved for optimal acceleration of car j at time step k.

2.3.5 Optimization Problem

Under the MPC framework, after we have an estimated speed and position of other cars, the controller of car j will construct an optimization problem to compute the optimal actions for the next K_p time periods. The optimization problem consists of an objective function $J(k)$ indicating the goal we want to achieve during this time period, and a group of constraints, which guarantee that the system is working in a specified manner and within certain conditions.

To achieve lower fuel consumption, one possible objective function is

$$J_\kappa(\mathbf{X}) = \sum_{i=1}^{N}\sum_{k=1}^{K_p}(a_i(k))^2$$

By minimizing this objective function, one can minimize the amount of acceleration (or deceleration) in the following K_p time steps. However, in order to maintain the cars with equal spacing in a platoon, we propose the following objective function:

$$J_\kappa(\mathbf{X}) = \sum_{i=1}^{N}\sum_{k=1}^{K_p}(a_i(k))^2 + W \cdot \sum_{i=1}^{N}\sum_{k=1}^{K_p}(x_i(k) - x_{i-1}(k) - Hv_i(k))^2 \qquad (2.2)$$

where H is the desired time headway (time used to travel the distance between two consecutive cars at current speed), W is the penalty coefficient of deviating from the desired headway. The second term in the objective function penalizes actions that will bring two consecutive vehicles too close or too far away from each other, thus tries to maintain a stable platoon system.

With the objective function (2.2), we define the optimization problem $P(j, \kappa)$ for car j, $j = 1, \ldots, N$ at time step κ as follows:

$$P(j, \kappa): \tag{2.3}$$

$$\text{minimize } J_\kappa(\mathbf{X}) = \sum_{i=2}^{N} \sum_{k=\kappa}^{\kappa+K_p} (a_i(k))^2 + W \cdot \sum_{i=2}^{N} \sum_{k=\kappa}^{\kappa+K_p} (x_i(k) - x_{i-1}(k) - Hv_i(k))^2 \tag{2.4}$$

$$\text{s.t.} \quad x_i(k+1) = v_i(k)T + x_i(k) \quad \text{for } i = 2, \ldots, N, \quad k = \kappa, \ldots, \kappa + K_p \tag{2.5}$$

$$v_i(k+1) = a_i(k)T + v_i(k) \quad \text{for } i = 2, \ldots, N, \quad k = \kappa, \ldots, \kappa + K_p \tag{2.6}$$

$$a_{\min} \leq a_i(k+1) \leq a_{\max} \quad \text{for } i = 2, \ldots, N, \quad k = \kappa, \ldots, \kappa + K_p \tag{2.7}$$

$$Lv_i(k) \leq x_i(k+1) \leq Uv_i(k) \quad \text{for } i = 2, \ldots, N, \quad k = \kappa, \ldots, \kappa + K_p \tag{2.8}$$

$$x_1(k) = \hat{x}_1(k) \quad \text{for } k = \kappa, \ldots, \kappa + K_p \tag{2.9}$$

$$v_1(k) = \hat{v}_1(k) \quad \text{for } k = \kappa, \ldots, \kappa + K_p \tag{2.10}$$

$$x_i(\kappa) = \hat{x}_i(\kappa) \quad \text{for } i = 2, \ldots, N \tag{2.11}$$

$$v_i(\kappa) = \hat{v}_i(\kappa) \quad \text{for } i = 2, \ldots, N \tag{2.12}$$

where $a_{\min}$ and $a_{\max}$ are the minimum and maximum acceleration, respectively. Constraint (2.7) guarantees the maximum acceleration and deceleration will not exceed the car's mechanical limits. For the convenience of demonstration, we set all cars have the same maximum acceleration, but one can easily change the value of these two parameters to apply the model to non-identical vehicle platoon.

The constraints (2.5) and (2.6) keep track of the movement of every vehicle in time and space appropriately. Constraints (2.9) to (2.12) are initial conditions of the model, the right-hand-side of which comes from the prediction results of the ARMAX model in the previous section.

Since this problem has linear constraints and quadratic objective function, it is a quadratic programming (QP) problem. Now we prove this is a convex programming problem.

Theorem II.1. *The objective function (2.4) is a convex function on* $\mathbf{X}$

Proof. To see that the objective function is convex, note that $J_\kappa(\mathbf{X})$ can be expressed as $\mathbf{X}^T Q \mathbf{X}$. Q is a positive semi-definite matrix because it can be expressed as sum of squares of variables. So the objective function is convex. □

Since the objective function is convex and all the constraints are linear, the optimization problem is a convex programming problem, thus can be solved efficiently by QP solver.

2.3.6 Overview of Platoon Control Using MPC

The overview of the control algorithm on car 2 is shown in Figure (2.3).

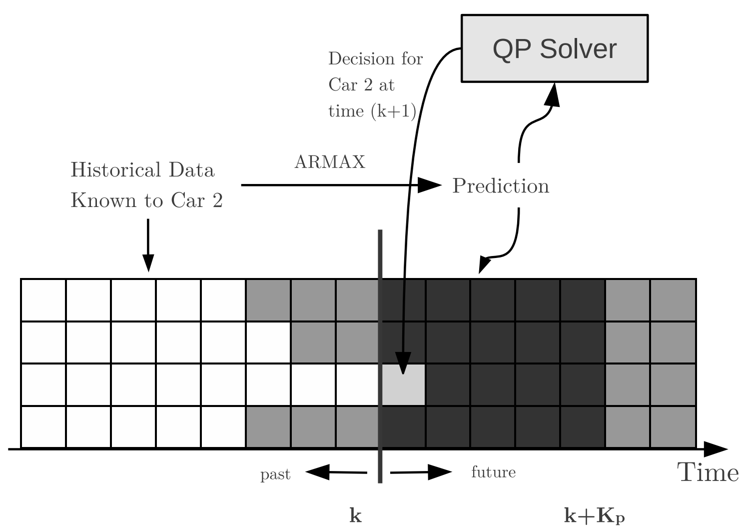


Figure 2.3: Illustration of Decisions made by Car 2 at Time Step k

In the illustration, we assume the platoon has only four cars. Each box in the grid represent the state of one vehicle at certain time step. Vehicle state contains the position, velocity and acceleration. The white boxes represent information already known to car 2, while the grey boxes represent information not known to car 2 due to communication latency. As in the illustration, car 2 knows every movement of itself up to time step k, but only has car 1 and 4's information up to time $k-3$. The black boxes represent predicted vehicle movement in the future, which is generated by ARMAX model. These predicted information will be used as the parameters of QP model $P(j,\kappa)$ described earlier. Then the QP solver will give the optimal acceleration at time $k+1$ for car 2.

2.4 Analysis of Robustness of Control Method

Since we use ARMAX model to predict future states of vehicles, it will inevitably introduce prediction errors into the MPC model. To test whether the MPC method is reliable enough, one needs to test how the prediction error affects subsequent optimization problem solutions. Here we use sensitivity analysis method on optimization problem $P(j,\kappa)$ and investigate how the optimal solution changes as the prediction of speed and position change. The goal of this analysis is two-fold: 1) to test under what conditions, the solutions can be used to maintain a stable system, 2) how large the estimation error can be tolerated by the optimization problem without jeopardizing the safety of the platoon.

2.4.1 General Form of QP

In order to achieve these goals, we can start from a general general form of quadratic programming problem. Firstly, we denote variable y as

$$y = (\mathbf{a}, \mathbf{v}, \mathbf{x})^T$$

where

$$\mathbf{a} = (a_1(k), \ldots, a_1(k+K_p), \ldots, a_N(k), \ldots, a_N(k+K_p))^T$$

$$\mathbf{v} = (v_1(k), \ldots, v_1(k+K_p), \ldots, v_N(k), \ldots, v_N(k+K_p))^T$$

$$\mathbf{x} = (x_1(k), \ldots, x_1(k+K_p), \ldots, x_N(k), \ldots, x_N(k+K_p))^T$$

Then we re-formulate the QP problem $P(j, \kappa)$ defined in (2.4)-(2.12) as follows:

$$\text{minimize}\, y^T Q y \tag{2.13}$$

$$\text{s.t.} \qquad Ay = b \tag{2.14}$$

$$By \leq c \tag{2.15}$$

where A and b are the coefficient matrix of the left-hand-side and right-hand-side of equalities constraint (2.6),(2.5), and (2.9)-(2.12), respectively. B and c are the coefficient matrix of the left-hand-side and right-hand-side of inequalities constraint (2.7)-(2.8), respectively.

2.4.2 Sensitivity Function

Then we apply a small perturbation ϵ to the right-hand-side of equalities constraints, and define the sensitivity function $y(\epsilon)$ as follows:

$$\bar{y}(\epsilon) = \text{argmin}_y\, y^T Q y \tag{2.16}$$

$$\text{s.t.} \qquad Ay = b + \epsilon \tag{2.17}$$

$$By \leq c \tag{2.18}$$

Here the estimation error ϵ is defined as

$$\epsilon = (\hat{x}_1(\kappa), \ldots, \hat{x}_1(\kappa + K_p), \hat{v}_1(\kappa), \ldots, \hat{v}_1(\kappa + K_p), \hat{x}_2(\kappa), \ldots, \hat{x}_N(\kappa), \hat{v}_2(\kappa), \ldots, \hat{v}_N(\kappa))^T$$

The goal is to derive the parametric function $\Delta y(\epsilon)$:

$$\Delta y(\epsilon) = \bar{y}(\epsilon) - \bar{y}(0) \tag{2.19}$$

2.4.3 Boundary-state Analysis

To facilitate our analysis, we start with the following definition:

Definition II.2. We define the system (2.6) - (2.12) at any given time k is in a *boundary state* if there exist $i \in \{2, \ldots, N\}$ such that at least one of the following inequalities is true:

$$\begin{aligned} a_{\min} &< a_i(k) \\ a_i(k) &< a_{\max} \\ Lv_i(k) &< x_i(k) \\ x_i(k) &< Uv_i(k) \end{aligned}$$

Then we make the following assumption:

Claim II.3. When the system is not in boundary state, there exists a small enough perturbation ϵ, such that the system at the next time step is still not in boundary state.

By making this assumption, we can thus assume all of the inequalities (2.7) - (2.8) are strict inequalities under small perturbation ϵ. Now we derive the KKT conditions, which are a system of linear equations:

$$Q(y + \Delta y) + A^T(\mu + \Delta\mu) = 0$$
$$A(\bar{y} + \Delta y) = b + \epsilon$$

Thus the relation of ϵ to Δy can be expressed as:

$$\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y \\ \Delta\mu \end{bmatrix} = \begin{bmatrix} 0 \\ \epsilon \end{bmatrix}$$

Since Q is a singular matrix, we partition it into four blocks:

$$\begin{bmatrix} Q_B & 0 & A_B^T \\ 0 & 0 & A_N^T \\ A_B & A_N & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y_B \\ \Delta y_N \\ \Delta\mu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \epsilon \end{bmatrix}$$

where Q_B is the columns and rows of Q that are not all zeros, and Δy_B are corresponding rows in Δy. We can show that Q_B is a non-singular matrix. Thus we have the following system of equations:

$$Q_B \Delta y_B + A_B^T \Delta\mu = 0$$
$$A_N^T \Delta\mu = 0$$
$$A_B \Delta y_B + A_N \Delta y_N - \epsilon = 0$$

Solving this system by substitution, we have

$$\Delta y_N = (A_N^T (A_B Q_B^{-1} A_B^T)^{-1} A_N)^{-1} (A_B Q_B^{-1} A_B^T)^{-1} \epsilon \tag{2.20}$$

Thus equation (2.20) reveals the linear relation between the error of estimation and deviation from optimal control actions. Thus it indicates how control decision $a_i(k), i = 2, \ldots, N$ is affected by the error of ARMAX model. For any given time, if we have an upper bound and lower bound on estimation error ϵ, we can numerically compute how large the control error can be in each time step.

2.5 Simulation Tests

In order to test whether this MPC method can effectively work under harsh communication environment, we set up a simulation test-bed to test the performance of this algorithm in two different scenarios. We first describe the test-bed setup and some implementation details, then present the results of two test scenarios.

2.5.1 Simulation Setup

2.5.1.1 Data Structure

The simulation test-bed and the control algorithm are implemented in MATLAB. We use three vectors of size N to respectively represent vehicle positions, velocities and accelerations. These vectors combined can be considered as the global system state. On the other hand, each vehicle is programmed as a separate object in the simulation. Each individual object has different "perceptions" of the system state due to different communication latency. Also as is described in the previous sections, each vehicle has a message buffer of size $\tau_{\max} + 3$, storing the history of movement of other vehicles.

2.5.1.2 Generating Latency

Since we assume that each message being sent has an independent loss ratio ρ, we can generate communication latency from vehicle i to j at time kT in the following way:

$$\tau_{i,j}(k) = \tau_0 + nK_s + (k - K_s\lfloor\frac{k}{K_s}\rfloor) \tag{2.21}$$

where n is the number of times a message being sent (or re-sent), and is a geometric distributed random variable with success rate $(1-\rho)$. Because every message is sent only at time step that is a multiple of K_s, thus the term $k - K_s\lfloor\frac{k}{K_s}\rfloor$ indicates how long since last time a message was sent (or re-sent).

2.5.1.3 Simulation Initialization

At the beginning of simulation, we initialize the program by setting the system at "stable state". So we let every vehicle driving at speed of 30 meters per second (67 mph), and at a distance of 30 meters (98 feet) apart. During the first $\tau_{\max}$ periods, we does not apply any acceleration to vehicles, thus let vehicles running at a constant speed. The reason for this initialization is to fill up the message buffer before enabling the ARMAX prediction model to work. After $\tau_{\max}$ time steps, we apply a series of acceleration and deceleration to the leading car, and record how the other cars react.

In the following two sub-sections, we test the MPC method under two scenarios: first in a good communication environment, then a harsh environment. In both cases, we use the same set of acceleration commands for the leading car

Table (2.1) is a brief description of parameter settings and their meanings in the model.

Table 2.1: Parameter Values

Parameter and Value	Meaning
$T = 0.05$ sec.	sampling interval (length of time step)
$N = 4$	number of vehicles in one platoon
$K_p = 10$	number of time steps we consider when solving QP problem
$\tau_{\max} = 50$	maximum delay in wireless communication
$W = 200$	penalty coefficient
$H = 1$ sec.	desired time headway
$L = 0.5$ sec.	minimum time headway
$U = 1.5$ sec.	maximum time headway
$a_{\min} = -12\text{m/s}^2$	minimum acceleration
$a_{\max} = 8\text{m/s}^2$	maximum acceleration
$\tau_0 = 1$	message encoding & decoding delay counted in time steps
$K_s = 2$ or 6	message sent interval
$\rho = 10\%$ or 25%	probability that a message is loss during transmission

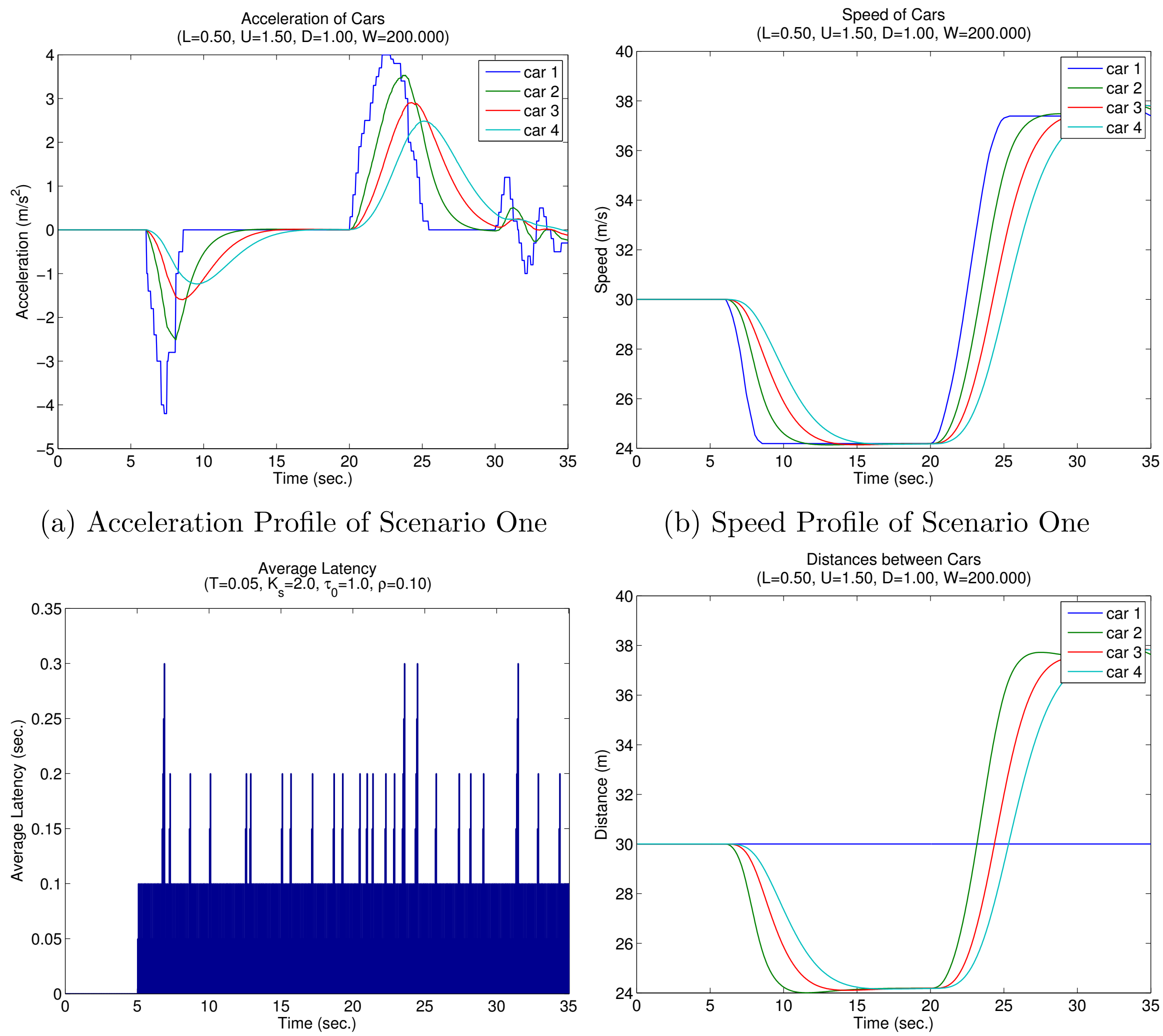
2.5.2 Scenario One - Low Latency Wireless Communication

The first test scenario is to simulate vehicles driving at a "normal" state, with only a few disruptions in wireless channel. Therefore we set $K_s = 2$, meaning that messages are sent every 0.1 seconds. Message loss rate is set at $\rho = 0.1$, which means 10% of the messages are loss during one transmission. Figure 2.4a shows the acceleration of vehicles in the platoon. Note that acceleration of vehicle one is not controlled by the MPC controller, but by pre-specified program input.

Figure 2.4c shows the average latency vehicles experienced along time. Notice that even most of the time latency is at 0.1 seconds, it will occasionally spike to 0.3 seconds. Despite these latency, the effect of sudden braking and accelerating of vehicle one is dampened when it propagates towards the end of platoon. Also Figure 2.4b and 2.4d demonstrate the speed and spacing between vehicles.

2.5.3 Scenario Two - High Latency Wireless Communication

The second test scenario is to test whether the MPC method can withstand noisy wireless communication. Therefore we set $K_s = 6$, meaning that messages are sent every 0.3 seconds. Message loss rate increases to $\rho = 0.25$, meaning that there are



(a) Acceleration Profile of Scenario One

(b) Speed Profile of Scenario One

(c) Latency Profile of Scenario One

(d) Spacing Profile of Scenario One

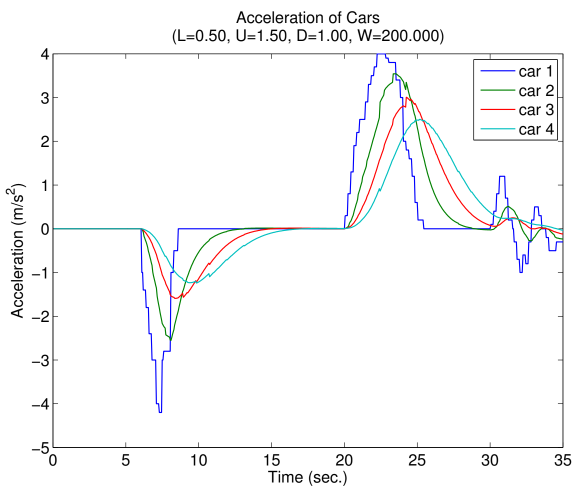
Figure 2.4: Scenario One

25% of chance that a message is loss during transmission. Figure 2.5a shows the acceleration of vehicles in the platoon. Note that acceleration of vehicle one is not controlled by the MPC controller, but is by a program input.

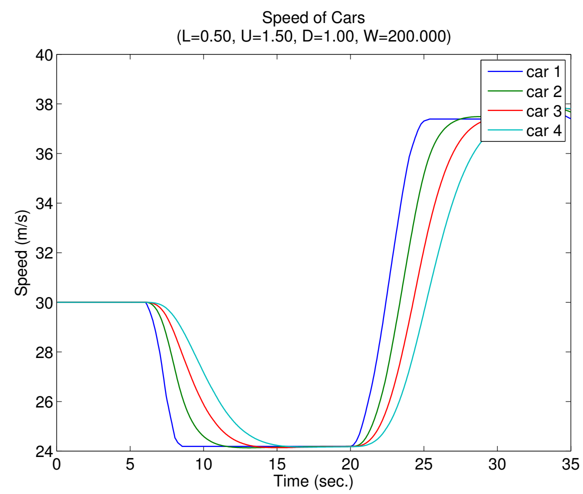
Figure 2.5c shows the average latency vehicles experienced along time. Notice in this case, communication latency is significantly higher compared to the first case: most of the latency numbers are between 0.1 to 0.3 seconds, while sometimes it takes more than 0.9 seconds for a message to reach its destination. Although this is very unlikely to happen in a real world application as is shown in experiments by (*Bai and*

Krishnan, 2006), it does provide a worst case scenario for us to test the robustness of MPC method.

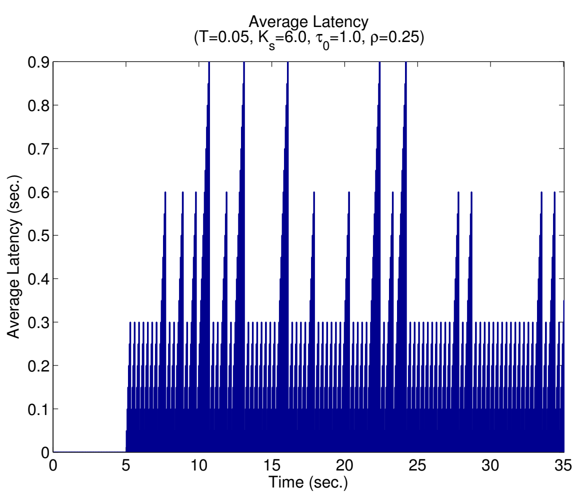
The result indicates large latency does affect the quality of control decisions of vehicle 2: there exists some jiggles in acceleration graph of car 2 and car 3 when the communication latency are high. However, even in these extreme high level of latency, the MPC algorithm still works properly, and operate every vehicle safely.



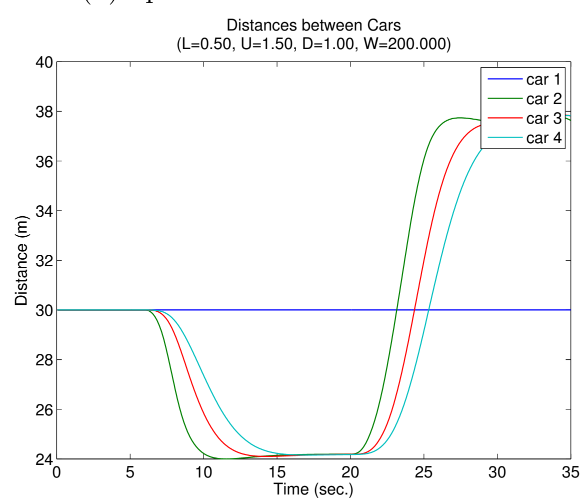
(a) Acceleration Profile of Scenario Two



(b) Speed Profile of Scenario Two



(c) Latency Profile of Scenario Two



(d) Spacing Profile of Scenario Two

Figure 2.5: Scenario One

2.6 Conclusion and Future Work

We have proposed a decentralized control method for controlling a platoon of vehicles under high-latency communication environment. We use MPC approach which combines a statistical prediction model with an optimization algorithm and give optimal control action for each time step. We also analyze the robustness of this method using sensitivity analysis methods. Simulation experiments are performed to test the effectiveness and safety of this control method. It has shown that the MPC controller can react quickly to sudden braking or accelerating of leading car, and dampen the effect of these actions as it propagates along the platoon. The simulation also demonstrates the potential of this method to operate vehicles safely in high-latency communication environment.

Future research includes more extensive case studies to test controller performance under different parameters settings. Quantitative measurement of the performance of the control method (i.e.,fuel efficiency, safety, ride quality) are needed to compare this model to other existing platoon control schemes. Another research topic is to reduce the size and complexity of optimization problem so that it can be computed efficiently in inexpensive on-board computers.

CHAPTER III

Using Combinatorial Auction and V2I Communication to Allocate Traffic

3.1 Introduction

Traffic congestion is a major problem in many parts of the world. For example, in the United States, the cost of increased travel times and fuel consumption alone is estimated to amount to hundreds of dollars per capita per year (*Schrank and Lomax*, 1999).

Many methods have been proposed for reducing congestion. A commonly studied and implemented method is that of congestion pricing or tolling. A vehicle travelling on a road increases the congestion and thus increases costs on other vehicles, leading to increased social costs. However, in the absence of tolls there is no incentive for individuals to consider the effect of their actions on the system. Tolling, on the other hand, is a price mechanism that shifts the social cost of travelling to individual vehicles, thus makes the traffic system more efficient. The idea of congestion pricing was recognized and advocated by (*Pigou*, 1912), and later promoted by William Vickrey's influential work (*Vickrey*, 1969).

As is stated in Vickrey's work, effective congestion pricing requires that tolls be set according to the severity of congestion. This then requires that tolls be a function of the time, location, type of vehicle, etc. Many scholars have proposed a variety of dynamic congestion pricing schemes. For example, Friesz et. al. present a sophisticated method (*Friesz et al.*, 2007) for dynamic congestion pricing. Even though, unfortunately most of these are computationally intensive and difficult to implement, some of them have been successful in the real world. As an example, in Singapore (*Toh and Phang*, 1997). (*Lindsey and Verhoef*, 2000) provides a good review of these pricing models under various settings, like pricing in network, heterogeneity of users, stochastic congestion and so on.

Another proposed methodology for relieving traffic congestion is the use of auctions. (*Teodorović et al.*, 2008) propose an auction-based congestion pricing scheme, which lets participating vehicles bid for time-slots for travel in the down-town area of a city. However, the auction is only used for controlling overall flow in an area, and does not allocate traffic at the network level.

Emerging technology such as Vehicle-to-Infrastructure (V2I) communication enables direct information exchange between vehicles and traffic controllers. Milanés et. al. propose an approach that uses Vehicle-to-Infrastructure (V2I) communication (*Milanés et al.*, 2012) to manage traffic. In this chapter, we propose a congestion control method based on a combinatorial auction implemented with V2I devices. The auction system determines the toll price according to individual vehicle 'bids', which are collected through a system of V2I devices.

This chapter is organized as follows: a detailed description of the mathematical model and the auction scheme is presented in section 3.2. In section 3.5, we test the auction in a small network with 5100 vehicles. In section 3.3 we analyze the computational complexity of the problem. In Section 3.4, we discuss issues related to the implementation of the auction in the real world.

3.2 Combinatorial Auctions

The fundamental reason for congestion is that too many vehicles compete for limited road resources. In economics, one efficient way to allocate scarce resources is through an auction. In this section, we will describe the motivation and details of an auction mechanism.

3.2.1 Introduction to Combinatorial Auctions

3.2.1.1 Background

Auctions have been used as a mechanism for exchanging goods or service since early human history. In the basic types of auctions, buyers offer bids, while sellers take bids, and then sell the item to the highest bidder. Auctions ask and answer the most fundamental questions in economics: who should get the goods and at what prices? In answering these questions, auction theory has been the most influential and widely studied field in economics of the recent decades.

To understand the idea of combinatorial auctions, it is useful to first look at some commonly used types of auctions. Open descending price auction, also known as Dutch auction, in which auctioneer begins with a high asking price which is lowered until some participant is willing to accept the auctioneer's price. The winning participant pays the last announced price.

Open ascending-bid auction, or English auction, is widely used in selling antiques and artwork, and recently used in online market, such as eBay. Participants bid openly against one another, with each subsequent bid required to be higher than the previous bid. The auction ends when no participant is willing to bid higher, at which point the highest bidder pays their bid.

In both of the aforementioned auctions, multiple buyers are bidding for a single item. The auction we use in here, however, requires multiple buyers (vehicles) bidding for multiple items (roads). This is called a combinatorial auction.

3.2.1.2 Basic Combinatorial Auctions

Combinatorial auctions allows bidders to bid not just for a particular item, but for sets of items, sometimes called bundles.

A general case of combinatorial auction has the following settings (*Bikhchandani et al.*, 2002): Let N be the set of bidders, who are bidding for a set M of items. For every set of objects $S \subseteq M$, let $b_i(S)$ be the bid of agent $i \in N$ on set S. Also denote that $x_i(S) = 1$ if the bundle $S \subseteq M$ is assigned to $i \in N$ and zero otherwise. Then the combinatorial auction can be formulated as an optimization problem:

$$\text{maximize} \sum_{i\in\mathbf{N}} \sum_{S\subseteq M} \quad b_i(S)x_i(S) \tag{3.1}$$

$$\text{s.t.} \sum_{i\in N} \sum_{j\in S} x_i(S) \leq 1 \qquad \forall j \in M \tag{3.2}$$

$$\sum_{S\subseteq M} x_i(S) \leq 1 \qquad \forall i \in N \tag{3.3}$$

$$x_i(S) = 0, 1 \qquad \forall S \subseteq M,\ i \in N \tag{3.4}$$

Constraint (3.2) ensures that overlapping sets of items are never assigned. Constraint (3.3) ensures that no bidder receives more than one subset of items. The objective function maximize the total revenue of the seller by summing up values of all combinations proposed by bidders.

As is seen in the optimization problem, the revenue of seller depends on the bids submitted $v_i(S)$, but there is no guarantee that the submitted bids approximate the actual values that bidders assign to the various subsets. Usually bidders have incentive to lie about these values in order maximize their own benefit, if the mechanism to prevent this is not present.

With carefully designed payment scheme, one can induce all the bidders to bid their true value. The most general class of such auction is characterized by Vickrey, Clarke, and Groves (*Vickrey*, 1961; *Clarke*, 1971; *Groves*, 1973). This is also known as Vickrey-Clarke-Groves (VCG) mechanism. The basic ideas is that after items are assigned to bidders, bidders are asked to pay the "opportunity cost" of winning the , rather than paying the bidding price. The details of computing payment is presented in section 3.2.5.

Examples applications of combinatorial auctions are the FCC spectrum auctions (*Cramton*, 1997), auctions for airport time slots (*Rassenti et al.*, 1982), railroad segments (*Brewer*, 1999), delivery routes(*Sheffi*, 2004) and network routing (*Hershberger and Suri*, 2001).

3.2.1.3 Using Combinatorial Auctions

In the context of using combinatorial auction to solve congestion problem, the items are the set of links, whereas the bidders are vehicles (drivers) who want to use these links. Usually vehicles travel through more than one link in a journey. Thus they must submit bids for "bundles" of links, which corresponds to paths in the traffic network.

Figure 3.1 illustrates an example of using combinatorial auction to allocate road resource to three vehicles.

Table 3.1: Vehicles

Vehicles	Origin	Destination	Bids
1	A	D	(AB,BD), (AB,BC,CD), (AC,CB,BD)
2	A	C	(AC), (AB, BC)
3	A	B	(AB), (AC, CB)

As is shown in table 3.1, vehicle 1 travels from A to D, thus needs to submit bids for four paths (bundles): (AB, BD), (AB, BC, CD), (AC, CB, BD) (assuming that we only consider simple paths, i.e., ignoring all paths that contain cycles). Similarly,

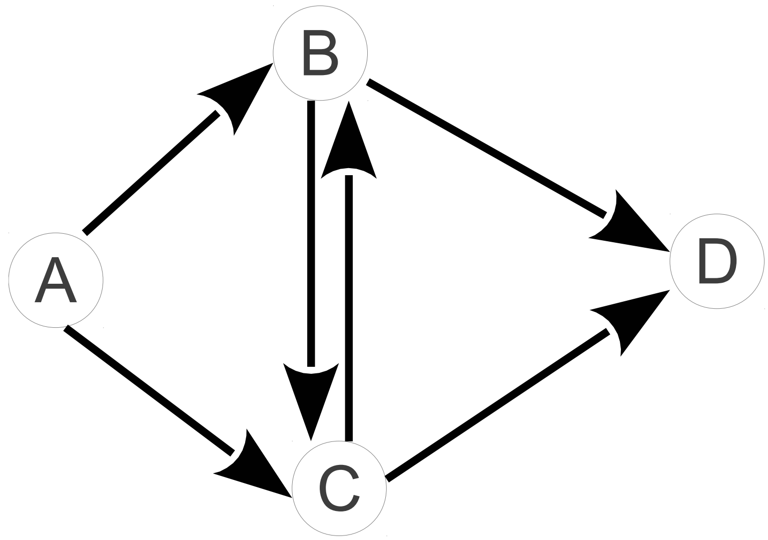


Figure 3.1: Network of Links

for vehicle 2 and 3, they each need to submit 2 bids for their travel.

In this chapter, a Vehicle-to-Infrastructure (V2I) communication system is used implement a combinatorial auction designed to efficiently allocate road resources. This V2I system enables a two-way communication between each vehicle and a central controller: vehicles send their "bids" to the central controller, and then the central controller sends back the path assignment and payment information to vehicles. V2I devices can be pre-installed in vehicles, or more conveniently, run as a specifically designed "apps" on smart-phones of drivers.

This auction is a type of VCG mechanism. In this mechanism, every vehicle submits a bid to the traffic controller for each path it wants to travel. The traffic controller assigns a path to each vehicle, and charges a certain price for taking the path. The price is set in such a way that drivers have no incentive to mis-report their truthful bid for each path. The outline of this mechanism is as follows:

3.2.2 Outline of the Auction Mechanism

The auction system is implemented on a subset of the links in a network. Each vehicle using these links submit its "bids" (the prices s/he is willing to pay) to the central controller. The central controller collects these bids, and solves an optimization problem to assign a path and the corresponding toll to each vehicle. Figure 3.2 illustrates this auction mechanism. On the left of the figure, is the network of links

The auction system works as follows:

1. Before travelling, each vehicle submits a "bid" to the traffic controller via V2I communication. Each bid consists of the following information:
   - Origin and destination of the travel
   - Estimated time to enter the network
   - Price s/he is willing to pay for each potential path s/he can travel.
2. After the submission deadline, traffic controller collects these bids, and uses this information to solve an optimization problem (details discussed in section 3.2.5). The controller then sends back the following instructions to each vehicle:
   - Path to take (path assignment)
   - Toll to pay (payment)
3. At start of travel, the vehicle is automatically charged through electronic devices installed in the car. Vehicles must take the assigned path. A penalty fee will be charged for any deviation from the assigned path.

In the rest of this section, we will address the following questions about the auction mechanism.

- What information must the drivers provide to the controller
- How does the controller determine who will be assigned to which path (path assignment)
- What is the toll a driver pays for taking the path. This can be different from what s/he bid. (payment scheme)

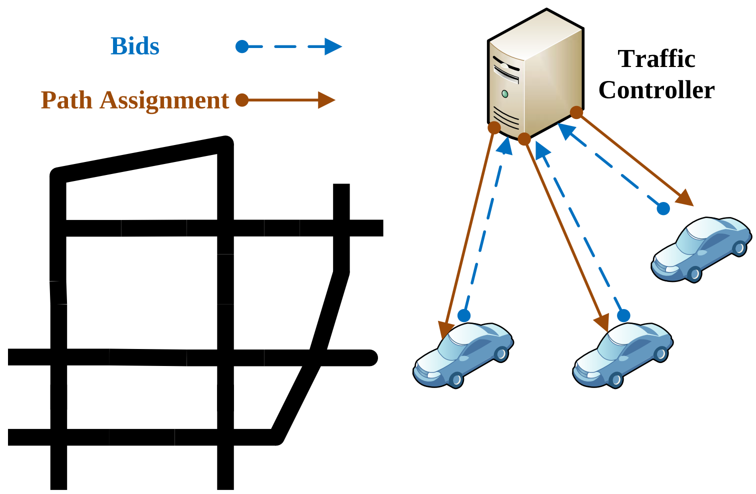


Figure 3.2: Auction Mechanism

- Mechanism to guarantee drivers bid their true valuation
- Is the solution efficient

In this model, we require that ALL vehicles are equipped with V2I devices and the required dedicated software to participate in the auction process. However, this system can also be implemented as a sub-system embedded in a larger network. For example, it can be used as the pricing model for a network of (High Occupancy Vehicle) HOV or HOT (High Occupancy Toll) lanes. In these cases, the requirement that every vehicle be equipped with V2I devices is not required. Details of this issue will be discussed in the section 3.4.

3.2.3 Assumptions

We make the following assumptions:

- Infrastructure

  1. Every vehicle is equipped with two-way V2I wireless devices (will be relaxed later)

2. The bidding process and toll collection is done through wireless communication.

- Traffic Controller

1. No congestion (free flow) for all the links in the auction network
2. Controller has mechanism to prevent drivers from deviating from assigned path (i.e., through penalty)
3. Vehicle must establish communication with controller before entering the network

- Drivers and Vehicles

1. Every vehicle must submit bids for all of the paths it can potentially use
2. All vehicles travel at free flow speed in the network
3. Bids are calculated and submitted by on-board computers
4. Each driver's cost function is independent of other drivers (private value)
5. Once assigned, the driver must accept the path and pay the toll

3.2.4 Mathematical Model

3.2.4.1 Network

Consider a network that consists of a set of $\mathbf{L} = \{1, 2, \ldots, L\}$ of links. For each link $l \in \mathbf{L}$, the free-flow capacity and travel time are denoted as C_l and T_l, respectively. A path is defined as a sequence of links from one origin to one destination. The set of all possible paths is denoted by $\mathbf{P} = \{1, 2, \ldots, P\}$. We denote a path $p \in \mathbf{P}$ as a sequence of links it contains:

$$p \equiv \{a^p_{(1)}, a^p_{(2)}, \ldots, a^p_{(|p|)}\}$$

where $|p|$ is the number of links contained in p, and $a^p_{(i)}$ is the ith link in path p. And we define the travel time of a path p as T^p [1]:

$$T^p = \sum_{k=1}^{|p|} T_{a^p_{(k)}}$$

3.2.4.2 Vehicles

There are N vehicles, denoted by set $\mathbf{N} = \{1, 2, \ldots, N\}$, using the traffic network. Vehicle $i \in \mathbf{N}$ will enter the network at time A_i.

The traffic controller's job is to assign a path to each vehicle. The path assignment for vehicle i is a vector $x_i = (x_i^1, x_i^2, \ldots, x_i^P)$, where $x_i^j = 1$ means vehicle i is assigned to path j, and 0 otherwise. Obviously, a valid assignment x_i requires that $\sum_{j=1}^{P} x_i^j = 1$.

Each vehicle has a utility function $U_i(x_i)$ that maps a valid assignment x_i to a real number:

$$U_i(x_i) : \{0, 1\}^P \rightarrow \mathbb{R}$$

It can be interpreted as the benefit vehicle i gets when travelling under assignment x_i.

Although this utility function can be of any form, for the purpose of demonstrating our model, we use a linear utility function:

$$U_i(x_i) = \sum_{j \in \mathbf{P}} v_i^j x_i^j$$

where v_i^j is the value of travelling in path j, by vehicle i. Note that the actual bid, $\hat{v}_i^j$, made by vehicle i for path j, may be different from the true value v_i^j.

[1] T^p does not change as no congestion is allowed

3.2.4.3 Time

The entire planning period is discretized into a set of intervals of equal length δ, denoted as $\mathbf{T} = \{1, 2, \ldots, T\}$. δ is set small enough so that the travel time of any link in the network is an integer multiple of δ, but not too small so as to make the problem computationally difficult (issues of computation will be discussed in section 3.3).

The typical planning period δT can be set to 24 hours, or to the duration of the peak hours when congestion is likely to happen.

3.2.5 Optimization Problem

Given a path assignment matrix for all drivers, $\mathbf{x} = (x_1, x_2, \ldots, x_N)$, and the bids $\hat{v}_i^j$ for vehicle i and path j, we evaluate the system performance using the sum of the utility of all vehicles (also called social utility function) $\mathbf{U}$:

$$\begin{aligned}\mathbf{U}(\mathbf{x}) &= \sum_{i \in \mathbf{N}} U_i(x_i) \\ &= \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{P}} \hat{v}_i^j x_i^j\end{aligned}$$

Assuming that all links are congestion free, we define the following delay operator τ_l^j for each $l \in \mathbf{L}, j \in \mathbf{P}$:

$$\tau_l^j = \sum_{k=1}^{k: a_{(k)}^j = l} T_{a_{(k)}^j}$$

in other words, τ_l^j is the time to travel to the entrance of link l, given that the vehicle is on path j.

Based on the above social utility function and delay operator τ_l^j, we formulate the following Path Assignment Problem to determine the optimal assignment:

$$\mathbf{U}^* = \text{maximize} \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{P}} \hat{v}_i^j x_i^j \tag{MAX 1}$$

$$\text{s.t.} \sum_{j \in \mathbf{P}} x_i^j = 1 \qquad \forall i \in \mathbf{N} \tag{3.5}$$

$$\sum_{j \in \mathbf{P}} \sum_{\substack{i: A_i > t - \tau_l^j - T_l \\ A_i \leq t - \tau_l^j}} x_i^j \leq C_l \qquad \forall t \in \mathbf{T},\, l \in \mathbf{L} \tag{3.6}$$

$$x_i^j = 0, 1 \qquad \forall i \in \mathbf{N},\, j \in \mathbf{P} \tag{3.7}$$

Constraint (3.5) ensures that each vehicle is assigned to exactly one path. Constraint (3.6) enforce the number of vehicles in each link l at each time period t does not exceed the total capacity of the link. This is done by summing up all the vehicles that have entered, but not yet exited link l at time t. $t - \tau_l^j - T_l$ is the time a vehicle arrives at the entrance of path j (also the entrance of network), given that it reaches the **entrance** of link l at time t. On the other hand, $t - \tau_l^j$ is the time a vehicle arrives at the entrance of path j, given that it reaches the **exit** of link l at time t. Constraint (3.7) ensures that the assignment variable x_i^j can only be zero or one.

On solving (MAX 1), we obtain an optimal assignment that maximizes the social utility function. This assignment is then distributed to individual vehicles via V2I wireless communication devices, informing them of the path to take.

3.2.6 Payment

The optimization problem (MAX 1), generates an optimal solution $\mathbf{x}^*$, determining the path assigned to each vehicle. We now determine the toll price for this assignment. We adopt a scheme similar to traditional VCG mechanism, which determines the toll as an "opportunity cost" it imposes on other vehicles, in other words, the marginal utility price. The procedure of computing toll for vehicle k is as follows:

Define

$$\mathbf{U}_{-k}(\mathbf{x}_{-k}) = \sum_{\substack{i\in\mathbf{N}\\ i\neq k}} \sum_{j\in\mathbf{P}} \hat{v}_i^j \cdot x_i^j$$

where $\mathbf{x}_{-k} = (x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_N)$. $\mathbf{U}_{-k}$ is the utility when vehicle k has been excluded.

We modify the optimization problem (MAX 1) to exclude vehicle k, and call it (MAX 1-k). Let its optimal value to be $\mathbf{U}^*_{-k}$. Thus :

$$\mathbf{U}^*_{-k} = \text{maximize} \sum_{\substack{i\in\mathbf{N}\\ i\neq k}} \sum_{j\in\mathbf{P}} \hat{v}_i^j x_i^j \qquad \text{(MAX 1-k)}$$

$$\text{s.t.} \sum_{j\in\mathbf{P}} x_i^j = 1 \qquad \forall i \in \mathbf{N}, i \neq k \qquad (3.8)$$

$$\sum_{j\in\mathbf{P}} \sum_{\substack{i:A_i>t-\tau_l^j-\tau_l\\ A_i\leq t-\tau_l^j\\ i\neq k}} x_i^j \leq C_l \quad \forall t \in \mathbf{T},\, l \in \mathbf{L} \qquad (3.9)$$

$$x_i^j = 0, 1 \quad \forall i \in \mathbf{N}, i \neq k,\, j \in \mathbf{P} \qquad (3.10)$$

Thus U^*_{-k} is the optimal social utility when vehicle k is not in the system.

If we denote $\mathbf{x}^*_{-k}$ as the optimal solution from (MAX 1), excluding vehicle k, then the toll π_k for vehicle k is

$$\pi_k = \mathbf{U}^*_{-k} - \mathbf{U}_{-k}(\mathbf{x}^*_{-k}) \qquad (3.11)$$

The first term in equation (3.11), is the optimal social utility without vehicle k, and the second term is the social utility of the optimal solution x^* of (MAX 1), without the vehicle k. The difference of these two terms is the increase in social utility when vehicle k is not included in the system, justifying it as a toll for vehicle k.

Note that the first term depends only on the bids of vehicles other than k. This is the desirable feature of VCG mechanism, which generates no incentive for vehicles to mis-report their true value (in other words, this mechanism guarantees $v_i^j = \hat{v}_i^j \quad \forall i \in \mathbf{N}, j \in \mathbf{P}$).

Theorem III.1. *Truthful reporting is an optimal strategy for each vehicle driver in the auction mechanism. Moreover, when each vehicle driver reports truthfully, the outcome of the mechanism is one that maximizes social utility.*

The proof of Theorem III.1 is in Appendix A.1.

3.3 Computational Issues

Both the path assignment problem (MAX 1) and payment problems (MAX 1-k) are Integer Programming (IP) problems, and thus NP-complete. These are also notoriously hard to solve for large problem size. Although medium-size problems like the example we used in section 3.5 can be solved relatively fast, it could take considerably longer to solve larger size problems with more vehicles and larger network. In this section, we will analyze the structure of these problems and propose some methods to reduce the complexity of computation.

3.3.1 Solving Path Assignment Problem

The constraint (3.7) of path assignment problem (MAX 1) requires that all variables be integer, this makes the problem an IP. A typical way of solving IP is to first solve Linear Programming (LP) relaxation of the IP problem and then use branch-and-bound method to find the optimal integer solution.

3.3.1.1 Structure of The Path Assignment Problem

Consider the constraints of the LP relaxation of (MAX 1).

$$\sum_{j\in\mathbf{P}} x_i^j = 1 \qquad \forall i \in \mathbf{N} \tag{3.12}$$

$$\sum_{j\in\mathbf{P}} \sum_{\substack{i:A_i>t-\tau_l^j-T_l\\ A_i\le t-\tau_l^j}} x_i^j + s_{t,l} = C_l \quad \forall t \in \mathbf{T},\, l \in \mathbf{L} \tag{3.13}$$

$$x_i^j \ge 0 \qquad \forall i \in \mathbf{N}, i \neq k,\, j \in \mathbf{P} \tag{3.14}$$

where $s_{t,l}, \forall t \in \mathbf{T}, l \in \mathbf{L}$ are the "slack" variables of each capacity constraint, which represents how many unit of capacity of link l is unused at time t.

This problem has $N \times P$ variables. There are N constraints in the first constraint group (3.12) and $T \times L$ constraints in the second constraint group (3.13). In the context of a Simplex method, a basis consists of $N + TL$ basic variables.

Based on the special structure of the basis, we can prove the following theorem:

Theorem III.2. *Solving the relaxed IP, the number of non-integer variables in any basic solution is bounded by $2TL$. Moreover, if no links are capacitated, the solution of LP relaxation of the problem is integer.*

Proof. Since for each vehicle i, a constraint in group (3.12) can provide at least one basic variable.

On the other hand, each constraint in group (3.13) must provide at least one basic variable.

If none of the links are capacitated, all of the TL slack variables $s_{t,l}$ should be positive, which make them basic variables. The coefficients of these slack variables (all equal to one) form a $TL \times TL$ identity matrix in the basis. In this case, each constraint i in group (3.12)

$$\sum_{j \in \mathbf{P}} x_i^j = 1 \tag{3.15}$$

must provide one, and only one basic variable, thus this constraint group provides an $N \times N$ identity matrix in the basis. So the structure of basis of the LP relaxation is

$$\begin{array}{l}\left[\begin{array}{c|c} I_N & 0 \\ \hline R & I_{TL}\end{array}\right]\begin{array}{l}\left.\vphantom{I_N}\right\} N \\ \left.\vphantom{I_{TL}}\right\} TL\end{array} \\ \;\underbrace{\qquad}_{N}\;\underbrace{\qquad}_{TL}\end{array}$$

where I_N is a $N \times N$ identity matrix, and I_{TL} is a $TL \times TL$ identity matrix. The upper part of the matrix corresponds to constraints in group (3.12), while the lower part of the matrix corresponds to constraints in group (3.13). In this case, since the entire basis is a identity matrix, the optimal solution to the LP relaxation is always integer.

If there are n links at capacity, slack variables corresponding to those links are non-basic, thus there are n more x_i^j as basic variables. These additional x_i^j will make some of the constraints in (3.12) contain more than one basic variable. In the worst case, the structure of the basis is

$$
\begin{array}{cl}
\begin{bmatrix} I_{N-n} & 0 & 0 & 0 \\ 0 & I_n & Q & 0 \\ R_1 & R_2 & R_3 & 0 \\ R_4 & R_5 & R_6 & I_{TL-n} \end{bmatrix} & \begin{array}{l} \}\, N-n \\ \}\, n \\ \}\, n \\ \}\, TL-n \end{array} \\
\begin{array}{cccc} \underbrace{\qquad\quad}_{N-n} & \underbrace{\qquad}_{n} & \underbrace{\qquad}_{n} & \underbrace{\qquad\quad}_{TL-n} \end{array} &
\end{array}
$$

where $R_1, R_2, R_3, R_4, R_5, R_6, Q$ are matrices containing only zero and one as elements. Thus, in the worst case, there will be at most $2n$ basic variables in matrix Q, thus give $2n$ non-integer solutions.

If all of the TL links are at capacity, there will be at most $2TL$ non-integer variables in the LP relaxation of (MAX 1). $\square$

In general, the proportion of non-integer variables is $2TL/NP$. In large network, $T \ll N$ and $L \ll P$, so only a small percentage of variables will be non-integer. In the test case of section 3.5, $(60 \times 4)/(5100 \times 4) = 1.18\%$ of variables will be non-integer at most.

3.3.1.2 Reducing Complexity

One method to reduce complexity of (MAX 1) is to use a smaller number of potential paths for each vehicle, and instead of letting vehicles choose from all of the available paths, we limit their choice to, say, at most six paths.

3.3.2 Solving Payment Problem

Although the initial path assignment problem (MAX 1) may itself be hard to solve, a bigger computational challenge is in solving N instances of payment problems (MAX 1-k).

Now we present two methods to reduce the computation time of solving payment problems:

1. Using approximate solution for payment problem.

   As is shown in (*Nisan and Ronen*, 2000), in order to maintain truthful reporting property of VCG mechanism, (MAX 1) must be solved to optimality, but the solution of (MAX 1-k) need not be optimum. So in order to reduce solving time for payment problems, we can set an optimality gap, say, 2%, when solving the payment problems. The branch-and-bound algorithm will stop when the obtained solution is at most 2% away from actual optimal solution. In this case, vehicles will pay less than what VCG mechanism will prescribe.

2. Solve payment problem only once for vehicles assigned to the same path at the same time.

   In our computation experiment, it is observed that if two vehicles enter the network at the same time and are assigned the same path, they pay the same toll. We have not yet proven this, so we propose this as a conjecture:

   Conjecture III.3. *If two vehicles arrive at the same entrance at the same time, and are assigned the same path, they will be charged the same amount.*

   However, a weaker theorem (III.4) can be proved. Intuitively, this theorem indicates that "rich" drivers will always pay no less than "poor" drivers :

   Theorem III.4. *If two vehicles k_1 and k_2 that*

   (a) share the same origin and destination,

   (b) arrive at the entrance of their trip at the same time

   (c) were assigned to the same path by the traffic controller

   (d) for vehicle $k = k_1, k_2$, $v_k^1 \leq v_k^2 \leq \cdots \leq v_k^P$ is always true

(e) *the value of paths by two vehicles satisfies* $v_{k_1}^{j_1} - v_{k_1}^{j_2} \leq v_{k_2}^{j_1} - v_{k_2}^{j_2}$ *for all* $j_1, j_2 \in \mathbf{P}$

then vehicle k_1 *would pay no more than* k_2

Proof. Consider two vehicles k_1 and k_2 which share the same arrival time $A_{k_1} = A_{k_2}$.

Suppose these two vehicles were assigned the same path j^*, then the payment of these two are:

$$\pi_{k_1} = \mathbf{U}^*_{-k_1} - (\mathbf{U}^* - v_{k_1}^{j^*})$$
$$\pi_{k_2} = \mathbf{U}^*_{-k_2} - (\mathbf{U}^* - v_{k_2}^{j^*})$$

To prove that $\pi_{k_1} - \pi_{k_2} \geq 0$, we will need to show that

$$\mathbf{U}^*_{-k_1} - \mathbf{U}^*_{-k_2} \geq v_{k_2}^{j^*} - v_{k_1}^{j^*}$$

Now if we let j_2 be the path assigned to vehicle k_2 in (MAX 1-k_1), that is, j_2 satisfies $x_{k_2}^{j_2} = 1$ in the optimal solution of (MAX 1-k_1). And similarly, let j_1 satisfy $x_{k_1}^{j_1} = 1$ in the optimal solution of (MAX 1-k_1). We claim that

$$j* \leq j_1 \leq j_2 \tag{3.16}$$

To prove this, use contradiction. If $j_1 > j_2$, then we can do one of the following

(a) assign path j_1 to vehicle k_2 in problem (MAX 1-k_1), that is, let $x_{k_2}^{j_1} = 1$ instead of $x_{k_2}^{j_2} = 1$

(b) assign path j_2 to vehicle k_1 in problem (MAX 1-k_2), that is, let $x_{k_1}^{j_2} = 1$ instead of $x_{k_1}^{j_1} = 1$

Define $\mathbf{U}_{-k_1,k_2}$ as the social utility excluding both vehicle k_1 and k_2. Also denote $x_{-k_2})^*_{-k_1}$ as the optimal solution to (MAX 1-k_2), removing vehicle k_1, while $x_{-k_1})^*_{-k_2}$ as the optimal solution to (MAX 1-k_1), removing vehicle k_2.

Note that payment problem (MAX 1-k_1) and (MAX 1-k_2) have identical feasible region (if we treat variables $x_{k_1}^j$ as $x_{k_2}^j$ and vice versa). The only difference between problem (MAX 1-k_1) and (MAX 1-k_2) is the objective coefficient $v_{k_1}^j$ as $v_{k_2}^j$.

For case 1, the change of objective value $\Delta\mathbf{U}_{-k_1}$ is

$$\begin{aligned}
\Delta\mathbf{U}_{-k_1} =& \left(\mathbf{U}_{-k_1,k_2}((x_{-k_2})^*_{-k_1}) + v_{k_2}^{j_1}\right) \\
&- \left(\mathbf{U}_{-k_1,k_2}((x_{-k_1})^*_{-k_2}) + v_{k_2}^{j_2}\right) \\
=&\mathbf{U}_{-k_1,k_2}((x_{-k_2})^*_{-k_1}) - \mathbf{U}_{-k_1,k_2}((x_{-k_1})^*_{-k_2}) \\
&+ v_{k_2}^{j_1} - v_{k_2}^{j_2}
\end{aligned}$$

For case 2, the change of objective value $\Delta\mathbf{U}_{-k_2}$ is

$$\begin{aligned}
\Delta\mathbf{U}_{-k_2} =& \left(\mathbf{U}_{-k_1,k_2}((x_{-k_1})^*_{-k_2}) + v_{k_1}^{j_2}\right) \\
&- \left(\mathbf{U}_{-k_1,k_2}((x_{-k_2})^*_{-k_1}) + v_{k_1}^{j_1}\right) \\
=&\mathbf{U}_{-k_1,k_2}((x_{-k_1})^*_{-k_2}) - \mathbf{U}_{-k_1,k_2}((x_{-k_2})^*_{-k_1}) \\
&+ v_{k_1}^{j_2} - v_{k_1}^{j_1}
\end{aligned}$$

Thus, according to assumption 2e

$$
\begin{aligned}
&\Delta \mathbf{U}_{-k_1} + \Delta \mathbf{U}_{-k_2} \\
=& v_{k_2}^{j_1} - v_{k_2}^{j_2} + v_{k_1}^{j_2} - v_{k_1}^{j_1} \\
>& 0
\end{aligned}
$$

This means at least one of $\Delta \mathbf{U}_{-k_1}$ or $\Delta \mathbf{U}_{-k_2}$ must be positive. So it is always possible to improve either $\mathbf{U}^*_{-k_1}$ or $\mathbf{U}^*_{-k_2}$, which contradict with the assumption that they are optimal. □

If the conjecture is true, it can be used to reduce running time of our mechanism: instead of solving payment problem for each vehicle, we solve payment problem (MAX 1-k) only once for each time step and each path.

3.4 Implementation Issues

3.4.1 Alternative Free Paths

We have assumed that every vehicle participating in the auction will be assigned to exactly one non-congested path. However, in most applications, this might be a too restricted, which can make problem (MAX 1) infeasible.

One extension of this auction model is to allocate at least one alternative free path between any pair of origin and destination. This alternative path, unlike other paths in the model, is toll-free, but is subject to congestion. We set $\mathbf{P}_f$ as the set of "free" paths. Each vehicle i is also required to submit a bid $\hat{v}_i^p$ for path $p \in \mathbf{P}_f$. We assume that

$$\hat{v}_i^p \leq \hat{v}_i^j \quad \forall p \in \mathbf{P}_f, j \notin \mathbf{P}_f$$

The rest of the auction mechanism remains the same, except that if a vehicle is assigned to a free path in problem (MAX 1), no payment problem (MAX 1-k) is solved, and the vehicle pays no toll.

3.4.2 Auction as a Tolling Sub-system for HOV or HOT Lanes

An issue that arises while implementing V2I devices, or generally, Intelligent Transportation System (ITS), is that early users of the systems gain little or no benefit when the market penetration of that device is low. Although the auction mechanism we present here is implemented as a stand-alone system, where all vehicles are required to be equipped with V2I devices, it can also be used as a tolling sub-system for High Occupancy Vehicle (HOV) or High Occupancy Toll (HOT) lanes.

In this case, the set of links **L** is defined as the set of HOV or HOT lanes. Regular lanes are treated as alternative links as in section 3.4.1. Only vehicles equipped with bidding devices are allowed to enter the HOT lanes, while vehicles without V2I devices can use regular lanes in the network. Thus, all vehicles can use the roads regardless of whether they are equipped with V2I bidding devices. At the same time, the system creates an incentive for vehicles to participate in the auction since then it allows non-congested travel.

3.4.3 Rolling Horizon

The current auction is operated off-line, meaning that all vehicles bid and get the assigned paths before starting travel. This limits the usability of the model. However, one can extend this model to a rolling horizon reservation system. In this system, we set up a main auction labelled B_0 which has a "cut-off" time, say, two hours before the start of planning period. Every vehicle that bids before this cut-off time will receive the path assignment and payment information immediately at the cut-off time. Vehicles who miss the cut-off time can still bid upon arrival at the entrance by

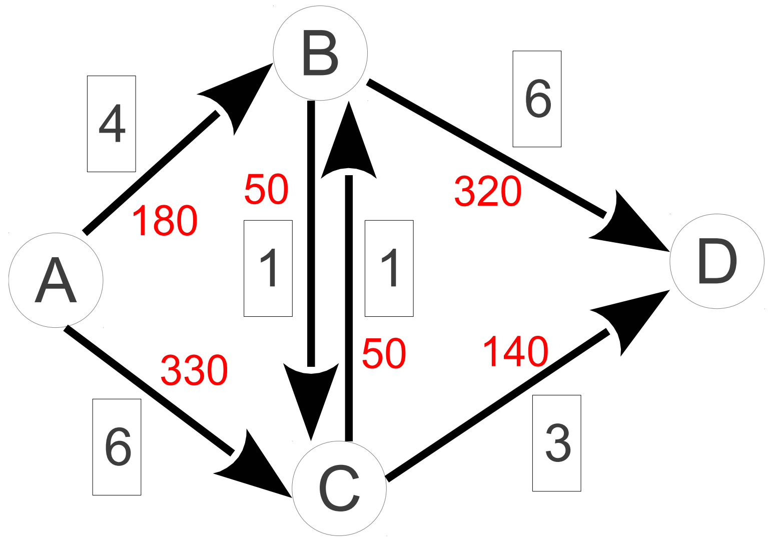


Figure 3.3: Network

participating in the following "rolling" auction:

The traffic controller will start a new round of auction B_t at every time period $t \in \mathbf{T}$. Vehicles arriving between time $t-1$ and t who did not bid before the cut-off time can participate in auction B_t. In auction B_t, we solve problems (MAX 1) and (MAX 1-k) by replacing the right-hand-side of constraint (3.9) by $C_{l,t}$, the "remaining capacity" of link l at time t. $C_{l,t}$ is calculated by subtracting the number of vehicles using link l at time t from the free-flow capacity C_l, using the prior vehicles' assignments from $B_0, B_1, \ldots, B_{t-1}$.

Since $C_{l,t}$ is always less than C_l, vehicles bid in auction B_t are likely to pay higher toll than those who bid in B_0.

3.5 Numerical Experiment

3.5.1 Experiment One: One Origin-Destination Pair

We test this model on the traffic network shown in Figure 3.3.

There are six links in this network. The free flow travel time T_l of each link l is shown in a box next to it. The free flow capacity C_l is also shown as a red number attached to link l.

We set up the number T_l such that four paths have different free flow travel time: 8 for Path ABCD, 9 for ACD, 10 for ABD, and 13 for ACBD. This makes the interaction of path choice and toll transparent.

To better understand the dynamic of traffic assignment and toll price, we assume that all vehicles are travelling only from A to D.

We assume that the number of vehicles arriving at the entrance follows a Poisson distribution with rate λ. Note that λ can be a function of time.

We assume vehicles' value v_i^j for travelling in path j is a linear function of the free-flow travel time of path j, i.e., $v_i^j = c_i T^j$. Here c_i can be viewed as the vehicle i's "willingness-to-pay" per unit travel time. We generate c_i with a log-normal distribution with mean $\mu = 1$ and standard deviation $\sigma = 0.5$.

To simulate the situation of real traffic, we generate an incoming vehicle flow in the following manner: from time 0 to 20, the number of vehicles arriving gradually increases from 60 to 100 vehicles per minute. Then the rate of arrival stays at 100 vehicles per minute from time 20 to 40 before it gradually decreasing to 60 vehicles per minute at time 60.

The parameters of this test are shown in Table 3.2

Table 3.2: Parameters of Simulation

Parameter	Meaning
$N = 5100$	Total Number of Vehicles
$\delta = 1$	Minutes per Time Period
$T = 60$	Number of Time Periods
λ	Vehicle arrival rate
$\mu = 1$	Mean of willingness-to-pay
$\sigma = 0.5$	Standard deviation of willingness-to-pay

3.5.1.1 Results of Experiment One

We generate input data according to the settings described above, and solve the path assignment (MAX 1) and payment problems (MAX 1-k) using CPLEX 12.0. A 30-CPU computer cluster was used to solve 5100 payment problems in parallel. It took about 5 minutes to solve the path assignment and all payment problems.

We now analyze the traffic flow on each path over time on Figure 3.4.

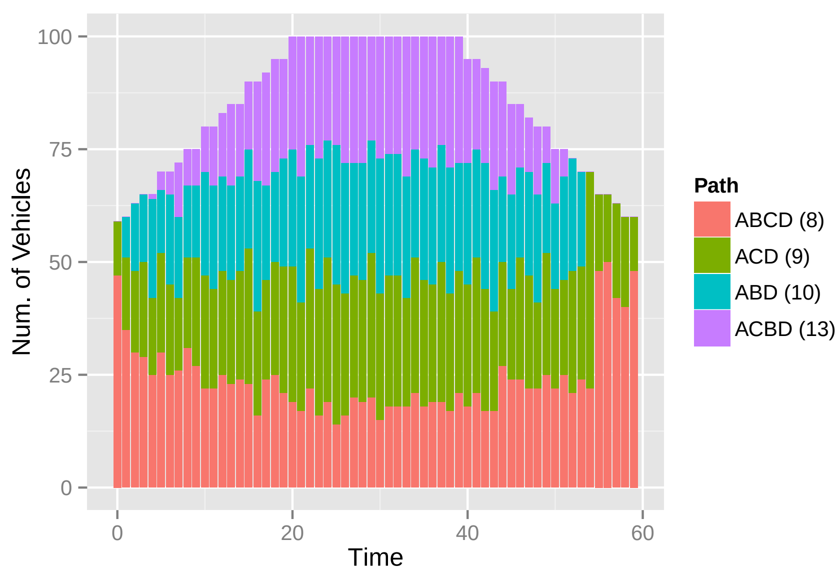


Figure 3.4: Number of Vehicles Using Each Path

As is shown in Figure 3.4, at the beginning when traffic is low, all of the traffic goes through the shortest two paths: ABCD and ACD. As traffic flow increases over time, more and more vehicles are assigned to longer paths.

At the same time, Figure 3.5 shows that the toll price also goes up as the traffic flow increases over time. Also the toll is higher for shorter path, and lower for longer path.

We also analyze the traffic flow of each link during the 60 minutes test period. As is shown in Figure 3.6, while flow in link AC and BD only reach link capacity during the peak time, flow in link AB and CD are very close or at the capacity most of the

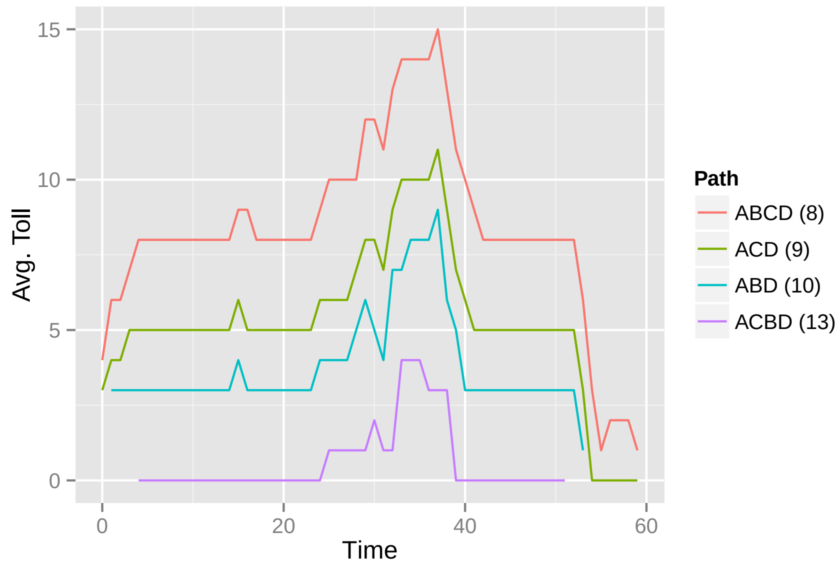


Figure 3.5: Average Toll Price

time during the test.

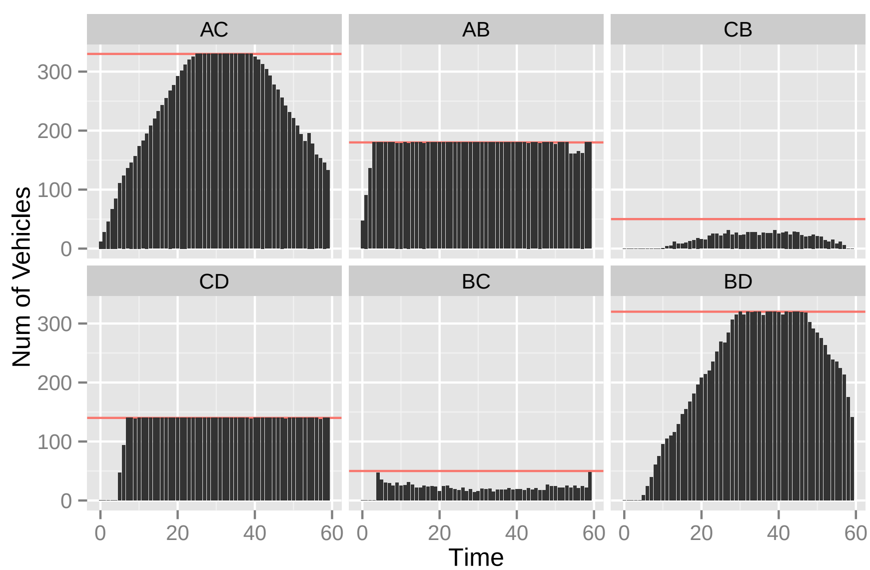


Figure 3.6: Number of Vehicles Using Each Link

To see if the mechanism distributes the toll “fairly”, we compare the relationship between payment and bid of each vehicle in Figure 3.7, which consist of four sub-figures, each representing one path. For each path j, we plot all vehicles assigned to this path in the following way: for vehicle k, arrival time A_k is plotted as x coordinate, whereas payment per unit travel time (π_k/T_j) is plotted as y coordinate, and the color of a dot represents the value per unit travel time (c_k), with red being lowest value, and purple being the highest.

Since many vehicles share the same arrival time and payment, to clearly distinguish each vehicle, we add a small random perturbation to each vehicle's x and y coordinates. As is already shown in Figure 3.5, vehicles assigned to shorter paths such as ABCD would pay more than vehicles assigned to longer paths such as ACBD. More importantly, this figure shows that vehicles assigned to shorter paths are driven by mostly "richer" people, i.e., people who has higher value of time c_i: most of the dots in the first sub-figure representing the shortest path ABCD are green and blue, which means these vehicles has value per unit travel time (c_i) greater than 4.

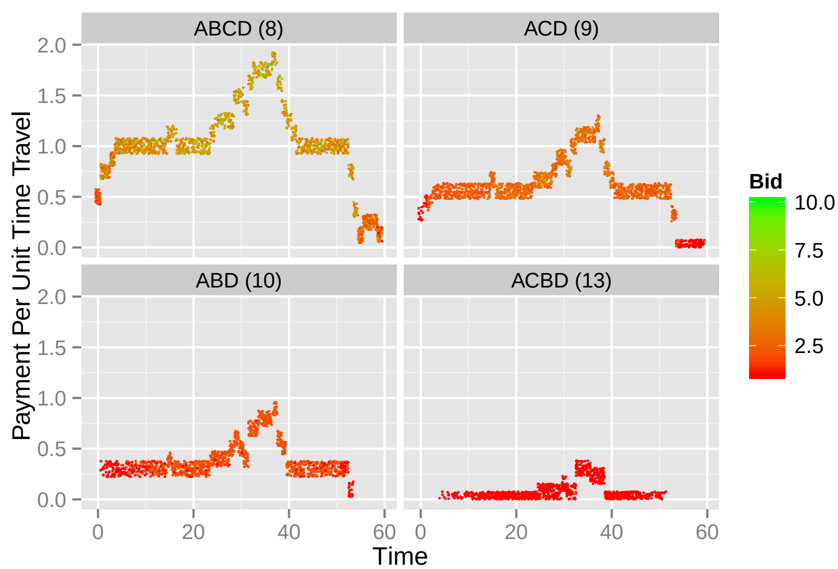


Figure 3.7: Payment and Bid Relationship Over Time

We also group all vehicle who has the same c_i together, and plot the distribution of payments (tolls) for each group using 'box-plot' in Figure 3.8. The meaning of this box-plot is as follows: the line within a box represents the median of payment within a corresponding c_i group. The upper and lower end of each box represent the first and third quartiles (25th and 75th percentiles) of payment in the group. The red line above the box extends from the first quartile to the highest value that is within 1.5 times inter-quartile range, or distance between the first and third quartiles. Similarly, the red line below the box ranges from the third quartile to the lowest value within 1.5 times inter-quartile. Payments not in these ranges (outliers) will be plotted as black dots.

From Figure 3.8, we can see that all tolls per unit travel time are less than their corresponding bid per unit travel time.

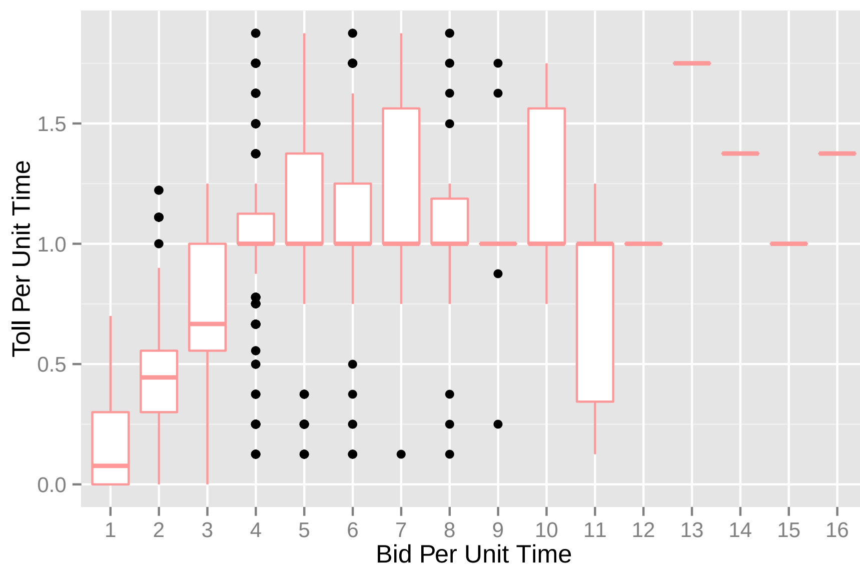


Figure 3.8: Payment and Bid Relationship

3.5.2 Experiment Two

In the second experiment, we simulate our auction mechanism in a highway network similar to the one near Los Angeles, CA. This area contains freeway I-5, I-405, CA-22 , CA-91 and CA-55. In this test, we use 42 nodes and 98 links. The structure of this network is shown in Figure 3.9.

Vehicles are generated to travel from 80 of the origin-destination (O-D) pairs, each pair with certain probability shown in table 3.3. We also limit three available paths

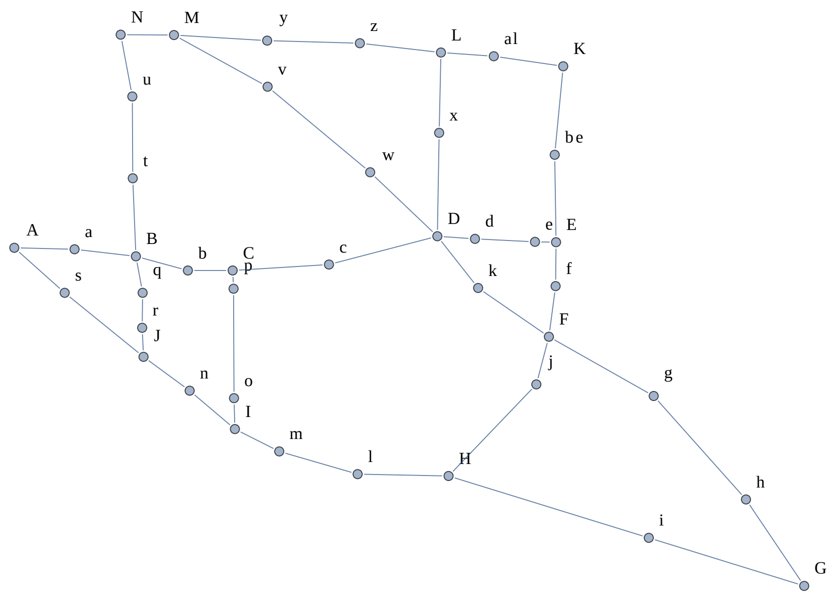


Figure 3.9: Traffic Network Near Los Angeles

between each O-D pair.

Figure 3.10 shows the histogram of vehicles willingness-to-pay per unit time c_i. As the same in Experiment One, this parameter is also a log-normal distributed with mean 1 and standard deviation 0.5.

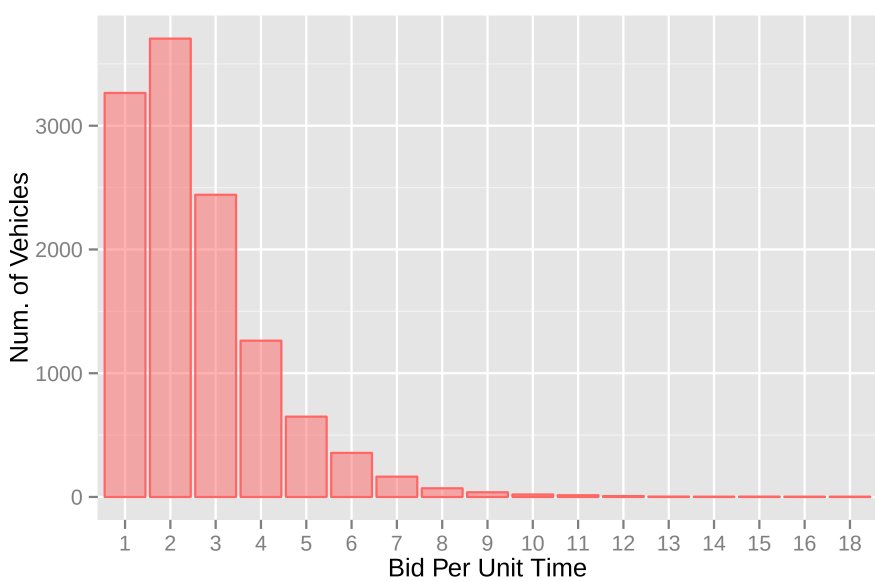


Figure 3.10: Histogram of c_i (Willingness-to-pay Per Unit Time)

Table 3.3: Origin and Destination Pairs of The Experiment

Origin	Destination	Percentage	Origin	Destination	Percentage
a	f	0.543	y	n	0.543
f	a	0.543	n	z	2.174
a	j	0.543	z	n	0.543
j	a	0.543	n	α	1.630
a	h	0.543	α	n	0.543
h	a	0.543	n	β	1.630
a	g	1.087	β	n	0.543
g	a	1.087	n	k	1.630
a	r	1.087	k	n	0.543
r	a	1.087	n	f	2.174
a	u	1.087	f	n	1.087
u	a	1.087	f	c	1.087
a	y	0.543	c	f	1.087
y	a	0.543	f	u	1.630
a	z	0.543	u	f	0.543
z	a	0.543	f	t	1.630
a	x	1.087	t	f	1.087
a	α	1.087	f	y	2.174
α	a	1.087	y	f	0.543
a	β	1.630	f	z	1.630
β	a	1.630	z	f	0.543
s	f	0.543	j	y	2.717
f	s	0.543	j	z	2.717
s	c	0.543	j	i	1.630
c	s	0.543	j	u	2.174
s	α	1.087	j	t	1.630
α	s	1.087	d	y	1.630
s	β	1.087	d	z	2.174
β	s	1.087	d	i	2.174
s	g	1.630	d	m	2.174
g	s	0.543	d	l	2.174
s	i	1.630	e	u	1.630
i	s	1.087	e	t	2.174
s	x	1.630	e	i	2.717
x	s	0.543	e	m	1.630
s	v	1.087	e	l	1.630
v	s	0.543	b	n	1.087
s	w	1.087	b	h	3.261
w	s	0.543	b	g	1.630
n	y	1.630	b	x	1.087

Other parameters are similar as in experiment one, except that this time we test 12000 vehicles in a period of 120 minutes. List of these parameters are shown in table 3.4.

Table 3.4: Parameters of Simulation

Parameter	Meaning
$N = 12000$	Total Number of Vehicles
$\delta = 1$	Minutes per Time Period
$T = 120$	Number of Time Periods
λ	Vehicle arrival rate
$\mu = 1$	Mean of willingness-to-pay
$\sigma = 0.5$	Standard deviation of willingness-to-pay

3.5.2.1 Results of Experiment Two

We test the auction mechanism in the aforementioned settings. It takes 22 minutes to solve the path assignment and all of the 12000 payment problems on a 30-CPU cluster running CPLEX 12.0. Figure 3.11 shows the percentage of capacity used for each link at different time during the test: flow in links that are more than 90% of the capacity is colored in red, while flow between 80% and 90% of capacity is colored in yellow, and the rest of links are colored in green.

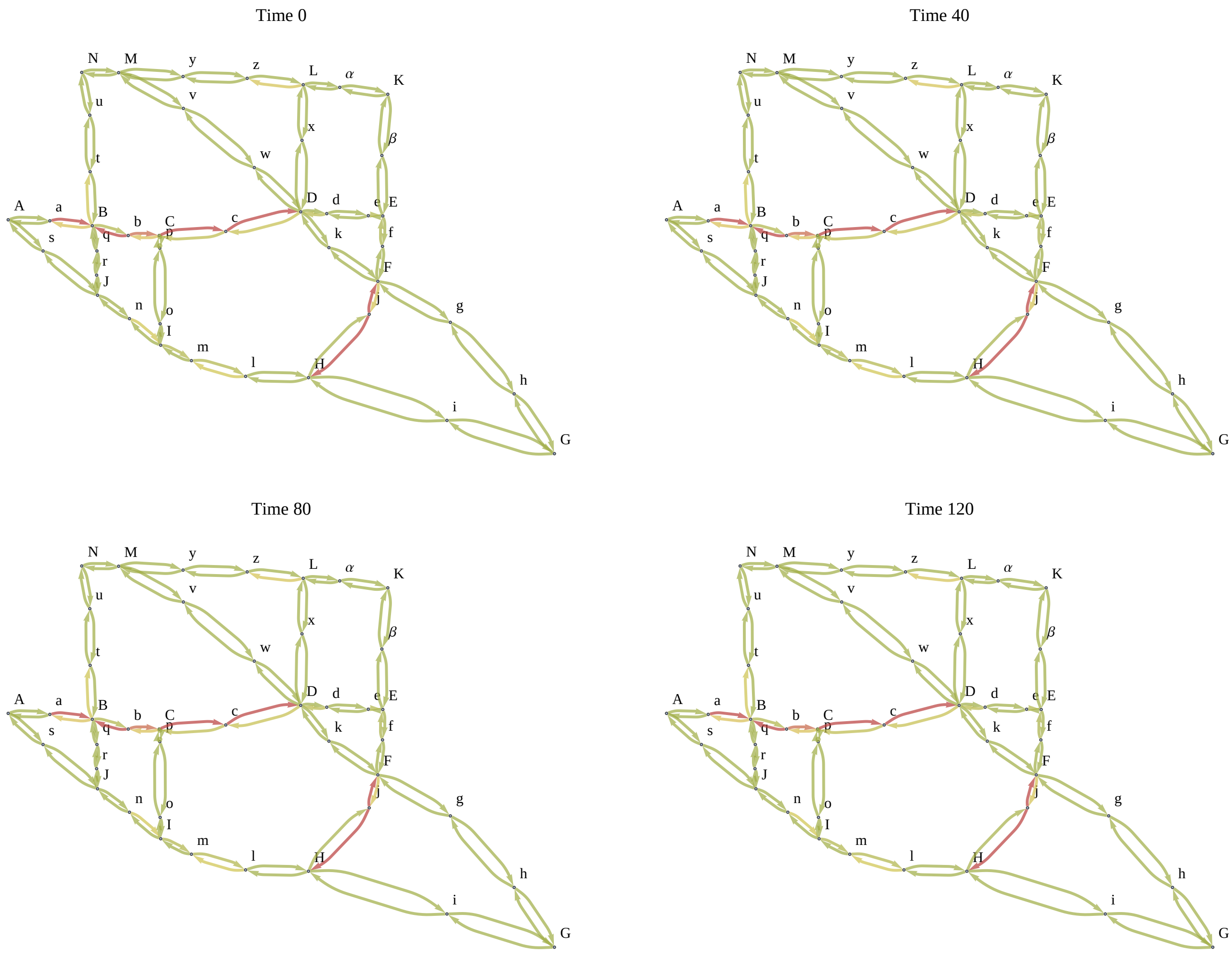
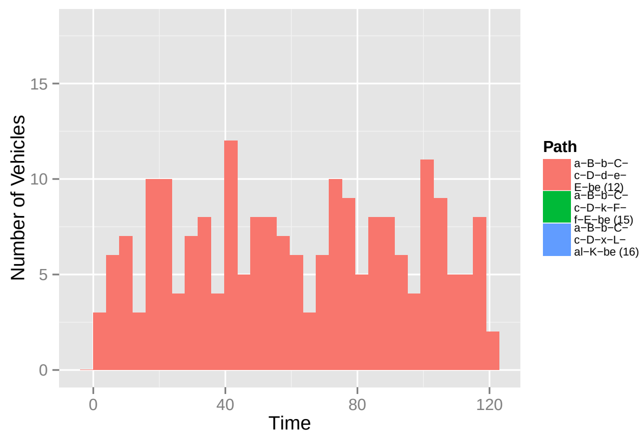
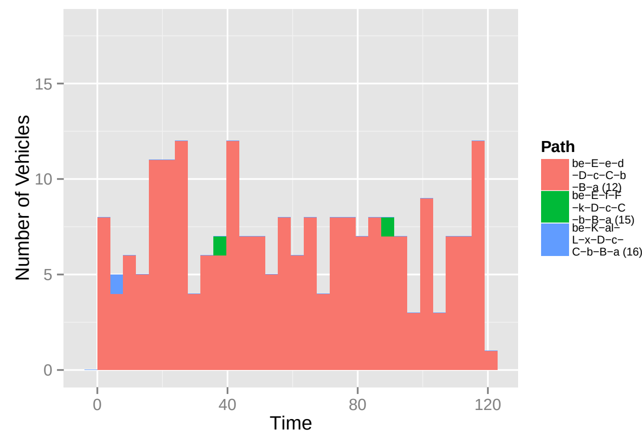


Figure 3.11: Vehicle Flow in Network

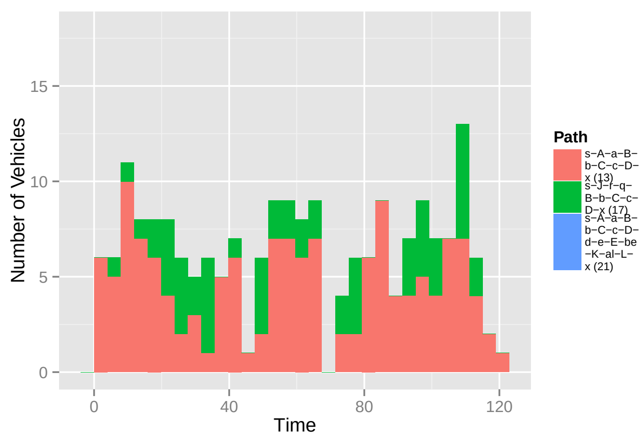
Figure 3.12 shows the traffic flow between certain orign and destinations



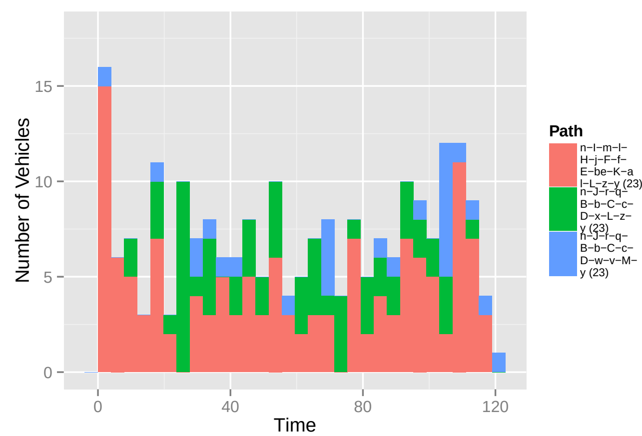
(a) $a \to \beta$



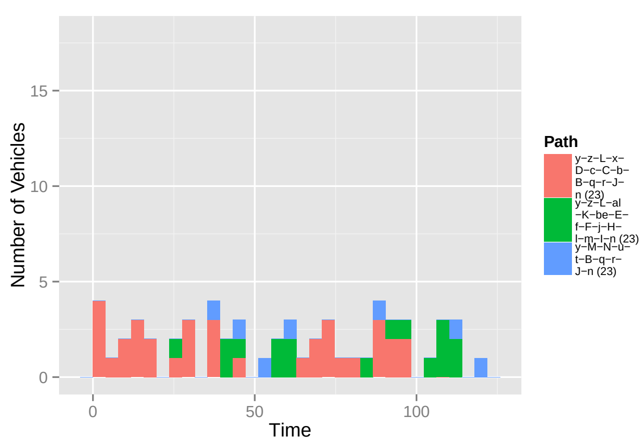
(b) $\beta \to a$



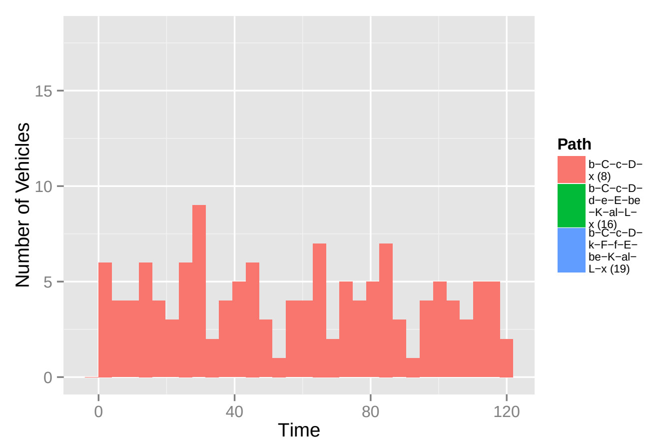
(c) $s \to x$



(d) $n \to y$



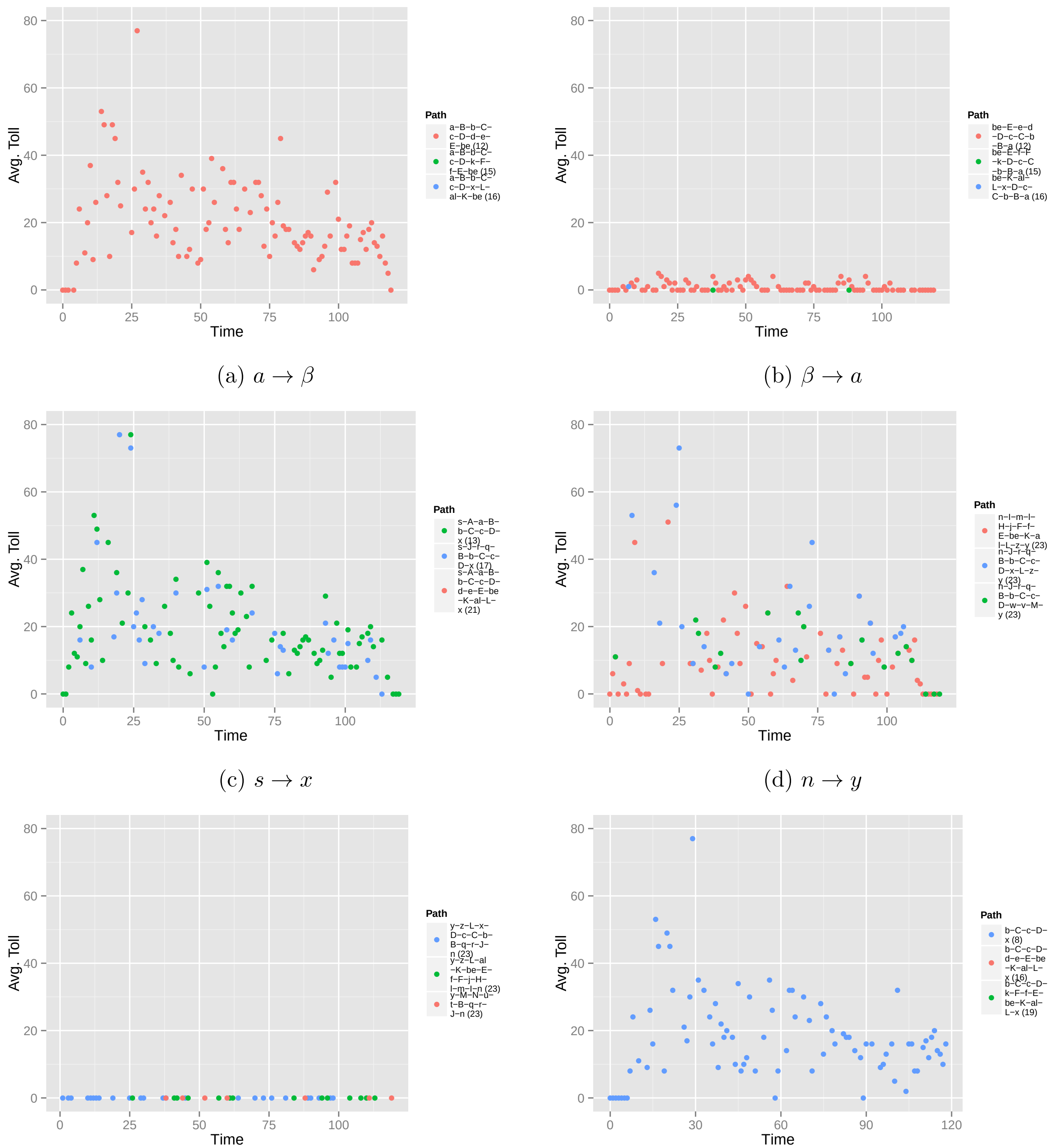
(e) $y \to n$



(f) $b \to x$

Figure 3.12: Flow of Vehicles of Various Origins and Destinations

Figure 3.13 shows the price that each vehicle paid for their assigned paths. Because of the limit of space, we only show these prices between six pairs of origin and destination: $a \to \beta$, $\beta \to a$, $\beta \to a$, $n \to y$, $y \to n$, $b \to x$.



(a) $a \to \beta$ (b) $\beta \to a$

(c) $s \to x$ (d) $n \to y$

(e) $y \to n$ (f) $b \to x$

Figure 3.13: Payment of Vehicles Travelling Between Six O-D Pairs

Notice that vehicles in certain paths such as in Figure 3.13a, 3.13c 3.13d and 3.13f are paying very high price, whereas vehicles in other paths such as in Figure 3.13b and 3.13e almost pay nothing. We will analyze this abnormal pricing in the next section.

3.5.3 Voluntary Participation Issues

As is seen in the result of experiment two, some vehicles are paying very high price for the path assigned. In fact, some of them are even paying higher than what they bid for the paths. In reality, vehicles having to pay more than they bid would not even participate in the auction.

This abnormal payment happens when the model is used in a multiple Origin-Destination (OD) pairs settings.

The reason for abnormal payment is illustrated in the following example shown in figure 3.14 and table 3.5.

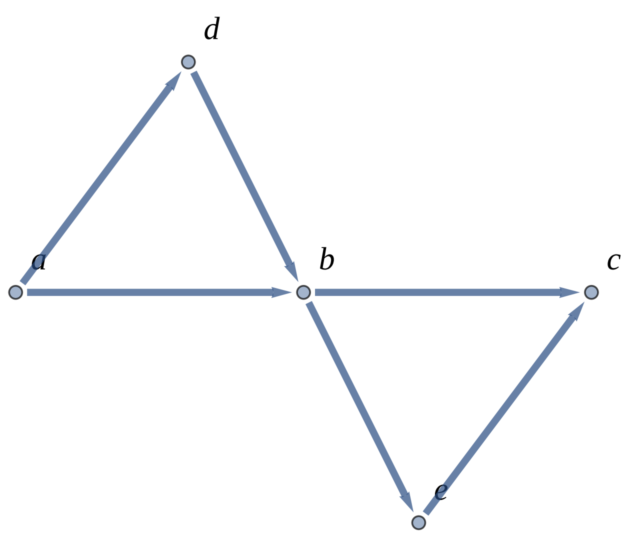


Figure 3.14: Abnormal Payment Example

Table 3.5: Paths and Bids of Abnormal Payment Example

(a) Paths Specification for Vehicles

Vehicle	Path 1	Path 2
A	abec	abc
B	ab	adb
C	bec	bc

(b) Vehicles Bids for Each Path

Vehicle	Path 1	Path 2
A	10	3
B	8	2
C	7	1

Every vehicle has two available paths to travel. The paths specification of each vehicle and their individual bids are shown in table 3.5.

Suppose each link only has one unit of capacity. Also assume that vehicle A and B arrives at node a at the same time, and vehicle A and C arrives at node b at the same time. So the only two feasible assignments are:

$$A \to 1, \quad B \to 2, \quad C \to 2$$

or

$$A \to 2, \quad B \to 2, \quad C \to 1$$

It is easy to see that the optimal assignment is to assign vehicle A, B and C to their path 1, 2 and 2 respectively. So the optimal social utility is

$$\mathbf{U}^* = 10 + 2 + 1 = 13$$

To calculate the payment of vehicle A, we "remove" it from the system, and calculate the social utility $\mathbf{U}^*_{-A}$:

$$\mathbf{U}^*_{-A} = 8 + 7$$

Since now without vehicle A, both vehicle B and C can be assigned to their respective "preferred" paths.

We then compute the payment of vehicle A as

$$
\begin{aligned}
\pi_A &= \mathbf{U}^*_{-A} - \mathbf{U}_{-A}(\mathbf{x}^*_{-A}) \\
&= \mathbf{U}^*_{-A} - (\mathbf{U}^* - v_A^1) \\
&= 15 - (13 - 10) \\
&= 12 > 10 = v_A^1
\end{aligned}
$$

As is seen in this example, the reason behind this phenomenon is that when some vehicle is assigned to a "critical path" in the network, it essentially becomes a bottleneck, forcing many other vehicles to be assigned to less desirable paths. And the payment π_k for vehicle k,

$$
\begin{aligned}
\pi_k &= \mathbf{U}^*_{-k} - \mathbf{U}_{-k}(\mathbf{x}^*_{-k}) \\
&= \mathbf{U}^*_{-k} - (\mathbf{U}^* - v_k^{j^*}) \\
&= v_k^{j^*} - (\mathbf{U}^* - \mathbf{U}^*_{-k})
\end{aligned}
$$

where j^* is the optimal path assigned to vehicle k. Usually the term $\mathbf{U}^* - \mathbf{U}^*_{-k}$ is positive, making the payment less than what k bid for the path, but when there is significant bottleneck caused by k, as is demonstrated by the previous example, $\mathbf{U}^* - \mathbf{U}^*_{-k}$ becomes negative, leading to payment higher than the bid.

3.5.3.1 Using Alternative Path

The reason for the bottleneck to form is that we assume each vehicle is always guaranteed one path. Sometimes this assigned path turns out to be a bottleneck for many other vehicles.

One method to fix this problem is to assume that there is an alternative path between every origin and destination pairs. This alternative path is free of charge, but could be congested. Call this free path $f(i)$ for vehicle i. Since $f(i)$ might be congested, we further assume that the bid for this path $v_i^{f(i)} = 0 \quad \forall i \in \mathbf{N}$.

Now the (MAX 1) problem becomes

$$\mathbf{U}^* = \text{maximize} \sum_{i \in \mathbf{N}} (\sum_{j \in \mathbf{P}} \quad \hat{v}_i^j x_i^j + \hat{v}_i^{f(i)} x_i^{f(i)}) \tag{MAX 1-F}$$

$$\text{s.t.} \sum_{j \in \mathbf{P}} x_i^j + x_i^{f(i)} = 1 \qquad \forall i \in \mathbf{N} \tag{3.17}$$

$$\sum_{j \in \mathbf{P}} \sum_{\substack{i: A_i > t - \tau_l^j - T_l \\ A_i \leq t - \tau_l^j}} x_i^j \leq C_l \qquad \forall t \in \mathbf{T},\, l \in \mathbf{L} \tag{3.18}$$

$$x_i^j = 0, 1 \qquad \forall i \in \mathbf{N},\, j \in \mathbf{P} \tag{3.19}$$

$$x_i^{f(i)} = 0, 1 \qquad \forall i \in \mathbf{N} \tag{3.20}$$

Because of the assumption $v_i^{f(i)} = 0$, this problem is equivalent to

$$\mathbf{U}^* = \text{maximize} \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{P}} \quad \hat{v}_i^j x_i^j \tag{MAX 1-F}$$

$$\text{s.t.} \sum_{j \in \mathbf{P}} x_i^j \leq 1 \qquad \forall i \in \mathbf{N} \tag{3.21}$$

$$\sum_{j \in \mathbf{P}} \sum_{\substack{i: A_i > t - \tau_l^j - T_l \\ A_i \leq t - \tau_l^j}} x_i^j \leq C_l \qquad \forall t \in \mathbf{T},\, l \in \mathbf{L} \tag{3.22}$$

$$x_i^j = 0, 1 \qquad \forall i \in \mathbf{N},\, j \in \mathbf{P} \tag{3.23}$$

Now we prove the following theorem to show that for any vehicle $i \in \mathbf{N}$, $\pi_i \leq v_i^{j^*}$, where j^* is the optimal path assigned to i.

Theorem III.5. *If there exists an uncapcitated free path between any pair of origin and destination, and every vehicle bids zero on this free path, then payment of every vehicle is no greater than its bid for the assigned path.*

Proof. By contradiction. Suppose vehicle i's payment $\pi_k > v_k^{j^*}$. Then

$$\begin{aligned} \pi_k = v_k^{j^*} - (\mathbf{U}^* - \mathbf{U}^*_{-k}) > v_k^{j^*} \\ \Rightarrow \mathbf{U}^*_{-k} > \mathbf{U}^* \end{aligned}$$

Denote $\mathbf{x}^*_{-k}$ as the optimal solution to $\mathbf{U}^*_{-k}$. Also define $\mathbf{x}'*$ as

$$\mathbf{x}' = \left(\mathbf{x}_{-k} = \mathbf{x}^*_{-k}, x_k^{f(i)} = 1, x_k^j = 0\right), \quad \forall j \in \mathbf{P}$$

i.e., $\mathbf{x}'$ assigns vehicle k to the free alternative path, and other vehicles according to the solution of $\mathbf{x}^*_{-k}$. Thus

$$\mathbf{U}(\mathbf{x}') = \mathbf{U}^*_{-k} + v_k^{f(k)} = \mathbf{U}^*_{-k} > \mathbf{U}^*$$

This violates the optimality of $\mathbf{U}^*$.

□

In real world, this combination of auctioned and free path system can be used in tolling the HOV/HOT lanes and regular lanes. Further details of this issue is discussed in section 3.4.2.

3.5.3.2 Solution With Alternative Path

We test the auction mechanism with alternative free paths. The result shows that 215 out of 12000 vehicles are assigned to the alternative free path, and the objective value (social utility) improve by 2323. Table 3.6 shows the difference of toll collection in these two scenarios.

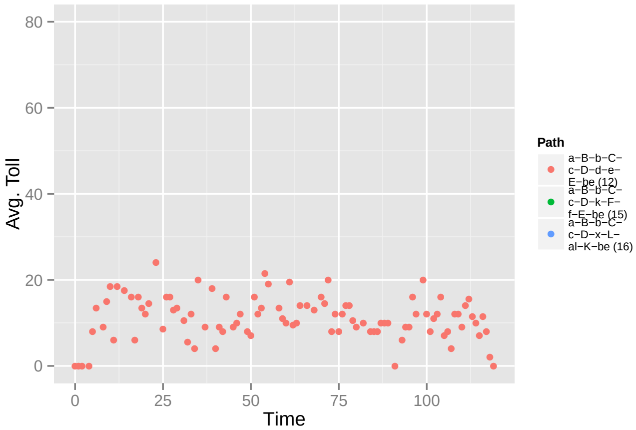
Table 3.6: Toll

	Total Toll	Avg. Toll
No Free Path	90621	7.55
With Free Path	43691	3.64

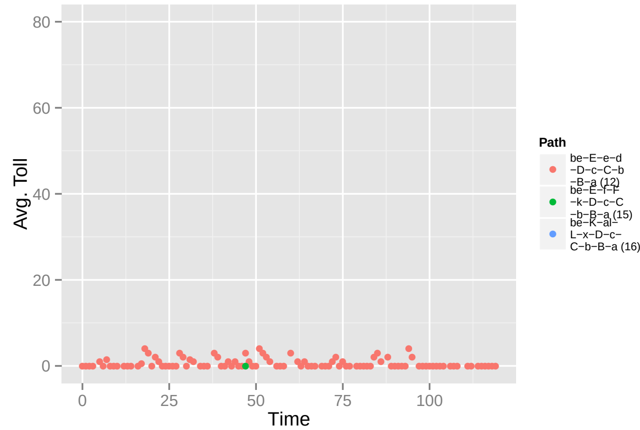
Although the auction with free path collect less toll (43691) than one without (90621), the actual collection will be closer to the smaller figure. This is because vehicles charged a toll higher than they bid may not show up.

Figure 3.15 shows the average toll paid by vehicles traveling between the same six origin-destination pairs as in Figure 3.13.

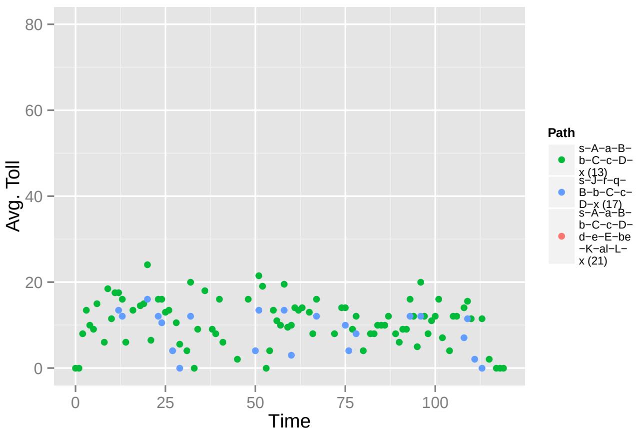
Figure 3.16 compares the payment and bid relation before and after adding free paths for the previous six origin-destination pairs. In each pair, we plot the payment and bid relation with no free path in the upper half, and and result of adding free path in the bottom half. Each panel in the figure represents a path. Each dot in the panel represents a vehicle's information: the x-axis is the time it arrives, y-axis is the price it pays, and the color of the dot is the per unit time bid for this vehicle (c_i). To clearly distinguish each vehicle, we add a small random perturbation to each vehicle's x and y coordinates.



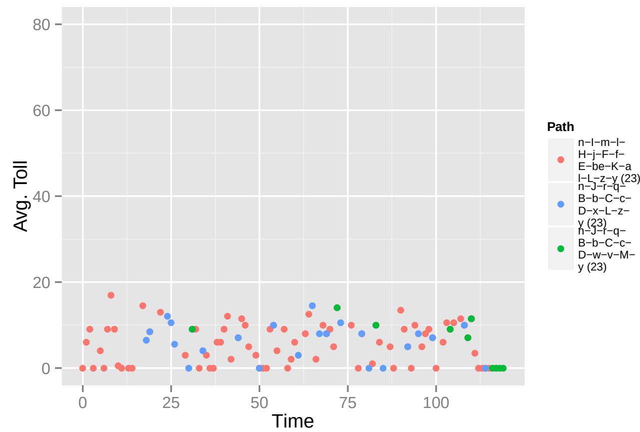
(a) $a \rightarrow \beta$



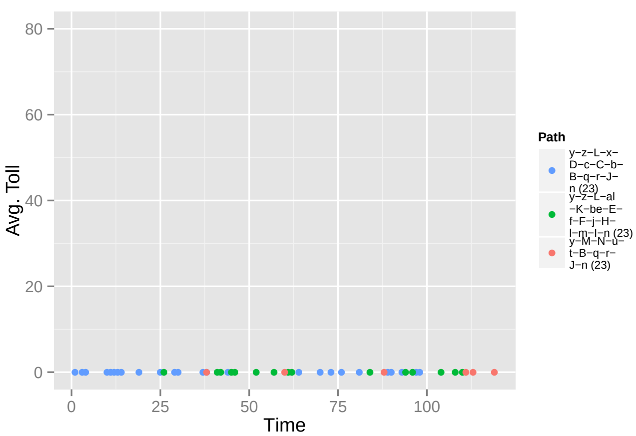
(b) $\beta \rightarrow a$



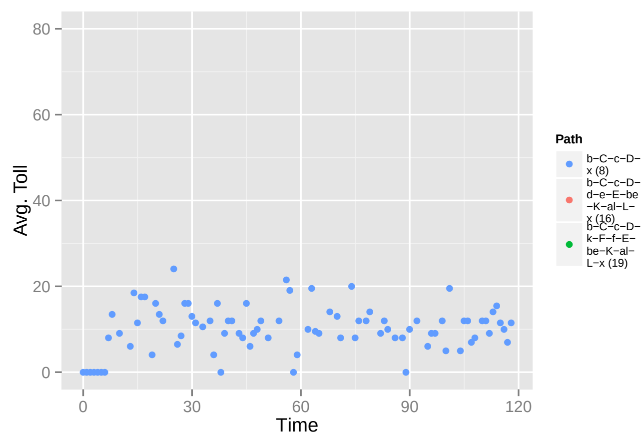
(c) $s \rightarrow x$



(d) $n \rightarrow y$

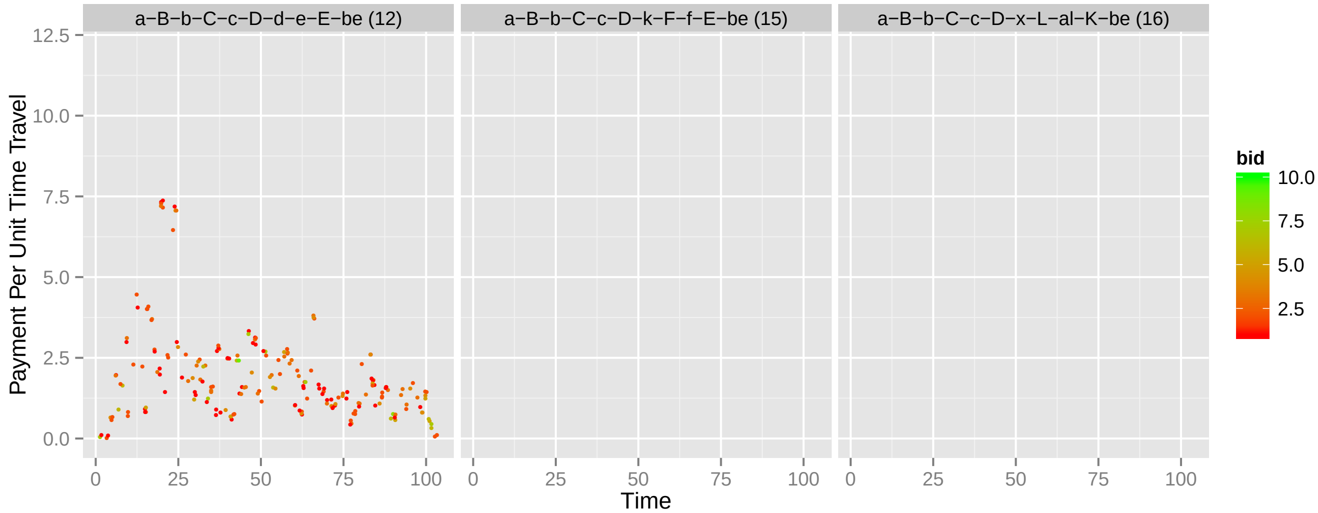


(e) $y \rightarrow n$

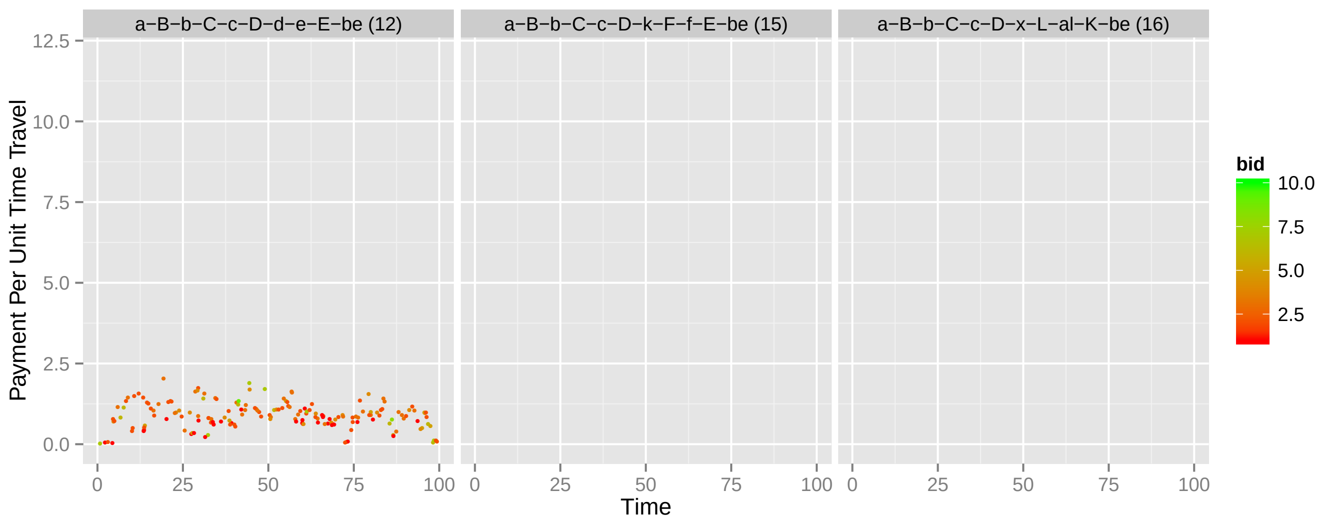


(f) $b \rightarrow x$

Figure 3.15: Payment of Vehicles Travelling Between Six O-D Pairs With Paths



(a) Without Free Path

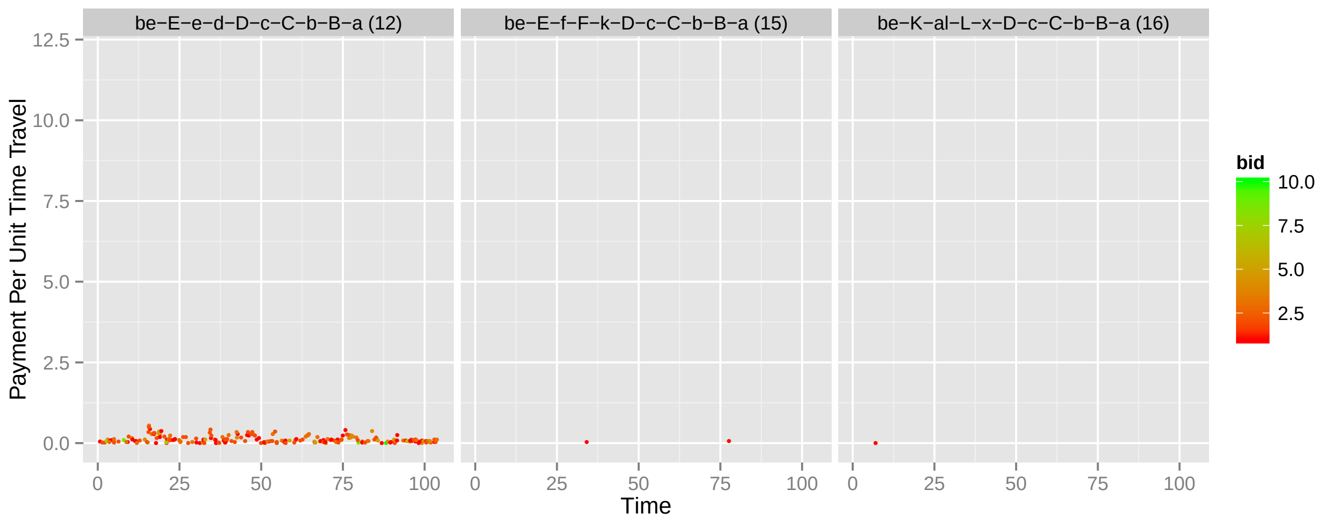


(b) With Free Path

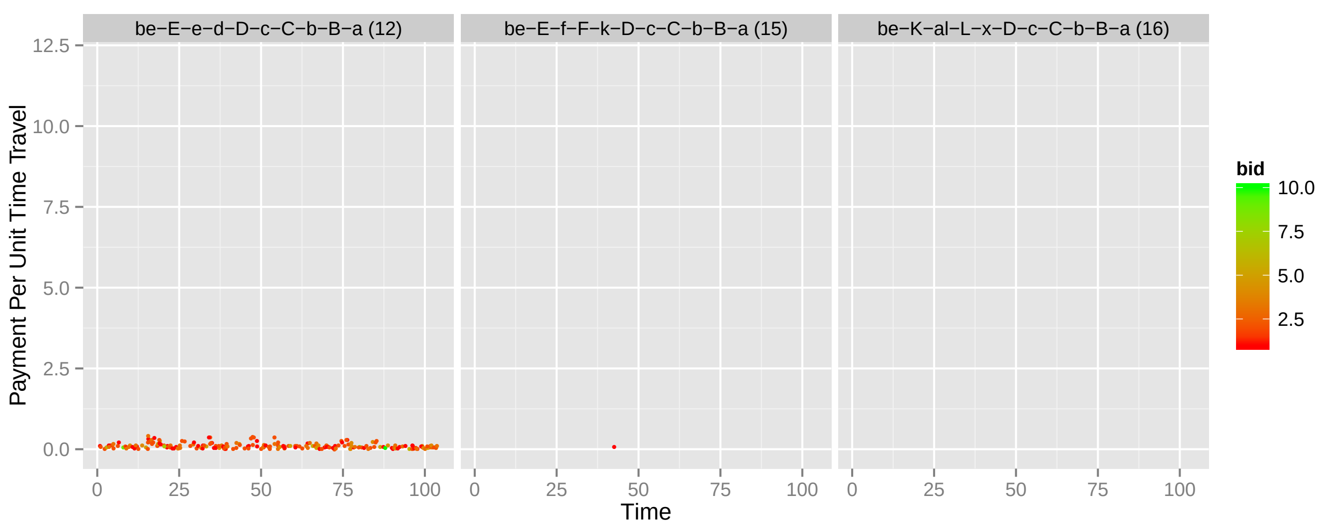
Figure 3.16: Payment and Bid Relation of O-D $a \to \beta$ With and Without Free Path

After adding free paths, all of the tolls are less than bids, as can be seen clearly by comparing Figure 3.16a and 3.16b.

Figure 3.22 shows how many vehicles are assigned to the free paths between these six O-D pairs. Among these vehicles in free paths, most of them are vehicles traveling from node a to β (colored in pink), some are traveling from s to x, and n to y. Tolls of these paths are reduced significantly, as can be seen in Figure 3.16, 3.18 and 3.19.

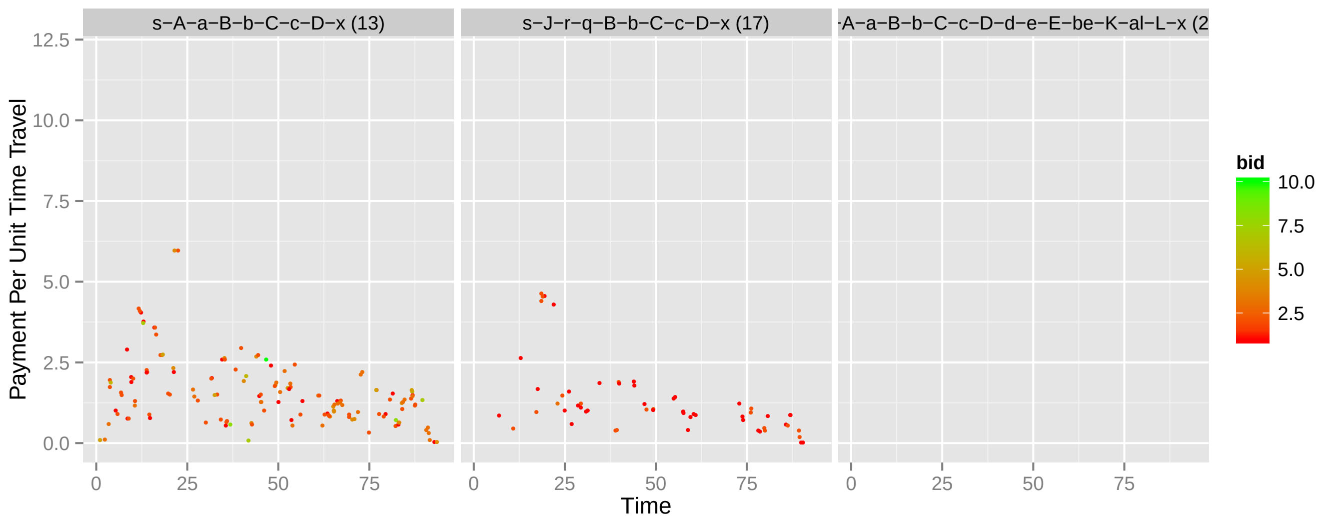


(a) Without Free Path

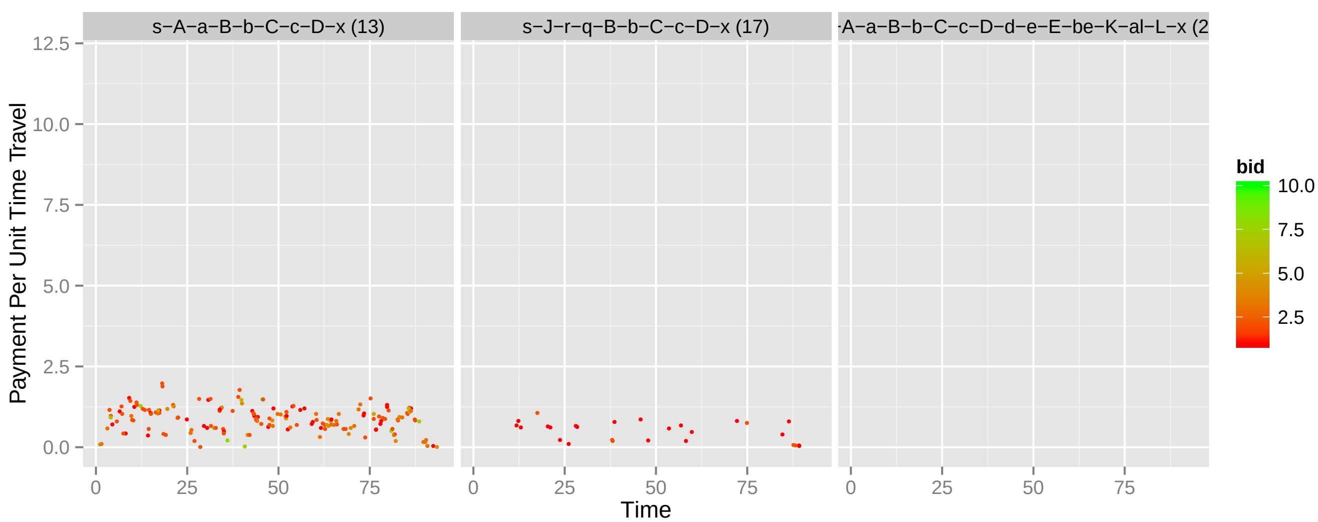


(b) With Free Path

Figure 3.17: Payment and Bid Relation of $\beta \rightarrow a$ With and Without Free Path

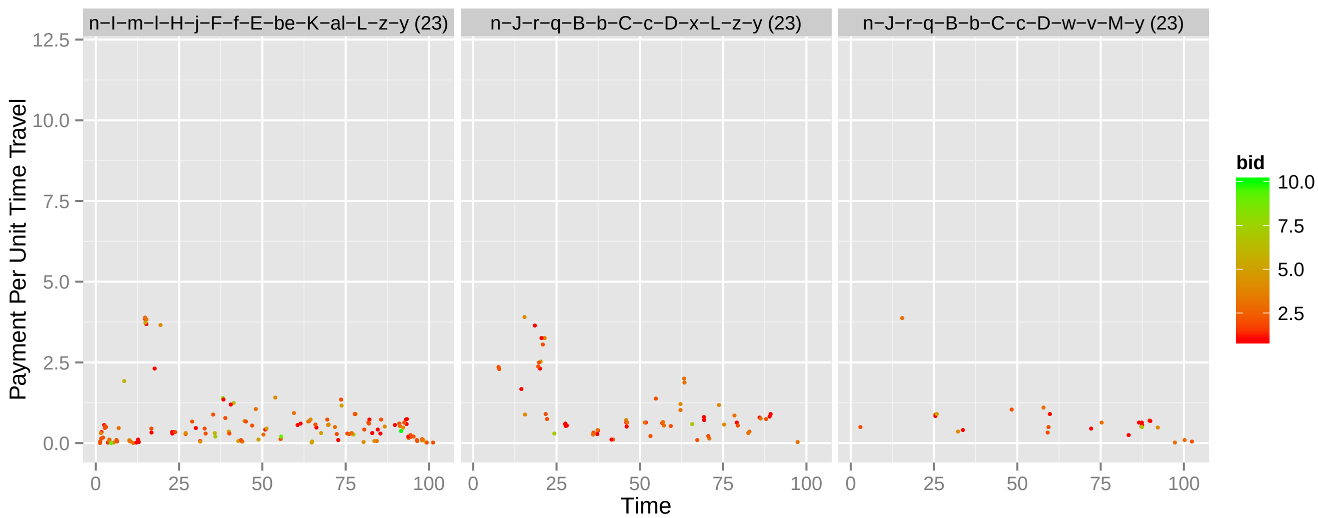


(a) Without Free Path

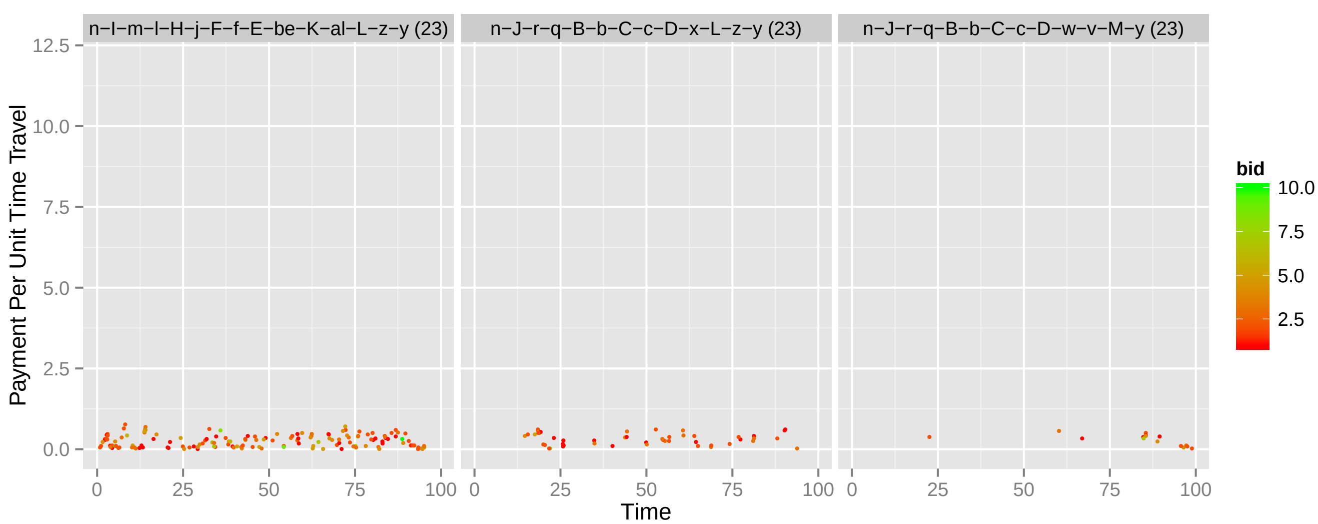


(b) With Free Path

Figure 3.18: Payment and Bid Relation of $s \rightarrow x$ With and Without Free Path

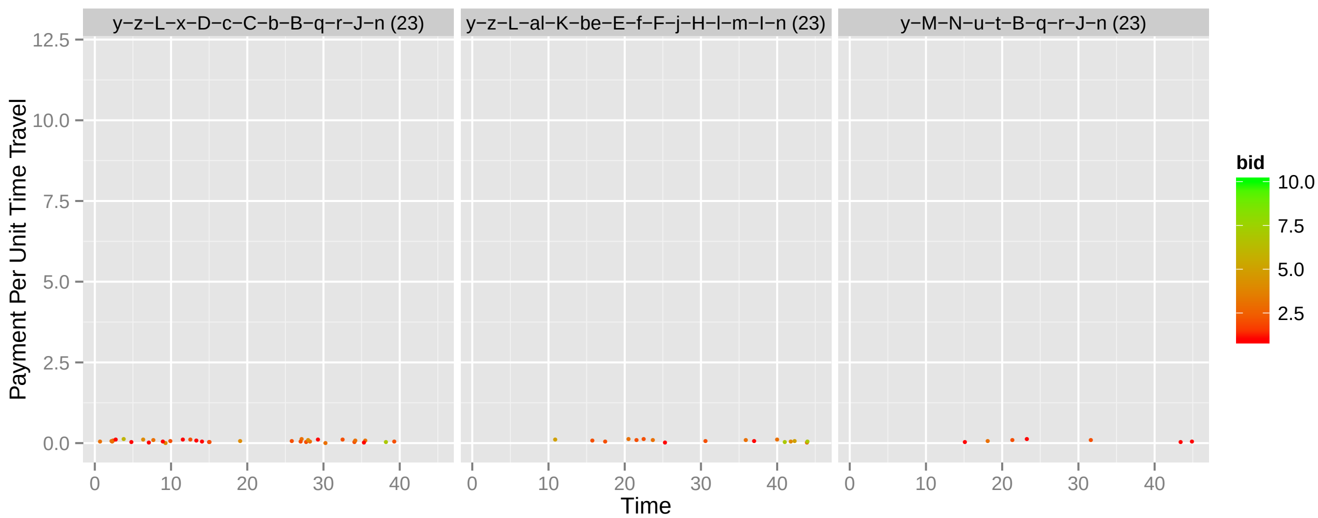


(a) Without Free Path

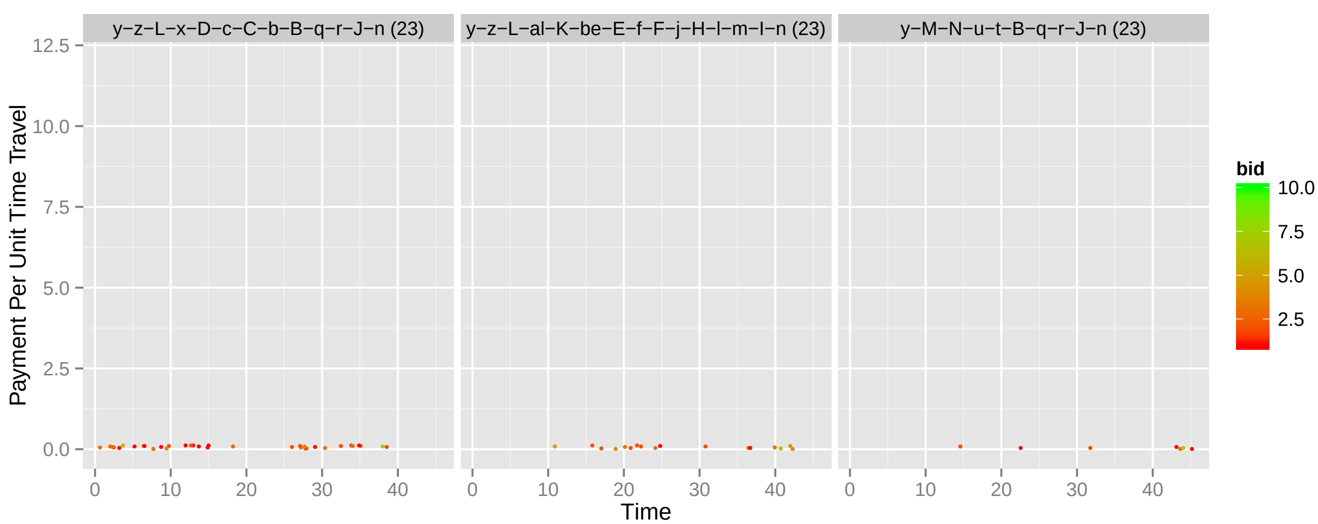


(b) With Free Path

Figure 3.19: Payment and Bid Relation of $n \rightarrow y$ With and Without Free Path

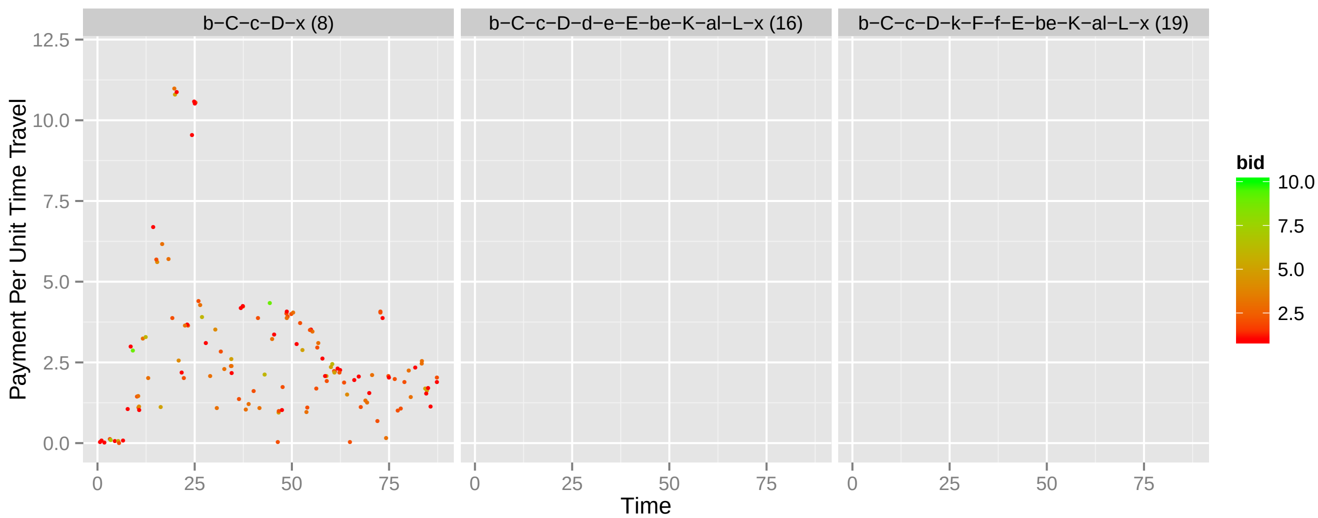


(a) Without Free Path

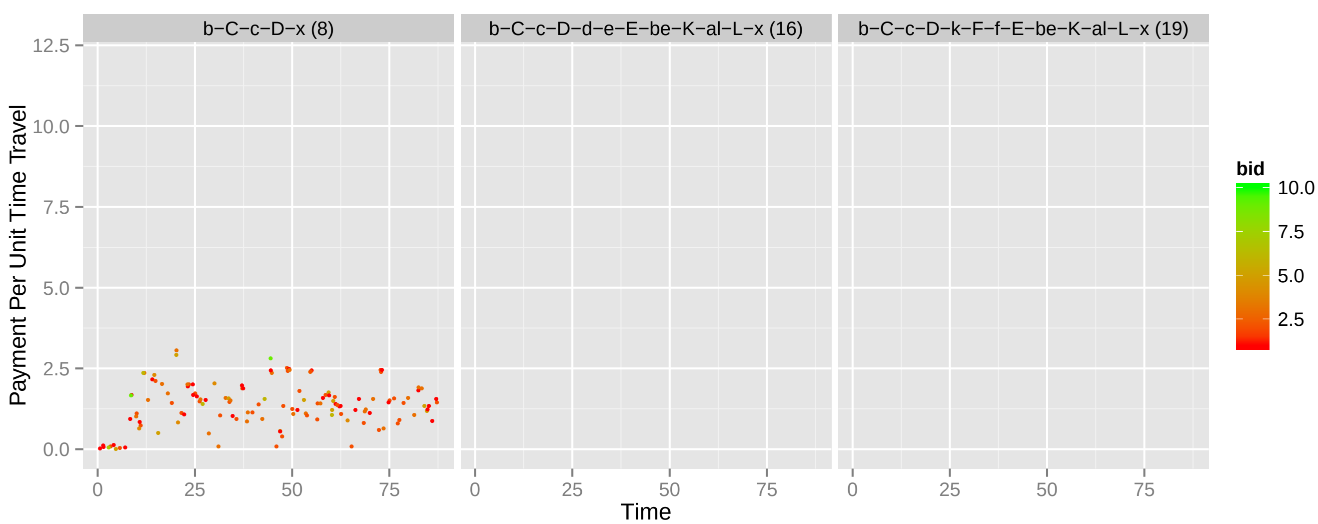


(b) With Free Path

Figure 3.20: Payment and Bid Relation of $y \to n$ With and Without Free Path



(a) Without Free Path



(b) With Free Path

Figure 3.21: Payment and Bid Relation of $b \to x$ With and Without Free Path

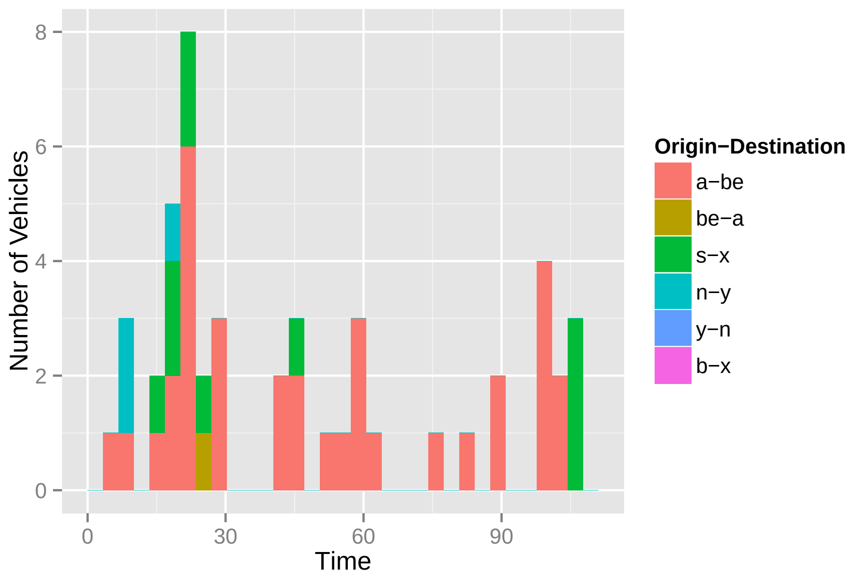


Figure 3.22: Number of Vehicles Assigned to Free Path

3.6 Conclusion and Future Work

We have proposed here an auction system implemented via V2I devices to toll and allocate traffic. Participating vehicles "bid" before travel. Using these bids, the traffic controller solves an optimization problem and assign paths and corresponding tolls to these vehicles. A mathematical model of the auction is presented and analyzed. The auctions system is based on VCG mechanism and thus guarantees truthful reporting of bids.

The auction scheme is tested on two numerical experiments: first in a small network of 6 links with 5100 vehicles, then in a large network of 98 links and 12000 vehicles. The computation of both of the experiments can be done in reasonable time limit. The result of experiment one shows that the auction indeed prescribes efficient tolls for each path at different time. Experiment two shows that in multiple origin-destination network, tolls can sometimes be larger than the corresponding bids. We fix this problem by adding an independent 'free path' for each origin-destination pair in the auction model. We show that by adding the free path, the tolls are guaranteed to be less than or equal to the bids.

We also analyze the computational difficulty of solving the payment problem and propose approaches to reduce the complexity. We also discuss methods to make this auction easier to implement in real world, such as using rolling horizon to allow for multiple-round auction, as well as the the possibility of implementing it as a tolling sub-system for HOV or HOT lanes.

There are three possible extensions of this work. 1) Changing the auction scheme or developing heuristics that reduces the computational complexity of auction. Here the classical VCG mechanism is used to determine the path assignment and toll, but there are other available auction mechanisms that do not involve solving Integer Programming problems. 2) Allow flexible travel time for vehicles. Instead of reporting a fixed travel time, vehicles can report a time window of travel. 3) Introducing stochasticity into the model. Instead of maintaining free-flow for each link in the network, we can allow congestion in certain links. This would require dynamically forecasting traffic flow in the network, and a more sophisticated model.

CHAPTER IV

Conclusion

Vehicle-to-Vehicle (V2V) and Vehcle-to-Infrastructure (V2I) communications are two important technologies used in Intelligent Transportation System (ITS). Here we develop two applications for ITS, one using V2V (Chapter II) and the other using V2I communication (Chapter III). Chapter II describes a vehicle platoon control model built on V2V devices, while Chapter III presents an application of a V2I system for a combinatorial auction mechanism to allocate traffic and generate tolls. These two models can be used separately, or used as sub-systems of an integrated ITS: the microscopic traffic is managed through platooning, and the macroscopic traffic is optimized by using a combinatorial auction.

In Chapter II, a vehicle platoon control method under high-latency communication environment is proposed. We use MPC approach which combines a statistical prediction model with an optimization algorithm and give optimal control action for each time step. We also analyze the robustness of this method using sensitivity analysis of quadratic programming. The simulation experiments performed show that the MPC controller can react quickly to sudden braking or accelerating of leading car, and dampen the effect of these actions as it propagates along the platoon. The simulation also demonstrates the potential of this method to operate vehicles safely.

In Chapter III, by using V2I devices, we apply a combinatorial auction in a network to toll and allocate traffic. Every participating vehicle bids in order to use a path in the network. We proposed a mathematical model to process these bids, and assign paths and corresponding tolls to vehicles. The auction system is based on VCG mechanism and thus guarantees truthful reporting of bids. We then test the auction mechanism in two numerical experiments: first in a small network of 6 links with 5100 vehicles, then on a large network of 98 links and 12000 vehicles. The result of experiment one shows that the auction indeed prescribes efficient tolls for each path at different time. Experiment two shows that in multiple origin-destination network, tolls can sometimes be larger than the corresponding bids. We fix this problem by adding an independent 'free path' for each origin-destination pair in the auction model. We show that by adding the free path, the tolls are guaranteed to be less than or equal to the bids. We also discuss methods for its implementation in the auction in real world, as well as the possibility of implementing it as a tolling sub-system for HOV or HOT lanes.

APPENDIX

APPENDIX A

A.1 Proof of Truthful Reporting Is a Best Strategy

Theorem 1. Truthful reporting is an optimal strategy for each vehicle driver in the auction mechanism. Moreover, when each vehicle driver reports truthfully, the outcome of the mechanism is one that maximizes social utility.

Proof. This is adapted from (*Cramton et al.*, 2006, Chap. 1).

Suppose each driver $i \in \mathbf{N}$ has a intrinsic value v_i^j for travelling in each path $j \in \mathbf{P}$. They report $\hat{v}_i^j$ to the central controller. Now we need to prove that reporting $\hat{v}_i^j = v_i^j, \forall j$ is a best strategy for each driver i.

Consider any fixed profile of reports $\{\hat{v}_i^j\}_{i \neq k}$ for all drivers besides k. Suppose that when driver k reports truthfully, the resulting allocation and payment vectors are denoted by $\mathbf{x}^* = \{x_i^j\}_{i \in \mathbf{N}, j \in \mathbf{P}}$ and $\boldsymbol{\pi}^* = (\pi_1, \pi_2, \ldots, \pi_N)$. But when driver k reports $\hat{v}_k^j$ for each path j, the resulting assignment are denoted as $\hat{\mathbf{x}} = (\hat{x}_1^*, \hat{x}_2^*, \ldots, \hat{x}_N^*)$, whereas the resulting payment is represented by $\hat{\boldsymbol{\pi}} = (\hat{\pi}_1, \hat{\pi}_2, \ldots, \hat{\pi}_N)$.

When vehicle k reports $\hat{v}_k^j$ for path j, his pay-off is:

$$
\begin{aligned}
&U_k(\hat{x}_k^*) - \hat{\pi}_k \\
=&U_k(\hat{x}_k^*) + \mathbf{U}_{-k}(\hat{\mathbf{x}}_{-k}^*) - \mathbf{U}_{-k}^* \\
\leq &\max_{\mathbf{x} \in S} \{U_k(x_k) + \mathbf{U}_{-k}(\mathbf{x}_{-k})\} - \mathbf{U}_{-k}^* \\
=&U_k(x_k^*) + \mathbf{U}_{-k}(\mathbf{x}_{-k}^*) - \mathbf{U}_{-k}^* \\
=&U_k(x_k^*) - \pi_k^*
\end{aligned}
$$

where S is defined as the set of $\mathbf{x}$ that satisfies constraint (3.8) to (3.10). $\square$

BIBLIOGRAPHY

BIBLIOGRAPHY

(2003), Standard specification for telecommunications and information exchange between roadside and vehicle systems 5 ghz band dedicated short range communications (dsrc) medium access control (mac) and physical layer (phy) specifications.

Bai, F., and H. Krishnan (2006), Reliability analysis of dsrc wireless communication for vehicle safety applications, *Proceedings of the IEEE Intelligent Transportation Systems Conference*.

Baskar, L., B. De Schutter, and H. Hellendoorn (2008), Model-based predictive traffic control for intelligent vehicles: Dynamic speed limits and dynamic lane allocation, in *Intelligent Vehicles Symposium, 2008 IEEE*, pp. 174 –179, doi: 10.1109/IVS.2008.4621307.

Baskar, L., B. De Schutter, J. Hellendoorn, and Z. Papp (2011), Traffic control and intelligent vehicle highway systems: a survey, *Intelligent Transport Systems, IET*, *5*(1), 38 –52, doi:10.1049/iet-its.2009.0001.

Bikhchandani, S., S. de Vries, J. Schummer, and R. V. Vohra (2002), Linear programming and vickrey auctions, *Mathematics of the Internet: E-auction and Markets*, *127*, 75–116.

Biswas, S., R. Tatchikou, and F. Dion (2001), Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety, *IEEE Communications Magazine*, *44*(1), 74 – 82, doi:10.1109/MCOM.2006.1580935.

Brewer, P. J. (1999), Decentralized computation procurement and computational robustness in a smart market, *Economic Theory*, *13*(1), 41–92.

Camacho, E., and C. Bordons (1995), *Model predictive control in the process industry*, Advances in industrial control, Springer-Verlag.

Clarke, E. H. (1971), Multipart pricing of public goods, *Public Choice*, *11*(1), 17–33.

Cramton, P. (1997), The fcc spectrum auctions: An early assessment, *Journal of Economics & Management Strategy*, *6*(3), 431–495.

Cramton, P., Y. Shoham, and R. Steinberg (2006), *Combinatorial auctions*, MIT Press.

Dunbar, W. B., and D. S. Caveney (2012), Distributed receding horizon control of vehicle platoons: Stability and string stability, *Automatic Control, IEEE Transactions on*, *57*(3), 620–633.

Friesz, T. L., C. Kwon, and R. Mookherjee (2007), A computable theory of dynamic congestion pricing, in *Transportation and Traffic Theory 2007. Papers Selected for Presentation at ISTTT17*.

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

Gupta, P., and P. Kumar (2000), The capacity of wireless networks, *Information Theory, IEEE Transactions on*, *46*(2), 388 –404, doi:10.1109/18.825799.

Hershberger, J., and S. Suri (2001), Vickrey prices and shortest paths: What is an edge worth?, in *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pp. 252–259, IEEE.

Horowitz, R., and P. Varaiya (2000), Control design of an automated highway system, *Proceedings of the IEEE*, *88*(7), 913 –925, doi:10.1109/5.871301.

Huang, C.-L., Y. P. Fallah, and R. Sengupta (2010), Adaptive intervehicle communication control for cooperative safety systems, *IEEE Network*, *24*(1), 6 – 13.

Lindsey, C. R., and E. T. Verhoef (2000), Traffic congestion and congestion pricing, *Tech. rep.*, Tinbergen Institute.

Liu, X., A. Goldsmith, S. Mahal, and J. Hedrick (2001), Effects of communication delay on string stability in vehicle platoons, in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pp. 625 –630, doi:10.1109/ITSC.2001.948732.

Liu, Y., F. Dion, and S. Biswas (2006), Safety assessment of information delay on performance of intelligent vehicle control system, *Transportation Research Record: Journal of the Transportation Research Board*, *1944*(1), 16–25.

Maciejowski, J. (2002), *Predictive control: with constraints*, Pearson Education, Prentice Hall.

Middleton, R. H., and J. H. Braslavsky (2010), String instability in classes of linear time invariant formation control with limited communication range, *Automatic Control, IEEE Transactions on*, *55*(7), 1519–1530.

Milanés, V., J. Villagra, J. Godoy, J. Simó, J. Pérez, and E. Onieva (2012), An intelligent v2i-based traffic management system, *Intelligent Transportation Systems, IEEE Transactions on*, *13*(1), 49–58.

Nisan, N., and A. Ronen (2000), Computationally feasible VCG mechanisms, *Electronic Commerce: Proceedings of the 2 nd ACM Conference on Electronic Commerce*, *17*(20), 242–252.

Pigou, A. C. (1912), *Wealth and welfare*, Macmillan and Co., limited.

Rajamani, R., and S. Shladover (2001), An experimental comparative study of autonomous and co-operative vehicle-follower control systems, *Transportation Research Part C: Emerging Technologies*, *9*(1), 15 – 31, doi:10.1016/S0968-090X(00)00021-8.

Rajamani, R., and C. Zhu (2002), Semi-autonomous adaptive cruise control systems, *Vehicular Technology, IEEE Transactions on*, *51*(5), 1186 – 1192, doi: 10.1109/TVT.2002.800617.

Rajamani, R., H.-S. Tan, B. K. Law, and W.-B. Zhang (2000), Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons, *Control Systems Technology, IEEE Transactions on*, *8*(4), 695 –708, doi:10.1109/87.852914.

Rassenti, S. J., V. L. Smith, and R. L. Bulfin (1982), A combinatorial auction mechanism for airport time slot allocation, *The Bell Journal of Economics*, pp. 402–417.

Schrank, D., and T. Lomax (1999), The 1999 annual mobility report: information for urban america.

Sheffi, Y. (2004), Combinatorial auctions in the procurement of transportation services, *Interfaces*, *34*(4), 245–252.

Shladover, S., et al. (1991), Automated vehicle control developments in the path program, *Vehicular Technology, IEEE Transactions on*, *40*(1), 114 –130, doi: 10.1109/25.69979.

Smith, R. S., and F. Y. Hadaegh (2007), Closed-loop dynamics of cooperative vehicle formations with parallel estimators and communication, *Automatic Control, IEEE Transactions on*, *52*(8), 1404–1414.

Swaroop, D., and J. Hedrick (1996), String stability of interconnected systems, *Automatic Control, IEEE Transactions on*, *41*(3), 349 –357, doi:10.1109/9.486636.

Teodorović, D., K. Triantis, P. Edara, Y. Zhao, and S. Mladenović (2008), Auction-based congestion pricing, *Transportation Planning and Technology*, *31*(4), 399–416.

Toh, R. S., and S.-Y. Phang (1997), Curbing urban traffic congestion in singapore: a comprehensive review, *Transportation Journal*, *37*(2), 24–33.

Varaiya, P. (1993), Smart cars on smart roads: problems of control, *Automatic Control, IEEE Transactions on*, *38*(2), 195 –207, doi:10.1109/9.250509.

Varaiya, P., and S. Shladover (1991), Sketch of an ivhs systems architecture, in *Vehicle Navigation and Information Systems Conference, 1991*, vol. 2, pp. 909 – 922, doi: 10.1109/VNIS.1991.205835.

Vickrey, W. (1961), Counterspeculation, auctions, and competitive sealed tenders, *The Journal of Finance*, *16*(1), 8–37.

Vickrey, W. S. (1969), Congestion theory and transport investment, *The American Economic Review*, *59*(2), 251–260.

Zhang, Y., B. Kosmatopoulos, P. Ioannou, and C. Chien (1999), Autonomous intelligent cruise control using front and back information for tight vehicle following maneuvers, *IEEE Transactions on Vehicular Technology,*, *48*(1), 319 –328, doi: 10.1109/25.740110.