2013

Fair Cost Sharing Auction Mechanisms in Last Mile Ridesharing

Duc Thien NGUYEN
Singapore Management University, dtnguyen.2011@smu.edu.sg

FAIR COST SHARING AUCTION MECHANISMS IN LAST MILE RIDESHARING

NGUYEN DUC THIEN

SINGAPORE MANAGEMENT UNIVERSITY

2013

Fair Cost Sharing Auction Mechanisms in Last Mile Ridesharing

by

Nguyen Duc Thien

Submitted to School of Information Systems in partial fulfillment

of the requirements for the Degree of Master of Science in Information Systems

Thesis Committee:

Lau Hoong Chuin (Supervisor/Chair)

Associate Professor of Information Systems

Singapore Management University

Cheng Shih-Fen (Co-Supervisor)

Assistant Professor of Information Systems

Singapore Management University

Pradeep Varakantham

Assistant Professor of Information Systems

Singapore Management University

Singapore Management University

2013

Fair Cost Sharing Auction Mechanisms in Last Mile Ridesharing

by Nguyen Duc Thien

Abstract

With rapid growth of transportation demands in urban cities, one major challenge is to provide efficient and effective door-to-door service to passengers using the public transportation system. This is commonly known as the Last Mile problem. In this thesis, we consider a dynamic and demand responsive mechanism for Ridesharing on a non-dedicated commercial fleet (such as taxis). This problem is addressed as two sub-problems, the first of which is a special type of vehicle routing problems (VRP). The second sub-problem, which is more challenging, is to allocate the cost (i.e. total fare) fairly among passengers. We propose auction mechanisms where we allow passengers to submit their willing payments. We show that our bidding model is budget-balanced, fairness-preserving, and most importantly, incentive-compatible. We also show how the winner determination problem can be solved efficiently. A series of experimental studies are designed to demonstrate the feasibility and efficiency of our proposed mechanisms.

Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I want to give thanks to God. I would like to attribute all of my ideas and results in this thesis to His providence.

I wholeheartedly thank my family, especially my beloved wife for her tremendous understanding and support to not only this thesis but also all parts of my life.

I would like to express my gratitude to my advisors, Cheng Shih-Fen and Lau Hoong Chuin, for their insightful guidance and support thorough all steps of my Master study. I am also thankful to William Yoeh and Pradeep Varakantham for beneficial discussion and advice to increase my research perception.

Chapter 1

Introduction

1.1 Transportation problems

In recent modern decades, along with explosion of population and rapid motorization, traffic congestion is rising as an urgent problem in many countries. Direct negative consequences are cost on wasted energy, travel delay and air pollution. In a statistic provided by Texas Transportation Institute [36], the cost of wasted fuel and extra time caused by congestion in urban America increases about 3 times from 21 billion $ in 1982 to 79 billion $ in 2000 and 101 billion $ in 2010. Among causes of congestion, Passenger Vehicle accounts for 74% of the cost by the statistic in 2010 [36]. In the same report [36], it was shown that without public transportation, the wasted fuel and time delay could have increased by 17% and 24% respectively. Consistent with the increasing level of congestion, transport grew fast by the increase 23.4% from 1990 to 2004 period [21] with 76.2 Mt CO2-e (megatons of carbon dioxide equivalent) in 2004. Contribution of transportation to global greenhouse gas emissions was just behind the stationary energy sector's [21]. Private cars again contributed majority of this figure with 41.7 Mt CO2-e.

Public transportation infrastructure is constructed and developed in many countries as a solution for handling congestion. To persuade more passengers to take public transport, it is necessary to provide more comfort and incentives. However, there are many inconveniences that discourage people from using public transportation. The main source of inconvenience comes from insufficiency of the service quality. Taking Singapore as an example, the government is trying to expand and improve current public transport with increasing additional number of buses from 16,309 in 2010 to 17,241 in 2012 [31]. It was shown in a recent survey of Singapore passengers on public transportation [30] that the satisfaction index has significantly dropped to -8.7% for Mass Rapid Transit (MRT, or subway) and -7.2% for public buses from 2011 to 2012. In another report [31], it reveals that private cars population keeps increasing by 13 thousands from 2011 to 2012 in comparison to 8 thousands from 2010 to 2011. It seems that the effort at improving public transport has yet to meet passenger expectations.

1.2 Ridesharing

The high ownership and use private cars for commuting are the main cause of congestion and air pollution. By The International Energy Agency's (IEA) forecast [20], from 2000 to 2050, car ownership in Asian developing countries could grow about twenty-fold. Beside causing traffic problems, it raises issues of land for parking, energy and natural resource for car manufacturing. Policy makers and transportation planners can encourage people to limit the use of private cars by devising good trip sharing plans and providing strong public transportation benefits.

Ridesharing is considered as an efficient measure to limit use of single-occupancy vehicles. Ridesharing entails the formation of trips by at least 2 passengers in a vehicle. It takes many forms: taxi sharing, carpooling, van pooling, private shuttle, etc. It could be recurring trips with long term commitment between passengers or one time shared trip between strangers. Beside the feature of small size vehicle (vs mass transport in public transportation individual routing and cost splitting makes ridesharing more passenger-oriented and distinguished from public transportation. In ridesharing, the group trip planning and execution is based on passenger demands. As the trip is designed to pick up and deliver passengers at (or close to) passengers' beginning and ending points, ridesharing provides a door-to-door service which is more convenient than public transportation.

Ridesharing is not a new idea. In the US, its first success story dates back to the 19^{th} century with Jitney Craze and World War II. Ridesharing was restored by the US government during the energy crisis in the 1970's [6]. Recently with economic recession and more concern about the environmental and congestion issues, ridesharing has been promoted by the US and European governments with many trial systems, for example German Flexible Operations Command and Control System (FOCCS), San Francisco Bay-area Ride Now, European Commission OPTI-TRANS, Smart Traveler in the U.S, etc [6]. Some companies even tried to encourage employees to travel more with ridesharing to receive monthly incentive gifts [35]. To facilitate matching of individual travellers for ridesharing, many web-based and mobile applications have emerged. Interested readers may refer to some popular websites http://www.carpoolworld.com/, http://www.myridebuddy.com/, or recently launched apps in Singapore http://www.gomywayapp.com/, http://www.split-it.sg/.

Beside positive social effects on road congestion, air pollution and energy resources, ridesharing incentivizes passengers by monetary benefit, especially in the economics recession and fuel crisis [6],[35]. Besides being a door-to-door service, it is much cheaper than private cars because the fuel cost of hired vehicles is fairly split among passengers. However, ridesharing might be less attractive than private cars because of additional travel distance and discomfort of sharing cars with strangers. The additional travel distance is inevitable in some cases in which the detour is necessary to serve different locations in the same trip.

1.3 First Mile and Last Mile problems and solutions

The segment incurred by public transport passengers from their homes to the nearest public transport (bus/train) station and vice versa is known as "First Mile" and "Last Mile" (LM) respectively. This inconvenience is often a major deterrence in utilizing public transport services.

A transportation solution to fill this First (Last) Mile gap is not easy. Expanding the bus and train network is long term and complicated problem. It is economically inefficient to design frequent routes to cover every residence, especially for underpopulated regions or to handle dynamic demands.

A straightforward idea to satisfy the LM demands is to establish a service fleet for each major transport hub. However, due to the fact that the demands for the LM transportation are irregular and distributed (both spatially and temporally), having a fixed-size service fleet is infeasible, for the following intuitive reasons:

1. Demands are highly irregular and uncertain. Therefore, to ensure that the fleet can cope with peaks in demands, the fleet has to run with spare capacity that would be underutilized most of times.

2. To ensure reasonable quality of service, the routes of the fleet have to sufficiently cover most of the service area (the travel time from any point in the area to the closest stop should be within certain minutes) with reasonable service intervals (this constrains longest waiting time). The fleet can operate statically with fixed routes, or it can operate dynamically with routes depending on passengers onboard; however, in either case, significant slacks have to be introduced in the fleet so as to handle the spatial and temporal demand uncertainties.

Due to the above issues, operating fixed-size fleets is cost-ineffective for most occasions except for the very limited cases where demands are consistently high.

A powerful idea in addressing unpredictable travel demands is *sharing*, or *resource pooling*. For example, in many European countries, the bike sharing [38] and car sharing [44], [37] schemes have been suggested as a way to bridge the gaps of public transport. In these instances, resources (bikes and cars) are pooled at fixed locations, and travelers will grab resources, if needed, to complete their travels. Resources are pooled and resource utilizations are independent. Ridesharing (car-pooling or taxi-pooling) is a such typical case.

Ridesharing provides an effective solution to solve this problem, typically with taxi-pooling for urban Asian cities like Singapore where the number of taxis is significantly large [31] in comparison with the size of public transportation. Statistically, over 50% of the time, a taxi is spent on idling (i.e. empty cruising or waiting in queues). The availability of taxis can be a potential support for a ridesharing solution to solve the Last (First) Mile problem. Especially a non-dedicated taxi system is efficient to cope with the dynamic demand nature of the problem, where drivers are called only by demanded and do not have to commit any recurring trip plan.

1.4 Mechanisms for Last Mile Ridesharing

In this thesis, we study the problem of ridesharing with a non-dedicated taxi fleet to solve the Last Mile problem. Given a batch of arriving passengers and a set of available taxis near the station, we need to design ridesharing trips and specify the fares of riders. For the sake of simplicity henceforth, we will refer this problem as the Taxi Sharing Last Mile problem (or simply Last Mile Problem).

The Last Mile problem contains 2 sub-problems: routing and cost allocation (or cost sharing). Complete solutions for the Last Mile problem need to provide the routes for the taxi fleet as well as the fare payment for passengers. In this thesis, we design market mechanisms to produce those solutions. More precisely, we consider two aspects in mechanisms design:

- From the routing standpoint, our problem is a special case of the Dial-a-Ride vehicle routing problems (DARPs). We derive an exact model from existing DARP models. We solve the routing problem with a specific objective function. The efficiency of our modified model is improved in comparison to the original one by leveraging the special structure of small capacity and other taxi-related constraints. To handle large scale problem instances, we propose heuristic algorithms, i.e. Tabu search and randomized search. Their solutions yield a small gap in comparison with exact method's solutions, and is capable of returning solutions in real-time, which is an important characteristic to make this approach operationally viable.

- To incentivize passengers to join ridesharing services, the fares should be fair and competitive. Acknowledging the heterogeneity of passengers joining the ridesharing service, we design protocol that allow passengers to specify their willing payments. In particular, we design an auction protocol where Last Mile passengers submit not only their destinations but also the willing payment to use the sharing service. The allocation of cost (payment) need not only to be operational optimum but also sustainable under passengers' rationality. Specifically, mechanisms are designed to be incentive compatible, i.e. there would be no incentive for passengers to be dishonest in disclosing their bid information. The general principle of our incentive compatible mechanisms is that passengers with high willing payment will have more chance to be served than low willing payment passengers.

In summary, the main purpose of this research is to study and propose mechanisms for the Last Mile problem. We need to compute solutions efficiently, and the mechanism must be fair and incentive compatible. In this thesis, we show theoretical properties of proposed mechanisms. Feasibility and efficiency of mechanisms are measured by a series of experiments.

For the sake of simplicity, in this thesis we assume the vehicle's velocity to be equal to 1, the meter rate of running the service to be 1. Hence, the travel distance is also equal to the travel time and driver's revenue. For real world implementations, it is straightforward to incorporate constant parameters and compute these values by linear transformation.

1.5 Thesis Structure

The structure of my thesis is as follows:

- The literature review on vehicle routing and cost sharing is given in chapter 2. We present vehicle routing models which are applicable to the routing problem in taxi sharing. For the cost sharing problem, we consider existing cost sharing approaches in vehicle routing as well as in other game settings. For each approach, we discuss the feasibility and drawback of applying to our Last Mile Ridesharing problem.

- In chapter 3, we propose and develop a Mixed Integer Program (MIP) and Local Search models to solve routing problem. An MIP model is considered to provide exact solution for Last Mile problem. However, MIP model is infeasible to handle large scale problems. Therefore, an heuristic search algorithm is developed to provide routing solutions in real time.

- A bidding protocol and cost sharing models with discount part are proposed in chapter 4. After analyzing pros and cons of each model, we choose the bidding protocol of meter rate and set the discount rate capped by meter rate. In this chapter, we also propose an optimal mechanism to produce the optimistic solution as a benchmark for incentive compatible mechanisms.

- We propose incorporated mechanisms for our cost sharing problem in chapter 5

  - Top-down mechanism solution is a baseline of incentive compatible mechanisms.
  - Bottom-up mechanism is an advanced version of Top-down mechanism.
  - Raising cost mechanism provides solution by differentiating meter rate of passengers.

  Routing models of these mechanisms have nonlinear objective function, which make problem unsolvable by normal MIP model. We propose a hill climbing search method to efficiently compute solutions of mechanisms.

- Chapter 6 contains experimental results of proposed mechanisms. We measure efficiency of mechanisms using synthetic and real data sets. With focus on maximizing the number of served passengers and total direct distance of served demands for large scale problems, we experiment on our Bottom-up mechanism with Local Search algorithm. The experimental results in real data show this approach is promising to serve almost all Last Mile demands.

- Chapter 7 is a summary of our contribution in this thesis and discussion of future work in implementing Ridesharing service for the Last Mile problem, including user interface and usability.

Chapter 2

Literature review

In this chapter, we review existing works in three aspects of Ridesharing:

- In the first section, we review some existing ridesharing systems, including mobile or e-ticket platforms for riders. We also report some contemporary results from different implementations of Ridesharing systems.
- In operations research, Ridesharing is considered as a special class of vehicle routing or traveling salesman problem. The challenge is to design routes for vehicles (in real time). We review mixed integer programming and heuristic methods to solve the problem.
- In cost sharing aspect, Ridesharing is associated with several popular game-theoretical concepts, e.g. coalition, Vickrey-Clarke-Groves (VCG) and auction mechanisms. In this thesis, we develop auction based mechanisms for Last Mile Ridesharing problem, so the review on cost sharing in this aspect will be presented.

2.1 Ridesharing Systems

In a recent survey on ridesharing [4], the authors defined several special characteristics, which are applied into our Last Mile taxi sharing service.

Dynamic Ridesharing planning is triggered by real-time demands of passengers. In our Last Mile taxi sharing problem, passengers send their requests a short period before arrivals. Future demands are not known apriori.

Cost-sharing A taxi sharing mechanism needs to provide a complete solution for each batch of passengers, which contains both the routes and the cost allocation.

Non-recurring trips Trips are non-recurring, which is different from standard vehicle routing problem where vehicles return to the depot after each round.

Automated matching The center system automatically provides solutions that assign each passenger to a taxi. The computation needs to be fast and reliable.

Combining trains with taxi sharing to serve the last mile is not a new idea. In fact, the concept of a demand adaptive system was first studied in [13] in which the authors suggested the combination of flexible line and conventional swift lines (train and express bus) to increase the coverage of the transportation system. They proposed an operational planning model to seamlessly connect the long distance transportation with flexible vehicles. Our taxi sharing solution is a variant of this demand adaptive concept, in which non-dedicated taxis play the role of a flexible line.

There are many studies on the technology side of ridesharing platforms. In [19], authors reported their work on construction of an efficient database structure to handle cab-sharing demand queries. Amey in [7] studied different problems and solutions of utilizing mobile phone technology in real time ridesharing. The paper [34] introduced an incorporated mobile context-based ridesharing platform named WEtransport which facilitated passengers to find ride-mates and share the train ticket or taxi fare. Authors in [43] showed a social media framework to support dynamic ridesharing scheme. The paper [41] introduced an agent-based ridesharing architecture via lightweight devices like mobile phones. As a special ridesharing service, Last Mile taxi sharing can inherit most of these existing technologies with some light modification or extension.

To prove the feasibility and effectiveness of ridesharing, researchers study implementation of ridesharing by different methods. Some researches verify ridesharing services under a simulation environment [5],[9]. Simulation data from matching personal drivers and riders shown in [5] demonstrated a potential and sustainable success of dynamic ridesharing. Focusing on taxi sharing, [9] reported his work on collective taxi system implementation at large scale in different models, including Last Mile transportation.

Success and failure factors of ridesharing raise an interesting research topic in ridesharing. Based on a survey about an implemented Carsharing service in Lisbon, [12] used a discrete choice model to recognize factors affecting ridesharing the most. To address the problem from the policy maker perspective, [6] showed the current trend of ridesharing by analyzing different implemented ridesharing schemes in Europe and the US. This work provided a detailed analysis of economics and social effects of ridesharing. In addition, it also suggested some solutions to improve and establish ridesharing services in the future.

2.2 Vehicle Routing Problems

Given a set of passengers and taxis, the goal of vehicle routing is to assign each passenger to a specific taxi and design a route for each taxi that optimizes a particular objective function. Common objectives in ridesharing routing are to minimize the total travel time/distance $T = \sum_{i,j} x_{ij}t_{ij}$ and maximize the number of served customers $\sum_{i,j} x_{ij}$ [4].

The routing sub-problem of our Last Mile problem is a specific instance of vehicle routing in Dial-a-Ride settings [27]. Dial-a-Ride is a demand-responsive system to provide door-to-door service with specific requirements on passenger's pickup and delivery locations and times. It comprises personal cars, vans or small buses in response to calls from passengers [29]. Dial-a-Ride problems (DARPs) belong to a class of vehicle routing problems with capacity and time window constraints. A detailed summary of characters and classification of Dial-a-Ride problem can be found in the review [10].

Similar to other NP-hard problems in vehicle routing, there is no efficient method to solve Dial-a-Ride routing problem. [17] modelled a generic Dial-a-Ride routing problem by mixed integer program (MIP) model and applied different subtour elimination constraints to strengthen the LP-relaxation of MIP. For the identical vehicle case where vehicle capacities are the same, [33] proposed a simplified version of [17]. However, [33] requires to enumerate a large number of subset constraints, which is not necessary in our model.

For complicated and large scale instances in which exact methods like MIP fail to obtain a solution in real time, heuristic and meta-heuristic methods are utilized. Studying DARPs, [11] proposed a Tabu search procedure to find locally optimal solutions, in which routing solutions were represented by ordered route vectors. In each local search iteration, the algorithm removed a pickup and delivery pair in one link and simply inserted into another link, with orders for other passengers were reserved. To gain diversity, they relaxed the time and capacity constraint by adding a penalty element into objective function. It is different from our local search procedure presented later. In our method, we do not represent routing solution in form of ordered routes but we consider cluster assignment (passenger-taxi) vectors. As a suitable local search operator is defined, the feasibility of solution is preserved over local moves. A genetic method for solving Dial-a-Ride problem is proposed in [23]. To solve DARPs by genetic algorithm, [23] proposed genetic operators in cluster vectors, where each element was index of passenger assigned to that cluster. Based on the same cluster structure, [46] proposed a simulated annealing algorithm and they reported that their algorithm produced better results than Tabu search [11].

Recently, [27] provided a summary of recent Dial-a-Ride algorithms to apply in ridesharing taxi. From this summary, we notice special structural characteristics of specific systems like taxi system was not addressed. We show in this thesis that adaptation and modification from generic models are essential to reduce the complexity for our sub problems, i.e. taxi routing. One example is reduction of redundant vehicle index variables. In addition, we can simplify pickup constraints into single pickup point because in our cases passengers arrive the same hub at the same time. In the next chapter, the detail of modification will be presented and integrated into our Last Mile taxi sharing problem.

2.3 Cost sharing in vehicle routing problem

Although the vehicle routing problem has a long history in operations research, the study of cost sharing has just drawn attraction very recently as an application of game theory. This section is aimed to briefly introduce some techniques on recent works related to our cost sharing problem.

2.3.1 Coalition mechanisms

The coalition value is a fundamental notion in cooperative game theory. It is the optimal cost or utility created by cooperation of a group of players. The core of a game is ideal solution for all players such that value of any group of players is not less than coalition value of it [40].

Formally, the coalition value is represented by a characteristic function as the cost or utility to serve each set (or coalition) of players $v : 2^{\mathbf{N}} \to \mathbb{R}$. The core is the cost allocation $\sum_{i \in \mathbf{N}} p_i = v(\mathbf{N})$ for all players such that there is not any of its subset value dominated by corresponding coalition value, in other words

$$\forall S \in \mathbf{N}, v(S) \geq \sum_{i \in S} p_i. \tag{2.1}$$

Core solution

The core value is suggested to be directly used to charge passengers in vehicle routing [15],[42]. To find core solution in vehicle routing game, a popular method is to use constraint generation procedure to iteratively solve the problem [15], [18]. Although core solution is ideal for cost allocating problem, there is no guarantee for existence of such solution in general vehicle routing game [42]. In some cases, we can only find an approximation core [16] instead, which is solution only recovering part of the service cost. In non-dedicated taxi service, trading off budget balance to get approximation solution is infeasible because taxi driver needs to get a sufficient amount from passengers to run the service.

Shapley solution

Another popular cost allocation method in cooperative game is Shapley value which is average marginal cost over all coalition [39]

$$p_i = \sum_{S \in N \setminus i} \frac{|S|!(n-|S|-1)!}{n!}(v(S \cup i) - v(S)), \tag{2.2}$$

in which $v(S)$ is the coalition value of coalition S.

The Shapley value is well known as a solution satisfying 4 Shapley axioms: budget balance, symmetry, linearity, zero player. In contrast to non-existence of core value in some cases, Shapley value always exists in any cooperative game. Shapley value is suggested to be used in vehicle routing game to allocate the service cost to passengers[15], [45].

2.3.2 Auction based mechanism

One of the shortcomings of conventional cost allocation mechanisms is the disregard for heterogeneous utility of passengers. In our taxi sharing problem, different routing solutions make passengers ”suffer” various extra travel distances. Moreover, passengers also have different utlity functions: some of them could require high compensation for extra travel distances; some of them could have low willing payment with the sharing service.

Given a set of available resources and willing payment of each passenger to use the service, auction based cost sharing studies how to distribute service and allocate the cost between served passengers. It requires a bidding protocol, where each passenger (or player) needs to submit not only her/his demand but also the willing payment. The common desirable property of auction based mechanisms is strategyproofness, which ensure a passenger to not have benefit to be dishonest about the bid information.

VCG mechanism

The VCG price scheme charges player by the marginal benefit he/she contributes into the service. This value is calculated for each player in game by formula $p_i = (V_{-i})^* - V_i^*$ where V_{-i}^* is total social benefit of all passenger minus social benefit of i in optimal solution for grand set N, and $(V_{-i})^*$ is the total social benefit of optimal solution of subset $N \setminus i$. This value can be simply characterized as coalition value.

Studying cost allocation in ridesharing, [24] proposed a VCG based mechanism to provide cost allocation for ridesharing participants. Although VCG is a truthful and budget balance scheme, it is not budget balancing. [24] tried to recover budget balance by trading off truthfulness and claimed that it was reasonable under bounded rationality. ”Semi” truthful and approximated budget balance can make non-dedicated taxi sharing service unstable for our Last Mile problem.

Second price based mechanism

In [26], Kleiner et al. proposed strategyproof auction based mechanism for ridesharing, where passengers submit their bids on willing payment to share the car with the driver. They provided a modified second-price auction mechanism which picked up the highest rank passenger that maximized benefit for driver then it charged the passenger a cost of the second highest bid. This modified second price auction mechanism is incentive compatible. Despite having this good property, this mechanism cannot apply to our problem because it is limited in bilateral scenario with matching algorithm to output solutions for pairwise driver-passenger. In our taxi sharing problem, we are challenged by clustering passengers in respect to their willing payment and spatial relation. In our Top-down and Bottom-up mechanisms described later, we modify the idea of second-price auction schema by setting a lower bound of first unserved passenger, corresponding to the second highest cost in the second price auction, to the cost of served passengers.

Raising cost mechanisms

Auction based mechanisms are studied in many game settings before vehicle routing application. Among them, set cover game [14] is considered as a general combinatorial game setting for different problems, where it requires different costs for different subsets of passengers. In fact, vehicle routing can be modeled as a set covering problem [25], in which each cluster with optimal routing is corresponding to a subset in set covering.

A popular technique used to solve these cost sharing problems is the cost raising procedure ([14],[32]) which often guarantees strategyproofness. In addition, the raising cost procedure could be combined with other techniques like primal-dual [28] or ”ghost processing” [32] to obtain the core solution or cross monotonic property but it usually needs to replace the budget balance by some approximations, which recover only a fraction of the service cost, for example in set covering it is only $1/H_n$, H_n is harmony number [14], in facility location it is one third of the cost [14]. Although approximation budget balance could help to gain interesting properties, it would be unrealistic in implementing of non-dedicated taxis because taxi driver will not agree to run the service unless money from passenger is sufficient. Therefore in non-dedicated taxi context, we focus on designing a strategyproof and budget balance solution, which recovers full service cost, rather than property of approximation budget balance solution.

Chapter 3

Taxi Sharing Routing Problem

Given a set of passenger demand and available taxi, routing is the problem of finding efficient trip to optimize an objective function. In this chapter, we study different routing solutions for taxis routing problem in Last Mile ridesharing service. A Mixed Integer Program (MIP) is adapted from DARPs model to provide exact solution for the problem. We show modification from original model to fit into specific domain and improve the scalability of the model. Although the new MIP is more scalable in comparison with its origin, it is still infeasible to handle large scale problem, i.e. instances with number of passengers above 32. Observe this limit of MIP, we propose a heuristic local search algorithm to provide solution in real time.

3.1 Mixed Integer Programming (MIP) Formulation

3.1.1 Last Mile routing problem and DARP

Given a set of pickup and delivery locations and their time windows, and a set of vehicle and their capacities, DARP is the problem of designing routes for the vehicle fleet to serve as many transport demands as possible. Routing in our Last Mile problem is a sub-problem of DARP, which could be optimized by eliminating redundant variables:

– Since the capacity of each taxi is assumed to be homogeneous, which is 4 (seats) in a normal taxi, it is possible to not enumerate the taxi index. Consequently, a large number of variables related to the taxi index can be simplified. Under this simplification to keep track of the sequence of serving order we could use an additional variable a_i .

– There is no time window constraint in our Last Mile problem. However to control the inconvenience caused by addition travel distance we need a variable to measure individual travel distance in the trip. For the sake of simplicity, we assume that travel time is equivalent to travel distance, i.e. velocity of taxi equal to 1.

– In a non-dedicated taxi system, after finishing the trip, taxi drivers could have flexible options of roaming or trip pick up instead of the obligation to return to depot. Hence we plan one way trip instead of round trip.

3.1.2 Clustering Constraints

Let **N** denote the set of passenger demands and K be available vehicles, each with capacity Q. The Last Mile routing problem is defined as a graph with depot and destinations as vertices $\{0, 1, \ldots, |\mathbf{N}|\}$. An edge in the graph between 2 nodes i and j is weighted by the travel distance t_{ij} between 2 locations.
To express the connection of 2 nodes in planned trips we use binary indicator

$$x_{ij} = \begin{cases} 0 & \text{if there is no trip traveling directly to } j \text{ from } i \\ 1 & \text{if there is trip traveling directly to } j \text{ from } i. \end{cases} \tag{3.1}$$

Let B_i, s_i denote individual trip travel distance and direct travel distance from depot (vertex 0) to passenger destination i.

To produce a valid ridesharing trip, the following constraints have to be satisfied [8]:

$$\sum_{i \in N} x_{ij} \leq 1, \forall j \in N, \tag{3.2}$$

$$\sum_{j \in N} x_{ij} \leq 1, \forall we \in N, \tag{3.3}$$

$$\sum_{j \in N} x_{ij} \leq \sum_{h \in N} x_{hi}, \forall i \in N, \tag{3.4}$$

$$\sum_{i \in N} \sum_{j \in N} x_{ij} \leq \min\{|N|, K \times Q\}, \tag{3.5}$$

$$\sum_{j \in N} x_{0j} \leq |K|, \tag{3.6}$$

$$B_j \geq B_i + t_{ij} - M\left(1 - x_{ij}\right), \forall i, j \in N, \tag{3.7}$$

$$B_j \leq B_i + t_{ij} + M\left(1 - x_{ij}\right), \forall i, j \in N, \tag{3.8}$$

$$B_j \geq 0, \forall j \in N, \tag{3.9}$$

$$a_0 = 0, \tag{3.10}$$

$$a_j \geq a_i + 1 - M(1 - x_{ij}), \forall i, j \in N, \tag{3.11}$$

$$a_j \leq a_i + 1 + M(1 - x_{ij}), \forall i, j \in N, \tag{3.12}$$

$$a_i \leq Q, \forall we \in N, \tag{3.13}$$

$$x_{ij} \in \{0, 1\}, \forall i, j \in N \tag{3.14}$$

$$\tag{3.15}$$

- $\sum_{i \in N} \sum_{j \in N} x_{ij}$ in (3.5) quantifies total number of served passengers. $\sum_{j \in N} x_{0j}$ in (3.6) is number of vehicles departing from Depot.

- Constraints (3.2) and (3.3) limits at most 1 arrival and 1 departure at each passenger vertex in the graph, consequently each passenger is served by at most 1 time by 1 vehicle. The in-out constraint (3.4) ensures vehicle departs from i only if it arrived i.
  Passenger j is served if there is trip traveling to her/him, or $\exists i, x_{ij} = 1$.

- Inequalities (3.7) and (3.8) are linearization from the sequencing constraint

$$B_j = x_{ij}(B_i + t_{ij}), \tag{3.16}$$

which implies that the trip travel distance of passenger j is the sum of her/his direct predecessor's travel distance and the travel distance from that predecessor. This relation is expressed by inequality (3.7) and (3.8) with big constant M to relax the bound for B_j if i is not a predecessor of j or tighten the bound otherwise.Particularly, when there is a trip traveling from i to j, $M(1 - x_{ij})$ is zero, combination of (3.7) and (3.8) directly leads to (3.16).

- Variable a_j is introduced to specify service order of passenger, e.g. if j is the first passenger arriving the final destination then $a_j = 1$, if he/she is the second one then $a_j = 2$ and so on. Similarly to B_j variable, (3.11) and (3.12) are linearization of service sequence constraint

$$a_j = x_{ij}(a_i + 1) \tag{3.17}$$

meaning that service order of j is greater than service order of its predecessor by 1.

The original model in DARPs uses the vehicle index k in x_{ij}^k to decide which vehicle will serve the trip directly traveling from i to j. Vehicle capacity constraint can be modeled by $\sum_{i,j \in N} x_{ij}^k \leq Q_k, \forall k$, in which Q_k is capacity of vehicle k. With assumption of homogeneous taxi vehicle, we discard vehicle index to reduce the complexity of the problem. Under this simplification, we express the capacity constraint in another way by the length of service sequence or $\max_{j \in N} a_j$ in inequality (3.13). It implies that the service order of any passenger cannot exceed the capacity of taxi. By removing vehicle index and using service order variable, we reduce number variable from $(|\mathbf{N}|^2 \times K)$ of x_{ij}^k variables to $(|\mathbf{N}|^2 + \mathbf{N})$ of x_{ij} and a_j variables. This reduction results in improvement of runtime in comparison with original formula.

Quality of routing solutions

There are some common criteria to measure the quality of a routing solution:

- In term of social welfare, we consider the objective function to maximize the total number of served demand. If there is more than one routing solution to serve the same number of passengers, the mathematics model would give priority to serve passengers with further destinations. This objective can be quantified by

$$\max \sum_{i,j \in N} x_{ij} + \frac{1}{M} \sum_{i,j \in N} x_{ij} s_j, \tag{3.18}$$

in which $\sum_{i,j \in N} x_{ij}$ is the total number of served passengers, $\sum_{i,j \in N} x_{ij} s_j$ is total direct distance of corresponding served demand. M is a big value to calibrate second term to set priority to first term.

- Minimizing travel cost is preferred by both passengers and transportation planner. It is usually related to minimize total travel distance $\sum_{i,j \in N} x_{ij} t_{ij}$ and minimize extra travel distance $\sum_{j \in N}(B_j - s_j)$.
- Sometimes, worst case criteria are considered. This quality of transport service can be measured by ratio between real travel distance and direct travel distance $\alpha = \max_{i \in N} \frac{B_i}{s_i}$. To improve quality of transportation, one can think of minimizing this α ration.

Modified MIP model experimental result

To verify improvement of our MIP model over original DARP model, we carry on experiment to find solution minimizing travel cost $\sum_{i,j \in N} x_{ij} t_{ij} + \sum_{j \in N}(B_j - s_j)$ under different problem sizes, i.e. 8, 20, 24, 32 passengers and the number of taxis is 4, 5, 6, 8 respectively.

Riders	Drivers	Time / Time using DARP Model
8	4	0.37s / 43s
20	5	24.06s / 3m14s
24	6	1m24s / no result after 3h
32	8	17m22 / no result after 1d

Table 3.1: Summary of the LM planning results for different problem sizes (all Δ_i are set to 1).

Table 3.1 summarizes the performance statistics we obtain under different problem sizes. The first thing to note is the significantly improved solution speed. For the largest instance, our simplified model returns solution within 17.5 minutes, while the classical DARP MIP model runs over one day without terminating. For all instances, our formulation is at least two orders of magnitude faster than the classical DARP MIP model.

3.2 Local search algorithm

MIP is a popular model to produce exact solution for linear problem. However, for problems with large number of variables and constraints, MIP is inefficient and unrealistic in real time computing system because it may take runtime of hours or days period to output final solutions. Alternative solutions from heuristic algorithms are useful in these cases. In this section, we propose a heuristic local search algorithm with random initial start point.

Our local search algorithm contains 2 phases: clustering and routing. The first phase is to specify the cluster of passengers, i.e. each passenger is assigned to a taxi. In second phase, from each cluster the detailed trip for each taxi is computed by exhaustively searching over all possible permutations. Since the capacity of taxis is small (typically 4), the number of permutations is scalable, e.g. $4! = 24$, which makes exhaustive search become an efficient method to produce solution in short runtime. To avoid recalculating the best permutation for the same cluster many times, in our implementation we use a hash data structure to store the best permutation for the corresponding cluster after each calculation.

Our local search algorithm is developed to solve 2 types of routing problems in our proposed mechanisms in Chapter 5

- to serve a given number of passengers. These solutions are meant for the ideal optimality and Bottom-Up mechanism.
- to serve some additional passengers and a fixed set of passengers. These solutions are meant for the Raising Cost mechanism.

3.2.1 Solution Structure

A local search solution is defined by k clusters corresponding to the k available taxis, each cluster contains a maximum of 4 elements corresponding to the taxi capacity. This cluster structure is dynamic in that elements can be added or removed from the cluster with constraint on capacity. An example of a local search solution is shown in Figure 3.1. From the cluster solution, we will compute the optimal route by exhaustive search over all permutations of elements within the cluster.

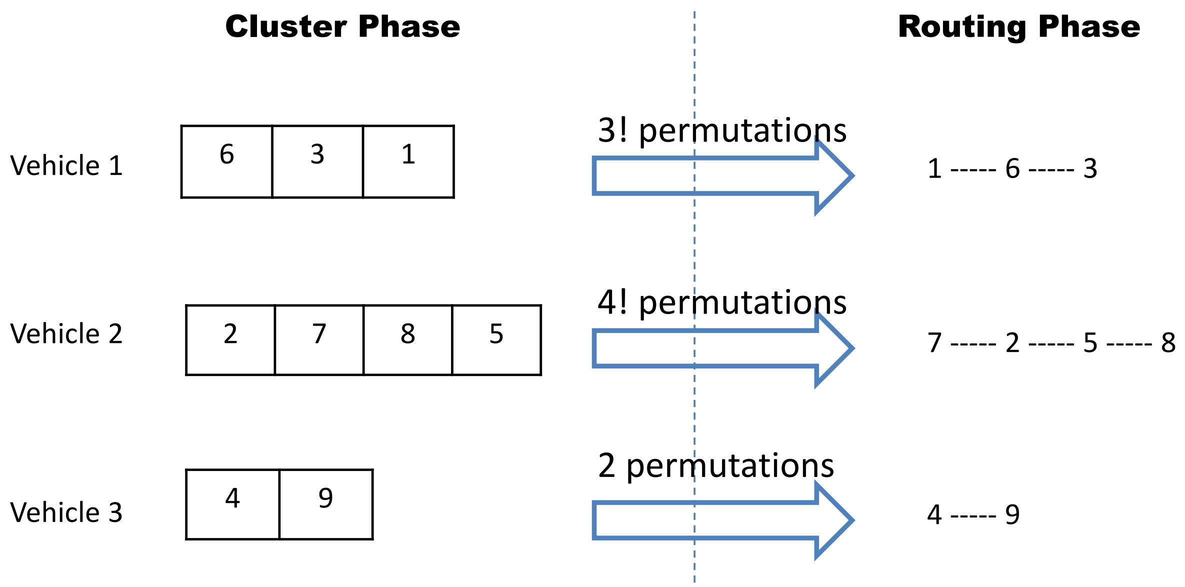


Figure 3.1: An example of Local Search.

3.2.2 Local Search algorithm

We propose a local search shown in Algorithm 1.

To explore the search space, we define the "Shuffle" operator as shown in Procedure 21 and illustrated in Figure 3.2. The $Shuffle(\{v(i)v(j)\})$ operator returns the best clusters formed by elements of 2 clusters i and j. First, it forms a list of passengers $\{p_1, \ldots, p_k\}$ from these 2 clusters. Then it computes all bi-partitions from this list to find the best bi-partition. In example 3.2, apply $Shuffle()$ to vehicle 2 and 3, a list of passengers $\{2, 4, 5, 7, 8, 9\}$ of vehicle 2 and 3

Algorithm 1: Local Search algorithm

Initialize: $v(1,\ldots,k) = ranInit(N)$;
while *(stop condition != false)* **do**
  for $i \leftarrow 1$ **to** k **do**
    for $j \leftarrow i+1$ **to** k **do**
      $v^{'} = Shuffle(\{v(i)v(j)\})$;
      if $computeCost(v^{'}) < computeCost(v)$ **then**
        $v = v^{'}$;
        break;
      end
    end
  end
end

is formed. A pool of $\binom{7}{4}$ combinations is enumerated. The best partition is chosen.

The local search algorithm begins with a randomly initialized solution then improves its solution by finding first improvement by the Shuffle operator in each step. We iteratively explore all pairs of 2 clusters to check if there is any improvement by applying $Shuffle()$. The local search terminates when the maximum time elapses or when it reaches a local optimum. To enhance the chance to reach an optimal solution, we diversify local search solutions by parallel runs with independently random initial solutions.

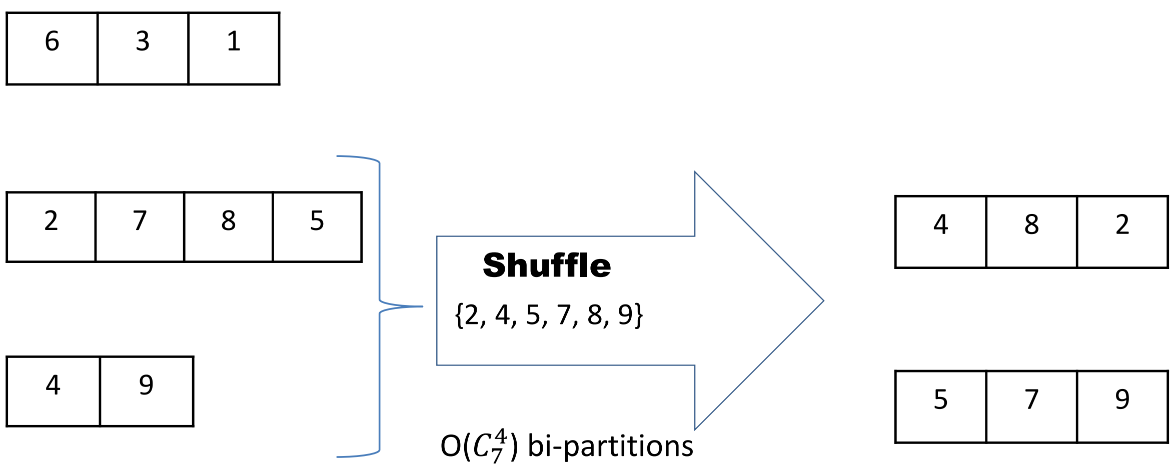


Figure 3.2: Shuffle Operator.

Procedure Shuffle($\{p_1, \ldots, p_k\}$)

$m_{best} = \infty$;
$v_{best} = v$;
forall the $[c_1, c_2] = partition(\{p_1, \ldots, p_k\})$ **do**
    $m = computeCost(c_1, c_2)$;
    if $m < m_{best}$ **then**
        Update v_{best} with $[c_1, c_2]$;
    end
end
return v_{best};

By exploiting structure of small capacity taxi to use the exhaustive search for each cluster, our local search algorithm can produce a solution almost the same as optimal solution produced by an exact method within a short time. The experimental results for local search algorithm can be found in chapter 6.

Chapter 4

Cost Allocation Model

Cost allocation is a critical part to ensure participation in ridesharing. By participating in ridesharing, passengers expect to get incentivized by a fairly distributed service cost. The payment of passengers should be aligned with their service utility. More challenging is the heterogeneity of passengers, in other words, passengers may have different utilities or willing payments on the travel service. In this chapter, we consider 2 problems: how to quantify cost division in regard to passengers' inconvenience and how to serve passengers with heterogeneous willing payment. For the cost division problem, we propose a linear cost allocating formula for ridesharing passengers which combines cost for direct travel and discount for additional distance. To address heterogeneity of passengers' willing payments, we propose and study several auction based protocols to elicit passenger willing payment. Among auction based protocol schemes, the meter rate bidding protocol is shown as a suitable scheme for our Last Mile problem.

4.1 Budget balance constraint

The Cost allocation problem needs to take into consideration the perspective of the taxi driver and the passengers. In this thesis, we consider solutions in which taxi drivers will get at least the amount as usual taxi service to run assigned route and this amount of money is directly paid by passengers. In the other words, the routing problem is constrained by the following budget balance condition:

$$\sum_{i \in S} p_i \geq \sum_{i \in N} \sum_{j \in N} x_{ij} t_{ij}, \tag{4.1}$$

in which $\sum_{i \in S} p_i$ is the sum of payment of served passenger. This total payment is at least total travel distance of taxi fleet $T = \sum_{i \in N} \sum_{j \in N} x_{ij} t_{ij}$ in the RHS. For the sake of simplicity, in this paper we consider simple linear cost function to interpret distance traveled T into revenue with 1\$ for 1 travel distance. Other fare formula in real implementation can be scaled by linear transforms.

In our routing model, since we simplify the taxi index k, there is no direct translation from total payment to single taxi payment. Instead, we need a separate phase to derive the single taxi portion c_k directly allocated based on travel distance of taxi k.

Notice that this assumption of budget balance is flexible to be adapted to other settings with monetary subsidy from the policy maker.

4.2 Cost sharing formula

A ride-sharing trip (or simply trip) is made up of a set of passengers assigned to a particular vehicle. The direct distance of a passenger refers to the distance the passenger will incur if he travels alone on the vehicle. Given a fixed trip comprising a number of passengers, the cost sharing problem is the problem of allocating cost (payment) to the passengers on this trip.

A usual method in some situations is to minimize total travel distance T and charge passenger proportionally to their direct distances [5]: $p_i = T \times \frac{s_i}{\sum_h s_h}$, where p_i and s_i are the payment and direct travel distance of passenger we respectively, and the denominator is the sum over all served passengers' direct travel distances. A flaw in this proportional cost formula is disregard of discount for extra travel distance, as a result of sharing the trip with other passengers. It is unfair (except for the first passenger arriving home) that some passengers travel longer than their direct distance due to the detour to serve the precedent passengers, but they do not receive any additional discount for this detour. Indeed, this extra travel distance makes ridesharing less attractive than normal taxi mode [22].

In order to compensate for this extra travel distance, we propose a new fare structure which gives passengers discounts that are proportional to their extra travel distance. Let m denote meter rate i.e. fare per unit of direct travel distance; B_i denotes trip travel distance of passenger i. The revised formula is given by:

$$p_i = ms_i - \Delta(B_i - s_i). \tag{4.2}$$

In the first term on the RHS ms_i is the direct cost, which is proportional to direct distance s_i to passenger i's destination. In the second term Δ is the discount rate for each unit of additional travel distance which is $(B_i - s_i)$, and B_i is the actual travel distance in the trip.

By this payment formula (4.2), the budget balance constraint (4.1) can be rewritten as

$$\sum_{i \in S} ms_i - \Delta(B_i - s_i) \geq \sum_{i \in N} \sum_{j \in N} x_{ij} t_{ij}, \tag{4.3}$$

4.3 Auction protocol

Auction protocols in ridesharing were recently introduced in [26]. In the request for a shared ride, besides the demand information on arrival time and destination, the auction protocol allows riders to specify their willing payment. The planned trip is not only dependent on the demand information but also passengers' willing payments. For example, in [26], a driver will go with the highest ranked passengers computed based on a weighted sum of willing payment and detour distance.

The cost sharing formula given in (4.2) can be directly applied to homogeneous passengers.

However, given the fact of heterogeneous passengers who could have different individual utility functions, we need to take account for their utility into our planning. A centralized solution would fail to serve these heterogeneous passengers when it charged them the fare greater than their willing payments. Moreover, it is difficult to provide passengers a "promised" fixed low meter rate in taxi sharing service a priori as the payment would be dynamically dependent on destinations of sharing passengers and availability of taxis case by case. What is needed is a real time protocol for passengers to submit their willing payment to participate in the Last Mile sharing service.

Contextualizing this on the cost sharing formula (4.2), the willing payment of passenger i is hence computed by their willing meter rate m_i and the discount rate Δ_i. However, asking a passenger to specify these two values is quite impractical in an auction setting as the cost could be minimized either by decreasing meter rate m or by increasing the discount rate Δ. To avoid ambiguity, we consider the auction protocols in which passenger only bid on 1 value. In this section, we propose and discuss on 2 possibilities: passengers could submit either required discount rate Δ or meter rate m. We show that although Δ bidding is feasible but it is inefficient to allocate the cost. It motivates to use meter rate m to charge passenger and decide the discount rate for passenger later.

Given this willing payment, passenger i's actual fare payment should then satisfy the following constraint (called the Individual Payment constraint):

$$p_i \leq m_i s_i - \Delta_i (B_i - s_i). \tag{4.4}$$

4.3.1 Discount rate bidding model

Consider a discount rate model in which meter rate is fixed to normal taxi meter rate, passengers will specify their required discount rate. A bid for passenger i is simply defined as

$$(s_i, \Delta_i), \tag{4.5}$$

in which s_i, Δ_i are respectively destination and required discount for each extra travel distance of passenger i.

Passenger i's fare p_i follows the individual and budget balance constraints

$$p_i \leq s_i - \Delta_i(B_i - s_i), \tag{4.6}$$

$$\sum_{i \in \mathbf{S}} p_i = T, \tag{4.7}$$

We consider 2 models to calculate the cost via discount rate

1. Charge passenger directly by $p_i = s_i - \Delta_i(B_i - s_i)$. A single discount rate value is determined for all passengers. To maximize compensation for additional travel distance inconvenience, the discount rate Δ can be raised whenever the money is still enough to run the service

$$\sum_{i \in \mathbf{S}} p_i^1 = \sum_{i \in \mathbf{S}} s_i - \Delta(B_i - s_i) \geq T. \tag{4.8}$$

2. We adapt proportional cost formula for discount rate model. Given a priori discount rate Δ, the cost is first calculated by proportional formula then discounted by additional travel distance.

   Recall the proportional cost without discount part for passenger i is

$$p_i = \frac{s_i}{\sum_{j \in \mathbf{S}}} T. \tag{4.9}$$

   However, with respect to discount for additional travel distance, we need to adjust passengers fares by increasing fare of non-suffered passengers to compensate suffered passengers. Specifically we maintain a discount budget to spend on compensation aside from trip budget T. The before-discount portion of passenger i now is

$$p_i^{'} = \frac{s_i}{\sum_{j \in \mathbf{S}}} (T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)). \tag{4.10}$$

$T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)$ is the combination of trip budget and discount budget. The final payment of passenger i is subtracted by his/her discount part

$$p_i^2 = p_i^{'} - \Delta(B_i - s_i) = \frac{s_i}{\sum_{j \in \mathbf{S}}}(T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)) - \Delta(B_i - s_i). \tag{4.11}$$

It can be shown that this formula satisfies budget balance requirement

$$\begin{aligned}
\sum_{i \in \mathbf{S}} p_i^2 &= \sum_{i \in \mathbf{S}} [\frac{s_i}{\sum_{j \in \mathbf{S}}}(T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)) - \Delta(B_i - s_i)] && (4.12)\\
&= \sum_{i \in \mathbf{S}} [\frac{s_i}{\sum_{j \in \mathbf{S}}}(T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)) - \sum_{i \in \mathbf{S}} \Delta(B_i - s_i) && (4.13)\\
&= (T + \sum_{j \in \mathbf{S}} \Delta(B_j - s_j)) - \sum_{i \in \mathbf{S}} \Delta(B_i - s_i) && (4.14)\\
&= T. && (4.15)
\end{aligned}$$

Drawback of discount rate

The main drawback of discount rate model is that it only takes account for inconvenience of suffered riders but not specify a fair division for sharers. As a result, it could end up in extremis by which a suffered passenger can get incentivized by most of cost savings in ridesharing and leave nothing to non-suffered passenger.

Let us look at an example illustrating the weakness of the 2 methods suggested above. This example is shown in Figure 4.1. In this example, there are only 2 passengers A and B with the distance Depot-A:5, A-B:1, Depot-B:5.5. The solid line represents the planned trip, in which passenger A is the first served passenger and passenger B is the second one with 0.5 extra travel distance.

For the first formula (4.8) we need to increase $\Delta = 9$

$$\begin{aligned}
p_A^1 &= 5 - 9 \times 0 = 5 && (4.16)\\
p_B^1 &= 5.5 - 9 \times 0.5 = 1. && (4.17)
\end{aligned}$$

Although it does not violate any individual constraints with any $\Delta_i \leq 8$, it is unfair as first passenger A has to pay most of the trip, passenger B gets most of the monetary saving and only needs to pay an amount.

For the second formula (4.11), there are several Δ values which can satisfy individual constraints

$\Delta = 1$ satisfies any $\Delta_i \leq 6$

$$p_A^2 = (6 + 1 \times 0.5)\frac{5}{5 + 5.5} - 8 \times 0 = 3.0952 < 5 - 6 \times 0 \tag{4.18}$$

$$p_B^2 = (6 + 1 \times 0.5)\frac{5.5}{5 + 5.5} - 1 \times 0.5 = 2.9048 < 5.5 - 6 \times 0.5 \tag{4.19}$$

...

$\Delta = 9$ satisfies any $\Delta_i \leq 9$

$$p_A^2 = (6 + 9 \times 0.5)\frac{5}{5 + 5.5} - 9 \times 0 = 5 \leq 5 - 9 \times 0 \tag{4.20}$$

$$p_B^2 = (6 + 9 \times 0.5)\frac{5.5}{5 + 5.5} - 9 \times 0.5 = 1 \leq 5.5 - 9 \times 0.5. \tag{4.21}$$

This formula also can produce unfair solution for passenger A.

The reason for iniquity of above 2 discount rate models is that they can not limit discount rate value in a reasonable range. As a result, they are not fairly distributed the service cost to all passengers. In the next section, by meter rate bidding model, we will see that meter rate model does not experience such problem.

4.3.2 Meter rate bidding model

We observe that meter rate is the reasonable upper bound to cap discount rate, by which passengers will not be discounted for each unit of extra travel distance more than their willing

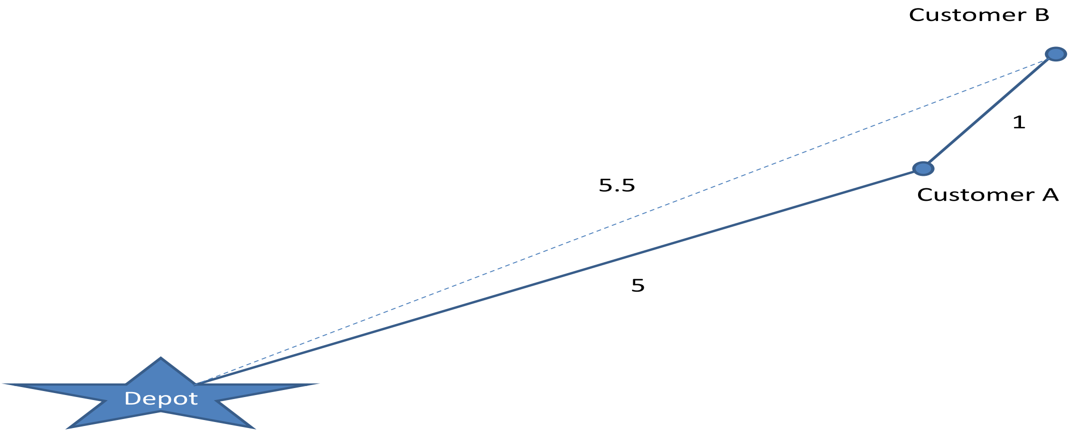


Figure 4.1: An example of routing.

payment on meter rate. With respect to this upper bound, we propose a meter rate bidding model, where discount rate and meter rate are equal. Intuitively speaking, if passenger i is willing to pay $1\$$ for each of his/her direct travel distance, he/she will get discounted by $1\$$ for each of his/her extra travel distance. In common sense, to use a better quality service, corresponding to higher discount rate, you need to pay more and vice versa.

As we will show later, this simple model has the property that the minimum fare for a passenger is easily computed by minimizing meter rate (against all other passengers)

Theorem 1. *A passengers fare is directly proportional to its meter rate.*

Proof. It follows the algebra transformation fact that

$$ms_i - m(B_i - s_i) < m^{'}s_i - m^{'}(B_i - s_i) \tag{4.22}$$

is equivalent to

$$m(2s_i - B_i) < m^{'}(2s_i - B_i) \tag{4.23}$$

when $m < m^{'}$. Notice that passenger needs to pay a positive cost to use the service, in other words $2s_i - B_i > 0$. □

As a corollary of this theorem, the fare is guaranteed to be competitive, and minimizing meter rate is dominant choice.

In regard to previous example in Figure 4.1, given optimal routing from Depot to A then B, consider one possibility to charge all passengers with the same meter rate, the meter rate model solution is

$$[m \times 5.5 - m \times (6 - 5.5)] + m \times 5 = 6 \Rightarrow m = \frac{6}{10} \tag{4.24}$$

$$p_A = 0.6 \times 5 = 3 \tag{4.25}$$

$$p_B = 0.6 \times 5.5 - 0.6 \times (6 - 5.5) = 3. \tag{4.26}$$

The meter rate solution is related to the minimizing travel distance and proportionally allocated cost solution. We notice that in budget balance situation $m = \frac{T}{S-(B-S)}$, in which T is trip travel distance, S is total direct distance demand, B is total individual trip travel distance. Hence minimizing m is equivalent to minimizing trip travel distance and total individual extra travel distance.

Chapter 5

Incentive Compatible Mechanisms

In this chapter, we study and propose incentive compatible mechanisms. We first define our mechanism as a routing and cost allocating solver. By a specific example, we show that incentive compatibility and optimality may be mutually exclusive. It also implies that designing incentive compatible mechanism is not trivial.

Based on auction based protocol, we propose 3 incentive compatible mechanisms:

1. Top-down mechanism is presented as a baseline when we compare incentive mechanisms. It expands feasible set of served passengers by including passengers with the order from high willing payment to low one. The fares of served passengers are computed by a mathematical program to minimize meter rate. Serving procedure is cut off at position of the first passenger fails to be added in the set. Meter rate bid value of this first unserved passenger will be used as lower bound in optimizing model. The notion of first unserved passenger is corresponding to notion of the second highest price passenger in classical second price auction.

2. Bottom-up mechanism is similar to Top-down mechanism in principle of giving precedence to serve passengers with high willing payment first. The difference is instead of checking servability from passengers with higher payment to passenger with lower payment, we will check the feasibility of solution to serve a maximal number of passengers

by a minimal charged lowest meter rate. At each iteration, if there is a violation of individual constraint (4.4) or $m^* > m_i$ for served passenger i, we will dismiss the worst violated passenger and solve again, otherwise we output the solution. Bottom-up mechanism is promising to solve more passengers than Top-down mechanism.

3. A modified version of raising cost procedure in Last Mile problem is proposed. It is different from the normal Raising-Cost in set covering and facility location games when instead of raising gross payment, we raise the meter rate. To speed up the algorithm, we replace the Raising-Cost by an equivalent procedure to repeatedly solve a mathematical program. The routing and cost allocation are essential in scheduling part of our Raising cost mechanism.

All of these 3 mechanisms can be decomposed into 2 phases

- In phase 1, we solve a routing problem without individual payment constraints. Only geography information of passengers' destination are considered.
- In phase 2, after solving the routing problem, we separately check if there is any violation of individual constraints of willing payment in outputted solution. If there is, we eliminate passengers in violated individual constraints and repeat the procedure from phase 1.

By following this 2 phases principle, our proposed mechanisms are proved to be incentive compatible.

5.1 Ridesharing mechanism

Our Last Mile ridesharing system is illustrated by diagram in figure 5.1. Before arriving at the hub, a batch of passengers will send their respective requests containing destination s_i and willing payment rate m_i. In addition, the number of available taxis needs to be updated.

Based on passenger demand and taxi supply, the system computes a routing and cost allocating solution. Each passenger will be informed with their trip B_i and payment p_i from the solution. Each taxi will be informed with assigned riders, routing solution and corresponding payments. To engage the payments from passengers to taxis, we can develop payment methods directly linking accounts of passengers and drivers to the center. After the confirmation of shared riders, the money will be debitted from riders' accounts and transferred to drivers' accounts. The

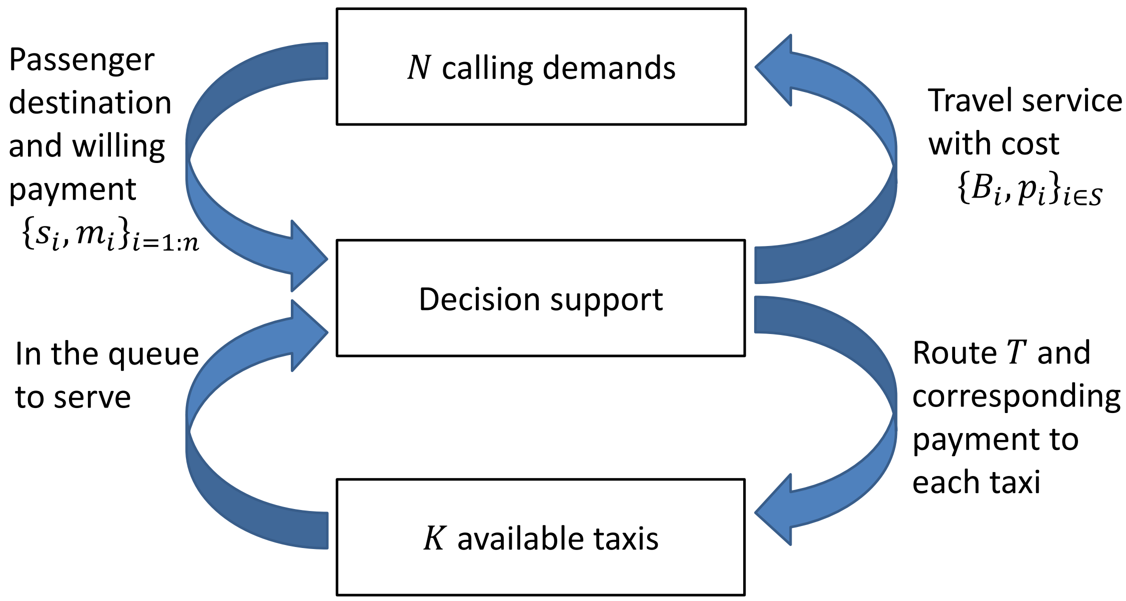


Figure 5.1: Ridesharing mechanism.

problem studied in this thesis is to develop a system to provide the routing and cost allocating solution. Specifically we want to develop mechanism defined as follows.

Definition 1. *A mechanism in Last Mile vehicle sharing is a mapping f from passenger demand set* $N = (s_i, m_i)$ *and available vehicle set* K *to routing solution* S *and vector or payment* (B_i, p_i)

$$f : (s_1, m_1) \times \ldots \times (s_n, m_n) \to (B_1, p_1) \times \ldots \times (B_n, p_n). \tag{5.1}$$

Because of budget balance constraint, each taxi driver is guaranteed to be paid by at least the amount charged for a normal (non-shared) taxi service.

In our bidding protocol, a passenger submits the information of destination s_i and willing payment rate m_i. We assume passengers will not lie on their destinations. But they can manipulate their willing payment value to gain benefit.

Definition 2. *A mechanism is incentive compatible if passengers have no incentive to bid any willing payment m_i dishonestly.*

5.2 Optimal number of served passengers mechanism

Given willing payments of participants, we propose an optimal model to provide an optimistic solution, i.e. the maximum number of passengers we can serve. Based on model, we show an example where optimality and compatibility can not happen at the same time, which illustrates the challenge of developing incentive compatible mechanism. Furthermore, optimal solution produced by this model will be used as an optimal benchmark for later experimental evaluation.

In the optimal model, passengers are charged directly by their willing payment, in other words the charged rate for passenger i, $\bar{m}_i$ is equal to his/her submitted rate m_i.

The mathematical model for this is as follows:

$$\max \sum_{i,j \in N} x_{ij} + \frac{1}{M}(\sum_{i,j \in N} x_{ij} s_j) \tag{5.2}$$

subject to

$$\sum_{j \in N} \sum_{i \in N} x_{ij}[m_j s_j - m_j(B_j - s_j)] \geq \sum_{i \in N} \sum_{j \in N} x_{ij} t_{ij}, \tag{5.3}$$

from constraint (3.2) to constraint (3.14) in routing model

The first term $\sum_{i,j \in N} x_{ij}$ in objective function is number of served passengers. With the big constant M, second part $\sum_{i,j \in N} x_{ij} s_j$ is the bias term for maximizing total direct distance of served demand.

Untruthfulness of Optimal Served Number Solution

Although above model gains the maximum efficiency in serving passengers given their willing payments, it does not ensure a truthful protocol as some passenger could lie on their willing payments to pay less. An example of strategic manipulation is shown in Figure 5.2. The distance matrix is presented in Table 5.1

Given 1 available vehicle, a solution serving 3 passengers would charge each passenger an

Distance	Depot	A	B	C
Depot		5	9	7
A			7	9
B				5

Table 5.1: This table shows some data

meter rate $m = \frac{5+7+5}{5+(9-3)+(7-10)}$, which is greater than meter rate $m^{'} = \frac{5+7}{5+(9-3)}$ of solution serving only 2 passengers A and B. To gain more benefit, passenger B could lower his willing payment on meter rate as he knows that he has advantage by bias term in objective function for the furthest destination.

5.3 Top-down mechanism

In the Top-down mechanism and its variant Bottom-up mechanism, all served passengers are charged by the same meter rate m, which is different from Raising-Cost procedure where the charged meter rate $\bar{m}_i$ can be differentiated.

Firstly, we order passengers by their submit values in a preprocessing step. We prioritize passengers with high willing payments on meter rate to be served first. Wining players should have higher willing payments than losing players.

Top-down mechanism initializes with the smallest set of feasible solutions of the high will-

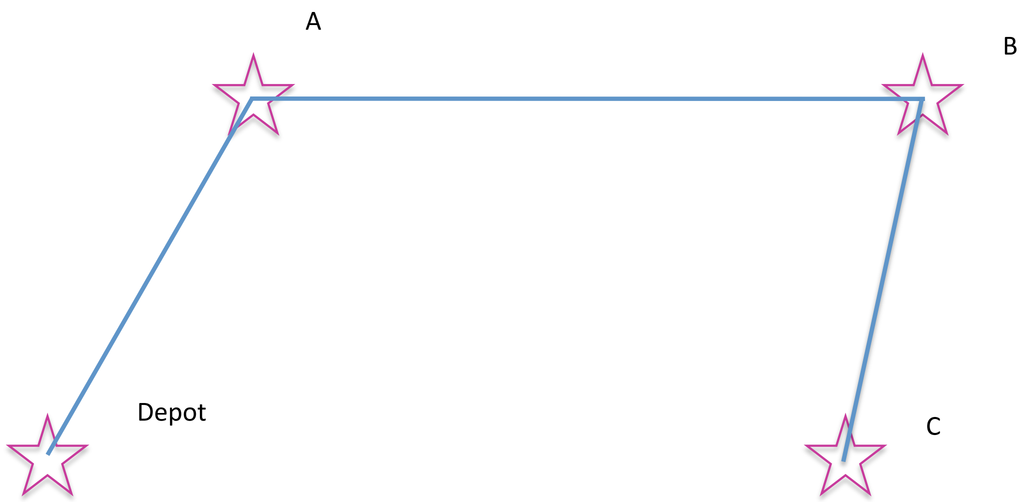


Figure 5.2: An example of Untruthful Bidding.

ing payment passengers. Then at each step, it includes an additional passenger with the highest willing payment in the remaining set. To find the routing solution for a set of passengers, Top-down solves a mathematical program minimizing the charged meter rate m. The meter rate solution is bounded by the bid value of first unserved passenger, i.e. the highest bid value in the remaining set. This lower bound plays a role to protect the mechanism from overbidding. The highest bid value of passengers in the unserved set is equivalent to the second highest bid in second price auction.

The detail of the Top-down mechanism is described by Algorithm 2. In the first phase from line 23 to line 29, it repeatedly expands the potential set from empty until a feasible solution is found. In the second phase from line 31 to line 37, it continues to expand the potential set until there is no feasible solution for expanded set. Notice that an expanding procedure is carried out by decreasing order of m_i. At each iteration, after adding the next highest willing payment passenger i into the served passengers set $S_i = S_{i-1}$ and set the lower bound $m_l = m_{i+1}$ by the first passenger outside of S_i, $Solve_1(S_i, m_l)$ will return solution m^* for S_i with constraint $m \leq m_i$ by solving a mathematical program $Solve_1$ with $\mathbf{S} = \mathbf{S}_i$:

$$Solve_1(N, m_l) : m^* = \min m \tag{5.4}$$

$$\sum_{i,j \in \mathbf{S}} x_{ij} t_{ij} \leq \sum_{i,j \in \mathbf{S}} [m s_j - m(B_j - s_j)] \tag{5.5}$$

$$\sum_{i \in \mathbf{S}} x_{ij} = 1, \forall j \in \mathbf{S} \tag{5.6}$$

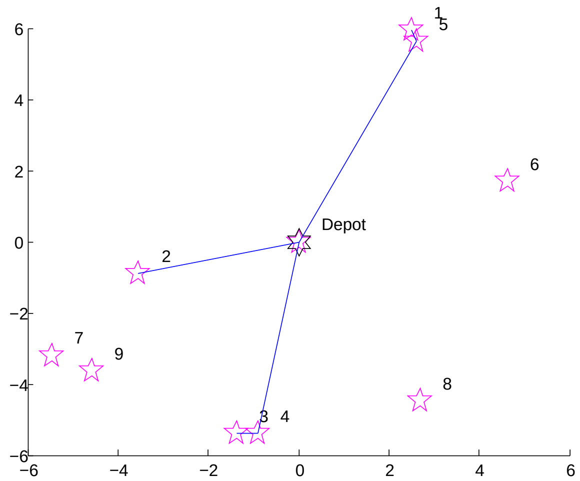
$$\sum_{i,j \in \mathbf{S}} x_{ij} = |\mathbf{S}| \tag{5.7}$$

$$m_l \leq m \leq m_i, \forall i \in \mathbf{S} \tag{5.8}$$

from constraint (3.2) to constraint (3.14) in routing model

In budget balance constraint (5.5), all served passengers are charged with the same meter rate m. Constraints (5.6) and (5.7) require to serve all passengers in the given set **S**. Constraint (5.8) indicates the m's lower bound and individual constraint checking. Note that in Top-down mechanism, individual constraint checking can be incorporated into mathematics program, which is not feasible in other incentive compatible mechanism.
m_l is value of the first unserved passenger, which is set in lines 24 and 32.
If there is no feasible solution, which is $m^* = \infty$ in line 35, i will be considered as cutoff point and we output solution of $\mathbf{S}_{i-1}$.

Figure 5.3 shows an example how Top-down mechanism works. Served passengers are $1, 2, 3, 4, 5$ as they are the 5 highest willing payment passengers that service can serve, there is no solution to serve 6 passengers from 1 to 6. We cut off at passenger 6 value because he is the first unserved passenger. m_6 will be used as the lower bound in the mathematics program to minimize m: $m_5 \geq m^* \geq m_{bound} = m_6$. In this case, the charged meter rate is minimized to be equal to lower bound $m_{bound} = m_6$. We do not charge served passengers by meter rate less than m_6 otherwise it does not guarantee the incentive compatibility.



Top Down

Customer	Willing meter rate	Served	Charged meter rate
1	0.98632646	✓	0.752338346
2	0.975447722	✓	0.752338346
3	0.95533129	✓	0.752338346
4	0.883909306	✓	0.752338346
5	0.765337227	✓	0.752338346
6	0.752338346	-	-
7	0.750611858	-	-
8	0.702040408	-	-
9	0.688559958	-	-

Figure 5.3: An example of Top-down mechanism.

Algorithm 2: Top-down algorithm

input : Set n passengers $\mathbf{N} = \{(s_i, \Delta_i)\}_i$
output: Set $\mathbf{S}$ of served passengers and their costs

Initialize: $\mathbf{S}_0 \leftarrow \emptyset$;
for $i \leftarrow 1$ **to** n **do**
    $m_l = m_{i+1}$;
    $S_i = S_{i-1} \cup i$;
    $m^* = Solve_1(S_i, m_l)$;
    if $m^* \neq \infty$ **then**
        break;
    end
end
for $j = i + 1$ **to** n **do**
    $m_l = m_{j+1}$;
    $S_j = S_{j-1} \cup j$;
    $m^* = Solve_1(S_j, m_l)$;
    if $m^* = \infty$ **then**
        output $\mathbf{S} = S_{j-1}$;
    end
end

Theorem 2. *The Top-down mechanism is incentive compatible.*

Figure 5.4: Manipulating bidding values

Proof. For unserved passengers, underbidding $m_{underbidding} < m_{passenger} \leq m_{bound}$ would not help as they are not served because of too low willing payment. Overbidding of unserved passengers $m_{bound} \leq m_{overbidding}$ could let passenger into the served passengers set but at the same time it would run a risk of overpayment: $m_{passenger} \leq m_{bound} < m^* \leq m_{overbidding}$.
For served passengers, overbidding obviously does not change the solution m^* as they are still above cutoff position. Similarly, underbidding of served passengers does not change the solution if $m^* \leq m_{underbidding} < m_{passenger}$. In additional, underbidding could dismiss passenger from service when their bidding is lower than the threshold of feasibility. □

5.4 Bottom-up mechanism

This mechanism is similar to Top-down in that it seeks to serve high willing payment passengers. But instead of expanding the served passengers by high willing payment order from an empty set, we gradually eliminate low willing payment passenger who is the worst violated passenger in individual constraint, which has the lowest bidding willing payment, and solve problem again.

The Bottom-up mechanism is based on a simple principle: include all demands initially, compute the assignment considering the destinations but not the meter rates. If a feasible cost allocation cannot be found from the particular assignment, the lowest paying passenger (in terms of submitted meter rate) will be eliminated from consideration and the assignment is recomputed. The process continues until we identify a feasible combination of assignment and cost allocation. For this reason, we call it the Bottom-up mechanism.

Algorithm 3: The Bottom-up algorithm.

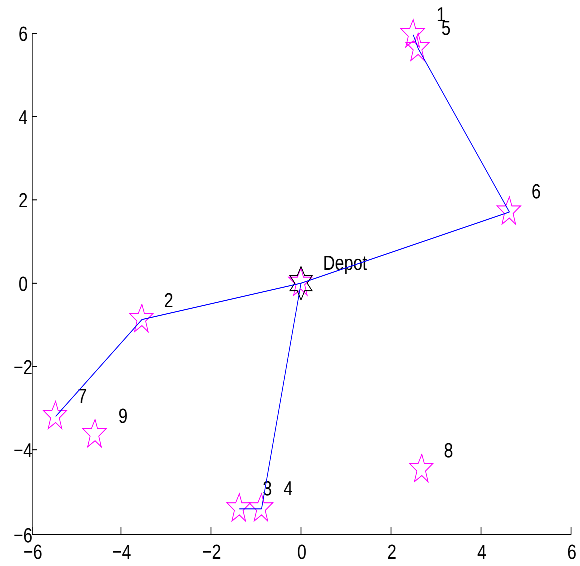
Input: $(N, \{(s_i, m_i)\}_{i \in N})$
Output: (S, m^*)
39 $S \leftarrow \emptyset$, $m_l = 0$
40 **while** $N \neq \emptyset$ **do**
41     $(m^*, S) = Solve_2(N, m_l)$
42   **if** $\min_{i \in S} m_i < m^*$ **then**
43         $m_l = \min_{i \in S} m_i$
44         $N \leftarrow N \setminus \{\arg\min_{i \in S} m_i\}$
45   **else**
46         $N \leftarrow \emptyset$
47   **end**
48 **end**
49 **return** (S, m^*)

Assuming that the auctioneer has collected the set of all bids $\{(s_i, m_i)\}_{i \in N}$ from all passengers in set N. The Bottom-up clearing process is illustrated in Algorithm 3. The mechanism first calls the mathematical program $Solve_2$ to obtain a candidate assignment. For each obtained assignment, we check whether individual payment constraint is violated for the lowest paying passenger (line 42). If a violation is detected, the lowest paying passenger is removed from set N, and the lower bound on the meter rate (m_l) is also updated (line 43). The above process repeats until an assignment that satisfies all individual payment constraints is found. At termination, the mechanism reports the set of passengers to be served (including their cluster and service orders) and the universal meter rate (m^*) to be paid by all served passengers.

$$Solve_2(N, m_l) : m^* = \min m \tag{5.9}$$

subject to

$$\sum_{i,j \in N} x_{ij}\, t_{ij} \leq \sum_{i,j \in N} x_{ij} \left(m s_j - m(B_j - s_j) \right), \tag{5.10}$$



Bottom Up

Customer	Willing meter rate	Served	Charged meter rate
1	0.98632646	✓	0.705940955
2	0.975447722	✓	0.705940955
3	0.95533129	✓	0.705940955
4	0.883909306	✓	0.705940955
5	0.765337227	✓	0.705940955
6	0.752338346	✓	0.705940955
7	0.750611858	✓	0.705940955
8	0.702040408	-	-
9	0.688559958	-	-

Figure 5.5: An example of Bottom-Up Mechanism

$$\sum_{i,j\in N} x_{ij} = \min\{|N|, |K| \times Q\}, \tag{5.11}$$

$$m \geq m_l, \tag{5.12}$$

together with constraints (3.2) – (3.14) *in routing model.*

An example of Bottom-up algorithm is shown in Figure 5.5. The mechanism begin with trying to serve $\min\{\mathbf{N}, \mathbf{K} \times K\} = 9$ passengers. There is no feasible solution, therefore we eliminate lowest willing payment passenger 9 and resolve again. In second iteration, a solution to solve 8 passengers is found but it violates individual constraint of passenger 8, therefore we need to dismiss passenger 8 from the potential set. Finally, there is no violation in solution to served 7 passengers, therefore we terminate and output the solution.

Note that the mechanism terminates when we eliminated all passengers in **N** or we find a non-violated solution.

Theorem 3. *The Bottom-up mechanism is incentive compatible.*

Proof. A passenger's bid contains two parts: the destination and the desired meter rate. Since this is a shared door-to-door service, a passenger should not cheat on the destination, and thus our analysis will focus only on the meter rate.

In the first phase of the mechanism, mathematics program $Solve_2$ is solved and selected destinations are assigned to clusters with service orders. If a passenger is not chosen in the first phase, truth telling is not an issue as this passenger's bid will not change the outcome. Suppose the passenger is selected in phase one, but eliminated in phase two because $m_i < m^*$, he will not want to raise his bid, since if he manages to stay on by raising bid, the new m^*, which originally is m_i, will be higher than m_i . The lower bound guarantees that he cannot be served if any bid higher than his value is eliminated. Lowering his bid in this case makes no difference.

Finally, if a passenger stays on in both phases one and two, raising the bid makes no difference, since he is already chosen, and he will be asked to pay m^* regardless of his own bid. Lowering the bid is also not desirable, since the charged rate, m^*, is the minimized meter rate, and if the passenger wants to benefit from the lower cost, he has to bid less than m^*, but this will result in him being eliminated instead. Therefore, in all cases, bid the true meter rate will not cause harm, and in two out of four cases, the bidder will perform strictly better than lying. □

5.5 Raising cost procedure in Last Mile vehicle routing problem

The raising-cost mechanism is assumed to receive identical inputs as the Bottom-up mechanism. The main idea of the mechanism is to select subsets of confirmed passengers iterative by gradually raising the clearing meter cost. Served passengers can thus have different meter rates. Similar clearing techniques have been utilized in constructing incentive-compatible mechanisms in other settings such as set covering and facility location game [14].

Algorithm 4 is a straightforward implementation of the raising-cost mechanism. In each iteration, the standing meter rate (that can be applied to unserved passengers) is increased by a small amount ϵ, and passengers having meter rates lower than m^* are then removed from

Algorithm 4: The raising-cost mechanism.

Input: $(N, \{(s_i, m_i)\}_{i \in N}, \epsilon)$

Output: $(S, \{\bar{m}_i\}_{i \in S})$

$t = 0, S_t \leftarrow \emptyset, N_t \leftarrow N, m^t = 0$

while $N \neq \emptyset$ **do**

  $m^t = m^t + \epsilon$

  for $i \in N$ **do**

    if $m^t > m_i$ **then**

      $N_t \leftarrow N_t \setminus i$

    end

  end

  $S_{t+1} = Solve_3(S_t, N_t, m^t)$

  if $S_{t+1} \neq \emptyset$ **then**

    for $i \in S_{t+1} \cap N_t$ **do**

      $N_t \leftarrow N_t \setminus i$

      $\bar{m}_i = m^t$

    end

    $N_{t+1} \leftarrow N_t$

    $t = t + 1$

  end

end

return $(S_t, \{\bar{m}_i\}_{i \in S})$

consideration. Given the sets of served and unserved passengers and the current meter rate m^*, $Solve_3$ is then invoked to find the set of served passengers. For new passengers served in this iteration, their meter rates ($\bar{m}_i$) are fixed at m^*. Note that we only need a feasible solution from $Solve_3$, thus it is a constraint satisfaction problem.

$$Solve_3(S, N, m^*) :$$

$$\sum_{i,j \in N \cup S} x_{ij} t_{ij} \leq \sum_{i \in N \cup S} \Bigg[\sum_{j \in N} x_{ij} \left(m^* s_j - m^*(B_j - s_j) \right) + \sum_{k \in S} x_{ik} \left(\bar{m}_k s_k - \bar{m}_k (B_k - s_k) \right) \Bigg], \tag{5.13}$$

$$\sum_{i \in N \cup S} x_{ij} = 1, \forall j \in S, \tag{5.14}$$

$$\sum_{i \in N \cup S} \sum_{j \in N} x_{ij} \geq 1, \tag{5.15}$$

together with constraints (3.2) – (3.14) *in routing model.*

In Raising-Cost algorithm 4, we have to continuously raise and check the meter rate by individual payment constraints until we find a new solution or any violation. If there is a violation in line 54, it will be removed from the remaining potential set N_t in line 55. $Solve_3(S_t, N_t, m*)$ in line 58 will find and return a solution of constraint satisfaction problem with constraints (5.17) to (5.22) to find an additional passenger in set N_t of remaining unserved passengers, if there is a solution found, charged meter rate of new served passengers will be fixed by current m^*. A passenger is removed from remain passengers set N_{t+1} if his/her bidding value is violated (line 55) or he/she is the new served passenger (line 61). If there is no solution found, we increase the meter rate m^* and solve again.

Instead of gradually raising the cost by small increment ϵ, due to special structure of vehicle routing problem, we can "jump" through a bigger increment from current solution to the next solution by a mathematical program to find the minimum m to serve at least 1 passenger in remaining set **N**. The number of iterations therefore reduced from $O(1/\epsilon)$ to N.

The modified algorithm for Last Mile problem is described by Algorithm 5.

The main difference is instead of gradually Raising-Cost by many intermediate steps, in line 74, $Solve_4(S_t, N_t, m_l)$ solves a mathematical model to find the minimum m^* for remaining passengers N_t who are still unserved. If a new routing solution is found, we need to check whether there is any violation for the new served passengers in line 77. A passenger is removed from remain passengers set N_{t+1} if his/her bidding value is violated (line 78) or he/she is the new served passenger (line 81).

$$Solve_4(S, N, m_l) : m^* = \min m \tag{5.16}$$

subject to

$$\sum_{i,j \in \mathbf{N} \cup \mathbf{S}} x_{ij} t_{ij} \leq \sum_{j \in \mathbf{N}} \sum_{i \in \mathbf{N} \cup \mathbf{S}} x_{ij} [m s_j - m(B_j - s_j)] \tag{5.17}$$

$$+ \sum_{j \in \mathbf{S}} \sum_{i \in \mathbf{N} \cup \mathbf{S}} x_{ij} [\bar{m}_j s_j - \bar{m}_j (B_j - s_j)] \tag{5.18}$$

Algorithm 5: Modified Raising Cost Mechanism in taxi sharing

```
Input: Set n passengers N = {(s_i, m_i)}_i
Output: Set S of served passengers and their costs
69 t = 0;
70 S_t ← ∅;
71 N_t ← N;
72 m_l = 0;
73 while N_t ≠ ∅ do
74     (m*, S_{t+1}) = Solve_4(S_t, N_t, m_l);
75     if S_{t+1} ≠ ∅ then
76         for i ∈ S_{t+1} ∩ N_t do
77             if m_i < m* then
78                 N_t = N_t \ i;
79                 go to 73;
80             else
81                 N_t = N_t \ i;
82             end
83         end
84         N_{t+1} = N_t;
85         m_l = m*;
86         t = t + 1;
87         go to 73;
88     else
89         output S;
90     end
91 end
```

$$\sum_{i \in \mathbf{N} \cup \mathbf{S}} x_{ij} = 1, \forall j \in \mathbf{S} \tag{5.19}$$

$$m \geq m_l \tag{5.20}$$

$$\sum_{i \in \mathbf{N} \cup \mathbf{S}, j \in \mathbf{N}} x_{ij} \geq 1 \tag{5.21}$$

$$\textit{from constraint (3.2) to constraint (3.14) in routing model} \tag{5.22}$$

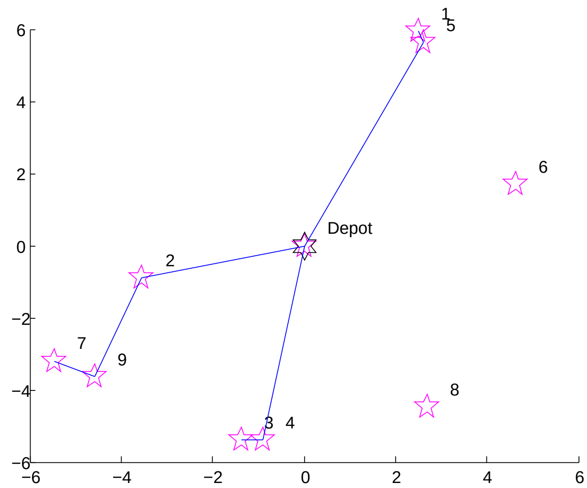
The equivalence of 2 mechanisms is proven by

Theorem 4. *Algorithm 5 returns the same solution with Algorithm 4.*

Proof. We show the proof by induction. Assume that until iteration k, solution of Algorithm 5 is still the same with solution of Algorithm 4. It is necessary that the next feasible solutions of these 2 algorithms are also identical. Denote m_k, $m_k^{'}$, m_{k+1} and $m_{k+1}^{'}$ as solutions of 4, 5 at step k, 4, 5 at step k+1 respectively. By induction hypothesis, $m_k = m_k^{'}$. By Algorithm 4, m_{k+1} will be raised therefore greater than m_k. By Algorithm 5, $m_k^{'}$ is the lower bound in next iteration solving, so $m_k^{'} < m_{k+1}^{'}$. If $m_{k+1} < m_{k+1}^{'}$, solution corresponding to $m_{k+1}^{'}$ is not the optimal one; if $m_{k+1} > m_{k+1}^{'}$ the normal Raising-Cost will found the solution of $m_{k+1}^{'}$ before m_{k+1}, both are contradiction. So $m_{k+1} = m_{k+1}^{'}$. $\square$

In Raising-Cost procedure, we maintain 2 sets of passengers at each iteration, set of served passengers **S** with their fixed meter rate $\bar{m}_i$ and set of the remaining unserved passengers **N** from which we need to find a solution minimizing m to serve at least one additional passenger. Notice that we do not maximum number of served passengers but minimize meter rate m, therefore there could be the case that only 1 additional passenger is added into served set **S** in each iteration.

An example of Raising cost mechanism is shown in Figure 5.6. The charged meter rate is raised over iterations. Solution for passengers $1, 5$ are found in the first iteration with meter rate 0.52. Then the in second iteration passengers $2, 7, 9$ are served with the higher meter rate 0.55. Finally passengers $3, 4$ are served in the third iteration with the highest meter rate 0.56.



Customer	Willing meter rate	Served Iteration	Charged meter rate
1.00	0.98	1	0.52
2.00	0.97	2	0.55
3.00	0.95	3	0.56
4.00	0.88	3	0.56
5.00	0.77	1	0.52
6.00	0.75	-	-
7.00	0.75	2	0.55
8.00	0.70	-	-
9.00	0.69	2	0.55

Figure 5.6: An example of Raising Cost Mechanism.

Again, notice that we do not include individual constraint in our mathematical program so we need to check after solving if there is any individual constraint violation for the new served passengers in set **N**. In post processing, if any individual constraint violation is found, we will dismiss all violated passengers and solve again.

Theorem 5. *The raising meter rate cost mechanism is incentive compatible.*

Proof. The incentive compatibility property of the raising-cost mechanism is inherent in its procedure. The meter rate is initialized at 0, and gradually increases at the interval of ϵ. If a passenger is not selected in the final served set, lowering his bid will not change his status. Raising his bid could delay his elimination to later iteration, and this passenger might be chosen in these additional iterations, however, it is not desirable as he will be paying meter rate higher than his tolerance.

If a passenger is selected in the final served set, raising his bid makes no difference. Let this passenger's bid be m_j, $m^t = m_j$, and let $\bar{m}_j$ be the meter rate that is actually charged. If $\bar{m}_j = m_j$, it is not desirable to lower the bid, since the passenger will be eliminated in iteration $(t-1)$. If $\bar{m}_j < m_j$, lowering the bid to the range of $[\bar{m}_j, m_j)$ makes no difference, since this passenger will still be charged $\bar{m}_j$. If the new bid is lower than $\bar{m}_j$, the passenger will be eliminated and this will not be desirable as well.

Therefore, in all cases, bidding trustfully does no harm, while deviating from the true meter rate will cause undesirable outcomes in three out of six cases. □

5.6 Optimal meter rate m^* search

The mathematical model in (5.4), (5.9) and (5.16) in fact is not linear because constraints (5.5), (5.10) and (5.18) contain the term $m \times B_j$, which is product of 2 continuous variables. The decomposition is difficult in this case. To the best of our knowledge, there is no available method to decompose product of 2 real variables in linear programming. Hence in this work, we propose an efficient hill climbing algorithm to optimize m, in which we repeatedly decrease

$m_{t+1} = m_t^* - \epsilon$ with m_t^* denoting solution in previous iteration. ϵ is the increment to move solution out of a local optimum, however it does not mean that the next solution will be improved only by ϵ.

In the Top-down mechanism, we can choose the start point for hill climbing process by the lowest bid For Bottom-up and Raising cost mechanism. Then we find the first feasible point by incremental search in bidding passenger values.

The quality of our hill climbing algorithm is determined by decrement ϵ

Theorem 6. *Solution m^* of hill climbing with decrement ϵ is a ϵ-approximation of optimal solution m^{opt}, in other words*

$$|m^* - m^{opt}| \leq \epsilon. \tag{5.23}$$

Proof. We prove the theorem by contradiction. We have m^* is the final solution of our hill climbing search. Assume $|m^{opt} - m^*| > \epsilon$, there is a solution for upper bound $m = m^* - \epsilon$, it means that the solution with meter rate m^* is not the final solution of our hill climbing search, which is a contradiction. □

To choose a heuristic replacement for nonlinear mathematical programming, firstly we observe that

Theorem 7. *Solution of the objective function $m^* = \min m = \min \frac{T}{S-(B-S)}$ is one Pareto solution of bi-objective functions $\min T$ and $\min B$*

Proof. This observation can be proved by contradiction. With a little abuse of notation, we use S as total direct distance demands of served passengers and $T(S), B(S)$ as total trip travel distance and total individual trip travel distance corresponding to that solution. If there was a solution S' dominating S^* of objective function $\min m$, we would have $T(S') \leq T(S^*)$ and $B(S') \leq B(S^*$ with at least one of them is strict inequality, then $m(S') = \frac{T(S')}{S'-(B(S')-S')} < \frac{T(S^*)}{S^*-(B(S^*)-S^*)} = m^*$, which is contradict to assumption that S^* is the optimal solution for $\min m$. □

As the decomposition of $m = \frac{T}{S-(B-S)}$ is hard, we try to find optimal m by repeatedly solving alternative mathematical programming with decreasing m. Notice solution of m^* is a Pareto set of $\min T$ and $\min B$, we heuristically use objective function $\min T + B$ which also returns a Pareto solution of $\min T$, $\min B$.

In each iteration in Meter Rate Search Algorithm 6, given a m value as the upper bound parameter from search procedure, we solve the MIP

$$A : \min \sum_{i,j \in \mathbf{N}} x_{ij} t_{ij} + \sum_{i \in \mathbf{N}} B_i \tag{5.24}$$

s.t.

constraints of normal model

Notice that if passenger i is not served, to minimize (5.24) B_i will be forced into 0.

The $m^* = \frac{T}{S-(B-S)} < m$ value found by solution of (5.24) will be used as the upper bound parameter for next iteration. If $m^* = m$, we need to decrease search bound by the decrement ϵ for the next iteration. The algorithm terminates and the best solution found is output when there is no more available solution for current upper bound meter rate m.

Algorithm 6: Meter Rate Search Algorithm

while $m > m_{lowerbound}$ **do**
    Solve A to find m^*;
    if *no solution* **then**
        output current solution found in previous iteration;
    else
        if $m^* < m$ **then**
            $m = m^*$;
        else
            $m = m - \epsilon$;
        end
    end
end

5.6.1 Hill climbing search experimental results

We carry out some test cases to verify efficiency of our algorithm. We find optimal meter rate m^* solutions for 10 instances of 20 passengers and 1 taxi with capacity 4, so the size of search space is $\binom{20}{4} = \frac{20!}{4!16!} = 4845$ with start point $m = 1$. We measure number of iterations, runtime and quality of solution with different decrement value ϵ, represented in Figure 5.7, 5.8, 5.9 respectively. We see that the average number of iterations is only 5.7 even with the small decrement value $\epsilon = 0.0001$, while to find the optimal solution $m^* = 0.41$ there are $\frac{1-0.41}{0.0001} = 5900$ intervals of $\epsilon = 0.0001$. When ϵ increases, number of iteration and runtime decrease, which is predictable. We observe after $\epsilon = 0.01$, quality of solutions reflected by meter rate value m^* are quickly reduced. Solution of $\epsilon = 0.01$ is similar to solution of $\epsilon = 0.0001$ but the runtime is better, hence in our experiments in next section, we choose the $\epsilon = 0.01$.

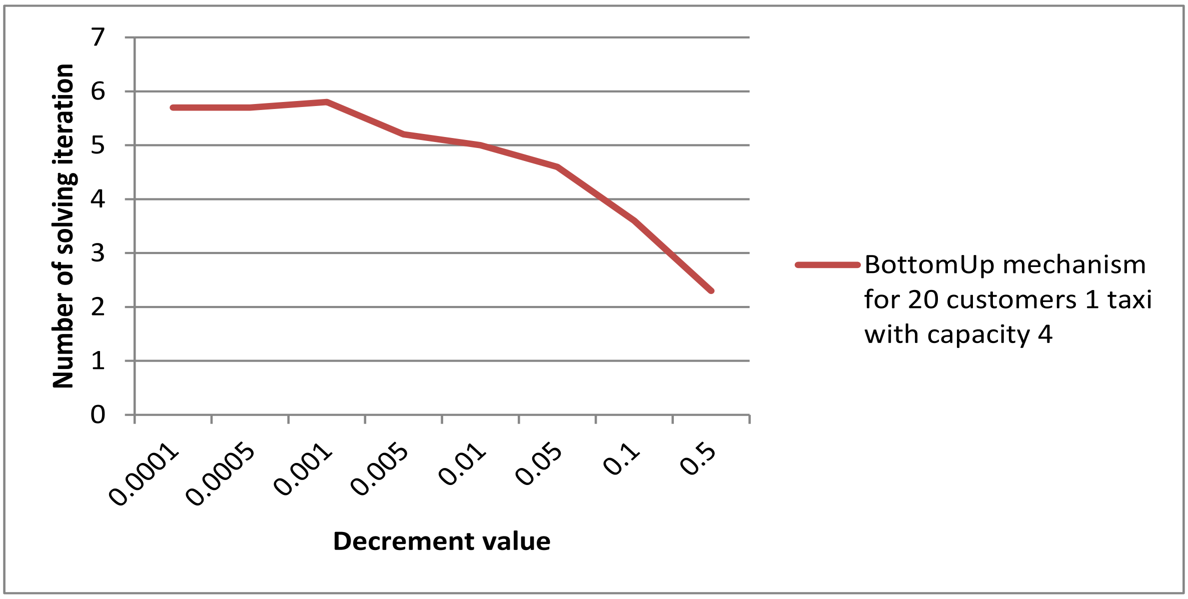


Figure 5.7: Number of iterations with varied ϵ

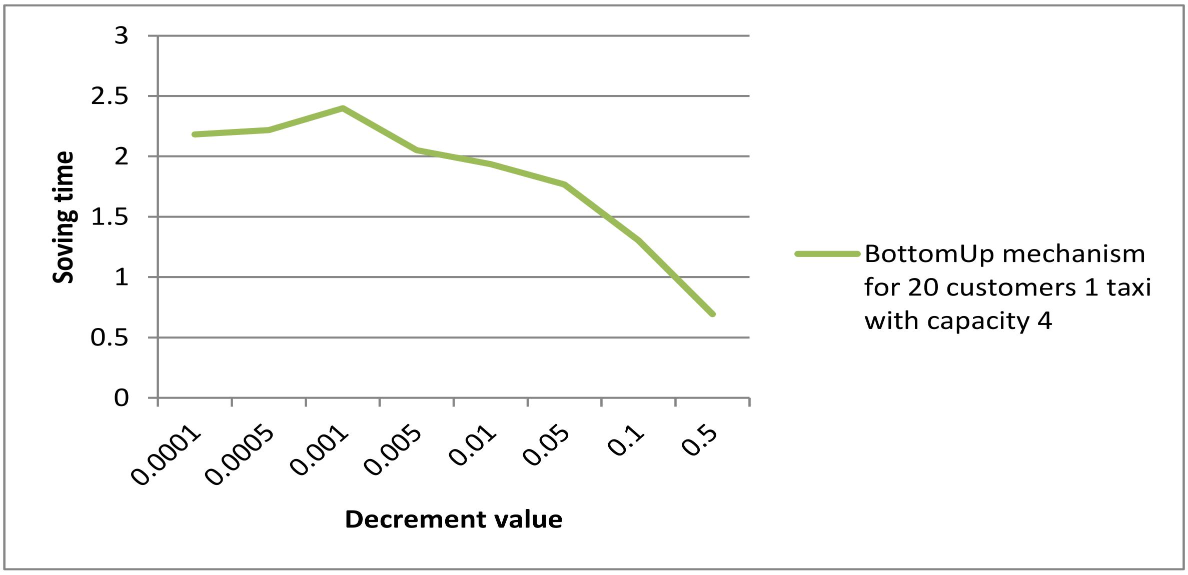


Figure 5.8: Runtime with varied ϵ

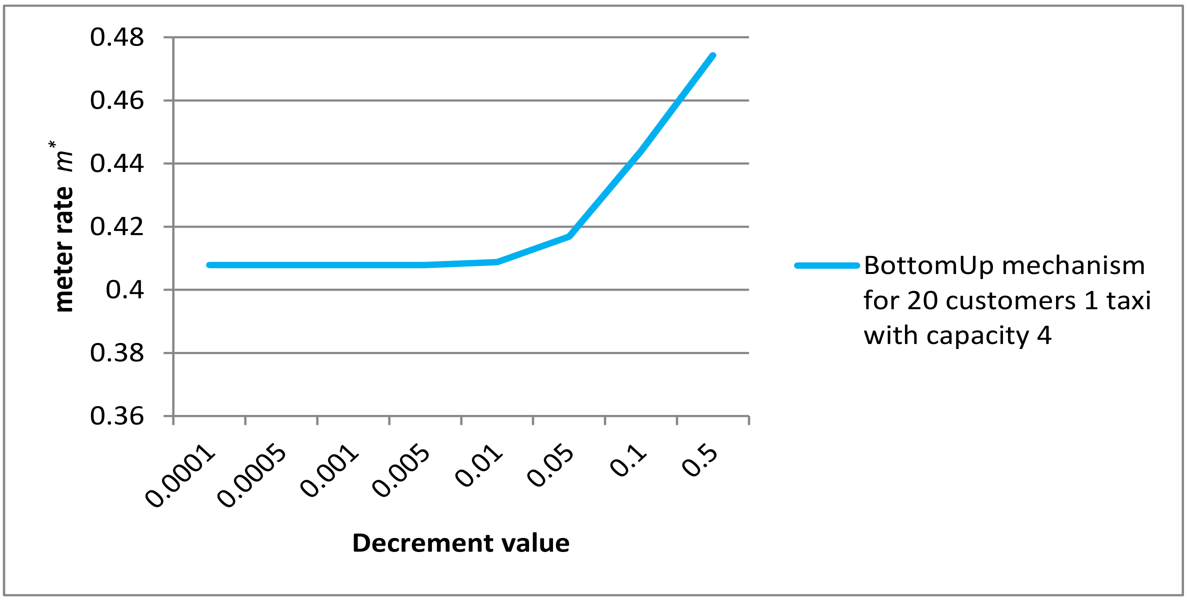


Figure 5.9: Meter rate m^* with varied ϵ

Chapter 6

Experimental Results

In this chapter, we study the effectiveness and efficiency of our proposed mechanisms by a series of experiments. Our experiments are established by 2 sets of data

- Synthetic data in which destinations are randomly generated in a square with depot as the center.
- Real data of Last Mile travel demand of passengers collected from Ang Mo Kio MRT Station in Singapore. We consider a peak hour from 6pm to 7pm and batches of passengers in time intervals of 3 minutes.

6.1 Synthetic data

In the synthetic set of data, passengers' destinations are generated in the square with edge length of 14 and depot in the center, willing payment on meter rate of each passenger query is generated uniformly in range [0.5,1]. We carry out experiment for 3 set of parameters: varying passenger numbers, varying taxi numbers and varying willing payment range. Each plotted result point is an average over solutions of 10 instances. We consider 3 metrics: number of served passengers $\sum_{i,j\in N} x_{ij}$, total direct distance demand $\sum_{i,j} x_{ij}s_j$ and total surplus, i.e. the difference between willing payment and real charged payment of served passengers $\sum_{i,j \in N} x_{ij}(m_j - \bar{m}_j) \times (2 \times s_j - B_j)$.

6.1.1 Varying number of passengers

In the first experiment set shown in Figures 6.1, 6.2, we observe results of mechanisms when the the number of passengers varies from 8 to 25. Figure 6.1 shows the number of the served passengers in solutions of different mechanisms. Figure 6.2 is the result in the total direct distance demand of served passengers. We see that results in these 2 metrics are consistent with each other in comparison of qualities of different mechanisms. The total direct distance increases along with the increment of number of served passengers. We observe when the number of passengers increases, Raising cost and Bottom-up Mechanism solutions qualities are closer to Optimal solution.

In all instances, the Top-Down mechanism runs as a baseline with the worst quality solution. When number of passengers is small, the quality of Raising cost mechanism dominates Bottom-up. When the number is large above 17 passengers, Bottom-up mechanism shows better result than Raising cost mechanism.

Figure 6.7 shows the result for total surplus of served passengers. By the result, we can see that Raising Cost by differentiating passengers' payments is the most beneficial mechanism for served passengers. Raising Cost produces the largest surplus, following is Bottom-Up and lowest is Top-down mechanism.

6.1.2 Varying number of vehicles

For second experiment, given 20 passengers we vary number of available vehicles in depot from 1 to 10 vehicles. Result for number of served passenger is shown in Figures 6.4 and result for total direct distance of served passenger is shown in Figure 6.5.

Similar to previous experiment on varying number of passengers, Top-down solution works

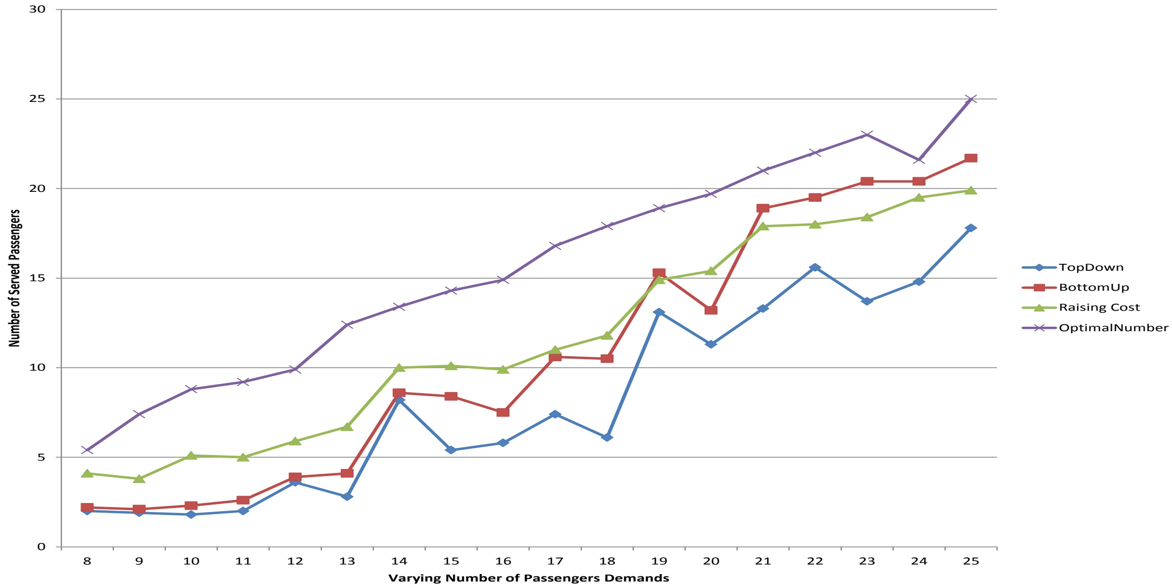


Figure 6.1: Number of served passengers: with increasing demand

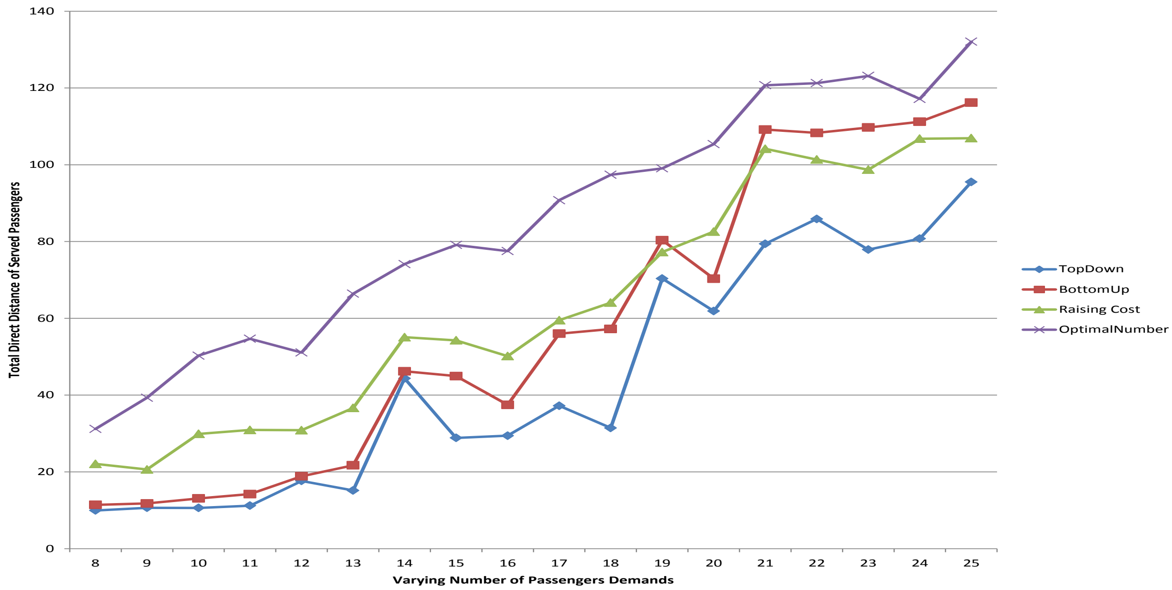


Figure 6.2: Total direct distance of served passengers: with increasing demand

as a lower bound in all cases.

The gap between incentive compatible mechanisms and optimal numbers of served passengers is widening when the number of vehicles increases. It is due to the fact that with more vehicles, the feasible solution set size, which is equal to $\binom{\mathbf{N}}{\mathbf{S}}$, decreases with more compulsiveness in serving passengers.

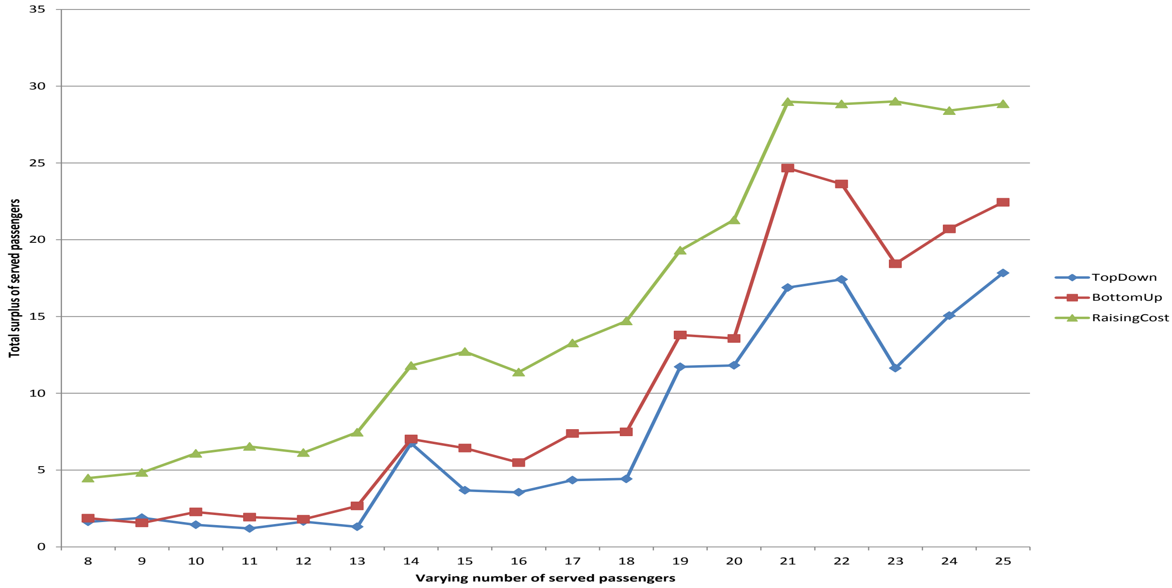


Figure 6.3: Total surplus of served passengers: with increasing demand

In respect to surplus values, when the number of vehicles is greater than 7, Bottom-Up and Top-down solution are closer to the top solution produced by RaisingCost mechanism.

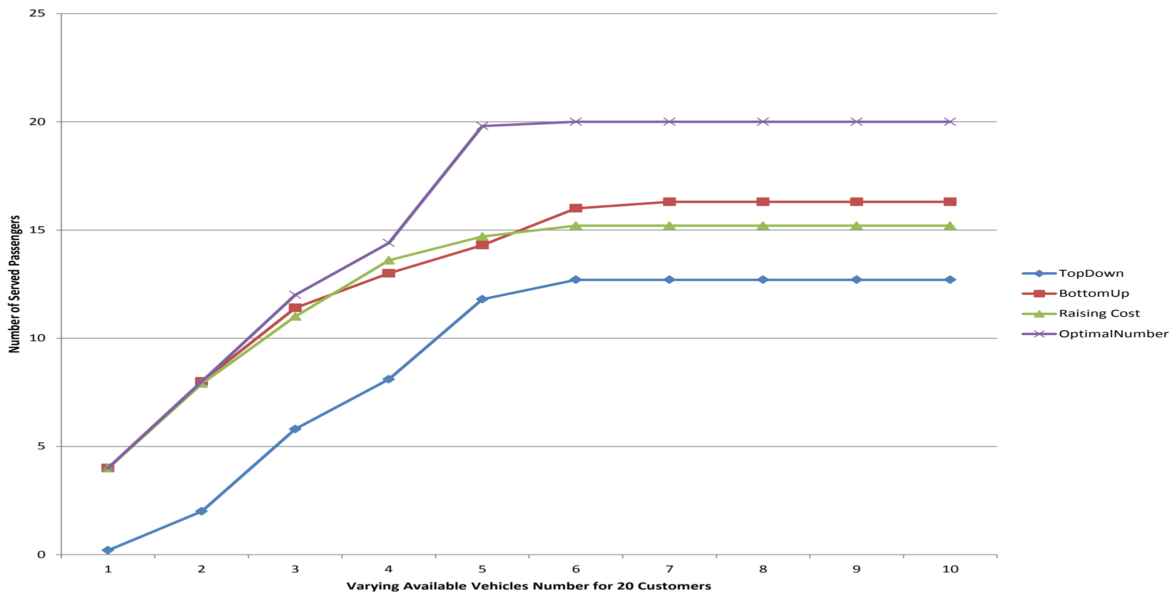


Figure 6.4: Number of served passengers: with increasing demand: with increasing fleet size

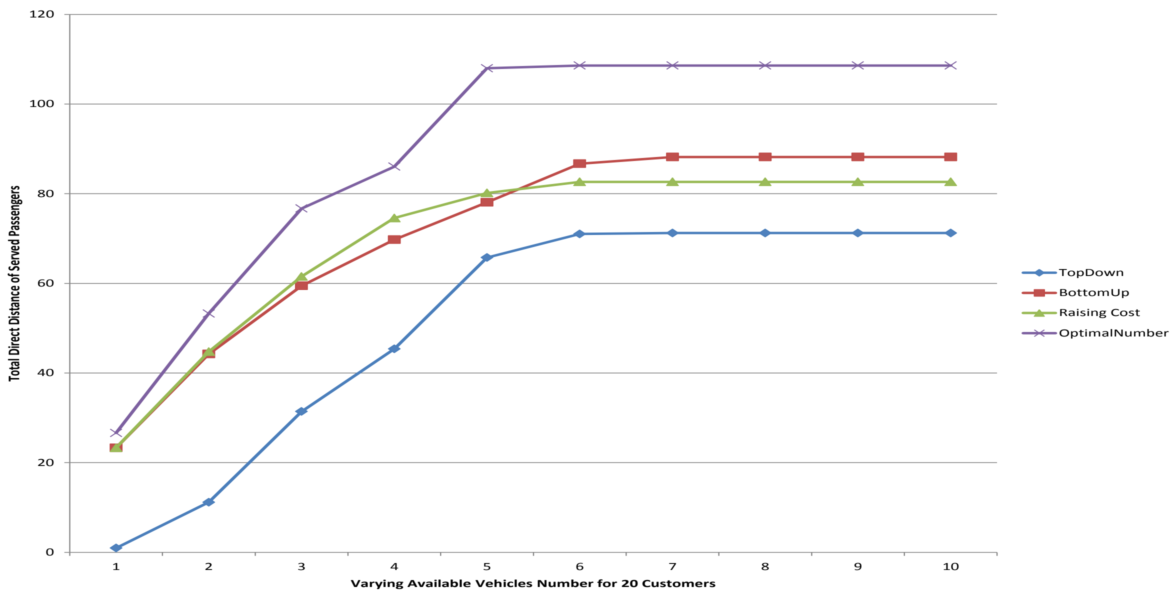


Figure 6.5: Total direct distance of served passengers: with increasing fleet size

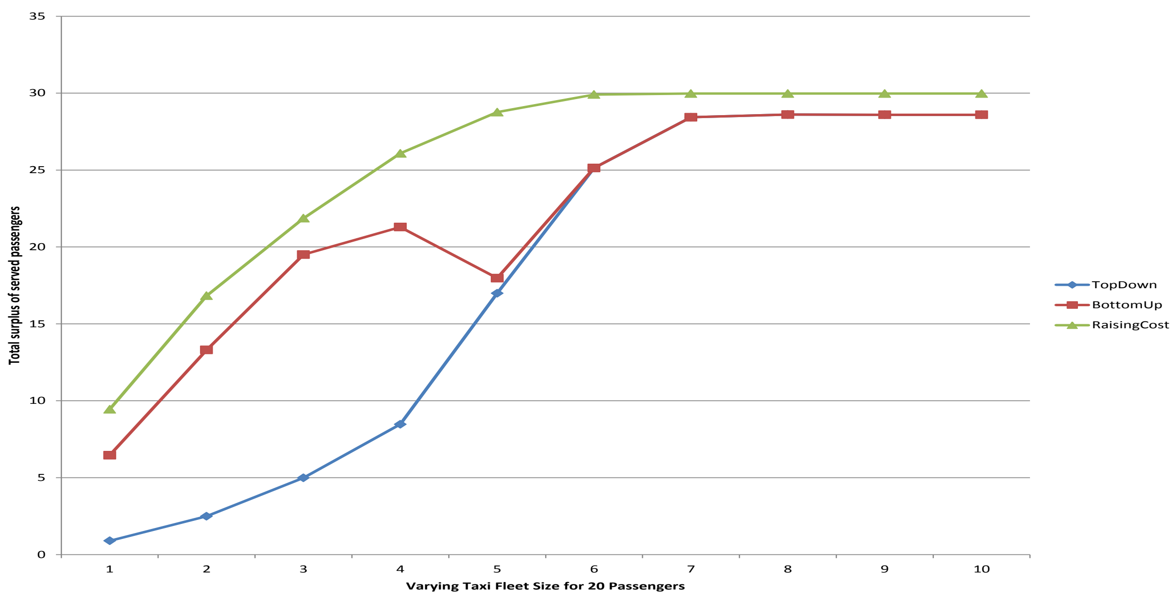


Figure 6.6: Total surplus of served passengers: with increasing fleet size

6.1.3 Varying range of bid values

In this experiment, we observe solution quality with varying bidding behaviour with measurement of number of served passengers and corresponding total direct distance. The experiment is set up by 20 passengers 5 taxis, bidding values of passengers are generated from 4 range $[0.5 - 1.0]$, $[0.6 - 1.0]$, $[0.7 - 1.0]$, $[0.8 - 1.0]$, $[0.9 - 1.0]$. When the lower bound of bid values increases, meaning bidders are willing to pay more money, solution qualities of our incentive compatible mechanisms Bottom-Up and Raising Cost are improved. These improvements are shown in both number of served passengers and total direct distance metrics. Among 2 mechanisms, Bottom-Up mechanism shows better improvement by increasing bid values with narrower gap converging to Optimal Number mechanisms. After first improvement at range $[0.6 - 1.0]$, Raising Cost solutions cannot improve further.

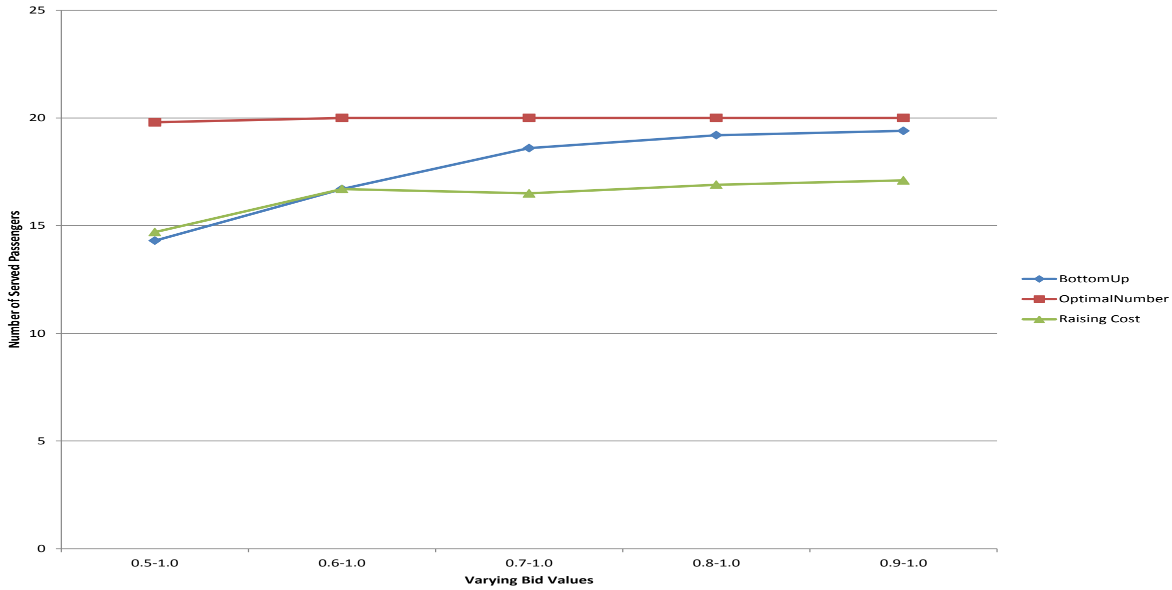


Figure 6.7: Number of served passengers: with increasing bid values

6.1.4 Local Search and Exact Solution

To verify the efficiency of applying local search algorithm into mechanisms, we compare performance of local search and exact methods in synthetic data in which we vary number of passengers.

Results for this experimental comparison of local search and exact solutions are shown in Figure 6.9 for the number of served passengers measure, Figure 6.10 for the total direct distance of served passengers measure and Figure 6.11 for total surplus of served passengers measure.

In all metrics, local search solutions are almost identical to the exact solution for Bottom-up mechanism. On the other hand, local search and exact solutions in Raising cost mechanism are

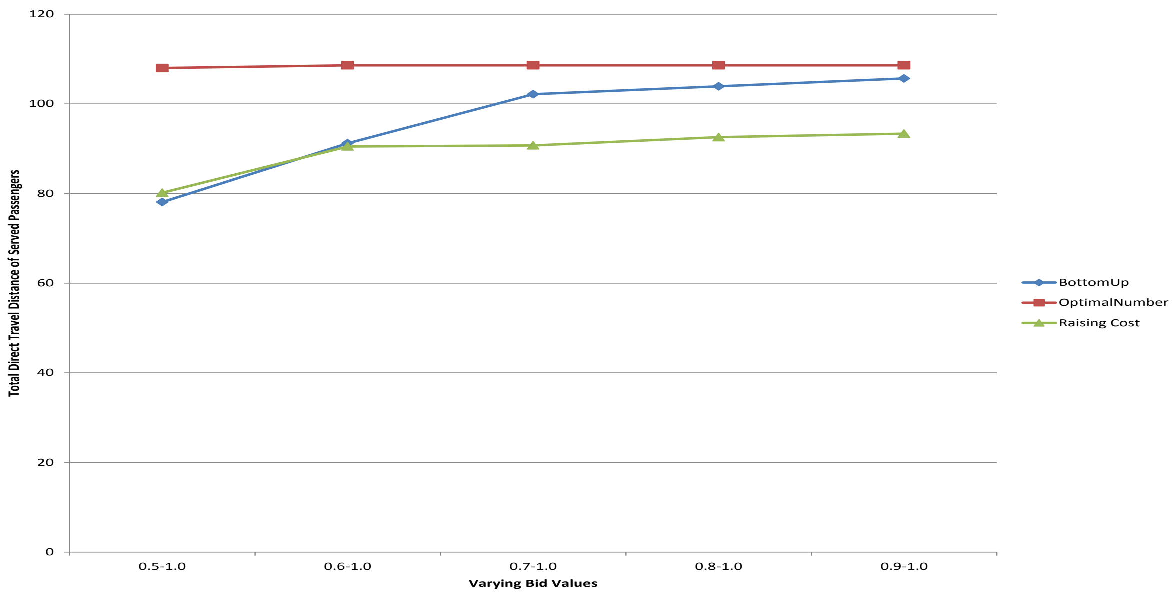


Figure 6.8: Total direct distance of served passengers: with increasing bid value

different from each other. In respect to total number and direct distance of served passengers, when the number of demands is small (from 8 to 16), some local search solutions are better than exact solutions. The reason is that exact solution provides better surplus for passengers, so other quantities might be traded-off. The surplus result is shown in Figure 6.11. While exact solutions provide more surplus than local search solution, local search algorithm shows better result in total direct distance and number of served passengers.

Even exact solutions promise better quality in different solution metrics, they are not scalable with large instances. Size of MIP model in exact method is, in fact, exponential with number of passengers. When the number of calling demands is greater than 29, the exact methods run for 1000 to 6000 seconds on average to output solutions. Meanwhile, runtime of local search algorithms is limited within 1 minute for these instances.

6.2 Ang Mo Kio data analysis

The purpose of this analysis is to choose a suitable mechanism and algorithm to implement in a real public transport hub with real Last Mile demand. In this part, we will begin with

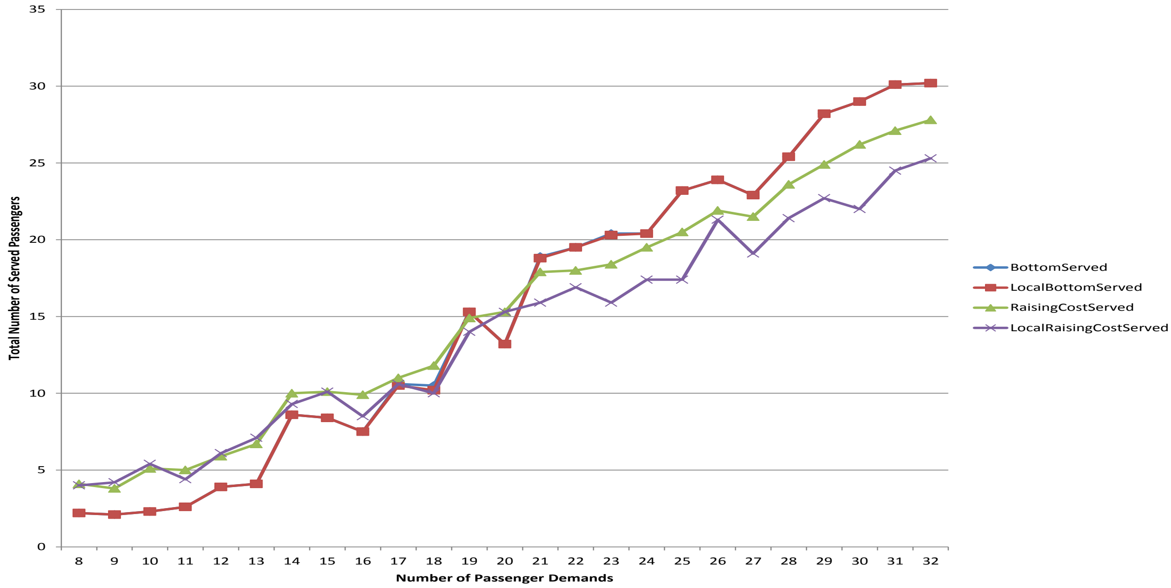


Figure 6.9: Number of served passengers: with increasing demand

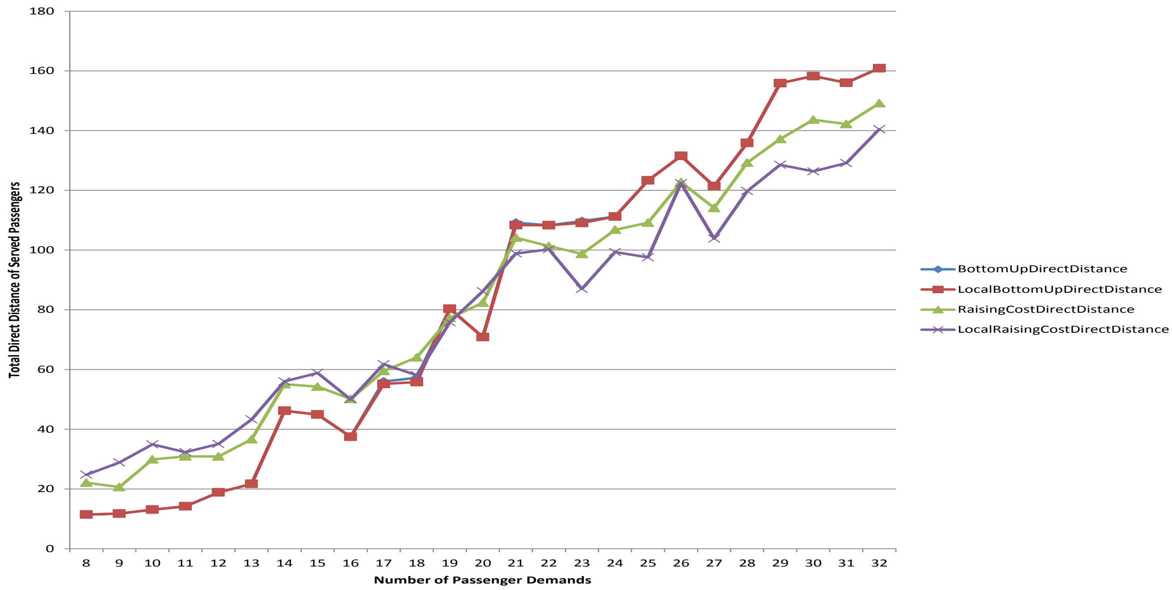


Figure 6.10: Total direct distance of served passengers: with increasing demand

experiments to compare local search and exact method in Bottom-up and Raising cost mechanisms. In all cases, Bottom-up mechanism with local search algorithm is shown to produce optimal solutions in short runtime. We choose this Bottom-up local search method to analyze large scale instances. We observe that these Bottom-up solutions can serve almost all calling demands in the data set.

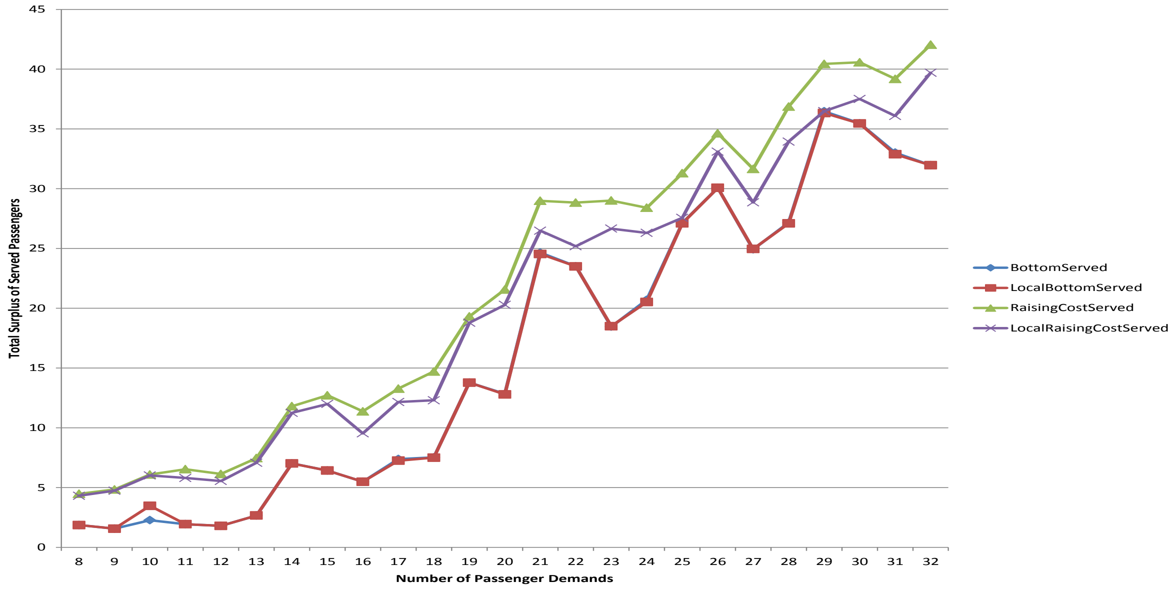


Figure 6.11: Total surplus of served passengers: with increasing demand

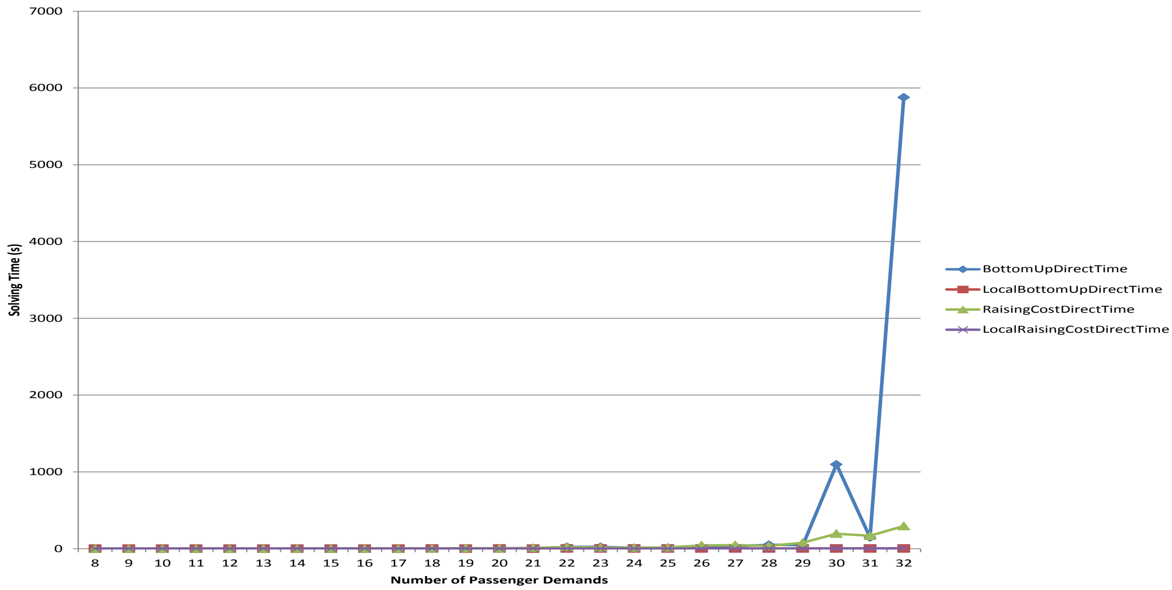


Figure 6.12: Runtime of algorithms: with increasing demand

6.2.1 Local and exact solution with varying number of passengers

Varying number of passengers instances are extracted from Ang Mo Kio data. Each solution result is the average of 10 Last Mile demands from 10 consecutive time intervals of 3 minutes. In this experiment, willing payment of passengers in this data is fixed by value of 0.7.

Solution quality is measured by number of served passenger, their total direct distance, surplus and cost saving. In all instances local search solution and exact solution is almost identical in Bottom-up mechanism. On the other hand, in Raising cost mechanism, local search solutions are different from exact solutions, specifically it sometimes may serve more passengers and direct distance demands in small scale problem (less than 14 passengers).

In respect to total surplus of served passengers, Raising cost mechanism with exact solutions provides the most surplus to passengers among all methods, followed by Raising cost mechanism with local search solutions. Bottom-up mechanism cannot optimize value of this surplus metric.

In addition, we measure the cost saving which is the difference between normal travel cost if all served passengers travel alone by single taxis and the ridesharing travel cost: $\sum_{i,j \in N} x_{ij} s_j - \sum_{i,j \in N} x_{ij} t_{ij}$. Bottom-up solution shows better result in this cost saving metric in comparison with Raising cost solution. Notice that this value is different from surplus. It is proportional to total saving in travel distance while surplus is total saving in utility.

The threshold for scalability of exact method in Ang Mo Kio data is 19 passengers. For instances with passenger number equal or above 19 need, the average runtime is above 1000 seconds, which make the solution computation infeasible in real time system.

By this experimental comparison, with purpose to maximize number of served passengers and the direct distance demands, we decide to focus on Bottom-up mechanism to implement in Ang Mo Kio transportation station. To solve the scale problem, we use local search algorithm whose solution quality is shown close to exact solution. The next 2 follow-up experiments are designed for Bottom-up mechanism with local search algorithm.

6.2.2 Varying bid values

To prove the effectiveness of ridesharing in real data, the following experiments are designed such that the willing payment of passengers increases from 0.5 to 0.9 and ratio of Last Mile demand calling the service increases from 1 to 19 percents.

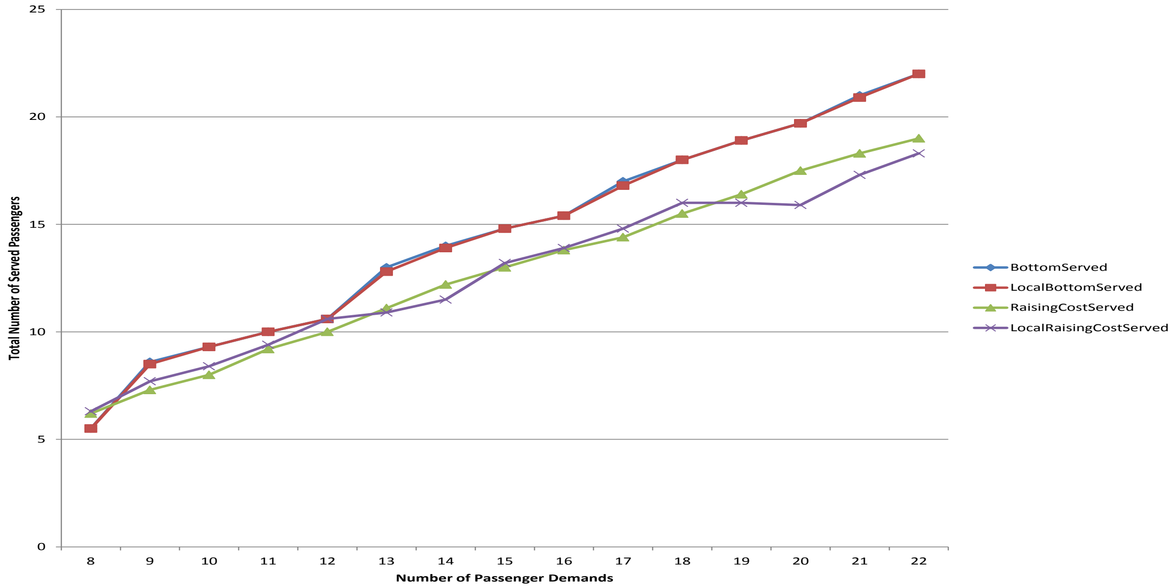


Figure 6.13: Number of served passengers: with increasing demand of Ang Mo Kio data

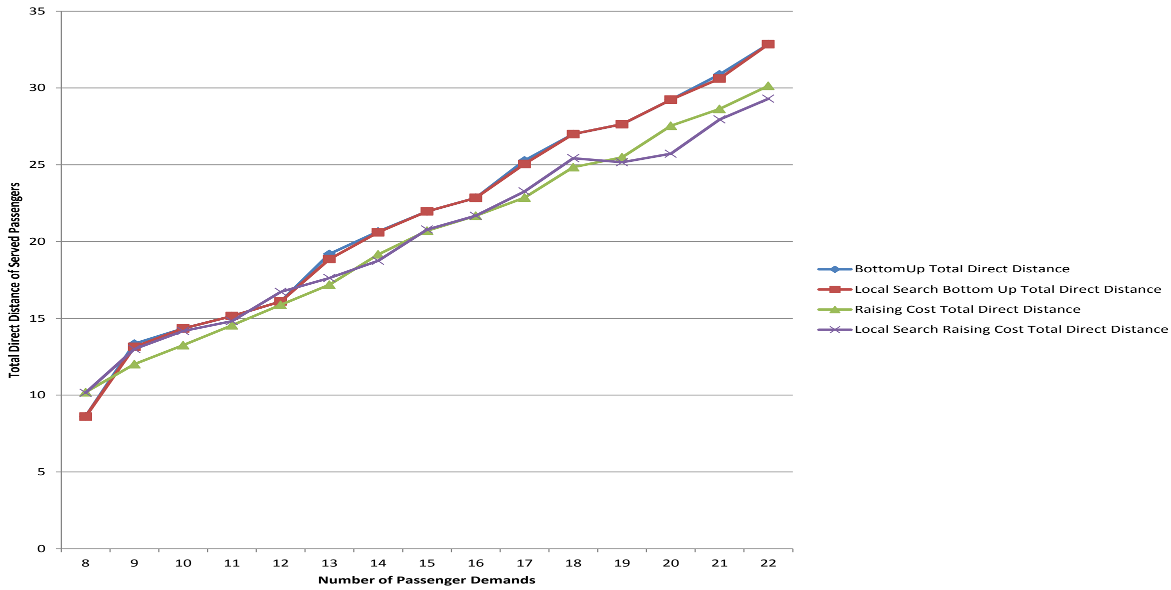


Figure 6.14: Total direct distance of served passengers: with increasing demand of Ang Mo Kio data

Results are shown in Figure 6.18 for the number of served passengers measure and Figure 6.19 for the total direct distance of served passengers measure with the diagonal blue line is number of calling demands. From these results, we observe that when the number of calling demands increases, it is more likely to serve all passengers. When the ratio of participating passengers is above 13 percent, almost all passengers are served.

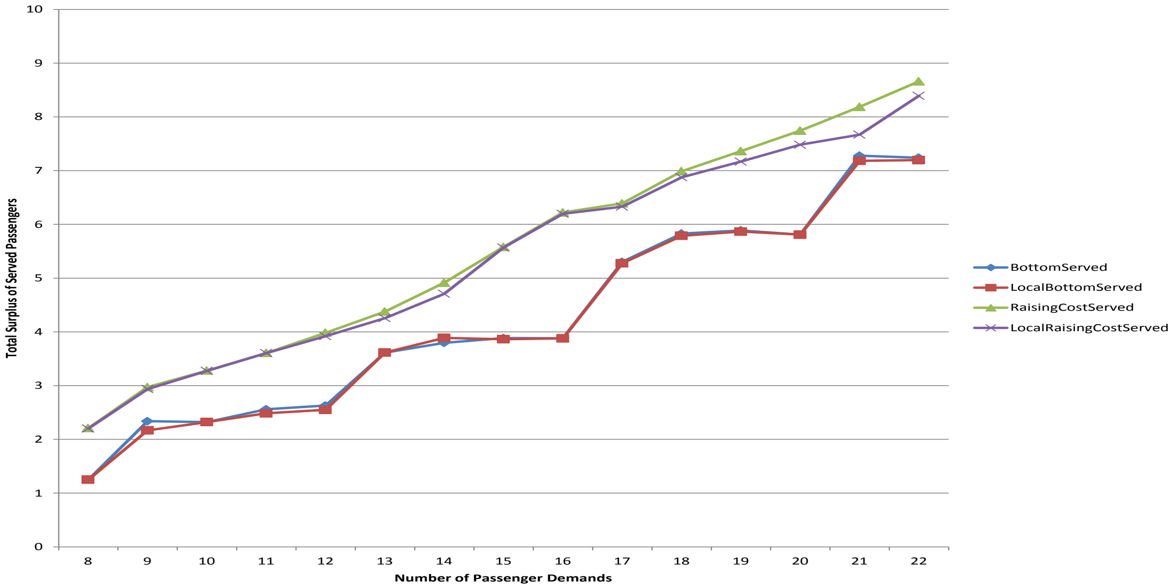


Figure 6.15: Total surplus of served passengers: with increasing demand of Ang Mo Kio data

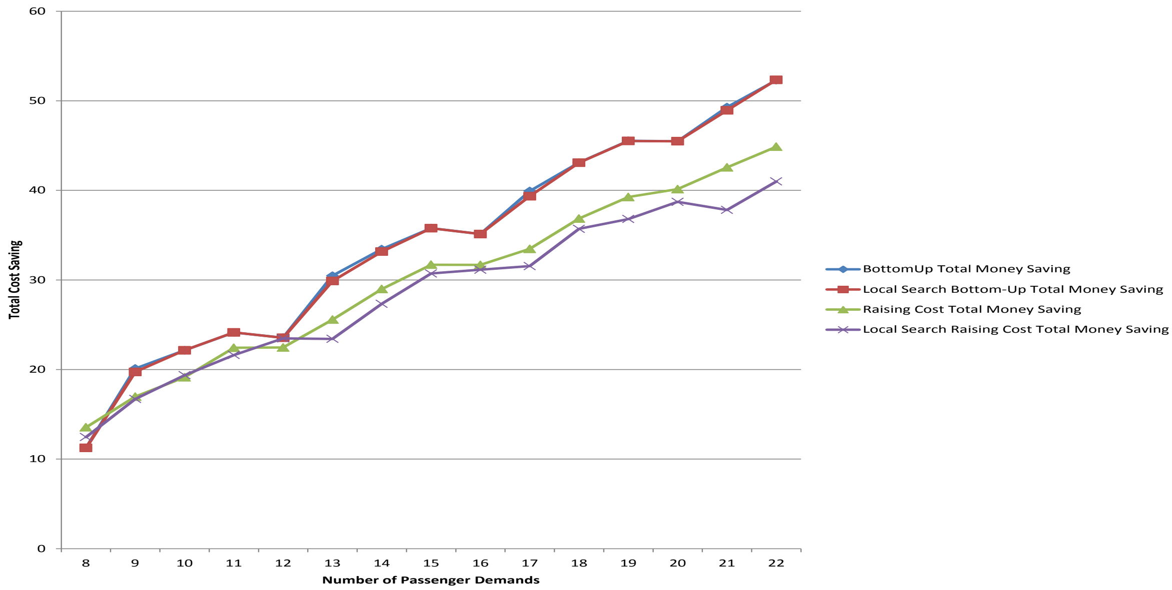


Figure 6.16: Total cost saving of served passengers: with increasing demand of Ang Mo Kio data

The number of served passengers is increased if passengers are willing to pay more. When the willing payment of passengers is 0.9, almost all passengers are served.

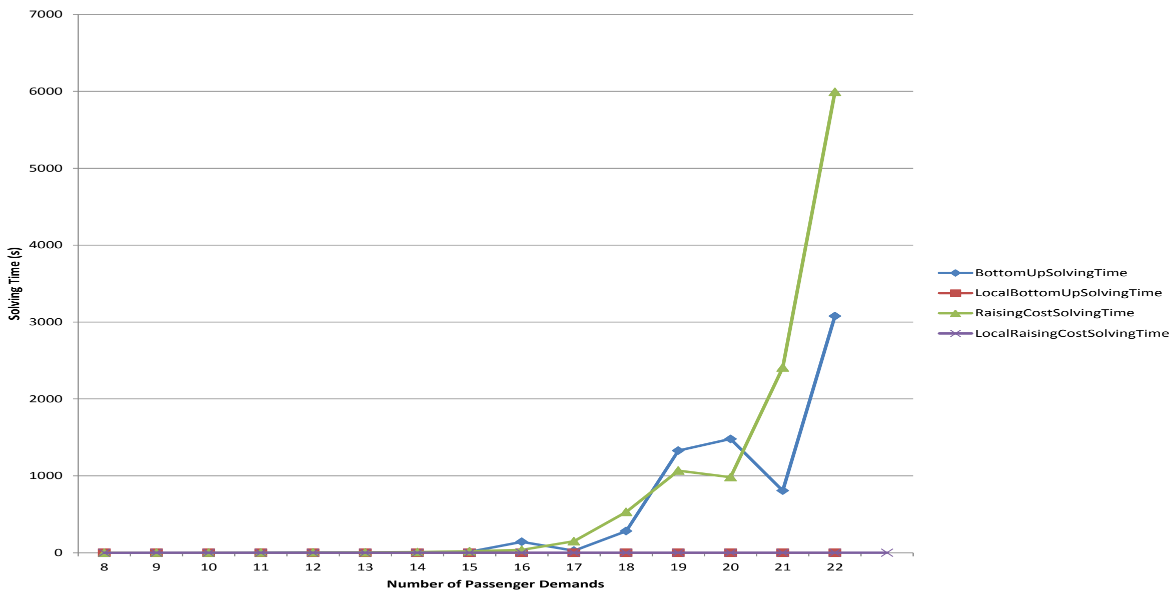


Figure 6.17: Runtime of algorithms: with increasing demand of Ang Mo Kio data

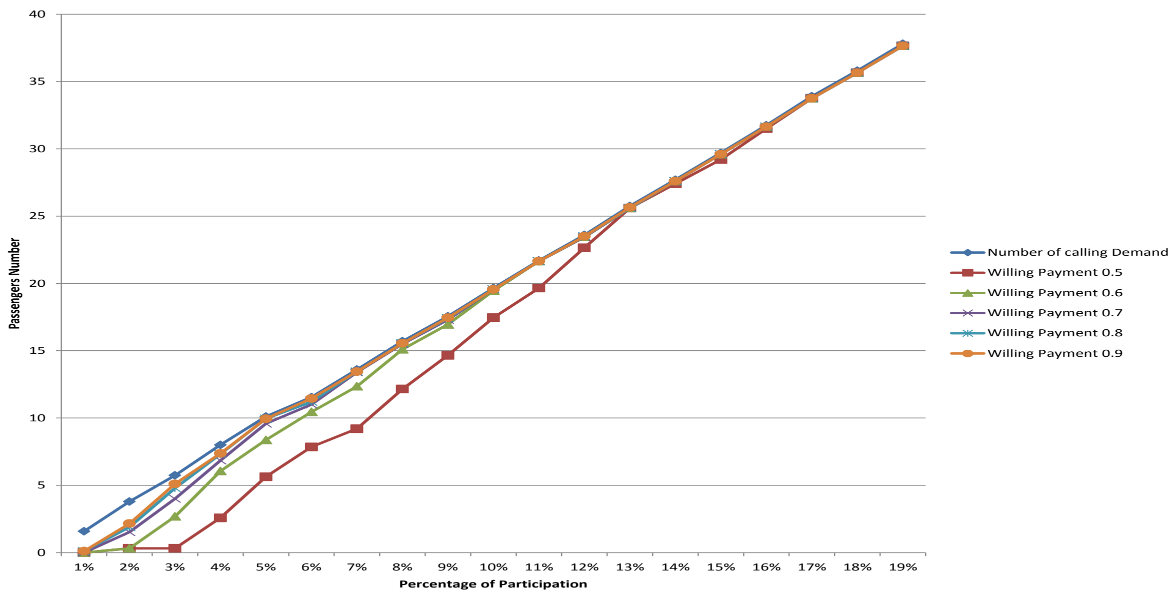


Figure 6.18: Number of served passengers: with increasing participation ratio and varying willing payment

6.2.3 Large number of participating passengers

The purpose of this final experiment is to verify the quality of solution of Bottom-up mechanism with local search algorithm in large scale instance of Ang Mo Kio demands.

The results are plotted in Figure 6.20 for the number of served passengers measure and

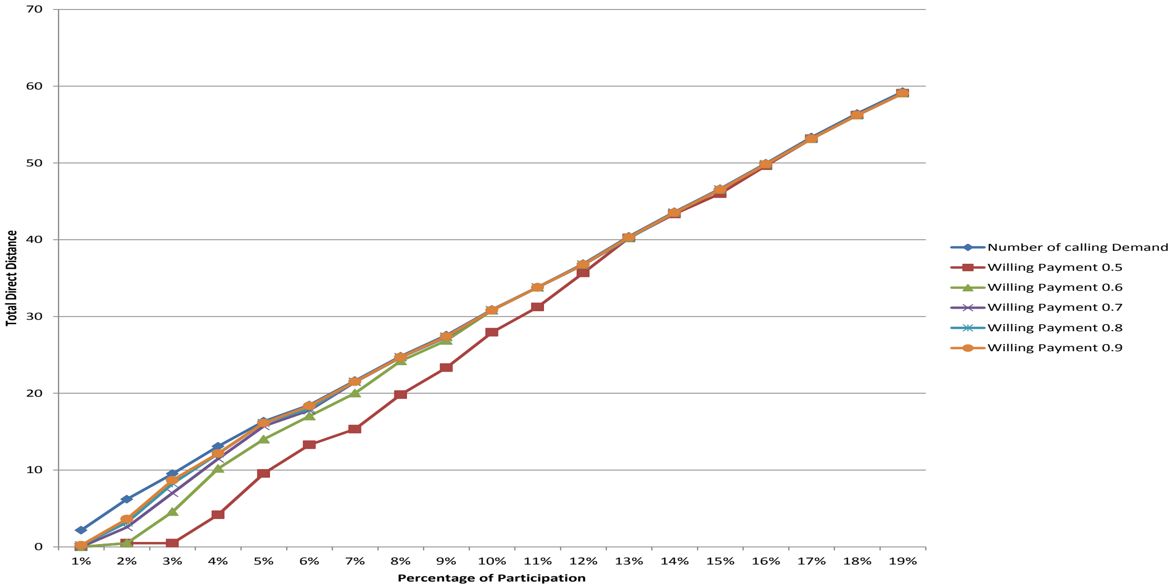
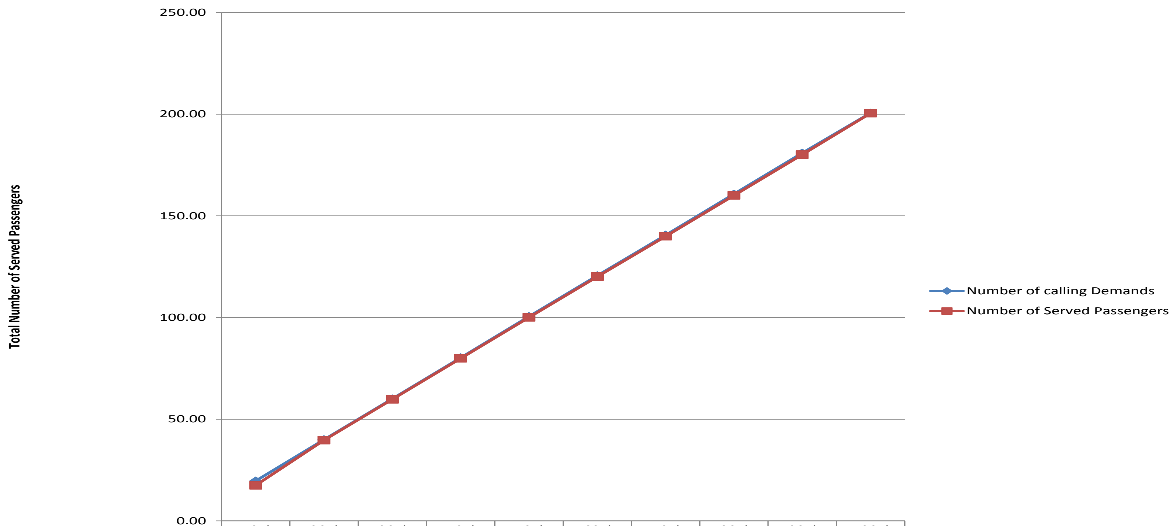


Figure 6.19: Total direct distance of served passengers: with increasing participation ratio and varying willing payment

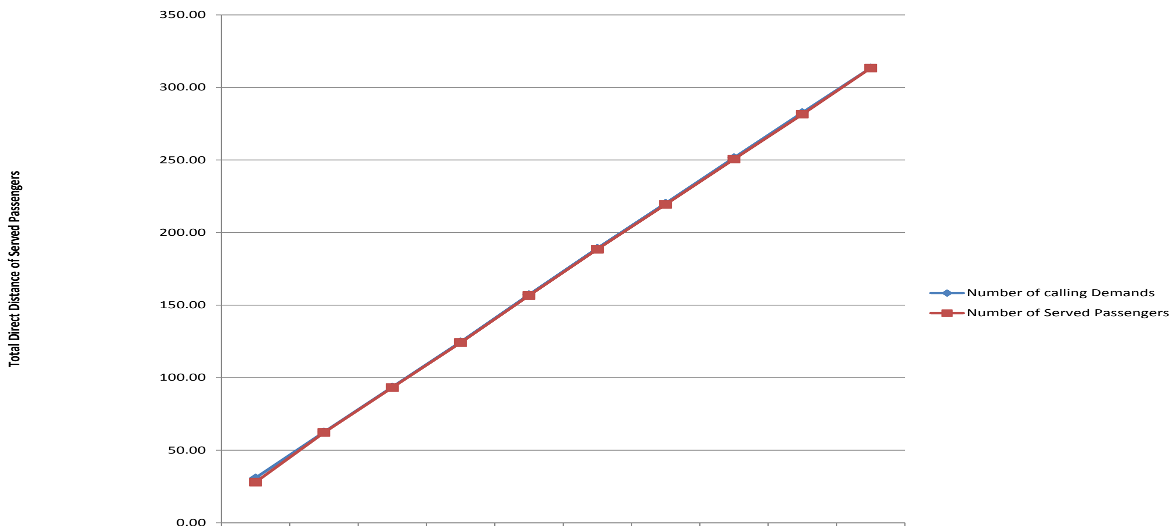
Figure 6.21 for the total direct distance of served passengers measure. It is inferred from this result that when the ratio of participation increases, the average number of missed passengers decreases. When only 10 percent Last Mile demands call for the service, average number of missed passengers is greater than 2. From 20 percent ratio value, number of missed passengers is smaller than 0.5. The reason for this is when more passengers join the service, it is more possible for more passengers to have close proximity in geographic locations.



	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Number of calling Demands	19.65	39.95	60.00	80.25	100.50	120.60	140.55	160.70	180.90	200.55
Number of Served Passengers	17.45	39.70	59.75	79.90	100.05	120.10	139.95	160.00	180.10	200.45

Participation Ratio

Figure 6.20: Number of served passengers: with increasing participation ratio



	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Number of calling Demands	30.91	62.52	93.50	124.67	157.19	189.15	220.20	251.53	282.67	313.39
Number of Served Passengers	27.95	62.17	93.13	124.14	156.52	188.42	219.35	250.52	281.46	313.27

Participation Ratio

Figure 6.21: Total direct distance of served passengers: with increasing participation ratio

Chapter 7

Conclusion and Future Work

7.1 Mechanism study in Last Mile Risharing problem

In this thesis, we study mechanisms that enable Last Mile non-dedicated taxi ridesharing. Our main contribution is to provide a framework for route planning and fair cost sharing. In the operational aspect, we develop a MIP and local search model to solve the specific case of the problem. In the game-theoretic and microeconomics aspect, we propose a cost sharing formula and different incentive compatible mechanisms, namely Top-down, Bottom-up and Raising cost. The efficiency and effectiveness of our proposed mechanisms and algorithms are carefully investigated by a series of experiments on synthetic and real (Ang Mo Kio) data set in different metrics. Based on experimental results that compare different methods, with the focus on fast response solution to maximize number of served demands, we decide to choose Bottom-up mechanism with local search algorithm to analyze the set of real (Ang Mo Kio) data. It can be concluded from these experiments that a ridesharing service can be effectively and efficiently solve Last Mile demands.

Although our work is on mechanisms for a Last Mile non-dedicated taxi sharing system, it can be extended to other systems. The simplest extension can be implemented by using other mode of transport different from non-dedicated taxis with fixed capacity to solve Last Mile demands, e.g. minibuses or vans. However, the extension to heterogeneous capacity of vehicle can trade-off the efficiency in runtime of our MIP model for routing problem. Further extension can be drawn out of ridesharing application, in other game-theoretic problems such as facility location game or set cover game, our framework can be applied to recover the complete budget balance.

7.2 Future work on user interface and usability

In this thesis, we do not consider problems of user interface and feedback in implementing a ridesharing service. However, they are important factors for success for any real-time passenger-centric service such as ours. The first user interface question is how we may implement a software application interface to facilitate interaction between passengers and the bidding system. It is not easy for first time user to specify their willing payment on an unfamiliar notion of the meter rate. In second problem, we need to adapt the model based on real psychological utility evaluation on the service of passengers. Wrong estimation of utility function can undermine effectiveness of the service. Hence what is needed is an indepth behaviorial study to understand the behavior of passengers.

The linear cost sharing formula proposed in this thesis is a simple baseline for implementation which allows problem to be solved by MIP methods. The framework can be adapted to other more complex cost sharing formulas.

To implement a Last Mile taxi sharing service, we can utilize existing mobile and web communication interface and database platforms such as [2], [1],[3]. An additional information element for willing payment can be simply embedded in request tube in current mobile or web based software applications. However, auction based protocols can be difficult for first time passengers in trying to determine their bid values. Tips of supportive information and suggestion for passengers are necessary. It is infeasible to explain detailed theoretical principles underlying operating mechanism. so the information presented should be brief but useful.

I suggest that the following information should be displayed:

- Passengers will be served with priority based on their willing payment on meter rate. This is the simple and intuitive rule for passengers participating in an auction based protocol.

- As in many route planning app, an approximate comparison among different modes of transport with Last Mile taxi sharing should be provided. It provides a bounded estimation for opportunity cost to use the taxi sharing service. For example, if the public transportation cost c_p, travel time of public transportation B_p, normal taxi cost 1, direct travel time of normal taxi s, the bounded taxi meter rate would follow

  $$c_p \leq m(2s - B_p) \tag{7.1}$$

  or

  $$m \geq \frac{c_p}{2s - B_p}, \tag{7.2}$$

  in which the RHS is lower bound for meter rate willing payment to use taxi sharing service.

  Furthermore, the willing payment rate can be represented as a discrete value instead of continuous value. Thereby, passengers can choose an option in a set of possible suggested rate.

- History of past meter rate charge. It can give passengers an intuition of how system is operated and help them to decide which willing payment should be bid to get served.

- An explicit information of routing and cost allocation solution from the system. For credibility, passengers should be informed of their trips before they confirm/accept.

- Although information in theoretical principles can be redundant for regular passengers, we should provide an access to detailed version of explanation. It would make the service more credible if passengers are content with the information presented.

- Feedback is prevalent in current ridesharing software applications. We could solicit comments about the fairness of cost sharing from the user of the system.

Bibliography

[1] http://www.carpoolworld.com/. Accessed January 21, 2013.

[2] http://www.gomywayapp.com/. Accessed January 21, 2013.

[3] http://www.split-it.sg/. Accessed January 21, 2013.

[4] Niels A. H. Agatz, Alan L. Erera, Martin W. P. Savelsbergh, and Xing Wang. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, 223(2):295–303, 2012.

[5] Niels A.H. Agatz, Alan L. Erera, Martin W.P. Savelsbergh, and Xing Wang. Dynamic ride-sharing: A simulation study in metro atlanta. *Transportation Research Part B: Methodological*, 45(9):1450 – 1464.

[6] A.M. Amey. Real-time ridesharing: Exploring the opportunities and challenges of designing a technology-based rideshare trial for the mit community. Technical report, Massachusetts Institute of Technology, Department of Civil and Environmental Engineering; and, (M.C.P.)–Massachusetts Institute of Technology, Department of Urban Studies and Planning, 2010.

[7] John Attanucci Andrew Amey and Rabi Mishalani. real-time ridesharing the opportunities and challenges of *Transportation Research Record: Journal of the Transportation Research Board*, 2217:103–110, 2011.

[8] Shih-Fen Cheng, Duc Thien Nguyen, and Hoong Chuin Lau. A mechanism for organizing last-mile service using non-dedicated fleet. Number 20866, page to appear. 2012 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT-12), 12 2012.

[9] F. Ciari. *Modeling collective taxis in a multi-agent traffic simulation framework*. [Arbeitsberichte Verkehrs- und Raumplanung]. ETH, Eidgenössische Technische Hochschule Zürich, Institute for Transport Planning and Systems (IVT), 2009.

[10] Cordeau, Jean Francois, Laporte, and Gilbert. The dial-a-ride problem: models and algorithms. *Annals of Operations Research*, 153(1):29–46, September 2007.

[11] Jean-Franois Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579–594, 2003.

[12] Gonalo Correia and Jos Manuel Viegas. Carpooling and carpool clubs: Clarifying concepts and assessing value enhancement possibilities through a stated preference web survey in lisbon, portugal. *Transportation Research Part A: Policy and Practice*, 45(2):81 – 90, 2011.

[13] Malucelli F. Crainic, T.G. and M. Nonato. Flexible many-to-few + few-to-many = an almost personalized transit system. *TRISTAN IV, So Miguel Azores Islands*, pages 435–440, 2001.

[14] Nikhil R. Devanur, Milena Mihail, and Vijay V. Vazirani. Strategyproof cost-sharing mechanisms for set cover and facility location games. *Decis. Support Syst.*, 39(1):11–22, March 2005.

[15] Stefan Engevall, Maud Göthe-Lundgren, and Peter Värbrand. The heterogeneous vehicle-routing game. *Transportation Science*, 38(1):71–85, February 2004.

[16] Ulrich Faigle, SndorP. Fekete, Winfried Hochstttler, and Walter Kern. On approximately fair cost allocation in euclidean tsp games. *Operations-Research-Spektrum*, 20:29–37, 1998.

[17] Jean francois Cordeau. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, 54:573–586, 2003.

[18] M. Göthe-Lundgren, K.O. Jörnsten, P. Värbrand, and Québec) Groupe d'études et de recherche en analyse des décisions (Montréal. *On the Nucleolus of the Basic Vehicle Routing Game*. Cahiers du GÉRAD. Groupe d'études et de recherche en analyse des décisions,1992., 1992.

[19] Torben Bach Pedersen Gyozo Gidfalvi. Cab-sharing: An effective, door-to-door, on-demand transportation service. In *Proceedings of the 6th European Congress on Intelligent Transport Systems and Services*, 2007.

[20] Yoshitsugu HAYASHI. Transport Solutions for Congestion and Climate Change Control in Developing Mega-Cities.

[21] David A. Hensher. Climate change, enhanced greenhouse gas emissions and passenger transport What can we do to make a difference?

[22] Sheldon H. Jacobson and Douglas M. King. Fuel saving and ridesharing in the us: Motivations, limitations, and opportunities. *Transportation Research Part D: Transport and Environment*, 14(1):14 – 21, 2009.

[23] R. M. Jørgensen, J. Larsen, and K. B. Bergvinsdottir. Solving the dial-a-ride problem using genetic algorithms. *The journal of the Operational Research Society*, 58:1321–1331, 2007.

[24] Ece Kamar and Eric Horvitz. Collaboration and shared plans in the open world: studies of ridesharing. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI'09, pages 187–194, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[25] S. Karaman and E. Frazzoli. Vehicle routing with linear temporal logic specifications: Applications to multi-UAV mission planning. In *AIAA Conf. on Guidance, Navigation, and Control*, Honolulu, HI, 2008.

[26] Alexander Kleiner, Bernhard Nebel, and Vittorio Amos Ziparo. A mechanism for dynamic ride sharing based on parallel auctions. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume One*, IJCAI'11, pages 266–272. AAAI Press, 2011.

[27] Yeqian Lin, Wenquan Li, Feng Qiu, and He Xu. Research on optimization of vehicle routing problem for ride-sharing taxi. *Procedia - Social and Behavioral Sciences*, 43(0):494 – 502, 2012.

[28] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.

[29] Cornelius Nuworsoo. Smart dial-a-ride for demand-responsive transit operations: Research and development of a concept of operations. Technical report, California Polytechnic State University, 2011.

[30] Institute of Excellence at SMU. Customer satisfaction index of singapore 2012. *Operations Research*, 2012.

[31] Institute of Excellence at SMU. MONTHLY DIGEST OF STATISTICS SINGAPORE, NOVEMBER 2012. *Department of Statistics, Ministry of Trade & Industry, Republic of Singapore.*, 2012.

[32] Martin Pal and Eva Tardos. Group strategyproof mechanisms via primal-dual algorithms. In *In Proceedings of the 44th Annual Symposium on Foundations of Computer Science (FOCS*, pages 584–593, 2003.

[33] Stefan Ropke, Jean-François Cordeau, and Gilbert Laporte. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Netw.*, 49(4):258–272, July 2007.

[34] Alireza Sahami, Thomas Kubitza, Florian Alt, Philipp Tarasiewicz, Andreas Bungert, Vladimir Minakov, and Albrecht Schmidt. Mobile context-based ride sharing. *Adjunct Proceedings of the Ninth International Conference on Pervasive Computing*, 2011.

[35] Jennifer Saranow. Carpooling for Grown-Ups — High Gas Prices, New Services Give Ride-Sharing a Boost; Rating Your Fellow Rider. *The Wall Street Journal.*

[36] David Schrank, Tim Lomax, and Bill Eisele. Urban mobility report 2010. Technical report, Texas Transportation Institute, 2010.

[37] Susan Shaheen. U.s. carsharing & station car policy considerations: Monitoring growth, trends & overall impacts. Institute of transportation studies, working paper series, Institute of Transportation Studies, UC Davis, 2004.

[38] Susan Shaheen, Stacey Guzman, and Hua Zhang. Bikesharing in europe, the americas, and asia: Past, present, and future. Institute of transportation studies, working paper series, Institute of Transportation Studies, UC Davis, 2010.

[39] L. S. Shapley. A value for n-person games. *Contributions to the theory of games*, 2:307–317, 1953.

[40] L. S. Shapley. On balanced sets and cores. *Naval Research Logistics Quarterly*, 14:453–460, 1967.

[41] Francesco Sottini, Sameh Abdel-Naby, and Paolo Giorgini. Andiamo: A multiagent system to provide a mobile-based rideshare service. Technical report, Departmental Technical Report.

[42] Arie Tamir. On the core of a traveling salesman cost allocation game. *Oper. Res. Lett.*, 8(1):31–34, February 1989.

[43] Gauresh Pandit Tejas Talele and Parimal Deshmukh. Dynamic ridesharing using social media. (4), 2012.

[44] Adam Millard-Ball; Jessica ter Schure; Christine Fox; Jon Burkhardt; Gail Murray; Transportation Research Board. *TCRP Report 108: Car-Sharing: Where and How It Succeeds*. The National Academies Press, 2005.

[45] Duygu Yengin. Characterizing the shapley value in fixed-route traveling salesman problems with appointments. *International Journal of Game Theory*, 41:271–299, 2012.

[46] Issam Zidi, Khaled Mesghouni, Kamel Zidi, and Khaled Ghedira. A multi-objective simulated annealing for the multi-criteria dial a ride problem. *Eng. Appl. Artif. Intell.*, 25(6):1121–1131, September 2012.