Title
Optimization for Urban Mobility Systems

Author
Lee, Jiung

Publication Date
2019

Optimization for Urban Mobility Systems

by

Ji Ung Lee

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Industrial Engineering and Operations Research

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Zuo-Jun (Max) Shen, Chair
Professor Philip Kaminsky
Professor Terry Taylor

Summer 2019

Optimization for Urban Mobility Systems

Abstract

Optimization for Urban Mobility Systems

by

Ji Ung Lee

Doctor of Philosophy in Engineering - Industrial Engineering and Operations Research

University of California, Berkeley

Professor Zuo-Jun (Max) Shen, Chair

In the recent decades, new modes of transportation have been developed due to urbanization, highly dense population, and technological advancement. As a result, design and operation of urban transportation have become increasingly important to better utilize the resources and efficiently meet demand. This dissertation was motivated by two problems on optimizing design and control of urban transportation. In the first one, we consider a problem of dynamically matching heterogeneous market parcitipants so as to maximize the total number of matching, which was motivated by practices of ride-sharing platforms. In the other problem, we study efficient design of elevator zoning system in high-rises with uncertainty in customer batching.

In Chapter 1, we consider a multiperiod stochastic optimization of a market that matches heterogeneous and impatient agents. The model was mainly motivated from carpooling products run by ride-sharing platforms such as Uber and Lyft, and kidney exchange market, where market participants are heterogeneous in terms of how likely they can be matched with others. In the case of a ride-sharing platform, one of the key operational decisions for carpooling is to efficiently match riders and clear the market in a timely manner. In doing so, the platform needs to take into account the heterogeneity of riders in terms of their trip types(e.g origin-destination pair) and different matching compatibility. For example, some customers may request rides within San Francisco, while others may request rides from San Francisco to outside the city. Since picking up and dropping off a customer within the city can be done within relatively short amount of time, those who want to travel within the city can be matched with any other riders for carpooling. However, the destinations of those who want to travel to outside the city may be very different, and in order to maintain customers'

additional transit time due to carpooling, it is likely that they can be only matched with those who want to travel within the city. In the case of kidney exchange where market participants arrive in the form of patient-donor pair, pairs with donor who can donate her kidney to most of patients (for example, blood type O) and patient who can get kidney from most of donors (for example, blood type AB) can be easily matched to other pairs. The opposite case would be hard-to-match pair that is incompatible for matching with most of other pairs. Our model is an abstraction of these two motivating examples, and considers two types of agents: *easy-to-match* agents that can be matched with either type of agents, and *hard-to-match* agents that can be only matched with easy-to-match ones. We first formulate a dynamic program to solve for optimal matching decisions over infinite time horizon in a discrete time setting, and characterize structure of optimal stationary policies. Inspired by practices in kidney exchange where the market is cleared for every fixed time interval, we connect the discrete time model to a continuous time setting by investigating the effect of the length of matching intervals on the matching performance. Results from numerical experiments indicate certain patterns in the relationship between the length of matching intervals and the maximum number of matching achieved, and provides valuable insights for future direction of research.

In Chapter 2, we consider a zoning problem for elevator dispatching systems in high-rises. In practice, zoning is frequently used to improve efficiency of elevator systems. The idea of zoning is to prevent different elevators from stopping at common floors, which may result in long service times of elevators and thus long waiting times of customers. Our goal is to provide a mathematical framework that can help a system planner decide optimal zoning design with some performance guarantee. To this end, we focus on uppeak traffic situation during morning rush hour, which is in general the heaviest traffic during the day. The performance in the uppeak traffic situation can be considered as the system's capacity, because if the system can handle uppeak traffic well, it can also serve other types of traffic with good performance. Thus, the performance measure in the uppeak traffic situation can be used as a metric to choose the optimal zoning configuration. One of the components that complicate the problem is customer batching, on which the system may not have a control. In view of this, we formulate an adversarial optimization problem that can measure the system performance of different zoning decisions. By considering the heaviest traffic situation of the day and using the adversarial framework, we provide a model that can be used for capacity planning of elevator systems. We formulate mixed-integer linear program(MILP)s to find the optimal zoning configuration. To solve the MILPs, we show that we can use simple greedy algorithms and solve smaller linear programs. We also provide a few illustrative examples as well as numerical experiments to verify the theoretical results and obtain insights for further analysis.

To my mother.

Contents

Contents ii

List of Figures iii

1 Dynamic Matching of Heterogeneous Agents with Abandonment 1
- 1.1 Introduction . . . 1
  - 1.1.1 Applications . . . 2
- 1.2 Literature review . . . 3
- 1.3 Setting . . . 5
  - 1.3.1 Continuous time setting . . . 5
  - 1.3.2 Discrete time setting . . . 7
  - 1.3.3 Relationship between discrete and continuous time settings . . . 8
- 1.4 Optimal policy: discrete time settings . . . 9
  - 1.4.1 Convergence of value function iteration . . . 9
  - 1.4.2 Structural properties of optimal policy . . . 11
- 1.5 Numerical Experiments . . . 23
- 1.6 Conclusion and future Directions of research . . . 25

2 Service Zone Design for Elevator Systems in High-Rises 27
- 2.1 Introduction . . . 27
- 2.2 Literature review . . . 29
- 2.3 Model . . . 30
  - 2.3.1 Setting . . . 31
  - 2.3.2 Zoning case . . . 33
  - 2.3.3 No-zoning: two elevators . . . 41
- 2.4 Numerical study . . . 47
  - 2.4.1 Uniform demand . . . 47
  - 2.4.2 Imbalanced demand . . . 50
- 2.5 Conclusion and future directions of research . . . 52

Bibliography 55

List of Figures

1.1 Convergence of the value of the myopic policy 24
1.2 Matching cardinality by interval: comparison by entrance rates 24
1.3 Matching cardinality by interval: comparison by types 25

2.1 Performance of different zoning configurations: uniform demand 50
2.2 Performance of different zoning configurations: imbalanced demand 53

Acknowledgments

My PhD years have been wonderful time that I could learn a lot from many people. I am grateful to have all of those who helped and taught me during this journey, and I cannot fully express my appreciation and gratitude to them.

First and foremost, I am deeply indebted to my advisor, Professor Max Shen, for his guidance and support. He gave me a lot of encouragement and inspiration to explore many interesting research areas, freedom to choose my research topic, and opportunities to improve myself as a future researcher. During the individual and group meetings, he has always guided me and other students to be better researchers and engineers. He has been a great mentor in my time at Berkeley, and I have been privileged to have had opportunities to learn from him.

I would like to thank Professor Kaminsky, Professor Taylor for their academic guidance and for serving on my dissertation committee and qualifying exam committee. Their invaluable feedback has helped me complement important details in this dissertation. Professor Kaminsky also taught a course in supply chain management in my first year at Berkeley, which helped me obtain fundamental knowledge on modeling approach that has been very useful for my research.

I am fortunate and grateful to the many friends and colleagues who have supported me during my unique journey in the past years. First of all, I would like to thank to Seunghwan Lim for his contribution to our research, which became the first chapter of this dissertation. My friends and colleagues made me go through the hard times that I had to struggle and taught me a lot of life lessons. I am very grateful to many of them, and will list only a few here: Kevin Li, Clay Campaigne, Joe Borja, Amber Richter, Heejung Kim, Shiman Ding, Cheng Lyu, Steward Liu, Erik Bertelli, Tugce Gurek, Quico Spaen, Alfonso Andres Lobos Ruiz, Tianyi Lin, Meng Qi, Sheng Liu, Ying Cao, Yanqiao Wang, Andy Mao, Junyu Cao, SangWoo Park, Han Feng, Arman Jabbari, HyeSoo Choi, Euiyoung Kim, Chaeyoung Shin, Donghyun Danny Choi, Hong-June Park, Taekjun Kim, Keunhong Jung, Titouan Jehl, Matt Olfat, Salar Fattahi, Dean Grosbard, Mark Velednitsky, Renyuan Xu.

I would like to thank my mother and family for their support and for believing in me. Their encouragement and trust have been the biggest help that I have received during my PhD years. I am especially grateful to my mother for her wisdom, prayers, optimism, and unlimited support and care. It has been the love and lessons from you that helped me endure difficult moments in the past years. Since I was very young, you have always inspired me to not give up, work hard and pursue my intellectural curiosity. I hope to continue to be your proud son, and that we can create values and contribute to the world as we have always talked. Last but not least, I would like thank my girlfiend, HyunJung Lee, for her love and support. You supported me through the hardest times in my life, and were with me for all the highs and lows. Moving forward together, I hope to be the person you can lean on the

same way.

Chapter 1

Dynamic Matching of Heterogeneous Agents with Abandonment

1.1 Introduction

In the last decade, dynamic matching has received increasing and broad attention from operation research and economics. The general structure of the dynamic matching problem can be described as follows. Agents of various types arrive to the platform over time. The platform can match agents, and gets the reward based on the match it created. The matched agents leave the platform, and does not comeback. Thus, given a existing pool of agents, the platform faces the trade-off between earning the payoff in the current period, and thickening the pool for better future match. Applications include wide range of problems including, but not restricted to, ride-sharing, carpooling, and kidney exchange.

One of the common practices in many real-world dynamic matching market is to clear the market in every period of certain time length. During the prescribed length of time, the platform just keeps the arriving agents and wait for the market to be thickened. Then, the platform clears the market, wins the reward, and repeats the process above. The unmatched agents can be carried over to the next period, or perished based on the setting of the problem. One of the most famous examples is organ exchange. Although different countries have different rules about specific matching policies, many of them share the common feature that the organ exchange in cleared periodically. This practice is also often used as a justification for building discrete-time model for many real-world matching platforms, which obviously run over continuous time.

To the best of our knowledge, our model is the first matching model with heterogeneous market participants and stochastic departures(abandonment). Also, it has been less studied what would be the optimal way to set the length of the time between consecutive matching operations, and decisions are often based on rule of thumb. In this paper, our goal is to build a stylized dynamic matching market model and provide insights for finding the optimal length of period for market to be cleared.

In our model, there are two-types of agents, easy-to-match (E) and hard-to-match (H). A type E agent can be matched to either a different type E agent or a type H agent. However, a type H agent can only be matched to type H agents. New agents arrive to the platform by Poisson processes, whose rates vary by the types of agents (E or H). Existing agents leave the market with exponential rate that is common for both types of agents. In this setting, the platform sets the length of time period and the matching rule to maximize discounted cardinality of matching. Thus, if the platform waits for longer period of time or keeps more agents in the market instead of matching them immediately, it faces two types of waiting cost. First, the reward will be discounted, and the second, the unmatched agents may perish and leave the market while waiting for the next matching period.

The rest of the present paper is organized as follows. In Section 1.1.1, we explain more about motivating examples in ride-sharing and kidney exchange market. In Section 1.2, we provide comprehensive literature review. In Section 1.3, we describe the original setting of our dynamic type matching model in continuous time. In Section 1.3.2, we define the discrete time setting given a fixed time window between consecutive matching operations, and explain its connection to the continuous time setting. In Section 1.4, we show theoretical results of the discrete time model and provide its structural properties. In Section 1.5, we provide results of numerical study that verify our theoretical results. In Section 1.6, we explain the current on-going works on additional numerical experiments for finding the optimal matching policy and optimal length of the time window, and potential directions of the research.

1.1.1 Applications

In this section, we briefly explain motivating examples of our model to provide more contexts. The first motivating example of our model with easy-to-match and hard-to-match agents is kidney exchange. When donor-patient pairs arrive to kidney exchange, pairs with donor who can donate her kidney to most of patients (for example, blood type O) and patient who can get kidney from most of donors (for example, blood type AB) can be easily matched to other pair. The opposite case would be hard-to-match pair that is incompatible with most of patient-donor pairs. Note that our model simplifies this problem in three dimensions. First, instead of considering various patient-donor combinations, there are only two types of patient-donor pairs. Second, the probability of match within and across types are simplified. Finally, actual kidney exchange uses various matching methods, such as chain or unconditional donor, but we only consider binary exchange in our model.

The second application is carpooling in ride-sharing context. Ride-sharing platforms like Uber and Lyft enable riders with different destinations to share the same car and pay lower price for their rides. Consider a platform trying to match riders in a city. There are two types of riders: those who travel within city and those who travel to outside of the city. The first type of riders can be matched to anyone, because destinations within the city are fairly close to each other. However, when destinations outside the city are fairly diverse, it would be almost impossible to match the second type of riders together. As in the first example, our model can be viewed as an abstraction of this example by assuming the probability that different types can be matched is zero. However, the model still capture the central idea that agents can be classified based on how they are likely to be matched to other agents.

1.2 Literature review

Optimization of matching in dynamic settings has been receiving increasing attention from economics and operations research literature. Most of these works are driven by real-life examples with significant impacts such as the college admission problem, kidney exchange, and the online ad allocation problem.(See Roth and Sotomayor (1990); Abdulkadiroğlu and Sonmez (2013); Mehta et al. (2013) and references therein.)

As alluded earlier, the kidney exchange is one of the most well-known applications that has motivated abundant research. Papers in this stream have focused on designing efficient matching mechanism subject to compatibility constraints using cycles or chains of patient-donor pairs. Earliest papers that study kidney exchange problem from the viewpoint of mechanism design include Roth et al. (2004) and Roth et al. (2007). Roth et al. (2004) suggests *top trading cycles and chains*, a variation of top trading cycle, as a potential candidate for kidney exchange, and shows that the mechanism is Pareto-efficient and strategy-proof. In Roth et al. (2007), there is a restriction on the number of patient-donor pairs that can be used to form a chain or a cycle, and the paper analyzes utility gain as this number increases. These papers consider static setting, and the main constraint that they consider is compatibility constraint based on patients' and donors' blood types. The present paper, however, considers a dynamic problem with simplified compatibility constraint. Ünver (2010) find the optimal mechanism for dynamic kidney exchange in the setting where different patient-donor pairs arrive over time. The setting looks familiar with that of the present paper, but there are some notable differences. In the present paper, the cost for waiting comes from unmatched agents leaving the platform. However, in Ünver (2010), there is no abandonment and the waiting cost is explicitly included in the objective. More importantly, while Ünver (2010) assumes discrete-time setting, we study the optimal way to build a discrete-time problem from the more general continuous real-time setting.

In the last decade, successes of sharing economy from ride-sharing(e.g Uber, Lyft) to food delivery(e.g Uber Eats, DoorDash) have motivated a stream of research that studies various operational problems to make supply meet demand efficiently(See Hu (2019, 2018); Benjaafar and Hu (2019) and references therein). One of the key problems in this context is making matching decisions between market participants(e.g drivers and riders in ride-sharing) over time so as to optimize performance measures of the marketplace such as profit generated from matched pairs and the number of successful matching.

Many matching models motivated by operations in sharing economy study dynamic matching decisions among heterogeneous agents. Given that matching reward depends on the types of matched agents, the optimal matching decision should be able to not only capture matches that need to be done in a timely manner, but also maintain the *market thickness* appropriately for future matches. In Hu and Zhou (2019), the authors analyze a dynamic program for matching heterogeneous agents with stochastic arrivals over a finite time horizon, where the matching reward depends on the types of matched agents. Their model assumes a bipartite matching compatibility structure, whereas our model has a non-bipartite matching compatibility and "flexible" type agents in our model can be matched between themselves. Baccara et al. (2018) studies a bipartite model with two types on each side, where agents arrive as a pair in each period. In their model, matching decisions are made so as to minimize the holding cost incurred by waiting agents, while in our model, the goal is to maximize the total number of matched agents where unmatched agents may abandon and leave the market. Ashlagi et al. (2019) considers a market with easy-to-match and hard-to-match agents similarly to the present paper. They focus on asymptotically analyzing the waiting time of agents under myopic policies.

Dynamic matching problems have been also studied as continuous time control problems using queueing framework(Gurvich and Ward (2014); Özkan and Ward (2019); Nazari and Stolyar (2019)). These models use greedy matching policies derived from certain optimization problems, and show that the greedy policies are asymptotically optimal as the system intensity(arrival rates of the market participants) becomes large. The model in Gurvich and Ward (2014) minimizes the holding cost of heterogeneous agents with a general matching compatibility, while Özkan and Ward (2019) and Nazari and Stolyar (2019) study problems of maximizing matching cardinality and general matching rewards, respectively.

More recently, models that considers matching operations as one of control levers to design efficient markets have been developed. Afèche et al. (2019) analyzes the optimal design of bipartite matching compatibility structure in a queueing system. They propose a Pareto optimal policy in a sense that minimizes average customer waiting time and maximizes matching rewards. Kanoria and Qian (2019) considers jointly optimizing matching operation with pricing or entry control in ride-sharing contexts, and proposes a near optimal state-dependent control policy that addresses certain challenges from ride-sharing operations. Johari et al. (2019) proposes a learning-and-earning problem that a platform needs to strike balance between learning about agents' preferences(*exploration*) and maximizing matching rewards(*exploitation*).

In Akbarpour et al. (2019), agents enter the market stochastically, and similarly to our model, unmatched agents leave the market after random amount of times. The key difference is that Akbarpour et al. (2019) assumes homogeneous agents, in the sense that every agent becomes compatible for match with each of others in the market with some fixed probability, while agents in our model are heterogeneous in terms of the(deterministic) matching compatibility structure. Ashlagi et al. (2019) considers heterogeneous agents that are compatible with others with different probabilities, and assumes that agents do not leave the market without being matched, whereas unmatched agents in our model may perish and leave the market.

One of the key features that differentiate our model from most of the works mentioned above is that we consider heterogeneous and impatient agents who leave the platform after random times. In this perspective, the model in Özkan and Ward (2019) is the closest to ours. However, their policy is nearly optimal in the sense that the optimality gap approaches to zero as the system intensity becomes large, while we focus on characterizing the structure of the optimal matching policy given arrival rates of agents. Furthermore, the goal of the present research is providing insights for finding the optimal length of matching intervals in continuous time setting using the results on the discrete time model. To the best of our knowledge, our work is the first to address the problem of finding the optimal length between consecutive matching intervals. In doing so, we build a model that takes as input the length of matching intervals and incorporates its effect on the market thickness through arriving and departure processes of the agents.

In Doval (2019), dynamic stability is defined in two-sided one-to-one matching market where agents arrive over time and the matching is irreversible. Then, the paper shows that dynamically stable matching always exists. Liu (2019) considers repeated two-sided matching market with short-lived agents on one side and long-lived agents on the other side. Applications include matching between medical students and hospitals, in which hospitals accept new students every year. In this setting, Liu (2019) characterizes self-enforcing arrangements and how statically non-stable matching can be implemented. Both of these papers consider agents' incentive to deviate from a given matching in dynamic environment, and the appropriate definition of stability in such setting. In the present paper, the platform has the power to enforce any matching between agents. Moreover, rather than taking a discrete-time matching environment as given, we study the optimal way to build discrete-time problem from a real-time setting.

1.3 Setting

1.3.1 Continuous time setting

We consider a market with two types of agents: easy-to-match(E) and hard-to-match(H) type agents. We assume that each E-type agent can be matched with another E or H-type agent, whereas each H-type agent can only be matched with a E type agent. Every matched pair of agents leaves the market immediately. In the real world, agents may leave the market after certain time of waiting to be matched, because they perish(e.g kidney exchange) or choose options outside the market. To model this behavior, we assume that once an agent enters the platform, she leaves it with exponential rate.

To formalize this setting, we consider a continuous-time two-dimensional Markov chain. First, time is continuous and denoted with $t \in [0, \infty)$. The first state is the number of E-type agents in the platform and the second state is the number of H-type agents in the platform. Assume that arrivals of E-type and H-type agents are independent Poisson processes with constant rates λ_E and λ_H, respectively. Once an agent in the market leave the market with exponential rate of μ, independent of other other agents' exit. Thus, if there is no outside intervention, the number of E-type agent is a birth and death process that jumps from state n to $n+1$ at rate λ_E and jumps to state n to $n-1$ at rate $n\mu$. The same characterization applies to the number of H-type agents in the platform.

The platform's objective is to maximize the expected discounted sum of matching cardinality. If a match is created at time t, the platform gets the payoff of e^{-rt} for a constant $r > 0$. Note that the platform gets the payoff of 1 *regardless* of the type of match it creates. In other words, both a match between two E-type agents and a match between a E-type agent and a H-type agent give the payoff of 1 to the platform. The problem of summing up payoffs over a continuous time will be described in more detail later. It is also important to note that unlike Akbarpour et al. (2019), the platform in our setting does not have any information about future entry or exit of agents. Thus, when the platform decides whether to create a match and what kind of match it will create, its decision is solely based on past history of agents' entrances and exits. Motivated by the practice of kidney exchanges in the United States and other countries, we assume that the planner executes matching among agents for every fixed length of time period, which it commits to follow throughout the whole matching process. This fixed length time period will be referred as *matching interval*, and each of timings that matching is conducted will be referred as *matching period* for the rest of the paper.

When the platform chooses the matching interval, it faces the following trade-off. On one hand, since an existing agent's probability of leaving the market increases by the waiting time, the platform may lose many agents if the matching interval is too large. On the other hand, the platform wants to choose the matching interval long enough so that there are sufficient number of compatible pairs of agents, i.e., the market is thick enough, at every matching period. In this research, we aim to identify the optimal matching interval that strikes the balance between these two effects. Once the platform chooses the matching interval, we can formulate the problem as a discrete time dynamic programming problem, in which matching is conducted in every matching period. In the next section, we assume that the matching period is fixed, and analyze the optimal matching policy for this discrete time dynamic matching problem.

1.3.2 Discrete time setting

To begin with, we first formulate a dynamic program over finite time horizon. For $k = 1, 2, ...,$, we use k to denote the time point that the planner makes a matching decision, when there are k *periods-to-go* onward. The parameters of the discrete time model that we use below depend on μ as well as the matching inverval in the continuous time setting in Section 1.3.1, but we omit the dependence in the notation for brevity.

We assume that the planner uses a stationary policy to choose a feasible matching operation for each period. Immediately after the matching, each of the remaining(unmatched) agents independently leaves the platform with probability $1 - p$ with $0 < p < 1$. Note that assuming the immediate departures of the remaining agents at the post-decision state here is equivalent to the continuous time departures in the original model. After the remaining agents' departures, new E-type and H-type agents enter the market. The number of new arrivals of E-type and H-type agents follow distributions F_E and F_H, respectively, which are known a priori. We let ρ_E and ρ_H be the mean of distributions F_E and F_H, respectively, and assume that $\rho_E < \infty, \rho_H < \infty$.

For $k = 1, 2, ...,$, we let x_k and y_k be the number of E type and H type agents at time k, respectively, and define (x_k, y_k) to be the system state at k. We also let $\Gamma(x_k, y_k)$ be the set of feasible matching decisions given (x_k, y_k), which is defined as

$$\Gamma(x_k, y_k) = \left\{(u, v) \in Z_+^2 : 2u + v \le x_k, v \le y_k, u, v \ge 0\right\},$$

where u is the number of matching between two E-type agents, and v is that of matching between one E-type and one H-type agent.

We define the objective function of the planner to maximize the discounted sum of the number of matched agents, or *the matching cardinality.* A matching decision $(u, v) \in \Gamma(x_k, y_k)$ generates a reward of $u+v$ immediately after the matching operation. Such matching decisions are made according to *matching policies* of the planner. More formally, we let $\pi = (\pi_1, \pi_2, ...)$ be a matching policy where π_i is a function that maps each state $(x, y) \in \mathbb{Z}_+^2$ to a distribution over $\Gamma(x, y)$. In fact, by Proposition 1, we can focus only on stationary and deterministic policies π, that is, each $\pi_i(x, y) \in \Gamma(x, y)$ is a deterministic function. Then, our goal is to find a stationary and deterministic policy that maximizes

$$\mathbb{E}_\pi \left[\sum_{k=1}^{\infty} \delta^k (u(x_k, y_k) + v(x_k, y_k)) \right],$$

where δ is the single-period discount factor, and $\mathbb{E}_\pi[\cdot]$ is the expectation with respect to the stochastic transitions of the system states, (x_k, y_k), when the matching decisions follow a policy π, as well as a distribution on the initial state.

1.3.3 Relationship between discrete and continuous time settings

The discrete time setting defined in the previous section is derived from the continuous time setting. We again emphasize that the discrete time model is derived from the continuous time model by clearing the market in every certain matching period. We define how the key parameters of the continuous model is related to those of the discrete time model and how the matching period t affects this relationship.

First, the discount factor δ represent how utility is discounted between two adjacent times that matching occurs. Since the discount rate in the continuous time model is r, if the market is cleared in every t time period, the corresponding discount factor would be $\delta = e^{-rt}$. Second, the exit probability p in discrete time model is the probability that an unmatched in the current period remains in the platform by the next period. Because an existing agent in the platform exits with Poisson rate μ, if the matching period is t, the probability that an agent does not exist during time t is the probability that a geometric random variable with mean $1/\mu$ is larger than t. Thus, $p = e^{-\mu t}$.

Finally, let us think about the distribution of number of agent of each type enter between two adjacent times that matching occurs. This is more complicated, because there is possibility that an agent entered between these time periods and than exit before the matching happens. Thus, both the rates that agents enter and exits matter in this case. Let $N(t)$ denote the random variable for the number of E-type agents entering and survives during a matching period. Then, its pdf can be calculated as follows.

$$\begin{aligned}\mathbb{P}(N(t) = n) &= \sum_{k \geq n} \frac{(\lambda t)^k}{k!} e^{-\lambda t} \binom{k}{k-n} \left(\int_0^t \frac{1}{t} \left(1 - e^{-\mu(t-s)}\right) ds \right)^{k-n} \left(\frac{1}{t} \int_0^t e^{-\mu(t-s)} ds \right)^n \\ &= \frac{(\lambda t)^n}{n!} e^{-\lambda t} \sum_{k \geq n} \frac{(\lambda t)^{k-n}}{(k-n)!} \left(1 - \frac{1}{\mu t}(1 - e^{-\mu t})\right)^{k-n} \left(\frac{1}{\mu t}(1 - e^{-\mu t})\right)^n \\ &= \frac{(\lambda t \beta(t))^n}{n!} e^{-\lambda t \beta(t)},\end{aligned}$$

where $\beta(t) = (1 - e^{-\mu t})/(\mu t)$. For there to be n agents enter and survive during matching interval of t, no less than n agents should enter in the first place. The total number of agents that entered during the interval of length t is denoted as k in the equation above. Given that an agent enter during this interval, she is equally likely to have entered at every time point. The term $1 - e^{-mu(t-s)}$ is the probability that agent entered at time point s survives until the end of the interval arrives at t. Then, this term is integrated by s from 0 to t with weight $1/t$ to find the unconditional probability that an agent enter and survive during the time interval t. Because agents enter and exit independently, the probability that n agents survive among k agents can be found with binomial distribution. According to this expression, $\rho_E = \lambda_E t \beta(t)$ and $\rho_H = \lambda_H t \beta(t)$.

1.4 Optimal policy: discrete time settings

1.4.1 Convergence of value function iteration

For the ease of exposition later, we also formulate a dynamic program that computes the optimal value function over a finite time horizon of length $k = 1, 2, ...$ Let $V_k(x_k, y_k)$ denote the optimal value, i.e., maximum expected discounted number of matched agents if there are x_k E-type and y_k H-type agents in the initial period. We formulate the dynamic program as follows:

$$\begin{gathered} V_{k+1}(x,y) = \max_{(u,v)\in\Gamma(x,y)} \left(u + v + \delta\mathbb{E}\left[V_k(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v)+\xi_H)\right]\right), \\ V_0(x,y) = 0, \\ \xi_E \sim F_E, \xi \sim F_H, \\ \eta_E(x-2u-v) \sim Binomial(x-2u-v,p), \eta_H(y-v) \sim Binomial(y-v,p). \end{gathered} \tag{1.1}$$

$V_{k+1}(x, y)$ is the discounted optimal value obtained over the $k + 1$ periods-to-go, starting from state (x, y). ξ_E and ξ_H are the number of exogenous arrivals of E-type and H-type agents, respectively. $\eta_E(x - 2u - v)$ and $\eta_H(y - v)$ are random variables that denote the the number of E-type and H-type agents, respectively, that are actually carried over to the next period among the remaining agents after matching according to (u, v). Recall that each of the remaining agents leaves the system independently with probability p, and thus $\eta_E(x - 2u - v)$ and $\eta_H(y - v)$ follow binomial distributions as above.

In this paper, we consider an infinite horizon model, and note that the convergence of the optimal value function (V_k) defined as in (1.1) is not trivial because the dynamic program has an countably infinite state space(Z^2_+), and the single-period reward function($u+v$) is not bounded, and most of well-known convergence results of value functions of Markov Decision Processes(MDP) are stated assuming finite state space and bounded reward functions.

To verify the convergence of V_k, we use Theorem 6.10.4 in Puterman (2005) by showing that our model satisfies certain conditions used in the theorem, which allows us to analyze the MDP with unbounded rewards. We begin our analysis by stating the result in Proposition 1.

Proposition 1 *The following holds for the dynamic program in* (1.1).

1. *There exists an optimal matching policy that is stationary and deterministic.*
2. *The optimal value function V is the unique solution of*

$$V(x,y) = \max_{(u,v)\in\Gamma(x,y)} \left(u + v + \delta\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v)+\xi_H)\right]\right), \tag{1.2}$$

for all $(x,y) \in \mathbb{Z}_+^2$*, where* $\xi_E, \xi_H, \eta_E(x-2u-v), \eta_H(y-v)$ *are independent random variables with distributions* F_E, F_H*, Binomial*$(x-2u-v,p)$*,and Binomial*$(y-v,p)$*, respectively.*

3. *For all* $(x,y) \in \mathbb{Z}_+^2$*,*

$$\lim_{k\to\infty} V_k(x,y) = V(x,y).$$

4. *Any stationary deterministic policy* π *satisfying*

$$\pi(x,y) \in \arg\max_{(u,v)\in\Gamma(x,y)} \left(u+v+\delta\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v)+\xi_H)\right]\right), \quad (1.3)$$

for all $(x,y) \in \mathbb{Z}_+^2$ *is an optimal policy.*

5. *For a stationary deterministic policy* π*,* V_π *is the unique solution of*

$$V_\pi(x,y) = \max_{(u,v)\in\Gamma(x,y)} \left(u+v+\delta\mathbb{E}\left[V_\pi(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v)+\xi_H)\right]\right).$$

Proof. For $(x,y) \in \mathbb{Z}_+^2$, let $w(x,y) = x+y+2(m_E+m_H)$. We show that the MDP with the transition structure and the reward function as in (1.1) satisfies Assumptions 6.10.1 and 6.10.2 of Puterman (2005). Then, the result follows from Theorem 6.10.4 in Puterman (2005). Assumption 6.10.1 in Puterman (2005) is satisfied since

$$\sup_{(u,v)\in\Gamma(x,y)} (u(x,y)+v(x,y)) \le x+y \le w(x,y),$$

where the first inequality follows from the condition defining $\Gamma(x,y)$. To verify that Assumption 6.10.2 holds, note that for all $u,v \in \Gamma(x,y)$,

$$\begin{aligned}
&\sum_{(\tilde{x},\tilde{y})\in\Gamma(x,y)} \mathbb{P}\left(\eta_E(x-2u-v)+\xi_E=\tilde{x}, \eta_H(y-v)+\xi_H=\tilde{y}\right)\cdot w(\tilde{x},\tilde{y})\\
&= m_E+m_H+\sum_{(\tilde{x},\tilde{y})\in\Gamma(x,y)} \mathbb{P}\left(\eta_E(x-2u-v)+\xi_E=\tilde{x}, \eta_H(y-v)+\xi_H=\tilde{y}\right)(\tilde{x}+\tilde{y})\\
&\le m_E+m_H\\
&+\sum_{(\tilde{x},\tilde{y})\in\Gamma(x,y)} \mathbb{P}\left(\eta_E(x-2u-v)+\xi_E=\tilde{x}, \eta_H(y-v)+\xi_H=\tilde{y}\right)\big(x+\xi_E+y+\xi_H\big)\\
&\le 2(x+y+m_E+m_H)\\
&= 2w(x,y),
\end{aligned}$$

and thus Assumption 6.10.2(a) of Puterman (2005) is satisfied. To show that Assumption 6.10.2(b) also holds, take a large integer $J > 1$ and a positive number κ such that $(J+1)\delta^J \le$

$\kappa < 1$. Then, for any finite sequence of policy $\pi = (\pi_1, ..., \pi_J)$ where $\pi_1, ..., \pi_J$ are Markovian deterministic policies, let (X_J, Y_J) be a random vector that denotes the state at time J when the system starts from (x, y) and the matching up to J is determined according to π. Then,

$$\begin{aligned}
&\delta^J \mathbb{E}_\pi \left[w\left(X_J, Y_J\right)\right] \\
&= \delta^J \mathbb{E}_\pi \left[X_J + Y_J + m_E + m_H \right] \\
&\leq \delta^J \left(x + y + (J+1)(m_E + m_H)\right) \\
&\leq \delta^J (J+1) w(x, y) \\
&\leq \kappa \cdot w(x, y),
\end{aligned}$$

where $\mathbb{E}_\pi[\cdot]$ denotes the expectation taken with respect to the distribution of (X_J, Y_J) when the matching is done according to the policy π up to J. The first inequality is true because

$$X_J + Y_J \leq (x + y) + \sum_{s=1}^{J} (\xi_{E,s} + \xi_{H,s}),$$

where $\xi_{E,s}$ and $\xi_{H,s}$ are i.i.d copies of ξ_E and ξ_H, respectively. ■

Proposition 1 implies that, in principle, we can solve the model using value iteration or (modified)policy iteration. However, this is not implementable because an infinite number of operations at each iteration would be required, and approximation approaches, for example, using a truncated model, can be used for computing the optimal value function(see Puterman (2005)). Our goal in the current section is to obtain insights to find the optimal matching interval in the original continuous time setting. In order to do so, by Proposition 1, we can focus on stationary and deterministic policies in the discrete time model, and characterize structures of those policies that are optimal.

1.4.2 Structural properties of optimal policy

We point out some immediate observations on V, which will be frequently used to prove our results in the remainder of the current section.

Observation 1 *(i) $V(x, y)$ is non-decreasing in both arguments.*

(ii) For all $c \geq 0$, $V(x, c - x)$ is non-decreasing in x for $0 \leq x \leq c$.

Observation 1 can be verified by proving it for V_k for all k using induction, and taking the limit $k \to \infty$ to extend the result to V. Observation 1(i) means that the more agents we have, the more agents will be matched. Observation 1(ii) means that given the total number of agents in the market, it is better to have more E-type agents. This is intuitive since a H-type agent can only be matched to an E-type agent, while an E-types agent can be matched to either type of agent.

Next, we analyze optimal solutions for (1.2) and reveal their structural properties. In the sequel, we let $(u^*(x,y), v^*(x,y))$ be the optimal solution for (1.2).

Proposition 2 *The followings hold true for for u^* and v^*.*

(a) If $v^*(x,y) < y$, then $u^*(x,y) = 0$.

(b) $v^*(x,y) = \min(x,y)$.

(c) Suppose $(x_1,y_1),(x_2,y_2) \in \mathbb{Z}_+^2$ with $x_1 - y_1 \geq x_2 - y_2 > 0$. Whenever there are multiple optimal choices for u^*, assume that u^* is chosen to be the smallest value. If $x_1 - y_1$ and $x_2 - y_2$ have the same parity, that is,

$$x_1 - y_1 = x_2 - y_2 + 2q,$$

for some $q \in \mathbb{Z}_+$, then

$$x_1 - y_1 - 2u^*(x_1,y_1) \geq x_2 - y_2 - 2u^*(x_2,y_2).$$

(d) For $x_1 > x_2 > 0$ with the same parity, if $x_2 \geq x_1 - 2u^*(x_1,0)$, then $x_1 - 2u^*(x_1,0) = x_2 - 2u^*(x_2,0)$.

Remark 1.4.1 *By Proposition 2(b), $v^*(x,y)$ is uniquely defined. In the sequel, when there are multiple values for $u^*(x,y)$, we assume that the smallest value is chosen to break the tie, as in Propositon 2(c).*

Proof.

(a) Suppose that $u = u(x,y) > 0$ and $v = v(x,y) < y$ for some $(u,v) \in \Gamma(x,y)$. Using the fact that a binomial random variable is a summation of independent Bernoulli random variables, we can write the continuation value in the next period in (1.2) as

$$\mathbb{E}\left[V(\eta_E(x-2u-v) + \xi_E, \eta_H(y-v-1) + \xi_H + \psi)\right],$$

where ψ is a Bernoulli random variable with success probability p, independent of all other random variables. Now, consider breaking one E-E match and replacing it with a new E-H match. This operation does not change the number of matches in the current period, while there is one more E-type agent and one less H-type agent than the original match. The continuation value of the new matching decision becomes

$$\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E+\psi, \eta_H(y-v-1)+\xi_H)\right].$$

Since all of the random variables are independent of each other in both cases, we can observe that

$$\begin{aligned}
&\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v-1)+\xi_H+\psi)|\psi=0\right] \\
&=\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E+\psi, \eta_H(y-v-1)+\xi_H)|\psi=0\right], \\
&\mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E, \eta_H(y-v-1)+\xi_H+\psi)|\psi=1\right] \\
&\leq \mathbb{E}\left[V(\eta_E(x-2u-v)+\xi_E+\psi, \eta_H(y-v-1)+\xi_H)|\psi=1\right].
\end{aligned}$$

The inequality holds by Observation 1(b). This shows that we can keep weakly improving the matching decision until either $u(x,y)$ decreases to zero or $v(x,y)$ increases to y, and the result follows.

(b) Let $(u_k^*(x,y), v_k^*(x,y))$ be the optimal matching for (1.1). We first prove that $v_k^*(x,y) = \min(x,y)$ for all k and (x,y). If this is the case, then we can write

$$V_k(x,y) = \min(x,y) + V_k(x-\min(x,y), y-\min(x,y)),$$

and the same equation holds for V by taking the limit $k \to \infty$ on the both sides.

To prove the above equation, we use the mathematical induction over k. The equation holds trivially for $k=1$, because if there is only one period remaining, it is optimal to create as many matches as possible. Assume that the claim holds for some $k \geq 1$, which implies that

$$V_k(x+1, y+1) = V_k(x,y) + 1,$$

for all $(x,y) \in \mathbb{Z}_+^2$. This is because in state $(x+1, y+1)$, $\min(x+1, y+1) = 1 + \min(x,y)$ number of E-H matching needs to be done by the induction hypothesis. If $v_{k+1}^*(x,y) = y$, then $x \geq 2u_{k+1}^*(x,y) + v_{k+1}^*(x,y) \geq v_{k+1}^*(x,y) = y$, thus $\min(x,y) = y = v_{k+1}^*(x,y)$. Suppose $v_{k+1}^*(x,y) < y$. By following a similar argument as in the proof of part (a), we can argue that $u_{k+1}^*(x,y) = 0$. Now, assume by contradiction that $x - v_{k+1}^*(x,y) > 0$, i.e., there is a remaining E-type agent after matching agents according to $(0, v_{k+1}^*(x,y))$. For brevity, let $v_{k+1}^* = v_{k+1}^*(x,y)$. We can make one more match of E-H from the remaining agents, and the value-to-go of this action is equal to

$$\begin{aligned}
&v_{k+1}^* + 1 + \delta\mathbb{E}\left[V_k(\eta_E(x-v_{k+1}^*-1)+\xi_E, \eta_H(y-v_{k+1}^*-1)+\xi_H)\right] \\
>&v_{k+1}^* + \delta\mathbb{E}\left[V_k(\eta_E(x-v_{k+1}^*-1)+\xi_E, \eta_H(y-v_{k+1}^*-1)+\xi_H)+1\right] \\
=&v_{k+1}^* + \delta\mathbb{E}\left[V_k(\eta_E(x-v_{k+1}^*-1)+\xi_E+1, \eta_H(y-v_{k+1}^*-1)+\xi_H+1)\right] \\
\geq&v_{k+1}^* + \delta\mathbb{E}\left[V_k(\eta_E(x-v_{k+1}^*-1)+\xi_E+\psi_E, \eta_H(y-v_{k+1}^*-1)+\xi_H+\psi_H)\right] \\
=&V_k(x,y),
\end{aligned}$$

where ψ_E, ψ_H follow $Bernoulli(p)$, and all the random variables are independent. The equality follows from the implication of the induction hypothesis mentioned above. The second inequality is due to Observation 1. The last equality is by the optimality of v^*_{k+1}, and the fact that $\eta_E(x - v^*_{k+1} - 1) + \psi_E \overset{d}{=} \eta_E(x - v^*_{k+1})$ and $\eta_H(y - v^*_{k+1} - 1) + \psi_H \overset{d}{=} \eta_H(y - v^*_{k+1})$. This implies that the value-to-go in period $k+1$ can be strictly improved, which is contradictory to the optimality of $(0, v^*_{k+1})$. It follows that $x = v^*_{k+1}(x, y) < y$, and thus $v^*_{k+1}(x, y) = \min(x, y)$.

(c) It is trivial when $x_1 - y_1 = x_2 - y_2$. Suppose $x_1 - y_1 = x_2 - y_2 + 2q$ for some integer $q \geq 1$. Define a function $h(w)$ as the continuation value when w E-type agents and zero H-type agents are carried to the next period:

$$h(w) = E[V(\eta_E(w) + \xi_E, \xi_H)]$$

Let $w^*(x_1, y_1) = x_1 - y_1 - 2u^*(x_1, y_1)$ and $w^*(x_2, y_2) = x_2 - y_2 - 2u^*(x_2, y_2)$. By Part (b), we know that $v^*(x_1, y_1) = y_1, v^*(x_2, y_2) = y_2$. Assume by contradiction $w^*(x_1, y_1) < w^*(x_2, y_2)$. From the optimality, we have

$$\begin{aligned}
&\frac{x_1 - y_1 - w^*(x_1, y_1)}{2} + y_1 + \delta h(w^*(x_1, y_1)) \\
&> \frac{x_1 - y_1 - w^*(x_2, y_2)}{2} + y_1 + \delta h(w^*(x_2, y_2)), \\
&\frac{x_2 - y_2 - w^*(x_2, y_2)}{2} + y_2 + \delta h(w^*(x_2, y_2)) \\
&\geq \frac{x_2 - y_2 - w^*(x_1, y_1)}{2} + \delta h(w^*(x_1, y_1)),
\end{aligned}$$

where the first inequality is strict since we break ties of u^* by choosing the smallest value(hence the largest value of w^*). This implies

$$\frac{w^*(x_2, y_2) - w^*(x_1, y_1)}{2} > \delta(h(w^*(x_2, y_2)) - h(w^*(x_1, y_1))) \geq \frac{w^*(x_2, y_2) - w^*(x_1, y_1)}{2},$$

which is a contradiction.

(d) Without loss of generality, we prove the claim when $x_1 = 2q + 2, x_2 = 2q$ for some

$q \in \mathbb{Z}_+$. Note that we can write $V(2q+2,0)$ in terms of $V(2q,0)$ as the following:

$$
\begin{aligned}
&V(2q+2,0)\\
&= \max_{0\le u\le q+1}\Big\{u + \delta\mathbb{E}\big[V(\eta_E(2q+2-2u)+\xi_E,\xi_H)\big]\Big\}\\
&= \max\Big\{\delta\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big]\\
&, 1 + \max_{0\le u-1\le q}\Big\{(u-1) + \delta\mathbb{E}\big[V(\eta_E(2q-2(u-1))+\xi_E,\xi_H)\big]\Big\}\Big\}\\
&= \max\Big\{\delta\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big], 1+V(2q,0)\Big\}.
\end{aligned}
$$

The second term on the right-hand side is the maximum value that can be obtained from matching at least a pair of E-type agents in the current period. It follows that if $u^*(2q+2,0) > 0$ and $2q+2-2u^*(2q+2,0) \le 2q$, then $u^*(2q+2,0) = 1+u^*(2q,0)$, and

$$(2q+2) - u^*(2q+2,0) = 2q - u^*(2q,0).$$

We can prove a similar claim for $x_1 = 2q+1, x_2 = 2q-1$ for $q \in \mathbb{Z}_+ \setminus \{0\}$.

■

The result in Proposition 2(b) is intuitive since each E-H matching uses one E-type agent while each E-E matching uses two of them. Recall that H-type agents can contribute to the matching reward by being matched with E-type agents and the matching rewards of the two types of matchings are homogeneous. To maximize the matching reward, it is optimal to first execute as many E-H matching as possible, each of which consumes one less "flexible"(E-type) agents than E-E matching. Then, the number of E-E matching, if there are any remaining E-type agents, is done so as to maximize the value-to-go given the state with positive E-type agents and zero H-type agents.

For any state $(x,y) \in \mathbb{Z}_+^2$, we can now write the optimal matching decision as

$$
\begin{aligned}
&v^*(x,y) = \min(x,y),\\
&u^*(x,y) = \arg\max_{0\le u\le\lfloor(x-\min(x,y))/2\rfloor} V\big(x-\min(x,y), y-\min(x,y)\big).
\end{aligned}
$$

In order to find $u^*(x,y)$, we need to be able to compute $u^*(x,0)$ for $x \in \mathbb{Z}_+$. Proposition 2(c) and (d) reveal some structural properties of u^*. Suppose there are $x = 2q$ or $x = 2q+1$ E-type agents available after executing E-H matches myopically. Proposition 2(c) states that the number of carried-over E-type agents(after executing E-E matching) should be monotone in q. Moreover, Proposition 2(d) implies that if $u^*(x,0) \ge 1$, $u^*(x',0)$ for all x'

with $x - 2u^*(x,0) \le x' \le x$ satisfies

$$u^*(x',0) = x' - x + 2u^*(x,0).$$

Although Proposition 2 provides important structural properties of u^*, we need more information on the structures of u^* to be able to completely characterize the optimal matching policy, if possible.

It turns out that when we need to be *myopic* enough, in the sense that $p\delta$ is small, the number of E-type agents carried over to the next period is uniformly bounded. Intuitively, if p is small, any carried-over E-type agent is likely to perish and not available in the next period, and it may be better off to myopically execute E-E matching as many as possible in the current period. When δ is small enough, matching reward obtained in future is significantly discounted, and benefit from potential matches in future by carried-over E-type agents is less than the immediate benefit from matching them in the current period.

We formally state the above result in Proposition 3. We first present two technical lemmas, Lemma 1.4.2 and Lemma 1.4.3 used in the proof of the proposition.

Lemma 1.4.2 *For any* $q \in \mathbb{Z}_+$,

(i) $V(2q+1,0) - V(2q,0) \le p\delta$,

(ii) $V(2q+2,0) - V(2q+1,0) \le 1$.

Proof.

(i) We can write

$$\begin{aligned}
&V(2q+1,0) - V(2q,0)\\
&= \max_{0\le u\le q}\left\{u + \delta\mathbb{E}\big[V(\eta_E(2q+1-u)+\xi_E,\xi_H)\big]\right\}\\
&- \max_{0\le s\le q}\left\{s + \delta\mathbb{E}\big[V(\eta_E(2q-s)+\xi_E,\xi_H)\big]\right\}\\
&= \max_{0\le u\le q}\left\{u + \delta\mathbb{E}\big[V(\eta_E(2q+1-u)+\xi_E,\xi_H)\big] - \max_{0\le s\le q}\left\{s + \delta\mathbb{E}\big[V(\eta_E(2q-s)+\xi_E,\xi_H)\big]\right\}\right\}\\
&\le \max_{0\le u\le q}\left\{\delta\mathbb{E}\big[V(\eta_E(2q+1-u)+\xi_E,\xi_H)\big] - \delta\mathbb{E}\big[V(\eta_E(2q-u)+\xi_E,\xi_H)\big]\right\}\\
&= \delta\max_{0\le u\le q}\mathbb{E}\big[V(\eta_E(2q-u)+\psi+\xi_E,\xi_H) - V(\eta_E(2q-u)+\xi_E,\xi_H)\big],
\end{aligned}$$

where *psi* is an independent *Bernoulli*(p) random variable. It follows that

$$
\begin{aligned}
&V(2q+1,0) - V(2q,0) \\
&\le p\delta \max_{0\le u\le q} \mathbb{E}\big[V(\eta_E(2q-u)+1+\xi_E,\xi_H) - V(\eta_E(2q-u)+\xi_E,\xi_H)\big] \\
&\le p\delta \max_{0\le u\le q} \mathbb{E}\big[V(\eta_E(2q-u)+1+\xi_E,\xi_H+1) - V(\eta_E(2q-u)+\xi_E,\xi_H)\big] \\
&= p\delta \max_{0\le u\le q} \mathbb{E}\big[1+V(\eta_E(2q-u)+\xi_E,\xi_H) - V(\eta_E(2q-u)+\xi_E,\xi_H)\big] \\
&= p\delta,
\end{aligned}
$$

where the second inequality is from Observation 1, and the equality is true due to Proposition 2.

(ii) As in the proof of Proposition 2(d), we can write

$$
\begin{aligned}
&V(2q+2,0) - V(2q+1,0)) \\
&= \max\Big\{1+V(2q,0), \delta\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big]\Big\} - V(2q+1,0) \\
&= \max\Big\{1+V(2q,0)-V(2q+1,0), \delta\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big] - V(2q+1,0)\Big\}.
\end{aligned}
$$

By Observation 1, $1+V(2q,0)-V(2q+1,0) \le 1$. To bound the second term, note that

$$
\begin{aligned}
&\delta\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big] - V(2q+1,0) \\
&\le \delta\Big(\mathbb{E}\big[V(\eta_E(2q+2)+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q+1)+\xi_E,\xi_H)\big]\Big) \\
&= \delta\Big(\mathbb{E}\big[V(\eta_E(2q)+\psi_1+\psi_2+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q)+\psi_1+\xi_E,\xi_H)\big]\Big) \\
&= \delta\Big(p^2\Big(\mathbb{E}\big[V(\eta_E(2q)+2+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big]\Big) \\
&+ p(1-p)\Big(\mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q)+\xi_E,\xi_H)\big]\Big)\Big)
\end{aligned}
$$

We can write

$$
\begin{aligned}
&p^2\Big(\mathbb{E}\big[V(\eta_E(2q)+2+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big]\Big) \\
&\le p^2\Big(\mathbb{E}\big[V(\eta_E(2q)+2+\xi_E,\xi_H+1)\big] - \mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big]\Big) \\
&= p^2\Big(1+\mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big] - \mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big]\Big) \\
&= p^2,
\end{aligned}
$$

where the first equality follows from Proposition 2, and ψ_1, ψ_2 follow *Bernoulli*(p), independent of all other random variables. Likewise,

$$p(1-p)\bigg(\mathbb{E}\big[V(\eta_E(2q)+1+\xi_E,\xi_H)\big]-\mathbb{E}\big[V(\eta_E(2q)+\xi_E,\xi_H)\big]\bigg)\le p(1-p).$$

It follows that

$$V(2q+2,0)-V(2q+1,0)\le\max\{1,\delta(p^2+p(1-p))\}=1.$$

■

Lemma 1.4.3 *For $\xi_1 \sim Poisson(\rho_1)$, $\xi_2 \sim Poisson(\rho_2)$, and $\eta_n \sim Binomial(n,p)$, $Pr(\xi_1 + \eta_n \ge \xi_2) \to 1$ as $n \to \infty$ for all $\rho_1, \rho_2 > 0$, and $p \in (0,1]$.*

Proof. Consider the following set of inequalities.

$$\begin{aligned}\mathbb{P}\left(\xi_1+\eta_n\ge\xi_2\right) &\ge \mathbb{P}\left(\eta_n\ge\xi_2\right)\\ &\ge \mathbb{P}\left(\xi_2\le n(1-\epsilon)p\right)\mathbb{P}\left(\eta_n\ge n(1-\epsilon)p\right)\end{aligned}$$

As $n \to \infty$, $n(1-\epsilon)p \to \infty$, and $\mathbb{P}\left(\xi_2 \le n(1-\epsilon)p\right)$ goes to 1 by the Monotone Convergence Theorem. By the law of large numbers, $\frac{\eta_n}{n} \to p$ in probability as $n \to \infty$, and thus $\mathbb{P}\left(\eta_n \ge n(1-\epsilon)p\right) \to 1$ as $n \to \infty$. ■

Proposition 3 *Let $(x,y) \in \mathbb{Z}_+^2$ with $x > y$. By Proposition 2(b), $v^*(x,y) = y$. If $F_E \sim Poisson(\rho_E)$ and $F_H \sim Poisson(\rho_H)$ for $\rho_E, \rho_H > 0$, the followings hold true for the optimal matching between E-type agents, $u^*(x,y) = u^*(x-y,0)$:*

(a) If $p \le \frac{1}{2\delta}$, $u^*(x,y) = \lfloor \frac{x-y}{2} \rfloor$ for all $(x,y) \in \mathbb{Z}_+^2$ with $x > y$, i.e., it is optimal to myopically match as many the remaining E-type agents as possible.

(b) If $\frac{1}{2\delta} < p < \frac{\sqrt{5}-1}{2\delta}$, there exists some positive integer $\bar{w}$ such that

$$w^*(x,y) = x-y-2u^*(x,y) \le \bar{w},$$

for all $(x,y) \in \mathbb{Z}_+^2$ with $x > y$.

Proof. When $x - y = 1$, we have $u^*(x,y) = 0$ and $v^*(x,y) = y$ by part (b). Assume $x - y \ge 2$.

(a) By Proposition 2(b), we can write

$$\begin{aligned}V(x,y) &= y + V(x-y,0)\\ &= y + \max_{0\le u\le \lfloor \frac{x-y}{2}\rfloor} \{u + \delta\mathbb{E}[V(\eta_E(x-y-2u)+\xi_E,\xi_H)]\}\\ &= y + \max_{0\le u\le \lfloor \frac{x-y}{2}\rfloor} \{u + \delta h(x-y-2u)\},\end{aligned}$$

where $h(\cdot)$ is defined as in the proof of Proposition 2(c). To show that the maximum is attained at $\lfloor \frac{x-y}{2}\rfloor$, we show that $h(w+1)-h(w)\le \frac{1}{2\delta}$ for all feasible z. If this is the case, then

$$h(z+2)-h(z) = (h(z+2)-h(z+1)) + (h(z+1)-h(z)) \le \frac{1}{\delta}.$$

That is, the expected gain of sparing one pair of E-type agents for the next period is less than or equal to 1, and thus it is (weakly) better off to match the pair at the current period. Note that

$$\begin{aligned}&h(w+1)-h(w)\\ &= \delta\left(\mathbb{E}\left[V(\eta_E(w)+\psi+\xi_E,\xi_H)\right] - \mathbb{E}\left[V(\eta_E(w)+\xi_E,\xi_H)\right]\right)\\ &= \delta p\left(\mathbb{E}\left[V(\eta_E(w)+1+\xi_E,\xi_H)\right] - \mathbb{E}\left[V(\eta_E(w)+\xi_E,\xi_H)\right]\right),\end{aligned}$$

where $\eta_E(w)\sim \mathit{Binomial}(w,p)$, $\xi_E\sim F_E$, $\xi_H\sim F_H$, and $\psi\sim \mathit{Bernoulli}(p)$, and all the random variables are independent. Let $p(m,n,l)$ be the joint probability mass function of $(\eta_E(w),\xi_E,\xi_H)$. Then,

$$\begin{aligned}h(w+1)-h(w) &= p\delta\sum_{(m,n,l)} p(m,n,l)\left(V(m+n+1,l)-V(m+n,l)\right)\right) \qquad (1.4)\\ &\le p\delta\sum_{(m,n,l)} p(m,n,l)\cdot 1\\ &= p\delta\\ &\le \frac{1}{2}.\end{aligned}$$

To see why the first equality holds, note that by Observation 1 and Proposition 2,

$$\begin{aligned}&V(m+n+1,l)-V(m+n,l)\\ &\le V(m+n+1,l+1)-V(m+n,l)\\ &= 1+V(m+n,l)-V(m+n,l)\\ &\le 1.\end{aligned}$$

This concludes the proof.

(b) We prove the claim by showing that

$$\lim_{w\to\infty}\left(h(w+2)-h(w)\right)<\frac{1}{\delta}.$$

If this is the case, then there exists $\bar{w}$ such that $h(w+2)-h(w)<1/\delta$ for all $w\geq\bar{w}$, which implies that the marginal expected matching reward of a pair of E-type agents is strictly smaller than that of immediate matching in the current period. Then, it is never optimal to carry over more than $\bar{w}$ E-type agents.

For $w\in\mathbb{Z}_+$,

$$\begin{aligned}
&h(w+2)-h(w)\\
&=\mathbb{E}\left[V\left(\eta_E(w)+\psi_1+\psi_2+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+\xi_E,\xi_H\right)\right]\\
&=p^2\left(\mathbb{E}\left[V\left(\eta_E(w)+2+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+\xi_E,\xi_H\right)\right]\right)\\
&+2p(1-p)\left(\mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+\xi_E,\xi_H\right)\right]\right)\\
&=p^2\left(\mathbb{E}\left[V\left(\eta_E(w)+2+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right]\right)\\
&+p(2-p)\left(\mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+\xi_E,\xi_H\right)\right]\right),
\end{aligned}\tag{1.5}$$

where ψ_1,ψ_2 follow *Bernoulli*(p), and all the random variables are independent. Using Proposition 2(b), the first term in (1.5) can be written as

$$\begin{aligned}
&\mathbb{E}\left[V\left(\eta_E(w)+2+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right]\\
=&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1=\xi_H\big)\big(V(1,0)-V(0,0)\big)\right]\\
+&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1<\xi_H\big)\big(1+V(0,\xi_H-\xi_E-\eta_E(w)-1)-V(0,\xi_H-\xi_E-\eta_E(w))\big)\right]\\
+&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1>\xi_H\big)\big(V(\xi_E+\eta_E(w)+2-\eta_H,0)-V(\xi_E+\eta_E(w)+1-\eta_H,0)\big)\right].
\end{aligned}\tag{1.6}$$

The first and second term in (1.6) vanish as $w\to\infty$ since

$$\begin{aligned}
&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1=\xi_H\big)\big(V(1,0)-V(0,0)\big)\right]\\
+&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1<\xi_H\big)\big(1+V(0,\xi_H-\xi_E-\eta_E(w)-1)-V(0,\xi_H-\xi_E-\eta_E(w))\big)\right]\\
\leq&\left(1+V(1,0)-V(0,0)\right)\mathbb{P}\left(\xi_H\geq\xi_E+\eta_E(w)\right),
\end{aligned}$$

by Observation 1, and the right-hand side converges to zero as $z\to\infty$ by Lemma 1.4.3. To bound the third term in (1.6), note that when $\xi_E+\eta_E(w)+1-\eta_H>0$,

$$\begin{aligned}
&V(\xi_E+\eta_E(w)+2-\eta_H,0)-V(\xi_E+\eta_E(w)+1-\eta_H,0)\\
&=\begin{cases}V(2q+1,0)-V(2q,0)\leq p\delta & ,\xi_E+\eta_E(w)+1=\eta_H+2q,q\in\mathbb{Z}_+\setminus\{0\}\\ V(2q+2,0)-V(2q+1,0)\leq 1 & ,\xi_E+\eta_E(w)+1=\eta_H+2q+1,q\in\mathbb{Z}_+,\end{cases}
\end{aligned}$$

by Lemma 1.4.2. Then, we can write

$$
\begin{aligned}
&\mathbb{E}\left[1\big(\xi_E+\eta_E(w)+1>\xi_H\big)\big(V(\xi_E+\eta_E(w)+2-\xi_H,0)-V(\xi_E+\eta_E(w)+1-\xi_H,0)\big)\right]\\
\le&p\delta\cdot\mathbb{P}\big(\xi_E+\eta_E(w)+1=\xi_H+2q \text{ for some } q\in\mathbb{Z}_+\setminus\{0\}\big)\\
+&\mathbb{P}\big(\xi_E+\eta_E(w)+1=\xi_H+2q+1 \text{ for some } q\in\mathbb{Z}_+\big)\\
=&p\delta\cdot\mathbb{P}\big(\xi_E+\eta_E(w)=\xi_H+2q+1 \text{ for some } q\in\mathbb{Z}_+\big)\\
+&\mathbb{P}\big(\xi_E+\eta_E(w)=\xi_H+2q \text{ for some } q\in\mathbb{Z}_+\big).
\end{aligned}
$$

To bound the right-hand side, note that

$$
\begin{aligned}
&\mathbb{P}\left(\eta_E(w)+\xi_E=\xi_H+2q+1 \text{ for some } q\in\mathbb{Z}_+\right)\\
&=\sum_{n\le w,\text{n: odd}}\mathbb{P}\left(\xi_H-\xi_E=n\right)\mathbb{P}\left(\eta_E(w)=n+2q+1 \text{ for some } q\in\mathbb{Z}_+\right)\\
&+\sum_{n\le w,\text{n: even}}\mathbb{P}\left(\xi_H-\xi_E=n\right)\mathbb{P}\left(\eta_E(w)=n+2q+1 \text{ for some } q\in\mathbb{Z}_+\right)\\
&=\mathbb{P}(\eta_E(w) \text{ is even})\mathbb{P}(\xi_H-\xi_E\le w,\xi_H-\xi_E \text{ is odd})\\
&+\mathbb{P}(\eta_E(w) \text{ is odd})\mathbb{P}(\xi_H-\xi_E\le w,\xi_H-\xi_E \text{ is even}) \qquad (1.7)\\
&=q(w)r_o(w)+(1-q(w))(r(w)-r_o(w)), \qquad (1.8)
\end{aligned}
$$

where $q(w)=\mathbb{P}(\eta_E(w) \text{ is even})$, $r(w)=\mathbb{P}(\xi_H-\xi_E\le w)$, and $r_o(w)=\mathbb{P}(\xi_H-\xi_E\le w,\xi_H-\xi_E \text{ is odd})$. Likewise, we can write

$$\mathbb{P}\left(\eta_E(w)+\xi_E=\xi_H+2q,q\in\mathbb{Z}_+\right)=(1-q(w))r_o(w)+q(w)(r(w)-r_o(w)).$$

Since $\eta_E(w)$ is a binomial random variable and $0<p<1$, we can show that

$$q(w)=(1+(1-2p)^w)\to\frac{1}{2},$$

as $w\to\infty$. Also,

$$\lim_{w\to\infty}r(w)=1,\ \lim_{w\to\infty}r_o(w)=\mathbb{P}(\xi_H-\xi_E \text{ is odd})=:r_o,$$

by the Monotone Convergence Theorem(**reference**). It follows that

$$
\begin{aligned}
&\lim_{w\to\infty}\mathbb{P}\left(\eta_E(w)+\xi_E=\xi_H+2q+1 \text{ for some } q\in\mathbb{Z}_+\right)\\
&=\lim_{w\to\infty}\mathbb{P}\left(\eta_E(w)+\xi_E=\xi_H+2q \text{ for some } q\in\mathbb{Z}_+\right)\\
&=\frac{1}{2}. \qquad (1.9)
\end{aligned}
$$

Taking $\limsup_{w\to\inf}$ on the both sides in (1.6), we have

$$\limsup_{w\to\infty} \mathbb{E}\left[V\left(\eta_E(w)+2+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right] \\ \leq \frac{1}{2}\big(1+p\delta\big).$$

For the second term in (1.5), we can use a similar approach to show that

$$\limsup_{w\to\infty} \mathbb{E}\left[V\left(\eta_E(w)+1+\xi_E,\xi_H\right)\right]-\mathbb{E}\left[V\left(\eta_E(w)+\xi_E,\xi_H\right)\right] \\ \leq \frac{1}{2}\big(1+p\delta\big).$$

Plugging the above results into (1.5), we obtain

$$\limsup_{w\to\infty} \big(h(w+2)-h(w)\big) \\ \leq \frac{1}{2}\Big(p^2\big(1+p\delta\big)\big)+p(2-p)\big(1+p\delta\big)\Big) \\ = p\big(1+p\delta\big).$$

The result follows by finding the condition in which $p(1+p\delta)<1/\delta$.

■

Combining the results in Proposition 2 and Proposition 3, we can characterize the optimal matching policy when $p\delta<(\sqrt{5}-1)/2$:

Theorem 1 *Let $(x,y)\in\mathbb{Z}_+^2$. The optimal matching policy when $p\delta<(\sqrt{5}-1)/2$ is described as the following:*

(i) $v^*(x,y)=\min(x,y)$, i.e., execute as many E-H matching as possible.

(ii) If $x>y$ and $p\delta\leq\frac{1}{2}$, myopically execute E-E matching on the remaining E-type agents, that is,

$$u^*(x,y)=\left\lfloor\frac{x-y}{2}\right\rfloor.$$

(iii) If $\frac{1}{2}<p\delta<\frac{\sqrt{5}-1}{2}$, there exists an integer $\bar{w}$ such that for any $x\geq\bar{w}$, and the largest integer $\hat{x}=x-2q\leq\bar{w}$ with $q\in\mathbb{Z}_+$, then

$$u^*(x,0)=u^*(\hat{x},0)+q.$$

Proof. The results in part (i) and (ii) are immediate from Proposition 2 and Proposition 3. The result in part(iii) is obtained by combining Proposition 2(d) and Proposition 3(b). ■

Theorem 1 shows that when $p\delta$ is not large enough the decision maker should not carry over too many E-type agents in anticipation of thick market and better matches in future. When $1/2 < p\delta < (\sqrt{5}-1)/2$, the result in Theorem 1(iii) implies that we only have to compute $u^(x, 0)$ for $x = 0, 1, ..., \bar{w}$ to dictate the optimal policy.

1.5 Numerical Experiments

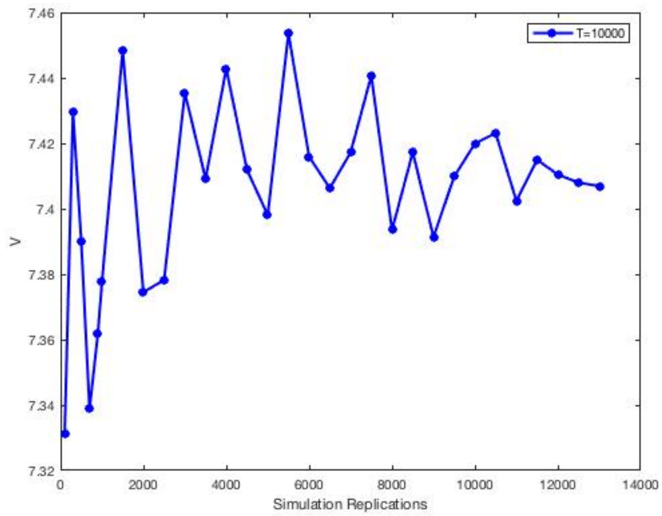
In this section, we conduct some numerical experiments that verify the theoretical results in the previous sections and provide insights for further analysis of the model. As alluded earlier, our goal of the present research is to find the optimal length of matching intervals that strikes the right balance between the two opposite effects on the market thickness: letting in more agents make the market thicker, while losing some agents due to their impatience.

In the first experiment, we show the convergence of the value function of the myopic matching policy. In every period in the simulation, we execute as many E-H matching as possible first, and if there are remaining E-type agents, execute as many E-E matching as possible. We use $p = 0.5$ and $\delta = 0.9$. Since $p\delta < 1/2$, the myopic matching policy is optimal by Theorem 1, and Figure 2.1 indirectly shows the convergence result of the optimal value function, V. The result is shown in Figure 2.1. The horizontal axis represents the value of the simulation replications. The vertical axis represents the estimate of the expected matching cardinality of the myopic policy, which was computed by taking sample average over the replications.

To verify the optimality of the myopic matching policy, we could directly use the value function iteration as in Proposition 1, and show that the value converges to the same value as that of the myopic policy in Figure 2.1. Since the model has countable state space, we could use finite-state approximation approach as in Puterman (2005), and analyze error bounds of the approximation using the structure of our model. This is a direction of on-going works on the present research, but we do not include the result in this dissertation.

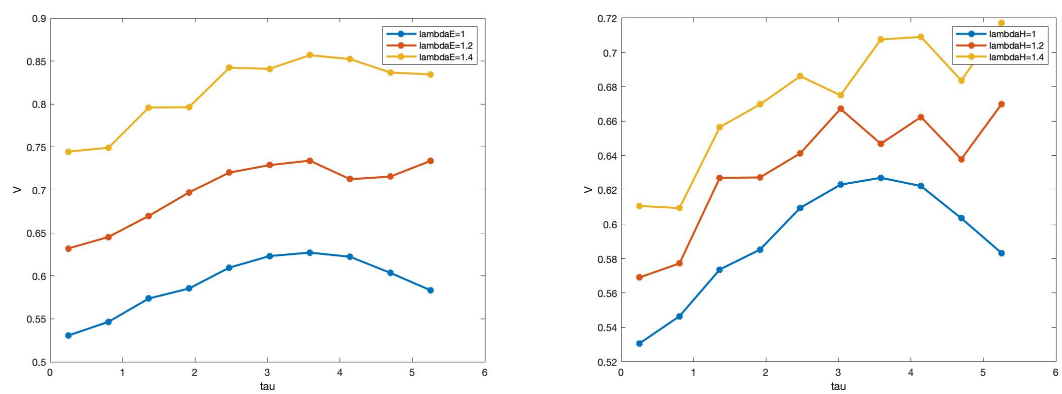
Next, we investigate numerically how the optimal expected matching cardinality varies for different lengths of matching intervals. In the following, we let τ denote the length of matching intervals. Also, we let $r = \mu = 1$, so $p\delta = e^{-2}\frac{1}{2}$ and the myopic matching policy is optimal by Theorem 1. For each case, we run simulations for $T = 10,000$ periods with $B = 2,000$ replications. Finally, the simulation was run for 10 different lengths of matching intervals, uniformly spaced between 0.25 and 5.25.

Figure 1.2 shows the matching cardinality obtained for different lengths of matching intervals. Figure 1.2a shows the results for the case that $\lambda_H = 1$ and $\lambda_E = 1, 1.2, 1.4$. Figure



(a) $\lambda_E = \lambda_H = 1$

Figure 1.1 Convergence of the value of the myopic policy

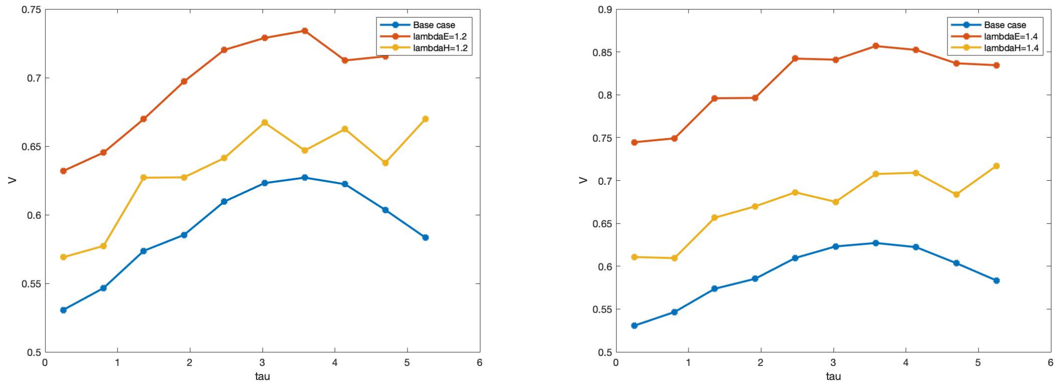


(a) $\lambda_H = 1$ (b) $\lambda_E = 1$

Figure 1.2 Matching cardinality by interval: comparison by entrance rates

1.2a shows the results for the case that $\lambda_E = 1$ and $\lambda_H = 1, 1, 2, 1.4$. A few important observations can be made here. First, the matching cardinality increases for small values of $\tau < 4$ for all cases, suggesting that there is gain from waiting for matching market to be thickened at the cost of agents' abandonment. Second, as Observation 1 suggests, the matching cardinality increases with λ. In other words, the faster agents enter the platform, the larger the matching cardinality becomes.

Figure 1.3 is a rearrangement of Figure 1.2. Figure 1.3a compares two cases that $\lambda_E = 1.2, \lambda_H = 1$ and $\lambda_E = 1, \lambda_H = 1.2$. Figure 1.3b compares two cases that $\lambda_E = 1.4, \lambda_H = 1$ and



(a) $\lambda_E = 1.2$ v.s. $\lambda_H = 1.2$ (b) $\lambda_E = 1.4$ v.s. $\lambda_H = 1.4$

Figure 1.3 Matching cardinality by interval: comparison by types

$\lambda_E = 1, \lambda_H = 1.4$. For both of the cases, the matching cardinality is larger when we increase λ_E compared to when we increase λ_H by the same degree. This confirms the intuition in Observation 1 that if the arrival rates increases by the same amount, the matching cardinality is larger when the arrival rate for E-type agent increases than when the arrival rate of H-type agent does.

1.6 Conclusion and future Directions of research

In this research, motivated by practices of carpooling product of ride-sharing platforms and kidney exchange, we developed a model for dynamically matching agents that have heterogeneous compatibility and stay in the market only for finite amount of time. In particular, our model takes as input a static length of time intervals between consecutive matching operations, which was inspired by matching execution with fixed time intervals in kidney exchange. The model is formulated as a discrete time Markov Decision Process(MDP) parametrized by the length of matching intervals with unbounded single-period reward. By showing that the MDP satisfy certain regularity conditions, we first proved that the value iteration converges, which also allowed us to focus on the class of optimal and stationary policies. We also characterized some of structures of optimal policies depending on the extent to which the platform needs to be myopic. We showed that it is optimal in every period to first execute matches between an easy-to-match agent and a hard-to-match agent in greedy fashion. Regarding the matches among any remaining easy-to-match agents, we showed that there is a bound on the number of the agents carried over to the next period, provided that the platform needs to be at least moderately myopic.

To obtain further insights in analyzing the optimal length of matching intervals, we conducted several numerical experiments. Based on these results and more extensive numerical study, we plan to analyze additional theoretical aspects such as bounds on the optimal length of matching intervals, structural properties of the value function(e.g unimodularity) as we vary the length of matching intervals. Another potential direction may be using simulation optimization approach(Marbach and Tsitsiklis (2001)). For example, when $p\delta \leq \frac{1}{2}$, myopically matching agents according to Theorem 1 is optimal, and thus the system state can be modeled as a Markov Reward Process whose transition dynamics is parametrized by the length of matching intervals.

Also, we want to conduct more fine-grained analysis on the discrete time model. In Proposition 3 and Theorem 1, it is still not clear if we can obtain structural properties of the optimal matching policy when $p\delta > \frac{\sqrt{5}-1}{2}$. Computing the value of $\bar{w}$ in Proposition 3 may be also an interesting direction of complementing our results in the discrete time setting. The value iteration in Proposition 1 needs to be done most likely by using finite-state approximation(Puterman (2005)). We want to propose a concrete procedure for such an approximation, analyze the error bound using the structure of our model, and verify that the iteration converges in a reasonable amount of time through extensive numerical experiments.

As an independent but related problem, we plan to dive deep into performance analysis of myopic matching policies. As many dynamic matching models mentioned in Section 1.2 and our main result in Theorem 1 suggest, the optimal matching policy may be computationally challenging or sensitive to the parameters. Since myopic policies are easy to implement, one may consider using such matching policies if the performance of such policies are at reasonable level. To this end, we hope to extend our model to a more general version with impatient and heterogeneous agents, and analyze performance bounds of myopic policies in a certain family of the models.

Chapter 2

Service Zone Design for Elevator Systems in High-Rises

2.1 Introduction

In this research, we study the problem of zoning design for elevators in high-rises. Our model is intended to compute conservative performance measures of a given elevator system, and provide insights for capacity planning of the elevator system using zoning approach.

In the recent decades, many cities around the world have become highly populated and urbanized, and it has been increasing important to build urban infrastructures that can deal with highly densed population more efficiently. Especially, due to technological advances in elevator industry and civil engineering sector, there have been increasing number of commercial and residential high-rises world-wide, and efficient and robust design of the buildings have become crucial accordingly.

There has been notable advancement in the hardware technology for elevators, which enabled buildings go as high as 1km(Rottweil and Tytyri (2017)). In particular, many of the high-rises around the world are mostly used for commercial purposes, and it is essential for elevator systems in such high-rises to be able to serve heavy traffic of demands during rush hours with certain performance guarantee. Inefficient control and design of elevator systems may critically harm the key performance measures such as average waiting time for elevators or total time to process certain amount of customers. As the scale of buildings increases, such suboptimal control and design can cause significant inconvenience and disutility of people who want to use elevators.

One of frequently used approaches for designing efficient elevator systems is *zoning*, which is assigning a designated set of floors to each elevator. Given a zoning configuration, elevators can only pick up and drop off customers at the designated floors. When every elevator can serve every floor, i.e., no zoning configuration, then it is possible that multiple elevators that are not fully capacitated make frequent stops at common floors, resulting in long service times and inefficiency of the system. The idea of zoning is to prevent different elevators from stopping at the same floors, by assigning destinations that the elevators can reach *a priori*, and thus "forcing" only customers whose destinations are the same or adjacent to each other take the same elevator.

In order to analyze the system performance given a zoning configuration, travel times of elevators need to be directly incorporated to a model. The analysis can be challenging due to several key components of elevator systems: heterogeneous demands for different floors, customer batching, and dependence of travel time of elevators on the demand profiles of the customer batches. It is crucial to capture how these three components interact and affect the system performance to determine the optimal zoning configuration. However, to the best of our knowledge, there is no framework that takes into account these components. Existing methods find a zoning configuration that minimize the estimate of round-trip times of elevators, which is obtained by using aggregate demands and somewhat crude estimates of factors that determine the travel times of elevators.(Barney (2015))

In this research, we want to develop a mathematical model for the zoning design that take into account demand patterns, customer batching, and the resulting service times of elevators. Our model can provide performance guarantee of elevator system to cope with congested traffic of customers, and thus can be used for capacity planning of elevator systems.

We consider conventional elevator systems where a customer's destination is not known until she gets into an elevator and click a button of her destination. The system cannot observe individuals' destinations in advance, and cannot control the combination of demands in each batch of customers. To isolate the effect of customer batching on service times of elevators, we assume that there is an adversary who knows customers' destinations and consolidates them to form batches so as to make the system performance as bad as possible. The adversarial framework not only makes the model tractable but also provides conservative performance measusre that can be used for capacity planning of elevator systems.

We consider an uppeak traffic situation where most of customers want to take elevators from the ground level to move to upper floors. In general, uppeak traffic is the heaviest traffic that a building can experience during the day, and thus the performance measure during the uppeak traffic situation is used to estimate the general system capacity in practice(Barney (2015)). If an elevator system performs well during uppeak traffic situation, it can deal with other types of traffic such as interfloor during lunch time or downpeak traffic during evening rush hour reasonably well.

The rest of the paper is organized as follows. In Section 2.2, we review the relevant literature. In Section 2.3, we explain the general setting of our model, propose a framework to analyze the performance of different zoning configurations, and provide some analytical results. In Section 2.4, we show results from various numerical study. In Section 2.5, we summarize the present work and propose future directions of research.

2.2 Literature review

There has been a stream of research on elevator dispatching/group control problems. This stream of research has addressed elevator dispatching problems in different contexts(e.g double-deck elevators, destination control systems) as new elevator products become available in the market by technological advances. Most of this research studies the problem of assigning online arrivals of a request while a group of elevators are moving along routes to serve previously assigned tasks. The online assignment problem is done by reoptimizing mixed-integer linear programs or network flow problems. Due to the combinatorial nature of the elevator dispatching problems, various methods have been employed including genetic algorithms(Chen et al. (2012); Fujino et al. (1997); Tyni and Ylinen (2006); Koh et al. (1999)), fuzzy logic(Kim et al. (1998); Ishikawa et al. (2000); Imasaki et al. (1995); Sasaki et al. (1991)), and neural networks(Kenji et al. (1996); Liu and Liu (2007); Stanley et al. (2007)). Tanaka et al. (2005a,b) formulate a dynamic program for control of a single elevator with destination control systems, and construct a branch-and-bound algorithm. Other than elevator dispatching problems, there is one interesting research on the problem of optimally parking multiple idle elevators(Parlar et al. (2006)), where the authors investigate state-dependent myopic policies.

Zoning problems were typically solved by genetic algorithms(Oh and Park (1995); Cortés et al. (2004); Tyni and Ylinen (2006)) to minimize the average waiting time or the average riding time(Bolat et al. (2010)). Ishihara and Kato (2013) studies a problem of dynamically assigning zones to multiple elevators in each shaft, and evaluate a heuristic policy using simulation. In Liu et al. (2010), optimal zoning strategy for energy saving is proposed in interfloor traffic situation, solved by a genetic algorithm, and the strategy is tested on a simulation platform. Yang et al. (2009) proposes a dynamic program formulation of dynamic zoning with destination control systems, and use fuzzy neural networks to implement the optimal policy.

Unlike these research on zoning approach that optimize the average waiting/riding time using (meta-)heuristics, our model provides the performance guarantee through the worst-case analysis of customer batching, and the solution method uses simple greedy algorithms and linear programs, which makes the result easy to interpret.

Some interesting early research on the elevator dispatching problem appeared in machine learning community. In Crites and Barto (1998); Touretzky et al. (1996), the authors modeled the problem as a multi-agent reinforcement learning problem, and perform Q-learning using a simple feedfoward neural network. Pepyne and Cassandras (1997) formulated the problem as a Markov Decision Process(MDP), proved that optimal policies are threshold-type policies, and showed how to compute the threshold. Their model is extended to Pepyne and Cassandras (1998) to an adaptive model using concurrent estimation.

Our model is similar to that of Pepyne and Cassandras (1997) in that we also focus on uppeak traffic situation and the system performance depends on round-trip time of elevators. However, Pepyne and Cassandras (1997) assumes that the round-trip time of elevators follows i.i.d exponential distribution with some fixed rate, and does not depend on the demand profile of the customer batch of each dispatch. On the contrary, we let the round-trip time for a batch depend on the number of distinct destinations and the highest floor to reach, and thus provide more realistic to be used for measuring the performance of the system.

Service zoning problems have been also studied in vehicle routing/dispatching literature. Bertsimas and Ryzin (1993a) studies *Traveling Repairman Problem*, where multiple vehicles with finite capacity need to visit uniformly distributed demand locations. The authors propose various policies to group the demands and design routes of the vehicles, and analyze average customer waiting time of the policies under heavy traffic situation using queueing theory. Bertsimas and Ryzin (1993b) extends the analysis to a setting with a general distribution of the demand. Carlsson and Delage (2013) and Carlsson et al. (2018) use distributionally robust optimization to solve problems of partioning a metric space, where each of the regions is then assigned to a vehicle that travels to serve customer locations within it. In all of the models in Bertsimas and Ryzin (1993a,b); Carlsson and Delage (2013); Carlsson et al. (2018), the demand locations are in the two-dimensional Euclidean space, and the well-known BHH theorem(Beardwood et al. (1959)) is used to approximate the "service time" of vehicles and simplify the analysis. In our model, since elevators move in one-dimensional shafts, the tour length of a round-trip is just a simple linear function of the highest floor that the elevator reaches. However, we need to incorporate the number of stops over the finite metric space(set of floors) into the round-trip time of the elevator, which complicates the cost function of the model.

2.3 Model

In this section, we propose an adversarial optimization framework to help the DM choose the optimal zoning configuration for the elevator dispatching system. As alluded earlier, our goal is to build a model that can measure the capacity the elevator system. In general, uppeak traffic during moring rush hour, when most of the people want to travel from the ground level to their destinations, is the heaviest traffic that elevator systems experience, and thus the system can handle traffic during all day well if it can deal with the uppeak traffic situation reasonably well.

We consider a building with two elevators and a single entrance at the ground level. We assume the uppeak traffic situation during morning rush hour, and most of people who want to use the elevators make requests and take the elevators at the lobby to travel to upper floors. Since we are considering on the uppeak traffic situation during morning rush hour, when most of the people want to get in to work, we assume that traffic of customers who want to take elevators at floors other than the lobby(ground level) is negligible.

In analyzing the influence of people's travel pattern on the performance of the system, we focus on traveling times of elevators to take batches of customers, drop them off at their destinations, and return to the lobby. We call the time for an elevator to make such a cycle *round-trip* time. When the traffic is intense as in morning rush hour, elevators will have to take a *batch* of passengers and stop at possibly many different destinations to drop them off, which will affect the patterns of round-trip times. To model this, we will assume that the mean round-trip time depends on the *demand profile* of the batch, that is, the exact combination of floors that the elevator needs to stop and drop off customers.

We assume a conventional hall call button system where customers can request an elevator at the lobby by pushing the up-call button upon their arrivals. We assume that all customers do not consider options of other transportation(e.g. escalators, stairs), and that they want to get off at their exact destinations. We also assume that customers form lines in front of elevators that they want to take, and that an elevator takes a batch of customers on each round-trip.

2.3.1 Setting

We consider a building with two elevators and K floors above the ground level or lobby. We let $k = 1, ..., K$ be the kth floor from the ground level. We assume that the building has a single entrance at the ground level, and the elevators can be only accessed from the lobby.

As alluded earlier, we want to build a model that can analyze the performance of the elevator system facing congested uppeak traffic during morning rush hour. In order to do this, we assume in our model that all customers are present at the lobby at the beginning of the system. This model is motivated by a situation in which a large number of customers arrive at the lobby around the same time during the morning rush hour, and there are no major incoming flow of people afterward. A similar model was used in Hu and Benjaafar (2009) to study partioning of servers in a queueing framework.

In real-world, analyzing the performance of elevator dispatching system even during just in uppeak traffic situation can be complicated by three components: (i) batching service of customers, (ii) dependence of round-trip times on the demand profile of batches, and (iii) stochastic in-flow of customers. In our model, we abstract away the last component and use stylizing assumptions on the customer demands. In this research, we want to propose a stylized model that captures the effect of demand profiles of batches on the system performance, and to build a framework with which the zoning decision can be made with certain performance guarantee.

For $k = 1, 2, ..., K$, we assume that there are $d_k = r_k C$ customers at the lobby who want to move up to floor k for some $r_k \in \mathbb{Z}_+$, i.e., demands are deterministic, integer multiples of the elevator capacity, and all of the customers are present at the lobby at the beginning of the system. Although these assumptions may be quite stylizing, they allow us to incorporate the batching service and the functional relationship between round-trip times and demand profiles, which will be specified later. Despite the very unique structure of the latter, the model remains tractable due to the assumptions and our modeling choice on the demands. From now on, we call customers whose destinations are $k = 1, ..., K$ type-k customers.

Before the system starts, the DM chooses the set of floors that each elevator will serve, that is, each elevator can stop and drop off customers only at the designated set of floors. In our model, the DM may let both elevators serve every floor, which we call *no-zoning* configuration, or may the entire floors into two subsets of consecutive floors and designate one elevator for each to serve, which we call *zoning* configuration.

Given the zoning decision made by the DM a priori, we assume that customers form a single queue in front of each elevator that serves their destinations. We assume that both elevators have limited capacity of $C \in \mathbb{Z}_+$, and an elevator can take customers up to its capacity. As soon as the customers form the queues, the elevators take customers in batches and start round-trips to drop them off. Each queue is served as a batch in FCFS basis. The customer at the head of a queue always requests the elevator immediately if she could not get into the previous batch due to the limited capacity.

During morning rush hour, it is likely that elevators take customers as batches up to the capacity. Round-trip time of an elevator depends on the set of floors that the elevator needs to visit, i.e., demand profile of the batch. More specifically, the main components that determine the round-trip time of an elevator are how many times the elevator needs to stop and drop off customers and how high the elevator needs to reach. The former is equal to the number of distinct destinations of customers in the batch, and the latter is the highest floor that a customer in the elevator is going to, which is called the *highest reversal floor* of the batch.

Although the actual functional relationship between the round-trip time and the demand profile of a batch can be highly nonlinear due to operational constraints of elevators such as acceleration, deceleration and jerk, we assume that round-trip time of a batch is a linear function of the number of distinct destinations and the highest reversal floor of the batch. In practice, such linear relationship is frequently used to measure the performance of elevator systems(Barney (2015)).

To be more specific, when an elevator servers a batch of customers that has m distinct destinations and has the highest reversal floor of h, the round-trip time of the batch, say τ, is defined as

$$\tau = ah + bm + \epsilon, \tag{2.1}$$

where a is the average time for the elevator to move a single floor, and b is the average time for the elevator to operate the door and transfer customers, and ϵ is a random noise with mean zero independent of m and h.

In conventional elevator system where a customer's destination is not known before she gets into an elevator and click a button of her destination, the system does not control the batching process, and the DM needs to make the zoning decision by taking into account the uncertainty in customer batching in order to guarantee performance of the elevator system. Given queues of customers in front of elevators given the zoning configuration of the elevators, we assume that an adversary who knows the customers' destinations permute themin each queue, and thus determines the exact demand profiles and orders of batches for each elevator. The adversary's goal is to maximize the total time to transfer all of the customers in the lobby, which will be defined more preciesely later. In anticipation of such adversarial batch formation, the DM's goal is to choose to whether to use zoning, and if he does, which set of floors to assign to each elevator.

In the remainder of the section, we explain the adversary's optimization problems to determine demand profiles of the customers in each queue, given the DM's decision on the zoning configuration.

2.3.2 Zoning case

In this section, we formulate the adeversary's problem of allocating customers over the batches when the DM chooses to use zoning. We consider non-overlapping zoning designs, i.e., the designated floors of the elevators do not have intersections. We let $z \in \{1, 2, ..., K-1\}$ be the DM's zoning decision, and assume that the DM choose z before the beginning of the morning rush hour. Given z, elevator 1 and 2 can only pick up customers who are going from the lobby to floors $1, 2, ..., z$ and $z+1, ..., K$, respectively. We assume that all customers take the elevator that can drop them off at their exact destinations according to the zoning configuration.

When the system starts at the beginning of morning rush hour, we assume that every customer can observe the zoning configuration, and join a queue in front of the elevator that serves her destination. We assume that an elevator takes a full batch of customers, i.e., exactly C customers, which is a reasonable assumption since we are considering congested situation during morning rush hour.

We let

$$n_1(z) = r_1 + \cdots + r_z \ , n_2(z) = r_{z+1} + \cdots + r_K,$$

be the number of batches that elevator 1 and 2 have to serve, respectively. We assume that the elevators start service and let the first batches immediately after the customers form queues in front of the elevator. Since our model assumes a given and deterministic set of customers present in the system from the beginning, it is sensible to define the total time to move all customers as the performance measure of the system.

Given the DM's zoning decision $z \in \{1, ..., K-1\}$, a queue of $n_i(z)C$ customers is formed in front of elevator i, $i = 1, 2$. Since the customers are served on FCFS in batches, the order of customers' destinations determine the sequence of batches to be served by each elevator, which will determine round-trip times, and thus the total time to finish serving all the assigned batches. As alluded earlier, we assume that the customers are permutated by the adversary who is aware of the customers' destinations so as to maximize the total time to transfer all customers in the queue. Using the adversarial batching model, we can obtain a conservative performance measure to digest congested traffic of customers, which can be viewed as a performance guarantee in capacity planning context of elevator systems.

For $i = 1, 2,,$ $j = 1, ..., n_i(z)$, we let B_{ij} be the set of destinations(demand profile) of the jth batch to be served by elevator i, which is determined by the adversary. We also let $H(B_{ij})$ and $S(B_{ij})$ be the highest reversal floor and the number of stops of B_{ij}. From (2.1), the expected time for elevator i to process all customer batches given z and (B_{ij}) can be written as

$$\sum_{j=1}^{n_i(z)} \left(aH(B_{ij}) + bS(B_{ij})\right), \tag{2.2}$$

by the linearity of expectation. The adversary's objective given z would be then allocating customers to the batches to maximize (2.2) for each $i = 1, 2$. Note that the adversary's customer allocation problems for the two elevators are separated by the DM's zoning decision z. To define the system performance measure, we let

$$\begin{aligned} f_1(z) &:= \max_{(B_{1j})} \sum_{j=1}^{n_1(z)} \left(aH(B_{1j}) + bS(B_{1j})\right), \\ f_2(z) &:= \max_{(B_{2j})} \sum_{j=1}^{n_2(z)} \left(aH(B_{2j}) + bS(B_{2j})\right), \end{aligned} \tag{2.3}$$

where the maximums on the right-hand sides are taken over all possible demand profiles of the batches given the customers in the queues. Recall that we define the total time to transfer all of the customers as the performance measure, which is similar to the idea of makespan in parallel machine scheduling literature. The "makespan" of the elevator dispatching system can then be written as

$$f(z) := \max\{f_1(z), f_2(z)\}.$$

Given the zoning configuration z and the corresponding queues of customers, the adversary's goal is to permutate customers, or equivalently, allocate them to the batches, so as achieve $f_1(z)$ and $f_2(z)$. Then, the best zoning decision that the DM can make is

$$z^* \in \arg \min_{z=1,...,K-1} f(z).$$

In order to find z^*, we first formulate mixed-integer linear programs(MILP), $P_1(z)$ and $P_2(z)$, to compute $f_1(z)$ and $f_2(z)$, respectively, as the following.

Definition 2.3.1 *Given the DM's zoning decision $z \in \{1, 2, ..., K-1\}$, for $i = 1, 2$, we define $P_i(z)$ to be the adversary's problem to allocate the customers over batches in front of elevator i, and $f_i(z)$ becomes the optimal objective value of $P_i(z)$. $P_1(z)$ is formulated as the following, and that of $P_2(z)$ is analogous:*

$$(P_1(z)) \quad \max_{\mathbf{x}_1,\mathbf{y}_1,\mathbf{w}_1} \quad \sum_{j=1}^{n_1(z)} \left(aw_{1j} + b \sum_{k=1}^{z} x_{1jk} \right)$$

$$s.t. \quad x_{1jk} \le y_{1jk} \ , j = 1, ..., n_1(z), k = 1, ..., z \ , \tag{2.4}$$

$$y_{1jk} \le Cx_{1jk} \ , j = 1, ..., n_1(z), k = 1, ..., z \ , \tag{2.5}$$

$$\sum_{k=1}^{z} y_{1jk} \le C \ , j = 1, ..., n_1(z), \tag{2.6}$$

$$\sum_{j=1}^{n_1(z)} y_{1jk} = d_k \ , k = 1, ..., z, \tag{2.7}$$

$$w_{1j} \le kx_{1jk} \ , j = 1, ..., n_1(z), k = 1, ..., z \ , \tag{2.8}$$

$$x_{1jk} \in \{0, 1\}, \ , j = 1, ..., n_1(z), k = 1, ..., z \ ,$$

$$y_{1jk} \in \mathbb{Z}_+, \ , j = 1, ..., n_1(z), k = 1, ..., z \ ,$$

$$w_{1j} \ge 0 \ , j = 1, ..., n_1(z).$$

In $P_1(z)$, x_{1jk} indicates whether any type k customer is allocated to batch j of elevator 1, and y_{1jk} is the number of type k customers allocated to batch j. w_{1j} at an optimal solution is the highest reversal floor of. Constraint (2.4) and (2.5) are used to ensure that elevator stops at a floor if and only if there is a customer in the batch who is getting off at the floor. Constraints (2.11) and (2.12) represent capacity and total demand restricitions, respectively, and (2.13) is used to make sure that w_{1j} becomes the highest reversal floor of batch j at optimal solutions.

Let $(\mathbf{x}_1^*, \mathbf{y}_1^*, \mathbf{w}_1^*)$ and $(\mathbf{x}_2^*, \mathbf{y}_2^*, \mathbf{w}_2^*)$ be optimal solutions for $P_1(z)$ and $P_2(z)$, respectively. It is easy to see that $a \sum_{j=1}^{n_1(z)} w_{1j}^*$ is the expected total time that elevator 1 spends to travel to the highest reversal floors for serving the *adversarial batches*. Also, $b \sum_{j=1}^{n_1(z)} \sum_{k=1}^{z} x^*_{1jk}$ is the expected total time that elevator 1 needs to spend to stop and drop off customers in the adversarial batches. Then, we can write

$$f_1(z) = \sum_{j=1}^{n_1(z)} \left(aw^*_{1j} + b \sum_{k=1}^{z} x^*_{1jk} \right), \tag{2.9}$$

$$f_2(z) = \sum_{j=1}^{n_2(z)} \left(aw^*_{2j} + b \sum_{k=z+1}^{K} x^*_{2jk} \right).$$

From (**??**), the DM's problem to find an optimal zoning configuration can be written as

$$\min \left\{ \min_{z=1,\ldots,K-1} \left(\max\{f_1(z), f_2(z)\} \right), f^{nz} \right\}, \tag{2.10}$$

where f^{nz} is the makespan the no-zoning case, i.e., both of the elevators can serve every floor. We will cover the no-zoning case and formulate a MILP to compute f^{nz} in the next section.

To begin our analysis, we show in the following lemma that $f_i(z)$ can be computed by first replacing $P_i(z)$ with a smaller problem for $i = 1, 2$.

Lemma 2.3.2 *$f_i(z)$ is equal to the optimal objective value of the following problem, $\bar{P}_i(z)$:*

$$(\bar{P}_i(z) \quad \underset{\mathbf{x}_i, \mathbf{w}_i}{max} \quad \sum_{j=1}^{n_i(z)} \left(aw_{ij} + b \sum_{k=1}^{z} x_{ijk} \right)$$

$$s.t. \quad \sum_{k=1}^{z} x_{ijk} \leq C \ , j = 1, ..., n_i(z), \tag{2.11}$$

$$\sum_{j=1}^{n_i(z)} x_{ijk} \leq d_k \ , k = 1, ..., z, \tag{2.12}$$

$$w_{ij} \leq k x_{ijk} \ , j = 1, ..., n_i(z), k = 1, ..., z \ , \tag{2.13}$$

$$x_{ijk} \in \{0, 1\}, \ , j = 1, ..., n_i(z), k = 1, ..., z \ ,$$

$$w_{ij} \geq 0 \ , j = 1, ..., n_1(z).$$

Proof.

Let $(\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{w}}_i)$ and $(\bar{\mathbf{x}}_i, \bar{\mathbf{w}}_i)$ be optimal solutions for $P_i(z)$ and $\bar{P}_i(z)$, respectively. Also, let $\bar{f}_i(z)$ be the objective function value of $(\bar{\mathbf{x}}_i, \bar{\mathbf{w}}_i)$ in $\bar{P}_i(z)$. Note that $(\hat{\mathbf{x}}_i, \hat{\mathbf{w}}_i)$ is feasible for

$P_i(z)$. Also, the objective functions of $P_i(z)$ and $\bar{P}_i(z)$ are identical, which does not depend on the value of $\hat{\mathbf{y}}$. It follows that $f_i(z) \leq \bar{f}_i(z)$.

To show that $\bar{f}_i(z) \leq f_i(z)$ also holds, we show that there exists some $\bar{\mathbf{y}}_i$ that makes $(\bar{\mathbf{x}}_i, \bar{\mathbf{y}}_i, \bar{\mathbf{w}}_i)$ feasible for $P_i(z)$. Note that for any customer type k_1, if $\sum_{j=1}^{n_i(z)} \hat{x}_{ijk_1} < d_{k_1}$ and $\hat{x}_{ij_1} = 0$ for some batch j_1, then $\sum_k \hat{x}_{ij_1k} = C$, i.e., batch j_1 must have been fully capacitated. This is because otherwise, we would have been able to strictly increase the objective value of $\bar{P}_i(z)$ by letting $\hat{x}_{ij_1k_1} = 1$, which is a contradiction to the optimality of $\bar{\mathbf{x}}_i$. This implies that if we want to finsh allocating the remaining customers who have not yet been assigned to any batches according to $\bar{\mathbf{x}}_i$, any such customer could be allocated only to a batch that already has the same type of customer assigned by $\bar{\mathbf{x}}_i$. Since the total remaining capacity across the batches should be equal to the number of the remaining customers, there exists a feasible allocation $\bar{\mathbf{y}}_i$ of customers that is consistent with $\bar{\mathbf{x}}_i$. i.e., $\bar{y}_{iik} > 0$ if and only if $\bar{x}_{ijk} = 1$. Then, $(\bar{\mathbf{x}}_i, \bar{\mathbf{y}}_i, \bar{\mathbf{w}}_i)$ becomes a feasible solution for $P_i(z)$, and its objective function value in $P_i(z)$ is the same as $\bar{f}_i(z)$, which implies $\bar{f}_i(z) \leq f_i(z)$. This concludes the proof. ■

Next, we further simplify how to solve $\bar{P}_i(z)$, $i = 1, 2$, for a given $z \in \{1, ..., K-1\}$. As the first step, we prove that (w^*_{ij}), the highest reversal floors of the adversarial batches for elevator i, can be found by a greedy allocation of the customers as in Algorithm 1. Starting from the highest type customers available, the algorithm assigns to each batch a single customer whose type(destination) becomes the highest type of the batch. The result is stated in Lemma 2.3.3, and the proof uses a swapping argument.

Lemma 2.3.3 *For $i = 1, 2$, (w^*_{ij}) can be obtained by Algorithm 1, and is unique up to permutation over the batch indices, $j = 1, ..., n_i(z)$.*

Proof. Algorithm 1 begins by allocating a single customer of the highest type available to each batch, starting from the first batch $j = 1$ through batch $j = n_i(z)$. Given $z \in \{1, ..., K-1\}$, the highest type available that the algorithm starts the allocation from is type-z if $i = 1$ and type-K if $i = 2$. If all of such customers have been allocated and there are still empty batches, the algorithm switches to the next highest type customers available, continues the allocation, and repeats the procedure until every batch has been assigned a customer who determines the highest reversal floor. This implies that the output of Algorithm 1 is unique up to the permutation over j's.

Now, assume by contradiction that there exists an optimal solution $(\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{w}}_i)$ for $P_i(z)$ such that $(\hat{\mathbf{w}}_i)$ is not a permutation of the output of Algorithm 1. Then, there must be two batches, say j_1 and j_2 such that

$$\hat{w}_{ij_1} < \hat{w}_{ij_2},$$

, i.e., the highest floor of batch j_1 is lower than that of batch j_2, and the latter has more than one customer going to $\hat{w}_{ij_2}$, its highest reversal floor. Let $k_1 := \hat{w}_{ij_1}$ and $k_2 := \hat{w}_{ij_2}$. Choose any customer from batch j_1 and swap her with one type-k_2 customer in batch j_2, and let $\tilde{\mathbf{x}}_i$ be a vector of the new "demand profile" of the batches. Since the highest type in batch j_2 is still $k_2(\tilde{x}_{ik_2} = 1)$, we can take $\tilde{w}_{ij_2} = \hat{w}_{ij_2} = k_2$. It is also easy to see that $\sum_k \hat{x}_{ij_2k} \geq \sum_k \tilde{x}_{ij_2k}$. On the other hand, for batch j_1, the swapping operation keeps the number of distinct stops of batch j_1 the same, i.e., $\sum_k \hat{x}_{ij_1k} = \sum_k \tilde{x}_{ij_1k}$, and more importantly, the highest reversal floor of the batch is now increased from k_1 to k_2 since $\tilde{x}_{ij_1k_2} = 1$. As a result, the value of the objective function of $P_i(z)$ is strictly increased, which is contradictory to the optimality of $(\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{w}}_i)$. ■

Algorithm 1: Greedy assignment of the highest reversal floors for elevator i, $i = 1, 2$

Result: w^*_{ij}, x^*_{ijk}, $j = 1, ..., n_i(z).$, $k = 1, ..., z$(if $i = 1$), $k = z + 1, ..., K$(if $i = 2$).
Set all $x^*_{ijk} = 0$. Set $\hat{d}_1 = d_1, ..., \hat{d}_K = d_K$. ;
If $i = 1$, set $k = z$. If $i = 2$, set $k = K$.;
Set $j = 1$.;
while $j \leq n_i(z)$ **do**
  while $\hat{d}_k = 0$ **do**
    $k \leftarrow k - 1$;
  end
  $w^*_{ij} \leftarrow k$;
  $x^*_{ijk} \leftarrow 1$;
  $\hat{d}_k \leftarrow \hat{d}_k - 1$;
  $j \leftarrow j + 1$;
end

Lemma 1 shows how the highest reversal floors, (w^*_{ij}) of the adversarial batches can be obtained given the DM's zoning decision z. Next, in order to compute (x^*_{ijk}), we use the output of Algorithm 1. For $i = 1, 2$, we let $A_i(z)$ be the set of customer type-batch pairs that are not "used" by Algorithm 1 to set the highest reversal floors. That is,

$$A_1(z) = \{(j, k) : 1 \leq k \leq z, 1 \leq j \leq n_1(z), x^*_{1jk} = 0, \hat{d}_k > 0\},$$
$$A_2(z) = \{(j, k) : z + 1 \leq k \leq K, 1 \leq j \leq n_2(z), x^*_{2jk} = 0, \hat{d}_k > 0\},$$

and let $B_i(z)$ be the set of customer types that can be still allocated to some batch after performing Algorithm 1:

$$B_1(z) = \{1 \leq k \leq z : (j, k) \in A_1(z) \text{ for some } j\},$$
$$B_2(z) = \{z + 1 \leq k \leq K : (j, k) \in A_2(z) \text{ for some } j\}.$$

Let $H_i(z)$, $i = 1, 2$ be the summation of the highest reversal floors of the adversarial batches obtained in Lemma 2.3.3:

$$H_1(z) := \sum_{j=1}^{n_1(z)} w^*_{1j}, \tag{2.14}$$

$$H_2(z) := \sum_{j=1}^{n_2(z)} w^*_{2j}.$$

From Lemma 2.3.3, we know that $\mathbf{x}^*_i$ needs to satisfy that $x^*_{ijk} = 1$ for $(j, k) \notin A_i(z)$, which implies that we can solve $\bar{P}_i(z)$ by plugging $x^*_{ijk} = 1$ for $(j, k) \notin A_i(z)$ and (w^*_{ij}) into the original constraints, and maximize $\sum\sum_{(j,k)\in A_i(z)} x_{ijk}$. Then, we can write

$$f_i(z) = aH_i(z) + b\left(n_i(z) + \sum\sum_{(j,k)\in A_i(z)} x^*_{ijk}\right), i = 1, 2, \tag{2.15}$$

where (x^*_{ijk}) denotes an optimal solution of $P_i(z)$ as before. The first term on the right-hand side is the total time for elevator i to travel up to the highest reversal floors of the adversarial batches, and the second term is the total time for stopping and dropping off customers at their destinations. In Proposition 2.3.4, we show that the summation in the parenthesis can be computed by solving a linear program.

Proposition 2.3.4 *For $i = 1, 2$, (x^*_{ijk}), $(j, k) \in A_i(z)$ is an optimal solution of the following linear program:*

$$\begin{aligned} \max \quad & \sum\sum_{(j,k)\in A_i(z)} x_{ijk} \\ s.t. \quad & \sum_{j:(j,k)\in A_i(z)} x_{ijk} \leq \hat{d}_k \ , k \in B_i(z) \\ & \sum_{k:(j,k)\in A_i(z)} x_{ijk} \leq C - 1 \ , j = 1, ..., n_i(z) \\ & 0 \leq x_{ijk} \leq 1 \ , (j, k) \in A_i(z) \end{aligned}$$

Proof. By Lemma 2.3.2, Lemma 2.3.3 and the argument following the latter, we know that

$(x^*_{ijk}), (j,k) \in A_i(z)$ is an optimal solution for

$$\begin{aligned} \max \quad & \sum\sum_{(j,k)\in A_i(z)} x_{ijk} \\ \text{s.t.} \quad & \sum_{j:(j,k)\in A_i(z)} x_{ijk} \le \hat{d}_k \ , k \in B_i(z) \\ & \sum_{k:(j,k)\in A_i(z)} x_{ijk} \le C-1 \ , j = 1, ..., n_i(z) \\ & x_{ijk} \in \{0,1\} \ , (j,k) \in A_i(z). \end{aligned}$$

It is easy to verify that the coefficient matrix of the constraints in the above problem is totally unimodular. Therefore, its optimal solution can be obtained by solving the LP relaxation, and the result follows. ■

Combining everything above, we show how to compute $f_i(z)$, time for elevator i to serve all the adversarial batches given the zoning decision z.

Proposition 2.3.5 *For* $z \in \{1, ..., K-1\}$,

$$f_i(z) = aH_i(z) + b\left(S_i(z) + n_i(z)\right),$$

where $H_i(z)$ *is given as* (2.14) *and* $S_i(z)$ *is defined as*

$$\begin{aligned} S_i(z) = \max & \sum\sum_{(j,k)\in A_i(z)} x_{ijk} \\ s.t. & \sum_{j:(j,k)\in A_i(z)} x_{ijk} \le \hat{d}_k \ , k \in B_i(z) \\ & \sum_{k:(j,k)\in A_i(z)} x_{ijk} \le C-1 \ , j = 1, ..., n_i(z) \\ & 0 \le x_{ijk} \le 1 \ , (j,k) \in A_i(z). \end{aligned}$$

Proof. The result follows by combining (2.9), Lemma 2.3.3, (2.14), (2.15), and Proposition 2.3.4. ■

Finally, we show how to compute $z^* \in \{1, ..., K-1\}$, the optimal decision among the non-overlapping zoning configurations. The proof uses the result of Proposition 2.3.5 to show the unimodularity of $f_i(z)$.

Theorem 2.3.6 *The optimal zoning* $z^* \in \{1, ..., K-1\}$ *can be found as the following:*

(i) If $aH_1(1) + bS_1(1) \geq aH_2(1) + bS_2(1)$*, then* $z^* = 1$.

(ii) If $aH_1(K-1) + bS_1(K-1) \leq aH_2(K-1) + bS_2(K-1)$*, then* $z^* = K-1$.

(iii) (Unimodularity) If $aH_1(1) + bS_1(1) < aH_2(1) + bS_2(1)$ *and* $aH_1(K-1) + bS_1(K-1) > aH_2(K-1) + bS_2(K-1)$*, then there exists* $z^* \in \{1, ..., K-1\}$ *such that* $z \mapsto \max\{aH_1(z)+bS_1(z), aH_2(z)+bS_2(z)\}$ *is non-increasing on* $z \leq z^*$ *and non-decreasing on* $z \geq z^*$.

Proof. From (2.14), it is easy to see that $H_1(z)$ and $H_2(z)$ are non-decreasing and non-increasing in z, respectively. Next, we claim that $S_1(z)$ and $S_2(z)$ are also non-decreasing and non-increasing in z, respectively. To see this, suppose that the DM wants to compare $S_1(z)$ and $S_1(z+1)$ for some $z < K-1$. There are r_{z+1} additional batches for elevator 1 to serve by increasing z by 1. Since there are $r_{z+1}C$ customers of type-$(z+1)$ with $C \geq 1$, at least r_{z+1} batches will have assigned type-$(z+1)$ customers in Algorithm 1. This implies that $A_1(z) \subset A_1(z+1)$ and thus $B_1(z) \subset B_1(z+1)$. Furthermore, for all $k \in B_1(z)$, the value of $\hat{d}_k$ in the linear program for $S_1(z+1)$ is greater than or equal to that of $\hat{d}_k$ in the linear program for $S_1(z)$. From the structure of the linear programs, it follows that $S_1(z+1) \geq S_1(z)$. We can show $S_2(z+1) \leq S_2(z)$ analogously. Therefore, $aH_1(z) + bS_1(z)$ and $aH_2(z)+bS_2(z)$ are non-decreasing and non-increasing in z, respectively, and the results in the theorem follow. ■

2.3.3 No-zoning: two elevators

In the no-zoning case, every elevator can take and drop off customers of all destinations. On one hand, the load to the system is now handled by the two identical 'servers', each of which serves batches consisting of all types of customers. On the other hand, the batches can contain any types of customers, and this may cause elevators to make frequent stops at common floors, resulting to system inefficiency. We want to formulate an optimization problem for the no-zoning case to compute f^{nz} in (2.10).

Recall that we assume $r_1 + \cdots + r_L = 2n$, $n \in \mathbb{Z}_+$. As in the zoning model, we assume that customers always join the shortest queue, and as a result, a single queue of nC customers is formed in front of each elevator. Elevators take C customers per round-trip, and each elevator serves n batches. The adversary can permutate customers and control the demand profiles of the n batches for each elevator. The system performance is defined as the makespan, the total time to transfer all of the customers, which can be written as

$$\sum_{i=1}^{2} \sum_{j=1}^{n} \left(aH(B_{ij}) + bS(B_{ij}) \right), \tag{2.16}$$

where B_{ij} is the jth batch for elevator i. The adversary's allocation problem can then be formulated as the following MILP:

$$
\begin{aligned}
(P) \min_{\mathbf{x},\mathbf{y},\mathbf{w},\xi,f_1,f_2} \quad & \xi \\
\text{s.t.} \quad & f_i = \sum_{j=1}^{n} \left(a w_{ij} + b \sum_{k=1}^{K} x_{ijk} \right) \ , i = 1,2 && (2.17) \\
& \xi \geq f_i \ , i = 1,2 && (2.18) \\
& x_{ijk} \leq y_{ijk} \ , i = 1,2 \ , j = 1,...,n \ , k = 1,...,K \\
& \sum_{k=1}^{K} y_{ijk} = C \ , i = 1,2 \ , j = 1,...,n \\
& \sum_{j=1}^{n} y_{ijk} = d_k \ , i = 1,2 \ , k = 1,...,K \\
& w_{ij} \leq k x_{ijk} \ , i = 1,2 \ , j = 1,...,n, k = 1,...,K \\
& x_{ijk} \in \{0,1\}, \ i = 1,2 \ , j = 1,...,n, k = 1,...,K \ , \\
& y_{ijk} \in \mathbb{Z}_+ \ , i = 1,2 \ , j = 1,...,n, k = 1,...,K \ , \\
& w_{ij} \geq 0 \ , i = 1,2 \ , j = 1,...,n.
\end{aligned}
$$

In Constraint (2.17), f_i is the time to transfer all customers at elevator i. Since P is a minimization problem, constraint (2.18) and (**??**) ensure that that the optimal objective function is equal to the makespan, $f^{nz} = \max\{f_1, f_2\}$ in (2.10) for any optimal solution.

In the following, we explain how to solve the MILP P efficiently. As in the zoning model, we first show in the following lemma that we can work on a smaller problem than P to find f^{nz} by getting rid of (y_{ijk}) in the formulation.

Lemma 2.3.7 *f^{nz} is equal to the optimal objective value of the following MILP, $\bar{P}$:*

$$
\begin{aligned}
(\bar{P}) \min_{\mathbf{x},\mathbf{w},\xi,f_1,f_2} \quad & \xi \\
s.t. \quad & f_i = \sum_{j=1}^{n} \left(a w_{ij} + b \sum_{k=1}^{K} x_{ijk} \right) \ , i = 1, 2 \\
& \xi \geq f_i \ , i = 1, 2 \\
& \sum_{k=1}^{K} x_{ijk} \leq C \ , i = 1, 2 \ , j = 1, ..., n \\
& \sum_{j=1}^{n} x_{ijk} \leq d_k \ , i = 1, 2 \ , k = 1, ..., K \\
& w_{ij} \leq k x_{ijk} \ , i = 1, 2 \ , j = 1, ..., n, k = 1, ..., K \\
& x_{ijk} \in \{0, 1\}, \ i = 1, 2 \ , j = 1, ..., n, k = 1, ..., K \ , \\
& w_{ij} \geq 0 \ , i = 1, 2 \ , j = 1, ..., n.
\end{aligned}
$$

Proof. The proof is similar to that of Lemma 2.3.2. Let $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{w}})$ and $(\mathbf{x}^*, \mathbf{w}^*)$ be optimal solutions for P and $\bar{P}$, respectively, and let $\hat{\xi}$ and ξ^* be the objective function values of the two solutions in P and $\bar{P}$, respectively. Since $(\hat{\mathbf{x}}, \hat{\mathbf{w}})$ is also feasible for P with the same objective function, we have $\hat{\xi} \geq \xi^*$. The proof of the inequality in the opposite direction, $\hat{\xi} \leq \xi^*$, follows a similar argument to that of the second part of the proof in Lemma 2.3.2, and we omit it here. ■

By Lemma 2.3.7, the performance of the no-zoning configuration can be obtained by solving $\bar{P}$. In the remainder of the section, we let $(\mathbf{x}^*, \mathbf{w}^*)$ be an optimal solution of $\bar{P}$. In the following lemma and proposition, we show that $\bar{P}$ can be solved by applying similar ideas used to solve $\bar{P}_1(z)$ in the previous section. .

Algorithm 2: Greedy assignment of the highest reversal floors in the no-zoning case

Result: w^*_{ij}, x^*_{ijk}, $i = 1$, $j = 1, ..., n$, $k = 1, ..., K$.

Set all $x^*_{ijk} = 0$. Set $\hat{d}_1 = d_1, ..., \hat{d}_K = d_K$. ;
Set $k = K$.;
for *i=1,2* **do**
  Set $j = 1$.;
  while $j \leq n$ **do**
    while $\hat{d}_k = 0$ **do**
      $k \leftarrow k - 1$;
    end
    $w^*_{ij} \leftarrow k$;
    $x^*_{ijk} \leftarrow 1$;
    $\hat{d}_k \leftarrow \hat{d}_k - 1$;
    $j \leftarrow j + 1$;
  end
end

Lemma 2.3.8 *(w^*_{ij}) in $\bar{P}$ can be obtained by Algorithm 2, and is unique up to permutation over the batches.*

Proof. Note that Algorithm 2 determines the highest reversal floors of the batches associated with one of the elevators(elevator 1) first in greedy fashion. That is, starting from type-K customer, the algorithm assigns to each batch for elevator 1 one customer who determines the highest reversal floor as high as possible, as in Algorithm 1. After finishing the allocation for elevator 1, the highest reversal floors of the batches for the other elevator are allocated similarly with the remaining customer pool.

Now, we prove that for any optimal solution $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{w}})$ of P, $\hat{\mathbf{w}}$ should be identical to $\mathbf{w}^*$ obtained by Algorithm 2 up to permutation over the batches. In order to do this, note that $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{w}})$ maximizes $\max\{f_1, f_2\}$. Assume by contradiction that $\hat{\mathbf{w}}$ is not a permutation of $\hat{\mathbf{w}}^*$. Similarly to the proof of Lemma 2.3.3, we can show that $\max\{f_1, f_2\}$ can be strictly increased by swapping customers who are going to the highest reversal floors between some batches. Note that such swapping can be done even between batches associated with different elevators, since every elevator can serve every type of customers. ■

To formulate a problem to compute $(\mathbf{x}^*, \mathbf{y}^*)$, we let A be the set of customer type-batch pairs that are not "used" by Algorithm 1 to set the highest reversal floors. That is,

$$A = \{(i, j, k) : 1 \leq i \leq 2, 1 \leq j \leq n, 1 \leq k \leq K, x^*_{ijk} = 0, \hat{d}_k > 0\}.$$

Also, we let B be the set of customer types that can be still allocated to some batch after allocating the highest reversal floors as in Algorithm 2:

$$B = \{1 \le k \le K : (i,j,k) \in A \text{ for some } 1 \le i \le 2, 1 \le j \le n\}.$$

As a result of Lemma 2.3.8, we can compute the summation of the highest reversal floors of the adversarial batches as the following:

$$\begin{aligned} H_1^{nz} &:= \sum_{j=1}^{n} w_{1j}^*, \\ H_2^{nz} &:= \sum_{j=1}^{n} w_{2j}^*, \\ H^{nz} &:= H_1^{nz} + H_2^{nz}. \end{aligned}$$

By Lemma 2.3.8, $\mathbf{x}^*$ has to satisfy that $x_{ijk}^* = 1$ for $(i,j,k) \notin A$. Then, similarly to the zoning case, we can solve P by plugging $x_{ijk}^* = 1$ for $(i,j,k) \notin A$ and (w_{ij}^*) into the problem and maximize $\sum_i \sum_j \sum_k x_{ijk}$. In Proposition 2.3.9, we further show that the latter can be actually computed by simply solving for the adversarial customer allocation for elevator 1, (x_{1jk}^*).

Proposition 2.3.9 *$\{x_{ijk}^* : (i,j,k) \in A\}$ can be obtained as an optimal solution of the following linear program. Together with the output of Algorithm 2, $\{w_{ij}^* : 1 \le i \le 2, 1 \le j \le n\}$ and $\{x_{ijk}^* : (i,j,k) \notin A\}$, $(\mathbf{x}^*, \mathbf{w}^*)$ becomes optimal for $\bar{P}$.*

$$\begin{aligned} \min_{\mathbf{x},\xi,f_1,f_2} \quad & \xi \\ s.t. \quad & f_i = aH_i^{nz} + b\left(n + \sum\sum_{(j,k)\in A} x_{ijk}\right), i = 1,2 \\ & \xi \ge f_i , i = 1,2 \\ & \sum_{k:(j,k)\in A} x_{ijk} \le C - 1 , i = 1,2 , j = 1,...,n \\ & \sum_{j:(j,k)\in A} x_{ijk} \le d_k , i = 1,2 , k \in B \\ & x_{ijk} \in \{0,1\}, \; i = 1,2 , (j,k) \in A \end{aligned}$$

Furthermore, the optimal objective value of the above problem, which is equal to f^{nz}, can be *obtained by solving the adversarial allocation problem only for elevator 1 as the following:*

$$\begin{aligned}
&\max_{\mathbf{x}_1} && aH_i^{nz} + b\left(n + \sum\sum_{(j,k)\in A} x_{1jk}\right) \\
&s.t. && \sum_{k:(j,k)\in A} x_{1jk} \le C-1 \ , j = 1, ..., n \\
& && \sum_{j:(j,k)\in A} x_{1jk} \le d_k \ , k \in B \\
& && 0 \le x_{1jk} \le 1 \ , (j,k) \in A
\end{aligned} \tag{2.19}$$

Proof. The first part of the proposition follows from Lemma 2.3.8. To prove the second part of the proposition, let

$$S_i^{nz}(\mathbf{x}_A) := \sum\sum_{(j,k)\in A} x_{ijk} \ , i = 1, 2,$$

for any feasible solution $\mathbf{x}_A := (x_{ijk}; (i,j,k) \in A)$ in the first MILP. Also, let k_0 be the lowest type of customers that are allocated to set the highest reversal floors in the algorithm. From Algorithm 2, there can be two kinds of result:

1. $k_0 = K$, or $k_0 < K$ and $\hat{d}_k = 0$: the former means there are at least $2n$ type-K customers and the highest reversal floors of all batches are K. The latter means that Algorithm 2 terminates exactly after allocating all of type-K,...,type-k_0 customers. In both of the cases, we have

$$A = \{(i,j,k) : 1 \le i \le 2, 1 \le k < k_0\}.$$

   Then, the sets of customer type-batch pairs, (j,k), that have the associated binary variable x_{ijk} are the same for $i = 1, 2$ in the first MILP in the proposition. This implies that the maximum value achievable for $S_1^{nz}(\mathbf{x}_A)$ is the same as that of $S_2^{nz}(\mathbf{x}_A)$.

2. $k_0 < K$ and $\hat{d}_k > 0$: in this case, Algorithm 2 terminates after allocating type-k_0 customers to some batches for elevator 2. Note that for all $k < k_0$, $(i,j,k) \in A$ for all $i = 1, 2$ and $j = 1, ..., n$. For all $k > k_0$, $(i,j,k) \notin A$ for all i, j. For $k = k_0$, note that we should have

$$|\{(1,j,k_0) \in A : 1 \le j \le n\}| > |\{(2,j,k_0) \in A : 1 \le j \le n\}| .$$

   This implies that the maximum value achievable for $S_1^{nz}(\mathbf{x}_A)$ is greater than or equal to that of $S_2^{nz}(\mathbf{x}_A)$ in the first MILP.

By the above observations and the fact that $H_1^{nz} \geq H_2^{nz}$, we know that the objective value, ξ, for any optimal solution $\tilde{\mathbf{x}}_A$ of the first MILP must be equal to f_1. Now, let $(x^*_{1jk}; (j,k) \in A)$ be an optimal solution of the second linear program in the proposition. Note that the constraint matrix must be totally unimodular, and thus $(x^*_{1jk}; (j,k) \in A)$ is integral, i.e., $x^*_{ijk} \in \{0,1\}$. It is easy to see that we can find $(x_{2jk}; (j,k) \in A)$ such that $\mathbf{x}^*_A := (x^*_{1jk}, x_{2jk}; (j,k) \in A)$ becomes feasible for the first MILP in the proposition. Moreover, it is obvious that

$$S_1^{nz}(\mathbf{x}^*_A) \geq S_1^{nz}(\mathbf{x}_A),$$

for any feasible solution $\mathbf{x}_A$ for the MILP, and this concludes the proof . ■

2.4 Numerical study

In this section, we introduce two examples to illustrate our model and the solution approach proposed in the previous section. In both of the examples, we consider a five-story building that has two elevators. Each elevator has capacity of five, and we use the parameters $a = 1, b = 2$ for the cost function. In the first example, we analyze a case of uniform demand, i.e., $d_1 = d_2 = \cdots = d_K$. In the second example, we cover a case of imbalanced demand, where there is one floor whose demand is significantly larger than those of other floors. These toy examples allow us to obtain closed-form solutions and to clearly show our solution approach. We hope to gain insights from these examples that can be used for further analyzing more general cases.

2.4.1 Uniform demand

As the first illustrative example, we study a case of uniform demands, i.e., $d_1 = d_2 = \cdots = d_K = d$. To better illustrate our approach, we introduce a small example with $K = 5$, $C = 5$, and $d = rC$ with $r = 2$. For $z = 1, 2, 3, 4$, we have $n_1(z) = rz = 2z$ and $n_2(z) = r(K - z) = 2(5 - z)$. It is easy to check that $w^*_{1j} = z$ for all $j = 1, ..., 2z$ and $w^*_{2j} = K = 5$ for all $j = 1, ..., 2(5 - z)$, $z = 1, 2, 3, 4$, and thus

$$H_1(z) = \sum_{j=1}^{2z} w^*_{1j} = 2z^2, \ H_2(z) = \sum_{j=1}^{2(5-z)} w^*_{2j} = 10(5 - z).$$

We let $S_i(z) := \sum_{(j,k) \in A_i(z)} x^*_{ijk}(z)$, where $(x^*_{ijk}(z)), i = 1, 2$ is an optimal solution of the inner maximization problem in Proposition 2.3.5 given $z \in \{1, 2, ..., K - 1\}$. To compute $S_1(z)$, note that the optimization problem in Proposition 2.3.4 becomes

$$\begin{aligned}
\max_{\mathbf{x}_1} \quad & \sum_{j=1}^{2z}\sum_{k=1}^{z-1} x_{1jk} \\
\text{s.t.} \quad & \sum_{j=1}^{2z} x_{1jk} \le 2C \ , k = 1, ..., z-1 \\
& \sum_{k=1}^{z-1} x_{1jk} \le C - 1 \ , j = 1, ..., 2z \\
& 0 \le x_{1jk} \le 1 \ , j = 1, ..., 2z \ , k = 1, ..., z-1
\end{aligned}$$

Note that this linear program is well defined only when $z > 1$, in which case we can verify that $x^*_{1jk} = 1$ for all $j = 1, ..., 2z, k = 1, ..., z-1$ is the unique optimal solution. Similarly, the LP in Proposotion 2.3.4 to compute $S_2(z)$ is

$$\begin{aligned}
\max_{\mathbf{x}_2} \quad & \sum_{j=1}^{2(5-z)}\sum_{k=z+1}^{4} x_{2jk} \\
\text{s.t.} \quad & \sum_{j=1}^{2(5-z)} x_{2jk} \le 2C \ , k = z+1, ..., 4 \\
& \sum_{k=z+1}^{5} x_{2jk} \le C - 1 \ , j = 1, ..., 2(5-z) \\
& 0 \le x_{2jk} \le 1 \ , j = 1, ..., 2(5-z) \ , k = z+1, ..., 4
\end{aligned}$$

, which is well defined only if $z \le 3$ and has the unique optimal solution $x^*_{2jk} = 1$ for all $j = 1, ..., 2(5-z), k = z+1, ..., 4$. For $z = 1, 2, 3, 4$, we let

$$S_1(z) = \sum_{j=1}^{2z}\sum_{k=1}^{z-1} x^*_{1jk} + 2z = 2z^2, \ S_2(z) = \sum_{j=1}^{2(5-z)}\sum_{k=z+1}^{4} x^*_{2jk} + 2(5-z) = 2(5-z)^2.$$

The summation on the right-hand side in the first equation is replaced by zero when $z = 1$, and $S_1(1) = 0 + 2\cdot = 2$. Note that this value is equal to $2z^2$ evaluated at $z = 1$, and we define $S_1(z)$ as above for all $z = 1, 2, 3, 4$ for convenience. Similarly, we define $S_2(z)$ as above for all $z = 1, 2, 3, 4$.

where the summation term for computing $S_2(z)$ is defined to be zero for $z = 4$. The second terms are added to account for the total time spent for stopping at the highest reversal floors, z and 5, respectively. Then, for $z = 1, 2, 3, 4$, the system performance of the zoning is

$$\begin{aligned} f(z) &= \max\{aH_1(z) + bS_1(z), aH_2(z) + bS_2(z)\} \\ &= \max\left\{2z^2 + 4z^2, 10(5-z) + 4(5-z)^2\right\} \\ &= \begin{cases} 2(5-z)(15-2z) & , z = 1, 2 \\ 6z^2 & , z = 3, 4. \end{cases} \end{aligned}$$

We can check that $z^* = 3$ achieve the minimum system cost among the zoning(partitioning) configuration with

$$f(z^*) = \min_{z=1,2,3,4} \max\{aH_1(z) + bS_1(z), aH_2(z) + bS_2(z)\} = 54.$$

Now, in order to compute the system performance of the no-zoning decision, note that $r_1 + \cdots + r_5 = 10$, we have $n = 5$ and each elevator serves five batches. By Lemma 2.3.8, it is easy to see that the highest reversal floors obtained from Algorithm 2 are $w^*_{1j} = w^*_{2j} = 5,\ j = 1, ..., 5$, and thus

$$H_1^{nz} = H_2^{nz} = 5n = 25.$$

It is also easy to check that

$$\begin{aligned} &A = \{(i, j, k) : i = 1, 2,\ j = 1, ..., 5, k = 1, ..., 4\}, \\ &B = \{1, 2, 3, 4\}. \end{aligned}$$

Using the result from Proposition 2.3.9, we can compute the system performance in the no-zoning configuration by solving

$$\begin{aligned} \max_{\mathbf{x}_1} \quad & \sum_{j=1}^{5}\sum_{k=1}^{4} x_{1jk} \\ \text{s.t.} \quad & \sum_{k=1}^{4} x_{1jk} \le C - 1\ , j = 1, ..., 5 \\ & \sum_{j=1}^{5} x_{1jk} \le 2C\ , k = 1, ..., 4 \\ & 0 \le x_{1jk} \le 1\ , j = 1, ..., 5\ , k = 1, ..., 4 \end{aligned}$$

We can check that $x^*_{1jk} = 1, \forall j = 1, ..., 5, k = 1, ..., 4$ is optimal, and

$$f^{nz} = H_1^{nz} + 2\left(5 + \sum_{j=1}^{5}\sum_{k=1}^{4} x^*_{1jk}\right) = 75.$$

Since $f^{nz} > f(z^*)$, the planner's optimal decision is assigning floor 1 through 3 to elevator 1, and floor 4 and 5 to elevator 2. The optimal system performance, which is total time for elevators to move all customers to their destinations and return to the lobby, is 54. The computation result is shown in Figure 2.1. The red and blue lines are the graphs of $f_1(z)$ and $f_2(z)$, respectively, i.e., the time for elevator 1 and 2 to drop off all the customers and return to the lobby given a zoning z. The black line shows f^{nz}, and the circles denote $f(z) = \max(f_1(z), f_2(z))$. The circle corresponding to z^* is filled in. The graphs in Figure 2.1 verify the unimodularity result of $f(z)$ in Theorem 2.3.6.

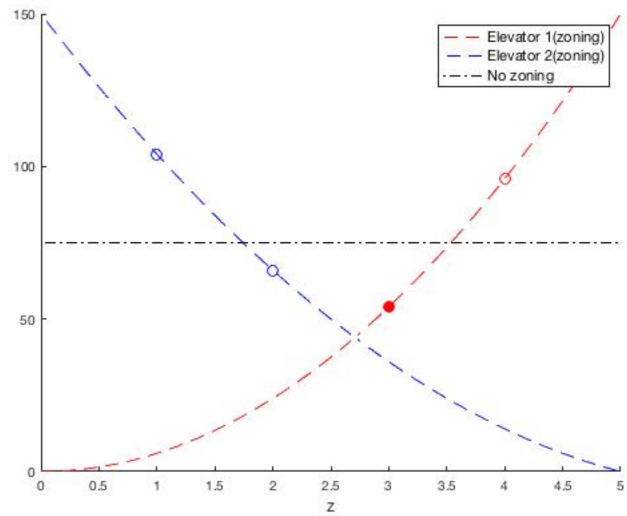


Figure 2.1 Performance of different zoning configurations: uniform demand

2.4.2 Imbalanced demand

In this section, we consider an example of highly imbalanced demands: we suppose $d_1 = \cdots = d_4 = C$ and $d_5 = 6C$. The total demand, $d_1 + \cdots + d_5 = 10C$, is the same as in the previous section. Note that the demand is highly imbalanced. In the zoning case, for $z = 1, 2, 3, 4$, elevator 1 and 2 need to serve $n_1(z) = z$ and $n_2(z) = 10 - z$ batches, respectively. It is easy to check that $w^*_{1j} = z$ for all $j = 1, ..., z$ and $w^*_{2j} = 5$ for all $j = 1, ..., 10 - z$. It follows that

$$H_1(z) = \sum_{j=1}^{z} w^*_{1j} = z^2, \; H_2(z) = \sum_{j=1}^{10-z)} w^*_{2j} = 5(10 - z).$$

$S_1(z)$ can be computed by solving the following linear program:

$$\begin{aligned}
\max_{\mathbf{x}_1} \quad & \sum_{j=1}^{z}\sum_{k=1}^{z-1} x_{1jk} \\
\text{s.t.} \quad & \sum_{j=1}^{z} x_{1jk} \leq C \ , k = 1, ..., z-1 \\
& \sum_{k=1}^{z-1} x_{1jk} \leq C - 1 \ , j = 1, ..., z \\
& 0 \leq x_{1jk} \leq 1 \ , j = 1, ..., z \ , k = 1, ..., z-1
\end{aligned}$$

Again, the above linear program is well defined only if $z > 1$, in which case $x^*_{1jk} = 1$ for all $j = 1, ..., z$, $k = 1, ..., z-1$ is the unique optimal solution, and we can define

$$S_1(z) = z(z-1) + z = z^2, \ z = 1, 2, 3, 4.$$

Similarly, for $z \leq 3$, $S_2(z)$ can be computed by solving

$$\begin{aligned}
\max_{\mathbf{x}_2} \quad & \sum_{j=1}^{10-z}\sum_{k=z+1}^{4} x_{2jk} \\
\text{s.t.} \quad & \sum_{j=1}^{10-z} x_{2jk} \leq C \ , k = z+1, ..., 4 \\
& \sum_{k=z+1}^{4} x_{2jk} \leq C - 1 \ , j = 1, ..., 10-z \\
& 0 \leq x_{1jk} \leq 1 \ , j = 1, ..., z \ , k = z+1, ..., 4.
\end{aligned}$$

Since $10 - z > C$ for $z = 1, 2, 3$, unlike the case of elevator 1, we cannot let x_{2jk} be one for all pairs of j, k. However, because $4 - (z+1) + 1 = 4 - z < C - 1$ for $z = 1, 2, 3$, the second constraint cannot be binding, and thus the optimal solution can be found by making the first constraint binding for every k, i.e., take a feasible solution (x^*_{2jk}) satisfying $\sum_{j=1}^{10-z} x^*_{2jk} = C$ for all $k = z+1, ..., 4$. Then,

$$S_2(z) = \sum_{k=z+1}^{4}\sum_{j=1}^{10-z} x^*_{2jk} + (10-z) = 5(4-z) + (10-z) = 30 - 6z,$$

for $z = 1, 2, 3, 4$, and the performance of zoning decision $z = 1, 2, 3, 4$ is

$$\begin{aligned}
f(z) &= \max\{aH_1(z) + bS_1(z), aH_2(z) + bS_2(z)\} \\
&= \max\left\{z^2 + 2z^2, 5(10-z) + 2(30-6z)\right\} \\
&= \begin{cases} 110 - 17z, z = 1, 2, 3 \\ 3z^2, z = 4. \end{cases}
\end{aligned}$$

It follows that $f(z)$ is minimized at $z^* = 4$ with $f(z^*) = 48$. In the no-zoning case, note that $r_1 + \cdots + r_5 = 10$, and $n = 5$ as in the previous section. We can check that the highest reversal floors of the batches obtained from Algorithm 2 are $w^*_{1j} = w^*_{2j} = 5,\ j = 1, ..., 5,$, and $H_1^{nz} = H_2^{nz} = 25$ as in the previous example. Again, we have

$$A = \{(i, j, k) : i = 1, 2,\ j = 1, ..., 5, k = 1, ..., 4\},$$
$$B = \{1, 2, 3, 4\},$$

and the linear program to compute the performance of the no-zoning configuration as in Proposition 2.3.9 is

$$\begin{aligned}
\max_{\mathbf{x}_1} \quad & \sum_{j=1}^{5}\sum_{k=1}^{4} x_{1jk} \\
\text{s.t.} \quad & \sum_{k=1}^{4} x_{1jk} \le C - 1\ , j = 1, ..., 5 \\
& \sum_{j=1}^{5} x_{1jk} \le C\ , k = 1, ..., 4 \\
& 0 \le x_{1jk} \le 1\ , j = 1, ..., 5\ , k = 1, ..., 4
\end{aligned}$$

Tbe optimal solution is $x^*_{1jk} = 1, \forall j = 1, ..., 5, k = 1, ..., 4$, the same as in the previous example, and thus $f^{nz} = 75 > f(z^*)$. Thus, the planner's optimal zoning decision is floor 1 through 4 to elevator 1, and floor 5 to elevator 2, and the total time for elevators to drop off all customers and return to the lobby is 48. The computation result is shown in Figure 2.2, which also confirms the uniformularity result of $f(z)$ as in Theorem 2.3.6.

2.5 Conclusion and future directions of research

In this research, we developed an adversarial framework to find the optimal zoning configuration of elevator systems in the perspective of capacity planning. We consider an uppeak traffic situation, which is the heaviest traffic that the building can experience during the day, and the performance is defined to be the total time to transfer all of customers to their destinations. The optimal zoning decision obtained from the model thus provides a conservative performance guarantee for the system to deal with congested traffic.

One of the challenges of the problem is to incorporate the effect of customer batching on round-trip times of the elevators. Round-trip time of an elevator depend on the demand profile of the customer batch through two components: (i) the number of stops that it needs to make to drop off customers, i.e., the number of distinct destinations in the batch, (ii) the highest destination that it needs to reach, i.e., the highest reversal floor of the batch. We take into account such a unique dependence of round-trip times on the demand profiles in

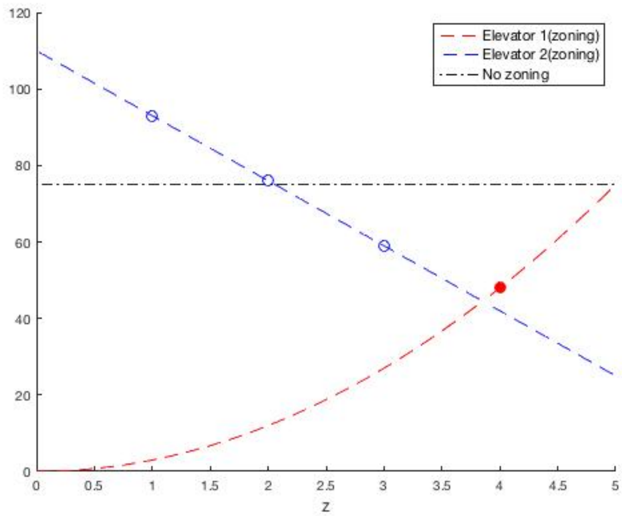


Figure 2.2 Performance of different zoning configurations: imbalanced demand

analyzing the system performance and finding the optimal zoning configuration. In order to do so, we introduce an adversary who can permutate customers in queues in front of the elevators, given the planner's zoning decision, and determine the demand profiles of the batches. Using the adversarial framework is also consistent with our goal of providing a capacity planning model for elevator systems. We formulated mixed-integer linear programs for computing performance different zoning decisions, and showed that they can be solved by simple greedy algorithms and reduction to smaller linear programs. We also proved the unimodularity of the performance function of the zoning case.

One of immediate extension of our model will be developing a similar model for a general setting with more than two elevators. Using the insight obtained from the model with two elevators, we may extend the mixed-integer linear program(MILP) approach to the general case, and show that the performance of each zoning decision can be computed by reducing the MILP to a greedy algorithm and a linear program. Also, we hope to exploit the unimodularity result in Theorem 2.3.6, and develop a bisection method to find the best zoning configuration. It will be an interesting direction to investigate the performance of the bisection method analytically and empirically.

In this research, we only considered zoning decisions without overlapping floors, that is, elevators are assigned set of floors with no intersection. On one hand, when different elevators serve common floors, the traffic of customers who want to move to those floors will be divided over the elevators, which may improve the system efficiency. On the other hand, overlapped zones may result in different elevators stopping at the common floors, which may increase round-trip times of the elevators and harm the system performance. It would be an interesting direction of future research to develop a model that allows overlapping between zones and find the optimal zoning decision.

We used a stylizing assumption that demand to each floor is an integer multiple of the elevator capacity. The assumption lets us analyze the customer batching effect using intuitive greedy algorithms and linear programs, and allows us to obtain analytic results. We may extend our model to a case of uncertain demands. One way to do so may be using robust optimization approach. For example, we may assume that the demand to each floor is a multiple of the elevator capacity with a random coefficient, which may still allow the robust counterpart of the formulation to be tractable using insights from the original model in this dissertation.

Bibliography

Abdulkadiroğlu, A., T. Sonmez. 2013. Matching markets: theory and practice. *Advances in Economics and Econometrics* .

Afèche, P., R. Caldentey, V. Gupta. 2019. On the optimal design of a bipartite matching queueing system.

Akbarpour, M., S. Li, S.O. Gharan. 2019. Thickness and information in dynamic matching markets. *forthcoming in Journal of Political Economy* .

Ashlagi, I., M. Burq, P. Jaillet, V. Manshadi. 2019. On matching and thickness in heterogeneous dynamic markets. *Operations Research* .

Baccara, M., S. Lee, Y. Yariv. 2018. Optimal dynamic matching.

Barney, G. 2015. *Elevator Traffic Handbook: Theory and Practice*. Routledge.

Beardwood, J., J. H. Halton, J. M. Hammersley. 1959. The shortest path through many points. *Mathematical Proceedings of the Cambridge Philosophical Society* .

Benjaafar, S., M. Hu. 2019. Operations management in the age of the sharing economy: What is old and what is new? *Manufacturing & Service Operations Management* .

Bertsimas, D. J., G. V. Ryzin. 1993a. Stochastic and dynamic vehicle routing in the euclidean plane with multiple capacitated vehicles. *Operations Research* .

Bertsimas, D. J., G. V. Ryzin. 1993b. Stochastic and dynamic vehicle routing with general demand and interarrival time distributions. *Advances in Applied Probability* .

Bolat, B., P. Cortés, E. Yalcin, M. Alisverisci. 2010. Optimal car dispatching for elevator groups using genetic algorithms. *Intelligent Automation and Soft Computing* .

Carlsson, J. G., M. Behroozi, K. Mihic. 2018. Wasserstein distance and the distributionally robust tsp. *Operations Research* .

Carlsson, J. G., E. Delage. 2013. Robust partioning for stochastic multivehicle routing. *Operations Research* .

Chen, T., Y. Hsu, Y. Huang. 2012. Optimizing the intelligent elevator group control system by using genetic algorithm. *Advanced Science Letters* .

Cortés, P., J. Larrañeta, L. Onieva. 2004. Genetic algorithm for controllers in elevator groups: Analysis and simulation during lunchpeak traffic. *Applied Soft Computing* .

Crites, R. H., A. G. Barto. 1998. Elevator group control using multiple reinforcement learning agents. *Machine Learning* .

Doval, L. 2019. Dynamically stable matching. *Working paper* .

Fujino, A., T. Tobita, K. Segawa, K. Yoneda, A. Togawa. 1997. An elevator group control system with floor-attribute control method and system optimization using genetic algorithms. *IEEE Transactions on Industrial Electronics* .

Gurvich, I., A. Ward. 2014. On the dynamic control of matching queues. *Stochastic Systems* .

Hu, B., S. Benjaafar. 2009. Partitioning of servers in queueing systems during rush hour. *Manufacturing and Service Operations Management* .

Hu, M. 2018. *Sharing Economy: Making Supply Meet Demand*. Springer.

Hu, M. 2019. From the classics to new tunes: A neoclassical view on sharing economy and innovative marketplaces.

Hu, M., Y. Zhou. 2019. Dynamic type matching.

Imasaki, N., S. Kubo, S. Nakai, T. Yoshitsugu, K. Jun-Ichi, T. Endo. 1995. Elevator group control system tuned by a fuzzy neural network applied method .

Ishihara, H., S. Kato. 2013. Multicar elevator control using dynamic zoning .

Ishikawa, T., A. Miyauchi, M. Kaneko. 2000. Supervisory control for elevator group by using fuzzy expert system which also addresses traveling time .

Johari, R., V. Kamble, Y. Kanoria. 2019. Matching while learning.

Kanoria, Y., P. Qian. 2019. Near optimal control of a ride-hailing platform via mirror backpressure.

Kenji, S., M. Sandor, N. Masami. 1996. Elevator group supervisory control system using neural network. *Elevator World* .

Kim, C., K. Seong, H. Lee-kwang. 1998. Design and implementation of a fuzzy elevator group control system. *IEEE Transactions on Systems, Man, and Cybernetics* .

Koh, E. L., J. O. Kim, P. H. Hahn. 1999. Group management control meth- od for elevator. *U.S. Patent 6 000 504* .

Liu, C. 2019. Stability in repeated matching markets. *Working paper* .

Liu, J., Y. Liu. 2007. Ant colony algorithm and fuzzy neural network-based intelligent dispatching algorithm of an elevator group control system. *IEEE International Conference on Control and Automation* .

Liu, Y., Z. Hu, Q. SU, J. Huo. 2010. Energy saving of elevator group control based on optimal zoning strategy with interfloor traffic .

Marbach, P., J. N. Tsitsiklis. 2001. Simulation-based optimization of markov reward processes. *IEEE Transactions on Automatic Control* .

Mehta, A., A. Saberi, U. Vazirani, V. Vazirani. 2013. Matching markets: theory and practice. *Advances in Economics and Econometrics* .

Nazari, M., A. L. Stolyar. 2019. Reward maximization in general dynamic matching systems. *Queueing Systems* .

Oh, S., D. Park. 1995. Self-tuning fuzzy controller with variable universe of discourse .

Özkan, E., A. Ward. 2019. Dynamic matching for real-time ridesharing. *forthcoming in Stochastic Systems* .

Parlar, M., M. Sharafali, J. Ou. 2006. Optimal parking of idle elevators under myopic and state-dependent policies. *European Journal of Operational Research* .

Pepyne, D. L., C. G. Cassandras. 1997. Optimal dispatching control for elevator systems during uppeak traffic. *IEEE Transactions on Control Systems Technology* .

Pepyne, D. L., C. G. Cassandras. 1998. Design and implementation of an adaptive dispatching controller for elevator systems during uppeak traffic. *IEEE Transactions on Control Systems Technology* .

Puterman, M. L. 2005. *Markov Decision Processes*. John Wiley & Sons, Inc.

Roth, A. E., T. Sönmez, M. U. Ünver. 2004. Kidney exchange. *Quarterly Journal of Economics* .

Roth, A. E., T. Sönmez, M. U. Ünver. 2007. Efficient kidney exchange: Coincident of wants in market with compatibility-based preferences. *American Economic Review* .

Roth, A. E., M. Sotomayor. 1990. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Cambridge University Press.

Rottweil, Tytyri. 2017. New lift technology is reshaping cities. *The Economist.* `https://www.economist.com/christmas-specials/2017/12/19/new-lift-technology-is-reshaping-cities` (visited: May 15, 2019).

Sasaki, K., K. Yokota, H. Hattori, N. Sata. 1991. Method and apparatus for controlling a group of elevators using fuzzy rules. *U.S. Patent 5 022 498* .

Stanley, J. A., H. Honma, D. S. Williams, T. Mori, P. Simcik. 2007. Elevator car dispatching including passenger destination information and a fuzzy logic algorithm. *U.S. Patent 20 070 045 052 (A1)* .

Tanaka, S., Y. Uraguchi, M. Araki. 2005a. Dynamic optimization of the operation of single-car elevator systems with destination hall call registration: Part i. formulation and simulations. *European Journal of Operational Research* .

Tanaka, S., Y. Uraguchi, M. Araki. 2005b. Dynamic optimization of the operation of single-car elevator systems with destination hall call registration: Part ii. the solution algorithm. *European Journal of Operational Research* .

Touretzky, D. S., M. C. Mozer, M. E. Hasselmo, eds. 1996. *Improving Elevator Performance Using Reinforcement Learning*.

Tyni, T., J. Ylinen. 2006. Evolutionary bi-objective optimisation in the elevator car routing problem. *European Journal of Operational Research* .

Ünver, M. U. 2010. Dynamic kidney exchange. *Review of Economic Studies* .

Yang, S., J. Tai, C. Shao. 2009. Dynamic partition of elevator group control system with destination floor guidance in uppeak traffic. *Journal of Computers* .