Repeated Selfish Routing with Incomplete Information

Repeated Selfish Routing with Incomplete Information

By

He Yu, M.Sc.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Science

McMaster University

MASTER OF SCIENCE (2013) McMaster University
(Computing and Software) Hamilton, Ontario

TITLE: Repeated Selfish Routing with Incomplete Information

AUTHOR: He Yu, M.Sc.

SUPERVISOR: Dr. George Karakostas

NUMBER OF PAGES: x, 78.

Abstract

Selfish routing is frequently discussed. The general framework of a system of non-cooperative users can be used to model many different optimization problems such as network routing, traffic or transportation problems.

It is well known that the Wardrop user equilibria (i.e. the user optima) generally do not optimize the overall system cost in a traffic routing problem. In order to induce the equilibrium flow to be as close to the optimal flow as possible, the term "toll" is introduced. With the addition of tolls, a traffic system does not show the actual cost to the users but the displayed cost of users, which is the summation of the actual cost and the toll. A common behavioral assumption in traffic network modeling is that every user chooses a path which is perceived as the shortest path, then the whole system achieves the equilibrium of the displayed cost. It is proved that there exists an optimal toll which can induce the equilibrium flow under displayed cost to be the optimal flow in reality.

However, this conclusion holds only if the selfish routing executes only once. If the game is played repeatedly, the users will detect the difference between the actual and displayed costs. Then, they will not completely trust the information given by the system and calculate the cost. The purpose of this thesis is to find out the optimal strategy given by the system–how to set tolls in order to maintain the flow as close to the optimal flow as possible.

Acknowledgments

Foremost, I gratefully acknowledge my supervisor, Dr. George Karakostas, the thesis was written under the guidance and with the help of him. His valuable advice and extended knowledge of the area helped me to do my best while working on the thesis. Frequently meetings between us was a strong sense of motivation of my research. It is impossible for me to finish the thesis without his supervision.

It is my pleasure to thank Dr. Antoine Deza, Dr. Jeffery Zucker and Dr. Ryszard Janicki for outstanding performances in teaching courses. Their lectures inspire me in the research of computer theory.

My special thanks are to the members of the examination committee: Dr. Ryan Leduc (Chair), Dr. Douglas Down and Dr. George Karakostas.

Also, I am indebted to thank my family and friends for their patience, understanding and continuous support.

Contents

Abstract **iii**

Acknowledgments **v**

List of Figures **ix**

List of Tables **x**

1 Introduction and Background Knowledge **1**

1.1 Motivation 1

1.2 Our Results 3

1.3 Application 4

1.4 Introduction to Game Theory 4

1.5 Stackelberg Games 8

1.6 Repeated Games with Incomplete Information and Reputation . 10

1.7 Thesis Outline 13

2 Repeated Non-atomic Selfish Routing **15**

2.1 Selfish Routing with Tolls and User Equilibrium 15

2.2 Repeated Selfish Routing and Reputation 23

2.3 Reputation and Estimation Mechanisms 28
2.4 Stochastic Games 31
2.5 Existence of Perfect Equilibrium in Markov Strategies 34
2.6 The Upper Bound of Perfect Equilibrium Costs and the Optimal Strategy for the Manager 36
2.7 An Example for the Optimal Strategy of the Manager 52

3 Stochastic User Equilibrium 55
3.1 Repeated Stochastic Selfish Routing and Stochastic User Equilibrium 55
3.2 Existence of Optimal Tolls in Stochastic Selfish Routing 57
3.3 An Example for Stochastic User Equilibrium 61

4 Repeated Atomic Splittable Selfish Routing 63
4.1 Atomic Splittable Selfish Routing 63
4.2 Repeated Atomic Splittable Selfish Routing with Limited History 65
4.3 An Example for Repeated Atomic Splittable Selfish Routing . . 67

5 Conclusions and Future work 70
5.1 Conclusion 70
5.2 Future Work 71

Bibliography 74

List of Figures

1.1 The Product-Choice Game 6

1.2 An Example of Chance Move 11

1.3 The Game as Seen by P_2 11

2.1 A General Network 16

2.2 The Curve of C when $t^* + \beta \leq ub$ 39

2.3 The Curve of C when $t^* + \beta > ub$ 40

2.4 Strategy $s^{\triangle}$ 45

2.5 A General Cycle Strategy s 46

2.6 Strategy s^* 47

2.7 Strategy s_1 47

2.8 Stage Cost Curve of Strategy s^* 49

2.9 Stage Cost Curve of Strategy s_1 49

2.10 Strategy s_2 51

2.11 Stage Cost Curve of Strategy s_2 51

2.12 The Network in Example 53

4.1 The Network for Atomic Splittable Selfish Routing Model 67

List of Tables

2.1 Strategy s_1 and the Corresponding Cost for the Manager 46

2.2 Strategy s_2 and the Corresponding Cost for the Manager 49

2.3 Optimal Flow Pattern with $K = 1$ 53

4.1 Optimal Flow Pattern with $K = 1$ 69

Chapter 1

Introduction and Background Knowledge

1.1 Motivation

In daily life, people often come across such a situation: plenty of entities or data are transported from side to side in a network, which happens frequently in logistics management, urban transportation, distribution network system; etc. Multiple users use such networks of edges and nodes. In general, there are two kinds of nodes: some nodes denote special points such as terminals of bus systems, network terminals and cargo dispatch centers, from which traffic flow outflows, or to which traffic flow inflows. These nodes are called source nodes or destination nodes. Other nodes denote ordinary flow inflows and outflows from different edges. Each user travels from her own origin nodes to destination nodes, and they share the edges in the network; some of them may use the same origin and destination nodes, so they are grouped together as a homogeneous commodity, and such origin and destination nodes are called "origin and destination" pairs.

Each user wants to pick up the optimal route for herself to maximize the utility, or minimize her cost, and she does not take others' action and benefit into account. After a period of chaos, the flow achieves a stable state gradually, in which no one wants to change her route unilaterally. In other words, no user can get a better utility by changing her route when the other users remain the same. This state is an equilibrium.

However, at equilibrium, the performance of the network may not be optimal. At optimum, the total utility of the network is as large as possible. In order to optimize the efficiency of the network, the first instinct is introducing a central governor to control the flow directly. Nevertheless, this is unrealistic and the cost may be very high in practice. For example, the manager of an urban transportation system cannot order drivers what to do. Therefore, only selfish routing is applicable, where each user maximizes her own payoff and there is no coordination.

If the efficiency of a network in equilibrium is much lower than the optimum (which could be expected), the network is a "bad" network. In [21], Koutsoupias and Papadimitriou defined the term "price of anarchy" to describe the ratio of the total cost of the network between equilibrium and optimum state. Then we can try to improve the network by reducing the price of anarchy. Roughgarden proved that the price of anarchy is independent of the network topology[31]. In [31], he showed that the price of anarchy is determined only by the simplest network, and under some weak hypotheses, the worst price of anarchy is achieved by very simple networks.

Although selfish routing is the mode of operation, we can introduce a manager to impact the user of the network indirectly. She can improve the efficiency of the network by manipulating the information about the network status delivered to the selfish users. This is the motivation for this thesis.

In addition, we should notice that in reality the selfish routing scenario is not executed only once, but repeatedly. Some methods listed above may not apply to repeated selfish routing because users can discover hidden information from their previous experience. Therefore, we extend the research into repeated selfish routing.

1.2 Our Results

First, we build a model of general network with finite edges and nodes. There are special nodes paired as source and destination pairs. For each pair, infinite users travel from the source to the destination, and they make up a single commodity. Different commodities can share edges and nodes and each user in any commodity carries an infinitesimal amount of flow. They travel in this network repeatedly and will take a latency time to traverse each used edge. There is a manager in charge of this network, who wants to minimize the social cost (total latency time) of the network. She can induce the flow by giving to the users an artificial latency time for each edge. We also assume that the users can retain some information from past plays in this repeated games, which influence their current behavior. We prove the optimum state for the manager under various conditions and the corresponding behavior in optimum state for the manager.

Furthermore, we extend our model to the case where users in the same commodity perceive random latency time for identical edges, and the latency time obeys some known probability distribution. We prove that the optimal flow can be kept in repeated games under certain conditions.

Finally, we build a similar model for users, each of whom carries an in-negligible amount of flow. We prove that the optimal strategy for the manager is similar to the case of users with infinitesimal amounts of flow.

1.3 Application

Our model has application in reality. In real life, there are systems containing different kinds of networks. Mostly, these networks are shared by selfish users. It is easy to see that, in general, the network state induced by these selfish users is not optimum. Therefore, improving the efficiency of the network is always a big problem. The method of centralized control has several deficiencies: there are lots of technical difficulties and each selfish individual does not want to obey the rules. However, the manager can use artificial "tolls" (time, money, etc) to optimize the selfish routing network repeatedly. Our model gives an efficient method for using artificial tolls repeatedly. For example, in urban transportation, a municipality can use our methods to display artificial delays in major road crossings to maintain a better traffic condition everyday. For a service system with multiple service windows in restaurants, pubs, banks, etc, the model can be applied to control the waiting time of people. For example, plenty of users go to a restaurant with several service windows. Each user chooses the window with the shortest latency. The manager can tell artificial delays to users to enhance the efficiency of the service system.

1.4 Introduction to Game Theory

From above we can see that selfish routing satisfies the following characteristics:

1. it contains multiple users each of which wants to maximize her own payoff;
2. each user has several choices;
3. each user competes with other users;
4. each user's payoff depends on not only her own choice, but also other users' choices.

The above characteristics can easily make us associate our situation to "games" and "game theory". Game theory reflects environments where an individual's payoff depends on the choices of others, and is applied in economics, political science, psychology, logic and biology.

Game theory became an independent discipline since John von Neumann published his paper in 1928[37]. Games come in a variety of forms, but most of them have the following common attributes:

1. at least two players;
2. alternation of moves;
3. a possible lack of complete knowledge;
4. a payoff function to be maximized.

In addition, the moves chosen by one player may be unknown to others; sometimes, a probability distribution can be given over several moves. We use $\mathbb{R}$, $\mathbb{R}_+$ and $\mathbb{R}_{++}$ to denote reals, and nonnegative and positive reals, respectively. From above, we have

Definition 1.4.1 *A game consists of*

1. *n players, numbered $1, ..., n$;*

2. *a strategy set S_i for each player $i, i = 1, ..., n$, which consists of all her choices;*

3. *a payoff function for each player $i, i = 1, ..., n$, $p_i(s_1, ..., s_n) : \Pi_i S_i \mapsto \mathbb{R}$.*

Players will make their choices, and the mechanism for each player to make her choice is her strategy. The intuitive meaning of a strategy is a plan for playing a game. Namely, in the player's mind, she is saying to herself, "If such and such happen, I will act in such a manner"[28]. The strategies defined in definition 1.1 are also called pure strategy. If a player uses a pure strategy, her action will be deterministic. In addition, players can also use strategies randomly, which leads to the idea of a mixed strategy.

Definition 1.4.2 *A mixed strategy for a player is a probability distribution over the set of her pure strategies.*

		2	
		h	l
1	H	2,3	0,2
	L	3,0	1,1

Figure 1.1: The Product-Choice Game

Figure 1.1 is an example of a game, the product-choice game. Player 1 is a firm which can exert either high effort(H) or low effort(L) in the production of a product; Player 2 is a consumer who can buy either a high-priced product, h, or a low-priced product, l. Therefore, $S_1 = \{H, L\}, S_2 = \{h, l\}$. Each entry in Figure 1.1 denotes the payoff for both players, for example, $p_1(H, l) = 0, p_2(H, l) = 2$.

As each player can choose what she wants, the game can have plenty of states according to different strategies chosen by the players. We are mostly interested in a very stable state – the equilibrium. In this state, no player wants to change her strategy unilaterally. The intuition is that in this situation, every player will lose payoff if she changes her strategy unilaterally.

Definition 1.4.3 *[25]*

A strategy vector $(s_1^*, \ldots, s_n^*) \in S_1 \times \ldots \times S_n$ *is said to be in Nash equilibrium, if and only if, for any* $i = 1, \ldots, n$, *and* $s_i \in S_i$,

$$p_i(s_1^*, \ldots, s_{i-1}^*, s_i, s_{i+1}^*, \ldots, s_n^*) \leq p_i(s_1^*, \ldots, s_n^*).$$

The product-choice game in Figure 1.1 has a unique equilibrium (L, l).

A game is said to be a *zero-sum game* if and only if the payoff function satisfies

$$\sum_{i=0}^{n} p_i = 0.$$

For any player i, we say a strategy $\widehat{s}$ *dominates* $\overline{s}$, where $\widehat{s}, \overline{s} \in S_i$, if and only if $p_i(\widehat{s}, s_{-i}) \geq p_i(\overline{s}, s_{-i})$ for all s_{-i} (s_{-i} denotes an action profile of all the players other than i). A strategy $\widehat{s}$ *strictly dominates* $\overline{s}$, where $\widehat{s}, \overline{s} \in S_i$, if and only if $\widehat{s}$ dominates $\overline{s}$ and there is at least one s_{-i} such that $p_i(\widehat{s}, s_{-i}) > p_i(\overline{s}, s_{-i})$.

1.5 Stackelberg Games

In some games, players do not move simultaneously, but sequentially. Therefore, players can be divided into leaders and followers. This kind of game is called a Stackelberg game. The Stackelberg game is a strategic game in which leaders move first and followers move sequentially, and have some knowledge about the leaders' actions.

In the following example, we use P_1 and P_2 to denote the leader and follower, respectively. In each stage, P_1 moves first and she does not know the move of P_2; she guesses the potential strategy of P_2 in order to choose the strategy which can maximize her payoff at the worst condition, which is determined by P_2.

The strength of P_1 is priority of moving. Her weakness is the lack of information of her opponent's move. Conversely, the strength of P_2 is that she can observe P_1's behavior in order to respond with her best strategy. Her weakness is that she should follow her opponent's move.

For example, let P_1 and P_2 be two companies, which produce a homogeneous product in the same market. What follows is an example from [23]. Their strategy is about the quantity of product. P_1 produces the product in Q_1 quantity and the unit cost of the product for P_1 is B_1, while the quantity produced by P_2 is Q_2. Moving after P_1, P_2 knows all the information about P_1. However, P_1 does not know the unit cost of P_2's product. It just knows that P_2's unit cost is B_2^H with probability Y^H and B_2^L with probability Y^L, and they satisfy $B_2^H \geq B_2^L$, $Y^H + Y^L = 1$, and $B_1 = Y^H B_2^H + Y^L B_2^L$. We also assume that the unit market price of this product is MP. Then, we define $\alpha = MP + Q_1 + Q_2$.

Accordingly, P_1's utility function is

$$U_1 = Q_1(\alpha - Q_1 - Q_2 - B_1),$$

and P_2's utility function is

$$U_2^H = Q_2(\alpha - Q_1 - Q_2 - B_2^H)$$

or

$$U_2^L = Q_2(\alpha - Q_1 - Q_2 - B_2^L).$$

If we want to find the equilibrium for this problem, we need to apply backwards induction: we solve P_2 first. As P_2 wants to maximize its profit when playing against P_1, it chooses Q_2 to be

$$Q_2^{H*} = \frac{1}{2}(\alpha - Q_1 - B_2^H)$$

or

$$Q_2^{L*} = \frac{1}{2}(\alpha - Q_1 - B_2^L).$$

Now, P_1 maximizes her profit under the condition of P_2's optimal response. It takes into account P_2's strategy. Therefore, it chooses the quantity which can maximizes the utility function:

$$Y^H U_1(Q_1, Q_2^{H*}) + Y^L U_1(Q_1, Q_2^{L*}).$$

It is easy to see that

$$Q_1^* = \alpha - B_1 - 1/2[Y^H(\alpha - B_2^H) + Y^L(\alpha - B_2^L)].$$

Let $\tau = \alpha - B_1$, $\tau^L = \alpha - B_2^L$ and $\tau^H = \alpha - B_2^H$. Therefore, we have $\tau = Y^H\tau^H + Y^L\tau^L$. Then, we can get that the optimal solution is:

$$Q_1^* = \frac{1}{2}\tau, Q_1^{*L} = \frac{1}{2}(\tau^L - \frac{\tau}{2}) \text{ and } Q_2^{H*} = \frac{1}{2}(\tau^H - \frac{\tau}{2}).$$

The corresponding optimal utility is

$$U_1^* = \frac{1}{8}\tau^2, U_2^{H*} = 14(\tau^H - \frac{\tau}{2})^2, \text{ and } U_2^{L*} = \frac{1}{4}(\tau^L - \frac{\tau}{2})^2.$$

Then, P_2's average profit function is

$$U_2^* = \frac{1}{4}Y^H(\tau^H - \frac{\tau}{2})^2 + \frac{1}{4}Y^L(\tau^L - \frac{\tau}{2})^2.$$

We can see that if $U_1^* < U_2^*$, P_1 may lose the advantage of moving first.

1.6 Repeated Games with Incomplete Information and Reputation

In this section we discuss repeated games with incomplete information. A repeated game refers to the game defined in definition 1.1, which is called stage game, being played round by round, either finitely or infinitely. In repeated games, we use *action* to denote the choice of players in stage game and *strategy* to denote the behavior of players in repeated game. At the end of a stage game, each player discovers some information, which may be incomplete, about her payoff and her opponents' actions in that stage. This information is used by her during future plays. In other words, a player may be uncertain about the payoffs related to some of her actions before playing the stage game[15].

We use a probability distribution to describe this uncertainty. For example, there are two players, P_1 and P_2, and two zero-sum games, G_A and G_B, denoted by matrices in Figure 1.2. P_1 is the row player picking the best row for her and P_2 is the column player picking the best column for her. In each game, each player has two strategies, "L" and "R". The numbers in each entry denote the payoffs which P_2 gives to P_1. There is only one game being played.

However, if any player does not know which game is actually being played, she is uncertain about the payoffs related to her actions. What she can do is give a probability distribution over G_A and G_B. If she gives a probability distribution $(\frac{1}{2}, \frac{1}{2})$ to G_A and G_B, she thinks she is playing the game in Figure 1.3 in her mind.

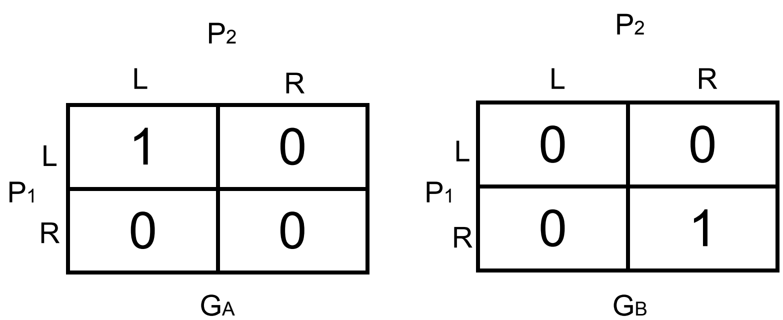

Figure 1.2: An Example of Chance Move

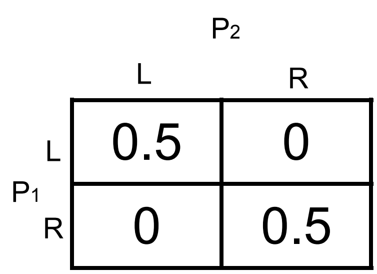

Figure 1.3: The Game as Seen by P_2

If one player has private information which is unknown by the opponents, repetition of a single stage game deeply changes her behavior. We give an example from [3], where the game is still Figure 1.2. In the example, P_1 knows which game is actually being played but P_2 does not. In G_A, "L" strictly dominates "R" for P_1. Therefore, ("L", "R") is in equilibrium and the equilibrium payoff is $(0, 0)$. Similarly, in G_B, ("R", "L") is in equilibrium and the equilibrium payoff is $(0, 0)$. If the stage game (Figure 1.2) is played once, we call it Γ_1, if it is played infinite rounds, we call it Γ_∞. If P_1 completely uses her information, she will always choose "L" when G_A is actually being played, or "R" when G_B is actually being played. For P_2, because the game seen by her is Figure 1.3, meaning that either choosing "L" or "R" gives the same expected payoff to her, thus, any pure or mixed strategy is optimal. We assume that P_2 chooses "L" with probability η, where $0 \leq \eta \leq 1$. Then, if G_A is being played, the equilibrium payoff of Γ_1 is $(\eta, -\eta)$; if G_B is being played, the equilibrium payoff of Γ_1 is $(1 - \eta, -1 + \eta)$. In either case, the equilibrium payoff of P_1 in Γ_1 is higher than in G_A or G_B, which means, P_1 can take advantage of her private information in one stage game Γ_1.

However, if G_A or G_B is played infinitely, P_2 can discover which game is actually being played from her opponent's action. If P_1 always chooses "L", P_2 knows that G_A is being played; if P_1 always chooses "R", P_2 knows that G_B is being played. Thus, P_2 can respond by choosing "R" or "L", respectively. This implies that P_1 only gets a payoff of 0 in each stage.

But P_1 can guarantee herself more than 1 in Γ_∞. She can play as if she does not know which game is actually being played by playing an infinite game Δ_∞, whose single stage Δ_1 is defined by the matrix in Figure 1.3.

In both Δ_1 and Δ_∞, the optimal strategies of each player is giving "L", "R" a probability distribution $(\frac{1}{2}, \frac{1}{2})$. Then P_1 gets a payoff of 0.25, which is higher than the payoff of P_1 in Γ_∞. In this case, P_1 can gain more than using her private information. Thus, we can conclude that the optimal strategy of P_1 in the repeated game is different from the single stage game.

However, if P_1 is myopic, meaning that she has a discounted payoff, she will try to maximize her payoffs in the initial stages and not care about that P_2 will discover which game is actually being played. Then, we can see that whether P_1 is patient or myopic ,her strategy is quite different. P_1's intention is her private information, which is not known by P_2. Therefore, P_2 assigns one "type" to each possibility of P_1. From this example, we know that for a player, her *type* represents her private information. Then, her opponents' uncertainty about her is described by a probability distribution over her types. This probability distribution is her reputation.

Definition 1.6.1 *If Player A is assigned a type set $\{T_i\}_{i=1}^{j}$ by other players, her reputation is the probability distribution over all the types $\{\mu(T_i)\}_{i=1}^{j}$, where $\mu(T_i) \geq 0$ and $\sum_{i=1}^{j} \mu(T_i) = 1$.*

1.7 Thesis Outline

In Chapter 2, we describe the general model of the selfish routing with a manager and specify the action of the manager. Then we extend the model to repeated games and explain the problems encountered in repeated games, but which do not arise in a single repetition. We introduce reputation mechanisms and estimation mechanisms to specify the problem accurately, and we generalize the model to a stochastic game. Then we show the properties of our game as a stochastic game and prove that certain kind of behaviors of the manager can be maintained in equilibrium paths. Then, we get optimal strategies for the manager under different conditions and the lower bound of the payoff for her. We show the influence of past actions on the current play and how reputation and estimation mechanisms work in repeated games. We also give examples to describe the whole procedure of a repeated selfish routing.

In Chapter 3, we modify the model defined in Chapter 2 to adapt stochastic user selfish routing. In stochastic user selfish routing, for any given flow, any edge cost perceived by users is not identical, so we treat it as a random variable. We give the definition of stochastic user equilibrium and optimal toll, which can induce the equilibrium flow to the optimal flow. Then we prove that in certain conditions, the social optimum can be maintained in the corresponding repeated game by using optimal tolls. Similar to Chapter 2, we give an example to illustrate the result in this chapter.

In Chapter 4, we modify the model defined in Chapter 2 again, but in another direction. We assume that users in each commodity are finite, and each user carries an innegligible amount of flow, which can be split into different paths. We introduce the same reputation mechanisms and estimation mechanisms described in Chapter 2 and show that the result proved in Chapter 2 can be maintained in this atomic splittable model if there are more restrictions on the cost function.

Chapter 2

Repeated Non-atomic Selfish Routing

2.1 Selfish Routing with Tolls and User Equilibrium

Selfish routing is a special kind of game. Selfish routing occurs in a multi-commodity flow network. The network is defined by a directed graph $G = (\mathcal{V}, \mathcal{E})$, with vertex set $\mathcal{V}$ and directed edge set $\mathcal{E}$. In the vertex set $\mathcal{V}$, there are some special vertex pairs called origin-destination(O-D) pairs. The set of all O-D pairs is denoted by $\mathcal{W}$. An acyclic sequence of edges connecting one origin to the corresponding destination is called a path, and for each O-D pair $w \in \mathcal{W}$, there is a corresponding non-empty path set $\mathcal{P}_w$, which consists of all paths connecting the given O-D pair. Then we define the path set $\mathcal{P} = \bigcup_{w \in \mathcal{W}} \mathcal{P}_w$. The cardinalities of $\mathcal{V}$, $\mathcal{E}$, $\mathcal{P}$, $\mathcal{W}$ and $\mathcal{P}_w$ are defined as V, E, P, W and P_w, respectively.

In this game, there are two kinds of players: users and a manager. Users are divided into W commodities and there is a one to one correspondence from commodities to O-D pairs. Each commodity contains a class of infinite users, each of whom travels with an infinitesimal amount of flow from the origin to the destination. The demand rate of commodity $w \in \mathcal{W}$ is r_w, which is fixed. The responsibility of the manager will be stated in following paragraphs.

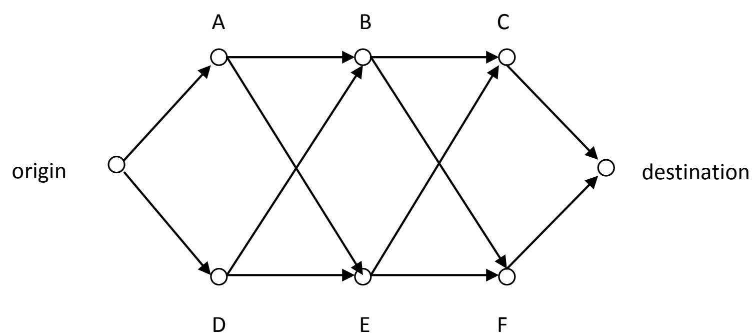

Figure 2.1: A General Network

Figure 2.1 is an example of a network containing one origin-destination pair and 6 other nodes, with 12 edges and 8 paths, so each user has 8 choices to travel from source to destination.

There are two kinds of selfish routing: atomic selfish routing and non-atomic selfish routing. In atomic selfish routing, each user carries a non-negligible amount of flow and if she changes her path, the total flow will be influenced; in non-atomic selfish routing, each user carries an infinitesimal amount of flow and an individual's change of strategy will not influence the total flow. In this chapter we discuss non-atomic selfish routing and we will discuss atomic selfish routing in Chapter 4.

We assume that all vectors are column vectors unless otherwise stated. We describe the routes chosen by users using a *path flow vector* h, each path flow vector can be denoted by $h = [h_p : p \in \mathcal{P}] \in \mathbb{R}_+^P$, and each component of

h denotes the flow on path p. We use the vector $h_w = [h_p : p \in \mathcal{P}_w]$ to denote the path flow vector for $O - D$ pair w, then h can be written in partitioned form as $h = [h_1^T, ..., h_W^T]^T$, where h_w^T is the transposed matrix of h_w. For each path flow vector h, there is a corresponding *edge flow vector*, f, which is a E-dimension vector. We define an *edge-path incidence matrix* to describe the relationship between edges and paths, the edge-path incidence matrix Θ is a $E \times P$ dimensional matrix and its entries are:

1. $\Theta_{ep} = 1$, if edge e is contained in path p;

2. $\Theta_{ep} = 0$ otherwise.

Of course, $\Sigma_p \Theta_{ep} > 0$ for each $e \in \mathcal{E}$, indicating each edge is a part of some paths.

Therefore, the relationship between f and its corresponding h is:

$$f = f(h) = \Theta h = [\Theta_1, ..., \Theta_W] \cdot [h_1, ..., h_W]^T = \sum_{w \in \mathcal{W}} \Theta_w h_w.$$

Notice that the edge-path incidence matrix, Θ, is expressed in partitioned form. As the demand rate of commodity w is r_w, we have $\Sigma_{p \in \mathcal{P}_w} h_p = r_w$ for all $w \in \mathcal{W}$. Also, we define the *demand rate vector* $r = [r_w, w \in \mathcal{W}] \in \mathbb{R}_{++}{}^W$.

In selfish routing, we discuss the opposite function of the payoff function, which is the "cost function". The *cost* for each user is the latency time for her to travel from the origin to the destination. The *edge cost function* is defined by $c^{edge}(f) : \mathbb{R}_+^E \mapsto \mathbb{R}_+^E$. We assume that cost functions are always nonnegative, continuous and nondecreasing. The corresponding *path cost function* c is $c(f) = \Theta^T c^{edge}(h)$, meaning that the cost of a path is the summation of the costs of the constituent edges. Notice that users discussed in this thesis are all homogeneous users, meaning that users in different commodities have the same sensitivity about the edge costs.

Then, we can define non-atomic selfish routing by a triple of the form (G, r, c). We denote the amount of flow using paths that contain the edge e.

Definition 2.1.1 *A feasible flow satisfies:*

1. $f_e \geq 0, \forall e \in \mathcal{E}$;

2. $\Sigma_{p \in \mathcal{P}_w} h_p = r_w, \forall w \in \mathcal{W}$.

Since we expect that each user attempts to minimize her cost, we arrive at the following definition[26],

Definition 2.1.2 *Let h be a feasible path flow for (G,r,c). The flow h is an equilibrium flow if, for each $w \in \mathcal{W}$ and every pair $p, p^* \in \mathcal{P}_w$ with $h_p > 0$,*

$$c_p(h) \leq c_{p^*}(h).$$

In other words, all paths used by users in an equilibrium flow h have minimum-possible cost[26]. In particular, all paths of a given commodity used in an equilibrium flow have equal cost. Beckmann et al. proved that every non-atomic instance admits at least one equilibrium flow and all equilibrium flows of a non-atomic instance have equal cost and identical edge flow vector[4].

The equilibrium strategy implies that no user can decrease her cost by changing her strategy unilaterally, thus, the cost of all paths for the same origin-destination pair are equal.

The total cost for (G, r, c) is $C(h) = \sum_{p \in \mathcal{P}} c_p(h) h_p$. In the edge flow form, we can express it as: $C(f) = \sum_{e \in \mathcal{E}} c_e^{edge}(f_e) f_e$. For an instance (G, r, c), we call a feasible flow an optimal flow if it minimizes the social cost over all feasible flows. Then we can have the definition of price of anarchy of non-atomic selfish routing.

Definition 2.1.3 *Price of anarchy of non-atomic selfish routing is the ratio between the cost of an equilibrium flow and that of an optimal flow.*

Roughgarden and Tardos proved that if the cost function of each edge is a linear function of edge flow, the price of anarchy is at most $4/3$; if the cost functions are assumed only to be continuous and nondecreasing in the edge flow, the price of anarchy is at most 2[32].

It is easy to see that the equilibrium flow is not equal to optimal flow in general. Recall that there is another player, the manager, playing this game. She is in charge of displaying the cost of each edge to users. The manager can add nonnegative artificial delays to the actual costs of all edges. The artificial delays for all edges are denoted by an *edge toll vector*. Then, the displayed cost of each edge for users is the summation of the actual cost and the toll. It needs emphasizing that the users do not experience the "toll" latency time, it is just a "trick" by the manager. A common behavioral assumption in traffic network modeling is that every user chooses a path that she perceives as being the shortest under the prevailing traffic conditions. The whole system achieves the equilibrium flow of the displayed cost. Karakostas, and Kolliopoulos[20], and Fleischer et al.[10] proved that this can be the optimal flow of the actual costs by adding a suitable nonnegative edge toll vector if the cost functions are nonnegative and nondecreasing. The edge toll vector inducing the optimal flow is the *optimal edge toll vector*.

The optimal edge toll vector can be calculated by (System 1)[20]. We use $u = [u_w : w \in \mathcal{W}]$ to denote the least costs for all commodities, $\widehat{h}$ to denote the optimal path flow vector. Notice that $t^{edge} = \Theta t$ and $\widehat{f} = \Theta \widehat{h}$.

$$\begin{aligned}
h_p(c_p(\widehat{h}) + t_p - u_w) = 0 &\quad p \in \mathcal{P}_w,\ w \in \mathcal{W} \\
c_p(\widehat{h}) + t_p - u_w \geq 0 &\quad p \in \mathcal{P}_w,\ w \in \mathcal{W} \\
u_w(\sum_{p \in P_w} h_p - r_w) = 0 &\quad w \in \mathcal{W} \\
\sum_{p \in P_w} f_p - r_w \geq 0 &\quad w \in \mathcal{W} \\
t_e^{edge}(f_e - \widehat{f}_e) = 0 &\quad e \in \mathcal{E} \\
f_e \leq \widehat{f}_e &\quad e \in \mathcal{E} \\
h_p, t_e^{edge}, r_w \geq 0 &\quad p \in \mathcal{P}, e \in \mathcal{E}, w \in \mathcal{W}
\end{aligned} \qquad \text{(System 1)}$$

Different commodities may have various sensitivites to the tolls. In this thesis, we assume that all commodities have the same sensitivity. In other words, they are homogeneous, meaning that users in different commodities have the same toll for the same edge. We denote the *optimal edge toll vector* by t^{edge*}. We also give an assumption about the manager's behavior.

Assumption 1 The *feasible edge toll vector set* is $\mathcal{T} = \{t^{edge} : t^{edge} = \theta t^{edge*}, 0 \leq \theta \leq q\}$, where $q \geq 1$. θ is the *toll factor*, and q is the upper bound of the toll factor.

Assumption 1 implies that any feasible edge toll vector is in the same direction of the optimal toll vector, the only difference between them is the scale. Then we can say that the manager cannot increase the toll on one edge faster than any other edges. Therefore, we can discuss tolls in a one-dimension space. Given a feasible edge flow vector f, the *displayed edge cost vector* is $c^{d,edge} = c^{edge}(f) + t^{edge}$. Similarly, the *feasible path toll vector set* is given by $\mathcal{U} = \{t : t = \Theta^T \cdot t^{edge}, t^{edge} \in \mathcal{T}\}$. The *optimal path toll vector* is $t^* = \Theta^T \cdot t^{edge*}$. So we can use a vector $ub = q \cdot t^*$ to denote the *upper bound* of the path toll vectors. The *displayed path cost vector* is $c^d = c(h) + t$. The *feasible displayed path cost vector set* is given by $\mathcal{Q} = \{c^d : c^d = c(h) + t, t \in \mathcal{U}\}$. If each element of a path toll vector is 0, we use $\mathbf{0}$ to denote it. We define a tiny threshold $0 < \omega << 1$, such that, if $t < \omega\, t^*$, the virtual players treat the path toll vector as $\mathbf{0}$.

For each $w \in \mathcal{W}$, we can create a corresponding *virtual player* $q(w)$ to represent the behavior of all users in w. Given a path flow vector h, we define the set of paths with largest cost by $\mathcal{B}(h) = \{\widetilde{p} : \widetilde{p} = arg\max_{p:p\in\mathcal{P}_w,h_p>0} c_p(h)\}$, and the set of other paths with positive flow by $\mathcal{A}(h) = \{p : p \in \mathcal{P}_w - \mathcal{B}(h), h_p > 0\}$. Let

$$M(h) = \max_{p:p\in\mathcal{P}_w,h_p>0} c_p(h). \tag{2.1.1}$$

The cost of $q(w)$ is given by $c_{q(w)} = M(h) + \sum_{p:p\in\mathcal{P}_w,h_p>0}(M(h) - c_p(h))$.

Lemma 2.1.1 *$\forall w \in \mathcal{W}$, all $p \in \mathcal{P}_w$ satisfying $h_p > 0$ have equal cost when the cost of $q(w)$ is minimized over all possible path flow vectors.*

Proof:

We assume that when the cost of $q(w)$ is minimized over all possible path flow vectors, the path flow vector is h^* and not all $p : p \in \mathcal{P}_w, h_p > 0$ have equal cost. Therefore, there exists some $\overline{p} \in \mathcal{A}(h)$, such that $M(h) > c_{\overline{p}}(h^*)$. Let $p^* \in \mathcal{B}(h^*)$ and ε be a small enough positive number. A path flow vector is given by $\overline{h} = [\overline{h}_{p^*} = h^*_{p^*} - \varepsilon, \overline{h}_{\overline{p}} = h^*_{\overline{p}} + \varepsilon, \overline{h}_p = h^*_p, \forall p \in \mathcal{P}_w - \{p^*, \overline{p}\}]$.

Let $f^* = \Theta h^*, \overline{f} = \Theta\overline{h}$. There is at least one edge e^* being used by path p^* but not used by path $\overline{p}$, thus, $\overline{f}_{e^*} = f^*_{e^*} - \varepsilon$. Because the edge cost function is nondecreasing, $c_{e^*}(\overline{f}_e) \leq c_{e^*}(f^*_e)$, therefore, $c_{p^*}(\overline{h}) \leq c_{p^*}(h^*)$. Similarly, $c_{\overline{p}}(\overline{h}) \geq c_{\overline{p}}(h^*)$.

1. If either $c_{p^*}(\overline{h}) < c_{p^*}(h^*)$ or $c_{\overline{p}}(\overline{h}) > c_{\overline{p}}(h^*)$ holds, $M(\overline{h}) \leq M(h^*)$;
2. If $c_{p^*}(\overline{h}) = c_{p^*}(h^*)$ and $c_{\overline{p}}(\overline{h}) = c_{\overline{p}}(h^*)$, we can increase ε to make either $c_{p^*}(\overline{h}) < c_{p^*}(h^*)$ or $c_{\overline{p}}(\overline{h}) > c_{\overline{p}}(h^*)$;
3. If when ε is increased to $h^*_{p^*}$, $c_{p^*}(\overline{h}) = c_{p^*}(h^*)$ and $c_{\overline{p}}(\overline{h}) = c_{\overline{p}}(h^*)$ still hold, which means $c_{p^*} > c_{\overline{p}}$ holds during this procedure. Then we can say that all flow on p^* should be moved to $\overline{p}$ when c_q is minimized and p^* should be deleted from $\mathcal{B}(h^*)$.

Because p^* is arbitrarily selected from $\mathcal{B}(h^*)$, the above conclusion holds for each $p^* \in \mathcal{B}(h^*)$, meaning that the cost of $q(w)$ has not been minimized yet, which contradicts our assumption.

Therefore, the state where the cost of virtual player $q(w)$ is minimized is equivalent to the equilibrium for users. Then we can use virtual player $q(w)$ to simulate the behavior of users in commodity w.□

A pure action of $q(w)$ is a feasible path flow vector h_w. The feasible path flow vector for commodity w is given by $\mathcal{Y}_w$. Thus the action profile set is $A = \Pi_{w\in\mathcal{W}}\mathcal{Y}_w \times \mathcal{Q}$, where each element is denoted by $a = (h_1, ..., h_W, c^d(h))$.

Therefore, we can specialize Definition 1.4.1 for our stage selfish routing,

Definition 2.1.4 *The stage selfish routing consists of*

1. $W + 1$ players, $q(w), w \in \mathcal{W}$ and manager m;

2. *the action set for* $q(w)$ $\mathcal{Y}_w$*, and the action set for* m Q*;*

3. *the cost function for* $q(w)$ $c_{q(w)} = M(h) + \sum_{p:p\in\mathcal{P}_w, h_p>0}(M(h) - c_p(h))$*, where* $M(h)$ *is defined in* (2.1.1)*, and the cost function for* m $C(h) = \Sigma_{p\in\mathcal{P}} c_p(h) h_p$.

2.2 Repeated Selfish Routing and Reputation

If the game is extended to a repeated game, the virtual players will discover that the actual cost is not equal to the displayed cost after several stages. Then, they will not fully trust the manager anymore, and work out the costs which are supposed to be the most possibly correct by themselves.

This is quite common in real life. Imagine that plenty of users go to a bank for deals. The bank has several service windows. Each user chooses the window for which the waiting time is the shortest, namely, choose the "shortest" path. According to the Wardrope principle[32], the costs of all used paths are equal and less or equal than unused paths at equilibrium. If the bank manager uses the "optimal toll" technology, she can keep the efficiency of the bank in the optimal status in the first few days. However, after the users discover that the displayed cost is not always equal to the actual cost, they will not fully trust the manager.

In this game, the manager has complete information: she knows the response of the virtual players to whatever alternative she chooses and the costs in any case, she knows the optimal flow, and she can use the optimal toll vector in the initial stages or not. If she always uses the optimal toll vector or the toll vector near it, she can maintain the network in a very efficient state during the initial stages. However, because the differences between the actual costs and the displayed costs are large, the manager will not be trusted and the virtual players will not follow her. Then the efficiency of network will decrease quickly.

If the manager chooses the opposite strategy–not add tolls on the edges or add tiny tolls, she cannot maintain the network in an efficient state in the initial stages but can make the virtual players always trust her strategy, which gives a profit to her in the future. Thus, our purpose is to find the optimal strategy of the manager.

Because the action sets of the virtual players and the manager are all continuous spaces, we can always treat their actions as pure actions[24]. In repeated games, some players are *long-run* players who play in all the stages, and some players are *short-run* players who play only once but observe some previous plays. The manager and the virtual players are all long-run players. However, because each user only carries an infinitesimal amount of flow, her behavior can be neglected in the total flow, and cannot affect the future behavior of any player (including herself). For this reason, users, whose individual behaviors are unobserved, are also called anonymous[24]. As there is no link between the current action of a user and her future treatment, she cannot make a higher payoff in the future stages by sacrificing her current payoff. Therefore, every user just wants to minimize her cost in the current stage. Consequently, any virtual player can be deemed as a short-run player[24].

We assume that the virtual players only remember the actions of the manager in the last K stages. The virtual players cannot remember their previous actions: if they could, they could infer the actions of the manager more than the last K stages[22]. A *history set* for virtual players is $\mathcal{H}^n =$

$\mathcal{U}^n, n = 0, \ldots, K$, where $\mathcal{U}^n$ is the n-fold product of $\mathcal{U}$. We use the toll vector set $\mathcal{U}$ instead of the displayed cost vector set $\mathcal{Q}$ because virtual players can discover the toll vectors in history, so they only need remember the toll vectors. A history $b^n \in \mathcal{H}^n$ is a list of n action profiles of the manager. Then we have that the set of *all possible histories* is

$$\mathcal{H} = \bigcup_{n=0}^{K} \mathcal{H}^n.$$

A *strategy for virtual player* $q(w)$ is a mapping

$$s_w : \mathcal{H} \mapsto \mathcal{Y}_w.$$

We assume that the manager has a full history of the previous plays. However, in the following, we can see that only the history remembered by the virtual players can influence this repeated game. Therefore, a *strategy for the manager* is a mapping

$$s_m : \mathcal{H} \mapsto \mathcal{Q},$$

and the strategy set for the manager is denoted by S_m.

For any history $b^n \in \mathcal{H}$, we define the *continuation game* to be the infinitely repeated game that begins in period n, following history b^n. For any strategy profile s, the *continuation strategies* of the virtual players and the manager induced by b^n, are

$$s_w|_{b^n}(b^\tau) = s_w(b^n b^\tau), \forall b^\tau \in \mathcal{H}, w \in \mathcal{W},$$

and

$$s_m|_{b^n}(b^\tau) = s_m(b^n b^\tau), \forall b^\tau \in \mathcal{H},$$

respectively, where $b^n b^\tau$ is the concatenation of the history b^n followed by the history b^τ.

In stage n, a strategy profile $s = s_1, ..., s_W, s_m$ yields an action profile $a^n(s)$, which implies a cost $C(a^n(s))$ for the manager. The normalized discounted cost for the manager from the infinite sequence of plays is

$$\mathcal{C}(s) = (1-\delta)\sum_{n=0}^{\infty} \delta^n C(a^n(s)).$$

Given history b^i, the *continuation cost* for the manager is

$$(1-\delta)\sum_{n=i}^{\infty} \delta^n C(a^n(s)).$$

As the virtual players myopically optimize in every single stage, the game played by them is still the stage game in Definition 2.1.4. Therefore, we can still use the concept of "Nash equilibrium" in Definition 1.4.3 for them. Nevertheless, the manager, who is a long-run player, should consider the scope of the infinitely repeated game. We give the definition of Nash equilibrium in repeated games:

Definition 2.2.1 *The strategy profile $s = (s_1, ..., s_W, s_m)$ is a Nash equilibrium of the repeated game for the manager if for all $\widehat{s}_m \in S_m$,*

$$\mathcal{C}(s_1, ..., s_W, s_m) \le \mathcal{C}(s_1, ..., s_W, \widehat{s}_m).$$

A *Nash equilibrium path* is an infinite sequence of action profiles induced by a Nash equilibrium. In repeated games, the Nash equilibrium is too permissive. If the players are given a truncated history which cannot appear in a Nash equilibrium path, this Nash equilibrium may not give the optimal behavior for following plays in this circumstance[24]. Therefore, we need a "stronger" equilibrium, which requires behavior to be optimal in all circumstances, both those which appear in equilibrium and those which appear out of equilibrium. We give the definition of "subgame perfect equilibrium" to describe the sequential rationality of the manager:

Definition 2.2.2 *A strategy profile s is a subgame perfect equilibrium for the manager if for all histories $b^n \in \mathcal{H}$, $s|_{b^n}$ is a Nash equilibrium of the repeated game.*

Similarly, a *subgame perfect equilibrium path* is an infinite sequence of action profiles induced by a subgame perfect equilibrium. Because the virtual players are myopic, given the actions of the manager, they should play a Nash equilibrium of the stage game. We define

$$B : \mathcal{Q} \mapsto \Pi_{w \in \mathcal{W}} \mathcal{Y}_w$$

to be the correspondence that maps pure actions of the manager to the corresponding set of static Nash equilibrium for the virtual players, which can also be deemed as the mapping from pure actions of the manager to the corresponding *best responses* of the virtual players. Notice that given a displayed cost, the equilibrium flow under this cost is unique. Then, we can get the manager's Stackelberg cost

$$C^* = \min_{c^d(h) \in \mathcal{Q}} \max_{h \in B(c^d(h))} C(h, c^d(h)),$$

and the manager's Stackelberg action

$$c^{d*}(h) = arg \min_{c^d(h) \in \mathcal{Q}} \max_{h \in B(c^d(h))} C(h, c^d(h)),$$

where $c^d(h)$ is the displayed cost vector given by the manager when the flow is h.

Because the virtual players move after the manager, our game is a Stackelberg game. However, they still have uncertainties about the game. They do not know the actual cost function of each path and they cannot predict what tolls the manager adds to each path in the current stage. As we said, in this repeated game, the manager should choose a balance between current and future benefits: she can use the optimal tolls or the approximate optimal tolls in the current stage to make a high profit, but build a bad reputation and damage the profit in the future, or not use tolls in the current stage to build a good reputation. The manager's type set is given by $\mathcal{X} = \{T_1, T_2\}$, where T_1 is the honest type that shows the actual cost and T_2 is the dishonest type that shows the cost with tolls.

2.3 Reputation and Estimation Mechanisms

In this section, we give the reputation mechanism for the manager and the estimation mechanism about the actual cost for the virtual players. We assume that these mechanisms are known by the manager. We use a superscript to denote the number of stages. In stage n, virtual player $q(w)$ learns history $b_w^n = \{t_w^k = [t_p^k : p \in \mathcal{P}_w]\}_{k=n-K}^{n-1}$. In stage n, the reputation (Definition 1.6.1) of the manager given by virtual player $q(w)$ is $\{\mu(T_1|b_w^n), \mu_w^n(T_2|b_w^n)\}$. Especially, the prior probabilities are $\mu(T_1|b^0) = 1$ and $\mu(T_2|b^0) = 0$, where $b^0 = \emptyset$, for all virtual players, meaning that the manager is trusted by virtual players at the beginning of the game. The reputation for the manager is a function of the history, which consists of toll vectors in past K stages. Therefore, the virtual players use toll vectors in the history to predict the probability of types in the current stage. Because the virtual players cannot predict exactly the probability distribution of the tolls, they use the nonparametric estimation to estimate the probability of types[1]. The Gaussian kernel estimator, which uses a smooth weight function to estimate the distribution of random variables from previous data, is the most popular estimator[1]. So we assume that the virtual players use the Gaussian kernel estimator. The Gaussian kernel[1] is:

$$\mathcal{K}(u) = \frac{1}{\sqrt{\pi}} exp\left(-\frac{u^2}{2}\right).$$

We define a discriminant factor ρ_w^n for each commodity in each stage. If $t_w^n < \omega t^*$, $\rho_w^n = 1$; otherwise, $\rho_w^n = 0$ (ω is defined in Page 21). Then we define two discriminant functions, $g_1(b_w)$ for T_1, and $g_2(b_w)$ for T_2, respectively. $g_1(b)(g_2(b))$ describes the possibility of $T_1(T_2)$ in virtual player $q(w)'s$ mind if receiving history b_w. We have

$$g_1(b_w^n) = \sum_{k=n-K}^{n-2} \mathcal{K}(\frac{t_w^{n-1} - t_w^k}{t^*})\rho_{w,1}^k + \frac{1}{\sum_{k=n-K}^{n-1} \mathcal{K}(\frac{ub - t_w^k}{t^*})(1 - \rho_{w,1}^k)}, \tag{2.3.2}$$

$$g_2(b_w^n) = \sum_{k=n-K}^{n-2} \mathcal{K}(\frac{t_w^{n-1} - t_w^k}{t^*})(1 - \rho_{w,1}^k). \tag{2.3.3}$$

Then,

$$\mu(T_1|b_w^n) = \frac{g_1(b_w^n)}{g_1(b_w^n) + g_2(b_w^n)},$$

$$\mu(T_2|b_w^n) = \frac{g_2(b_w^n)}{g_1(b_w^n) + g_2(b_w^n)}.$$

We have following properties about the reputation mechanism.

1. If $b_0 = \{\underbrace{\mathbf{0}, \ldots, \mathbf{0}}_{K"\mathbf{0}"s}\}, \mu(T_1|b_0) = 1.$

2. If $t^n > \widehat{t^n}, \mu(T_1|\{b^{n-1}, t^n\}) < \mu(T_1|\{b^{n-1}, \widehat{t^n}\}).$

The first property shows that if the manager is honest in the history, she will be trusted; the second property shows that if she uses a smaller toll vector, she will have a better reputation. In general, the virtual players do not fully trust the displayed cost vector, but they cannot observe the actual cost vector. Therefore, after receiving a history $b \in \mathcal{H}^K$ and the current displayed cost vector c^d, they calculate the *perceived cost vector* $c^r = c^r(c^d, b)$ and they believe that the perceived cost vector is the actual cost vector. For each path $p \in \mathcal{P}_w$, the virtual player $q(w)$ learns the tolls added on p in history, which are $\{t_p^k\}_{k=n-K}^{n-1}$. Similarly, the virtual players still use the Gaussian kernel estimator to estimate the density of the tolls. Therefore, the estimated density function of t_p^n is

$$\hat{p}(t_p^n) = \frac{1}{K d_K(t_p)} \sum_{k=n-K}^{n-1} \mathcal{K}\left(\frac{t_p^n - t_p^k}{d_K(t_p^n)}\right), \tag{2.3.4}$$

where $d_K = \max_k |t_p^k - t_p^n|$. The expectation of t_p^n is

$$\epsilon_p^n = \int_{-\infty}^{+\infty} t_p^n \hat{p}(t_p^n)\, dt_p^n. \tag{2.3.5}$$

Because $\hat{p}(t_p^n)$ is the average of K independent normal distributed random variables, the expectation of $\hat{p}(t_p^n)$ is the average of expectations of all the normal distributed variables, which are $t_p^k, k = n-K, ..., n-1$, so $\epsilon_p^n = \frac{1}{K}\sum_{k=n-K}^{n-1} t_p^k$.

If virtual player $q(w)$ believes that the manager is of T_2, she subtracts the expectation of t_p^n, $\forall p \in \mathcal{P}_w$ from the displayed costs. Therefore, in stage n, the perceived cost of path $p \in \mathcal{P}_w$ is $c_p^{r,n} = \mu_(T_1|b^n)c_p^{d,n} + \mu_(T_2|b^n)(c_p^{d,n} - \epsilon_p^n) = c_p^{d,n} - \mu_(T_2|b^n)\epsilon_p^n$.

We define $\beta_w^n = [\beta_p^n : p \in \mathcal{P}_w]$, where $\beta_p^n = \mu(T_2|b^n)\epsilon_p^n$, and $n = 0, 1, \ldots$. Then, we define the *signal vector* corresponding to history b^n:

$$\beta^n = [\beta_1^{nT}, \ldots, \beta_W^{nT}]^T \tag{2.3.6}$$

(if we need not specify the stage, we just use β). Therefore, we can conclude that the perceived cost vector at stage n is

$$c^{r,n} = c^{d,n} - \beta^n. \tag{2.3.7}$$

Notice that in the initial stages, the length of memory is less than K, so we should specify the estimation mechanism in this situation. We set $\epsilon_p^n = \frac{1}{K}\sum_{k=0}^{n-1} t_p^k$ in the initial stages $n < K$.

2.4 Stochastic Games

From Section 2.3, we can see that before playing each stage game, virtual players always receive a signal β (2.3.6), and the signal varies from stage to stage. In other words, stage games are no longer the same, and the players no longer simply play the identical stage game repeatedly. In the stage game, some events are deterministic and some are random. This kind of game is called "stochastic game". In a stochastic game, players repeatedly play games from a set of normal-form games defined in Definition 1.4.1. The game played at any stage depends on the previous game played and the action taken by all players in that game. Notice that the stochastic game is a generalization of the identical-stage repeated games discussed in previous sections.

The formal definition of a stochastic game is

Definition 2.4.1 *A stochastic game is a tuple* (Q, N, A, P, R)*, where*

1. *Q is a finite set of states.*

2. *N is a finite set of players.*

3. *A is the available action set.*

4. *$L : Q \times A \times Q \mapsto [0, 1]$ is the transition probability function. $L(q_1, a, q_2)$ is the probability of transitioning from sate q_1 to state q_2 after action a.*

5. *R is the payoff function.*

From Definition 2.4.1, we know that in a stochastic game the play proceeds by steps from position to position, according to transition probabilities controlled jointly by the two players[34]. Then, we can transform our repeated game to a stochastic game. What we should add to the existing model are the state set and transition probability function.

Because any difference in histories may lead to different state transitions, the *state set* is the history set $\mathcal{H}$. Notice that in our model, the transition rule from the current state to the subsequent state is deterministic, which is a special kind of stochastic game. So we can use a *successor function* $suc(b, t) : \mathcal{H} \times \mathcal{U} \mapsto \mathcal{H}$ to replace the transition probability function.

Then, we can have

Definition 2.4.2 *Our stochastic game model at stage $n, n = 0, 1, \ldots$ consists of*

1. *a set of states $\mathcal{H}$;*

2. *$W + 1$ players, $q(w), w \in \mathcal{W}$, and manager m;*

3. *the action sets for $q(w)$ $\mathcal{Y}_w$, and the action set for m $\mathcal{Q}$;*

4. *the transition probability function, which is the successor function $suc(b,t) = \{t^{n-K+1}, \ldots, t^{n-1}, t\}$, where $b = \{t^{n-K}, t^{n-K+1}, \ldots, t^{n-1}\}$;*

5. *the cost function for $q(w)$ $c_{q(w)} = M(h) + \sum_{p:p\in\mathcal{P}_w, h_p>0}(M(h) - c_p^d(h) - \beta_p^n)$($M(h)$ is defined in (2.1.1)), and the cost function for m $C(h) = \Sigma_{p\in\mathcal{P}} c_p(h) h_p$.*

At each stage, as the manager knows the actual cost of each path, she plays the same stage game, but for virtual players, they play a different selfish routing at each stage. The cost of selfish routing for the virtual players is $c^{d,n} - \beta^n, n = 0, 1, 2, \cdots$. We use $h^{NE}(c)$ to denote the *equilibrium flow* for the cost vector c. Therefore, receiving a displayed cost vector $c^{d,n}$, the best response for virtual players is

$$B(c^{d,n}) = h^{NE}(c^{d,n} - \beta^n), n = 0, 1, 2, \cdots .$$

Then, the Stackelberg cost for the manager is

$$C^{n*} = \min_{c^{d,n}(h)\in\mathcal{Q}} C(h^{NE}(c^{d,n} - \beta^n)).$$

And the Stackelberg action is

$$c^{d,n*} = arg \min_{c^{d,n}\in\mathcal{Q}} C(h^{NE}(c^{d,n} - \beta^n)).$$

Because $t^n = c^{d,n} - c^n$,

$$\min_{t^n\in\mathcal{U}} C(h^{NE}(c^n + t^n - \beta^n)) = \min_{c^{d,n}(h)\in\mathcal{Q}} C(h^{NE}(c^{d,n} - \beta^n)).$$

Therefore, the optimal toll vector for the manager in stage n is

$$t^{n*} = arg \min_{t^n\in\mathcal{U}} C(h^{NE}(c^n + t^n - \beta^n)).$$

We define $C^{NE} = C(h^{NE}(c))$ and $C^{SO} = C(h^{NE}(c+t^*))$. If $(t^* + \beta^n) \in \mathcal{U}$, the optimal flow can be maintained, so $t^{n*} = t^* + \beta^n$ and $C^{n*} = C^{SO}$.

Notice that the transition rule just changes the stage cost for the virtual players and only the manager's action influences the transition rule. The virtual players always use the best response to the manager's stage action. Then the normalized discounted cost of the manager is

$$\mathcal{C} = (1-\delta)\sum_{n=0}^{\infty} \delta^n C(B(c^{d,n} - \beta^n), c^{d,n}).$$

2.5 Existence of Perfect Equilibrium in Markov Strategies

We introduce a special kind of strategy, Markov strategy. The action for the manager induced by the Markov strategy only depends on the state, no matter in which stage she is playing. Markov strategy is defined as

Definition 2.5.1 *A strategy, $s_m^\diamond$, for the manager is called a Markov strategy if $s_m^\diamond(b,n) = s_m^\diamond(b,n')$, where $b \in \mathcal{H}$ and $n, n' = 0, 1, \ldots$.*

We want to know the behavior of the manager in equilibrium paths. The "simplest" strategy we can imagine is the Markov strategy. If we can prove the existence of perfect equilibrium when the manager commits to Markov strategies, we can have an understanding of the manager's behavior in equilibrium paths.

[16], [27], [29], and [30] proved the existence of Markov equilibrium strategies in stochastic games, but their model does not contain short-run players. However, we can prove the same result in stochastic games with short-run players and limited history.

Theorem 2.5.1 *The manager has a Markov perfect equilibrium strategy in the stochastic game defined in Definition 2.4.2.*

Proof:

We use $F = \{f : \mathcal{H} \mapsto \mathbb{R}\}$ to denote the set of all possible costs for the manager. Then, given $f \in F$ and $b \in \mathcal{H}$, we define a one-shot game $G_f(b)$, where the cost to the manager is given by

$$(1-\delta)C(B(c^d-\beta), c^d) + \delta f(suc(b)). \tag{2.5.8}$$

Notice that virtual players always give best response to manager's action, the current state is b and the corresponding signal is β.

In the one-shot game $G_f(b)$, f specifies the cost of the continuation game. For each $b \in \mathcal{H}$, Equation (2.5.8) has a Nash equilibrium point[30]. Let $z(b)$ be the corresponding Nash equilibrium cost. Then we can define a mapping set $N_f = \{b \mapsto z(b) : b \in \mathcal{H}\}$.

Lemma 3 in [30] proved that there exists a f^* such that $f^* \in N_{f^*}$, which means $f^*(b)$ is the equilibrium cost for each $b \in \mathcal{H}$ in the one-shot game $G_{f^*}(b)$. Then we can show that f^* is an equilibrium cost in the stochastic game and find the corresponding Markov strategy for the manager.

[30] proved the existence of action profile $\gamma^*(b, \mathbb{P})$ yielding payoff $\mathbb{P}$ at state b. Then we have that $\gamma^*(b, f^*(b))$ yields the Nash equilibrium cost $f^*(b)$ in $G_{f^*}(b)$, and it is easy to see that $\gamma^*(b, f^*(b))$ is a Markov strategy. Because $\gamma^*(b, f^*(b))$ is an equilibrium strategy for the one-shot game $G_{f^*}(\cdot)$ at any state, and the continuation cost is also induced by this Markov strategy, we can conclude that it is a perfect equilibrium strategy for the manager in this stochastic game. $\square$

Notice that in our proof, $\mathcal{H}$ only contains truncated histories.

2.6 The Upper Bound of Perfect Equilibrium Costs and the Optimal Strategy for the Manager

Our purpose in this section is to find out the upper bound of normalized discounted costs in subgame perfect equilibrium and the corresponding optimal strategy for the manager. If there is not any bound for toll vectors, the manager can always maintain the optimal flow by adding suitable (maybe quite large) toll vectors. However, with the bound, the manager's action is restricted, so we want to get the scope of possible costs the manager can achieve. There is a class of theorems which states that in repeated games, any outcome can be a feasible solution concept for equilibrium if certain conditions are satisfied. This class of theorems is called "folk theorems". In [14], Fudenberg and Maskin proved that any individually rational and feasible payoff can be maintained as an equilibrium outcome in repeated games with particular incomplete information. In [11], Fudenberg, Kreps, and Maskin proved a folk theorem for repeated games with short-run players. In [9], Dutta proved the folk theorem for general stochastic games with long-run players. He proved that under some weak conditions, any feasible and individually rational payoff can almost be a subgame perfect equilibrium payoff if the discount factor is sufficiently near to 1. In [17], Hörner et al. extended this result to a stochastic game with short-run players and presented a recursive algorithm to calculate the set of perfect equilibrium payoffs as the discount factor tends to 1. To get a similar result in our model, we give several relevant definitions.

If the initial state (history) is b, the total cost for the manager is

$$\mathcal{C}(b) = (1-\delta)\sum_{n=0}^{\infty}\delta^n C^n(b,s),$$

where $s = [B(s_m, b^n), s_m], s_m \in S_m$, meaning that virtual players' actions in s are always the static best responses to s_m in each stage. The maxmin cost for the manager is

$$\overline{\mathcal{C}} = \max_{s_w: w\in\mathcal{W}} \min_{s_m}(1-\delta)\sum_{n=0}^{\infty}\delta^n C(s).$$

If the manager adds no tolls in any state, then $C(s) = C^{NE}$ (the stage cost for the manager with equilibrium flow), so $\overline{\mathcal{C}} = C^{NE}$. This gives an upper bound of the manager's cost. What we are interested in is the minimal cost the manager can obtain during repeated games.

The Markov strategy set of the manager is denoted by $S_m^{\diamond}$, the set of total costs for the manager generated by Markov strategies is

$$\mathcal{F} = \{\mathcal{C}(s), \exists s = [s_1, ..., s_W, s_m], s.t. s_m \in S_m^{\diamond}\}.$$

Let $\mathcal{F}^{\dagger}$ be the set of feasible total costs for the manager. Lemma 1 in [9] says that all feasible payoffs can be realized by one-shot public randomization over pure Markov strategies. Therefore, we obtain that

$$\mathcal{F}^{\dagger} = co\mathcal{F},$$

meaning that the set of feasible total costs is the convex hull of $\mathcal{F}$ (the convex hull of $\mathcal{F}$ is the smallest convex set containing $\mathcal{F}$).

Then we can define the set of feasible and individually rational costs for the manager,

$$\mathcal{F}^* = \{\mathcal{C} \in \mathcal{F}^{\dagger} : \mathcal{C} \leq \overline{\mathcal{C}}\}.$$

Theorem 2 in [17] says that if players have the full history, any element in $\mathcal{F}^*$ can be reached by the perfect equilibrium cost as $\delta \to 1$. However, in our model, limited history makes some parts in $\mathcal{F}^*$ unreachable. Therefore, we cannot prove a similar folk theorem, but we can prove the existence of a cost of the manager smaller than C^{NE} in a network of parallel links.

Assumption 2 In the network discussed in Section 2.6, each path contains only one edge, meaning that $E = P$ and Θ is a diagonal matrix

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix},$$

where

$$\Theta_{i,j} = \begin{cases} 1 & if \quad i = j \\ 0 & if \quad i \neq j \end{cases}, and\ 0 \leq i, j \leq E.$$

Definition 2.6.1 *Suppose that $\mathcal{A} \subseteq \mathcal{U}$. $\forall t_1, t_2 \in \mathcal{A}$ and $t_1 \leq t_2$, if $C(t_1) \leq C(t_2)$, $C(t)$ is a weakly increasing function in $\mathcal{A}$; if $C(t_1) \geq C(t_2)$, $C(t)$ is a weakly increasing function in $\mathcal{A}$.*

Lemma 2.6.1 and Lemma 2.6.2 specify relationships between the manager's cost and her action.

Lemma 2.6.1 *If $t^* + \beta \leq ub$, $C(t)$ is a weakly decreasing function in $[0, t^* + \beta]$ and a weakly increasing function in $(t^* + \beta, ub]$; if $t^* + \beta > ub$, $C(t)$ is a weakly decreasing function in $[0, ub]$.*

Proof: Let $t_1 = \theta_1 t^*, t_2 = \theta_2 t^*$, where $0 \leq \theta_1 \leq \theta_2 \leq q$ and $t_1, t_2 \leq t^* + \beta$. Therefore, $t_1 - \beta \leq t_2 - \beta \leq t^*$. Because of Assumption 2, we have $C(h^{NE}(c + t_1 - \beta)) \geq C(h^{NE}(c + t_2 - \beta)) \geq C^{SO}$. Similarly, let $t_1 = \theta_3 t^*, t_2 = \theta_4 t^*$, where

$0 \leq \theta_3 \leq \theta_4 \leq q$ and $t_3, t_4 > t^* + \beta$. Therefore, $t^* \leq t_3 - \beta \leq t_4 - \beta$. Then we have $C^{SO} \leq C(h^{NE}(c + t_3 - \beta)) \leq C(h^{NE}(c + t_4 - \beta).\square$

Figure 2.2 and Figure 2.3 are general curves illustrating the properties of tolls proved in Lemma 2.6.1. From Lemma 2.6.1, we can directly have

Lemma 2.6.2 *In every subgame perfect equilibrium path,*

$$\mathcal{C}(suc(b, t_1)) \leq \mathcal{C}(suc(b, t_2))$$

for any $t_1 \leq t_2$.

Lemma 2.6.2 says that using a smaller toll vector at the current stage must give an equal or better continuation payoff.

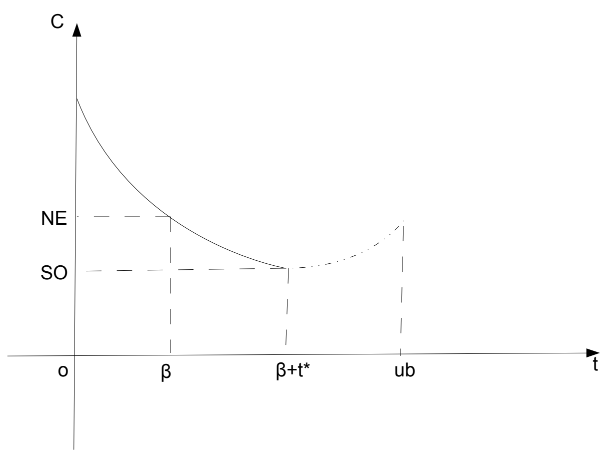

Figure 2.2: The Curve of C when $t^* + \beta \leq ub$

Notice that when $t^* + \beta \leq ub$, the manager will never use tolls in interval $(t^* + \beta, ub]$. To demonstrate this, we have Lemma 2.6.3:

Lemma 2.6.3 *In the stage game with signal* $\beta \leq ub - t^*$ *and the corresponding state(history)* b*, the action* $t^* + \beta$ *strictly dominates every action* $t \in (t^* + \beta, ub]$.

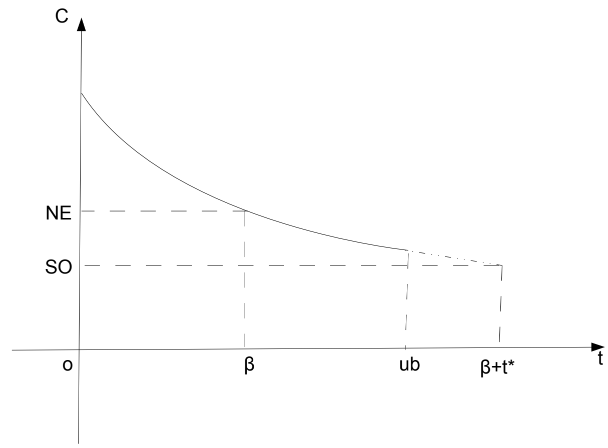

Figure 2.3: The Curve of C when $t^* + \beta > ub$

Proof:

For every action $t \in (t^*+\beta, ub]$, the cost for the manager is $\mathcal{C}_1 = (1-\delta)C(B(c+t-\beta), t)+\delta\mathcal{C}(suc(b,t))$; for $t^*+\beta$, the cost for the manager is $\mathcal{C}_2 = (1-\delta)C(B(c+t^*), t^*+\beta)+\delta\mathcal{C}(suc(b,t^*+\beta))$. Because $t > t^*+\beta$ and Lemma 2.6.1, we have $C(B(c+t-\beta),t) > C(B(c+t^*), t^*+\beta)$ and $\mathcal{C}(suc(b,t)) \geq \mathcal{C}(suc(b, t^*+\beta))$, so $\mathcal{C}_1 > \mathcal{C}_2$. $\square$

Therefore, if $t^*+\beta \leq ub$, the manager only uses toll vectors in $[0, t^*+\beta]$. We have

Definition 2.6.2 *The rational space for the manager is* $[0, t^*+\beta]$ *if* $t^*+\beta \leq ub$, *or* $[0, ub]$ *otherwise.*

We can conclude that given any signal β, the cost function for the manager is always a decreasing function of t. This result shows the myopic incentive of the manager: using $t^*+\beta$ (if possible) or ub is a strict dominant action in a single repetition.

Now we show why reputation is valuable in each stage. Suppose at a given stage, the manager has the best reputation ($\mu(T_1) = 1$), then $\beta = \mu(T_2) \cdot \epsilon = 0 \cdot \epsilon = 0$, so the optimal action is t^* and the corresponding cost is $C(B(c + t^*), t^*) = C^{SO}$. In contrast, if the manager has the worst reputation ($\mu(T_1) = 0$), then β meets its maximum ϵ(defined in (2.3.5)), the optimal action is $t^* + \epsilon$ (if $t^* + \epsilon \leq ub$) or ub (if $t^* + \epsilon > ub$), and the corresponding cost is $C(B(c + t^*), t^* + \epsilon)$ or $C(B(c + ub - \epsilon))$. Therefore, the stage cost for the manager with better reputation is less or equal than with worse reputation. From Lemma 2.6.2, the manager with better reputation has less continuation cost because $\mathcal{C}(suc(b, t^*)) < \mathcal{C}(suc(b, t^* + \epsilon))(\mathcal{C}(suc(b, ub)))$.

In most repeated games, such as games discussed in [11], [12], [13], [14], [22], and [24], the perfect equilibrium payoff attained by long-run players is always equal to or higher than the equilibrium payoff attained in a single stage game because she can build a good reputation for some "friendly" action. Then the short-run players will give the best response to the "friendly" action, which is better than the static equilibrium action. However, our game is different, the manager can do better in a single repetition because she can have the optimum in stage game, but if she commits to any action in the repeated game, she can only reach the equilibrium flow cost. Nevertheless, the manager can still get a profit by building a good reputation as we showed above. In addition, virtual players get not only reputation but also signal β(defined in (2.3.6)) from history, which is another difference between our model and the models discussed by others before.

A *clean history* is given by b_0 and a *clean phase* denotes the K stages where no tolls are used. Similarly, an *exploit phase* denotes the stages where tolls are used. The behavior of using no tolls for K stages is denoted by *clean the history.*

Definition 2.6.3 *A strategy s is a cycle strategy if the manager plays as following repeatedly : using tolls induced by s for $L(s)$ stages and does not use tolls for K stages.*

Notice that a cycle strategy is a Markov strategy. To make the cycle strategy useful, we add another assumption to the reputation mechanism,

Assumption 3 C increases faster when the signal is larger, which is

$$|C(t_1, \beta_1) - C(t_2, \beta_1)| > |C(t_1, \beta_2) - C(t_2, \beta_2)|,$$

where $\beta_1 > \beta_2$ and t_1, t_2 are in the rational space.

We define a cycle strategy s^*(Figure 2.6), with which the manager keeps using $t^* + \beta$ for $L(s^*)$ stages to maintain the optimal flow as long as possible, then keep using no tolls for K stages. Among all the cycle strategies hitting ub during the exploit phase, s^* has the shortest exploit phase. We should set that the shortest exploit phase is still longer than K, in other words, $L(s^*) > K$. We assume that when the manager uses s^*, the signals during the exploit phase are $\beta^{*0}, \beta^{*1}, \ldots, \beta^{*L(s^*)}$. $\forall 0 \leq n \leq L(s^*)$, we have

$$\beta^{*n} = \frac{1}{K}(Kt^* + \sum_{i=n-K}^{n-1} \beta^{*i}) = t^* + \frac{1}{K} \sum_{i=n-K}^{n-1} \beta^{*i} < t^* + \beta^{*n-1}.$$

Therefore, $\beta^{*n} - \beta^{*n-1} < t^*$, and $\beta^{*L(s^*)} < L(s^*)t^*$. Because $\beta^{*L(s^*)} > ub$, $q = \frac{ub}{t^*} < L(s^*)$. Therefore, we have

Assumption 4 $K < q$.

Notice that after using tolls for some stages, the manager can also use very small toll vectors, such as $\omega\ t^*$, to increase $\mu(T_1)$. However, we can prove that in any subgame perfect equilibrium, the manager must use **0** other than $\omega\ t^*$ increase $\mu(T_1)$.

Lemma 2.6.4 *If the manager wants to increase $\mu(T_1)$, she must clean the history.*

Proof: If the manager uses **0** to increase $\mu(T_1)$, then

$$g_1(b_0) = 1 \text{ and } g_2(b_0) = 0.$$

Therefore,

$$\mu(T_1|b_0) = 1 \text{ and } \mu(T_2|b_0) = 0.$$

The signal corresponding to b_0 is $\beta_0 = 0$.

If the manager does not use **0** to increase $\mu(T_1)$, the smallest toll vector she can choose is ωt^*. We define $b_1 = \{\underbrace{\omega\ t^*, \ldots, \omega\ t^*}_{K\text{"}\omega\ t^*\text{"}s}\}$. Then we have

$$g_1(b_1) = \frac{1}{K\mathcal{K}(q-\omega)} \text{ and } g_2(b_1) = (K-1)\mathcal{K}(0) = K-1.$$

Therefore,

$$\mu(T_1|b_1) = \frac{\frac{1}{K\mathcal{K}(q-\omega)}}{\frac{1}{K\mathcal{K}(q-\omega)} + K - 1} = \frac{1}{1 + (K^2-K)\mathcal{K}(q-\omega)},$$

and

$$\mu(T_2|b_1) = \frac{(K^2-K)\mathcal{K}(q-\omega)}{1 + (K^2-K)\mathcal{K}(q-\omega)}.$$

And the corresponding signal is

$$\beta_1 = \mu(T_2|b_1)\omega\ t^* = \frac{(K^2-K)\mathcal{K}(q-\omega)\omega\ t^*}{1 + (K^2-K)\mathcal{K}(q-\omega)}.$$

Because $K < q$, b_0 and b_1 only affect the following K stages. Suppose that in any stage of the following K stages, the manager uses toll vector t. For any β in the clean phase, we have $C(h^{NE}(c+t-\beta_1)) - C(h^{NE}(c+t-\beta_0)) > C(h^{NE}(c+0-\beta)) - C(h^{NE}(c+\omega\ t^*-\beta))$. $\square$

Lemma 2.6.5 *If the manager wants to increase $\mu(T_1)$, she must use the cycle strategy.*

Proof:

We assume that in stage 0, the manager uses toll vector t_1 and the displayed cost vector is c_1^d. Therefore, $\forall t_2 \in \mathcal{U} - \{t_1\}$ (the corresponding displayed cost vector is c_2^d), we have

$$(1-\delta)C(B(c_1^d), c_1^d) + \delta\mathcal{C}(suc(b_0, t_1)) > (1-\delta)C(B(c_2^d), c_2^d) + \delta\mathcal{C}(suc(b_0, t_2)). \tag{2.6.9}$$

Lemma 2.6.4 implies that if the manager decides to rebuild her reputation(increase $\mu(T_1)$) after some stages of using tolls, she must clean the history. Therefore, after the clean phase, the history is b_0. (2.6.9) implies that the manager should use t_1 when have a history b_0. Similarly, the manager should repeat each action played before.$\square$

However, there is still one case where the manager should not use the cycle strategy. If the manager uses strategy $s^\triangle$ (Figure 2.4), which is using some tolls in the initial stages and keeping using ub after hitting ub, she reaches the cost

$$\mathcal{C}(s^\triangle) = (1-\delta)(\sum_{i=0}^{L(s^\triangle)-1} \delta^i C^i(s^\triangle) + \sum_{i=L(s^\triangle)}^{\infty} \delta^i C^{NE}) = (1-\delta^{L(s^\triangle)})C'(s^\triangle) + \delta^{L(s^\triangle)} C^{NE},$$

where $C^{NE} < C^i(s^\triangle) \leq C^{SO}$ and $C'(s^\triangle) = (1-\delta)\sum_{i=0}^{K} \delta^i C^i(s^\triangle)$. When the manager is patient enough ($\delta \to 1$), $\mathcal{C}((s^\triangle)) \to C^{NE}$.

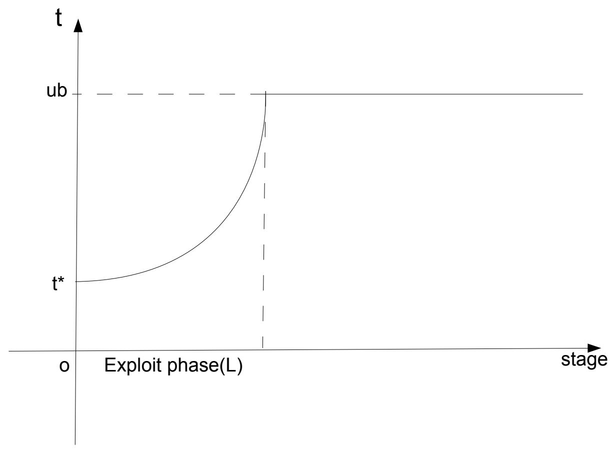

Figure 2.4: Strategy $s^{\triangle}$

If the manager uses the cycle strategy (Figure 2.5), the cost for the manager in the first cycle is

$$\mathcal{C}^{cycle}(s) = (1-\delta)(\sum_{i=0}^{L(s)-1} \delta^i C^{SO} + \sum_{i=L(s)}^{L(s)+K-1} \delta^i C^i(s)) = (1-\delta^{L(s)})C^{SO} + \delta^{L(s)}(1-\delta^K)C'(s).$$

Because the manager has the same cost in all cycles, we have

$$\mathcal{C}(s) = \frac{(1-\delta^{L(s)})C^{SO} + \delta^{L(s)}(1-\delta^K)C'(s)}{1-\delta^{L(s)+K}}.$$

If

$$C'(s) < \frac{1}{\delta^{L(s)} - \delta^{L(s)+K}} C^{NE} - \frac{1-\delta^{L(s)}}{\delta^{L(s)} - \delta^{(L(s)+K)}} C^{SO},$$

then $\mathcal{C}(s) < C^{NE}$, and the best strategy is s.

Therefore, we can conclude that if

$$C'(s) > \frac{1}{\delta^{L(s)} - \delta^{L(s)+K}} C^{NE} - \frac{1-\delta^{L(s)}}{\delta^{L(s)} - \delta^{(L(s)+K)}} C^{SO},$$

the manager should not use the cycle strategy. The optimal strategy for her is $s^{\triangle}$, this case is called **Case 1**.

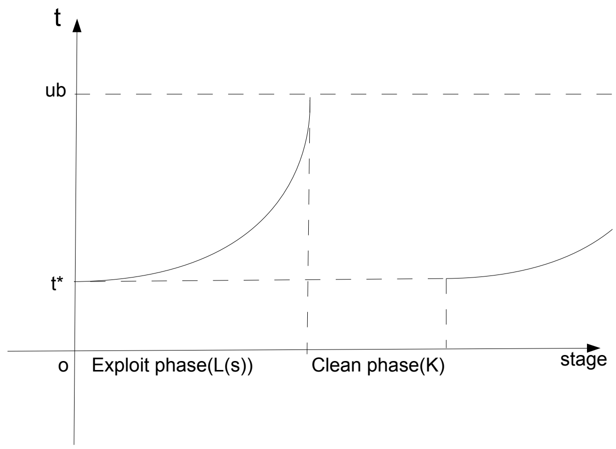

Figure 2.5: A General Cycle Strategy s

Stage	Toll	Cost for Manager
0 to $L(s_1)-1$	$t^*+\beta$	C^{SO}
$L(s_1)$ to $L(s_1)+J-1$	ub	C^{NE}
$L(s_1)+J$ to $L(s_1)+J+K-1$	0	C^{NE-}

Table 2.1: Strategy s_1 and the Corresponding Cost for the Manager

In **Case 2**, we have

$$C'(s) < \frac{1}{\delta^{L(s)} - \delta^{L(s)+K}} C^{NE} - \frac{1-\delta^{L(s)}}{\delta^{L(s)} - \delta^{(L(s)+K)}} C^{SO},$$

under which the manager should use the cycle strategy. We want to figure out her optimal behavior in the exploit phase. We know that $t^*+\beta$ increases if the manager wants to maintain the optimal flow, and we wonder whether the manager should keep using ub for a period after hitting ub. So we define a cycle strategy s_1, such that, in a cycle, the manager uses $t^*+\beta$ from the stage when the history is clean until hitting ub and uses ub for J stages ($J>0$), then cleans history. s_1 is showed in Table 2.7 and Figure 2.7.

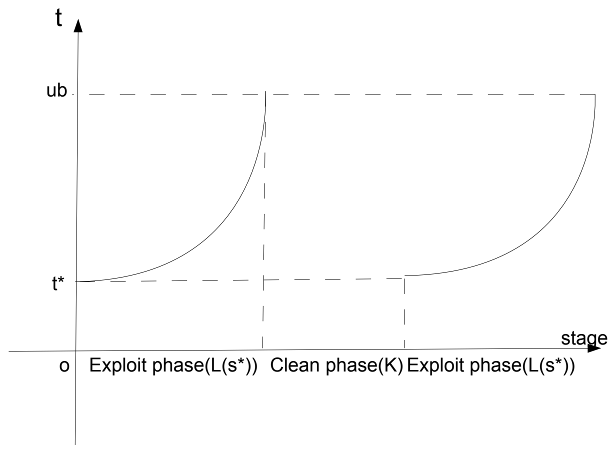

Figure 2.6: Strategy s^*

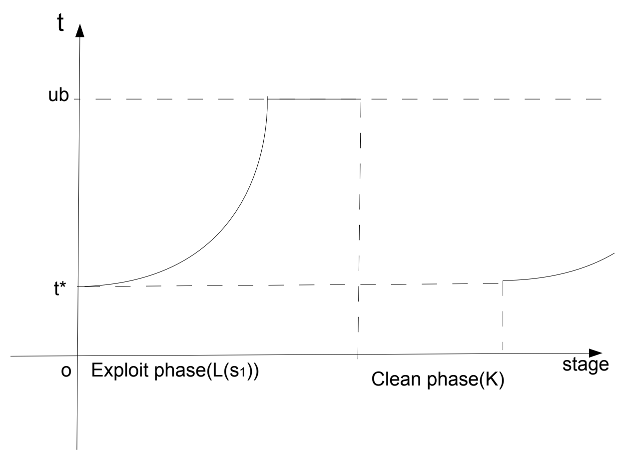

Figure 2.7: Strategy s_1

Lemma 2.6.6 *In Case 2, s^* strictly dominates s_1.*

Proof:

$$\begin{aligned}\mathcal{C}^{cycle}(s_1) &= (1-\delta)(\sum_{i=0}^{L(s_1)-1}\delta^i C^{SO} + \sum_{i=L(s)}^{L(s)+J-1}\delta^i C^{NE} + \sum_{i=L(s)+J}^{L(s)+J+K-1} C^i(s_1)) \\ &= (1-\delta^{L(s_1)})C^{SO} + \delta^{L(s_1)}(1-\delta^J)C^{NE} + \delta^{L(s_1)+J}(1-\delta^K)C'(s_1).\end{aligned}$$

Then,

$$\mathcal{C}(s_1) = \frac{(1-\delta^{L(s_1)})C^{SO} + \delta^{L(s_1)}(1-\delta^J)C^{NE} + \delta^{L(s_1)+J}(1-\delta^K)C'(s_1)}{1-\delta^{L(s)+J+K}}.$$

Because before the clean phase the tolls used by s_1, ub, are greater than the tolls used by s^*, we have $C'(s_1) > C(s^*)$. In addition, $L(s_1) = L(s^*)$ and

$$C'(s_1) < \frac{1}{\delta^{L(s_1)} - \delta^{L(s_1)+K}}C^{NE} - \frac{1-\delta^{L(s_1)}}{\delta^{L(s_1)} - \delta^{(L(s_1)+K)}}C^{SO},$$

therefore

$$\frac{\mathcal{C}(s_1)}{\mathcal{C}(s^*)} > 1.\square$$

The stage cost for the manager in s^* and s_1 are illustrated in Figure 2.8 and Figure 2.9, respectively. Lemma 2.6.6 says that if keeping using ub after hitting ub is not good for the manager, then maintaining ub for any length of stages is not good for the manager. Therefore, after hitting ub, the manager should begin the clean phase immediately.

Then, we consider whether the manager should begin the clean phase before hitting ub. So we define a cycle strategy s_2, such that, in a cycle, the manager uses $t^* + \beta$ from the stage when the history is clean for L stages ($L < L(s_1) = L(s^*)$), then begins the clean phase. s_2 is showed in Table 2.2 and Figure 2.10.

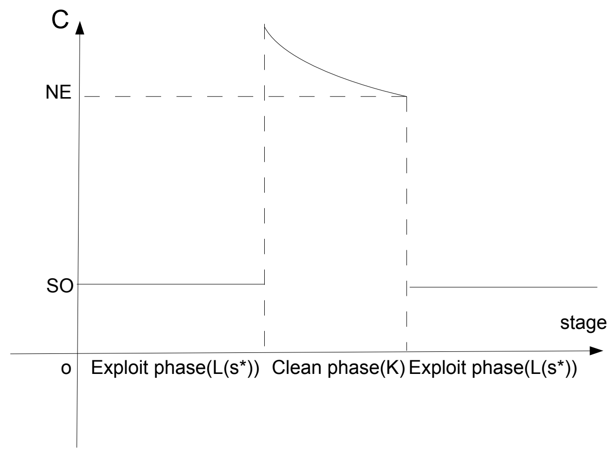

Figure 2.8: Stage Cost Curve of Strategy s^*

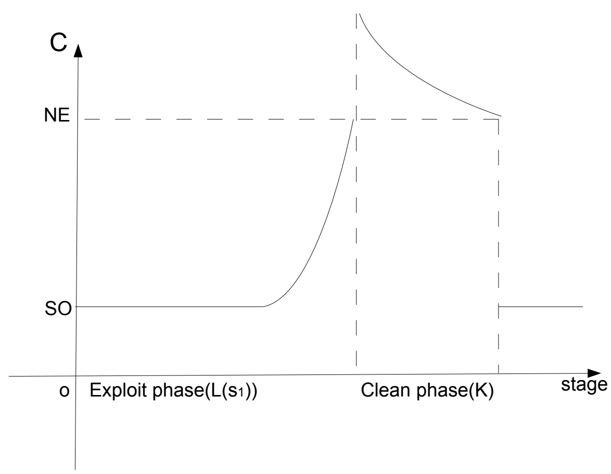

Figure 2.9: Stage Cost Curve of Strategy s_1

Stage	Toll	Cost for Manager
0 to $L-1$	$t^* + \beta$	C^{SO}
L to $L+K-1$	0	C^{NE-}

Table 2.2: Strategy s_2 and the Corresponding Cost for the Manager

Lemma 2.6.7 *In Case 2, s^* strictly dominates s_2.*

Proof:

$$\begin{aligned}\mathcal{C}^{cycle}(s_2) &= (1-\delta)(\sum_{i=0}^{L-1}\delta^i C^{SO} + \sum_{i=L}^{L+K-1}\delta^i C^i(s_2)) \\ &= (1-\delta^L)C^{SO} + \delta^L(1-\delta^K)C'(s_2),\end{aligned}$$

then,

$$\mathcal{C}(s_2) = \frac{(1-\delta^L)C^{SO} + \delta^L(1-\delta^K)C'(s_2)}{1-\delta^{L+K}}.$$

Because $L < L(s^*)$,

$$\frac{1-\delta^L}{1-\delta^{L+K}} < \frac{1-\delta^L(s^*)}{1-\delta^{L(s^*)+K}}.$$

In addition,

$$C'(s_2) < \frac{1}{\delta^{L(s_2)} - \delta^{L(s_2)+K}}C^{NE} - \frac{1-\delta^{L(s_2)}}{\delta^L(s_2) - \delta^{(L(s_2)+K)}}C^{SO},$$

and because of Assumption 3, we have $C'(s_2) - C^{SO} > C'(s^*) - C'(s_2)$. Therefore

$$\frac{\mathcal{C}(s_2)}{\mathcal{C}(s^*)} > 1.\square$$

Lemma 2.6.7 says that if keeping using ub after hitting ub is not good for the manager, the manager should maintain the optimal flow as long as possible, the stage cost of s_2 is illustrated in Figure 2.11.

We define a cycle strategy s_3 such that, in s_3, the manager deviates from keeping using $t^* + \beta$ in some stage before the $L(s_3)$-th stage, but still increases tolls until hitting ub, then begins the clean phase. Therefore, the cost of s_3 is

$$\mathcal{C}(s_3) = \frac{(1-\delta^{L(s_3)})C^{SO} + \delta^{L(s_3)}(1-\delta^K)C'(s_3)}{1-\delta^{L(s)+K}},$$

where $L(s_3) > L(s^*)$ and $C'(s_3) < C'(s^*)$. Just as in Lemma 2.6.7, we have

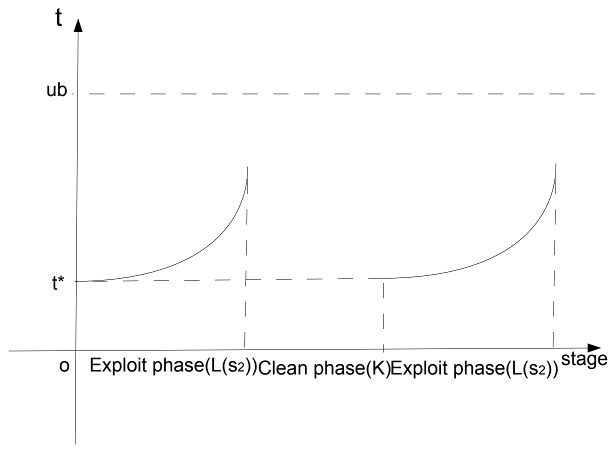

Figure 2.10: Strategy s_2

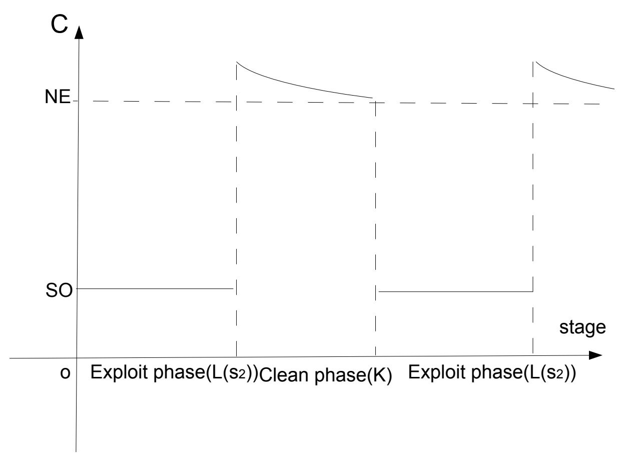

Figure 2.11: Stage Cost Curve of Strategy s_2

Lemma 2.6.8 *In Case 2, s^* strictly dominates s_3.*

s_1, s_2 and s_3 cover all possible cycle strategies deviating from s^*. Then, from Lemma 2.6.6, Lemma 2.6.7 and Lemma 2.6.8, we can conclude the following theorem,

Theorem 2.6.9

1. *in Case 1, the optimal strategy for the manager is $s^{\triangle}$ and the upper bound of cost is C^{NE};*

2. *in Case 2, the optimal strategy for the manager is s^* and the upper bound of cost is $\mathcal{C}(s^*)$.*

2.7 An Example for the Optimal Strategy of the Manager

Now we use an example containing a simple network with one commodity and linear cost functions (Figure 2.12) to explain the result in Section 2.6. We assume that the demand of commodity is $r = 1$, so the optimal toll for stage game is $t^* = [0, 0.5]^T$ and the optimal flow is $h^* = [0.5, 0.5]^T$. In the repeated game, we make following assumptions:

1. number of stages in history is $K = 1$;

2. the upper bound of the toll factor is $q = 4$, so $ub = [0, 2]^T$;

3. if any toll is realized in history, μ_2 becomes 1.

Then, we have the optimal strategy and the flow induced by it in Table 2.3, and we have following observations:

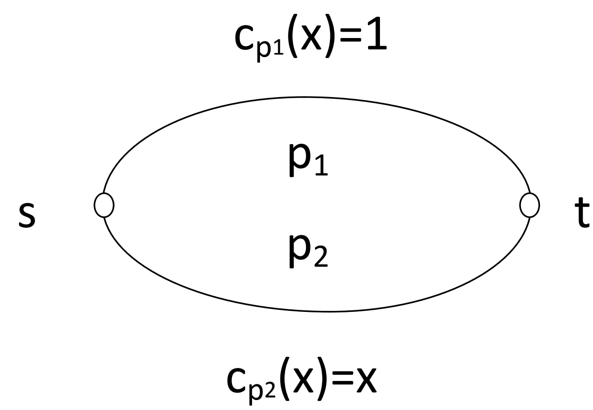

Figure 2.12: The Network in Example

Stage	Toll	Flow	μ_2	β	Perceived Cost
0	$[0, 0.5]^T$	$[0.5, 0.5]^T$	0	$[0, 0]^T$	$[1, 1]^T$
1	$[0, 1]^T$	$[0.5, 0.5]^T$	1	$[0, 0.5]^T$	$[1, 1]^T$
2	$[0, 1.5]^T$	$[0.5, 0.5]^T$	1	$[0, 1]^T$	$[1, 1]^T$
3	$[0, 2]^T$	$[0.5, 0.5]^T$	1	$[0, 1.5]^T$	$[1, 1]^T$
4	$[0, 0]^T$	$[0, 1]^T$	1	$[0, 2]^T$	$[1, 0]^T$
5	$[0, 0.5]^T$	$[0.5, 0.5]^T$	0	$[0, 0]^T$	$[1, 1]^T$

Table 2.3: Optimal Flow Pattern with $K = 1$

1. At stage 0, virtual player completely trusts the manager and give 0 to μ_2, then the manager gets the optimal flow.

2. At stage 1, μ_2 becomes 1, which means the virtual player does not believe what manager shows at all, however, the manager can still maintain the optimal flow from stage 1 to stage 3.

3. At stage 4, $[0, 2.5]^T$ exceeds the upper bound of $\mathcal{U}$, so the manager cannot use it to maintain the optimal flow. Then she decides to tell the truth to the virtual player, making her get a "bad" flow $[0, 1]^T$. But by doing

this, the manager rebuilds her reputation for the next stage.

4. At stage 5, the situation is identical with stage 0. In this round of loop, the manager can get the optimal flow all the time except one stage.

As action of each stage only depends on states(history of virtual players), this is a Markov strategy for manager. We can prove that it is the optimal strategy for the manager.

Lemma 2.7.1 *The strategy described in Table 2.3 is a subgame perfect equilibrium(Definition 2.2.2 on Page 27).*

Proof:
At each stage, if the manager uses a toll on P_2 smaller than the toll specified in Table 2.3, the cost of P_2 will always be greater or equal than 1; if the manager uses a toll on P_2 greater than the toll specified in Table 2.3, the cost of P_2 will always be less than 1. In other words, any one-shot deviation from the strategy described in Table 2.3 will induce a flow $[0, 1]^T$ or $[1, 0]^T$, which means there are no profitable one-shot deviation. Proposition 2.2.1 in [24] says that a strategy profile is subgame perfect if and only if there are no profitable one-short deviations. Then we can conclude that this is a subgame perfect equilibrium strategy.$\square$

Chapter 3

Stochastic User Equilibrium

3.1 Repeated Stochastic Selfish Routing and Stochastic User Equilibrium

In the model described in Chapter 2, the costs of paths and edges are deterministic, meaning that although the displayed cost may be different from the actual cost, it can be observed or measured directly. However, this may not be the case in some application and furthermore, many of the attributes that influence the users' costs cannot be observed and must therefore be treated as random. Consequently, the costs are modeled as random, meaning that choice models can give only the probability with which alternatives are chosen, not the choice itself[35]. Because the real psychological process is quite complicated, it is reasonable to treat the perceived costs as random variables. From the second stage, users get some information from previous play, and different users may get different information because of different experiences. Therefore, although receiving the same displayed costs, the perceived costs for different users are different. To incorporate the effect of this attribute, we do not use the estimation mechanism described in Section 2.3 to get a number vector from history.

At stage n, we express the perceived cost as a random variable consisting of a deterministic component $c^{d,n}$, and an additive random variable vector: the "error term" $\xi^n = [\xi_1^n, ..., \xi_W^n]^T$. So the perceived path cost vector at stage n is $c^{r,n} = c^{d,n}(h^n) + \xi^n = [c_1^{r,n}(h^n), ..., c_W^{r,n}(h^n)]^T$.

The distribution of the perceived cost is a function of the flow h. Therefore, the probability that path p will be chosen, Pr_p, can be related to the path flow vector h. The function relating Pr_p to h is known as the choice function. The probability that path p is chosen by a given flow h is the fraction of individuals in a large population who choose path p. At stage n, the choice probability is the probability that $c_p(h)$ is lower than the cost of any other paths for the same O-D pairs, when flow h is given, which is

$$Pr_p(h^n) = Pr[c_p^{r,n}(h^n) < c_{p^*}^{r,n}(h^n), p^* \in \mathcal{P}_w - \{p\}], p \in \mathcal{P}_w, w \in \mathcal{W},$$

and $Pr_p(h)$ satisfies,

1. $0 < Pr_p(h) \leq 1, p \in \mathcal{P}$;

2. $\Sigma_{p \in \mathcal{P}_w} Pr_p(h) = 1, w \in \mathcal{W}$.

The first property means that the probability for each path to be chosen is positive, this is the *positive assumption* on choice probabilities; the second property means that the summation of the probabilities for all the paths in every commodity is 1. At equilibrium, the flow satisfies $h_p^n = r_w Pr_p(h^n), p \in \mathcal{P}_w, w \in \mathcal{W}$. We set $Pr_w = [Pr_p, p \in \mathcal{P}_w]^T, w \in \mathcal{W}$.

Definition 3.1.1 *A feasible path flow vector h^n satisfying $h_w^n = r_w Pr_w(h^n), w \in \mathcal{W}$ is called stochastic user equilibrium.*

Once the distribution of the error term ξ is specified, the distribution of the cost can be determined, and the choice function can be calculated explicitly.

3.2 Existence of Optimal Tolls in Stochastic Selfish Routing

From (2.3.7) in Section 2.3, we know that the expectation of ξ^n is $-\beta^n$. Let $\delta^n = -\beta^n$. The expectation vector of $c^{r,n}$ is $c^{d,n}+\delta^n$. We assume that the choice probabilities for the paths are given by the logit model defined in (3.2.1),

$$Pr_p(h^n) = \frac{exp(-c_p^{d,n} - \delta_p^n)}{\Sigma_{p^* \in \mathcal{P}_w} exp(-c_{p^*}^{d,n} - \delta_{p^*}^n)}, p \in \mathcal{P}_w, w \in \mathcal{W}. \tag{3.2.1}$$

If we can prove the existence of a suitable toll which can lead the optimal flow to be a stochastic user equilibrium flow, we can conclude that the manager can maintain the optimal flow status in this stochastic user selfish routing model. We call the suitable toll “optimal toll”.

Our discussion applies to each stage in this repeated game. For the latter part in this chapter, we omit the superscript n for convenience. Proving the existence of the optimal toll is equivalent to proving the existence of the solution t in (3.2.2).

$$\widehat{h}_w = r_w Pr_w(\widehat{h} + t), w \in \mathcal{W}. \tag{3.2.2}$$

To express (3.2.2) in edge form, we have

$$\widehat{f} = \Sigma_{w \in W} r_w \Theta_w Pr_w[\Theta_w^T(c^{edge}(\widehat{f}) + t^{edge})], \tag{3.2.3}$$

where $\widehat{f} = \Theta\widehat{h}$.

We define $S_w, w \in \mathcal{W}$ as the satisfaction function, which is the expectation of the minimum cost,

$$S_w = E(\min_{p \in \mathcal{P}_w}(c_p^r)), w \in \mathcal{W}.$$

For our logit model, from [35], we have

$$\begin{aligned} S_w &= ln\Sigma_{p\in\mathcal{P}_w} exp(c_p^d + \delta_p) = ln\Sigma_{p\in\mathcal{P}_w} exp(c_p + \delta_p + t_p) \\ &= ln\Sigma_{p\in\mathcal{P}_w} exp(c_p + \delta_p + \Theta_{,p}^T t^{edge}), \end{aligned}$$

($\Theta_{,p}$ is the row vector denoting path p).

If $t_p = t_{\widetilde{p}}, \forall p, \widetilde{p} \in \mathcal{P}_w$, the path choice probability of commodity w does not change. Similarly, if t has already been given, adding or subtracting the same increment to each $p \in \mathcal{P}_w$ does not change the path choice probability[19]. We introduce the *uniform unit-vectors* $1_w = (1, \ldots, 1)^T \in \mathbb{R}^{P_w}$ for all $w \in \mathcal{W}$, and vector $q = [q_w : w \in \mathcal{W}] \in \mathbb{R}^W$, which satisfies

$$\Theta_w^T t^{edge} = q_w 1_w. \tag{3.2.4}$$

(3.2.4) means that t^{edge} gives the same path toll for each commodity. Then we have

Definition 3.2.1 *A neutral toll space is*

$$\Omega = \{t^{edge} \in \mathbb{R}^E : (3.2.4) \textit{ holds for some } \alpha \in \mathbb{R}^W\}.$$

In order to prove the existence of optimal tolls, we give the definition of the direction of recession [direction of constancy], which comes from [19]:

Definition 3.2.2 *For any convex function, $g : \mathbb{R}^E \mapsto \mathbb{R}$, a vector $d \in \mathbb{R}^E$ is designated as a direction of recession [direction of constancy] for g iff $(g(x + \lambda d) \leq g(x))[g(x + \lambda d) = g(x)]$ for all $x \in \mathbb{R}^E$ and scalars $\lambda > 0$.*

Now we prove the existence of the optimal tolls:

Theorem 3.2.1 *For the optimal flow $\widehat{h}$, there exists a solution to* (3.2.3)*, iff $\widehat{h} \in \mathbb{R}^P_{++}$, which means $\widehat{h}$ is a fully positive vector.*

Proof:

The positivity assumption on choice probabilities implies that solution to (3.2.3) can only exist for fully positive path flow vectors. So it suffices to establish the converse. Adapting Proposition 3.6 in [19] to our model, we can conclude that there exists a optimal toll for $\widehat{f}$ iff the function

$D(t^{edge}) = -\Sigma_{w \in \mathcal{W}} r_w ln \Sigma_{p \in \mathcal{P}_w} exp(c_p + \delta_p + \Theta^T_{,p} t^{edge}) + \widehat{f}^T t^{edge}$ achieves a minimum.

From lemma 3.7 in [19] we know that a convex function $g : \mathbb{R}^E \mapsto \mathbb{R}$, achieves its minimum if the only directions of recession for g are directions of constancy. First, we prove that each $d \in \Omega$ is a direction of constancy for $D(t^{edge})$. Given $\lambda > 0$, we have

$$
\begin{aligned}
& D(t^{edge}+\lambda d) = -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T(t^{edge}+\lambda d)) + \widehat{f}^T(t^{edge}+\lambda d) \\
= & -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge}+\Theta_{,p}^T\lambda d) + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge})\cdot exp(\Theta_{,p}^T\lambda d) + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge})\cdot exp(\lambda q_w) + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} r_w ln exp(\lambda q_w) \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge}) + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} r_w [\lambda q_w + ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge})] + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} \lambda r_w q_w - \Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge}) + \widehat{f}^T t^{edge} + \widehat{f}^T\lambda d \\
= & -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}^T t^{edge}) + \widehat{f}^T t^{edge} - \Sigma_{w\in\mathcal{W}} \lambda r_w q_w + \widehat{f}^T\lambda d \\
= & D(t^{edge}) - \Sigma_{w\in\mathcal{W}} \lambda r_w q_w + \widehat{f}^T\lambda d
\end{aligned}
$$

Notice that

$$\widehat{f}^T\lambda d = \Sigma_{w\in\mathcal{W}} h_w^T \Theta_w^T \lambda d = \Sigma_{w\in\mathcal{W}} \lambda \widehat{h}_w^T q_w 1_w = \Sigma_{w\in\mathcal{W}} \lambda q_w \widehat{h}_w^T 1_w = \Sigma_{w\in\mathcal{W}} \lambda r_w q_w.$$

Therefore, we can conclude that

$$D(t^{edge}+\lambda d) = D(t^{edge}), \tag{3.2.5}$$

which means every $d\in\Omega$ is a direction of constancy for $D(t^{edge})$. Therefore, it remains to show that no $d\in\mathbb{R}^E-\Omega$ can be a direction of recession for $D(t^{edge})$. From the proof of Theorem 3.8 in [19], we know that it suffices to show that for sufficiently large λ we must have

$$D(t^{edge}+2\lambda d) > D(t^{edge}+\lambda d). \tag{3.2.6}$$

Applying the method in the proof of Theorem 3.8 in [19], we need to show that

$$\begin{aligned}\lambda^{-1}D(\lambda d) &= \lambda^{-1}[-\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp(c_p+\delta_p+\Theta_{,p}\lambda d)+\widehat{f}^T\lambda d] \\ &= -\Sigma_{w\in\mathcal{W}} r_w ln \Sigma_{p\in\mathcal{P}_w} exp[\lambda^{-1}(c_p+\delta_p)+\Theta_{,p} d]+\widehat{f}^T d.\end{aligned}$$

As $\lambda\to\infty$, $\lambda^{-1}(c_p+\delta_p)\to 0$, we have

$\lim_{\lambda\to\infty}\lambda^{-1}D(\lambda d) = -\Sigma_{w\in\mathcal{W}} r_w \Sigma_{p\in\mathcal{P}_w} exp(\Theta_{,p}^T d)+\widehat{f}^T d.$

Since $d\in\mathbb{R}^E-\Omega$, we have $\widehat{f}^T d-\Sigma_{w\in\mathcal{W}} r_w \Sigma_{p\in\mathcal{P}_w} exp(\Theta_{,p}^T d)>0$, therefore,

$$\begin{aligned}\lim_{\lambda\to\infty}\frac{D(2\lambda d)}{\lambda d} &= \lim_{\lambda\to\infty}\left[\frac{2\lambda^{-1}D(2\lambda d)}{(\lambda)^{-1}D(\lambda d)}\cdot\frac{2\lambda}{\lambda}\right] \\ &= \frac{\widehat{f}^T d-\Sigma_{w\in\mathcal{W}} r_w \Sigma_{p\in\mathcal{P}_w} exp(\Theta_{,p}^T d)}{\widehat{f}^T d-\Sigma_{w\in\mathcal{W}} r_w \Sigma_{p\in\mathcal{P}_w} exp(\Theta_{,p}^T d)}\cdot 2 = 2.\end{aligned}$$

Therefore, (3.2.6) holds for sufficiently large λ. Together with (3.2.5), we proved the existence of optimal tolls.□

3.3 An Example for Stochastic User Equilibrium

In this example, we still consider the network defined in Figure 2.2. There is still one commodity with demand $r=1$ using this network. The probabilities to choose paths in the users' mind are

$$Pr_{p_1}=\frac{exp(-c_{p_1}^d-\delta_{p_1})}{exp(-c_{p_1}^d-\delta_{p_1})+exp(-c_{p_2}^d-\delta_{p_2})},$$

and

$$Pr_{p_2}=\frac{exp(-c_{p_2}^d-\delta_{p_2})}{exp(-c_{p_1}^d-\delta_{p_1})+exp(-c_{p_2}^d-\delta_{p_2})}.$$

Let the toll vector added by manager denoted by $[t_{p_1}, t_{p_2}]$. Therefore,

$$Pr_{p_1} = \frac{exp(-1 - t_{p_1} - \delta_{p_1})}{exp(-1 - t_{p_1} - \delta_{p_1}) + exp(-h_{p_2} - t_{p_2} - \delta_{p_2})},$$

and

$$Pr_{p_2} = \frac{exp(-h_{p_2} - t_{p_2} - \delta_{p_2})}{exp(-1 - t_{p_1} - \delta_{p_1}) + exp(-h_{p_2} - t_{p_2} - \delta_{p_2})}.$$

At stochastic user equilibrium, the manager uses the optimal toll vector t^*, therefore,

$$h^*_{p_1} = Pr_{p_1} = \frac{exp(-1 - t^*_{p_1} - \delta_{p_1})}{exp(-1 - t^*_{p_1} - \delta_{p_1}) + exp(-h^*_{p_2} - t^*_{p_2} - \delta_{p_2})},$$

and

$$h^*_{p_2} = Pr_{p_2} = \frac{exp(-h^*_{p_2} - t^*_{p_2} - \delta_{p_2})}{exp(-1 - t^*_{p_1} - \delta_{p_1}) + exp(-h^*_{p_2} - t^*_{p_2} - \delta_{p_2})}.$$

As we know, $h^*_{p_1} = 1 - h^*_{p_2}$, and because the cost of p_1 is a constant, the manager only needs add toll on p_2, so we can assume that $t^*_{p_1} = 0$. Then, we can have the following equation:

$$h^*_{p_2} = \frac{exp(-h^*_{p_2} - t^*_{p_2} - \delta_{p_2})}{exp(-1 - \delta_{p_1}) + exp(-h_{p_2} - t^*_{p_2} - \delta_{p_2})}. \tag{3.3.7}$$

The only unknown number in (3.3.7) is $t^*_{p_2}$, and it is can be solved by Newton's method.

Chapter 4

Repeated Atomic Splittable Selfish Routing

4.1 Atomic Splittable Selfish Routing

In Chapter 2 we discussed non-atomic selfish routing, now we turn to discuss atomic selfish routing. We discuss the stage game at first. Atomic selfish routing is defined as a non-atomic one: a directed graph $G = (\mathcal{V}, \mathcal{E})$ with W O-D pairs, a demand rate vector $r \in \mathbb{R}^W_{++}$, a nonnegative, continuous, nondecreasing edge cost vector $c^{edge} : \mathbb{R}^E_+ \mapsto \mathbb{R}^E_+$ and the corresponding path cost vector $c : \mathbb{R}^P_+ \mapsto \mathbb{R}^P_+$. The difference is that in the non-atomic selfish routing, each commodity represents a large population of users, each of whom controls a negligible amount of traffic; in the atomic selfish routing, each commodity represents a single user who must route a significant amount of traffic and different commodities can share O-D pairs. In non-atomic selfish routing, if different commodity share O-D pair, we combine them to a single commodity, but for commodities $w, \widehat{w} \in \mathcal{W}$ sharing the same O-D pair, we still use different path sets $\mathcal{P}_w, \mathcal{P}_{\widehat{w}}$ to distinguish them. For example, for the network described in Figure 2.1, we assume that w and $\widehat{w}$ share the origin-destination pair, therefore, $origin \to A \to B \to C \to destination$ can be treated as a path $p \in \mathcal{P}_w$ and a path $\widehat{p} \in \mathcal{P}_{\widehat{w}}$, that is to say, h_p denotes the flow assigned by w to $origin \to A \to B \to C \to destination$ and $h_{\widehat{p}}$ denotes the flow assigned by $\widehat{w}$ to $origin \to A \to B \to C \to destination$.

The atomic selfish routing discussed in [26] requires that each commodity routes all her traffic on a single path. In our model, we permit commodity w to route traffic fractionally over all paths in $\mathcal{P}_w$. This kind of atomic selfish routing is called atomic splittable selfish routing. Therefore, moving from a non-atomic model to an atomic splittable one can be viewed as identifying groups of previous independent and noncooperative traffic into single strategic agents[33]. It is easy to see that non-atomic selfish routing is a special case of atomic splittable selfish routing, when $W \to \infty$ and $r_w \to 0$ for all $w \in W$. Atomic splittable selfish routing becomes non-atomic selfish routing.

For an instance (G, r, c) in atomic splittable selfish routing, the relationship between edge cost and path cost remains the same as in non-atomic selfish routing. Cost for commodity w is $C_w = \sum_{p \in \mathcal{P}_w} h_p c_p(h)$. The social cost of flow h, which is still the cost for manager, is defined by $\sum_{w \in \mathcal{W}} C_w$ or, equivalently, $\sum_{e \in \mathcal{E}} c_e^{edge}(f_e) f_e$.

The definition of a feasible flow in the atomic splittable selfish routing is the same as definition 2.1.1, and we define the feasible path flow set for (G, r, c) as $\mathcal{F}$ and the feasible path flow set for commodity w as $\mathcal{F}_w$, respectively. At equilibrium, no commodity can decrease her cost when flows of other commodities are held fixed:

Definition 4.1.1 *Let $h = [h_1, ..., h_w, ..., h_W] \in \mathcal{F}$ for (G,r,c), the flow h is an* *equilibrium flow if, for each $w \in \mathcal{W}$,*

$$C_w(h) \leq C_w(h^*), \text{ where } h^* \in \mathcal{F} \text{ satisfying } h^* = [h_1, ..., h_w^*, ..., h_W].$$

In [5], Bhaskar et al. proved that the equilibria of an atomic splittable selfish routing is not unique in general. In [2], Altman et al. proved the uniqueness of the equilibria of an atomic splittable selfish routing if the edge cost functions are all monomials of the same degree, or they are all polynomials of degree ≤ 3. Therefore, we assume that the condition holds in our model. In [36], C. Swamy proved the existence of optimal tolls in atomic splittable selfish routing. Therefore, the manager can maintain the optimal flow in stage game.

Similar to the model in Chapter 2, the available edge toll vector set, path toll vector set and action set of manager are still $\mathcal{T}$, $\mathcal{U}$ and $\mathcal{Q}$, respectively. Then we can define our stage game for atomic splittable selfish routing,

Definition 4.1.2 *The stage selfish routing consists of*

1. *$W + 1$ players, $w, w \in \mathcal{W}$ and manager m;*
2. *the action set for w $\mathcal{F}_w$ and the action set for m Q;*
3. *the cost function for w $C_w = \sum_{p \in \mathcal{P}_w} h_p c_p(h)$ and the cost function for m $\sum_{w \in \mathcal{W}} C_w$.*

4.2 Repeated Atomic Splittable Selfish Routing with Limited History

As demonstrated in Chapter 2, if followers can observe the full history of the repeated game. Finally they will form the actual equilibrium flow, which is bad for the manager. So we assume that each commodity remembers the action of the manager in the last K stages and cannot observe the action of previous followers at all. The history set for the commodities at stage n is $\mathcal{H}^n = \mathcal{Q}^n$, and the set of all possible histories is

$$\mathcal{H} = \bigcup_{n=0}^{K} \mathcal{H}^n.$$

A strategy for the manager is a mapping

$$s_m : \mathcal{H} \mapsto \mathcal{Q}.$$

A strategy for commodity w is a mapping

$$s_w : \mathcal{H} \mapsto \mathcal{F}_w.$$

At stage n, a strategy profile $s = \{s_1, ..., s_W, s_m\}$ yields an action profile $a^n(s)$, which implies cost $C^n(a^n(s))$ for the manager. The normalized discounted cost of manager from the infinite sequence of plays is

$$(1-\delta)\sum_{n=0}^{\infty} \delta^n C^n(a^n(s)).$$

We suppose that commodities still use the estimation method described in section 2.3, which means they can get a signal β at each stage, then give the best response $B(c^d - \beta)$. As β varies by stages, this repeated game is also a stochastic game. The results of Chapter 2 still hold in this atomic splittable case. Then, we can conclude that the manager can always use a Markov(stationary) strategy as a sub-game perfect equilibrium. The manager should also use the cycle strategy defined in Section 2.6 to minimize her cost in Case 2.

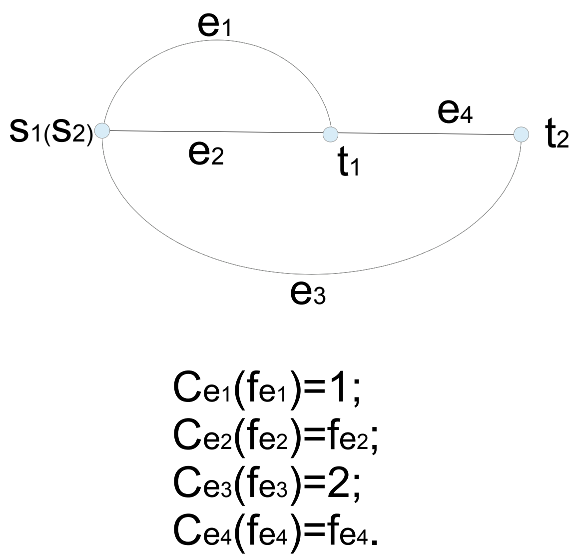

Figure 4.1: The Network for Atomic Splittable Selfish Routing Model

4.3 An Example for Repeated Atomic Splittable Selfish Routing

We use the following example to explain the result in Section 4.2. The network is defined in Figure 4.1. Notice that if there is only one player (commodity) w in the game, cost for manager is $\sum_{p \in \mathcal{P}_w} h_p c_p(h)$, which is equal to the cost for commodity w, and the manager need not add tolls. Therefore, there must be at least two commodities in the game. We assume that there are two commodities, w_1 and w_2, they share the origin node $s_1(s_2)$, and the destination nodes are t_1 and t_2, respectively. In addition, $r_{w_1} = r_{w_2} = 1$. The cost functions of all edges are illustrated in Figure 4.1.

For commodity w_1, $\mathcal{P}_{w_1} = \{p_1, p_2\}$, where $p_1 = \{e_1\}$ and $p_2 = \{e_2\}$; for commodity w_2, $\mathcal{P}_{w_2} = \{p_3, p_4, p_5\}$, where $p_3 = \{e_1, e_4\}$, $p_4 = \{e_2, e_4\}$ and

$p_5 = \{e_3\}$. Then we can have the cost of each path:

$$\begin{cases} c_{p_1} &= 1 \cdot h_{p_1} \\ c_{p_2} &= (h_{p_2} + h_{p_4}) \cdot h_{p_2} \\ c_{p_3} &= 1 \cdot h_{p_3} + (h_{p_3} + h_{p_4}) \cdot h_{p_3} \\ c_{p_4} &= (h_{p_2} + h_{p_4}) \cdot h_{p_4} + (h_{p_3} + h_{p_4}) \cdot h_{p_4} \\ c_{p_5} &= 2 \cdot h_{p_5}. \end{cases}$$

Therefore,

$$C_{w_1} = c_{p_1} + c_{p_2} = h_{p_1} + (h_{p_2} + h_{p_4})h_{p_2} = 1 - h_{p_2} + (h_{p_2} + h_{p_4})h_{p_2}.$$

If $h_{w_1} = [\frac{1+h_{p_4}}{2}, \frac{1-h_{p_4}}{2}]^T$, C_{w_1} reaches its minimum.

Similarly,

$$C_{w_2} = h_{p_3} + (h_{p_3} + h_{p_4})^2 + \frac{h_{p_4}(1 + h_{p_4})}{2} + 2 - 2(h_{p_3} + h_{p_4}).$$

Because $h_{p_2}^{SO} = \frac{1-h_{p_4}}{2}$, we have

$$c_{p_3}^{SO} = h_{p_3}^{SO} + (h_{p_3}^{SO} + h_{p_4}^{SO})h_{p_3}^{SO},$$

and

$$c_{p_4}^{SO} = (\frac{1 - h_{p_4}}{2} + h_{p_4}^{SO})h_{p_4}^{SO} + (h_{p_3}^{SO} + h_{p_4}^{SO})h_{p_4}^{SO}.$$

If $h_{w_2} = [0, 1, 0]^T$, C_{w_2} reaches its minimum.

Thus, the equilibrium flow is $h_{w_1}^{NE} = [1, 0]^T, h_{w_2}^{NE}[0, 1, 0]^T$ and $C^{NE} = 1 + 2 = 3$. However, the optimal flow is $h_{w_1}^{SO} = [0.5, 0.5]^T, h_{w_2}^{SO} = [0, 0, 1]^T$ and $C^{SO} = 0.75 + 2 = 2.75$. Thus, manager needs to add tolls to optimize the network. It is easy to see that the optimal edge toll vector is $t^{edge*} = [0, 0.5, 0, 0]^T$, so $t^* = [0, 0.5, 0, 0.5, 0]^T$.

We assume that the upper bound of the toll factor is $q = 4$. Therefore, the upper bound of the toll vector is $ub = [0, 2, 0, 2, 0]^T$. Similar to Section

Stage	Toll	Flow	μ_2	β	Perceived Cost
0	$[0, 0.5, 0, 0.5, 0]^T$	$[0.5, 0.5, 0, 0, 1]^T$	0	$[0, 0, 0, 0, 0]^T$	$[1, 0, 0, 1, 0]^T$
1	$[0, 1, 0, 1, 0]^T$	$[0.5, 0.5, 0, 0, 1]^T$	1	$[0, 0.5, 0, 0.5, 0]^T$	$[1, 0, 0, 1, 0]^T$
2	$[0, 1.5, 0, 1.5, 0]^T$	$[0.5, 0.5, 0, 0, 1]^T$	1	$[0, 1, 0, 1, 0]^T$	$[1, 0, 0, 1, 0]^T$
3	$[0, 2, 0, 2, 0]^T$	$[0.5, 0.5, 0, 0, 1]^T$	1	$[0, 1.5, 0, 1.5, 0]^T$	$[1, 0, 0, 1, 0]^T$
4	$[0, 0, 0, 0, 0]^T$	$[1, 0, 0, 1, 0]^T$	1	$[0, 2, 0, 2, 0]^T$	$[0, 1, 0, 1, 0]^T$
5	$[0, 0.5, 0, 0.5, 0]^T$	$[0.5, 0.5, 0, 0, 1]^T$	0	$[0, 0, 0, 0, 0]^T$	$[1, 0, 0, 1, 0]^T$

Table 4.1: Optimal Flow Pattern with $K = 1$

2.7, the optimal strategy for the manager and the flow pattern are shown in Table 4.1. The manager reaches the social optimum in all exploit stages. In the clean phase, she only achieves the Nash equilibrium flow for one stage.

Chapter 5

Conclusions and Future work

5.1 Conclusion

This thesis deals with the repeated selfish routing with incomplete information. In this kind of game, there are two kinds of players: users and a manager. In the network, users travels from sources to destinations, each of whom carries some flow. They are grouped by commodities. We introduce a virtual player to simulate the behavior of users in a commodity. The manager, who wants to minimize the social cost in all stages, is in charge of the network. The manager can show artificial costs of paths (but with some restrictions) and predict the users' behavior. But the users cannot predict the manager's behavior accurately because the manager has private information.

First, we assume that there are infinite users in one commodity and each user carries an infinitesimal amount of flow. If such a game played once, the manager can always keep the flow to be the optimal flow by giving proper tolls. However, this does not happen when the selfish routing is played repeatedly because the users can discover the actual costs in history and change their strategies, not just play according to the current action of the manager.

In order to find the optimal strategy for the manager, we build a reputation mechanism for her. The reputation mechanism describes how much users trust the manager by her previous actions. Then we build an estimation mechanism for the users, by which the users can choose their best response to the manager's current action. The manager knows the reputation mechanism, the estimation mechanism and the users' strategy, so she can make a decision upon all previous actions and the potential actions of the users in the current stage. Then we find out her optimal strategies in different cases.

Second, we change the model to make the cost function of each edge more complicated: we assume that even if the flow is fixed, the cost of each edge is not a deterministic number, but a random variable, which obeys a given probability distribution. This kind of game is called stochastic selfish routing. We give the definition of stochastic user equilibrium in our model. Then we prove that if each path flow is positive, the manager can always maintain the optimal flow by showing proper artificial costs.

Finally, we return to the deterministic selfish routing and assume that the number of users is finite and each of them carries an unnegligible amount of flow. Therefore, a single user should be denoted by one commodity. We prove that in this situation, the manager's optimal strategy is similar to the situation with infinite users.

5.2 Future Work

The model applied in repeated selfish routing can be improved in the following aspects:

1. We give several assumptions on the reputation mechanism in this thesis.

People can remove some assumptions and prove the optimal strategy for the manager for certain topology of the network and certain cost functions.

2. People can set the lower bound of tolls less than 0 in the model in Chapter 2. In other words, negative tolls are admitted. Negative tolls will sometimes make reputation useless because even with very bad reputation, the manager can use negative tolls to maintain the optimal flow. People can find out the optimal strategy for the manager in this case.

3. The model discussed in Chapter 3 for stochastic user equilibrium is simpler than Chapter 2 because we do not introduce the reputation and estimation mechanisms into it. People can build the same reputation mechanism for stochastic selfish routing and use the probability distribution in Equation 2.3.3 as the result of the estimation mechanism. Afterwards, people can find out the best response for the users and the optimal strategy for the manager.

4. For the model in Chapter 4, as each user carries an unnegligibe amount of flow, people can extend them to be long-run players. This extension is useless for the model in Chapter 2 as we explained in Section 2.2. Then, people can also build a reputation mechanism for the users. Therefore, the users should also consider their long-run payoff and will not be myopic. We guess that in order to build a good reputation, users should pretend to trust the manager and do what the manager wants for some stages. People can find out the optimal strategies for both sides.

5. The network discussed in this thesis just contains edges without capacity

(an upper bound of flow). One can extend all the discussion to capacitated networks[7].

Bibliography

[1] E. Alpaydin. *Introduction to Machine Learning.* The MIT press, pp. 163-172, 2010.

[2] E. Altman, T. Basar, T. Jimnez, and N. Shimkin. *Competitive Routing in Networks With Polynomial Costs.* IEEE Transactions on Automatic Control, Vol. 47, No. 1, pp. 92-96, January 2002.

[3] R. Aumann, and M. B. Maschler. *Repeated Games With Incomplete Information.* The MIT Press, Cambridge, Massachuset, pp. 71, 1995.

[4] M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics of Transportation.* Yale University Press, 1956.

[5] U. Bhaskar, L. Fleischer, D. Hoy, and C. Huang. *Equilibria of Atomic Flow Games are not Unique.* In Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms, pages 748-757, 2009.

[6] C. K. Chau, and K. M. Sim. *The Price of Anarchy for Non-Atomic Congestion Games With Symmetric Cost Maps and Elastic Demands.* Operations Research Letters, pp. 327-334, 2003.

[7] J. R. Correa, A. S. Schulz, and N. E. Stier-Moses. *Selfish Routing in Capacitated Networks.* Mathematics of Operations Research, Vol. 29, No. 4, pp. 961-976, 2004.

[8] C. Dellarocas. *Reputation Mechanisms.* Handbook on Economics and Information Systems, 2004.

[9] P. K. Dutta. *A Folk Theorem for Stochastic Games.* Journal of Economic Theory 66, 1-32, 1995.

[10] L. Fleischer, K. Jain, and M. Mahdian. *Taxes for Heterogeneous Selsh Users in a Multicommodity Network.* In Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, 2004.

[11] D. Fudenberg, D. Kreps, and E. Maskin. *Repeated Games With Long-Run and Short-Run Players.* The Review of Economic Studies, Vol. 57, No. 4, pp. 555-573, 1990.

[12] D. Fudenberg and D. K. Levine. *Reputation and Equilibrium Selection in Games With a Patient Player.* Econometrica, Vol. 57, No. 4, pp. 759-778, Jul., 1989.

[13] D. Fudenberg and D. K. Levine. *Maintaining a Reputation When Strategies are Imperfectly Observed.* The Review of Economic Studies, vol. 59, No. 3, pp. 561-579, 1992.

[14] D. Fudenberg, and E. Maskin. *The Folk Theorem in Repeated Games With Discounting or With Incomplete Information.* Econometrica, 54, 533-554, 1986.

[15] J. C. Harsanyi. *Games With Incomplete Information Played by "Bayesian" Players*. Management Science Vol. 14, No. 3, 1967.

[16] C. J. Himmelberg, T. Parthasarathy, T. E. S. Raghavan, and F. S. V. Vleck. *Existence of p-Equilibrium and Optimal Stationary Strategies in Stochastic Games.* Proceedings of the American Mathematical Society, Vol. 60, No. 1, pp. 245-251, 1976.

[17] J. Hörner, T. Sugaya, S. Taknhashi, and N. Vieille. *Recursive Methods in Discounted Stochastic Games: an Algorithm for* $\delta \to 1$ *and a Folk Theorem.* Econometrica, Vol. 79, No. 4, 1277-1318, 2011.

[18] J. Hörner, T. Sugaya, and N. Vieille, *On the Limit Equilibrium Payoff Set in Repeated and Stochastic Games.* Cowles Foundation Discussion Paper No. 1848, 2012.

[19] B. Johansson, and L. Mattsson. *Road Pricing: Theory, Empirical Assessment and Policy.* Kluwer Academic Publishers, 65-87, 1995.

[20] G. Karakostas, and S. G. Kolliopoulos. *Edge Pricing of Multicommodity Networks for Heterogeneous Selfish Users.* In Proceedings of the 45th Annual Symposium on Foundations of Computer Science (FOCS): pp. 268-276, 2004.

[21] E. Koutsoupias, and C. Papadimitriou. *Worst-Case Equilibria.* In Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science. pp. 404413, 1999.

[22] Q. Liu, and A. Skrzypacz. *Limited Records and Reputation.* Report, University of Pennsylvania, 2011.

[23] C. F. Lo, and D. Kiang. *Quantum Stackelberg Duopoly With Incomplete Information.* Physics Letters A 346, pp.65-70, 2005.

[24] G. J. Mailath and L. Samuelson. *Repeated Games and Reputations.* Oxford University Press, 2006.

[25] J. F. Nash. *Equilibrium Points in n-Person Games.* Proceedings of the National Academy of Sciences 36(1): 48-49, 1950.

[26] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, 2007.

[27] A. S. Nowak. *Existence of Equilibrium Stationary Strategies in Discounted Noncooperative Stochastic Games with Uncountable State Space.* Journal of Optimization Theory and Applications: Vol. 45, No. 4, 1985.

[28] G. Owen. *Game theory.* Academic Press, San Diego, USA, pp. 2,11,96–101, 1995.

[29] T. Parthasarathy. *Discounted, Positive, and Noncooperative Stochastic Games.* International Journal of Game Theory. Vol. 2, pp. 25-37, 1973.

[30] T. Parthasarathy, and S. Sinha. *Existence of Stationary Equilibrium Strategies in Non-zero Sum Discounted Stochastic Games With Uncountable State Space and State-Independent Transitions.* International Journal of Game Theory (1989) 18:189-194.

[31] T. Roughgarden. *The Price of Anarchy is Independent of the Network Topology.* Journal of Computer and System Sciences-STOC 2002 Vol: 67, 2, 2003.

[32] T. Roughgarden, and E. Tardos. *How Bad is Selfish Routing?* Journal of the ACM, Vol 49: 2, 2001.

[33] T. Roughgarden. *Selfish Routing With Atomic Players.* Proceedings of the 16th annual ACM-SIAM symposium on Discrete algorithms, Vancouver, January, 2005.

[34] L. S. Shapley. *Stochastic Games.* Proceedings of the National Academy of Sciences, Vol. 39(10): pp. 1095-1100, 1953.

[35] Y. Sheffi. *Urban Transportation Networks.* Prentice Hall, pp. 262-285, 1985.

[36] C. Swamy. *The Effectiveness of Stackelberg Strategies and Tolls for Network Congestion Games.* In Proc. 18th ACM Symp. on Discrete Algorithms, 2007.

[37] A. W. Tucker and R. D. Luce. *On the Theory of Games of Strategy in A. W. Tucker and R. D. Luce*, Contributions to the Theory of Games, vol. 4, pp. 13-42, 1959.