MECHANISM DESIGN FOR COMPLEX SYSTEMS: BIPARTITE MATCHING OF DESIGNERS AND MANUFACTURERS, AND EVOLUTION OF AIR TRANSPORTATION NETWORKS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Joseph D Thekinen

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2018

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Jitesh H. Panchal, Chair

School of Mechanical Engineering

Dr. Daniel DeLaurentis

School of Aeronautics and Astronautics

Dr. Ilias Bilionis

School of Mechanical Engineering

Dr. William A. Crossley

School of Aeronautics and Astronautics

Approved by:

Dr. Jay P. Gore

Head of the School Graduate Program

Dedicated to Mom and Dad.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my parents and family for all the motivation and support they have provided me from childhood. I remember right from childhood they have been stressing on the importance and power of education, they remunerated me with rewards ranging from chocolates to small deposits for school performance, they always invested in developing my soft skills, they helped me identify my niches, and they had given me infinite freedom to pursue knowledge and study as long as I want. I am thankful for the emotional support they have given me during tough times. It is impossible to express my gratefulness for the role they have played in a few words and I owe them the most in this world.

I would like to thank Dr. Jitesh Panchal for being the best advisor one could wish to do a PhD with. Beyond doubt, he has single-handedly played the most pivotal role during my PhD right from the first day until the finish line both technically and emotionally. Thank you for introducing me to this topic, guiding me in the best direction identifying my strengths and weaknesses, motivating me throughout the journey, and enlightening me with alternate directions during tough times. One aspect that particularly stands out is that he consistently stressed on imparting the skills of an independent researcher to me be it research thinking or be it technical writing; he never edits my writing himself but rather points out the mistakes so that I could learn as I correct. He stresses on the growth of a PhD student as an independent researcher as much as the quality of the research output. And there are many more reasons for me to feel grateful for which the space is here is too short to express. Thank you for everything!

Next, I would like to acknowledge my gratitude towards the professors: Dr. DeLaurentis, Dr. Bilionis, and Dr. Crossley for their continued guidance and valuable advice. They added a lot of my value in my research my expanding my horizons of thinking and providing me with constructive criticism for my work during meetings and presentations. I am also thankful to all my teachers at Purdue who taught me various courses which improved my knowledge and helped me accelerate my understanding of the subject. At this point, I also remember my undergraduate teachers back at IIT, Kharagpur, all of them, with immense gratitude; particularly my B.Tech and M.Tech project advisor Dr. Datta.

I would like to thank my lab mates: Adam Dachowitz, Ashish Chaudhari, Han Yupeng, Murtuza Shergadwala, Naman Mandhan, Piyush Pandita, Sharmila Karumuri, Siva Chaduvula, and Vikrant Reddy who were like a family away from home giving the perfect atmosphere for fun and work. I will cherish my friendship with them forever. I would also like to thank two of my ex-labmates who had a mentor figure during my PhD studies: Zhenghui Sha and Ahmad Taha. They had given me a lot of support during the initial phases. I am also thankful to my collaborators Kushal Moolchandani, Navin Davendralingam, Apoorv Maheswari from the aerospace engineering department for providing a fruitful collaborative experience. I enjoyed working together with you. I would also like to thank all my friends here at Purdue, my roommates, an organization named Bridges International, my friends at the church St. Tom's Aquinas for providing me a wonderful social life. You never made it hard for me being far away from family. In addition, I thank from friends back in India some of whom I am still in touch.

Finally, I would like to acknowledge NSF Grant1360631, Purdue ME B.F.S. Schaefer Grant for providing the financial assistantship. My acknowledgement list is incomplete without mentioning my gratitude towards Purdue University particularly School of Mechanical Engineering for providing all the facilities and environment for my graduate studies. I am thankful to the graduate staff.

Once again thank you everyone for helping and being alongside me during this long journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

ABBREVIATIONS

ATS	Air Transportation System
ATN	Air Transportation Network
CBDM	Cloud-based Design and Manufacturing
CES	Complex Engineered Systems
CNS	Complex Networked System
FAA	Federal Aviation Administration
MCMC	Markov chain Monte Carlo
MH	Metropolis-Hastings
AP	Airport Presence
NE	Nash equilibria
HR	High service rate
LR	Low service rate
UA	United Airlines
DL	Delta airlines

NOMENCLATURE

$\lvert S \rvert$	Number of service seekers
S	Set of service seekers
P	Set of service providers
$\lvert P \rvert$	Number of service providers
q_{p_i}	Vacancies offered by service provider p_j
s_{-i}	Set of all service seekers excluding s_i
p_{-i}	Set of all service providers excluding p_i
X	Set of all attributes
u_{ij}	Utility of agent i being matched to alternative j
w_{ki}	Weight for attribute X_k for agent i
p	Probability distribution of the attribute values
$f(X)$	Single attribute utility
$j \succeq_i k$	i prefers j over k
$j \succ_i k$	i strictly prefers j over k
$M(s_i)$	Service provider to which service seeker s_i is assigned through matching mechanism M
t	Time period of each matching cycle (in days)
t_k	time of implementation of k^{th} matching cycle
$M_{s_i}(t_k)$	Service provider to which service seeker s_i was matched in k^th matching cycle
$M_{p_j}^{-1}(t_k)$	Service seekers being served by service provider p_j in k^th matching cycle
T	Total time duration during which matching mechanisms will be implemented (in days)

τ_{ij}	Time to process service request of service seeker s_i by service provider p_j
$\overline{\tau}$	Average service processing time
λ	Mean arrival rate of service seekers
μ_j	Mean service processing time of service provider p_j
h_j	Working hours of service seeker p_j per day
$\|M\|$	Total number of stable matches possible under mechanism M

ABSTRACT

Thekinen, Joseph D Ph.D., Purdue University, December 2018. Mechanism design for complex systems: bipartite matching of designers and manufacturers, and evolution of air transportation networks. Major Professor: Jitesh H. Panchal Professor, School of Mechanical Engineering.

A central issue in systems engineering is to design systems where the stakeholders do not behave as expected by the systems designer. Usually, these stakeholders have different and often conflicting objectives. The stakeholders try to maximize their individual objective and the overall system do not function as expected by the systems designers.

We specifically study two such systems- a) cloud-based design and manufacturing system (CBDM) and b) Air Transportation System (ATS). In CBDM, two stakeholders with conflicting objectives are designers trying to get their parts printed at the lowest possible price and manufacturers trying to sell their excess resource capacity at maximum profits. In ATS, on one hand, airlines make route selection decision with the goal of maximizing their market share and profits and on the other hand regulatory bodies such as Federal Aviation Administration tries to form policies that increase overall welfare of the people.

The objective in this dissertation is to establish a mechanism design based framework: a) for resource allocation in CBDM, and b) to guide the policymakers in channeling the evolution of network topology of ATS.

This is the first attempt in literature to formulate the resource allocation in CBDM as a bipartite matching problem with designers and manufacturers forming two distinct set of agents. We recommend best mechanisms in different CBDM scenarios like totally decentralized scenario, organizational scenario etc. based on how well the properties of the mechanism meet the requirements of that scenario. In addition to

analyzing existing mechanisms, CBDM offers challenges that are not addressed in the literature. One such challenge is how often should the matching mechanism be implemented when agents interact over a long period of time. We answer this question through theoretical propositions backed up by simulation studies. We conclude that a matching period equal to the ratio of the number of service providers to the arrival rate of designers is optimal when service rate is high and a matching period equal to the ratio of mean printing time to mean service rate is optimal when service rate is low.

In ATS, we model the evolution of the network topology as the result of route selection decisions made by airlines under competition. Using data from historic decisions we use discrete games to model the preference parameters of airlines towards explanatory variables such as market demand and operating cost. Different from the existing literature, we use an airport presence based technique to estimate these parameters. This reduces the risk of over-fitting and improves prediction accuracy. We conduct a forward simulation to study the effect of altering the explanatory variables on the Nash equilibrium strategies. Regulatory bodies could use these insights while forming policies.

The overall contribution in this research is a mechanism design framework to design complex engineered systems such as CBDM and ATS. Specifically, in CBDM a matching mechanism based resource allocation framework is established and matching mechanisms are recommended for various CBDM scenarios. Through theoretical and simulation studies we propose the frequency at which matching mechanisms should be implemented in CBDM. Though these results are established for CBDM, these are general enough to be applied anywhere matching mechanisms are implemented multiple times. In ATS, we propose an airport presence based approach to estimate the parameters that quantify the preference of airlines towards explanatory variables.

1. INTRODUCTION

The primary objective in this dissertation is to establish mechanism design based approaches in different types of complex engineered systems (CES) where multiple stakeholders, often with conflicting objectives, interact with one another. CES consists of multiple stakeholders, each with individual and often conflicting objectives, interacting with one another. These systems differ from traditional engineered systems in that they are not designed by a central designer or team of designers and display the fundamental characteristics of self-organization [1]. Some of the examples of CES are transportation networks such as Air Transportation System, communication networks, matching markets such as kidney exchange program where kidney donors are matched to patients [2], national residency matching program where medical residents are matched to hospitals [3]. In my research, I study the behavior of interacting entities in CES, evaluate the performance of existing mechanisms in CES applied in domains other than CES, identify the challenges and research gaps in applying these mechanisms into CES, and provide mechanism design recommendations to address these challenges.

1.1 Research Overview

Complex Engineered Systems (CES) consist of multiple stakeholders with multiple objectives interacting with one another. In CBDM for example, the stakeholders are service seekers and service providers; service seekers are interested in desired part quality at a minimum price, while service providers are generally interested in maximizing revenue from their available capacity. On the other hand in ATS the stakeholders are policymakers such as the Federal Aviation Administration (FAA), service providers such as airlines and service users such as air passengers. Policymakers aim to maximize the social welfare while airlines try to maximize their profit. Therefore, there are as many different objectives as there are participants in these systems.

In systems engineering literature, designing such systems is based on the assumption that the stakeholders behave in a certain manner as laid out by the systems designer. The drawback of using this assumption is that it does not consider the objectives of the independent interacting stakeholders and ignores that the interacting entities are self-interested utility maximizers.

Therefore, an alternate approach based on mechanism design is used. Mechanism design is the science of rule making [4]. A *mechanism* is an institution, procedure or game for determining outcomes [5]. The agent who gets to choose the mechanism is called the mechanism designer. The mechanism designer is usually part of the setting; for example, in the case of auctioning of public goods, the government who provides the goods is also choosing the mechanism through which the goods are allotted [5]. The larger goal of a mechanism is to design mechanisms by which selfish behavior of the agents leads to socially optimal outcome [6].

The stakeholders operate within the rules laid out by the mechanism designer. Overall the system behavior is driven by the actions of the stakeholders which, in turn, depends on the rules laid. The objectives of all interacting entities are considered and the central system is driven by a mechanism such that the overall system is driven towards better performance and efficiency given the self-interested entities behave to maximize their individual utilities.

In this dissertation, I focus on two such systems: Cloud-Based Design and Manufacturing (CBDM) and Air Transportation System (ATS). Although the evolution of ATS and resource allocation in CBDM sounds as two different design problems with unique characteristics, the general approach under the mechanism design framework shares commonalities as shown in Table 1.1. In a mechanism design framework we identify the stakeholders in the system, assess the objectives of each stakeholder which are usually different and conflicting with one another, infer the behavior of the

Table 1.1.
Common framework of CBDM and ATS from mechanism design standpoint.

	CBDM	**ATS**
Stakeholders	Service seekers, service providers, and matching platform	Airlines, FAA, passengers, and airports
Goal of mechanism	Optimal matching considering true preferences of all agents	Direct the network towards better performance
Mechanism Designer	Matching platform	Regulatory body
Inferring behavior	Theoretical analysis; simulation studies	Historical data on rout decisions
Research question	How can service seekers be optimally matched to service providers?	How do airlines make routing decisions under competition?
Research challenges	There are no studies on the frequency at which the matching mechanism needs to be implemented to optimize the matched outcomes	Most of the decision information are proprietary and private to the airlines

stakeholders, and design mechanism based on the information. For example, in ATS two such stakeholders are airlines such as United Airlines, Delta, and policymakers such as FAA. While the goal of airlines is primarily to maximize their profits, the primary objective of FAA is to provide quality and affordable service to the passengers with fewer traffic delays, more network connectivity, and other disruptions. On the other hand, two such stakeholders in CBDM are service seekers and service providers. The goal of service seekers is to get their parts manufactured with desired quality but at the lowest possible cost, whereas hand service providers strive to maximize the utilization of their excess capacity and profits. The mechanism designer is the matching platform that facilitates the interaction between service seekers and service providers in CBDM, whereas in FAA the mechanism designer is the policymakers who try to influence the route decisions made by airlines so as to channel the evolution towards a better performing one. In both the scenarios, the individuals who do have the information about strategies and data required to make decisions have their own private objectives and may not have the incentive to behave in the way the mechanism designer wants them to behave. In ATS, airlines make decisions based on proprietary data which is not available to the policymakers. Airlines reveal only that information which is required by law such as total passengers carried, cost etc. Most of the variables are hidden. Similarly in CBDM application such as 3D Hubs [7] for example, they keep the formula to rank the hubs proprietary to prevent the strategic behavior from the participating manufacturing hubs [8]. In ATS, we address this challenge by developing a predictive model based on openly available data while including the effect of competition. In CBDM, we perform a critical analysis of the nature of the strategic behavior of the interacting agents and propose mechanisms based on this analysis. Sometimes, the CES presents additional challenges which have not been addressed in the mechanism design literature. For example, if the matching mechanisms need to be implemented over a long duration with the stochastic arrival of service seekers and providers, then there are no studies on the frequency at which these mechanisms need to be implemented.

Now, in Sections 1.1.1 and 1.1.2 we discuss the motivation, research gaps, and research questions individually in CBDM and ATS, that are addressed in this dissertation, in detail.

1.1.1 Cloud-Based Design and Manufacturing

Cloud-based design and manufacturing (CBDM) is a decentralized, service-oriented design and manufacturing model where participants utilize product development resources, such as CAE tools and manufacturing equipment, using cloud computing, and other related technologies [9]. CBDM involves interactions among two groups of participants: service seekers and service providers as shown in Figure 1.1. *Service seekers* are designers need to manufacture or use computational resources but do not possess the capabilities to do so. *Service providers* own and operate equipment or other resources and are ready to offer users instantaneous access to these capabilities. The equipment may be 3D printers, CNC machines or some other manufacturing resources.

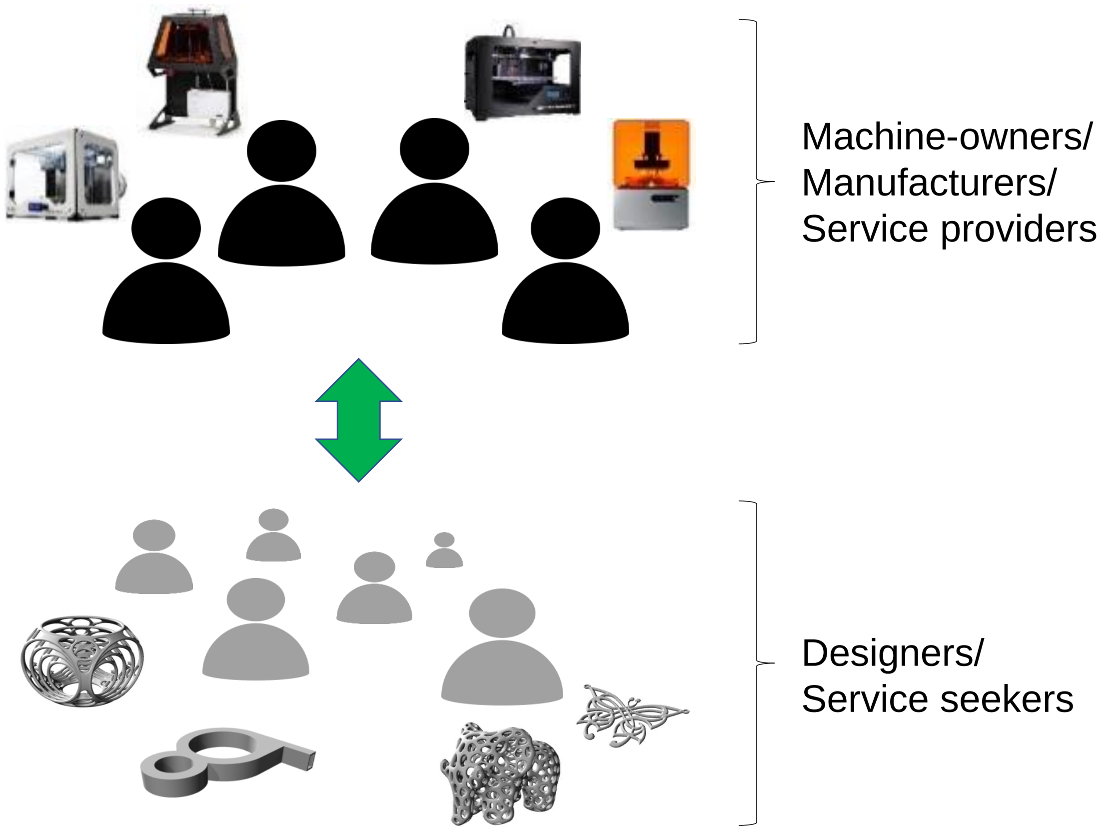


Figure 1.1. Decentrally distributed machine-owners sell their excess capacities to designers.

There are online service platforms such as 3D Hubs [7] and Shapeways [10] that try to facilitate the interaction between decentrally distributed designers and manufactures but function based on mechanisms that have several drawback. 3D Hubs has more than 25000 registered 3D printer machine owners worldwide [11]. Figure 1.2 shows the global geographical outreach of 3D Hubs. Designers who do not own a 3D printer use this platform to benefit from the manufacturing resources. In brief, 3D Hubs work as follows- the designer uploads the *stl* format of the design file into their server, their internal software lists a set of compatible and available machine owners after processing the file, the designer selects a desired machine among the available choices after which the machine owner prototypes the design and dispatches the prototype to the designer. Here, the designer are assigned to machine owners based on the choice of the designer on a first come first serve basis (FCFS). This FCFS based framework to match designers to machine owners comes with several drawbacks. Some of the drawbacks of FCFS are (a) the participating agents may have to indefinitely wait in the queue, (b) the strategic nature of interacting agents is ignored, (c) preferences of only the designers are considered while those of manufacturers are ignored. All these drawbacks are discussed in-depth in Chapter 2. There is a need to establish mechanisms that can allocate the decentrally distributed manufacturing resources to the decentralized designers more optimally.

Figure 1.2. This figure shows the world-wide geographical distribution of manufacturers registered with 3D Hubs (source: 3D Hubs [7]).

Existing resource allocation methods in job-scheduling and operations research([12], [13], [14], [15]) do not address the drawbacks discussed earlier associated with FCFS.

For example, Smith [12] proposed a job scheduling algorithm for a single machine but makes implicit assumptions that the participating agents will act as instructed. Nisan and Ronen [14] proposed a job-scheduling algorithm that accounts for the strategic behavior of the participants and Heydenreich et al. [15] extended this idea to a strategic setting where the participants may manipulate the job processing time, arrival time of job, and the cost of waiting time. However, in both the studies the focus is only on optimizing some global objective function such as overall completion time or cost and the individual objectives of the independent agents are ignored. As a result, there is a need to establish a mechanism design based resource allocation method.

Conventionally, mechanism design, as a branch of science, was developed to tackle problems in economics and more recently in computer science. Existing mechanism design methods cannot be directly applied in these applications because of the unique challenges that CES offers which has not been addressed in the historic applications for which these mechanisms were developed. For example, if matching mechanisms are used to allocate manufacturing resource in CBDM, there are no existing studies in mechanism design literature on how frequently such mechanisms need to be implemented. This is because the matching theory was developed for economics-related applications such as matching students in National residency program [16] or matching kidney donors to patients [2]. In these applications, implementing the matching mechanism is a one-time process. But in CBDM, where service seekers and service providers arrive continuously over a long period of time, these mechanisms need to be implemented repetitively over several matching cycles. Choosing the optimal frequency at which these mechanisms need to be implemented is a mechanism design issue. Another challenge is that the objective that a systems designer strives to optimize varies from one application to another. In one scenario the focus may be on maximizing the total utility attained, whereas in another it may be on the fairness of their distribution. In yet another scenario, the resource allocation approach needs to be robust towards dynamical arrival and exit of service seekers or providers. No mechanism addresses all of these objectives simultaneously. Depending on the target application the mechanism that meets the requirements needs to be chosen after analysis.

One of the primary requirements of a CBDM platform is to determine an *optimal allocation of resources considering the objectives of all the participants*. The service seekers and the service providers have different, often conflicting, objectives. Service seekers are interested in desired part quality at a minimum price, while service providers are generally interested in maximizing revenue from their available capacity. Hence, in CBDM, the goal is service matching, which involves determining which service providers will serve different service seekers.

With the aim of achieving the goal of optimal service matching in CBDM three research questions are addressed:

- **RQ1.1** How do the utilities and quality of matches attained by service seekers and service providers by implementing existing bipartite matching mechanisms compare against FCFS under resource scarce and resource abundant conditions?

- **RQ1.2** How can service seekers be optimally matched to service providers in different decentralized design and manufacturing scenarios, considering the true preferences of all agents?

- **RQ1.3** What is the optimal frequency of implementing matching mechanisms so as to maximize the matching objectives such as service seeker utility, service provider utility, and fairness in their distribution?

We use matching theory, which has been used for different matching problems such as matching students to schools, kidney donors to patients for transplant [2], and residents to hospitals [3]. This is the first application of matching theory within the CBDM context. An illustration of resource allocation in CBMD as a bipartite matching problem is shown in Figure 1.3. The applicability of different matching algorithms in different decentralized design and manufacturing scenarios and the effects of the strategic behavior of participants on the efficiency of the matching are analyzed. The influence of dynamic entry and exit of service seekers and providers on the optimality of matches, which is crucial in a CBDM framework, is studied. Insights on the effects of market thickness and resource availability on these matching algorithms are drawn using simulation studies.

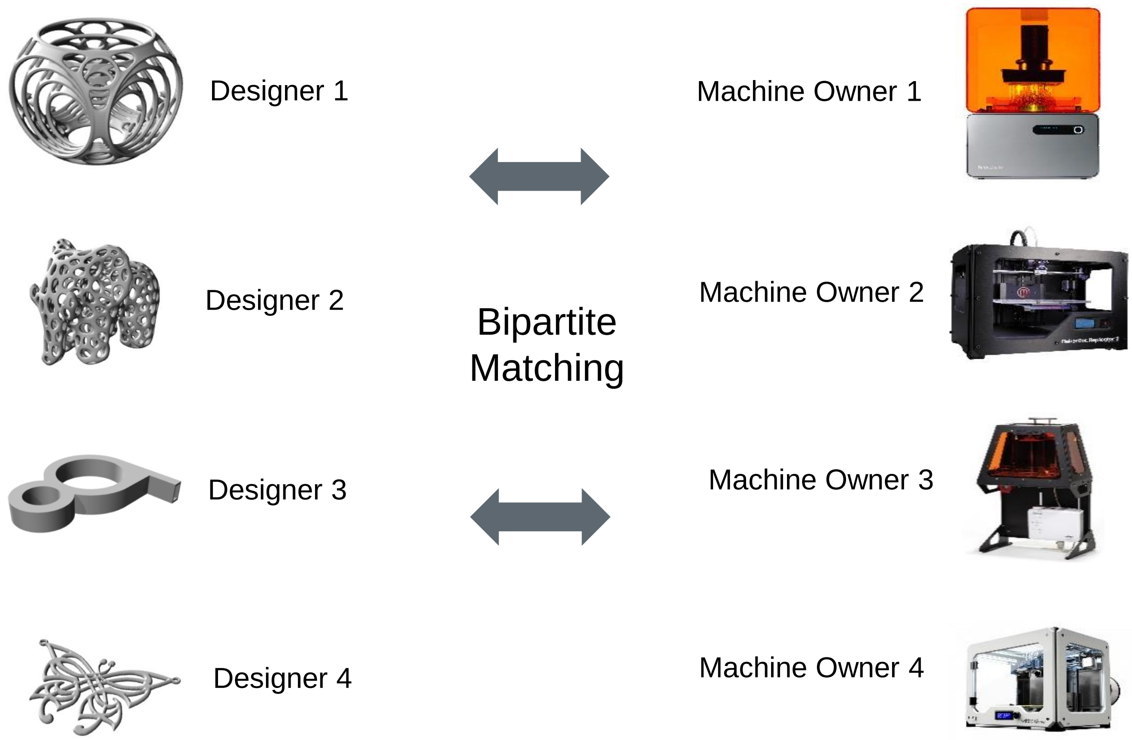


Figure 1.3. Illustrating resource allocation in decentralized design and manufacturing as a matching problem.

The influence of frequency of implementing the matching mechanisms on the optimality is studied. For example, if the frequency is too low then the quality of matches is low due to the lack of market thickness. On the other hand, if matching is performed infrequently then the waiting time is high thereby decreasing the social utility. The best matching mechanisms identified by answering $RQ1.2$ is studied. Each CBDM scenario is subjected to different resource setting (resource surplus, resource-scarce) and the performance of the mechanism in those conditions is studied.

There are **three primary contributions** in the field of cloud-based design and manufacturing from this part of the dissertation: a) framing the resource allocation problem as a matching problem and evaluating the performance of matching mechanisms in comparison to current FCFS based approach in resource-scarce and resource surplus scenarios, b) classifying CBDM applications into three main scenarios, studying the nature of interaction between the participants in these scenarios, and proposing best matching mechanisms in each of those scenarios, and c) proposing optimal frequency of implementation of matching mechanisms.

1.1.2 Air Transportation System

The US Air Transportation System (ATS) is a complex network with airports as nodes and routes connecting the airports as links [17]. Routes are operated by airlines and all such routes collectively determine the topology of the network. Frequently new routes get added and existing routes gets deleted by airlines. The evolutionary dynamics of the network topology depends on these decisions.

Such decisions on whether to operate a new route or delete an existing route are made by airlines based on a number of factors such as route characteristics such as passenger demand, distance, operating cost and competition from other airlines. These factors depend on the decisions made by other stakeholders in the Air Transportation System such as the Federal Aviation Administration (FAA), Airport Authorities, air passengers, etc. For example, FAA could incentivize certain links thereby lowering the operating cost in that link. Passenger preference towards different modes of transportation influence the passenger demand.

In ATS the topology evolves through time based on routing decisions. Every year, the number of new routes getting added and the number of existing routes getting deleted cumulatively sum to nearly 10% of the total number of routes [18]. Figure 1.4 shows the route map of DL in July 2018. It also shows the future routes that DL is planning to add and the new routes that were added in the month of July 2018.

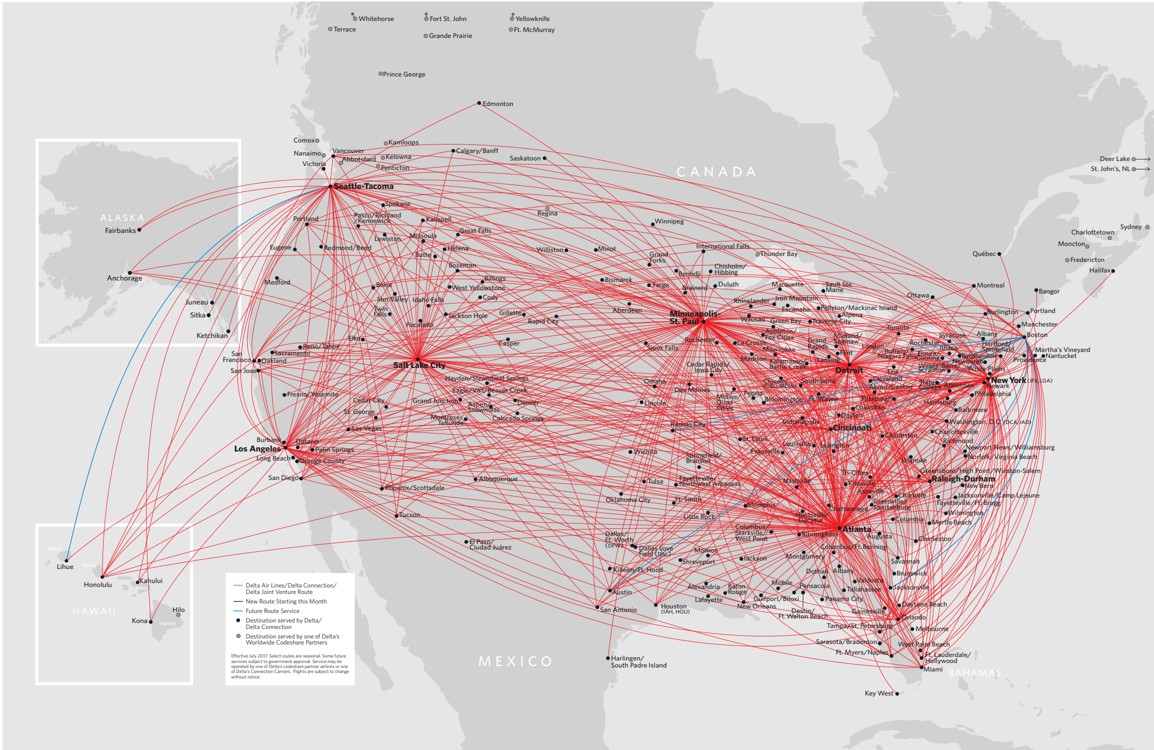


Figure 1.4. This figure shows the route map of Delta airlines in the year 2018 (source: Delta airlines [19]).

Multiple stakeholders with conflicting objectives make decisions leading to the evolution of the network in ATS. In ATS, airlines such as United Airlines (UA) and Delta (DL) frequently add new routes and delete existing ones. These route selection decisions are made in response to federal airline policies by regulatory bodies such as FAA. The airlines make these route selection decisions with the objective of maximizing their profits, market share etc. Regulatory bodies such as FAA make policies so as to channel the evolution of the network topology towards better performance. Throughout history, several policies undertaken by the FAA have shaped the evolution of the US ATS. For example, the Airline Deregulation Act of 1978 transformed the network structure of major airlines from a point-to-point into a hub-and-spoke one [20]. Figure 1.5 shows the snapshots of the historic network structure of UA.

It can be seen how UA transformed from a point-to-point structure in 1950s and 1960s into a hub-and-spoke one in 1980s. The network structure of hub-and-spoke is more convenient from the standpoint of the passengers as it offers more options on routes whereas point-to-point is less costly for airlines. The hub-and-spoke network is used by legacy carriers like UA and DL, whereas low-cost carriers like Southwest use point-to-point structure. Over time, legacy carriers use a hybrid of point-to-point and hub-and-spoke structure to capture both economies of scale and cost benefits.

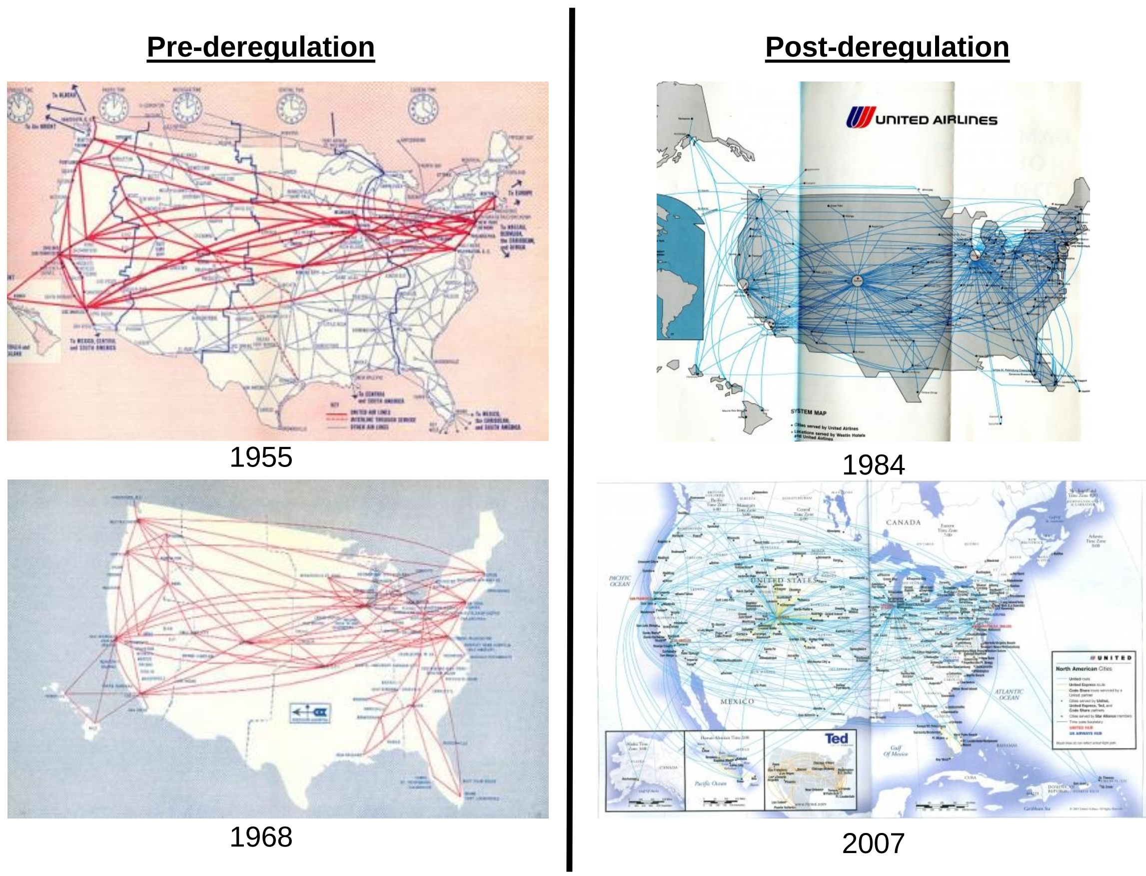


Figure 1.5. This figure shows the route map of Delta airlines in the year 2018 (source of the yearly maps: Metabunk.org [21]).

In addition to federal policy decisions by FAA, market variables such as passenger demand, operating cost, network level factors like presence of hubs, competition from other airlines, and several other factors affect the route selection decisions of airlines.

Most of the decision variables are unknown to the policymakers and stakeholders other than the airlines making these decisions. Therefore, it is important to develop a decision model that explains how airlines make route selection decisions as a function of observable factors such as market demand and operating cost. Some of these observable factors such as operating cost can be controlled by the policymaker by levying taxes or adjusting fuel prices. Such a model can guide policymakers in making informed policy mechanism decisions that channel the evolution of the network towards desired performance.

Several desirable properties such as connectivity, lack of congestion, robustness depend on the topology of the network. Route decisions made by the airlines determine the topology of the network. Hence it is important to understand how such route decisions are made by the airlines and design incentive schemes to influence such decisions so as to steer the network towards better performance.

Understanding the decision model of airlines and to design incentive schemes for policy makers. The actual decision model of the airline is not public knowledge. Airlines base their decisions on data that are available only to the airlines. Federal policy makers aim to maximize social welfare in an environment where airlines make routing decisions with the sole intention to maximize their profit. To address this challenge two research questions are answered:

- **RQ2.1** How do the airlines make routing decisions in the presence of competition from other airlines?

- **RQ2.2** How are the route decision strategies of the airlines affected by varying the explanatory variables such as market demand and operating cost?

In this dissertation the analysis is limited to the route structure only. While route structure is a critical strategic choice there are other strategic decisions made by the airlines such as how many flights to allot, which aircraft and how much load to carry in those, etc. However, this is not a limitation of the model. The approach developed can be extended to study such decisions as well. We focus our studies on

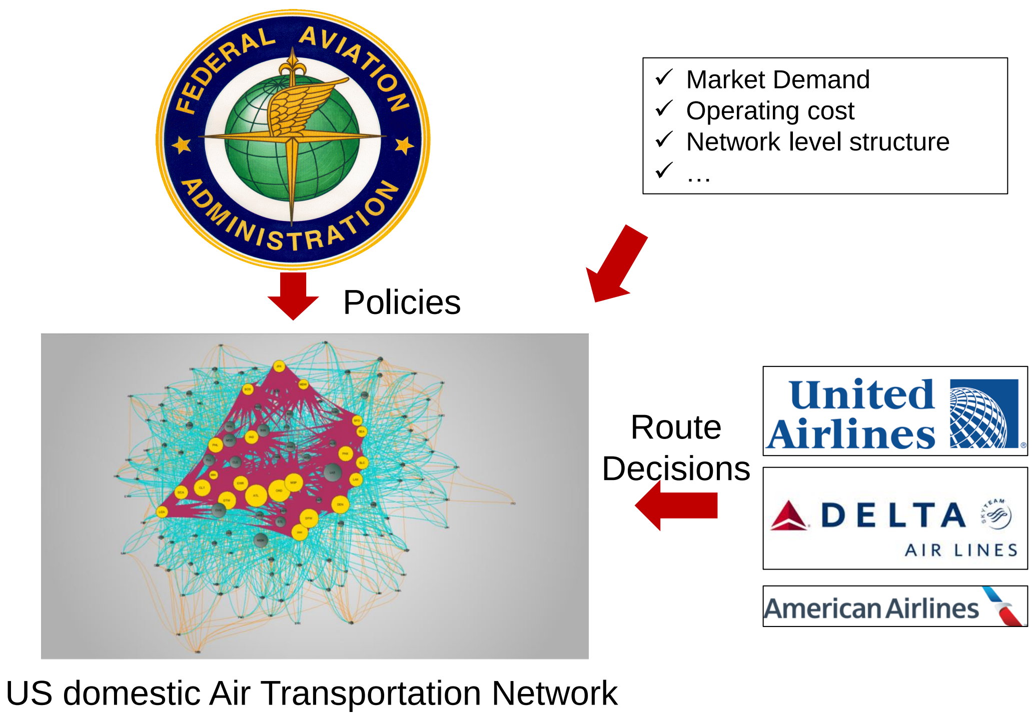


Figure 1.6. Interaction between multiple stakeholders resulting in the topology evolution of the US ATS.

route structure because the topology forms the backbone of the network. Properties such as robustness, connectivity, resilience depend on the topology. Hence, it is in the interest of the policymakers to guide the evolution of the network topology into the targeted structure so as to achieve targeted performance.

Data on past decisions made by the airlines within the US ATS provide inferences on the decision parameters. Because the decision on whether or not to operate on a route is a discrete choice and involves the decision made by a competitor, our method is based on the discrete games. At each route, the airlines play a strategic game on whether to operate or not to operate. The strategic decision of whether to operate is based on preferences for route characteristics such as demand, cost, and competition from other airlines. First, preference parameters are solved ignoring competition using discrete choice analysis assuming route decisions are independent. Using the information gained on the preference parameters as prior, and using the historic routing decisions under competition as a basis for likelihood functions Markov Chain Monte-Carlo (MCMC) Metropolis-Hastings algorithm is used to obtain estimates on the decision parameters. The decision model so developed using those parameters are to predict the topological evolution of the network.

Using the inferences gained on routing decisions by airlines we aim to guide policymakers to design mechanisms that can channel the network into better performance. The topology evolves from one period to next based on the routing decisions. Once we develop a model that mimics the routing decisions of the airlines, the next step is to control the airline decisions through policy decisions. It is important to understand the implications of the policy while designing such policies. Our aim is to guide such policymakers in designing the right policies based on its implications. This broad objective is achieved in two steps: first, a network-level welfare metric that can assist the policymakers in quantifying the effectiveness of their policies, perform different policy experiments and evaluate its implications on network performance (quantified by the metric developed). An important policy decision under question is single-till versus double-till regulation in airports. There is an ongoing debate about which policy is more effective from a social welfare point of view. All the existing studies quantify social welfare based on total profits that the airlines and airports make. Different from those, we quantify social welfare based on network-level performance and aim to compare the effectiveness of both these policies to guide the regulatory bodies. Finally, we validate our approach based on important historic policies made such as the Wright Amendment, on how the actual implications compared with predicted implications.

The **two primary contributions** in ATS are: a) using an airport presence based approach to estimate parameters of discrete games to improve prediction accuracy, and b) using forward simulation to understand the behavior of the airlines in response to variation in explanatory variables.

1.2 Overview of the Dissertation Document

The rest of the document is structured as follows:

Chapter 2 formulates resource allocation in CBDM as a matching problem. The steps involved in implementing the matching mechanism in CBDM are described. Further, three different matching mechanisms are analyzed. Simulation of an illustrative decentralized additive manufacturing setup is made to compare the efficiency of the three mechanisms against the FCFS approach. Conclusions are drawn to compare the efficiency of the three mechanisms in both resource scarce and resource surplus conditions.

Chapter 3 presents three different scenarios in cloud-based design and manufacturing. The strategic behavior of the participants, information, and other interaction is analyzed and compared for these scenarios. Based on the analysis, the requirements of an appropriate resource allocation mechanism in the three scenarios are studied. Matching mechanisms are recommended for each of the three CBDM scenarios based on how well the properties of the matching mechanism satisfy the requirements in those scenarios.

In Chapters 2 and3 a one-time only implementation of the matching mechanism is presented. However, in practice, these matching mechanisms are employed over a long period of time and would need to be implemented multiple times. It is necessary to determine the optimal frequency at which these mechanisms are implemented so as to maximize various matching objectives such as utility attained by the participating agents, number of successful matches, and fairness in their distribution. In Chapter 4 we study the effect of frequency of implementing mechanisms on the outcome of matching and propose optimal frequencies as a function of variables that characterize a CBDM scenario. Some of these variables are the arrival rate of designers, service rate of manufacturers, average job processing time, number of service providers etc.

In Chapter 4 the frequency recommendations are restricted to a certain idealized environment with strong assumptions such as deterministic arrival process, constant job processing time etc. Some of these assumptions are not valid in real-world applications. In Chapter 5 we see the effect of relaxing these assumptions on the optimal matching frequency. We also simulate a stochastic CBDM matching scenario relaxing all the assumptions to mimic a real-world setting. In the simulation, we study how the frequency recommendations drawn from the studies generalize into the real-world environment.

In Chapter 6 we use a discrete game based approach to model the decision-making behavior of the airlines in the presence of competition. We use airport presence to evaluate the parameters of the discrete games from historic decisions. The parameters studied are used to study the preference characteristics of the airlines towards different explanatory variables.

In Chapter 7 we perform forward simulation to predict the evolution of US ATS. Prediction accuracies from the forward simulation are compared with actual decisions and other competing models. The effects of varying the route characteristics on the evolutionary behavior of the network are studied.

Table 1.2. : Summary of research questions, approaches, and contributions from this dissertation in CBDM and ATS.

Case	Research Questions	Research Approach	Contributions
CBDM	(RQ1.1) How do the utilities and quality of matches attained by service seekers and service providers by implementing existing bipartite matching mechanisms compare against FCFS under resource scarce and resource abundant conditions?	Simulation studies were performed on an illustrative 3D printing service framework.	• New understanding on the influence of resource availability on existing bipartite mechanisms; • Insights on the effect of resource scarce and resource surplus conditions on the number of successful matches, utility attained from matches by both the set of service seekers and service providers.
	(RQ1.2) How can service seekers be optimally matched to service providers in different decentralized design and manufacturing scenarios, considering the true preferences of all agents?	• Compiling the properties of mechanisms that are crucial for resource allocation in CBDM; interpreting the properties in the context of CBDM; • Analyzing stakeholder behavior and nature of interaction in 3 CBDM scenarios; • Analyzing the major requirements in the scenarios; • Recommending best matching mechanisms for the scenario based on how the properties of the mechanism match to the requirements in the scenario; some insights from resource simulation results by answering RQ1.1 were also used	Best matching mechanisms were recommended for three CBDM scenarios that generalizes to a broad range of applications

Table 1.2. : Summary of research questions, approaches, and contributions from this dissertation in CBDM and ATS.

	(RQ1.3) What is the optimal frequency of implementing matching mechanisms so as to maximize the matching objectives such as service seeker utility, service provider utility, and fairness in their distribution?	• Theoretical proposition for optimal period at which the matching mechanisms need to be implemented under certain assumptions about distribution of job processing time and service rate; • Studying the effect of relaxing the assumptions through simulation; • Validating the approach by simulating an illustrative CBDM setting to see how well the results generalize.	• Recommendation on the optimal period at which matching mechanisms should be implemented given the four input conditions: a) job processing time, b) service rate, c) number of service providers, and d) arrival rate of service seekers; • These results generalize even in applications outside of CBDM.
ATS	(RQ2.1) How do the airlines make routing decisions in the presence of competition from other airlines?	Discrete games based model was used to model the route level decisions of the airlines. An airport presence based Bayesian estimation technique was used to estimate the parameters of the discrete games. The data of historic decisions of two major airlines, UA and DL, were used to test the prediction accuracy of the model.	A model for network evolution of airlines based on route-level decisions made by the airlines under competition. An airport presence based approach to estimate the parameters of the discrete games.
	(RQ2.2) How are the route decision strategies of the airlines affected by varying the explanatory variables such as market demand and operating cost?	Forward simulation on the decision model of the airlines created by answering RQ2.1.	Understanding how the variations in explanatory variables affect the Nash equilibria strategies of the airlines.

2. MECHANISM DESIGN APPROACH TO RESOURCE ALLOCATION IN CBDM

CBDM involves interactions among two groups of participants: service seekers and service providers. *Service seekers* need to manufacture or use computational resources, but do not possess the capabilities to do so. *Service providers* own and operate equipment or other resources and are ready to offer users instantaneous access to these capabilities. An effective CBDM platform must be able to effectively support the important tasks of resource discovery, service scheduling, service matching. Several research efforts have been focused on issues such as resource virtualization technologies, resource and service publication and discovery [22], service composition, efficiency [23], reliability and security management [24]. A review of challenges and research gaps in these emerging manufacturing models is provided by Tao and co-authors [25].

2.1 Illustrative Example of CBDM: 3D Printing Services

In this section we use a 3D Printing service platform as an illustrative example to describe the stakeholders and the nature of interaction between them in CBDM.

Additive manufacturing is bridging the gap between designers and manufacturers by enabling rapid transition of concepts into physical prototypes and final products. The increasing popularity of additive manufacturing is partly due to the availability of mid-level consumer grade 3D printers, and access to robust 3D modeling software for the creation of geometric models.

To serve designers for whom it is economically not viable to own different printers for their needs, there has been an emergence of service organizations, such as Shapeways [10], who own a variety of 3D printing machines. The machines range from desktop printers for plastic parts to industrial scale metal printers, giving designers a myriad of options to choose from based on their needs. Designers can submit geometric models to these organizations and get them printed at the quoted price. These organizations typically also offer quality checks and assistance to designers to help them market and sell their products in return for a commission.

In addition to 3D printing service organizations, an alternate, decentralized scenario exists where designers who do not possess the necessary prototyping resources are able to connect with independent individuals who own those resources. These interactions are facilitated by service matching organizations, such as 3D Hubs [7], where designers upload their 3D models and are able to choose from the available providers. Machine owners can register their services at these platforms and advertise their printing resources, and designers can avail these services on a FCFS basis by choosing the machines that best suit their needs. The machine owner completes the 3D printing task for a price decided based on designer's requirements.

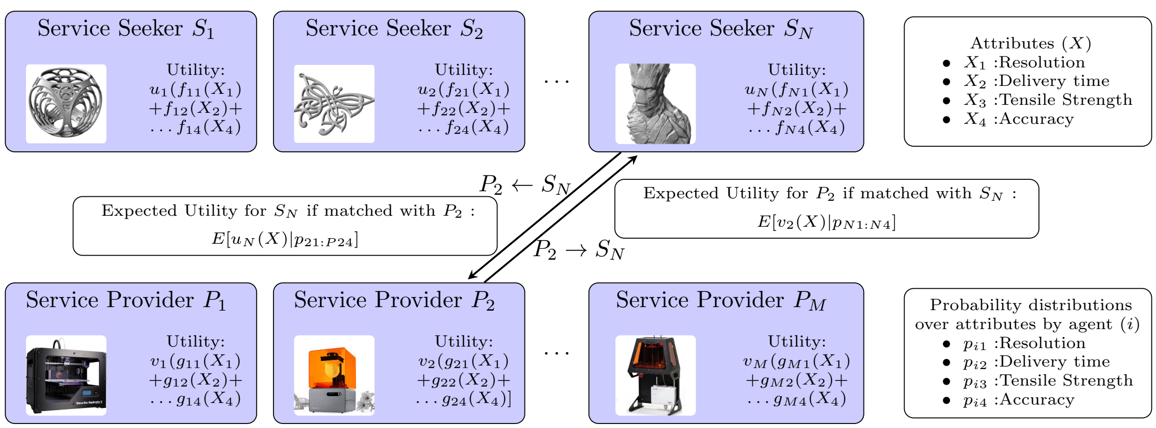


Figure 2.1. Matching in decentralized design and 3D printing services.

The scenario is illustrated in Figure 2.1.

2.1.1 Interacting Agents

There are three main staheholders interacting with one another. They are service seeker, service providers, and the individual or organization implementing the matching mechanism.

In CBDM, the service seekers are designers who are trying to get a job done but do not possess the resources to get the job done and are seeking the same from the system. For example, these may be designers who are trying to get their design parts prototyped but do not possess the machines to do so. The seekers can also be individual or organization needing computational resources for a task.

Service providers are machine owners who own resources with excess capacity and are willing to sell the capacity by processing job requests of service seekers. For example, a manufacture may be an individual owning a CNC machine or production facility or 3D printer.

The third agent, matching agent, connects the set of service providers to the set of service seekers. Sometimes the matching agent would overlap with the service provider. For example, all the resource would be owned by the same organization and the organization itself would be responsible to create the matching platform. Shapeways [10] Depending on the application the objective that the matching platform are trying to optimize would vary. For example, if all the organization who owns all the resources are implementing the matching mechanism then their objective would be only to maximize their profit. On the other hand, if it is a third party that is creating the matching platform then they would consider optimizing the objectives of both service providers and service seekers.

2.1.2 Attributes and Preference Characteristics of the Interacting Agents

Each designer and machine owner has different preferences from each other. The designers' preferences for machines are based on machine-owner's attributes such as resolution, tensile strength, and maximum build size. Resolution is an important attribute because the quality, detail capability, finish and accuracy of the final printed part depend on it. The strength of the final part affects its usability as a functional prototype. The machine-owner preferences are based on design attributes, such as resolution requirement, geometric properties, and printing time. Resolution is important to machine-owners because (i) the machine-owners prefer to fully utilize their resolution capabilities to maximize profits, and (ii) machine-owners may prefer to print other products in the same run and would consider a particular resolution range as an ideal match. Geometric properties, such as size, are important because they affect needs for the machine's build area.

2.2 Why Use Matching Mechanisms for Resource Allocation in CBDM?

The adoption of additive manufacturing (AM) and other advanced manufacturing technologies appears to herald a future in which value chains are shorter, smaller, more localised, more collaborative, and offer significant sustainability benefits [26].

2.2.1 Issues With Conventional Resource Allocation Methods

Conventional resource allocation methods based on multi-objective optimization are inappropriate for matching resources to service seekers in decentralized scenarios because they optimize the objective of one party only. The commonly used approach for matching in decentralized scenarios is a *marketplace* where service providers display capabilities and prices at a central location (e.g., on a website), and the service seekers self-select the providers based on their needs. This is a first-come first-serve approach (FCFS) because if a service provider's resource is available, it can be used by the service seeker who requests it first, and is willing to pay the asking price. Such a model is adopted in 3Dhubs [7], an online 3D printing service platform with around 25,000 service providers.

In a CBDM setting using FCFS to match service seekers to service providers leads to several inefficiencies because of the nature of the interacting environment. Firstly, the service seekers have to wait indefinitely in a queue if they opted for service from a highly sought after service provider. There is no effective way of quantifying waiting time. Even if some heuristics are proposed to quantify such waiting periods, the service seekers have to make hard decisions such as choosing the trade-off between high waiting period or lower quality service. This problem is more severe when the resources are scarce. Secondly, in FCFS service seekers select their most preferred choice or set of choices. The allocation is made based on these choices in the order of arrival time. There is no provision to consider the preferences of the service providers. Service providers are passive agents in this setting who processes service requests of their allocated jobs. The preferences of the service provider being not explicitly considered may lead to dissatisfaction among the service providers. For example, a service provider with a high resolution 3D printer, which is more suitable to print jobs that demand higher detail capability, may be chosen first by a seeker who does not need such capability. A major reason for the failure of online B2B markets for manufacturing back in early 2000s was the dissatisfaction of the service providers [27]. Third, even in cases where two service seekers have first preference for the same service provider, the utility that each service seeker gains may be significantly different. FCFS considers only the order of arrival times without considering these utilities. Therefore, the match obtained from FCFS may not be optimal from the standpoint of the entire set of participants. Fourth, it is possible for participants to try and "game" the system by exhibiting strategic behavior, i.e., considering other participants' objectives and stating preferences that are different from their true preferences. For example, a service seeker may consider how much delay would result if he/she seeks the resource that best matches his/her requirement as there may be several other seekers in the queue prolonging the response time. When this happens it is not optimal for a service seeker to state his/her true preferences, but rather based on expectations about other service seekers' preferences. Finally, FCFS does not account for the specific requirements of different organizational scenarios. For example, for a central service provider organization, such as Shapeways [10], where all the resources are owned by the same company and the service seekers are independent designers who are interested in printing their parts, the objective is to allocate the jobs to the resources to maximize the total utility gained by the organization. On the other hand, in a totally decentralized scenario, such as 3Dhubs [7], the utilities of all service providers and seekers need to be accounted for in the matching algorithm. These two scenarios present diverse set of challenges.

2.2.2 Why Mechanism Design based Approach

To address the limitations of the FCFS matching mechanism, the central question addressed in this chapter is: *How can service seekers be optimally matched to service providers in different decentralized design and manufacturing scenarios, considering the true preferences of all agents?* We propose the use of matching theory, which has been used for different matching problems such as matching students to schools, kidney donors to patients for transplant, and residents to hospitals. To the best of our knowledge, this is the first application of matching theory within the CBDM context. We analyze the applicability of different matching algorithms in different decentralized design and manufacturing scenarios. The effects of strategic behavior of participants on the efficiency of the matching are analyzed. We also study the influence of dynamic entry and exit of agents on the optimality of matches, which is crucial in a CBDM framework. Finally, we draw insights on the effects of market thickness and resource availability on these matching algorithms through simulation studies.

2.3 Steps for Matching

We propose three steps in optimal matching of designers and manufacturers within the CBDM framework. The steps are shown in Figure 2.2. The first step involves quantification of preferences of designers and manufacturers using the expected utility theory [28]. The second step involves ranking of alternatives of each participant based on the expected utilities. The third step is to analyze the CBDM scenario and utilize the matching algorithm based on the most relevant criteria based on interaction, objectives and private information of agents. These steps are discussed in detail in Sections 2.3.1 through 2.3.3.

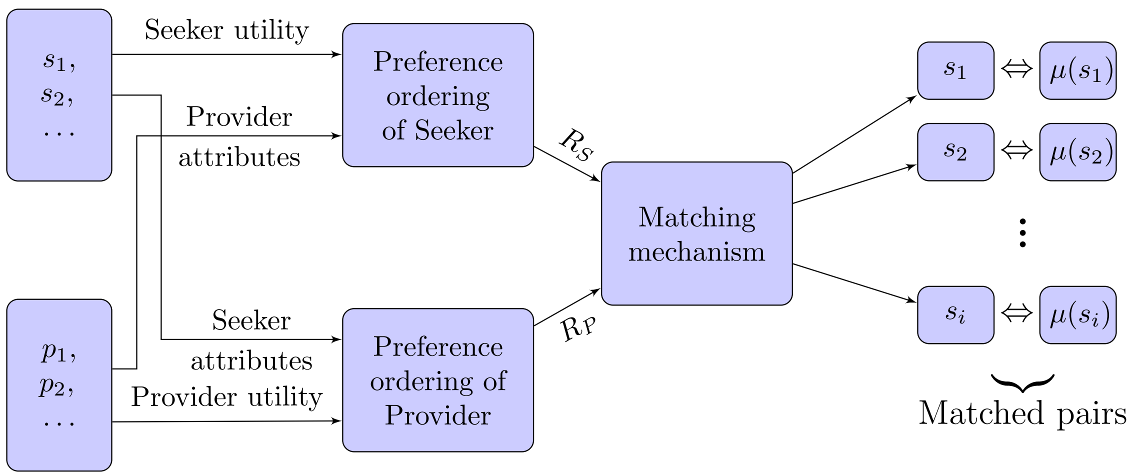


Figure 2.2. The above flow chart summarizes the approach followed.

We consider a set of service seekers (i.e., designers), $S = \{s_1, s_2, \ldots, s_{|S|}\}$, availing manufacturing services from a set of service providers (i.e., manufacturers), $P = \{p_1, p_2, \ldots, p_{|P|}\}$. Service seekers and service providers are collectively referred to as agents $A = S \cup P$ and total number of agents $|A| = |S| + |P|$. Agents in P constitute the alternative set for agents in S, and vice-versa. Each service provider, p_j, offers a cap on the maximum number of service seekers it can serve, which we call vacancy denoted by q_{p_j} with $q_{p_j} \geq 1$. We assume that the cap is in terms of the number of designs a service provider can manufacture, and not the amount of time the manufacturer is willing to work.

2.3.1 Quantification of Preferences of Service Seekers and Service Providers

We use the utility-based procedure for quantification of agents' preferences, as demonstrated for Fernández et al. [29]. To quantify the preference characteristics of the agents, the first step is to identify all the attributes that service seekers and service providers consider while evaluating their preferences. All service seekers in S value a set of attributes of the service providers in P, e.g., machine resolution and the attributes of the material offered. Similarly, the service providers in P value certain service-seeker attributes such as printing time. We represent the union of service seeker and service provider attributes as $X = \{X_1, X_2, \ldots, X_n\}$. In the rest of the paper, s_i and p_j denote i^{th} service seeker and j^{th} service provider respectively, while X_k denote the k^{th} attribute.

Based on the preferences for each attribute X_k, service seeker s_i's single-attribute utility functions f_{ki} are obtained using standard utility assessment procedures (see [28] for details). The single attribute utility functions f_{ki} are then combined into s_i's multi-attribute utility function, u_i. We have $u_i(X) = u(f_{1i}(X_1), f_{2i}(X_2), \ldots, f_{ni}(X_n))$. For illustration, assuming the additive form of the multi-attribute utility function, s_i's utility function is $u_i(X) = \sum_{k=1}^{n} w_{ki} f_{ki}(X_k)$, where w_{ki} is the weight that s_i associates to attribute X_k, such that $\sum_k w_{ki} = 1$. Note that each service seeker $s_i \in S$ may care only about a subset of relevant attributes in the set X. These subsets are generally different for different agents, particularly between service providers and service seekers. The weights for those attributes that the agents do not care about are assigned as zero.

The next step is to define the probability function of the attribute value. For service seeker s_i we are interested in the probability function of the attributes of service provider p_j. We represent the pdf of attribute X_k associated with service provider p_j as $p_{kj}(X_k)$. The probability distribution and the multi-attribute utililty functions are combined to obtain the expected utility that s_i gains by being matched to each of the s_i's alternatives p_j. For the case of additive multi-attribute utility functions the following equation is valid for expected utility calculation,

$$E[u_{ij}(X)] = \sum_{k=1}^{n} w_{ki} \int [f_{ki} p_{kj}] dx \quad (2.1)$$

where $E[u_{ij}(X)]$ represent the utility of service seeker s_i for provider p_j. Repeating the same steps for service providers, the expected utility gained by service provider p_j being matched to each of its alternatives s_i is given by

$$E[u_{ji}(X)] = \sum_{k=1}^{n} w_{kj} \int [f_{kj} p_{ki}] dx \quad (2.2)$$

In cases where additive independence of attributes is not valid the expected utility expression is re-formulated. However, a detailed discussion of such special cases is beyond the scope of this paper. Fernández et al. [29] provide additional insights on such reformulations for the 3D printing scenario.

2.3.2 Ranking of Alternatives

Based on the expected utilities thus generated, the alternatives for each agent are rank ordered. Higher the expected utility, lower is the rank. We include the agent itself in the set of its alternatives. All the alternatives that are unacceptable to the agent are ranked below the agent. An agent matched to itself represents an unmatched agent. For example, if the design cannot be printed on any of the available machines because the volume exceeds the capacity of each machine, then the design remains unmatched to any of the machines. Therefore, the set of alternatives for a seeker (s_i) is the set of providers ($P \cup s_i$), and the set of alternatives for a provider (p_j) is the set of seekers ($S \cup p_j$).

The ranking of the alternatives of each agent s_i is represented as R_{s_i}. The preference ordering thus generated is assumed to be complete, anti-symmetric and transitive as utility theory is based on assumption of rational behavior. We define $R_{s_{-i}}$ as the preferences of all service seekers except s_i and R_S as the preference ordering of all service seekers combined. We have $R_S = R_{s_{-i}} \otimes R_{s_i}$. Here, $\otimes$ symbolizes the product of vector spaces. The preference ordering of each agent is a vector and falls in a vector space. Preference ordering of a collection of agents falls in product of these vector space. Similarly, for all service providers, $R_P = R_{p_{-j}} \otimes R_{p_j}$.

2.3.3 Matching Algorithms

Matching algorithms are aimed at implementing matching mechanisms [30]. In this paper, we use the words *mechanism* and *algorithm* interchangeably. A mechanism is a mathematical structure that models institutions through which economic activity is guided and coordinated [31]. A mechanism results in a matching $\mu : S \to P \cup S$ that

(i) assigns each service seeker to an alternative, i.e., $\forall s_i \in S : \mu(s_i) \in P \cup s_i$, where $\mu(s_i)$ is the service provider to which service seekers s_i is matched and

(ii) implements the matching without exceeding the vacancy of any service provider, i.e., $\forall p_i \in P : |\mu^{-1}(p_i)| \leq q_{p_i}$; where $\mu^{-1}(p_i)$ is the set of service seekers matched to service provider p_i

The best matching mechanism μ^* matches S and P with the preference profile $R = R_S \otimes R_P$ in the most optimal way. The notion of optimality is based on the scenario and the desired properties for that scenario.

In the following, we present three different matching mechanisms: Deferred Acceptance (DA), Top Trading Cycles (TTC), and Munkres' mechanism.

Deferred Acceptance (DA) mechanism

The first DA mechanism was proposed by Gale and Shapley [32] as a solution to the stable marriage problem[1]. Gusfield [3] discusses the extensions and implications of this mechanism in detail. This mechanism guarantees a stable matching solution in finite time. Even 50 years after it was first formulated, DA is still applied in the National Residency Matching Program [16] where medical residents are assigned to hospitals based on the preference ordering of both residents and hospitals. It has also been used in matching workers to firms [34]. Within the service seeker and service provider context, the mechanism can be used as follows:

Algorithm 1 Deferred Acceptance (DA) mechanism

Set each $s_i \in S$ as unassigned and each $p_i \in P$ as totally unsubscribed
while($\exists p_i \in P$ who is undersubscribed) and ($\exists s_k \in R_{p_i}$ not provisionally assigned to p_i) **do**

1. s_i is first such s_k in R_{p_i} and s_i is provisionally assigned to p_j
2. unassign s_i from p_j and provisionally assign s_i to p_i
3. for each successor p_k on R_{s_i} remove p_k and s_i from each other's list

end while

Top Trading Cycles (TTC) mechanism

TTC was proposed by Shapley and Scarf [35]. TTC has been successfully applied for kidney exchange [2] and matching students to schools [36]. The matching generated by TTC enjoys useful properties such as group-strategy proof and efficiency, but lacks stability (see further details in Section 3.3.2). The mechanism is implemented in the service matching scenario as follows:

---

[1]Extensions of the basic mechanism have been proposed, for example [33]. Here we focus only on the fundamental mechanism

Algorithm 2 Top Trading Cycle (TTC) mechanism

Set each $s_i \in S$ as unassigned and each $p_i \in P$ as totally unsubscribed
while$(\exists p_i \in P)$ or $(\exists s_i \in S)$ **do**

1. All $s_i \in S$ points to top preferred $p_j \in P$ and vice-versa;
2. any agent who prefers to remain unmatched points to itself forming self-cycle;
3. each s_i is allotted the p_j it points to and vice-versa;
4. all allotted s_i are removed and capacity of all allotted p_j is reduced by one;
5. remove all p_j whose capacity reduces to zero

end while

Munkres' mechanism

Munkres' mechanism [37] can optimize the expected utility only for one set of agents. Unlike TTC and NRMP, Munkres optimizes the absolute value of expected utility attained by one set of agents. The details of the mechanism are as follows:

Algorithm 3 Munkres' mechanism

Populate matrix $B_{o \times o}$, where $o = \max(|S|, |P|)$ such that element $b_{i,j}$ of B represent utility of $s_i \in S$ for $p_j \in P$; empty cells in the matrix are replaced by the largest entry

while(the number of lines covering the zero elements equals the number of rows in matrix B) **do**

1. For each element in a row subtract the minimum value;
2. repeat the above step for columns;
3. cover each zero in the matrix with minimum lines
4. add the minimum uncovered element to every covered element; if covered twice, add the minimum element twice

end while

while(either $S \neq \phi$ and $P \neq \phi$) **do**

1. select a matching pair by choosing a set of zeros
2. reduce the capacity of matched manufacturers by one;
3. remove those $p_j \in P$ whose capacity reduced to zero and remove those $s_i \in S$ who were matched;
4. repopulate matrix B with the updated S and P;

end while

2.4 Simulation Results

For the three matching algorithms we simulate an illustrative CBDM scenario and we do simulation studies to compare the efficiency of the DA, TTC, and Munkres mechanism mechanism against the FCFS method. We evaluate the performance under various resource conditions: resource scarce (where resources are scarcer than the demand for them), resource balanced, and resource surplus.

In the decentralized service seeker-provider scenario, discussed in Section 2.3, there are $Permutation(\sum_i^y q_{p_i}, x)$ possible unique matching combinations. The best mechanism for a scenario is the one that best satisfies the properties in Table 3.2. In

Section 3.3, a comparison of the performance of the mechanism is presented based on various properties. The comparison does not provide information about the effects of uncertainty, such as uncertainty in the order of arrival of service requests and preference characteristics. Additionally, the implications of scarcity or abundancy of resources on the performance of the mechanisms are also not clear based on these properties. As an example, if resources are abundant such that for every new service request there is an available slot on the first preferred resource provider then FCFS would perform well. To analyze the effects of uncertainty and resource availability, simulation studies are carried out. Simulation results related to the performance of the mechanisms under scarce, balanced and surplus supply of resource are discussed in this section.

Two measures are used to compare the efficiency of different mechanisms: (i) average rank, and (ii) total expected utility. An agent s_i has rank r if matched to its r^{th} choice in the agent's preference ordering. The average rank of designers (manufacturers) is obtained by averaging the rank of all designers (manufacturers) matched using the mechanism. The total expected utility of designers (manufacturers) is obtained by aggregating the utility gained by all the designers (manufacturers) through the matching mechanism. By definition utility attained by an unmatched designer is zero. The total utility of a set quantifies the collective performance of the set of agents, whereas average rank over all realized matches in a set quantifies the extent to which individual preferences are met. Thus, both these measures are used to evaluate the performance of different matching mechanisms.

The following three cases of resource availability are simulated:

- resource scarce case, where $\sum_i^y q_{p_i} < x$;
- resource balanced case, where $\sum_i^y q_{p_i} = x$; and
- resource surplus case, where $\sum_i^y q_{p_i} > x$.

As an illustrative scenario, there are 25 designers being matched to 5 manufacturers. Assuming that all manufacturers serve the same number of designers, $q_{p_i} < 5$

represents a resource scarce case, $q_{p_i} = 5$ is a resource balanced case, and $q_{p_i} > 5$ is resource surplus. To represent the three cases, simulation results are presented for $q_{p_i} = 3$, $q_{p_i} = 5$, and $q_{p_i} = 7$ for each manufacturer. The insights drawn are consistent for other values of q_{p_i} as well.

2.4.1 Illustrative Example: 3D printing Service Framework

Consider a scenario where a set of 25 designers ($|S| = 25$ where service seeker set S is the set of designers) are seeking services from a set of 5 manufacturers ($|P| = 5$). The 25 designs used in this example were downloaded from Thingiverse [38]. Each of the 5 manufacturers owned one of these five 3D printers: Makerbot 2 (P1), Ultimaker 2 (P2), Witbox (P3), B9 Creator (P4), Form 1+ (P5). First three machines are FDM (Fusion Deposition Material) machines while the last two are based on the SLA (StereoLithography) process. Each manufacturer owned materials compatible with their respective machine. Material data was collected from iMaterialise [39] for different FDM and SLA materials. Representative images of the designs used are shown in Figure 2.3.

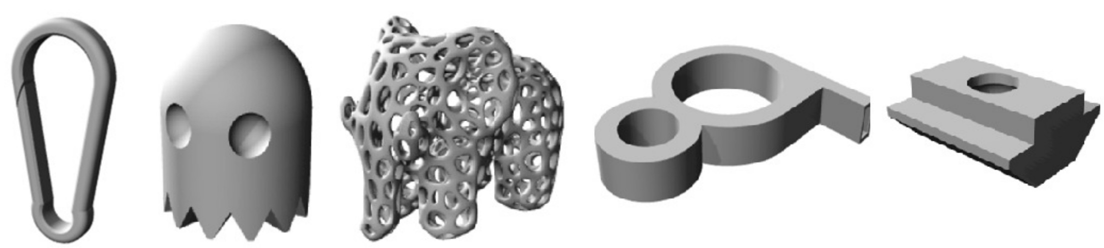


Figure 2.3. Sample designs used in the illustrative scenario.

The manufacturer attributes concerning the designers are machine volume, machine resolution, material tensile strength, manufacturer proximity whereas designer attributes concerning the manufacturers are printing time, material requirement, and design dimensions. The resolution capabilities of the machines varied from 5 to 100 microns. Designers could list their job on an urgency scale from 1 to 5. Further the designers could state their resolution and tensile strength range requirements. The numerical values used for the design printing time and machine dimension attributes are listed in Table 2.1.

Table 2.1.
Printing time and design dimensions corresponding to each designer-manufacturer pair. The machines are numbered from p_1 to p_5 and 25 designs are numbered from s_1 to s_{25}, The 25 designs used in this example were downloaded from website thingiverse [38]. 3D printers used were Makerbot 2 (p_1), Ultimaker 2 (p_2), Witbox (p_3), B9 Creator (p_4), Form 1+ (p_5). Material data was collected from iMaterialise [39]. Empty cells denote incompatible design-machine pairs.

	Printing Time in machine type (hr)					**area (cm^2)**	**vol (cm^3)**
	p_1	p_2	p_3	p_4	p_5		
s_1	0.28	0.35	0.40	0.12	0.17	18.6	15.55
s_2	0.15	0.13	0.15	0.13	0.15	304.41	94.08
s_3	0.13	0.08	0.12	0.25	0.20	407.77	13.44
s_4	0.03	0.03	0.05	0.33	0.23	0.62	0.04
s_5	2.27	1.77	1.83	0.72	0.70	4.28	11.13
s_6	0.75	0.83	1.13	0.97	0.60	23.31	1.06
s_7	0.77	1.37	1.07	1.20	0.88	49	28.92
s_8	1.93	2.02	2.28	2.15	1.43	13.14	0.67
s_9	3.15	3.93	4.90	2.67	1.73	57.5	9.99
s_{10}	6.25	7.67	8.97	3.30	2.40	103.58	26.71
s_{11}	2.25	1.55	1.88	3.68	2.25	74.01	21.23
s_{12}	4.73	6.15	6.85	4.27	2.68	38.36	29.62
s_{13}	9.48	11.28	13.00	4.65	6.25	56.91	199.65
s_{14}	7.73	10.13	11.15	5.13	5.33	1.32	0.42
s_{15}	2.07	1.92	2.00	5.77		56.5	330.65
s_{16}	12.92	16.88	18.52	7.35	8.20	6.25	11.27
s_{17}		2.00				116.91	16.43
s_{18}	3.88	4.10	4.78		0.97	76.42	298.12
s_{19}	3.48	3.62	3.93		1.13	57.3	286.37
s_{20}		11.02	12.50			58.87	55.05
s_{21}	13.82	17.28	19.10		6.65	72.4	77.79
s_{22}	24.87		34.15			43.37	2.3
s_{23}	11.98	12.80	16.92		5.38	99.97	1227.76
s_{24}			10.37			30.98	12.73
s_{25}			48.92			340.62	54.06

Utility functions of individual agents are derived and preference ordering is generated using the procedure described by Fernández et al. [29]. An example of expected utility of service seeker (designer) s_1 for service provider (manufacturer) p_2 is described. Service provider s_1 preferred the attribute tensile strength with a weight of 0.8 and resolution with a weight of 0.2. The left hand side utility function of designer s_1 for attribute resolution x_1 is defined as $f_{11}(x_1) = -0.003\ {x_1}^2 + 0.049\ x_1 - 0.9446$ in the range $22\mu m < x_1 \leq 60\mu m$, and the right hand side utility defined between $60\mu m < x_1 \leq 90\mu m$ is $f_{11}(x_1) = -0.0004\ {x_1}^2 + 0.0333\ x_1 + 0.6$. For designer s_1, the single attribute utility for tensile strength varies from 60 MPa to 75 MPa and the function is given by $f_{12}(x_2) = -0.001\ {x_2}^2 + 0.010\ x_2 + 1.200$. Assuming additive independence of the attributes, multi-attribute utility function of designer s_1 is $u(X) = 0.2f_{11}(x_1) + 0.8f_{12}(x_2)$. Manufacturer p_2 offers resolution between 50 μm and 300 μm and tensile strength between 22 MPa and 41 MPa with uniform probability. Using these function values the expected utility gained by designer s_1 being matched to manufacturer p_2 was calculated as $E[u_{12}(X)] = 0.353$. Single attribute utility functions of manufacturer p_2 were $g_{23}(x_3) = -0.000\ {x_3}^2 + 0.0024\ x_3 + 0.1975$, $g_{21}(x_1) = -0.000\ {x_1}^2 - 0.0009\ x_1 + 1.052$ and $g_{24}(x_4) = -0.0327\ {x_4}^2 + 0.4327\ x_2 - 0.2082$. Again assuming additive independence, multi-attribute utility function for manufacturer p_2 is defined as $u(X) = 0.375g_{23}(x_3) + 0.375g_{21}(x_1) + 0.25g_{24}(x_4)$. Using these functions the expected utility gained by manufacturer p_2 being matched to s_1 was calculated as $E[u_{21}(X)] = 0.071$. Similarly, expected utilities of each agent are calculated for each of their potential matches. Some of the matches are incompatible, e.g., the design may not fit in the build envelope of the machine. In such cases, the utility of matching is zero. Table 2.2 shows the results of expected utilities calculated for each agent for each alternative.

Based on the expected utilities the designers are matched to manufacturers using the Munkres, NRMP, TTC, and FCFS mechanisms. The total expected utility gained, and the average rank of the designers and manufacturers are then calculated for each mechanism. The utility and rank are dependent on the preferences of the agents. In FCFS these may also depend on the order of arrival of the service requests. All the illustrative numerical values listed in Tables 2.1 and 2.2 are for a particular preference distribution and the order of arrival of the service requests. Thus, there is variability associated with the performance of the mechanism. To account for the variability, several combinations of preference distributions were used, and the results were analyzed statistically. Urgency, resolution and material requirement of each designer and manufacturer and the relative weightage of these attributes were randomized for each run. For each random preference distribution so generated,

Table 2.2.
Expected utility achieved by each agent for each of their alternatives. Numerical value tabulated in row i and column j indicates the expected utility achieved by service seeker s_i being matched to service provider p_j and the entry in bracket indicates the expected utility achieved by p_j in return. '-' represents that the match is not feasible.

Manufacturer / Designer	(p_1)	(p_2)	(p_3)	(p_4)	(p_5)
(s_1)	0.355 (0.196)	0.353 (0.071)	0.355 (0.313)	0.000 (0.191)	0.004 (0.527)
(s_2)	0.000 (0.001)	0.047 (0.001)	0.043 (0.001)	-	-
(s_3)	0.000 (0.001)	0.004 (0.002)	0.000 (0.001)	-	-
(s_4)	0.019 (0.000)	0.004 (0.001)	0.000 (0.000)	0.349 (0.088)	0.000 (0.056)
(s_5)	0.000 (0.000)	0.022 (0.001)	0.017 (0.000)	0.470 (0.088)	0.003 (0.000)
(s_6)	0.379 (0.196)	0.377 (0.071)	0.379 (0.313)	0.000 (0.197)	0.004 (0.527)
(s_7)	0.355 (0.039)	0.453 (0.021)	0.326 (0.063)	0.148 (0.11)	0.129 (0.577)
(s_8)	0.233 (0.039)	0.324 (0.021)	0.233 (0.063)	0.094 (0.108)	0.101 (0.576)
(s_9)	0.012 (0.000)	0.005 (0.142)	0.000 (0.013)	0.430 (0.091)	0.000 (0.001)
(s_{10})	0.307 (0.196)	0.283 (0.072)	0.284 (0.314)	-	0.000 (0.542)
(S11)	0.612 (0.040)	0.875 (0.021)	0.637 (0.063)	0.246 (0.191)	0.247 (0.577)
(s_{12})	0.326 (0.097)	0.513 (0.021)	0.386 (0.063)	0.131 (0.315	0.140 (0.520)
(s_{13})	0.490 (0.040)	0.680 (0.021)	0.490 (0.063)	0.197 (0.256)	0.199 (0.521)
(s_{14})	0.431 (0.039)	0.567 (0.021)	0.408 (0.063)	0.178 (0.130)	0.161 (0.520)
(s_{15})	0.016 (0.000)	0.005 (0.001)	0.000 (0.000)	0.385 (0.091)	0.000 (0.001)
(s_{16})	0.000 (0.000)	0.004 (0.001)	0.000 (0.000)	0.282 (0.088)	0.006 (0.000)
(s_{17})	-	0.053 (0.001)	-	-	0.009 (0.001)
(s_{18})	0.031 (0.072)	0.002 (0.170)	0.000 (0.020)	-	0.000 (0.016)
(s_{19})	0.008 (0.000)	0.006 (0.050)	0.000 (0.031)	-	0.000 (0.057)
(s_{20})	-	0.629 (0.021)	0.453 (0.063)	-	0.179 (0.577)
(s_{21})	0.000 (0.000)	0.004 (0.001)	0.000 (0.000)	-	0.006 (0.001)
(s_{22})	0.227 (0.196)	-	0.287 (0.313)	-	0.011 (0.542)
(s_{23})	0.355 (0.196)	0.353 (0.072)	0.355 (0.314)	-	0.004 (0.527)
(s_{24})	-	-	0.490 (0.063)	-	0.199 (0.520)
(s_{25})	-	-	0.350 (0.063)	-	-

all the four matching mechanisms were implemented and the overall efficiency was compared.

2.4.2 Influence of Resource Availability

Figures 2.4 through 2.7 show a comparison of the total utility and average rank by each of the mechanisms under various resource conditions. All the mechanisms were run for 100 randomly generated preference ordering for each of resource scarce, resource balance and resource surplus setting. The boxplots of the total utility attained by designers and manufactures under various resource conditions are shown in Figures 2.4 through 2.6. While Figures 2.4 through 2.6 show total utility, Figure 2.7 shows the average rank attained by designers and manfacturers under resource balanced conditions.

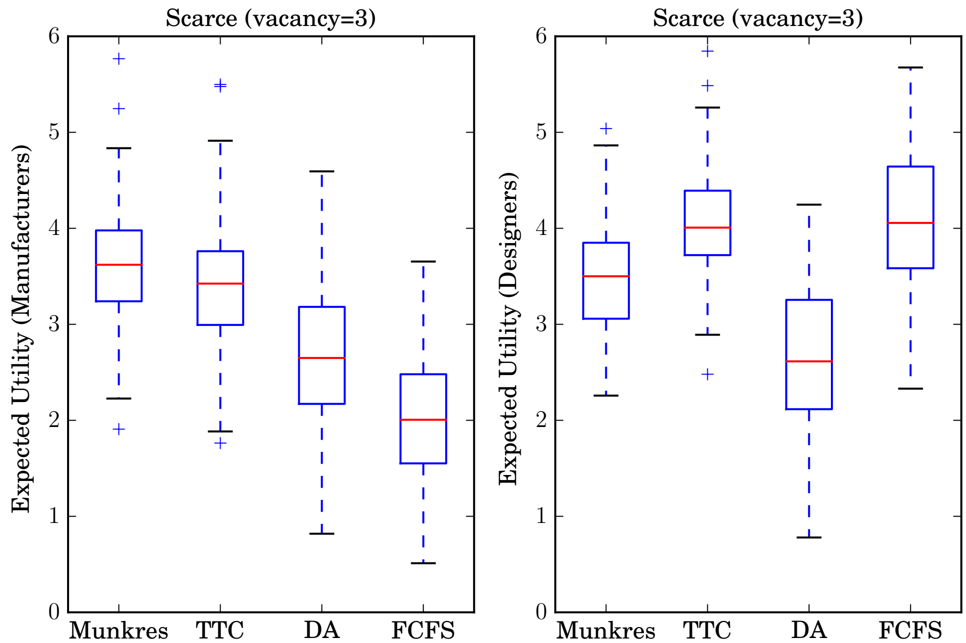


Figure 2.4. Total expected utility attained by manufacturers and designers in resource scarce case.

Comparing DA and TTC in Figure 2.4, it is observed that the total manufacturer utility gained by TTC is more than in DA even though a PODA (manufacturer optimal) mechanism is implemented. This is because PODA guarantees stability and can operate only in the stable space of solutions. But as the resource availability increases from scarce to balanced and abundant, the performance of TTC and DA is nearly similar in terms of the manufacturers' utilities. This is because the set of stable solutions grows rapidly as the resources get abundant and the formation of weak and strong cycles [3] decreases and efficiency of DA matches that of TTC with respect to manufacturer utility. Thus, in fully decentralized market when demand and supply of services and resources are balanced, DA performs as well as TTC for manufacturers even though it operates in the space of stable solutions. In addition DA mechanism offers stable solutions and hence a mechanism based on DA is the best mechanism in

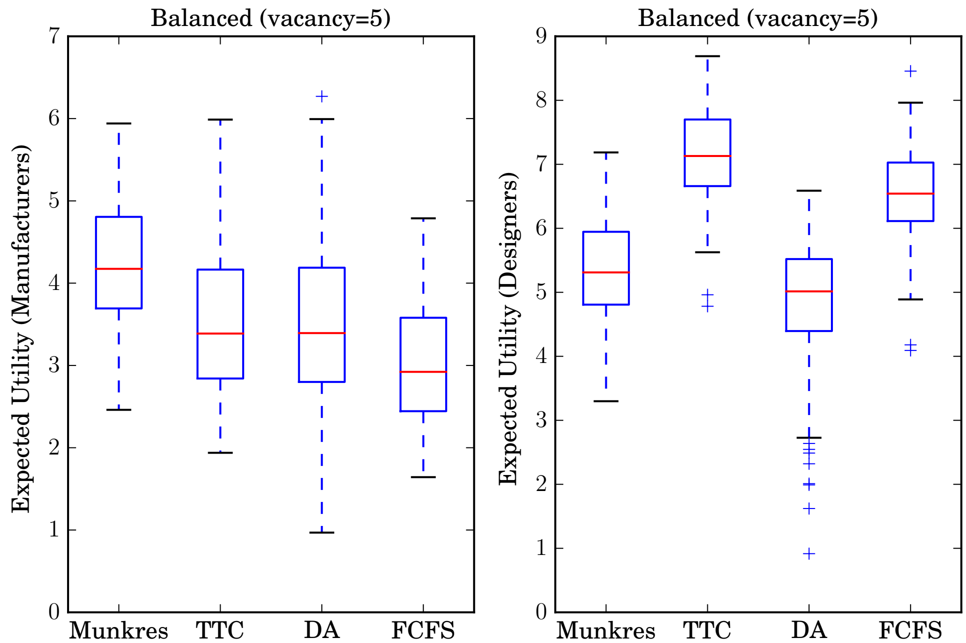


Figure 2.5. Total expected utility attained by manufacturers and designers in resource balanced case.

a totally decentralized CBDM market. The total expected utility of the designers by implementing TTC or DA increases from resource scarce to resource balanced, but after that it depends on the set of feasible matchings.

In the case of FCFS, the designer utility grows from resource scarce to resource surplus as the availability of most preferred manufacturers increases with resource availability. The average designer rank (see Figure 2.7) of FCFS on the other hand is better than all other mechanisms. However, this average is only calculated over the completed matches. In FCFS the number of completed matchings is arbitrary even for a given instantiation depending on the order of arrival of service requests. Additionally, FCFS does not have any of the useful properties listed in Table 3.2 Therefore FCFS is not a preferred mechanism in any of the three scenarios discussed.

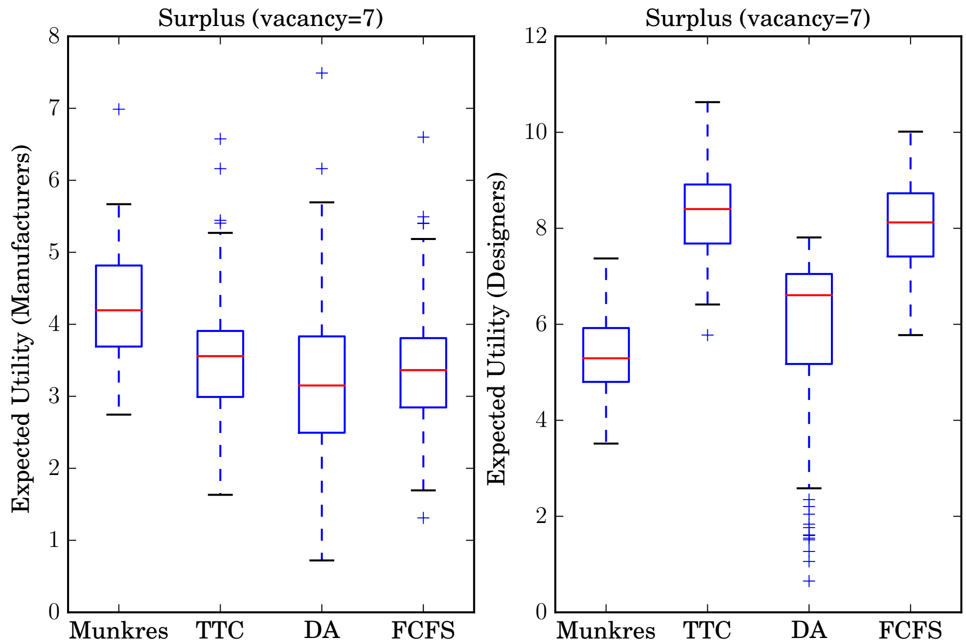


Figure 2.6. Total expected utility attained by manufacturers and designers in resource surplus case.

Munkres offers the highest manufacturer utility regardless of scarcity of resources. But the average rank attained by the manufacturers is better in DA and TTC as compared to Munkres. This is because DA and TTC are based on the ordinal preference ordering whereas Munkres is based on total cardinal utility attained by the entire set of agents. The difference between Munkres and TTC, DA is clear particularly in scarce resource settings.

2.5 Conclusions

The performance of the mechanisms depends on the availability of resources which in turn is based on the market thickness. However, FCFS is not a preferred mechanism in any of the three resource scenarios. Total manufacturer utility gained by TTC is

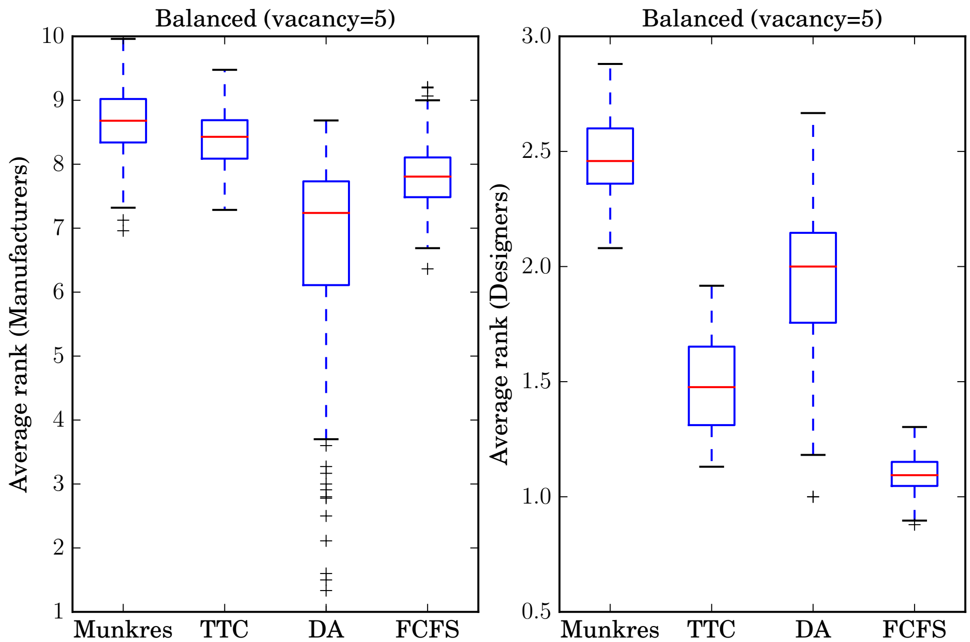


Figure 2.7. Average rank over completed matches attained by manufacturers and designers in resource balanced case.

more than PODA (manufacturer optimal DA) as PODA operates only within the stable space of solutions limiting its efficiency. This effect is more pronounced in resource scarce setting. Once resources become more abundant, the set of stable matching options increases thereby reducing the gap in efficiency between TTC and DA. Munkres offer higher manufacturer utility than DA and TTC in all three resource settings. However, the average rank attained by manufacturers is better in DA and TTC as compared to Munkres.

However, in this chapter the efficiency was quantified based on only the utility and rank attained by the matched outcomes. In practice, several other properties become crucial depending on the application. In Chapter 3 we analyze the properties of DA, TTC, and Munkres mechanism in the context of three specific CBDM scenario. Based on the analysis of properties and based on the insights drawn from the resource simulations shown in this chapter, we recommend best mechanisms to be implemented in those scenarios.

3. BEST MATCHING MECHANISM IN CBDM SCENARIO

In this chapter the characteristics of three different CBDM scenarios are studied and matching mechanism are recommended for the three scenarios. The CBDM scenarios studied have unique characteristics and generalizes to a wide range of applications. The properties of the mechanisms are analyzed in the context of the objectives of the scenario. Based on how well the properties of the mechanism suit the requirements and objectives of the scenario, recommendations are made for the best mechanisms. This chapter is structured as follows: Section 3.1 discusses some of the desirable properties of matching mechanisms in CBDM environment, Section 3.2 describes the characteristics of three different CBDM scenarios, Section 3.3 formalizes the properties of suitable matching mechanism, Section 3.4 proposes ideal matching mechanism in each of three CBDM scenarios based on its properties and the resource simulation results, and Section 3.5 discusses closing comments and research gaps for future work.

3.1 Desirable Properties of Optimal Bipartite Matching in CBDM

In the last Chapter, in Section 2.3 the steps involved in matching service seekers (designers) to service providers (manufacturers) were discussed. Based on their preferences, machine owners can rank order the designers; and designers can rank order the machine owners. Each designer would like to be matched to a machine owner who is at the top of their rank ordered list. Similarly, each machine owner would like to be matched to the top designer in his/her list. The match is easy if each machine owner is at the top of only one designer's rank ordered list, and each designer is at the top of only one machine owner's list. However, this is a very restricting case, and is generally not true. In real scenarios, many designers may be interested in using the same 3D printer. Therefore, it is rarely possible for everyone to achieve their first preference. Given that some of the participants will not achieve their first preference, the goal is to match designers and machine owners in a way that is optimal, in some sense, for the market as a whole.

Since there are multiple decision makers with different objectives, optimality can be defined in many ways. One possible definition is based on the maximization of total utility achieved by a set of agents (e.g., the set of all service providers). Such an optimal matching can be achieved by generalized assignment algorithms, such as the Munkres algorithm [37]. Such algorithms are appropriate only if the set of agents belongs to the same organization, and have a collective preference for the entire set. The algorithm fails when agents in the set are independent and the utilities of both the service seekers and the providers need to be considered.

In addition to optimality, the match should also have a number of other desirable properties. First, the matches should be compatible with the preference structures. For example, there should not be any designer-manufacturer pair, who prefers to be matched with each other, but are not matched by the CBDM system. If there is such a pair, then that pair has an incentive to collude outside the CBDM platform. Second, participants can dynamically enter or exit the system in a decentralized environment such as CBDM. Therefore, the matching mechanism should be insensitive to participants entering and leaving the system. For example, if a matched designer-manufacturer pair leaves the system, there should not be any change in the matches for the rest of the participants. Third, addition of service providers should only help the service seekers. Finally, the system should prevent "gaming", i.e., misrepresentation of information by individuals (or group of) participants. A matching mechanism should avoid strategic behavior, which results in the stated preferences being different from the true preferences. The matching mechanism should enable participants to base their preference ordering solely on their true preferences. This is called the *truthful revelation* property of the mechanism.

These desirable properties are formally described and quantified in Section 3.3.1. No matching algorithm can satisfy all these properties. Based on the scenario in question, certain properties may be more important than others. For example, the likelihood of different types of gaming is different depending on the CBDM platform that the agents are operating on. Therefore, the goal is to find the best possible matching mechanism depending on the scenario. To achieve this, the first step is to analyze the possible scenarios. Three broad scenarios are discussed in Section 3.2.

3.2 Typical scenario in CBDM

Consider three representative scenarios covering a broad range of applications: a) fully decentralized scenario, b) monopolistic scenario, and c) organizational scenario. The strategic characteristics of the agents on both sides, service seekers (S) and service providers (P), are compared for the three scenarios in Table 3.1.

Table 3.1.
Comparison of strategic behavior of agents in the three scenarios.

Scenario	**Set Size**	**Service Seekers (S)**	**Service Providers (P)**	**Coalitions**
Fully decentralized	$\|S\| > 1, \|P\| > 1$	Strategize	Strategize	Unlikely
Monopolistic	$\|S\| > 1, \|P\| = 1$	Strategize	Not Strategize	No coalition
Organizational	$\|S\| > 1, \|P\| = 1$	Strategize	Not Strategize	Likely

Fully Decentralized Scenario

This is a completely decentralized market scenario where independent service seekers avail services from independent service providers. The service seekers and providers will be collectively referred to as agents. The service seekers are designers who do not possess necessary resources to make physical prototypes of their designs, and service providers are companies or individuals owning resources such as CNC machines or 3D printers. It is assumed that each designer and manufacturer is an independent strategic entity aiming to maximize his/her own payoff. It is also assumed that each agent can engage in strategic behavior such as revealing preferences that are different from their true preferences.

The agents may exhibit the following types of strategic behavior: (i) a service seeker and a provider may sign contracts outside the platform, (ii) a service provider may manipulate the system by providing false information about their capacity, and (iii) service seekers may submit manipulated preference characteristics to increase the probability of matching with the desired or most sought after service providers. The objectives of the matching algorithm in the fully decentralized scenario are: it should be immune to the strategic behavior of agents, it should optimize the utility of both service seeker and provider, the optimal matching should remain the same even if some agents leave the system once matching is done.

Monopolistic scenario

In this scenario, a single organization (e.g., Shapeways [10]) owns a wide variety of resources and independent designers avail services from this organization. Designers submit their design requirements to these companies and get them printed or manufactured. In this scenario, the service provider is a single agent or company, which is also responsible for matching service seekers to the resources. Hence, the organization does not exhibit strategic behavior. Unlike in the totally decentralized scenario, here both the resource provider and algorithm implementer is the same agent whose sole objective is to maximize its utility by matching seekers to the resources.

The objective of an appropriate matching mechanism is only to maximize the payoff of the organization. The utility gained by the service seekers is not considered here. The preferences of the service seekers are indirectly accounted for through customer satisfaction, which is generally a part of the service-provider's preferences.

Organizational scenario

This scenario is different in the sense that both service seekers and service providers belong to the same organization. All the service providers are owned by the same organization and hence are non-strategic in nature. This scenario can represent students of a university availing 3D printing resources of the university to print their design projects, or designers in a R&D company printing prototypes to validate their designs.

Service seekers who belong to the same company have a greater incentive to strategize as the information and strategies about service seekers and service providers are readily available. The probability of strategic coalitions is higher in this scenario as the agents belong to the same organization. An appropriate matching mechanism should be strategy-proof to coalition formation. It should also consider the utility functions of both the service seekers and service providers.

3.3 Evaluation of Matching Mechanisms for CBDM

In this section we describes the properties that we use to compare the mechanisms against each other. Section 3.3.1 define the properties and describes the meaning of those properties in CBDM application. In Section 3.3.2 we describe how we compare the mechanisms using these properties.

3.3.1 Criteria for Evaluation

This section describes the set of criteria relevant to a diverse set of matching scenarios in CBDM framework. The important criteria include (i) individual rationality [40], (ii) stability [32], (iii) consistency [41], (iv) resource monotonicity [42], (v) population monotonicity [43], (vi) strategy proofness [30], (vii) group strategy proofness [30], (viii) Pareto efficiency [40], (ix) absolute majority [44], and (x) effective cardinal efficiency [45].

Individual Rationality: A mechanism μ is individually rational if (i) every agent matched through μ prefers its matched partner over being unmatched, and (ii) there is no under-utilized service provider who is more preferred by any service seeker than its match. Here, an under-utilized service provider p_j is one who is matched to a lower number of service seekers than its capacity q_{p_j}. Mathematically, individual rationality can be written as:

(i) $\forall s_i \in S$, $\mu(s_i) \succeq_{s_i} s_i$

(ii) $\nexists s_i$, p_j such that $p_j \succ_{s_i} \mu(s_i)$ and $q_{p_j} > \mu^{-1}(p_j)$

Practical examples where individual rationality gets violated are: (a) a company, outsourcing a manufacturing job to the cloud to save time and cost incurred to the company, matched to a service provider who demands a higher price than carrying out the job in-house; (b) a service seeker's requirements exceed the capacities of a service provider being matched to (e.g., design volume exceeding the build volume of the 3D-printer owned by the service provider). In all these examples the agent is better off remaining unmatched. Hence, the matching mechanism should eliminate those match combinations where individual rationality assumption breaks.

Stability: The mechanism is said to be pairwise stable if it is individually rational and has no blocking pair. Service seeker s_i and service provider p_j are said to be a blocking pair in μ if

(i) a different service seeker s_k is assigned to p_j under the mechanism μ, i.e., $\mu(s_k) = p_j$,

(ii) s_i strictly prefers p_j over its assignment, i.e., $p_j \succ_{s_i} \mu(s_i)$, and

(iii) reciprocally, p_j strictly prefers s_i over its current assignment s_k, i.e., $s_i \succ_{p_j} s_k$

If a mechanism is not stable then there is an incentive for the blocking designer-manufacturer pair to collude outside the CBDM application platform affecting the efficiency of the matching process.

Consistency: The mechanism is consistent if the optimal allocation remains the same even if some agents leave along with their matched pairs. For a consistent mechanism μ, if

(i) $\phi \subset S' \subseteq S$ and $\phi \subset P' \subseteq P$, and

(ii) $\mu : S \to S \cup P$ and $\mu' : S' \to P' \cup S'$

then $\mu(s_i) = \mu(s_i')$ for all $s_i \in S$ and $s_i' \in S'$; where S' and P' are the set of designers and manufacturers who leave the matching mechanism after matching is performed. CBDM is a dynamic system with large number of agents entering and leaving the system to seek and provide services. In such a setting, it cannot be guaranteed that the agents would accept the assignment generated by the platform. Consistency property ensures that the efficiency of a matching mechanism is not lost due to dynamic entry and exit of agents.

Monotonicity: The mechanism is called monotone in a bilateral matching if the welfare of each agent on one side increases (decreases) by addition (removal) of agents from the other side. If agents on the service provider side are the ones being added or removed and the welfare achieved by the each service seeker either strictly increases or decreases, then we call the matching mechanism resource monotone. Mathematically, if P' is a set of service providers different[1] from set P, with either $P' \subseteq P$ (some service providers left from original set P) or $P \subseteq P'$ (new service providers joined the original set P), then either $\mu_{R_S \otimes R_P}(s_i) \succeq \mu_{R_S \otimes R_{P'}}$ (welfare of all service seekers decrease from P to P') or $\mu_{R_S \otimes R_P}(s_i) \preceq \mu_{R_S \otimes R_{P'}}(s_i)$ (welfare increase from P to P') for each $s_i \in S$. Similarly the mechanism is called population monotone if the welfare of each resource provider is increased (decreased) by the addition (removal) of service seekers.

For example, consider matching in '3DHubs' [7]. Addition of a new machine into the mechanism should only help the designers in getting a higher ranked match,

[1]it can also be the same set of service providers offering a reduced or increased cap on vacancy i.e., $q'_{p_i} \neq q^{p_i}$

and addition of a new designer should only increase the number of suitable matches available to the service provider. Monotonicity ensures that this property holds true for matching. Monotonicity also provides the added advantage that welfare of each service seeker increases by adding a new machine, and not just the total welfare of all service seekers.

Strategy-proof: Mechanism μ is strategy-proof if no single agent is better off misrepresenting the preferences. Agent s_i would exhibit strategic behavior if $\mu_{R_{s_{-i}} \otimes R'_{s_i}}(s_i) \succ_{s_i} \mu_{R_{s_{-i}} \otimes R_{s_i}}(s_i)$, where R is the real preference structure and R' is the misrepresented preference structure. If the rule is immune to such behavior then it is strategy-proof. Consider the case where agents repeatedly outsource their needs to the CBDM framework. Over time the agents may learn how the matching takes place and the matching mechanism is susceptible to loss in efficiency due to strategic behavior from the agents. For example, in FCFS a designer can increase the probability of being matched to a highly sought after machine owner by switching the first and second choice. The machine owners could misrepresent the capacity of machines to increase the probability of getting matched. Hence, there is a need to make the platform immune to such strategies.

Group Strategy-proof: The mechanism is group strategy-proof if even a coalition of agents is not better off misrepresenting the preference ordering of all the individuals in the coalition. Agents in a set $S'(\subseteq S)$ have an incentive to form coalition and falsify their preference ordering in a rule μ if (i) all agents in set S' do not decrease their welfare by colluding, and (ii) at least one agent strictly increases its welfare.

(i) $\mu_{R_{S/S'} \otimes R'_{S'}}(s_i) \succeq_{s_i} \mu_{R_S}(s_i) \, \forall s_i \in S'$ and

(ii) $\mu_{R_{S/S'} \otimes R'_{S'}}(s_i) \succ_{s_i} \mu_{R_S}(s_i)$ for some $s_j \in S'$

If agents who participate in the matching know each other well then the mechanism would be susceptible to coalition strategies [3]. In the scenarios discussed this gaming behavior is probable in the organizational scenario where the agents involved in the matching process are co-workers.

Pareto-efficiency: A mechanism μ is Pareto-efficient with respect to a set of agents if there is no other mechanism μ' that strictly increases the utility of a subset of agents keeping the utility of the rest of the agents the same. Rule μ is Pareto efficient with respect to set S if $\forall R \nexists \mu'$ such that

(i) $\mu'(s_i) R_{s_i} \mu_{s_i}$; $\forall s_i \in S$, and

(ii) $\exists s_k \in S$ with $\mu^{'}(s_k) P_{s_k} \mu_{s_k}$

Similarly, the definition can be extended to Pareto efficiency with agents of set P. Mechanism μ is Pareto efficient if it is efficient with respect to both S and P.

Absolute majority: A mechanism is said to satisfy the absolute majority property if it maximizes the number of agents who are matched to their first choice in their submitted preference ordering. In many cases, the designers would rate their first desired manufacturer much above the subsequent ones; they would be indifferent between second and third choices. Hence, it is always desirable to have a mechanism that matches the maximum number of agents to their first choice.

Effective cardinal efficiency: For sets S and P, effective cardinal efficiency is the total expected utility achieved by each agent in sets S and M respectively.

3.3.2 Using the Criteria to Compare the Mechanisms for the CBDM problem

Table 3.2 shows competing properties while implementing mechanism design for decentralized design and manufacturing in a strategic setting. There is no universal mechanism that satisfies all these properties. For example, no mechanism is both stable and efficient [40]. However, depending on the scenario, the mechanism that has the properties most relevant to the scenario in concern can be chosen.

DA mechanism can be implemented in two ways, based on the set of agents that proposes [3]. DA mechanism has the property that it is optimal with respect to the agents who propose during its implementation. Thus, if the service providers

Table 3.2.
Comparison of mechanisms in terms of its properties.

Criterion	DA	TTC	Munkers
Individual rationality	✓	✓	✓
Stable	✓	if strongly acyclic	✗
Resource monotone	if weakly acyclic	if strongly acyclic	✗
Population monotonic	if weakly acyclic	if strongly acyclic	✗
Consistency	if weakly acyclic	if strongly acyclic	✗
Strategy-proof	✓	✓	✓
Group-strategy proof	if weakly acyclic	✓	✗
Pareto efficiency	if weakly acyclic	✓	✓
Absolute majority	✗	✗	✗
Effective cardinal efficiency	✗	✗	✓

in P propose, then it is P-optimal, whereas if the seekers propose then it is S-optimal. The former provides the optimal stable matching for agents of set P whereas the latter results in optimal matches for agents in set S. In the rest of the paper, these mechanisms are labeled PODA (Provider Optimal DA) and SODA (Seeker Optimal DA) respectively. In addition to being P-optimal, PODA is strategy-proof with respect to agents in P however agents in S can strategize. Similarly, SODA is strategy-proof only with respect to agents in S. Thus PODA is not group-strategy proof with agents in S. Kojima [46] shows that PODA is not group-strategy proof if there is at least one $p_i \in P$ with $q_{p_i} > 1$. In such cases one can generate scenarios where agents in P can manipulate via misreporting vacancies or being involved in prearranged matches outside the application platform.

DA always produces stable matching. It is resource monotonic and population monotonic but not always consistent. Although DA Pareto dominates any other stable allocation, it is not optimal. Also, DA is strategy proof but not group-strategy proof. Whereas TTC is efficient and group-strategy proof but lacks consistency and is not resource and population monotone.

Unlike TTC and DA which only account for the ordinal ranking of the preference ordering, Munkres mechanism maximizes the cardinal value of total expected utility for a set. Hence, when the aim is to maximize expected utility of non-strategic agents belonging to the same set (either S or P) then Munkres is the best choice. For example, organizations such as Shapeways can adopt Munkres mechanism to match the uploaded designs to the right machines. But it breaks down in 3Dhubs where the individuals agents in either set are utility maximizing strategic agents. Moreover, Munkres mechanism is not population monotone, resource monotone, consistent, stable and strategy proof. In a bipartite matching situation the utility of the two sets of agents needs to be maximized. Munkres can optimize only a single set at a time.

3.4 Evaluation of Matching Mechanisms for the Three Scenarios

Based on critical analysis of the properties of the mechanisms and their performance under various resource conditions the best mechanisms for the three scenarios are listed in Table 3.3. For the *monopolistic scenario*, the best mechanism is the one based on the Munkres mechanism. From Figures 2.4 to 2.6, it is observed that total expected utility attained by set of manufacturers is maximum for the Munkres mechanism, irrespective of availability of resources. In monopolistic scenario the company who owns all the resources is in charge of carrying out the matching process. Monopolist will try to maximize its utility and hence would resort to the Munkres mechanism.

In *two-sided strategic scenario* DA is the most suited mechanism in the case of totally decentralized scenario despite the lower efficiency compared to TTC. This is because it offers useful properties such as consistency, resource and population monotonicity, and stability. Balinski and Sonmez [40] show that if DA is not consistent for a problem, then TTC is not consistent for it either; the converse is not true. Similarly if DA fails to be stable, TTC also fails; but the converse is not true. Resource monotone and population monotone means even if some service providers or service seekers (or both) leave after the final matching allotments are declared, the solution for the reduced set of agents remains the same. This is highly possible in totally decentral-

Table 3.3.
Relevant properties and best mechanisms for the three scenarios.

Scenario	**Relevant properties**	**Best mechanism**
Totally decentralized	Strategy proof, stable, Resource monotonic, Population monotonic, consistent, individually rational, bilateral utility	PODA
Monopolistic	efficiency, cardinally optimal	Munkres
Organizational	group-strategy proof, bilateral utility, efficiency	Top-Trading Cycle

ized scenario with independent strategic agent being the stakeholders. Consistency of the DA mechanism ensures that the most optimal matching remains unchanged even if some agents leave the system. Resource and population monotone nature of DA ensures fairness even if some service seekers or providers leave. Finally, there is evidence from real world applications [16] which shows that stable mechanisms often succeed over unstable ones. Balinski and Sonmez [40] highlight this by stating that if one needs to consider both stable and optimal solutions then DA ranks ahead of TTC.

Now that we have rated DA over TTC in this scenario, still we need to choose among the two types of DA (i.e., SODA and PODA). SODA is strategy-proof with respect to agents in S, i.e., the designers, while PODA is strategy proof with respect to manufacturers only. Manufacturers have a higher probability to behave strategically because unlike designers, manufacturers can strategize both by capacity manipulation and outside settlement. Additionally, manufacturers are repeatedly involved and have more learning effect than designers. Hence we propose the use of PODA over SODA to make it strategy-proof with respect to the manufacturers. Further, we rule out FCFS and Munkres in this scenario as it takes care of only one set of agents and do not offer properties such as stability and consistency, which are important in a dynamic environment.

In an *organizational scenario*, we need to enforce a canonical response[2] unlike totally decentralized scenario. For example, students using 3D-printing resources in a university can group-strategize to gain unfair advantage. Therefore, in addition to optimizing the preferences of both sets of agents, the mechanism needs to be group-strategy proof as the system is more prone to coalition strategies. Therefore, the mechanism based on TTC is best suited in this scenario.

3.5 Closing Comments

There is no single best matching mechanism for all scenarios. Depending on the strategic behavior and resource availability the right mechanism should be implemented. If the objective is to maximize the utility of a single monopolistic resource provider then Munkres is the best mechanism. When the preference ordering of both service seekers and providers need to be considered TTC and DA perform better. In a totally decentralized scenario, where stable solutions are important, DA is the best mechanism. In an organizational scenario such as students using 3D printing resources in a university, TTC is the best mechanism. Table 3.3 summarizes the general approach and mechanism for the three scenarios considered in this paper. The performance of the mechanisms also depends on the availability of resources which in turn is based on the market thickness. Hence, the insights drawn from the simulations can also be used to choose appropriate mechanism based on the frequency of matching and the type of the scenario.

There are several mechanism design related issues specific to CBDM for further investigation. First, the present analysis is based on the assumption that agents are substitutes and not complements. But CBDM is about integrating collective resources to get the manufacturing task done. Thus, manufacturing resources may

[2]the response depends only on the individual's preference characteristics and rules of the mechanism and is not influenced by strategies and preferences of other agents.

become complements depending on the needs of the service seeker. Second, resource discovery is a challenging task in CBDM. It may be impossible for all agents to provide an exhaustive list of their alternatives. The agents are better off revealing the preference characteristics towards attributes. There is a need to design strategy proof mechanisms when agents reveal attribute preferences instead of alternatives. This is because many properties of the mechanisms change when preferences are revealed as attributes and not as alternatives. Third, CBDM involves exchange of transferable utility such as money. This assumption has been relaxed in the analysis. Fourth, the resources may not be perfectly divisible. For example, in the NRMP application, when medical residents are matched to hospitals there is a fixed vacancy to be filled. But in CBDM a few resource owners may be willing to manufacture the product for pre-determined amount of time and for the available time the resource can be divided among multiple service seekers. Thus, the matching problem in CBDM is unique as compared to the previous applications of the matching mechanisms. These differences bring in new challenges. As an example, when resources are not perfect substitutes there will not exist any stable matching.

4. OPTIMAL MATCHING FREQUENCY OF MULTI-PERIOD MATCHING MECHANISMS

Matching markets such as CBDM in addition to being two-sided are also *real-time* in nature as there is a continuous stream of service providers and service seekers entering and exiting the system. Service providers such as machine owners enter the system when they have extra capacity to receive new orders and exit the system as they are assigned new service orders and re-enter the system after they process the assigned service request. New designers (or service seekers) who needs to get their parts manufactured come and exit out of the system once they get matched or if they do not find any acceptable manufacturers in the system.

A central issue in resource allocation in such *real-time* markets where agents interact over a long period of time is scheduling of the allocation mechanism. In Chapter 3 we proposed DA as the most optimal mechanism in two-sided total decentralized market settings. Now, if DA matching mechanism is employed for resource allocation it needs to be implemented repeatedly over multiple matching cycles. There is a need to tune the frequency at which the matching mechanism is implemented so as to optimize various matching objectives such as utility attained by the participating agents, fairness in their distribution, and number of successful matches.

The interval at which the matching mechanism is implemented is called matching period. It is the length of the matching cycle. Higher the matching period lower the frequency of implementation of matching mechanism. In this chapter, we aim to study the design of optimal period of matching mechanism.

4.1 Literature Review and Research Gap

Existing literature provides mechanisms that satisfy useful properties such as stability in a single matching cycle, but they lack studies on the effect of the period of matching cycle on the optimality.

4.1.1 Deficiency of conventional resource allocation mechanisms

Conventional resource allocation methods are inappropriate for matching resources to service seekers in decentralized scenarios because the designer of the algorithm makes implicit assumptions that the participating agents will act as instructed. With the emergence of distributed and cloud-based manufacturing with independent resource providers, this assumption is no longer valid. It is more reasonable to assume that each participating agent will manipulate preferences for selfish gains at the cost of efficiency of the mechanism. The field of 'mechanism design'studies on designing algorithms where agents act rationally. The misrepresentation of information by individuals (or group of) is called 'strategic behavior'and the mechanisms that penalize such behavior are said to be 'strategic-proof' [30]. Efficiency loss because of strategic behavior need not be small and the cost of not having a strategy-proof mechanism is much harder to measure.

In addition to strategy-proofness, other useful properties of mechanisms include stability [32], individual rationality [40], consistency [41], monotonicity with respect to demand [43], and supply [42]. There is no mechanism that satisfies all these properties [40]. Therefore, mechanism must be specifically designed for each application. In an earlier work, the authors, modeled the problem of resource allocation in CBDM as a bipartite matching problem [47] and proposed best matching mechanisms [48] among existing mechanisms in different CBDM scenarios by analyzing their properties in the context of requirements in different CBDM scenarios. However, in a stochastic environment like CBDM where the arrival of service seekers and availability of service providers is a continuous process there are no studies on the optimal frequency of implementation of such mechanisms. Therefore, additional advances need to be made on existing mechanism design literature to develop optimal matching mechanism suited for the requirements of CBDM.

4.1.2 Why not job scheduling algorithms?

Job scheduling literatures studies the timing of jobs to optimize objectives such as total cost, total utility, and total project tim while allocating resources in a stochastic environment. For example, Smith [12] proposed a job scheduling algorithm for a single machine. However, they assumed that the mechanism is centralized with complete information on all the jobs, their significance, and processing time. Anderson and Potts [13] extended it to a scenario where the algorithm does not have access to complete knowledge of all the jobs. In all the above job scheduling algorithms, the designer of the algorithm makes implicit assumptions that the participating agents will act as instructed. Nisan and Ronen [14] proposed a job-scheduling algorithm that accounts for the strategic behavior of the participants. Heydenreich et al. [15] extended this idea to a strategic setting where the participants may manipulate the job processing time, arrival time of job, and the cost of waiting time. However the resulting mechanism is not decentralized and the equilibrium of the game is a myopic best response based (a weaker condition than Dominant strategy equilibrium). Christodoulou et al. [49] extended the LP-relaxation job scheduling problem into a mechanism design framework to also account for the strategic behavior of the participants. Jain et al. [50] developed an algorithm for allocating jobs in the cloud which is truthful-in-expectation which is primarily suited for cloud computing applications. Andelman et al. [51] showed a Fully Polynomial Time Approximation Scheme algorithm for scheduling jobs on a fixed number of machines that elicit truthful revelation with the goal of minimizing overall completion time.

In all of the job-scheduling algorithms the focus is only on optimizing some global objective function such as overall completion time or cost and they ignore individual objectives of the independent agents. Moreover, they do not have useful properties such as stability, individual rationality, and consistency.

4.1.3 Research gap

In job-scheduling literature, the focus is only on optimizing some global objective function while ignoring important properties such as stability, individual rationality, and consistency. The field of mechanism design focuses on designing mechanisms that have useful properties, but they lack studies on the optimal frequency of implementation of such mechanisms. In situations where agents repeatedly interact with one another, manipulations and strategic behavior are much more probable because of the knowledge about historic data. Therefore, there is a need to identify optimal frequency of implementation of mechanisms so that they satisfy useful properties and produces optimal matches when implemented in situations where agents repeatedly interact with one another over long periods of time.

To address this gap the central research question in this chapter is: *what is the optimal period of matching for a given service arrival rates considering matching objectives such as average utility attained, number of successful matches and fairness in the distribution of utility?* We use simulation studies on a synthetic CBDM scenario to identify the optimal period of matching for various arrival rate of designers and perform Sobol sensitivity index [52] to study the robustness of the design period to the variabilities in the seeker arrival rate and availability of providers.

4.2 Modeling Resource Allocation in CBDM as a Stochastic Matching Problem

We model CBDM as a resource allocation problem, where service seekers (S) avail manufacturing resources from service providers (P). The sets of service seekers, and service providers are denoted by $S = \{s_1.s_2....,s_{|S|}\}$, and $P = \{p_1,p_2....,p_{|P|}\}$ respectively. Service seekers and providers will be collectively referred to as agents.

The set of service seekers constitute the alternative set of service providers and vice-versa. Together, they form a bipartite set, and the resource allocation problem is formulated as a bipartite matching problem.

The first step in matching involves quantification of preferences of seekers and providers using the expected utility theory [28]. The next step is to generate the preference rank ordering of alternatives for each participant based on the expected utilities. The matching algorithm is then implemented to match the service seekers to the most suited service providers. This is called a single-period matching. The steps involved in single-period matching have been discussed in Section 2.3.1.

In a stochastic environment, where the service seekers and providers arrive and exit the system as a continuous process over a long period of time (T) the matching mechanism needs to be implemented multiple times. The mechanism is implemented recursively after every fixed interval of time, t_{design}. The recursive implementation of the single-period matching after every fixed interval of time is called multi-period matching. The interval between two successive implementations of the matching mechanism is referred to as a matching cycle. During this period new service seekers place their service requests, the service providers complete their jobs assigned in previous cycle and become available. A suitable designed matching period t_{design} optimizes the outcome of the mechanism. Section 4.2.1 elaborates the modeling of multi-period matching in CBDM and Section 4.2.2 describes multi-period implementation of matching mechanisms.

4.2.1 Stochastic Modeling of Multi-period Matching Scenario

The demand for service and availability of resources determine the market thickness in CBDM. Both arrivals of service requests and processing of services are modeled as stochastic processes. This section describes the modeling of these stochastic processes.

Arrival of Service Seekers

The stochastic arrival of service seekers is modeled as a Poisson process with mean arrival rate λ. We chose a Poisson process to model the number of service seekers because (a) Poisson distribution models discrete events that occur in a finite and continuous interval of time, and (b) service seekers arrive from a wide range of sources independent of one another. This assumption is valid in a totally decentralized scenario where there is a large number of independent designers or groups of designers trying to get their parts printed or manufactured. The sources are designers or groups of designers trying to get their parts printed or manufactured. They are independent because the arrival time of a designer does not depend on the arrival time of other designers in the system. The set of service seekers on k^{th} matching cycle is denoted as S_k. The probability density function $\pi(|S_k|)$ over number of service seekers $|S_k|$ is given by Equation (4.1).

$$\pi(|S_k|) = \frac{e^{-\lambda}\lambda^{|S_k|}}{|S_k|!} \tag{4.1}$$

The mean arrival rate (λ) is a characteristic of the target population on which the mechanism will be implemented. The higher the demand for cloud-based services higher the value of (λ). The mean arrival rate varies from one setting to the next. The demand cannot be controlled by the mechanism designer. The goal is to design the matching period (t) for a given demand (or arrival rate λ) of the target population.

Availability of Service Providers

The set of service providers who are available on the k^{th} matching cycle is denoted by S_k; we have $S_k \subseteq S$. The availability of service provider p_j in a matching period depends on the complexity of the job assigned to them in the previous periods, the type of manufacturing resource they possess, and their average working time in a day (h_j). The complexity of the job depends on factors such as resolution requirement, design dimensions, detail capability etc. The complexity of service request (of the service seeker s_i that he/she got matched to) and the type of manufacturing resource (possessed by service provider p_j) determine the service time (τ_{ij}) needed for processing a request. For example, a service provider with a machine that uses a Stereolithography (SLA) process can print a design faster than a service provider possessing a machine that uses Fused Deposition Modeling (FDM) printing process. Comparison of the printing time required for 100 representative designs for various 3D printing processes and machines are discussed in further detail in Section 5.2.2. We assume that each service provider is available in a matching cycle if he/she completes the previous service request in the preceding matching cycles. $\frac{\tau_{ij}}{h_j t}$ is the number of matching cycles after which service seeker p_j is available after being matched to service seeker s_i. A service provider who is able to meet the service requirements faster is more available and therefore derives more utility during the matching duration T by being available on a higher number of matching cycles.

4.2.2 Multi-period implementation of Matching Mechanisms

In Chapter 3 it was shown that the Deferred Acceptance (DA) algorithm is the most suited matching mechanism in a totally decentralized design and manufacturing setting due to its properties such as stability, individual rationality, consistency, immunity to gaming behavior of the participating agents, monotonicity with respect to demand, and supply. Now, we extend this matching mechanism into a multi-period setting and determine the optimal period of implementation of the mechanism (t). The extension of the DA mechanism in a multi-period setting is described in Algorithm 4.

Algorithm 4 multi-period DA mechanism

Set $avail_i = 1 \; \forall \;\; p_i \in P$
for ($time \leftarrow t$ to T step t) **do**
  if($time$ is t) **then**
    Set each $p_i \in P$ as unassigned;
  else
    Unassign those $p_i \in P$ who have $avail_i = 1$
  end
  while ($\exists p_i \in P$ unassigned) and ($\exists s_k \in \; R_{p_i}$ not assigned to p_i) **do**
   s_i is first such s_k in R_{p_i} and s_i is provisionally assigned to p_j
     unassign s_i from p_j and assign s_i to p_i
    for each successor p_k on R_{s_i} remove p_k and s_i from each other's list
  end
  Set $avail_i = 0$ for all p_i in P who is assigned a seeker in this matching cycle
  Set $avail_i = 1$ for those p_i who completed previous assignment
textbfend

4.3 Analyzing the Effects of Matching Period

The period of matching (which is inverse of matching frequency) is designed depending on the characteristics of the scenario to which the matching mechanism is applied. Section 4.3.1 describes the main variables that characterize a target scenario. Efficiency of the matching mechanism is quantified based on several objectives. No matching mechanism uniquely optimizes all the objectives. Objectives that are important will differ from scenario to scenario. For example while fairness in the distribution of outcomes will be more important in a particular scenario, cumulative utility attained by all the service providers might be the most important one in another setting. Section 4.3.2 details the objectives considered and the metric used to assess them in our analysis.

Figure 4.1 illustrates the mechanism design model, the key variables in the inputs, and the metrics that quantifies the outcomes of the matching mechanism. Section 4.3.1 discusses the inputs to the model in detail. The inputs characterizes the application onto which the matching mechanism needs to be applied. Section 4.3.2 describes the metrics used to evaluate the efficiency of the matching mechanism. We propose optimal matching period in two different supply demand settings. Subsection 4.3.3 describes these two settings.

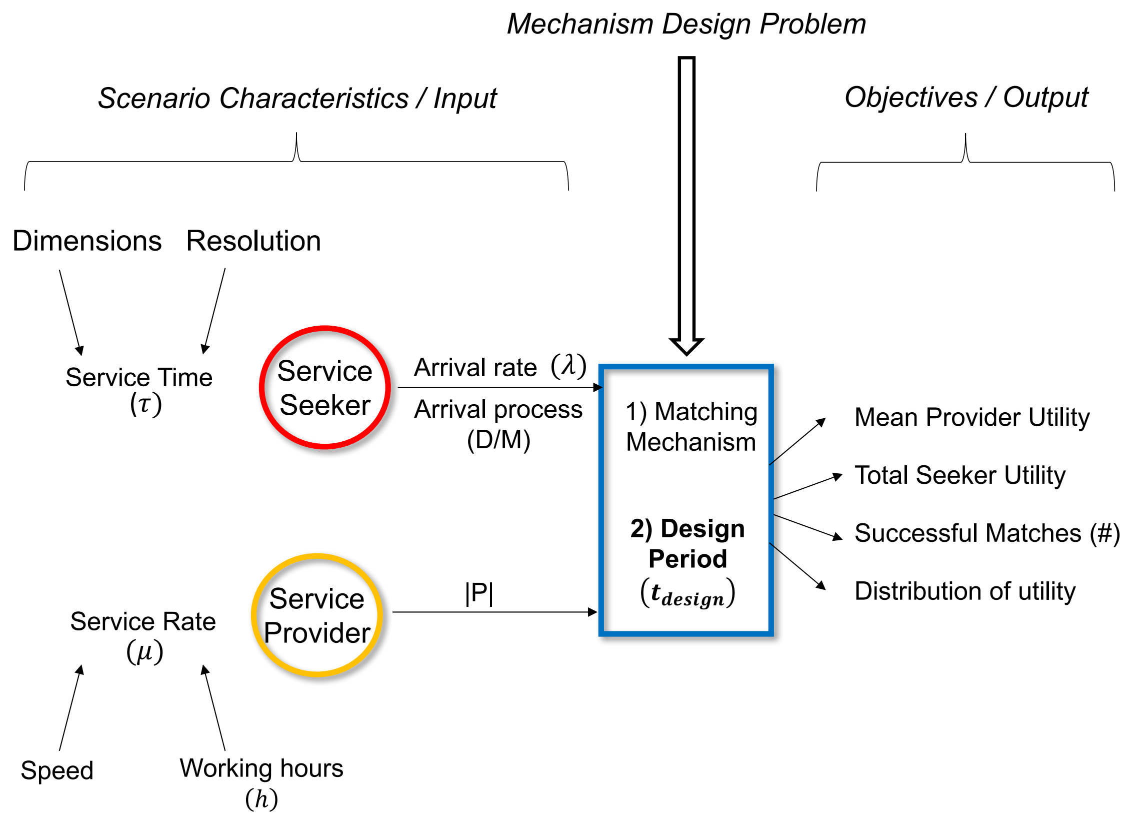


Figure 4.1. Block diagram illustrating the inputs and outputs of the mechanism design model.

4.3.1 Modeling the Stochastic Matching Scenario

The key variables that are altered to study the effect of matching frequency are: a) the arrival characteristics, b) the service characteristics, and c) the nature of interaction between service seekers and providers.

a) The arrival characteristics consists of the (i) arrival process of the service seekers, (ii) the mean rate of arrival (λ), and (iii) the complexity of the jobs that the service seekers are trying to print. (i) The arrival process is either deterministic or stochastic. Deterministic arrival indicates constant rate of arrival with time whereas Poisson arrival is changing with respect to time where the fluctuation is modeled by a Poisson process. Though deterministic arrival process might not mimic any real-world resource allocation application, it provides insights on the impact of matching period on the efficiency of matching outcome as a function of other input variables. In Chapter 5 we analyze the effect of relaxing this deterministic assumption. (ii) The mean rate of arrival indicates the average number of service requests per unit time. This is a characteristic of the population. A high arrival rate indicate that the demand is high in the target population. (iii) The complexity of the jobs depends on the size of the part that needs to be manufactured, the resolution requirements, detail capability etc. The mean arrival rate and job complexity together characterizes the demand in the population. A high arrival rate and a high job complexity represents a high demand.

b) The service characteristics consist of: (i) the number of service providers ($|P|$), (ii) the service rate (μ). The service rate depends on the speed at which the manufacturer processes the job, and the working time of the manufacturer (h). The speed is primarily a characteristic of the machine. The speed at which the manufacturer processes the job or equivalently how fast the job is printed in the 3D printer. For example 3D printers based on Stereolithography (SLA) printing process prints the same job with same dimensions and part complexity faster than the ones based on Fusion Deposition Modeling (FDM). The working time (h) is decided by the machine owner depending on the availability and excess resource capability. Service rate is a combination of both these factors. A high μ^j indicate high service rate for service provider p_j.

c) the nature of interaction between service seekers and providers consists of (i) the distribution of utility that the service seekers and provider gain by being matched, and (ii) the distribution of service (τ_{ij}) to prototype design i in machine j. (i) The distribution of the utility that the provider gains from being matched to a service seeker and vice-versa determines the level of compatibility, diversity in design, and resource choices. The utility is modeled to lie between 0 and 1 with 0 being the lowest and 1 the highest utility attainable. If all seeker gain utility 1 being matched to any service provider and vice-versa then this means that all the job requests can be completed in any of the service provider machine and the providers are invariant between the jobs they are matched to. If the distribution is binomial with a parameter close to 0 then this indicates that most of machines are incompatible to prototype the service request. In the analysis effects of various distribution of utility such as Binomial, Beta, Uniform is studied. (ii) The complexity of the job request by the service seeker and the speed of the machine that the service provider possess combine to determine the job time for that seeker manufacturer pair. In practice, the printing time is exponentially distributed for each service provider with a parameter that is characteristic of the machine. The distribution is exponential because there will be a small number of designs that needs really large printing time and most of the designs need a low printing time. The parameter of the exponential distribution is characteristic of the machine. A faster machine will have lower mean printing time for all the designs in the population.

4.3.2 Evaluating the Outcome of the Mechanism

We define an optimal matching mechanism based on four different objectives: a) number of successful matches, b) total expected utility attained by the entire set of manufacturers (denoted by $\overline{EU_P}$), c) total expected utility attained by the set of designers (denoted by $\overline{EU_S}$), and d) fairness in the distribution of utility among service providers (denoted by $\overline{\sigma_{EU_P}}$). The total expected utility attained by the set of manufacturers is

$$EU_P(t_k) = \sum_{j=1}^{|P_k|} E[u_{ji}]\mathbb{1}_{\mathbb{M}^{-1}} \quad where, \begin{cases} \mathbb{1}_{\mathbb{M}^{-1}} = 1, & M_{p_j}^{-1}(t_k) \in S_k \\ \mathbb{1}_{\mathbb{M}^{-1}} = 0, & otherwise \end{cases} \tag{4.2}$$

The total expected utility attained by set of designers is

$$EU_S(t_k) = \sum_{i=1}^{|S_k|} E[u_{ij}]\mathbb{1}_{\mathbb{M}} \qquad where, \begin{cases} \mathbb{1}_{\mathbb{M}} = 1 & M_{s_i}(t_k) \in P_k \\ \mathbb{1}_{\mathbb{M}} = 0 & otherwise \end{cases} \tag{4.3}$$

We use standard deviation of the utility distribution among the service providers as a measure of fairness in their distribution. For k^{th} matching the standard deviation in utility distribution among the service providers in that cycle ($|P_k|$) is expressed as Equation (4.4).

$$\sigma_{EU_P}(t_k) = \sum_{j=1}^{|P_k|} \left(\left(E[u_{ji}] - \frac{EU_p(t_k)}{|P_k|} \right)^2 \mathbb{1}_{\mathbb{M}^{-1}} \right) \tag{4.4}$$

The matching mechanisms are implemented over a total duration T. The objectives are assessed by calculating the cumulative over the entire matching duration as shown in Equation (4.5). In Equation (4.5), O_i indicates the measure of objective O at i^{th} matching cycle and $\lfloor \frac{T}{t} \rfloor$ is the total number of matching cycles. Here, $\lfloor\rfloor$ denotes floor function. O is one of either EU_P, EU_S, σ_{EU_P}, the fraction of successful matches depending on the objective of the mechanism we consider for efficiency.

$$\overline{O} = \sum_{i=1}^{\lfloor \frac{T}{t} \rfloor} O_i \tag{4.5}$$

4.3.3 Two supply-demand settings

We present analytic results for optimal matching frequency in two different supply-demand settings. This section describes the two different supply-demand settings.

The product of the number of service providers $|P|$ and the rate at which they offer service μ represents the supply of resources in the population. The product of rate of arrival of service seekers λ and the average job-processing time τ represents the demand for services in the population. When demand is lower than supply we refer to it as low supply (LS) setting and otherwise we refer it it as high supply (HS) setting. Equation 4.6 shows this in mathematical notation.

$$\begin{aligned} HS &: |P|\mu \geq \lambda\tau \\ LS &: |P|\mu < \lambda\tau \end{aligned} \tag{4.6}$$

Therefore, the goal is to optimize the matching period for a given demand and supply. Demand depends on the arrival rate of service seekers, arrival process, and the printing time to process the service requests. Supply depends on the number of service providers offering service, their availability, and service speed. We study the effect of matching period on multi-period Munkres, DA, and TTC algorithms. The results showing the effect of matching period on each of the four objectives are shown in Sections 4.4 and 4.5. Section 4.4 discusses the results in HS setting and Section 4.5 discusses the results in LS setting.

4.4 Optimal matching period under high supply setting

In this section we consider the HS setting when the supply is higher than demand as defined in Section 4.3.3. In HS, we have $\lambda\tau \leq |P|\mu$. Here, $|P|$ is the number of service providers and μ is their service rate. λ is the rate of arrival of service seekers into the system and τ is their average job-processing time.

We had considered four objective in designing optimal matching period as defined in Section 4.3.2. For the objective number of successful matches, we propose the following theorem in HS setting.

Theorem 1: If $|P|\mu \geq \lambda\tau$ with deterministic arrival of service seekers with arrival rate λ, a constant job processing time τ and a constant service rate μ, then the total number of matches for a given matching period t over a total matching duration T is given by

$$\text{No. of matches} = \begin{cases} \frac{T}{|P|/\lambda}, & if \quad t \leq \frac{|P|}{\lambda} \\ \frac{T}{t}, & if \quad t > \frac{|P|}{\lambda} \end{cases} \tag{4.7}$$

where, $|P|$ is the number of service providers and λ is the rate of arrival of the service seeker.

Proof: We consider three separate cases

Case 1: $\frac{\tau}{\mu} \leq t \leq \frac{|P|}{\lambda}$

At every time-interval t, λt service seekers enter the system

Since $t \geq \frac{\tau}{\mu}$, each service provider completes the assigned task in the same matching cycle in which it was assigned. As a result, all P service providers are available in every matching cycle.Thus, $P_i = P$ where P_i is the number of service providers in i^{th} matching cycle.

All the λt service seekers gets matched as $\lambda t \leq |P_i|$ and every match is perfectly compatible with utility 1. The mean utility attained by service providers in i^{th} matching cycle $EU_P = \frac{\lambda t}{|P|}$. Mean utility attained over the entire matching duration T is given by

$$\overline{EU_P(t)} = EU_P\frac{T}{t} = \frac{T}{|P|/\lambda} \qquad when \;\; \frac{\tau}{\mu} \leq t \leq \frac{|P|}{\lambda}$$

Case 2: $t < \frac{\tau}{\mu}$

At every t interval λt seekers arrive. $\lambda t \leq |P|$ but not necessarily $|P_i|$ as now not all

providers are available in every matching cycle since $\mu t < \tau$. The service providers take $\frac{\tau}{\mu t}$ number of matching cycles to complete the matched service task.
We have $\frac{|P|}{\tau/\mu t}$ number of service providers available in each matching cycle.
$|P_i| = \frac{|P|}{\tau/\mu t} \geq \frac{|P|}{|P|/\lambda t} = \lambda t$.
Therefore, $|P_i| \geq \lambda t$. As a result all λt service seekers gets matched in each matching cycle. Following the procedure similar to Case 1, we obtain

$$\overline{EU_P(t)} = \frac{T}{|P|/\lambda} \qquad \textit{when} \;\; t < \frac{\tau}{\mu}$$

Case 3: $t > \frac{|P|}{\lambda}$
Since $t > \frac{|P|}{\lambda} \geq \frac{\tau}{\mu}$, we have $P_i = P \;\; \forall i$. Now number of service seekers exceeds the number of service providers in every matching cycle. As a result $|P|$ service seekers among the λt total gets matched in every matching cycle.
$EU_P = 1$ Mean utility attained over the entire matching duration T is given by

$$\overline{EU_P(t)} = \frac{T}{t} \qquad \textit{when} \;\; t > \frac{|P|}{\lambda}$$

In Theorem 1 we expressed one of the matching objectives as a function of matching period. However, we had four objectives to consider to design the optimal matching period. In Sections 4.4.1 and 4.4.2 we analyze the effect of matching period on the other objectives under certain assumptions about the distribution of utility attained by agents being matched.

4.4.1 Binary Utility Setting

In this section we evaluate the matching objectives as a function of matching period assuming that the utility on being matched is binomially distributed in the population. The binomial parameter is denoted by α. α indicates the fraction of successful matches in the target population. For a randomly drawn service seeker service provider pair, the utility of being matched is 1 with probability α and is 0 with probability $1 - \alpha$. Thus the utility is either 0 or 11. A utility of 1 indicates a compatible pair and 0 indicates a non-compatible pair. In applications such as Kidney Exchange where the donor and receivers are either compatible or incompatible the utility can be modeled using this assumption. In distributed manufacturing the utility can be modeled as a binary variable where utility is 1 if the design can be printed in a machine and 0 otherwise.

We begin by the most simplistic case where the binomial parameter is unity ($\alpha = 1$). This means that any service seeker or service provider upon being matched to any of their alternatives attains a utility of 1. This represents a situation where the service providers are invariant to the service seeker they are being matched to and the service seeker do not differentiate the service provide that processes their service request as long as it is processed. Every service request can be processed in any machine and the designer attains identical utility. Similarly, the machine owner attains identical utility on processing service request of any designer he/she is matched to.

If any service provider or service seeker attains a utility of 1 upon being matched, then the total service provider utility and the total service seeker utility is numerically equal to the number of successful matches. Under this setting, the three objectives number of successful matches, total service seeker utility, and total service provider utility are identical. An optimal matching period maximizes the number of successful matches, total service seeker utility, and the total service provider utility. All the three objectives remain at a constant maximum value for $0 < t \leq \frac{|P|}{\lambda}$ and hyperbolically decreases for $t > \frac{|P|}{\lambda}$. Therefore, any matching period in the range $(0, \frac{|P|}{\lambda}$ is optimal considering the number of successful matches, total service seeker utility, and the total service provider utility. For the final objective fairness, which is quantified by standard deviation in the distribution of utility among the service providers, higher the matching period the better. If the period of matching is small then the more sought after provider keeps getting matched every cycle increasing the standard deviation in the distribution of matches and thereby the fairness. The standard deviation decreases to 0 as matching period increases from 0 to $\frac{|P|}{\lambda}$ and remains 0 for $t \geq \frac{|P|}{\lambda}$. Therefore, any matching period in the range $[\frac{|P|}{\lambda}, \infty]$ is optimal considering fairness as the matching objective. In conclusion, when matching period t is $\frac{|P|}{\lambda}$ all the four objectives are optimized. Thus the optimal matching period is when $t_{optimal} = \frac{|P|}{\lambda}$.

We simulated this stochastic environment using the values of the parameters tabulated in Table 4.1.

Table 4.1.
Values of the parameters used for the simulation study for the results presented in this section.

Parameter	**Value**
λ (Arrival rate)	12 per day
τ (Job time)	1.5 units per hour
μ (Service speed)	5 units per hour
$\lvert P\rvert$ (Number of service providers)	50

Comparison of the theoretical result with simulation for the scenario presented in Table 4.1 is shown in Figure 4.2. From the theorem, the number of successful matches remains constant for $t \leq \frac{|P|}{\lambda}$ and hyperbolically decays for $t > \frac{|P|}{\lambda}$. From Figure 4.2 we observe that the simulation results match well with the prediction. The minor staircase effect is because the simulation is performed for a finite value of $T = 300$. Floating point error occurs when the simulation is performed for a small finite matching duration. For example when say $T = 30$, changing the matching period from 10 to 10.1 decreases the number of matching cycles from 3 to 2. These errors can be addressed by choosing a total matching duration T much higher than the range of frequency analyzed. In the simulation studies conducted, a period of $T = 300$ was chosen to assess the matching period ranging from 0 to 20. If the simulation is extended for longer duration these fluctuations will die out and converge to the theoretical result. However, the fluctuations are negligible even for finite duration truncation.

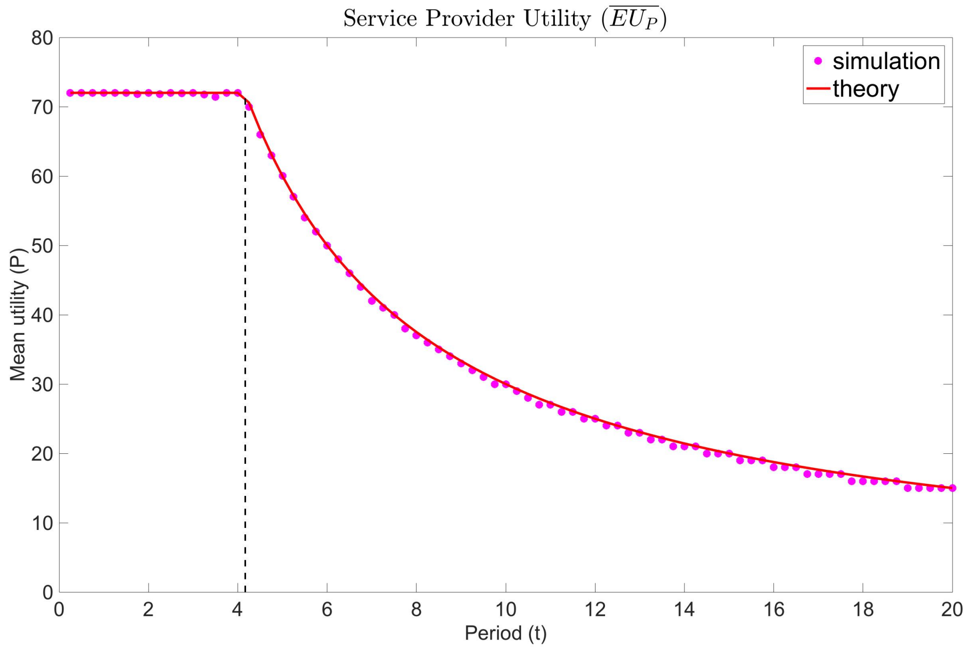


Figure 4.2. Comparison of theoretical prediction to simulation studies for the effect of matching period on the mean service provider utility attained by implementing multi-period Munkres mechanism under high service rate setting. The parameters used in the simulation are tabulated in Table 4.1.

Figures 4.3, 4.4, and 4.5 show the effect of period of matching on the four different matching objectives which are service provider utility, service seeker utility, fairness in the distribution of utility, and the number of successful matches. Figure 4.3 shows the mean service provider and the service seeker utility for various periods of matching (t). For service seeker utility total utility was chosen as the metric to assess the efficiency of match as unlike the service provider they are not fixed. The number of service seekers depends on the period of matching itself and they change from one cycle to another. Both service seeker and service provider utility is constant for $t \leq \frac{|P|}{\lambda}$ or $t \leq 4.17$. Mean service provider utility in this range of matching period is a constant value and is given by $\overline{EU_P(t)} = \frac{T}{|P|/\lambda} = 72$. Total service seeker utility when $t \leq 4.17$ is given by $\overline{EU_S(t)} = 3600$.

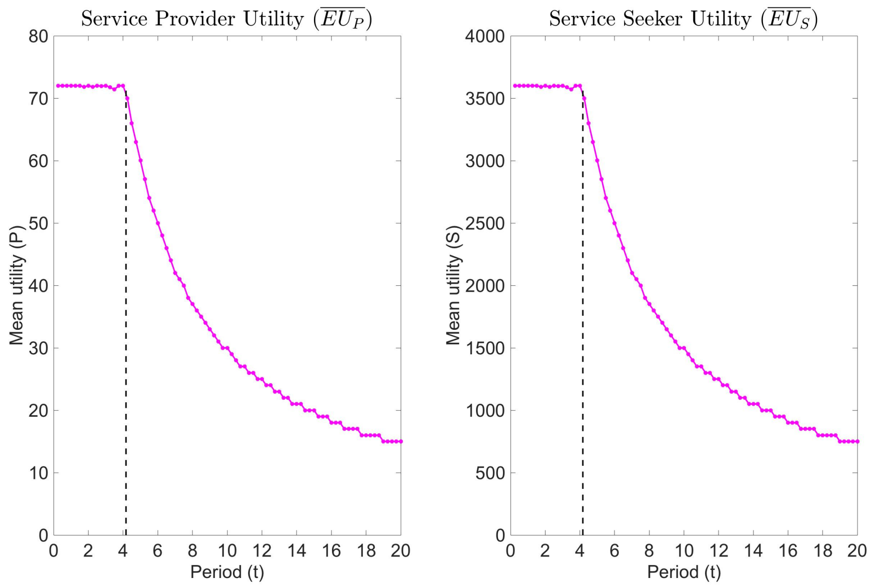


Figure 4.3. Effect of matching period on the mean service provider utility and total service seeker utility attained by implementing the multi-period Munkres mechanism under high service rate setting.

Figure 4.4 shows the fairness in the distribution of utility among the service providers. Lower the standard deviation in the distribution of utility among the service providers, more equitable the distribution. When matching period t is small the more preferred service providers keep getting matched period after period. $t < \frac{|P|}{\lambda}$ not all service providers gets matched in every cycle. As the service rate is high, the service providers who are matched complete their service request in the same matching cycle in which they got matched. Hence, all service providers are available in every matching cycle regardless of being matched in the immediate previous matching cycle. When this happens the more sought after service providers keep getting matched in every matching cycle causing unfairness in the distribution of utility among service providers. When t increases to $\frac{|P|}{\lambda}$, more service providers participate in every matching cycle thereby decreasing the variance or unfairness in the distribution of utility. For $t \geq \frac{|P|}{\lambda}$ all service providers gets matched in every matching cycle attaining identical utility and has zero variance or maximum fairness.

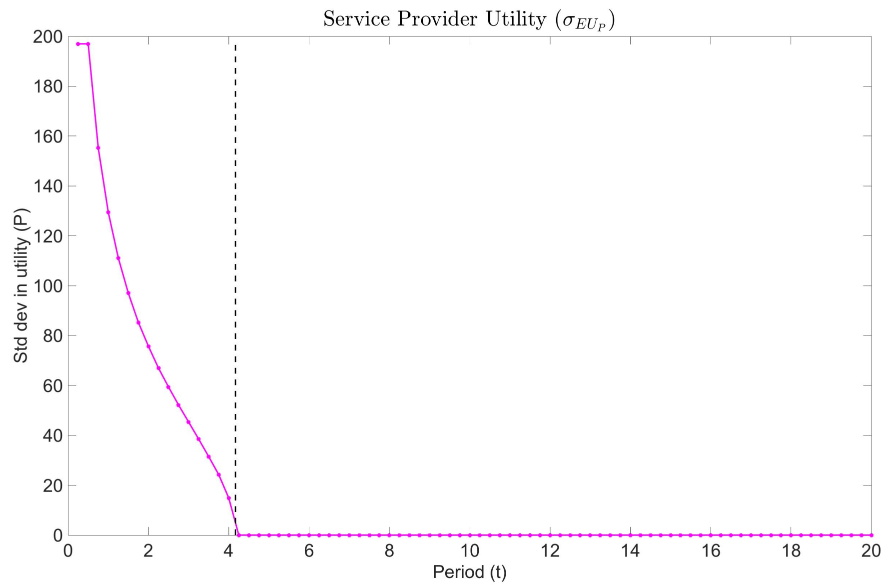


Figure 4.4. Effect of matching period on the distribution of utility among service providers by implementing the multi-period Munkres mechanism under high service rate setting.

Figure 4.5 shows the number of matches and standard deviation of the distribution of total matches among the service providers. The matches are counted for each service provider over a total matching duration T during which the mechanism is implemented. Equation (4.8) shows the number of total matches over the entire matching duration T for this setting. If the period of matching is small then the more sought after provider keeps getting matched every cycle increasing the standard deviation in the distribution of matches and thereby the fairness.

$$\text{number of matches} = \begin{cases} \lambda T, & if \quad t \leq \frac{|P|}{\lambda} \\ |P|\frac{T}{t}, & if \quad t > \frac{|P|}{\lambda} \end{cases} \tag{4.8}$$

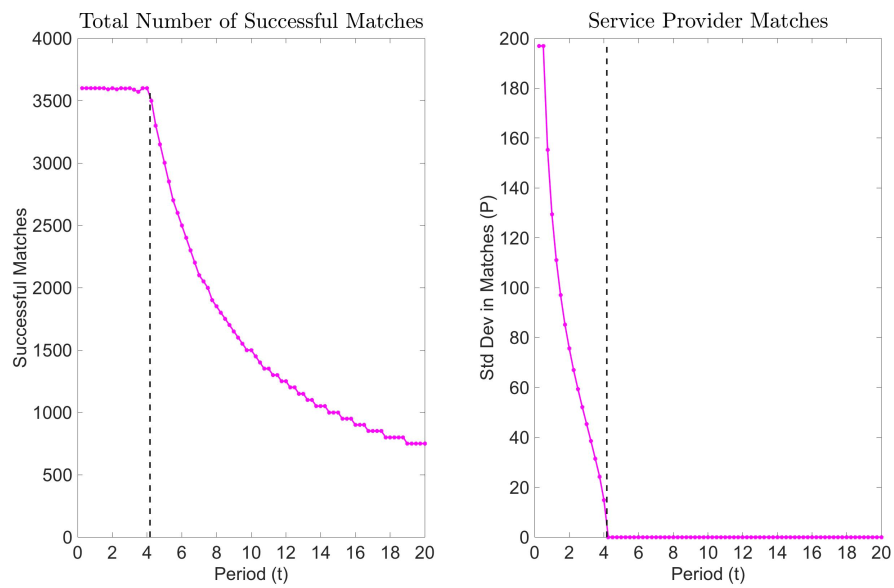


Figure 4.5. Effect of matching period on the total number of successful matches and distribution of matches among service providers by implementing the multi-period Munkres mechanism under high service rate setting.

Now we consider the generic case where $\alpha \in [0, 1]$. Here, we relax the assumption that all the possible matches are perfect compatible. The utility being matched is given by the binomial distribution i.e. *utility* $B(1, \alpha)$, where α is the parameter of the distribution. The special case considered earlier was the perfectly compatible setting when $\alpha = 1$. When $\alpha = 0$ no matching is compatible. As α increases from 0 to 1 the number of possible compatible matches increases. α denotes the probability with which matching between a randomly chosen service seeker is compatible. For the generic case we discuss the effect of matching period on multi-period Munkres, DA, and TTC mechanisms.

(i) Multi-period Munkres

Figure 4.4.1 shows the mean utility attained by service providers and total utility attained by service seekers for various values of the utility parameter α. When α is decreased from 1 to 0.1 i.e. by a factor of 10 times, the mean service provider utility is mostly unaffected. Consider a matching setting where $|P|$ service providers are being matched to $|P|$ service seekers. Each service provider has $|P|$ alternatives with a fraction $\alpha|P|$ of the matches being compatible (utility = 1). The Munkres assignment mechanism is applied to match service providers to service seeker. We order the service providers in the set as follows: $\{1, 2, ..., \alpha|P|, \alpha|P| + 1, ..., \alpha|P| + n, ..., |P|\}$. Each service provider has $\alpha|P|$ compatible matches and hence the first $\alpha|P|$ service providers are assigned to a compatible service seeker and attains a utility of 1. The service provider $\alpha|P| + 1$ receives a compatible match with probability $1 - \frac{{}^{\alpha|P|+1}P_{\alpha|P|}}{{}^{|P|}P_{\alpha|P|}}$ i.e. it receives a compatible match provided all its potentially compatible alternatives are not already matched. In general, for a service provider $\alpha|P| + n$ we can obtain a lower bound on the utility assuming all the previous $\alpha|P| + n - 1$ service providers is matched. The service provider $\alpha|P| + n$ finds a compatible match with probability $1 - \frac{{}^{\alpha|P|+n}P_{\alpha|P|+n-1}}{{}^{|P|}P_{\alpha|P|}}$. This probability is also the expectation of utility attained by the service provider as the utility is 1 if found a compatible match and 0 otherwise. For the simulation performed with $|P| = 50$. When $\alpha = 0.1$ the mean expected utility of service providers is 99% of that when $\alpha = 1$. But the service seeker utility decreases more steeply. This is because Munkres mechanism optimizes the utility of only one set of agents in the bipartite matching setting.

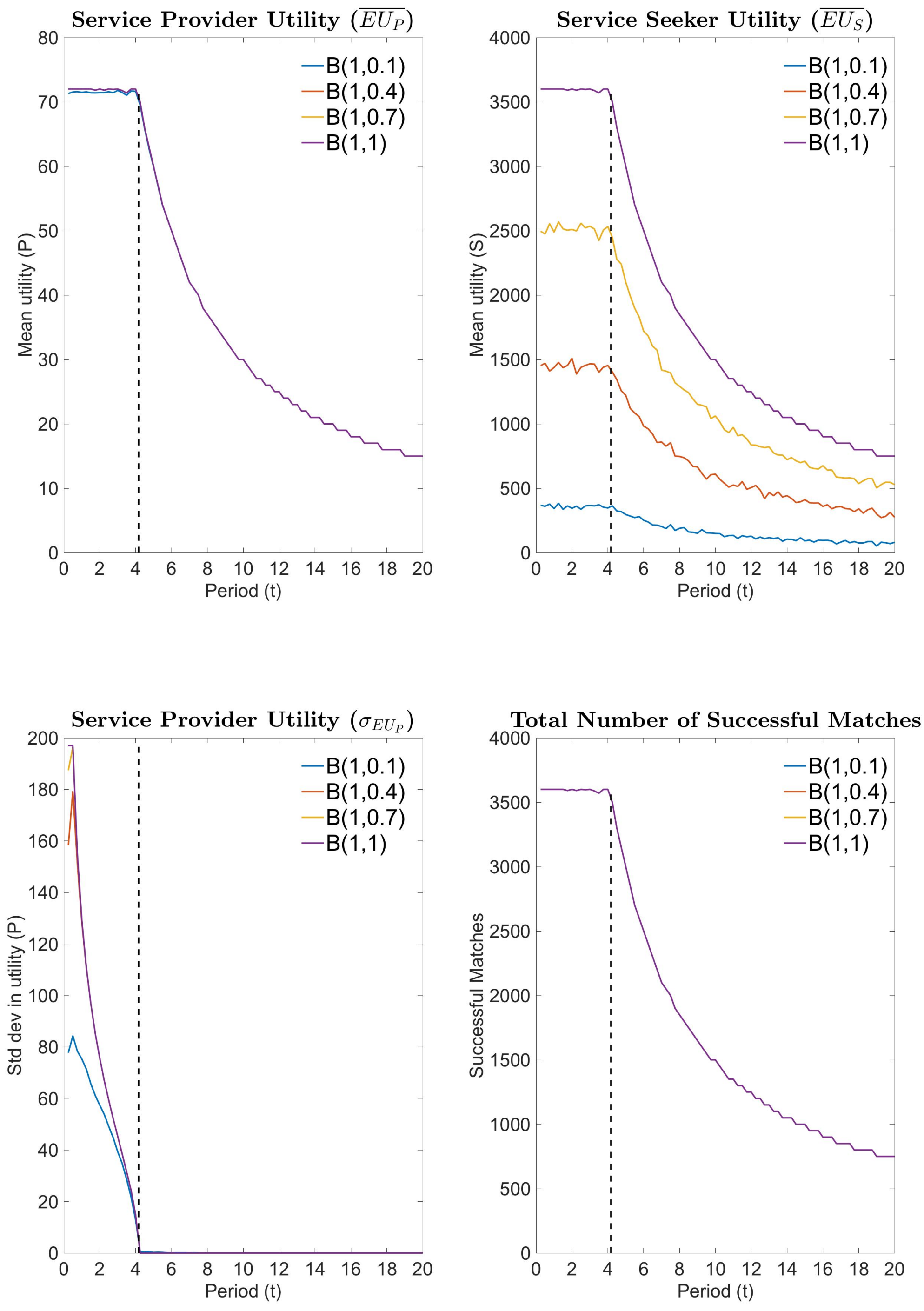


Figure 4.6. Performance of multi-period Munkres mechanism for different matching period (t) under high service rate setting with utility of being matched following a binomial distribution ($B(1, \alpha)$). Results are shown for α ranging between 0.1 and 1.

From the Figure 4.6 we notice that even though mean expected utility dropped by 10 times, the total attained utility by service providers did not change. In Figure 4.4.1 shows the results of service provider and service seeker utility for α varying between 0.01 and 0.1. Now, the fraction of incompatible matches start affecting the mean service provider utility. For a set of $|P| = 50$ service providers being matched with $|P| = 50$ service seekers with only a 0.01 fraction of it being compatible, the mean expected utility of service providers is 49% of that when $\alpha = 1$. The service seeker utility fluctuates around a constant value for $t \leq \frac{|P|}{\lambda}$ and drops down hyperbolically. This is because the total service seeker utility follows the same curve as the total number of matches and the fluctuations are because Munkres mechanism ignores optimizing the utility of service seekers.

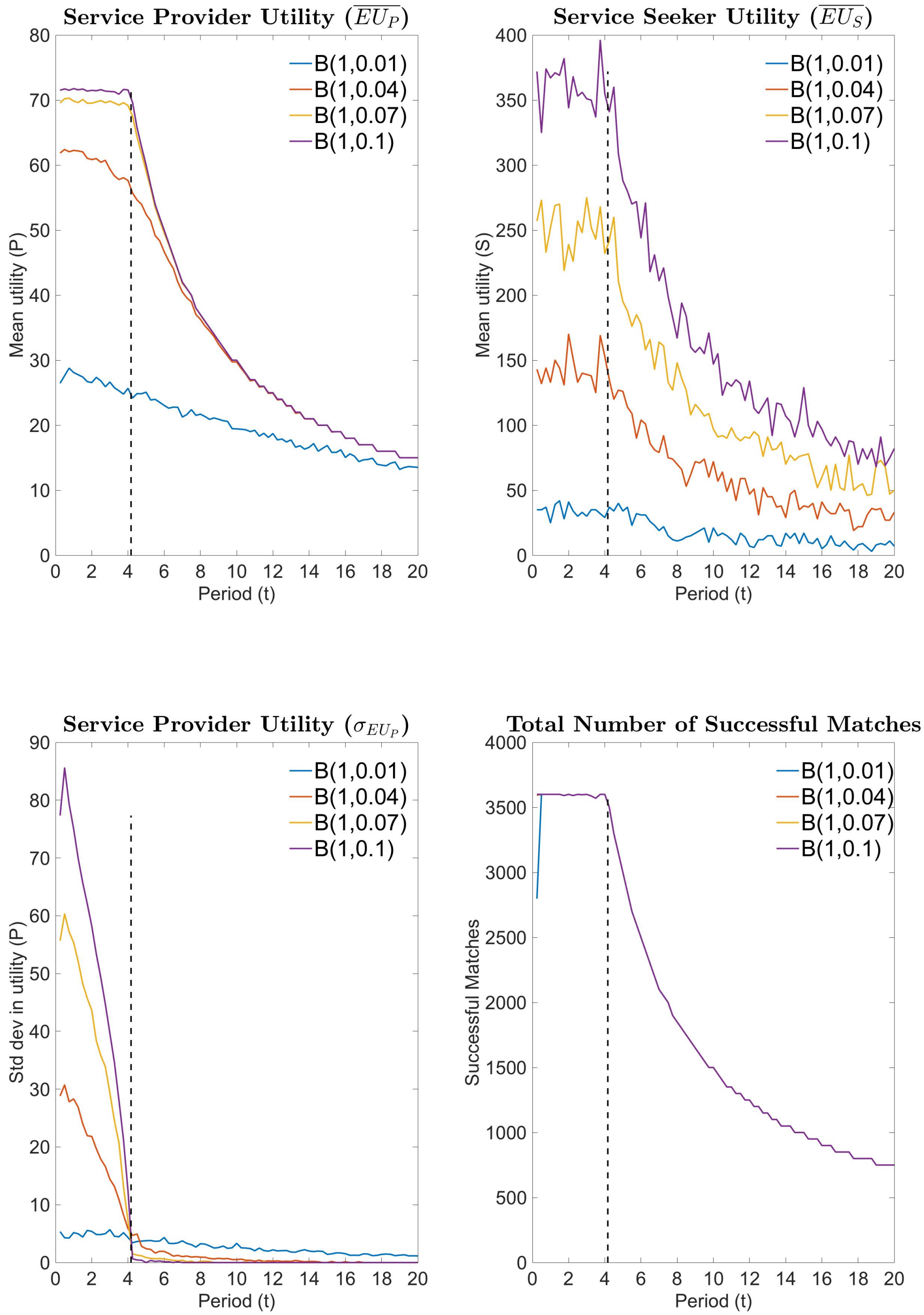


Figure 4.7. Performance of multi-period Munkres mechanism for different matching period (t) under high service rate setting with utility of being matched following a binomial distribution ($B(1, \alpha)$). Results are shown for α ranging between 0.01 and 0.1.

Figures 4.6 and 4.7 shows the fairness in distribution of utility among the service providers for α in the range $[0.1, 1]$ and $[0.01, 0.1]$ respectively. The fairness follows a trend similar to the one discussed for $\alpha = 1$. When matching period t is small the more preferred service providers keep getting matched period after period. $t < \frac{|P|}{\lambda}$ not all service providers gets matched in every cycle. The more sought after service providers keep getting matched in every matching cycle. For the same matching period t a higher α result in higher unfairness as a higher fraction of the matches take place leading to higher difference in distribution of utility attained by more sought after service providers and the less sought after ones. When α is extremely small the standard deviation in utility is positive even for some of the matching period $t > \frac{|P|}{\lambda}$. For example this is seen for $\alpha = 0.01$ in Figure 4.7. However, it eventually goes down to zero when there are enough service seeker alternatives to compensate for a large fraction of incompatible matches.

The number of successful matches and unfairness in their distribution is shown in Figure 4.6 for α in the range $[0.1, 1]$. Regardless of the degree of incompatibility the standard deviation in service seeker utility drops down steeply from $t = 0$ to $t = \frac{|P|}{\lambda}$. The mean service provider utility, total service seeker utility, and the number of successful matches remains constant for $t < \frac{|P|}{\lambda}$ after which it hyperbolically drops down to 0 for matching periods $t > \frac{|P|}{\lambda}$. Any matching period $t \leq \frac{|P|}{\lambda}$ is optimal from the utility and number of successful matches standpoint. However, unfairness in the distribution of utility and number of successful matches among the agents is least when $t = \frac{|P|}{\lambda}$. Therefore, considering both the utility attained and unfairness in their distribution, the optimal matching period is when $t = \frac{|P|}{\lambda}$ in high service setting.

(ii) Multi-period DA

We consider the scenario summarized in Table 4.1 except that now we study the effect of multi-period DA algorithm.

Similar to Munkres, we begin by considering the perfectly compatible setting i.e. when $\alpha = 1$. Since DA produces only stable matches regardless of the number of service seekers in the system only one match take place per matching cycle. Consider the setting when there is one service seeker and 2 service providers p_1, p_2. The only stable match is to match the service seeker to his/her most preferred service provider. Since the service provider is indifferent between both the service providers (as it is a perfectly compatible setting) matching either p_1 or p_2 to the service provider is stable. Now, if the matching period is long enough to bring in another service seeker to the system, then the only stable solution is to match one of the service provider to one of the service seekers (say the service seeker who arrived first in time). Any other matches blocks this rendering the outcome unstable. Therefore, waiting longer does not yield additional utility but only decreases the number of possible matching cycles over the matching duration T. Thus the total number of matches, service seeker, and service provider utility hyperbolically drops down as matching period t increases in a perfectly compatible setting as seen in the simulations results shown in Figures 4.3 and 4.5.

(iii) Multi-period TTC For multi-period TTC, the behavior is similar to multi-period Munkres because there is no constraint of stability as in DA. Figure 4.4.1 shows the matching objectives as a function of matching period for different values of the binomial parameter α. The optimal matching period is same as that of multi-period Munkres.

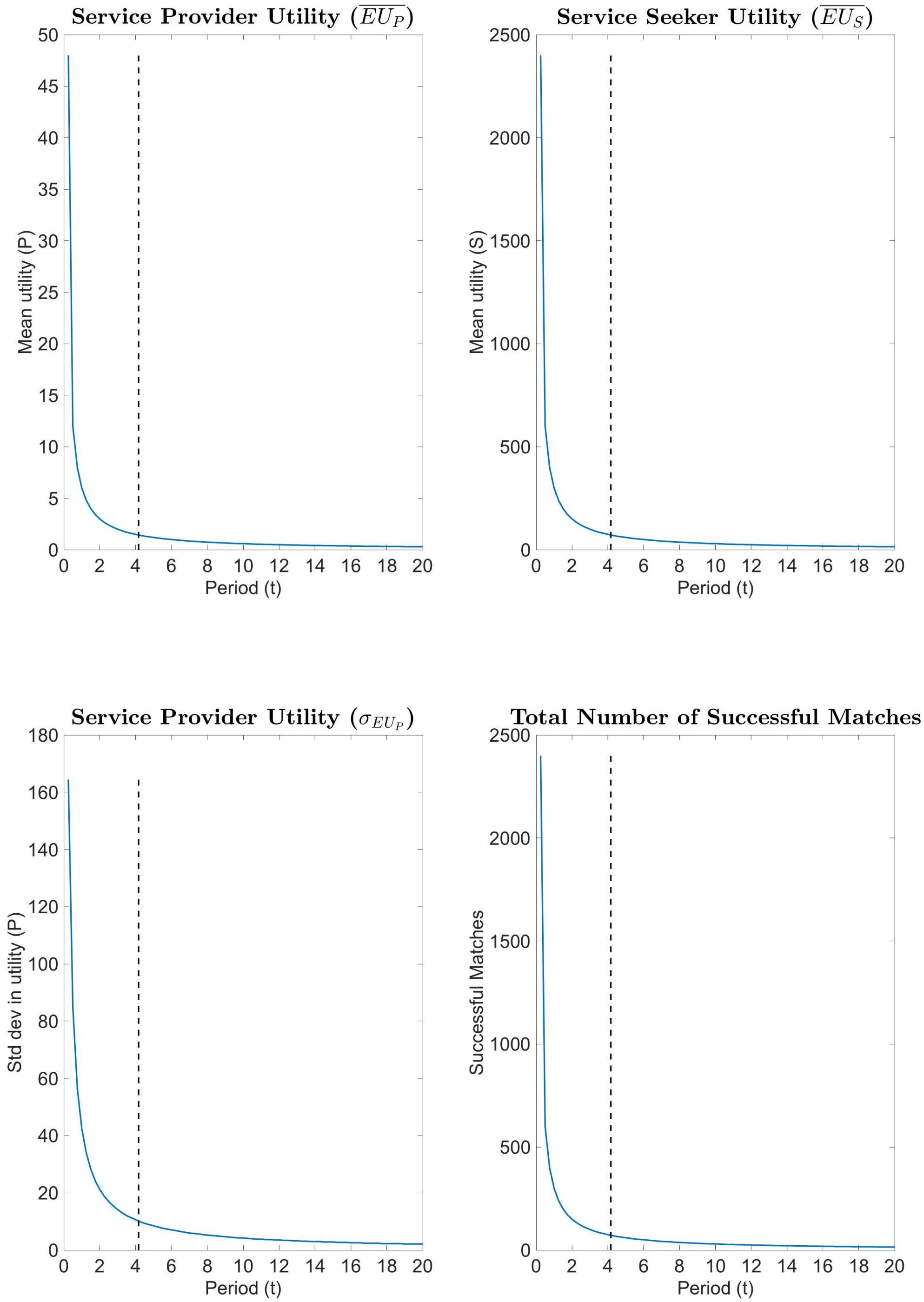


Figure 4.8. Performance of multi-period DA mechanism for different matching period (t) under high service rate setting with perfectly compatible matches.

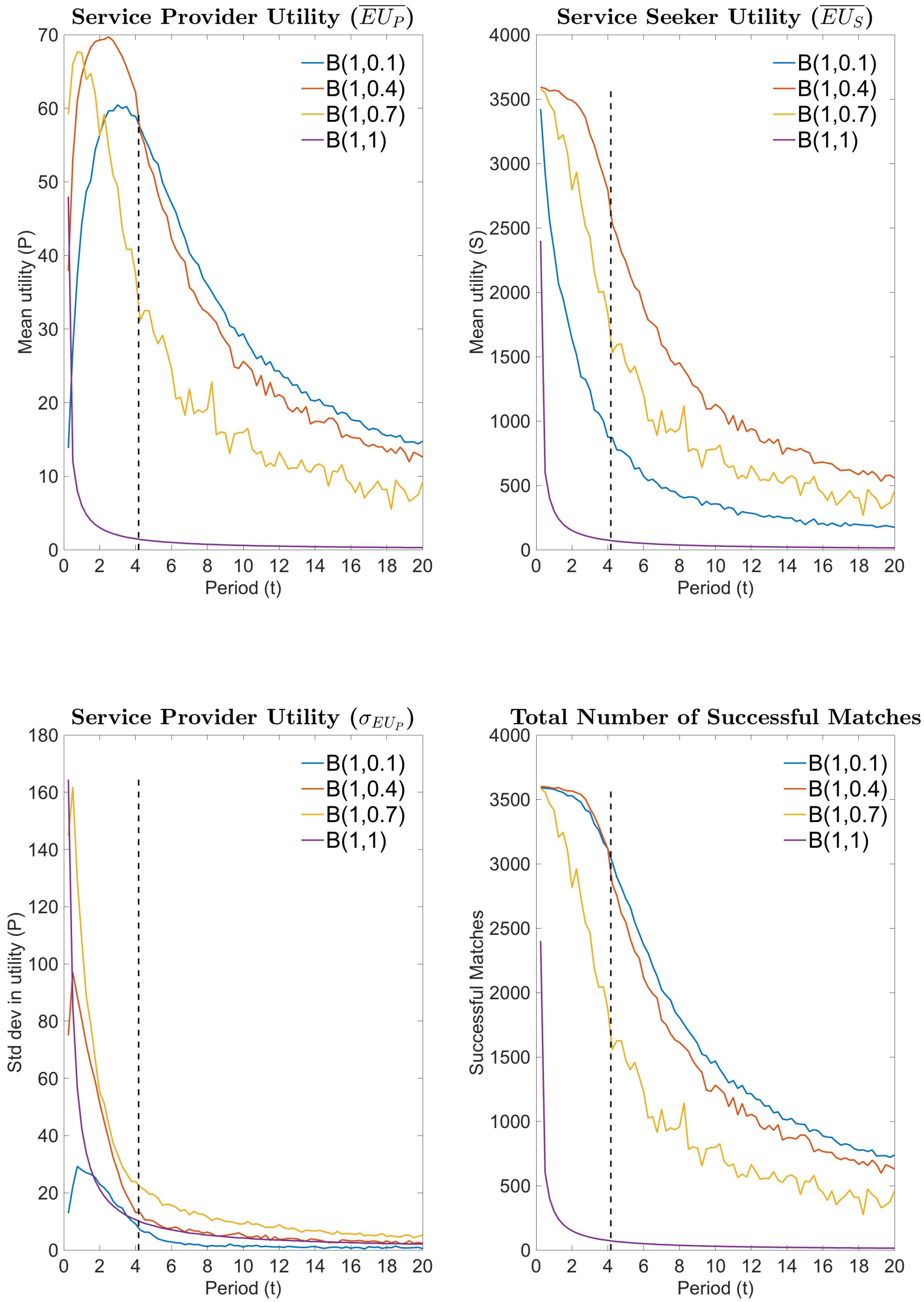


Figure 4.9. Performance of multi-period DA mechanism for different matching period (t) under high service rate setting with utility of being matched following a binomial distribution ($B(1, \alpha)$). Results are shown for α ranging between 0.01 and 0.1.

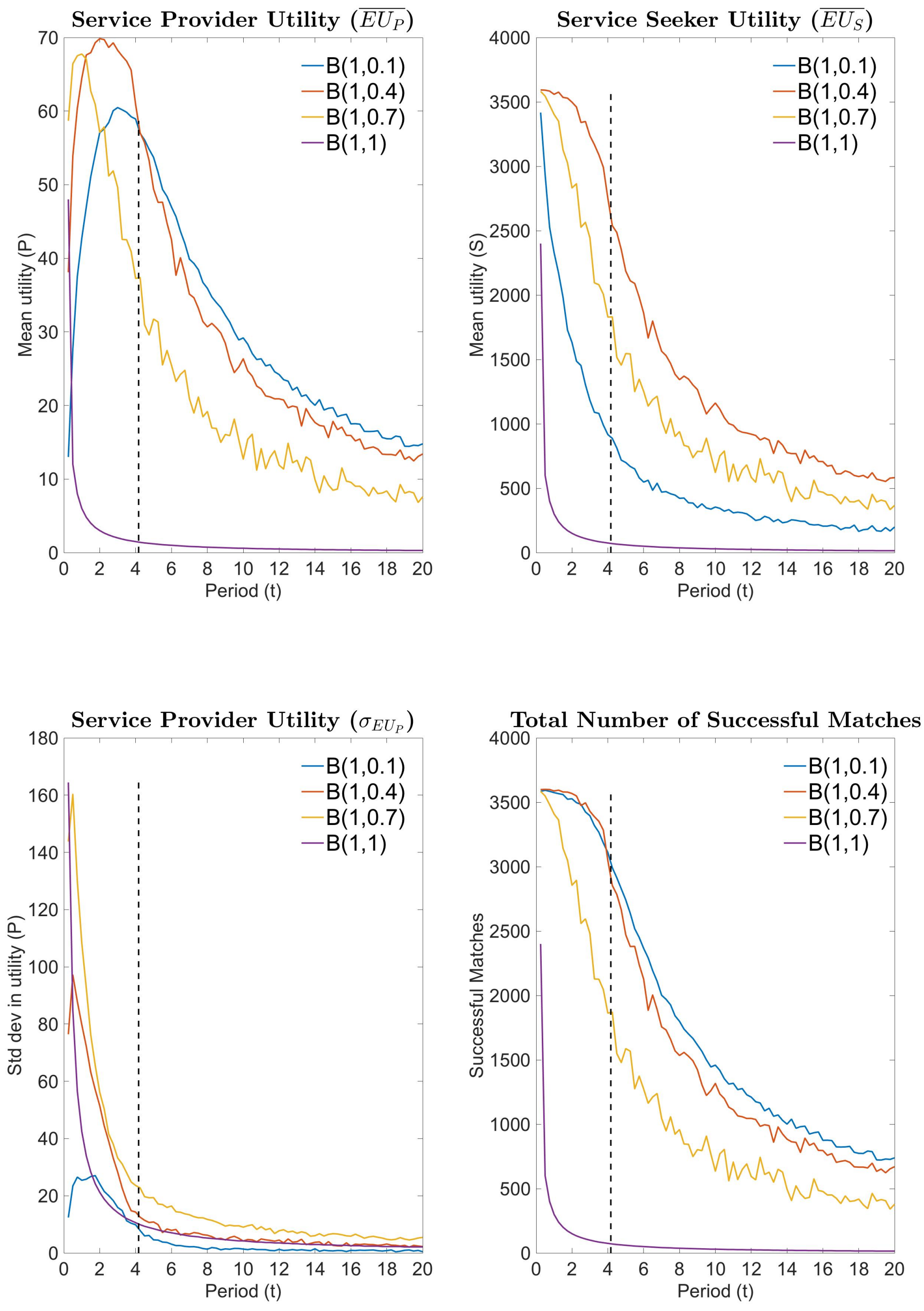


Figure 4.10. Performance of multi-period TTC mechanism for different matching period (t) under high service rate setting with utility of being matched following a binomial distribution ($B(1, \alpha)$). Results are shown for α ranging between 0.1 and 1.

4.4.2 Beta Distribution of Utility

In Section 4.4.1, the utilities of being matched where either 0 or 1 which indicated situations where the matching is either compatible or incompatible. However, in many applications this might not be the case. For example, in CBDM when multiple machines (or service providers) meet the size, and other requirements of the service seeker but one service provider offers more suited material than the other assigning both the service providers utility = 1 is not desirable. At the same time, there needs to be a metric to distinguish the utility of being matched to either of the service provider. Therefore allowing the utility to assume any value between 0 and 1 provides additional flexibility in distinguishing such matches. In this section, we assume the utility of being matched follows a general beta distribution and study how it affects the design matching period recommended in Section 4.4.1. We analyze the effect of matching period on multi-period Munkres, DA, and TTC mechanisms.

(i) Multi-period Munkres

If $|P|\mu \geq \lambda\tau$ with deterministic arrival of service seekers with arrival rate λ, a constant job processing time τ and a constant service rate μ with utility of service seeker and service provider being matched being drawn from Beta distribution ($beta(a, b)$), then the total utility attained by service seekers and service providers for a given matching period t over a total matching duration T is given by Equations (4.9) and (4.10).

$$\overline{EU_P(t)} = Number\ of\ matches(t)\ max_{|P|} \tag{4.9}$$

$$\overline{EU_S(t)} = Number\ of\ matches(t)\ \frac{a}{a+b} \tag{4.10}$$

where, $\max_{|P|}$ is the expected maximum of $|P|$ random numbers drawn from a beta distribution $Beta(a, b)$ and number of matches is given by Equation (4.7). For example, when $a = b = 1$, beta distribution is same as standard uniform distribution and we have $\max_{|P|} = \frac{|P|}{|P|+1}$. Figure 4.11 compares the theoretically proposed utility of service providers with simulation experiments for different values of $|P|$. Other parameters used in the simulation is same as the ones tabulated in Table 4.1.

Proof: The proof is similar to the proof of proposition 1. For the three separate cases the number of matches remain unchanged.

Case 1: $\frac{\tau}{\mu} \leq t \leq \frac{|P|}{\lambda}$

Total number of matches in the i^{th} matching cycle is $\frac{\lambda t}{|P|}$. The utility is drawn from a beta distribution with parameters a, b. Munkres mechanism chooses the largest of $|P|$ random numbers selected from a standard uniform distribution. Therefore, the utility attained by each matched service provider in expectation is given by $\frac{|P|}{|P|+1}$. Mean utility attained in i^{th} matching cycle is $EU_P = \frac{\lambda t}{|P|+1}$ Mean utility attained by the service providers over the entire matching duration T is given by

$$\overline{EU_P(t)} = \frac{T}{{}^{(|P|+1)}/_{\lambda}} \qquad when \;\; \frac{\tau}{\mu} \leq t \leq \frac{|P|}{\lambda}$$

Case 2: When $t < \frac{\tau}{\mu}$

following the procedure similar to Case 2 in the proof of proposition 1 and applying the maximum rule for $|P|$ random numbers we have

$$\overline{EU_P(t)} = \frac{T}{{}^{(|P|+1)}/_{\lambda}} \qquad when \;\; t < \frac{\tau}{\mu}$$

Case 3: When $t > \frac{|P|}{\lambda}$

we have $|P|$ service seekers among the λt total gets matched in every matching cycle. $EU_P = \frac{|P|}{|P|+1}$ Mean utility attained over the entire matching duration T is given by

$$\overline{EU_P(t)} = \frac{|P|T}{(|P|+1)t} \qquad when \;\; t > \frac{|P|}{\lambda}$$

Figure 4.11 compares the theoretical proposition with simulation experiments for different values of $|P|$. The simulation is performed maintaining other parameters as the ones described in Table 4.1. The utility is from a standard uniform distribution. Therefore the parameters of the Beta distributions are $a = 1$, $b = 1$.

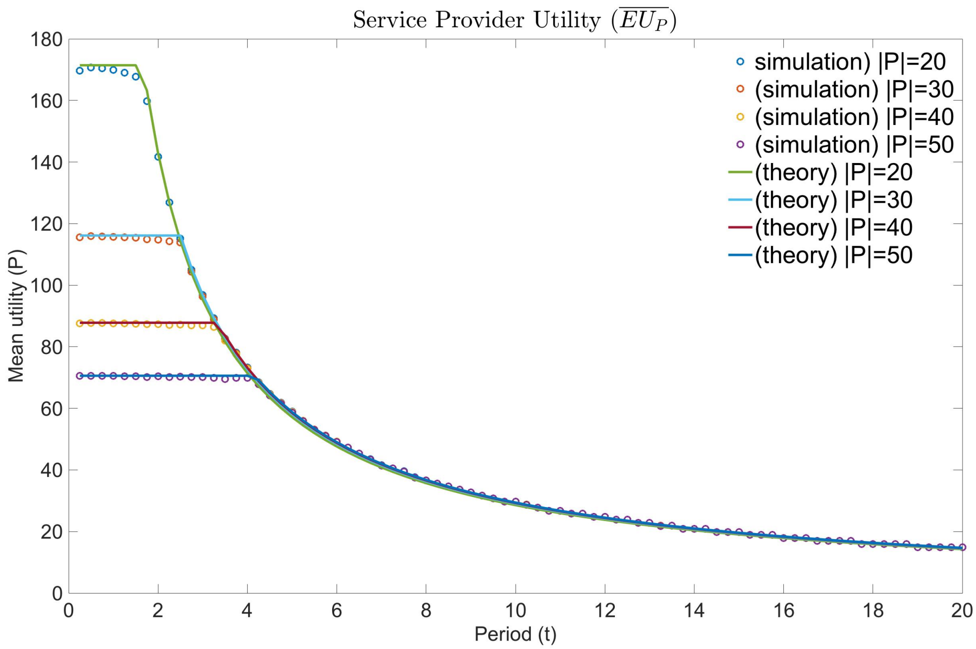


Figure 4.11. Comparison of theoretical prediction to simulation studies for the effect of matching period on the mean service provider utility attained by implementing multi-period Munkres mechanism under high service rate setting with utility of being matched drawn from a standard normal distribution.

Figure 4.12 shows the mean utility attained by service providers and total utility attained by service seekers when matching period is varied from 0 to 20 for the scenario presented in Table 4.1. Service provider utility follows the curve as discussed in Proposition 2. Now, unlike the scenario where all matches are perfectly compatible, we have the total service seeker utility at a matching period equaling approximately half of the total number of expected matches for the same period. This is because the Munkres mechanism optimizes the utility of only the service provider set. Utility of service seekers are not considered in the algorithm. Since the utility of service seeker being matched to an alternative is obtained from a standard uniform distribution, in expectation the matches result in a utility of 0.5 per match. Thus the total utility attained by service seekers in expectation follows Equation (4.11). Figure 4.3 also shows the comparison of the service seeker utility as proposed by Equation (4.11) with simulation studies for the parameters defined in Table 4.1.

$$\overline{EU_S(t)} = \begin{cases} \frac{0.5|P|T}{(|P|+1)/\lambda}, & if \quad t \leq \frac{|P|}{\lambda} \\ \frac{|P|^2 T}{(|P|+1)t}, & if \quad t > \frac{|P|}{\lambda} \end{cases} \tag{4.11}$$

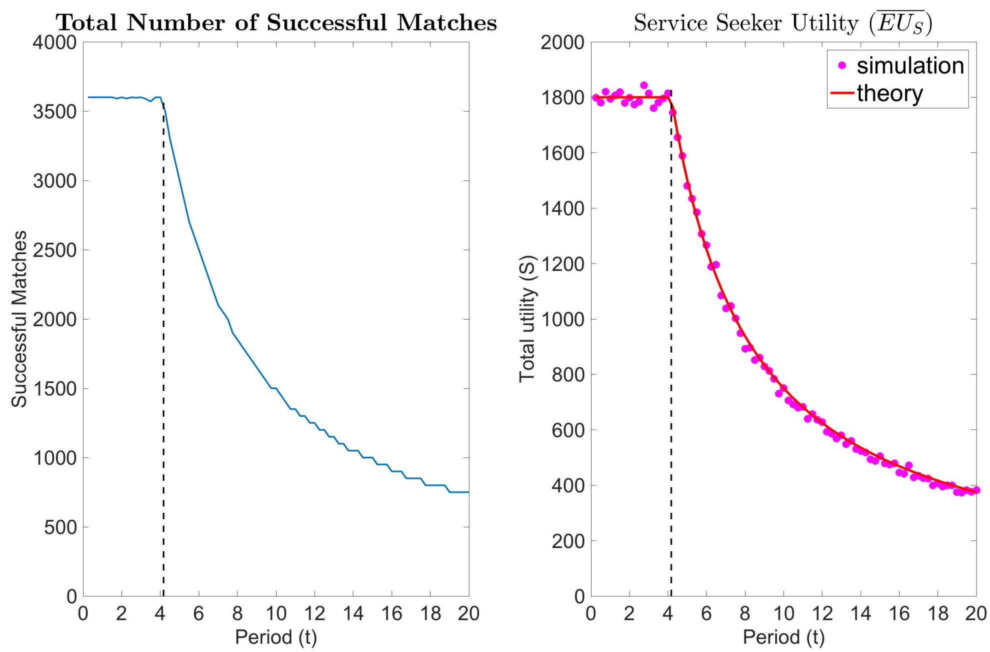


Figure 4.12. Effect of matching period on the mean service provider and total service seeker utility attained by implementing multi-period Munkres mechanism under high service rate setting with utility of being matched drawn from a standard uniform distribution.

Figure 4.13 shows the effect of matching period on the distribution of utility among the service providers. Similar to the case for perfect compatibility (as shown in Figure 4.4) the standard deviation in the distribution of utility drops down drastically as t increases from 0 to $\frac{|P|}{\lambda}$. However, unlike in the case of perfect compatibility the standard deviation does not drop to 0 for $t = \frac{|P|}{\lambda}$and remain 0 for $t > \frac{|P|}{\lambda}$. There is still a non-zero standard deviation in the distribution of utility among service providers even at $t = \frac{|P|}{\lambda}$. This is because for $t \geq \frac{|P|}{\lambda}$ only the standard deviation in the distribution of matches goes to 0, there is still a non-zero standard deviation due to randomness in the utility attained by matches. In contrast to the case of perfect compatibility where all utility were 1, here we have utility drawn from a standard uniform distribution. However, as t increases from $\frac{|P|}{\lambda}$ to ∞ the standard deviation of this distribution decays down to 0 as the service providers have more matching options waiting for a longer period in individual matching cycles. The standard deviation in the distribution of utility drops sharply from $t = 0$ to $t = \frac{|P|}{\lambda}$. At $t = \frac{|P|}{\lambda}$ the standard deviation is only 2.5% of the maximum value. For $t > \frac{|P|}{\lambda}(= 4.17)$ the gain in standard deviation drops much less drastically as seen from Figure 4.13. This is because for $t > \frac{|P|}{\lambda}$ the decrease in standard deviation is because of marginally improved quality of matches of the less sought after service provider. Whereas, when $t < \frac{|P|}{\lambda}$, increasing t increase the number of successful matches of the less sought after service provider, which has a much more dominant effect on the standard deviation of the distribution. Lower the standard deviation in the distribution, the better as it denotes more fairness in the distribution of match outcomes. At the same time in Figure 4.12 we saw that the utility of matching decreases hyperbolically for $t > \frac{|P|}{\lambda}$ for both service seekers and service providers and remains constants for $t \leq \frac{|P|}{\lambda}$. Thus considering both the objectives $t = \frac{|P|}{\lambda}$ produces the most desired outcome which we propose as the design matching period under high service rate setting with utility of being matched following a standard uniform distribution. Therefore $t_{design} = \frac{|P|}{\lambda}$.

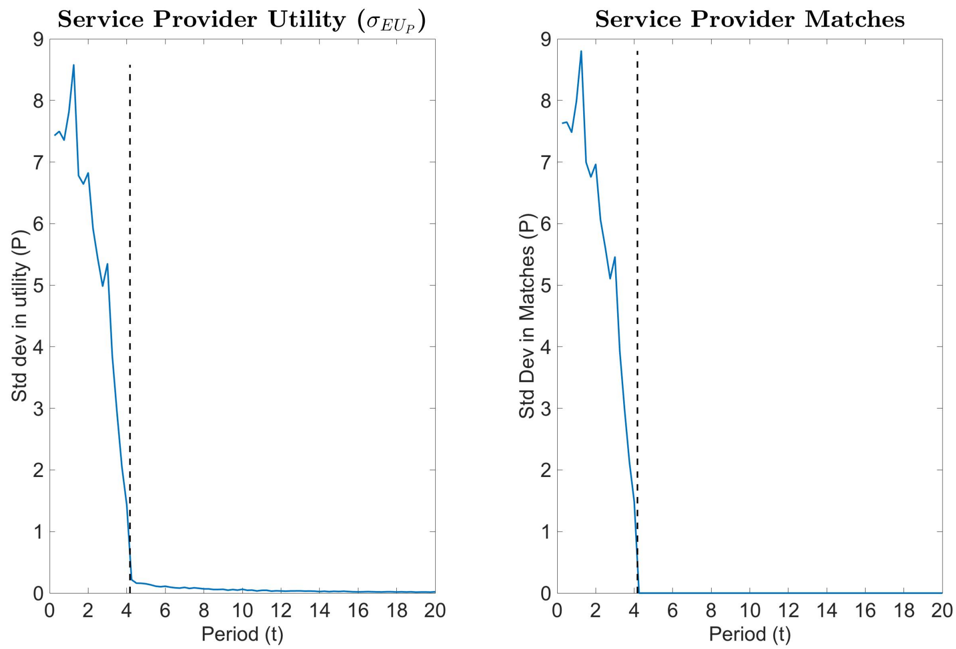


Figure 4.13. Effect of matching period on the distribution of utility and number of matches among service providers by implementing multi-period Munkres mechanism under high service rate setting with utility of being matched drawn from a standard uniform distribution.

Figure 4.14 shows the results for other values of parameter a, b in a general beta distribution ($Beta(a, b)$). For other values of a and b we observe that the general trend of mean seeker, total provider utility, and the distribution of provider utility is unchanged from that of standard uniform distribution (i.e. when $a = b = 1$). Only their magnitudes have decreased as the value of parameters b relative to a increases. The expected value of beta distribution is given by $\frac{a}{a+b}$. When b increases the expected value decreases. Lowering the magnitudes of utility decreases the magnitude of the effect, however the general trend of the effects is preserved. This is because the number of successful matches for a given matching period t is unchanged even if the values of parameters a, b deviates from that of a standard uniform distribution. Therefore for $0 \leq t \leq \frac{|P|}{\lambda}$ the mean service seeker utility and total service provider utility remains constant in this period after which it hyperbolically decreases. The mean service provider utility is expressed as in Equation (4.12).

$$\overline{EU_P(t)} = Number\ of\ matches \times \delta_{max} \tag{4.12}$$

where, δ_{max} denote the maximum of $|P|$ random numbers drawn from $Beta(a, b)$. This is the reason why even though value of b increases from 1 to 9 (nearly 9 times), the service provider utility is majorly unchanged as the value of δ_{max} does not drop down much. For example, when $a = b = 1$, we have $\delta_{max} = \frac{|P|}{|P|+1}$. In a perfectly compatible setting we have $\overline{EU_P(t)} = Number\ of\ matches$ i.e. $\delta_{max} = 1$ which is not much different from $\delta_{max} = \frac{|P|}{|P|+1} = 0.98$.

The total service provider utility in a high frequency setting with utility following a beta distribution ($Beta(a, b)$) is expressed as in Equation (4.13).

$$\overline{EU_S(t)} = Number\ of\ matches \times \frac{a}{a+b} \tag{4.13}$$

.

This is because Munkres mechanism focuses only on the service provider side maximizing their utility and therefore the utility of the service seeker side is a random number drawn from $Beta(a, b)$ which in expectation has the value $\frac{a}{a+b}$. In both Equations (4.12) and (4.13), the number of matches is given by Equation (4.14).

$$Number\ of\ matches = \begin{cases} \lambda T, & if\ \ t \leq \frac{|P|}{\lambda} \\ \frac{|P|\lambda}{t}, & if\ \ t > \frac{|P|}{\lambda} \end{cases} \tag{4.14}$$

For example when $a = 1, b = 9$, we have $\frac{a}{a+b} = 0.1$ and total service provider utility ($\overline{EU_S}$)=$0.1\lambda T = 360$ for $t \leq \frac{|P|}{\lambda} = 4.17$ as seen in Figure 4.14.

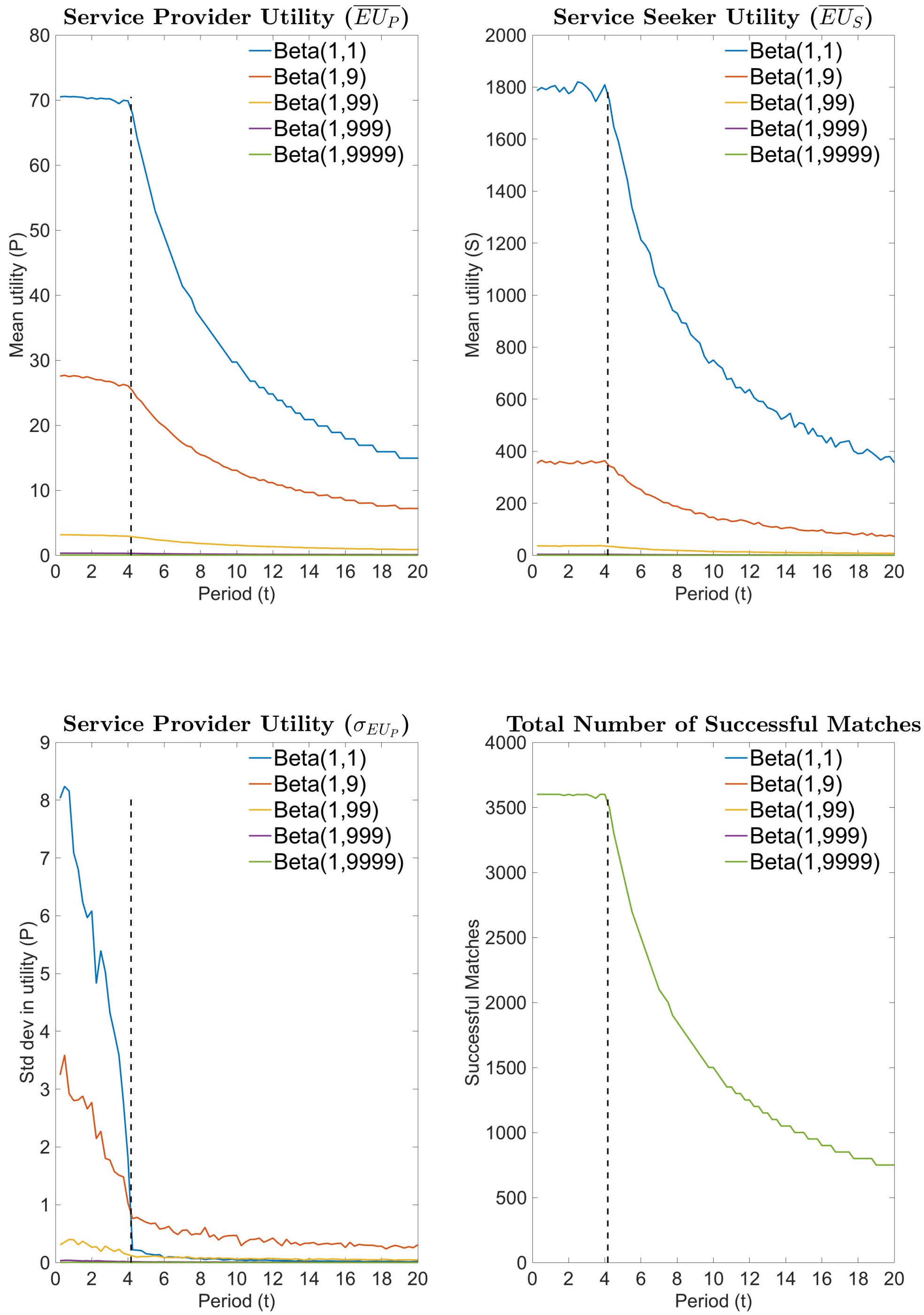


Figure 4.14. Effect of matching period on different matching objectives by implementing multi-period Munkres mechanism under high service rate setting with utility following a beta distribution, $Beta(a, b)$. Parameters used in the simulation are tabulated in Table 4.1.

Multi-period DA

While applying Multi-period Munkres when job time was constant we observed that for high service rate setting the utility remained constant for $t \in (0, \frac{|P|}{\lambda})$ and the utility decreased from its optimal value hyperbolically for $t > \frac{|P|}{\lambda}$ with the optimal value being at $\frac{|P|}{\lambda}$. When the matching mechanism whose frequency is being optimized is DA instead of Munkres, for the same setting a slightly different pattern is observed. Utility attained and fairness in the distribution of utility among service providers and service seekers for this scenario are shown in Figure

When the period of matching cycle is increased, manufacturer (or service provider) utility ($\overline{EU_P}$) gets affected due to two reasons: a) the sample size of service seekers increases thereby increasing the number of alternatives to choose from, b) the number of matching cycles decreases over a fixed duration T thereby decreasing the average utility attained over an assessment duration T. Both of these causes have opposing effects on the average manufacturer utility.

When the matching period t is increased from 0, at low values of t, the effects due to a low sample size of service seeker alternative are more dominant considering the average utility attained by all service providers. This is because most of them remain unmatched in each matching cycle. Moreover, if the average service request processing time τ is lower, then the unfairness in utility distribution is more prominent at small values of matching period t. This is because the more sought-after manufacturer will be matched in each cycle while the others remain unmatched. If τ is high, then the less desirable manufacturers get matched due to the lack of availability of the manufacturers already matched in every matching cycle.

At large values of t, the marginal effect from an increased sample size due to increased t is less prominent. Now, the effect of a decreased number of matching cycles becomes more prominent. Therefore, as t is varied from 0 to large numbers there is an increase in service provider utility initially, followed by a decrease. The point where the effect of marginal increase in period on the service provider utility undergoes a transition from increase to decrease, is the optimal matching period.

From simulation studies, we obtain that the optimal matching period is $t_{design} = \frac{|P|}{\lambda}$. This is the period at which service providers attain the highest utility. This is not an exact point as the optimum shifts mildly due to stochasticity in arrival patterns, randomness in utility distribution, the working time of service providers. But over a wide range of arrival rate λ, t_{design} is the optimal matching period. The reason is that when $0 < t < t_{design}$ there are not enough service seekers in the system (in expectation) at the instance of matching to match all the service providers. Thus, when t is increased from 0 to t_{design} the effect of an increased number of service seekers is more pronounced than the effect of a decreased number of matching cycles. However, when $t > t_{design}$ there is sufficient number of service seekers to match all the service providers and only the quality of alternatives or number of matching outcomes to choose from improves. As a result, the effects due to a decreased number of matching cycles start taking more precedence, thereby decreasing the overall utility. Also, the presence of a $|P|$ number of service seekers does not guarantee that all service providers in P will get matched. This is because DA produces only stable matches and there may be no stable solution that gets all $|P|$ agents matched. In practice, the required number of service seekers is slightly higher than $|P|$ because of this restriction of stability. This is why in the simulation studies the actual optimal point is slightly higher than the design optimal point as seen in Figure 4.4.2. For example, when $\lambda = 1$ we have $t_{design} = 5$ but the actual optimum occurs at $t = 6$, and when $\lambda = 2$ we have $t_{design} = 2.5$ when the actual optimum is at 3. Thurber [53] showed that the number of stable matches under DA mechanism (denoted by $|M|$) when λt men and women are being matched is

$$|M| > \frac{1.509^{\lambda t}}{1 + \sqrt{3}} \quad when \quad |\lambda t| \geq 1 \tag{4.15}$$

As a result, the actual optimum is not too far away from t_{design} as the number of stable matches grows in power of t.

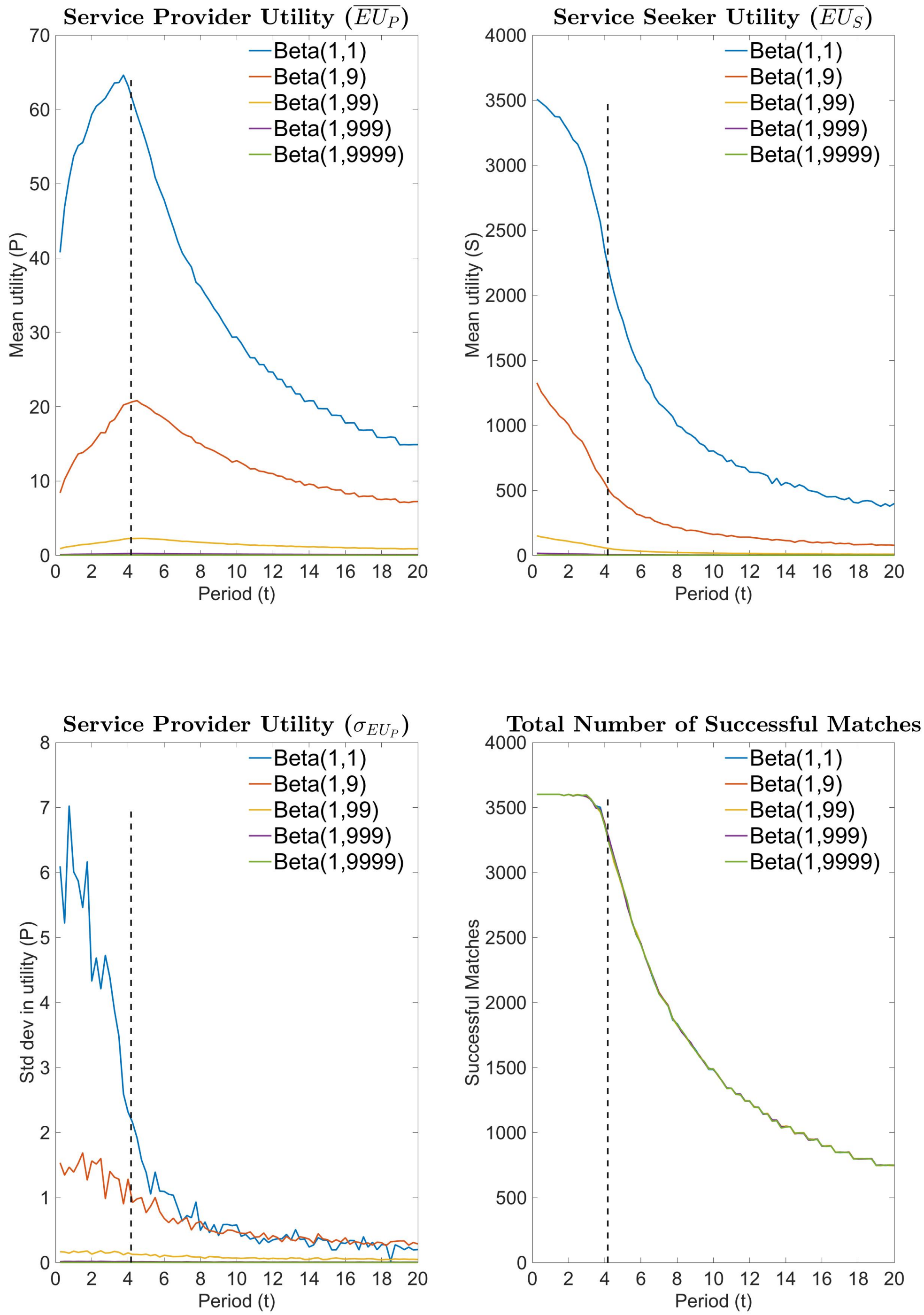


Figure 4.15. Effect of matching period on different matching objectives by implementing multi-period DA mechanism under high service rate setting with utility following a beta distribution, $Beta(a, b)$.

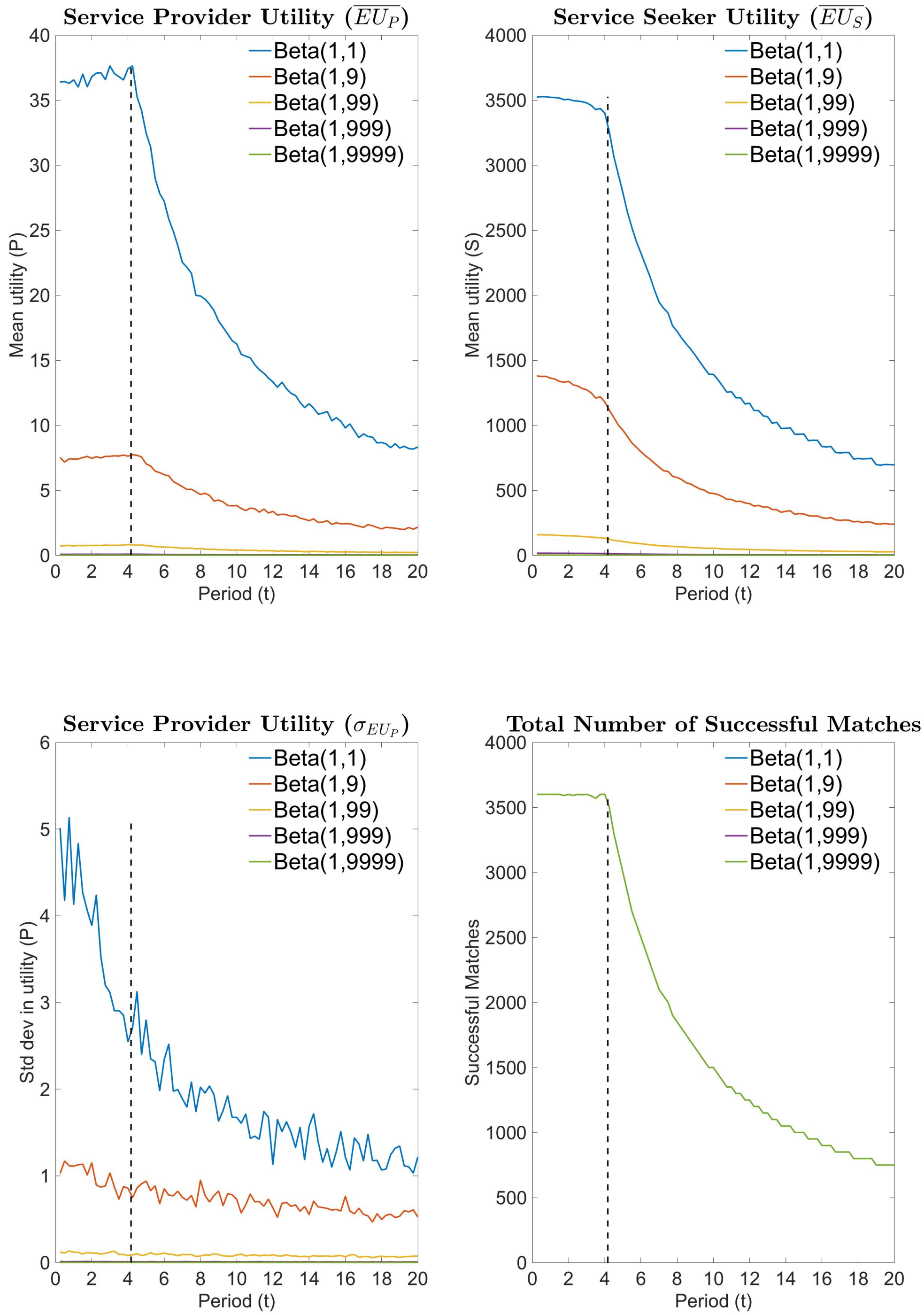


Figure 4.16. Effect of matching period on different matching objectives by implementing multi-period TTC mechanism under high service rate setting with utility following a beta distribution, $Beta(a, b)$.

4.5 Results B: Effect of Matching Period Under Various Utility Distribution With Low Service Rate

Now we consider the setting where service rate is low compared to arrival rate i.e. $\frac{\tau}{\mu} > \frac{|P|}{\lambda}$. We study the effect of period of matching on multi-period implementation of Munkres, DA, and TTC mechanisms in this section.

4.5.1 Binomially Distributed Utility

In this section we assume that the utility of being matched follows a binomial distribution with parameter α. The parameters for the simulation experiments used in this section are summarized in Table 4.2. We have $\frac{\tau}{\mu} = 5$ and $\frac{|P|}{\lambda} = 4.17$ for the the parameters chosen for simulation studies. Thus they satisfy the low service rate criteria.

Table 4.2.
Values of the parameters used for the simulation study for the results presented in this section.

Parameter	**Value**
Utility	B(1,α)
Arrival Process	Deterministic
λ (Arrival rate)	12 per day
τ (Job time)	1.5 units per hour
Job time distribution	Fixed
μ (Service speed)	0.3 units per hour
h (Service time)	1 hour per day
P (Number of service providers)	50

Multi-period Munkres

For the special case when $\alpha = 1$ we propose the following utility distribution for the service providers participating in the matching mechanism.

Proposition 3: If $\frac{\tau}{\mu} > \frac{|P|}{\lambda}$ with deterministic arrival of service seekers, a constant job processing time τ and constant service rate μ, then the mean utility attained by service providers for a given matching period t over a total matching duration T is given by

$$\overline{EU_P(t)} = \frac{T}{\tau/\mu + slack} \tag{4.16}$$

where, $|P|$ is the number of service providers, λ is the rate of arrival of the service seeker. Slack is defined as $slack = (i+1)*t - \frac{\tau}{\mu}$ if i is a whole number such that $i*t \leq \frac{\tau}{\mu} < (i+1)t$.

Proof: We consider two separate cases:

Case 1: when $t \leq \frac{\tau}{\mu}$

Service provider needs $\left\lceil \frac{\tau}{\mu t} \right\rceil$ matching cycles to complete the assigned service task where $\lceil\rceil$ is the ceiling function

Thus, it takes $\left\lceil \frac{\tau}{\mu t} \right\rceil$ matching cycles for the service provider to be available again for matching

A matched service provider is available after $\left\lceil \frac{\tau}{\mu} \right\rceil$. It takes $\left\lceil \frac{|P|}{\lambda t} \right\rceil$ matching cycles for service seeker to outnumber the service providers

As $\left\lceil \frac{\tau}{\mu t} \right\rceil \geq \left\lceil \frac{|P|}{\lambda t} \right\rceil$ and $t \leq \frac{\tau}{\mu}$ a matched service provider gains a utility of 1 after a time period of $\frac{\tau}{\mu} + slack$.

Thus, in the total duration T the mean utility of service providers is given by $\overline{EU_P(t)} = \frac{T}{\tau/\mu + slack}$ when $t \leq \frac{\tau}{\mu}$

Case 2: when $t > \frac{\tau}{\mu}$

a matched service provider gains a utility of 1 after a time period of t

Thus, in the total duration T the mean utility of service providers is given by $\overline{EU_P(t)} = \frac{T}{t}$

When $t > \frac{\tau}{\mu}$, $t = \frac{\tau}{\mu} + slack$ as $i = 0$ by definition of $slack$

$\overline{EU_P(t)} = \frac{T}{\tau/\mu + slack}$ when $t > \frac{\tau}{\mu}$

Comparison of this theoretical prediction to numerical simulation of the scenario summarized in Table 4.2 is shown in Figure 4.17. The simulation results follow the theoretical curve proposed in Proposition 3. From the theorerical curve for service provider utility we see that the mean utility is the highest when $slack = 0$ as the parameters T, τ, and μ are characteristics of the target scenario. Only $slack$ depends on the period of matching. We have $slack = 0$ when matching period $\frac{\tau}{\mu}$ is perfectly divisible by the matching period t. This is also explained by that when matching period perfectly divides $\frac{\tau}{\mu}$, there is no waiting period after the completion of the assigned service task until the next match is received. In conclusion, the mean service provider utility attains the maximum value of $\frac{T}{\tau/\mu}$ for those matching period t that perfectly divides $\frac{\tau}{\mu}$. For $t > \frac{T}{\tau/\mu}$ the mean service provider utility hyperbolically drops down to 0.

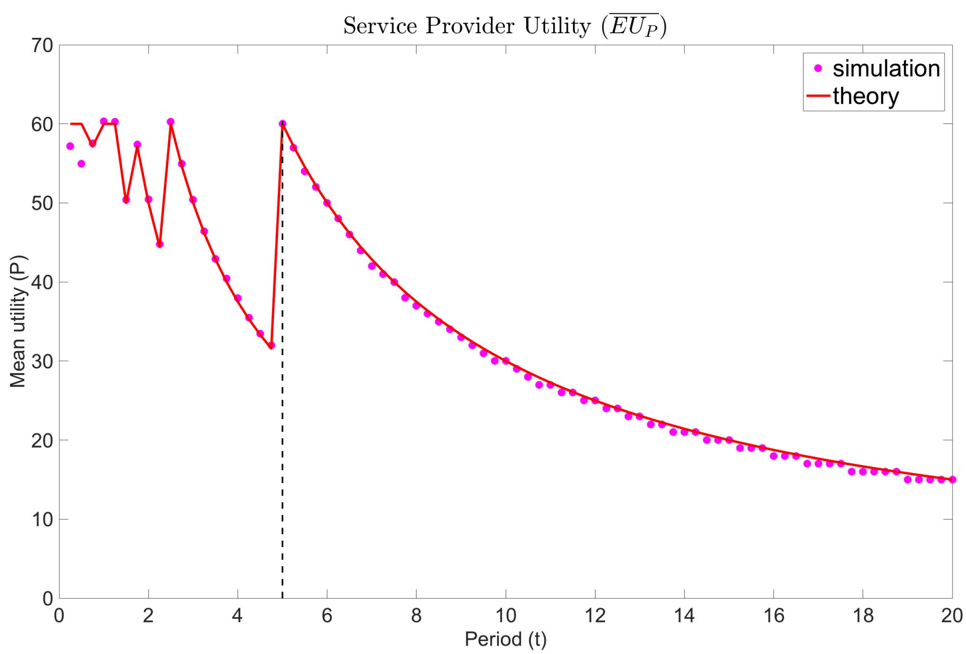


Figure 4.17. Comparison of theoretical prediction to simulation studies for utility attained by service providers as a function of matching period t under low service rate setting with all matched being perfectly compatible. The parameters used in simulation are the same as the ones tabulated in Table 4.1.

Figures 4.5.1 shows the mean utility attained by service providers and total utility attained by service seekers for various matching period under low service rate and perfect compatibility assumptions for the scenario summarized in Table 4.2. The mean service provider utility follows the curve described in Proposition 3. The total service seeker utility over the matching duration is given by $EU_P(t) = 50EU_S(t)$. The maximum occurs whenever the matching period t perfectly divides $\frac{\tau}{\mu}$. The number of matches follows the same curve as described in Proposition 3.

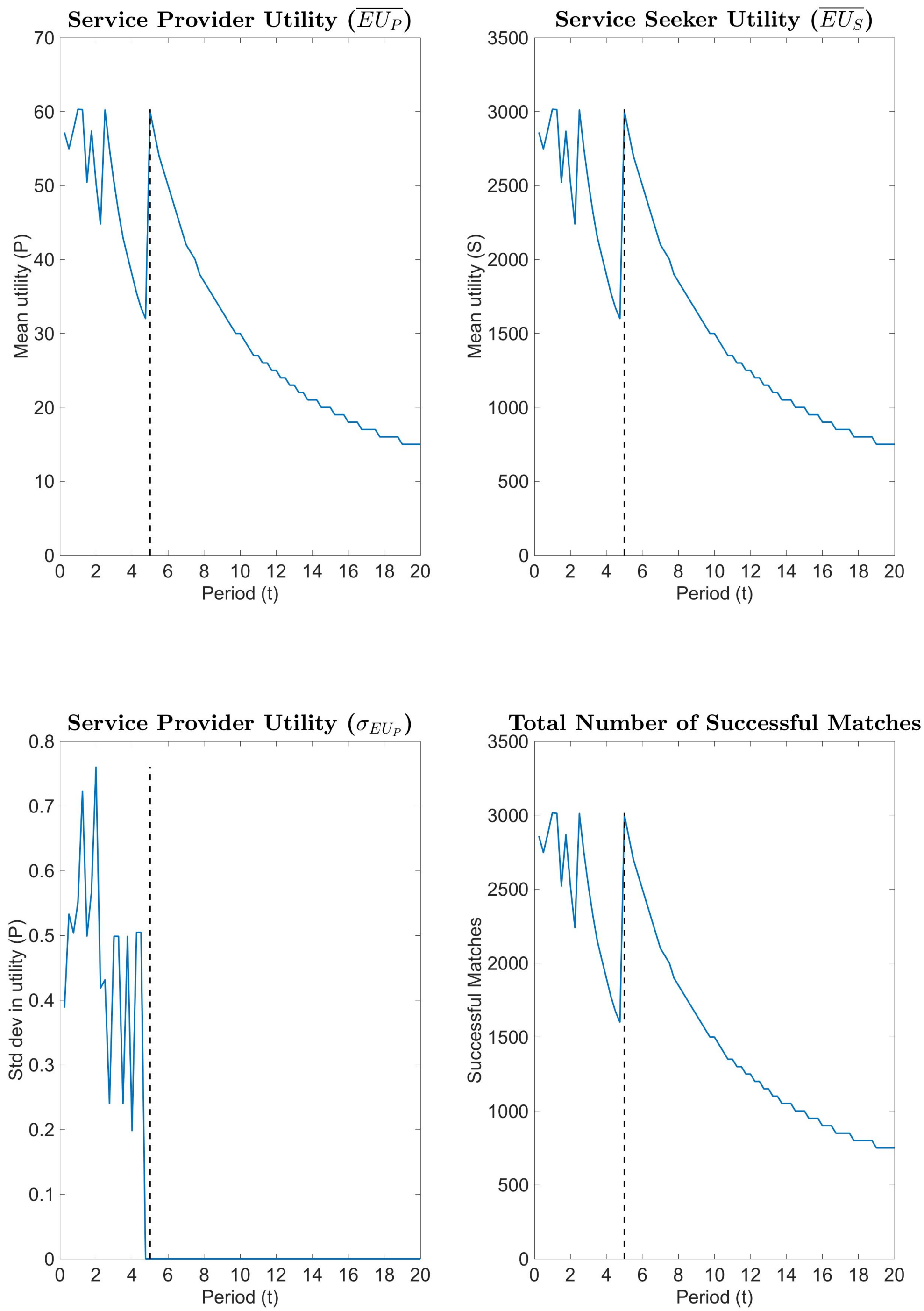


Figure 4.18. Effect of matching period on various matching objectives for multi-period Munkres mechanism under low service rate setting with perfectly compatible utility setting.

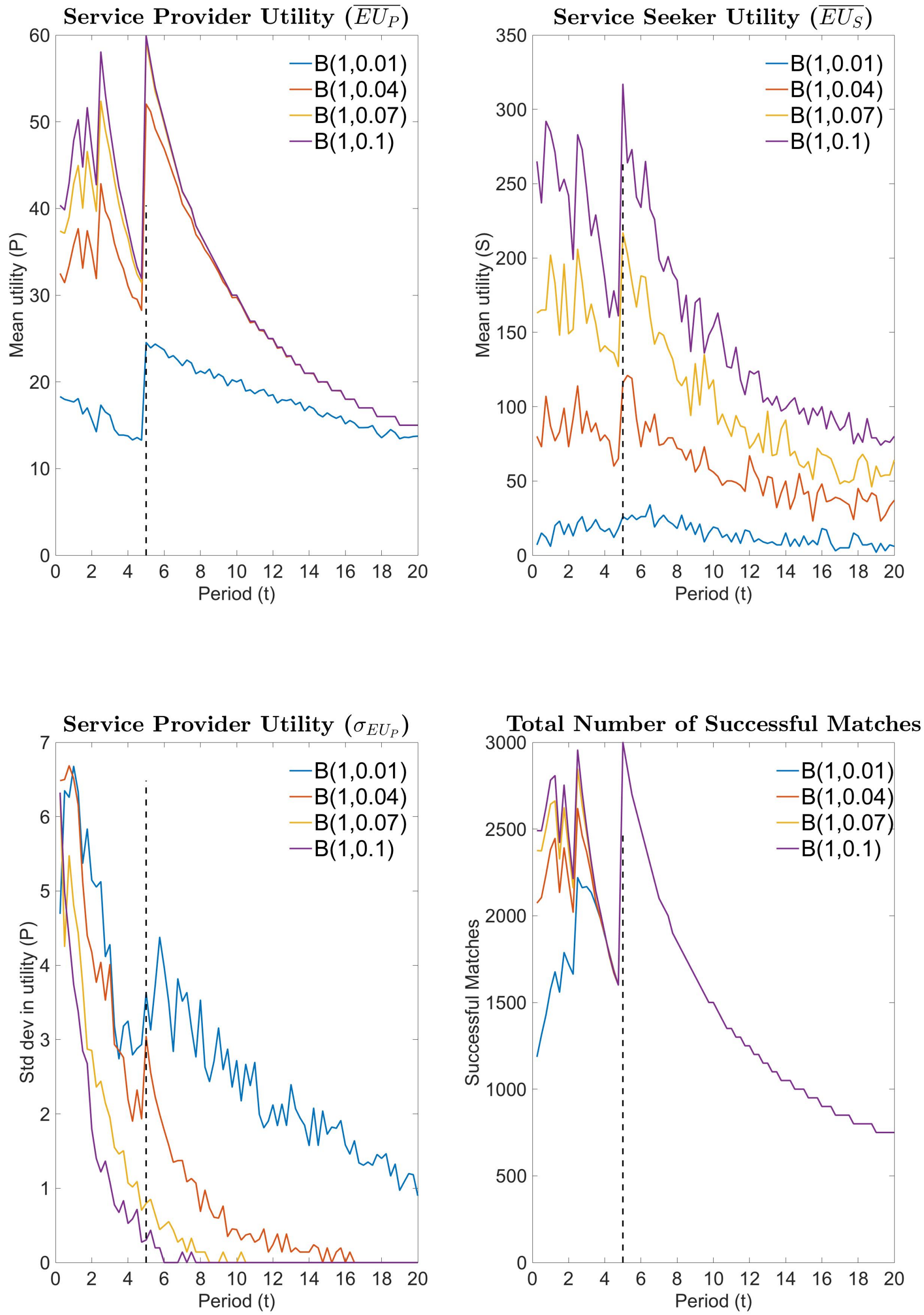


Figure 4.19. Effect of matching period on various matching objectives for various matching period of Munkres mechanism when service rate is low and utility follows a binomial distribution, $B(1, \alpha)$.

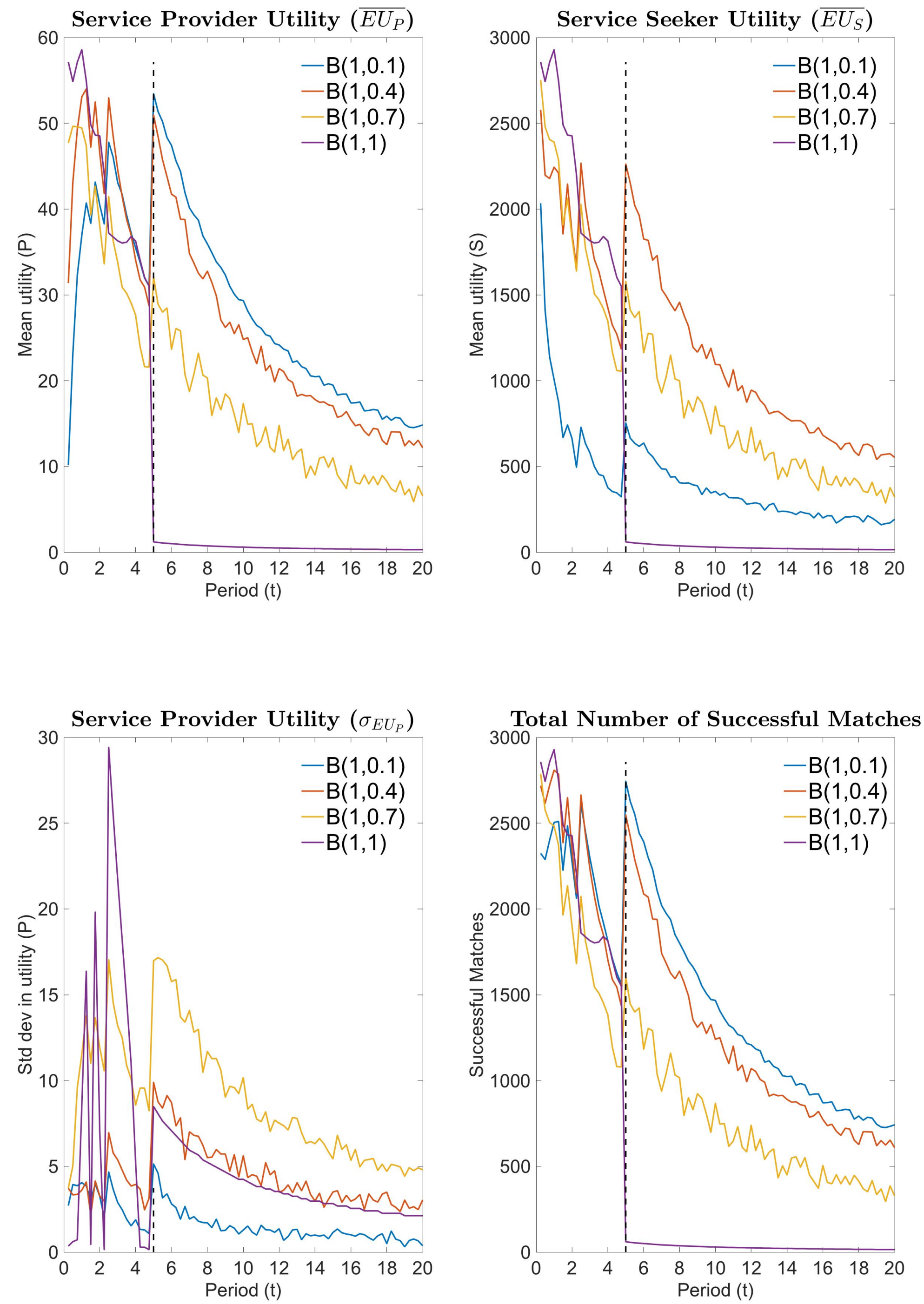


Figure 4.20. Effect of matching period on various matching objectives for various matching period of DA mechanism when service rate is low and utility follows a binomial distribution, $B(1, \alpha)$.

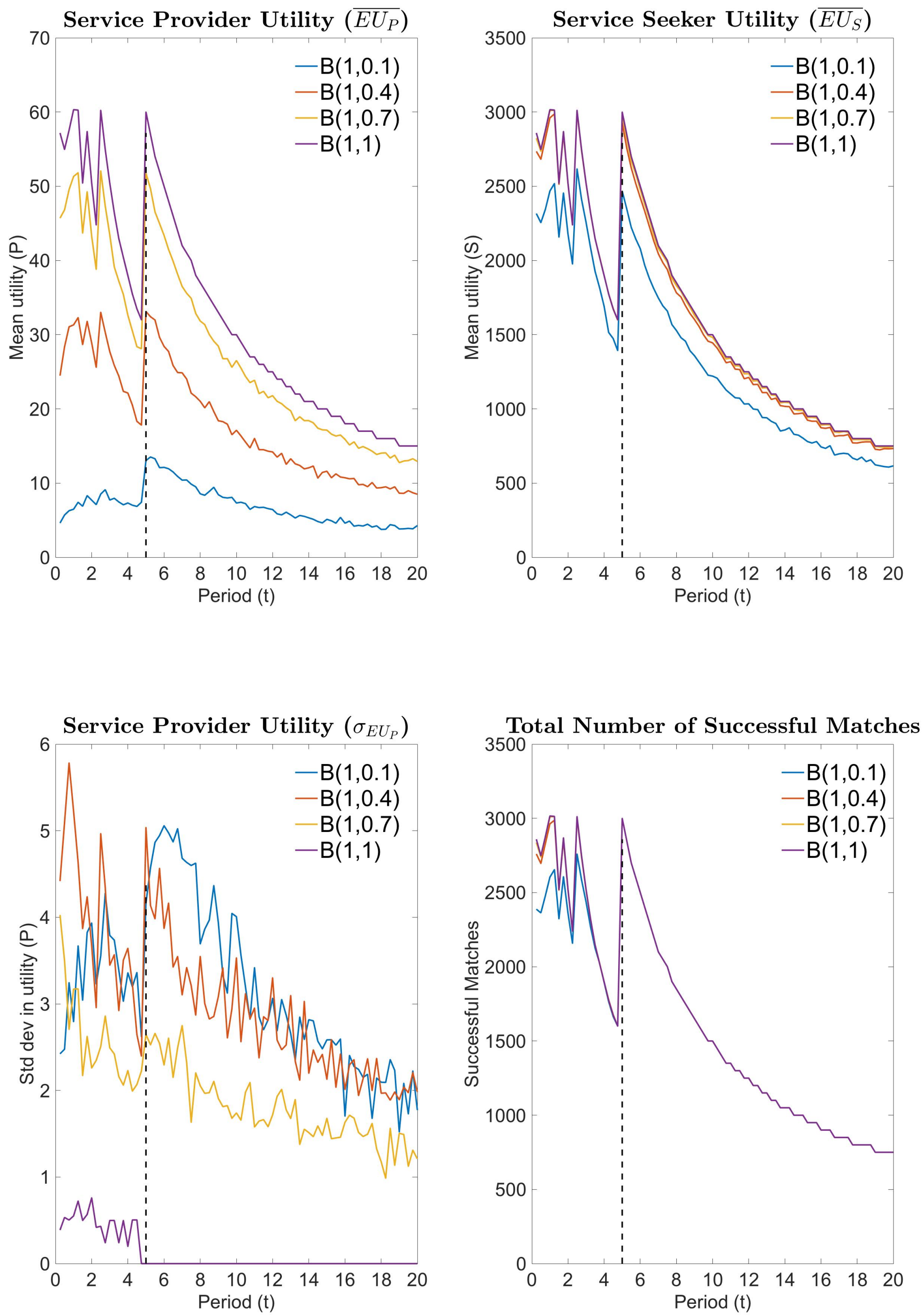


Figure 4.21. Effect of matching period on various matching objectives for various matching period of TTC mechanism when service rate is low and utility follows a binomial distribution, $B(1, \alpha)$.

4.5.2 Beta Distributed Utility

Now, in the low frequency setting we show the results of how the matching period affects the outcome of multi-period Munkres, DA, and TTC mechanism when utility of being matched follows a Beta distribution. The parameters used in the results present in this section are tabulated in Table 4.3.

Table 4.3.
Values of the parameters used for the simulation study for the results presented in this section

Parameter	**Value**
Utility	Beta(a, b)
Arrival Process	Deterministic
λ (Arrival rate)	12 per day
τ (Job time)	1.5 units per hour
Job time distribution	Fixed
μ (Service speed)	0.3 units per hour
h (Service time)	1 hour per day
P (Number of service providers)	50

When utility follows a beta distribution under low-service setting the effect of matching period on multi-period Munkres mechanism is similar to that of perfectly compatible setting under low-service rate. However in multi-period Munkres the seeker utility is maximum whenever matching period t perfectly divides $\frac{\tau}{\mu}$. In contrast to that, in multi-period DA whenever t perfectly divides $\frac{\tau}{\mu}$ there is a local maximum of the utility but is lower than the global maximum that occurs when $t = \frac{\tau}{\mu}$. This is because of the effect similar to the one observed in high service rate setting as explained in Section 4.4.2.

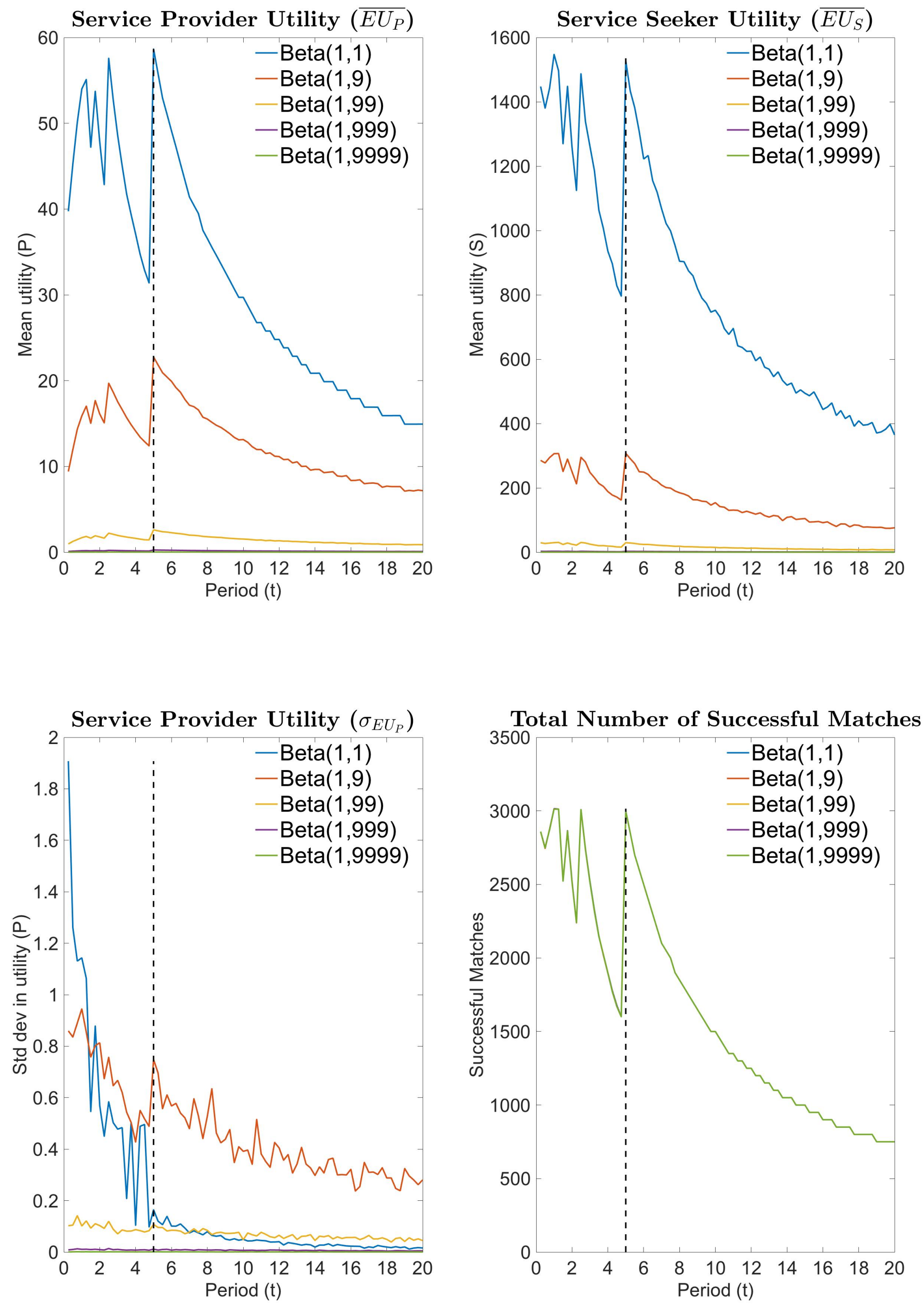


Figure 4.22. Effect of matching period on various matching objectives for various matching period of multi-period Munkres mechanism when service rate is low and utility follows a beta distribution, $Beta(a, b)$. The parameters used in this simulation of these results are summarized in Table 4.3.

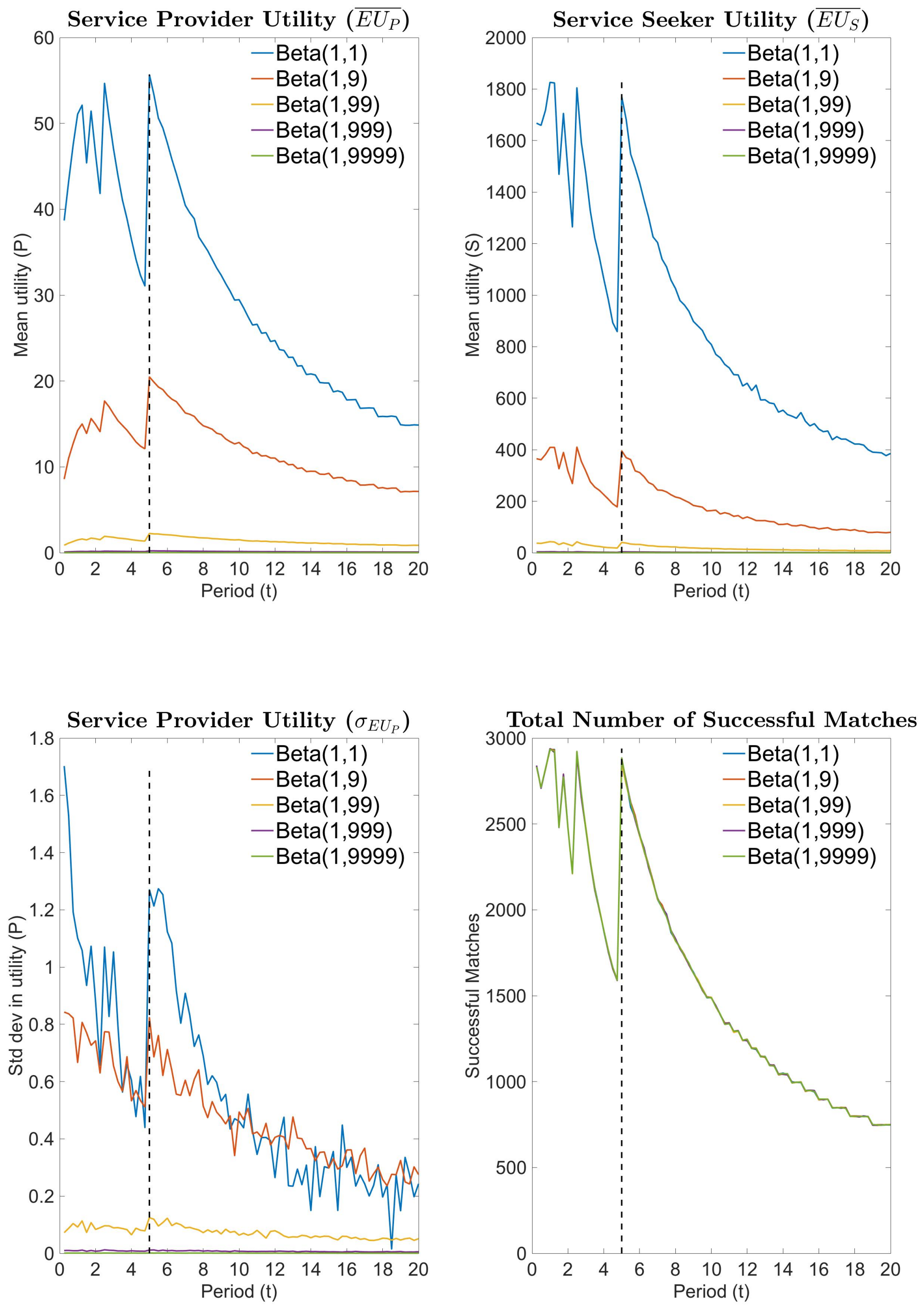


Figure 4.23. Effect of matching period on various matching objectives for various matching period of multi-period DA mechanism when service rate is low and utility follows a beta distribution, $Beta(a, b)$.

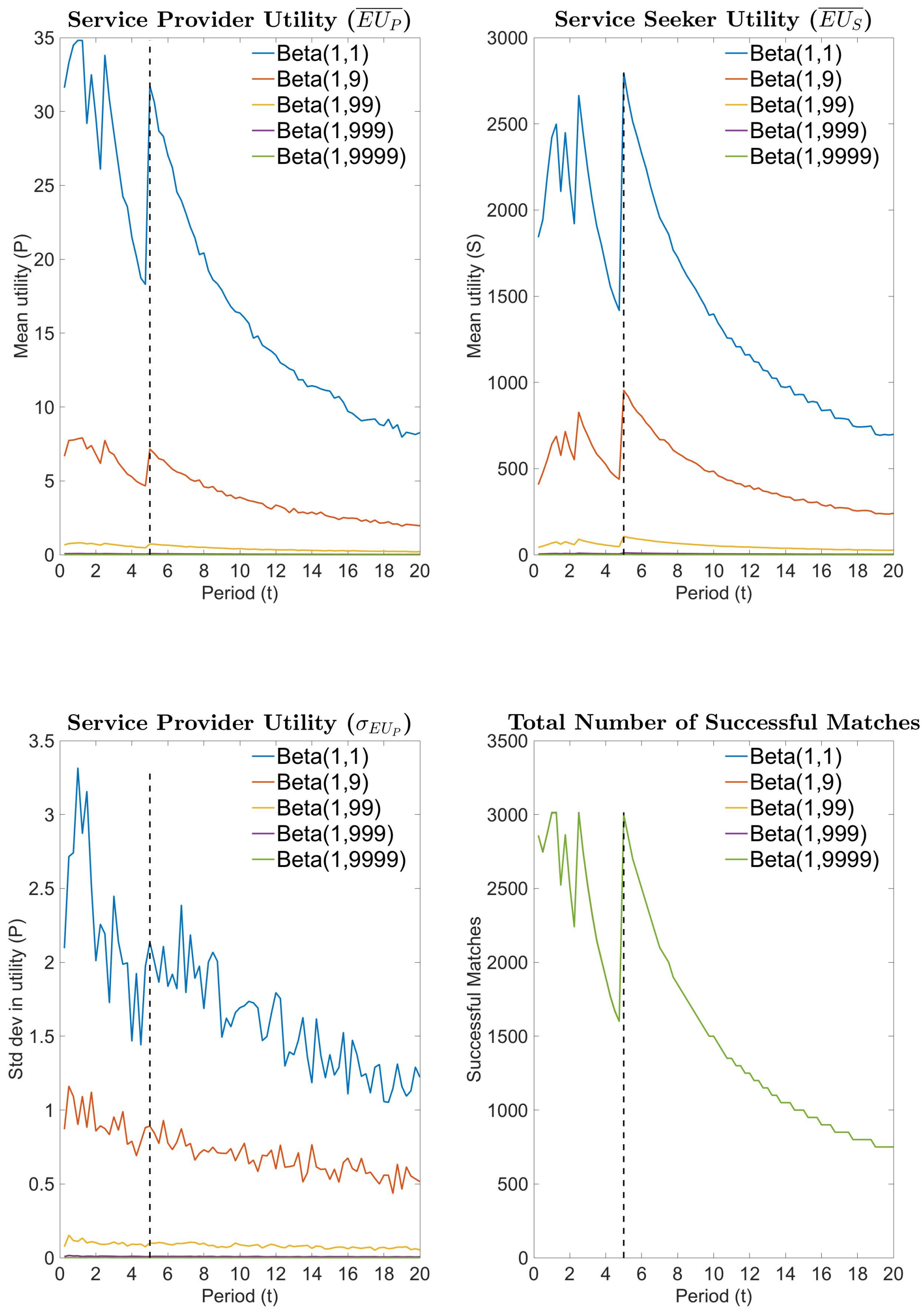


Figure 4.24. Effect of matching period on various matching objectives for various matching period of multi-period TTC mechanism when service rate is low and utility follows a beta distribution, $Beta(a, b)$.

4.6 Conclusions

In high service rate setting, i.e. when we have $\frac{|P|}{\lambda} \geq \frac{\tau}{\mu}$, with a deterministic arrival of service seekers, all matches being perfectly compatible and constant job processing time, the optimal period of matching is $\frac{P}{\lambda}$. For any matching period in the range $0 < t \leq \frac{|P|}{\lambda}$ the Munkres mechanism yields the same maximum utility for both service seekers and service providers in high service setting. It also yields the same maximum number of successful matches in this range of matching period. For $t > \frac{|P|}{\lambda}$, the utility and the number of successful matches hyperbolically decreases. Therefore, any matching period in this range is optimal for a Munkres mechanism considering the number of successful matches and the utility as the objective for optimization. But from the point of consideration of fairness, which is quantified by the standard deviation in the distribution of utility among service providers, the most optimal matching period is when $t = \frac{|P|}{\lambda}$. For $t < \frac{|P|}{\lambda}$ the standard distribution is higher than when $t = \frac{|P|}{\lambda}$. The standard deviation decreases as t increases from 0 to $\frac{|P|}{\lambda}$. Therefore considering all three objective the most optimal period to implement the matching mechanism is when $t = \frac{|P|}{\lambda}$.

If the assumption of perfectly compatible matches is relaxed, the optimal period is still unchanged. When the matches are perfectly compatible the utility follows the same curve as the number of successful matches. Now, if we relax the assumption of perfect compatibility to the utility being drawn from a beta distribution only the expected utility from the match changes. The number of matches is same as proposed.

For TTC and DA mechanism the same results are obtained for the optimal period of matching. In DA mechanism an additional observation is that the utility and successful increases (instead of staying constant at the maximum value) in the range $[0, \frac{|P|}{\lambda}]$. This is because DA always produces stable outcomes. This limits the set of possible outcomes only into a small regime of stable solutions. In Munkres mechanism any bipartite matched output is possible. The set of stable solution increases exponentially as the set of service seekers to choose from increases. This is why when matching period is increases DA has an additional effect of increasing the utility. The design point is unchanged. The optimal period continues to be $\frac{|P|}{\lambda}$.

Under low service rate setting, i.e. when we have $\frac{|P|}{\lambda} < \frac{\tau}{\mu}$, with a deterministic arrival of service seekers, all matches being perfectly compatible and constant job processing time, the optimal period of matching is $\frac{\tau}{\mu}$. The mean utility attained by service providers for a given matching period t over a total matching duration T is given by $\overline{EU_P(t)} = \frac{T}{\tau/\mu + slack}$, Slack is defined as $slack = (i+1) * t - \frac{\tau}{\mu}$ if i is a whole number such that $i * t \leq \frac{\tau}{\mu} < (i+1)t$. Whenever the matching period is a perfect divisor of $\frac{\tau}{\mu}$ we have $slack = 0$. At this matching period the total number of successful matches, the mean service seeker utility, and the total service provider utility is maximized. Similar to the high service rate setting the fairness increases as matching period increases from 0 to $\frac{\tau}{\mu}$. Thus considering all the objectives the optimal matching period is $\frac{\tau}{\mu}$. These results extend to both DA and Munkres mechanism under any utility distribution assumptions.

5. GENERALIZING OPTIMAL MATCHING PERIOD IN REAL-WORLD APPLICATIONS

There are three primary assumptions that were made for the results presented in Section 4.4 and 4.5 in Chapter 4. These were: a) the arrival process was deterministic with a constant arrival rate, b) the job time for processing the service request was constant, and c) the utility of being matched was drawn from standard distribution like Binomial and Beta. In this chapter we discusses the effect of relaxing these assumptions on the optimal matching period recommended in the previous chapter. In Section 5.1 we discuss the effect of relaxing the assumption of deterministic arrival and constant job processing time. In Section 5.2 we simulate an illustrative decentralized design and manufacturing scenario relaxing all the assumptions considered including the utility of being matched following standard probability distributions. We shows how the results presented generalizes to the illustrative simulated scenario. Finally, Section 5.3 presents the concluding remarks.

5.1 Effects of Relaxing Some of the Idealistic Assumptions

In this section we relax the assumption of deterministic arrival and constant job processing time. Instead of a deterministic arrival process, we model it as a Poisson process and study the effect of that on the optimal matching period in Section 5.1.1. In Section 5.1.2 we model the job processing time as exponentially distributed among the population of designers and study its effects on the optimal matching period.

5.1.1 Effects of Relaxing the Assumption of Deterministic Arrival

In Chapter 4 we assumed that the arrival of service seekers are deterministic with a constant arrival rate. However, in practice the arrival pattern is not deterministic.

In a totally decentralized setting, the service seekers arrive from a wide range of independent sources. The arrival time of a service request from a service seeker is independent from the arrival time of service requests from other service seekers. A Poisson process better models the arrival pattern in practice than a deterministic approximation.

Figure 5.1 shows the effect of Poisson arrival in high service rate setting. The parameters used for the results presented in Figure 5.1 are the same as in Table 4.1 for the Deterministic and Poisson simulation curves. The general behavior of the effect of matching period on the matched outcome is unchanged from deterministic to Poisson arrival process. The total service seeker and mean service provider utility remains constant for $t \leq \frac{|P|}{\lambda}$ after which it hyperbolically drops down and the standard deviation in their distribution drops drastically as t increases from 0 to $\frac{|P|}{\lambda}$. The Poisson arrival only causes some fluctuation around the mean deterministic behavior. These fluctuations are more prominent when $t \leq \frac{|P|}{\lambda}$. This is because the main effect of a Poisson arrival is the variation in the number of service seekers to choose from in a matching cycle. When $t > \frac{|P|}{\lambda}$ there are already more service seekers than the available service providers in expectation and the fluctuation due to a Poisson arrival is not much evident in the matched outcome. To verify this effect further, Gaussian process regression was performed on the observed data for which the results are presented in Figure 5.2. From the deterministic case, we know that the matching period has entirely different effects in the range $0 < t \leq \frac{|P|}{\lambda}$ and when $t > \frac{|P|}{\lambda}$ two separate Gaussian regressions were performed on the observed data in these regions. The regression yields a straight line as in the case of deterministic arrival process in the region $0 < t \leq \frac{|P|}{\lambda}$ and it yields a hyperbolically decaying curve for $t > \frac{|P|}{\lambda}$. When averaged over 10 samples the uncertainty band diminishes over a single-sample simulation. Moreover, when averaged over 10 samples the mean curve of the Gaussian regression shifts closes to the deterministic curve as observed from Figures 5.2a and 5.2b.

Figure 5.3 repeats the simulation under low service rate setting. The parameters used in the deterministic and Poisson simulation are the ones tabulated in Table 4.2. Similar to the high service rate setting, the Poisson arrival only adds a fluctuation around the deterministic arrival process. The mean effects produced by a Poisson arrival are the same as deterministic process.

Therefore, in conclusion for both high and low service rate setting Poisson arrival process only causes some fluctuation around the deterministic deductions of the effects of matching period. The optimal matching period concluded for deterministic arrival is unchanged for high and low service rate setting even if the arrival process transitions to Poisson.

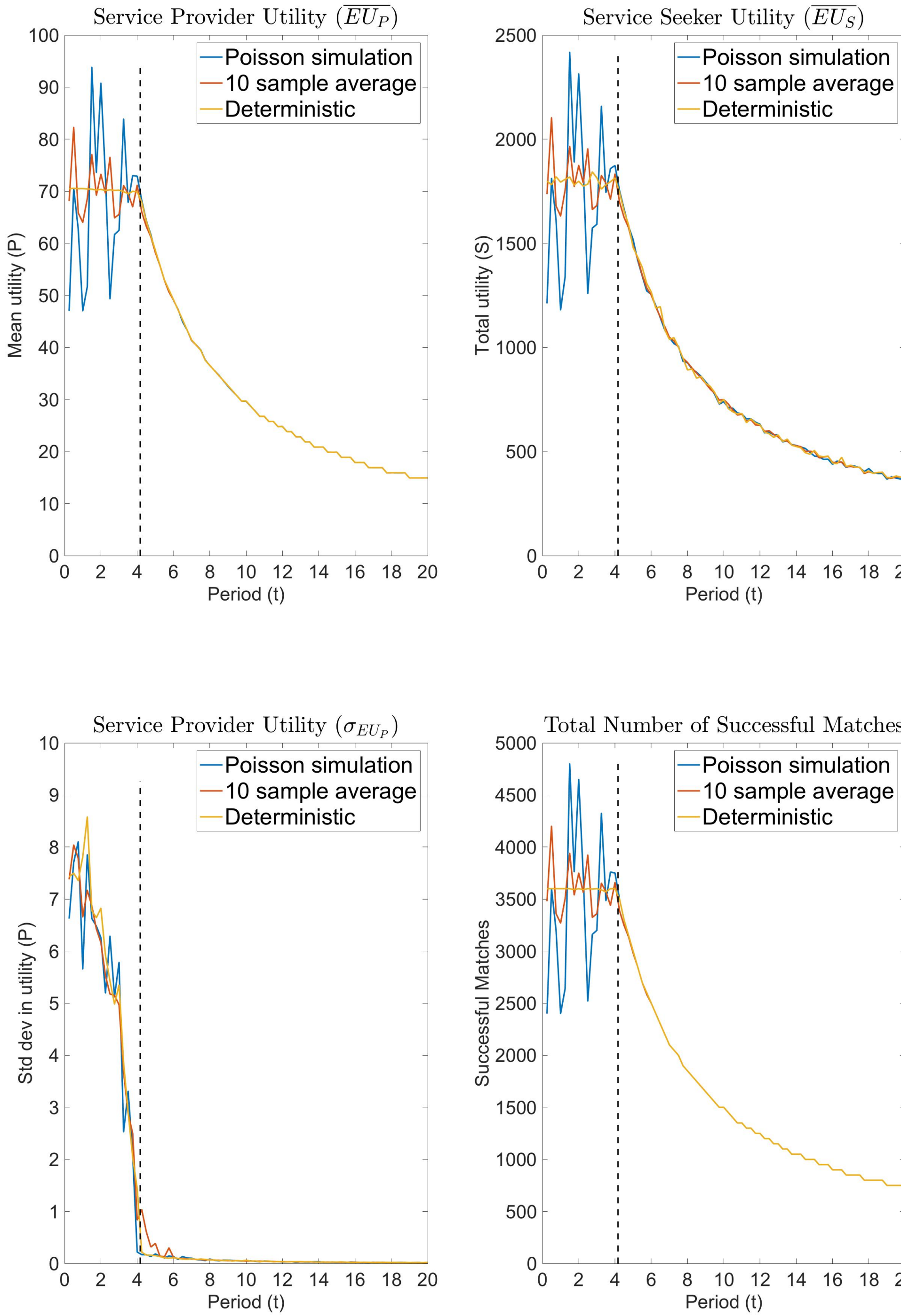
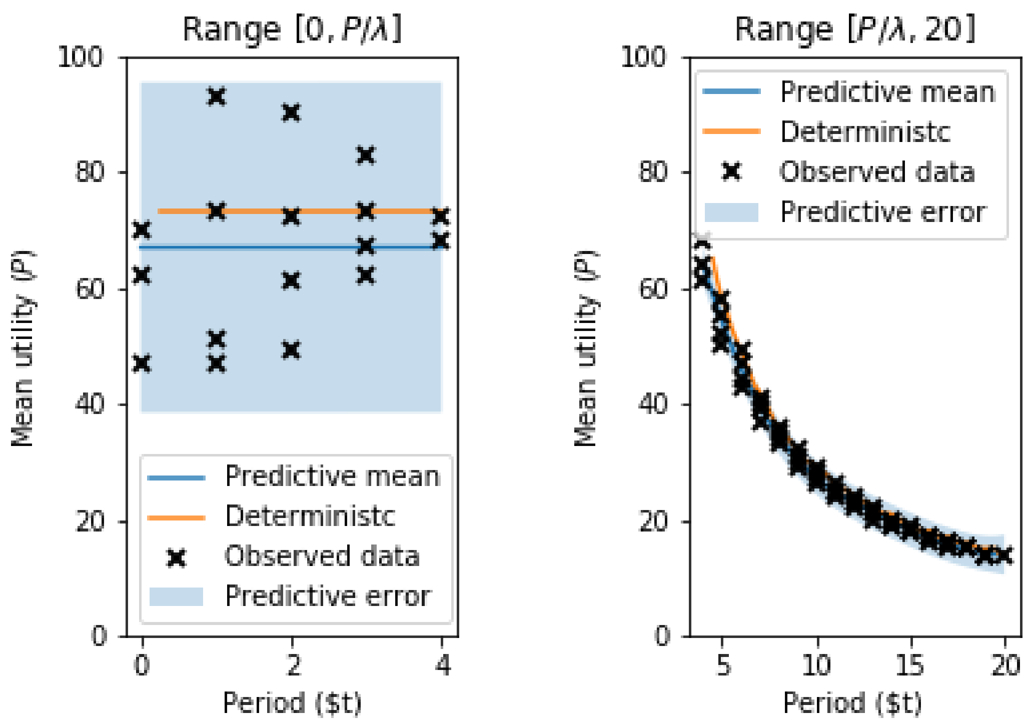


Figure 5.1. Comparison of Poisson arrival to deterministic arrival for multi-period Munkres mechanism under high service rate setting.

(a)



(b)

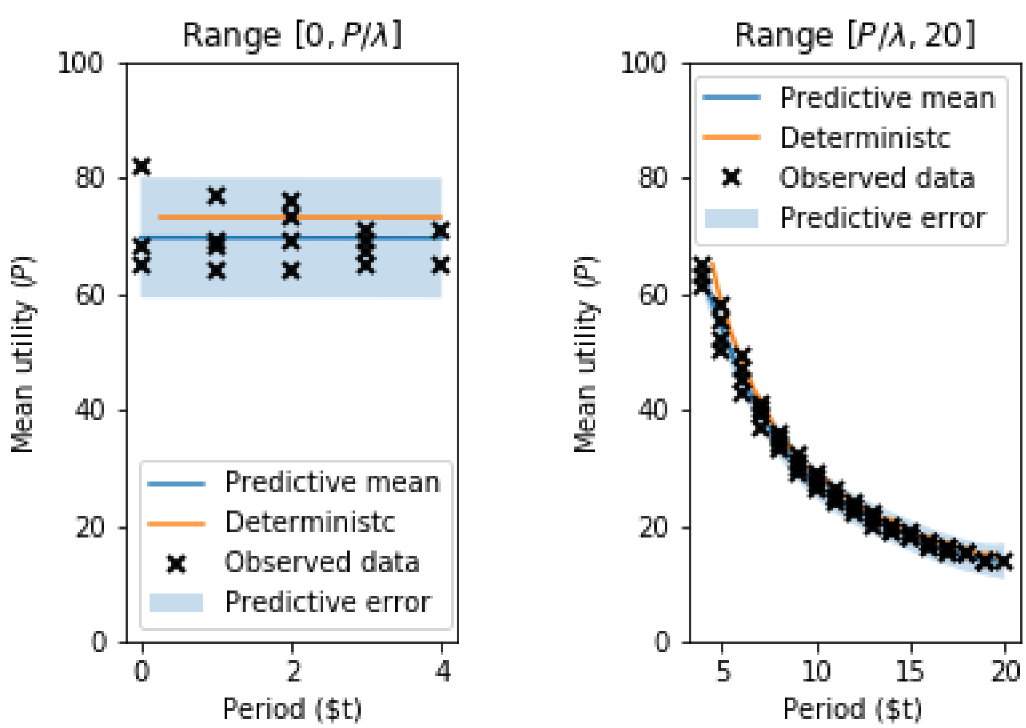


Figure 5.2. (a) Gaussian process regression applied to Poisson arrival. (b) Gaussian process regression applied to 10 sample average of Poisson arrival.

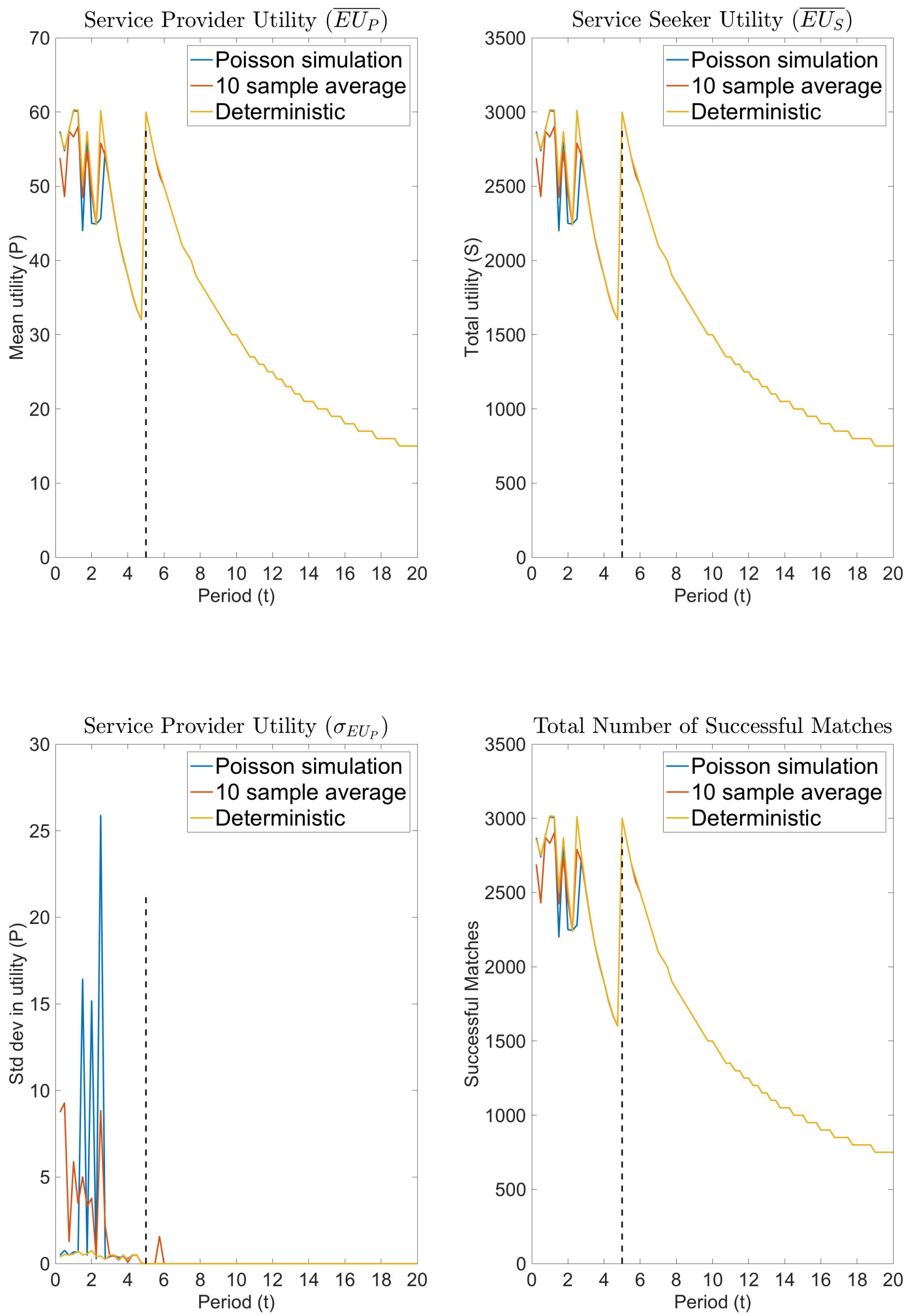


Figure 5.3. Comparison of Poisson arrival to deterministic arrival for multi-period Munkres mechanism under low service rate setting with utility following a perfectly compatible setting.

5.1.2 Effects of Relaxing the Assumption of Uniform Job Processing Time

Previously we had assumed that the job processing time is constant. In real applications it is not constant and from practice it is observed that the job processing time is exponentially distributed in the population. This is because of a characteristic of a general population that a large number of service requests will have relatively small processing time, but at the same time there will be quite a few number of service requests with extremely large processing time. The fraction of service requests above a threshold processing time exponentially decays as the threshold increases. From the data we had collected this exponential trend was observed which is discussed in further detail in Section 5.2.2.

In Figures 5.4 and 5.5 the effect of job processing time being exponentially distributed is shown. Figure 5.4 shows the effect on high service rate setting. For this particular comparison the utility is drawn from standard uniform distribution. For this particular simulation, the parameters used in the simulation are same as the ones tabulated in Table 4.1 with only difference being job processing time distributed exponentially instead of fixed and the utility is drawn from standard uniform distribution. The mean of the exponential distribution is 1.5 (same as the fixed processing time constant). Figure 5.5 shows the effect on low service rate setting. The parameters used in Figure 5.5 are the same as the ones tabulated in Table 4.2.

From the comparison between fixed and exponential distribution we observe that under high service rate setting the optimal matching period is unchanged even if the job processing time is exponentially distributed. But, when the service rate is low the optimal matching period for exponentially distributed job processing time is different from the fixed one. As matching period t increases both mean service provider utility, total service seeker utility, and number of successful matches decreases monotonically. This is because for any given population since the processing time is exponentially distributed there are large number of service requests with extremely small processing time and decreasing the length of matching cycles increases the total number of matches performed during the entire mechanism implementation duration T. Therefore from the perspective of optimizing the utility or number of successful matches, lower the matching period t the better. The standard deviation in the distribution of utility also monotonically decreases as matching period t increases. This is because when matching period t is low the service provider who got matched to jobs that needed lower processing time gets matched in a higher number of cycles thereby giving them a higher overall utility in the entire match duration T. This effect becomes less pronounced as matching period t increases. So from the perspective of optimizing the fairness in the distribution of utility among the agents higher the matching period t the better.

In conclusion, there is no optimal matching period considering all the matching objectives when service rate is low and printing time is exponentially distributed. There is a tradeoff between utility gained and fairness in their distribution as matching period t is varied. The optimal matching period is application specific and needs to be designed based on the specific objectives of the target application. The mechanism designer needs to design the matching period based on which objective is more important whether the cumulative utility or fairness in their distribution.

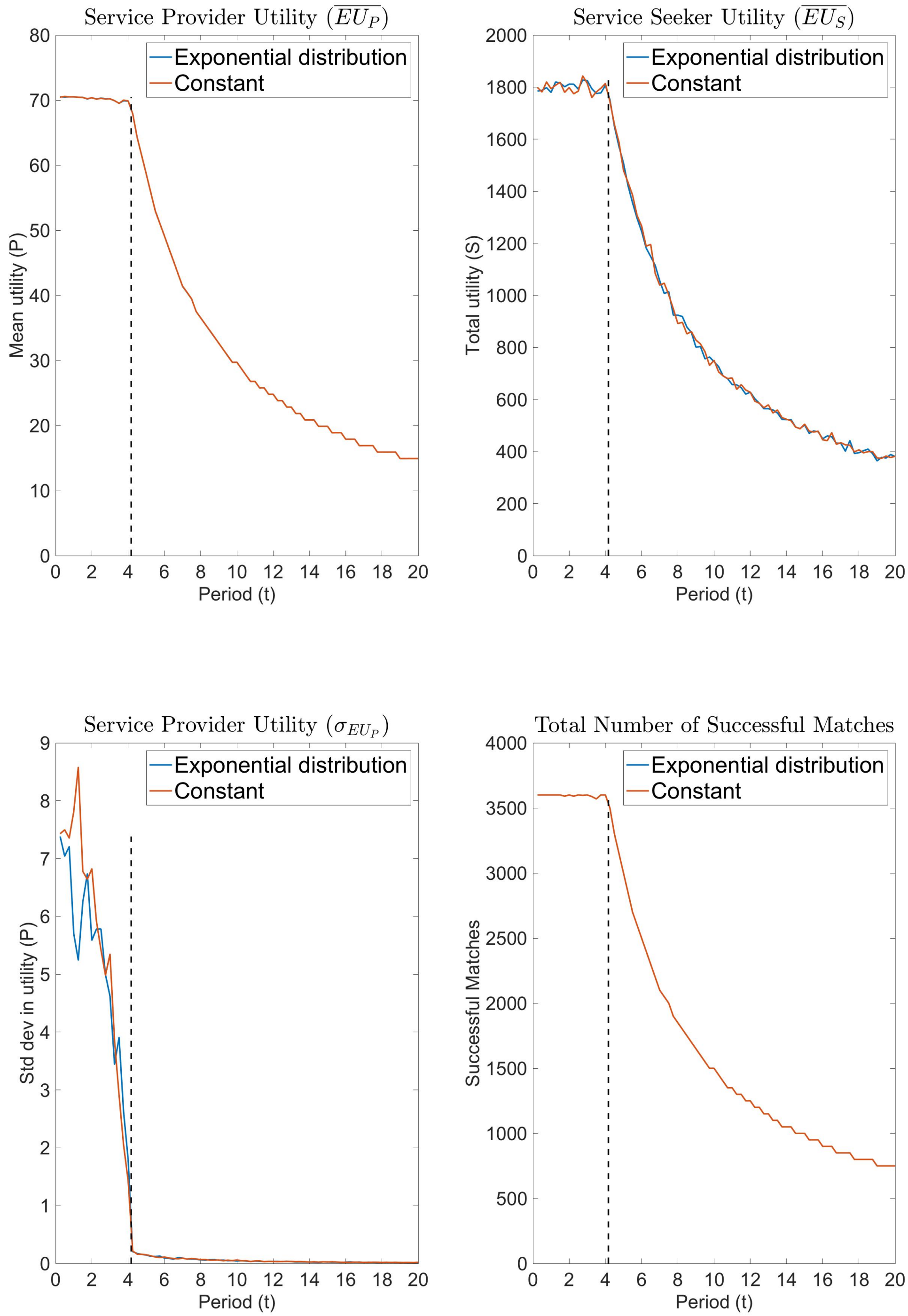


Figure 5.4. Comparison of exponential distribution of job processing time and fixed job processing time for multi-period Munkres mechanism under high service rate setting with utility drawn from standard uniform distribution.

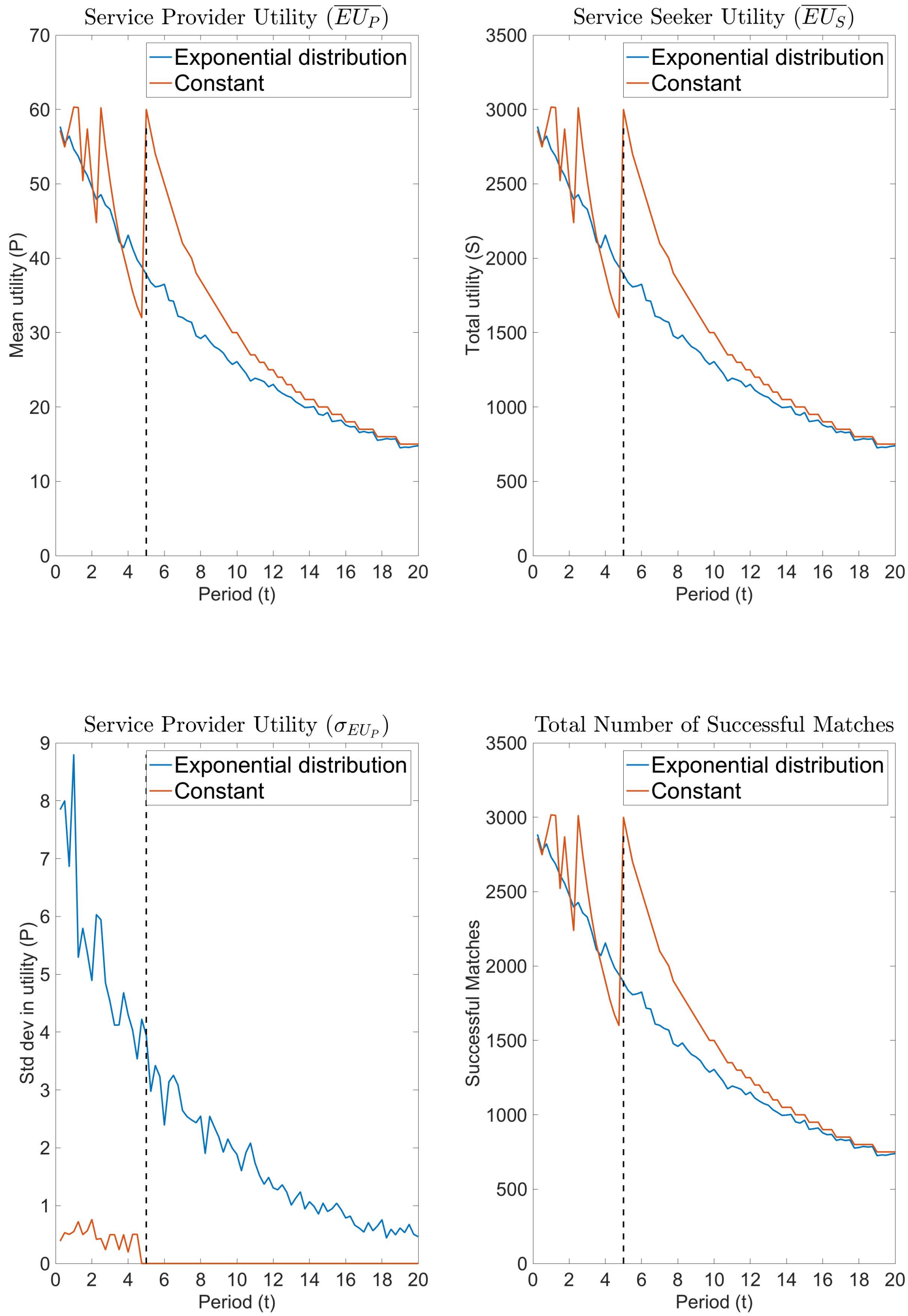


Figure 5.5. Comparison of exponential distribution of job processing time and fixed job processing time for multi-period Munkres mechanism under low service rate setting in a perfectly compatible utility setting.

5.2 Simulation Studies of an Illustrative Decentralized Manufacturing Scenario

We consider an illustrative scenario where 50 independent 3D-printer machine owners are offering manufacturing services. Service seekers are designers from a population who are trying to get their designs prototyped in the 3D printers. The service providers and service seekers are referred to as manufacturers and designers respectively. The arrival of designers is modeled as a Poisson process. The manufacturer p_j offer h_j working hours per day. Matching is done after every t days. We assume that each manufacturer can be matched to at most one designer in a matching cycle. This is not a limitation of the model but has been assumed for analysis purposes.

5.2.1 Data Collection

The manufacturers' attributes considered are machine volume, machine resolution (Res), the tensile strength (TS) of the material offered, manufacturer proximity whereas designer attributes were printing time, material requirement, and design dimensions (Vol). To generate the attributes of the designers, 100 different designs are downloaded from Thingiverse [38] and their characteristics such as design dimensions, printing time required in different 3D printers are recorded. Some of the sample designs used are shown in Figure 5.6. Attributes of a large sample size of designers are generated from these recorded attributes. Manufacturer data concerning the machine attributes are collected from the Senvol [54] database. The machine search mode on the database is used for searching machine features. Material properties of the material used in these 3D printers are collected from iMaterialise [39]. 50 unique material machine combinations are used to define the attributes of 50 manufacturers. Ranges of attributes of the designs used in the simulation studies are summarized in Table 5.1.

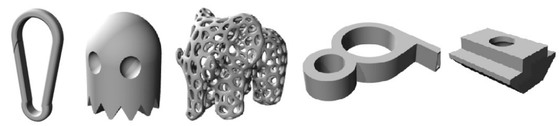


Figure 5.6. Samples of designs used in the simulation studies.

Figure 5.7. Examples of some of the 3D printers used in the simulation studies.

Table 5.1.
Range of values used for the attributes in the simulation studies.

Attribute	Area (in^2)	Vol (in^3)	Res (mm)	TS (MPa)
min	3.4	4.5	0.01	14
max	279.3	67875.4	1	1800

5.2.2 Setting Parameters of the Simulation Studies

The working time (h) denotes the availability of the manufacturers on a day and this in combination with working speed determined the service rate of the manufacturer. The working speed depends on the 3D printing process. For example Stereolithography (SLA) is faster than Fusion Deposition Modeling (FDM). Figure 5.2.2 compares the printing time for the 100 designs downloaded from Thingiverse [38] on MakerBot and Form 1+ machines. As expected the printing time on each machine is exponentially distributed across the designer population with the mean characteristic of the machine. For example in MakerBot machine the printing time for the

100 designs followed an exponential distribution with mean printing time 5.77 hours. Form 1+ had a lower mean printing time of 1.73 hours as compared to the 5.77 hours of MakerBot as Form 1+ uses SLA process and MakerBot uses relatively more time consuming FDM process. For five different machines the printing time calculation was made using Cura [55] software after which an exponential curve was fitted. The results of the parameter of the exponential curve on seven different machines are tabulated in Table 5.2. It is not possible to calculate printing time for several tens of thousands of designs on 50 different machines. Therefore, to simulate the characteristic of a representative decentralized design and manufacturing population the following sampling technique was used. The upper 95% confidence bound on the mean printing time for the 100 designs on the slowest machine (Makerbot Replicator 2) was 7.17 hours and the lower 95% confidence bound on the fastest machine (Form 1+) was 1.39 as observed from Table 5.2. Other mean printing time was in between these two bounds. 50 samples were drawn from a uniform distribution between 1.5 and 7.5 to represent mean printing time for 50 different machines. For example $\overline{\tau_j} \sim Uniform(1.5, 7.5)$ denotes the mean printing time of manufacturer p_j. For the designer s_i being matched to the service provider p_j, the printing time is generated from an exponential distribution with parameter being the mean printing time of service provider p_j i.e $\tau_{ij} \sim exp(\overline{\tau_j})$.

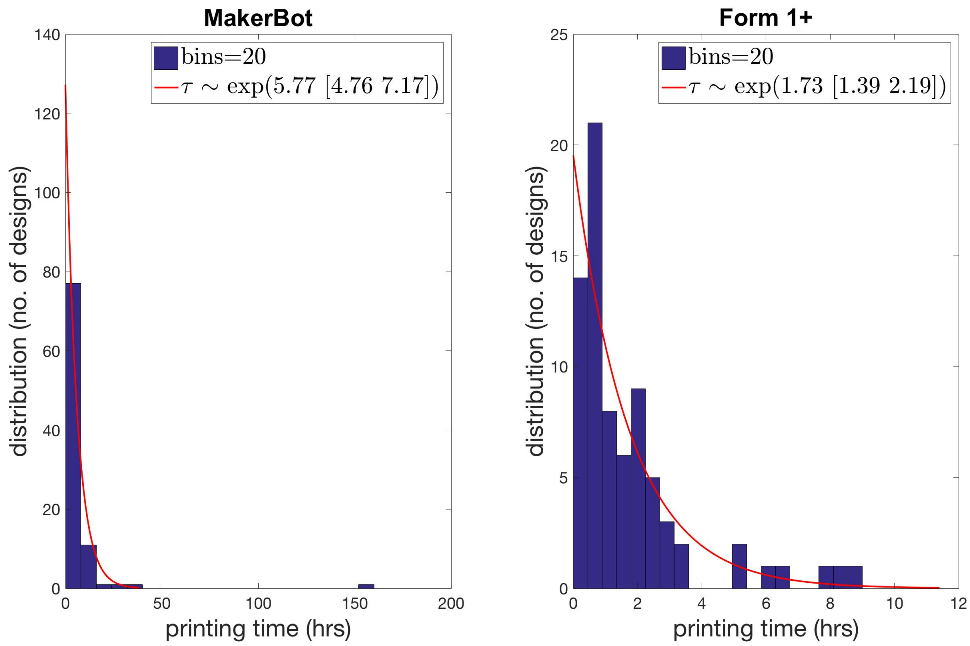


Figure 5.8. Printing time for 100 designs downloaded from Thingiverse on MakerBot and Form 1+ as calculated using Cura [55] software.

Table 5.2.
Parameters of the exponential distribution of printing time of the 100 designs on seven different machines

Machine	**Mean (hours)**	**95 % CI**
Makerbot Replicator 2	5.77	$[4.76, 7.17]$
Form 1+	1.73	$[1.39, 2.19]$
Ultimaker 2	4.49	$[3.71, 5.55]$
Lulzbot TAZ 4	6.19	$[5.14, 7.60]$
Witbox	6.05	$[5.02, 7.44]$
Prusa Mendel i3	4.53	$[3.75, 5.58]$
B9 Creator	1.98	$[1.56, 2.60]$

For dimensions like area and volume similar sampling technique was used based on the parameters calculated using the 100 representative designs as these attributes are also exponentially distributed in a random population. Working hours was set to $h = 5$ hours per day. Without loss of generality, service rate, $\mu = 5$ for all manufacturers. Service rate is a combination of the effect of working hours and service speed. But, however the effect of service speed was already incorporated into the printing time while sampling using different parameters based on the service speed of each service provider.

The mean printing time of the selected designs on the selected machines for simulation studies is approximately 5.99 hours. The efficiency of the mechanism is assessed over a duration of 30 days ($T = 30$) since this was sufficiently long compared to the average job processing duration ($\overline{\tau} = 5.99$ hours). The arrival process of service seekers is assumed to be distributed as Poisson with a mean arrival rate of 12 designs per day ($\lambda = 12$). The matching when period (t) is varied from 0.25 days to 20 days in increment of 0.25 days.

To simulate the distribution of utility, we began by first calculating the expected utility that manufacturers gain being matched to designers and vice-versa for the 100 representative designs and the 7 representative machines using the procecure discussed in Section 2.3.1 in Chapter 2. From these calculations we found that in between 90% and 100% of the designs in the population could be processed in these machines. Incompatibility arose primarily due to volume and build area dimension constraints. The utility calculated for the compatible matches was used to obtain the parameters of the beta distribution for the individual machines. Superimposing the binomial distribution over the beta distribution the utility of individual designer to manufacturer matches were obtained. These distributions were heterogeneous among the service providers resulting in preferential treatment of certain service providers over others. For example, the utility for being matched to one of the service provider was generated from $Beta(1, 0.9)$ whereas for another provider it was $Beta(1, 0.1)$ resulting in preferential treatment of the former.

All the above conditions used in the simulation study is summarized in Table 5.3.

Table 5.3.
Parameters and assumptions used in the CBDM illustrative scenario simulation.

Parameter	**Value**
Utility	Generated from Data
Arrival Process	Poisson
λ (Arrival rate)	12 per day
$\overline{tau_j}$ (Mean Job time for p_j)	Uniform(1.5,7.5) hours
Job time distribution	Exponential
μ (Service speed)	5 units per hour
h (Service time)	5 hours per day
$\lvert P \rvert$ (Number of service providers)	50

5.2.3 Results of Simulation Studies

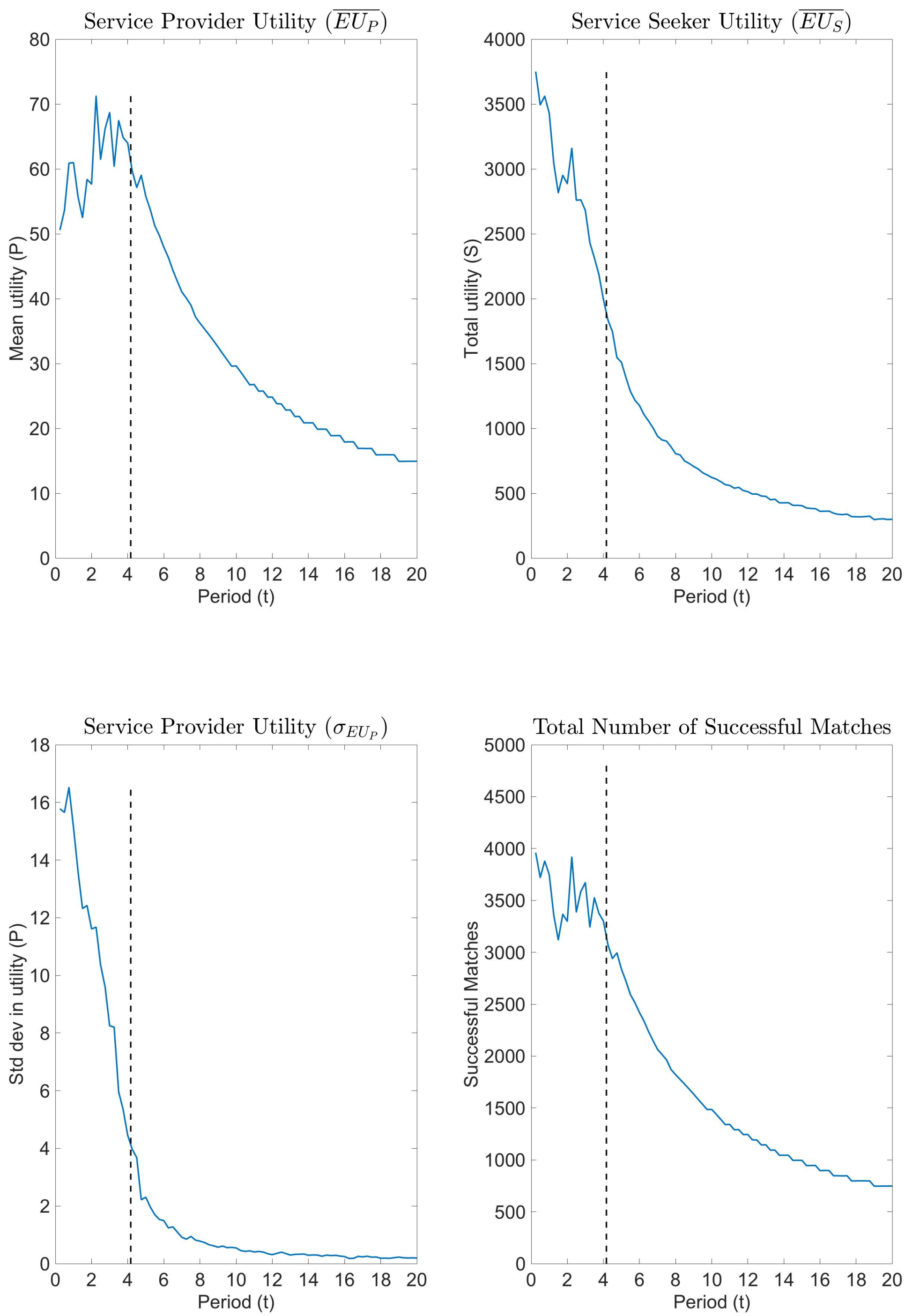


Figure 5.9. Results for effect of matching period on various matching objectives for the illustrative CBDM scenario.

Figure 5.9 shows the results for mean service provider utility, total service seeker utility, number of matches, and fairness in the distribution of utility among service providers for various period of matching t. For the simulation scenario we had $\frac{\tau}{\mu} = \frac{4}{5}$ and $\frac{|P|}{\lambda} = 4.17$. Since $\frac{|P|}{\lambda} \geq \frac{\tau}{\mu}$ this represents a high service rate setting. From the conclusions made in the analytic studies under standard utility distribution assumption it was proposed that $\frac{|P|}{\lambda}$ is the optimal matching period. From Figure 5.9 we see that $\frac{|P|}{\lambda}$ is the optimal matching period considering service provider utility, number of matches, and fairness in their distribution as the service provider utility and number of matches is maximum when $t \leq \frac{|P|}{\lambda}$ and decreases for $t > \frac{|P|}{\lambda}$. The service seeker matches decreases as t increases as Provider optimal DA primarily focuses on optimizing the ordinal (and thereby cardinal) efficiency of service providers. Some fluctuations are observed observed around the mean trend because the arrival process is Poisson and not deterministic in the simulation.

5.3 Conclusions

The effect of relaxing the idealistic assumptions used in the analysis presented in Chapter 4 were discussed. Relaxing the assumption of Deterministic arrival into more practical Poisson arrival process did not change the optimal matching period for both high and low service rate setting. Poisson arrival only causes some fluctuation around the mean predicted behavior for deterministic case. Optimal matching period is $\frac{|P|}{\lambda}$ in high service rate setting and $\frac{\tau}{\mu}$ under low service rate setting even with Poisson arrival process. In expectation, the matching objectives are optimal when the matching period is tuned to this value. However, relaxing the assumption of fixed printing time to a variable one which is exponentially distributed across the target population changes the optimal matching period under low service rate setting. Under low service rate setting with printing time exponentially distributed, lower the matching period more optimal the matched objectives. Both service seeker and service provider utility and fairness in their distribution increases as matching period t increases. But, in high service rate setting the optimal matching period, $\frac{|P|}{\lambda}$, is unchanged even when the printing time is exponentially distributed. Finally, an illustrative CBDM scenario was simulated by relaxing all the assumptions- the arrival process was modeled as Poisson, the service rate was heterogeneous among the service providers, the printing time was exponentially distributed among the designer population which varied from one service to another, with utility following no distributional assumptions. The optimal matching period for high service rate $\frac{|P|}{\lambda}$ generalized to this scenario.

6. MODELING EVOLUTIONARY DYNAMICS OF AIR TRANSPORTATION SYSTEM

To analyze the effects of policies within the air transportation network, there is a need to model how policies affect the decisions made by airlines. To model this effect we need to understand how airlines make decisions. Often such decisions are not independent, and is made in the presence of decisions by competing airlines. In this chapter, we present a model to understand how airlines make decisions including the effect of competition. The model is calibrated based on historic data.

The challenge is that the airline decisions are made based mostly on proprietary information. However, for such models to be useful for target stakeholders such as Federal Aviation Administration in the design of policies they need to rely only on openly available data sources. Therefore we develop a predictive model of airline route selection decisions based only on openly available data.

The model accounts for airline competition and parameters such as operating cost which can be influenced by the policymakers. The model is illustrated using a dataset from two major airlines in US domestic Air Transportation Network.

The dataset and the cost model are used for Bayesian estimation of model parameters, which are then used to predict the effects of cost and demand on the evolution of the network topology. From the estimates obtained on the preference parameters, it is found that decreasing the operating cost and increasing the market demand increases the probability of operating service on the route for airlines, and the operating cost has a greater effect than market demand and route distance in the route selection decisions.

6.1 Introduction

Airline decisions depend on a large number of variables. The data on many of these variables are proprietary, and therefore not publicly available. Several stakeholders, who do not have access to all the decision variables of the airlines, have an interest in understanding how airlines make these decisions. For example, regulatory bodies such as the Federal Aviation Administration (FAA) can use the entry decision model to direct the evolution of the Air Transportation Network (ATN) topology [56] towards improved connectivity, robustness, and resilience. Hence, there is a need to model such routing decisions using publicly available data.

In addition to being publicly available, it is important to base these decision models on factors that policymakers can influence. Air Transportation Network (ATN) evolves based on network decisions made by the airlines. If we construct a model that mimics network decisions of airlines by understanding their preferences towards factors that policymakers can influence, then such models can be used to provide useful guidance to regulatory bodies such as FAA to play an active role in channeling the network evolution towards targeted performance. For example, some of the factors such as operating cost can be directly influenced by the policymakers through incentives or imposing taxes. A decision model that estimates the preferences of airlines based on operating cost when making route decisions can be effectively used by policymakers in co-evolving the network.

Various approaches have been used in the past to model airline route selection decisions using publicly available data on route characteristics such as market demand, operating costs, route length. These approaches were primarily based on approaches such as linear programming [57], integer programming [58], and machine learning [59]. All these approaches are focused on predicting the evolution of the ATS network. They do not explicitly model the decisions made by airlines in response to changes in factors that policymakers can influence. Sha et al. [60] develop decision-based models to estimate the importance of decision variables that can be influenced by policymakers. These models are based on discrete choice, and use procedures to quantify the preferences of airlines towards route characteristics. The limitation of this model is that it does not account for the effect of competition among the airlines. In addition to route characteristics, potential competition from other airlines also influences these decisions. The profit that the airlines gain by operating a service on a route is affected by the presence of a competitor. The competitor may lead to a decrease in profit by reducing the market share or an increase in profit by reducing overall operating cost. Based on the nature of interaction the effect of the competitor on profit may be positive or negative. A discrete games model developed by Bresnahan and Reiss [61] has been effectively used in the past to model the effect of competition on airline decisions. These models study the effect of demographic factors such as city population of connecting airports or per capita income, nature of destination such as vacation or commercial, and geographic factors such as distance. Policymakers do not have influence over these demographic and geographic factors. In summary, discrete choice-based models help to understand the preferences of airlines towards factors that the policymakers can influence but they do not account for competition. On the other hand, discrete games models [62] address competition but are limited in terms of usefulness to policymakers.

In this paper, we address these limitations by estimating preferences towards factors that the policymakers can influence, while including the effect of competition. Once the preferences are estimated, we study how the airline's entry decisions are affected by variations in these factors. In particular, we study how the strategic route decisions taken by airlines in the presence of competition from other airlines is affected by these variations. With the motivation of understanding how airlines make route selection decisions under competition, we answer the following research questions: (1) how can the preferences of airlines towards factors that the policymakers can influence be estimated including the effect of competition? and (2) how do the variations in route characteristics affect the route decisions and the number of routes operated by the airlines?

To answer these questions, we build on the discrete games model by Ciliberto and Tamer [62] and estimate the preferences of airlines towards operating cost, market demand, and distance while making the route selection decisions. Operating cost, market demand, and distance are referred to as the explanatory variables. The strategy of whether or not to operate a service on the route by each airline is based on the preferences for explanatory variables, and presence of the competitor in the route. The Markov Chain Monte Carlo (MCMC) method is used to obtain posterior distribution on these preference parameters using data from decisions made by these airlines in the past. We assume that the airlines played Nash Equilibria strategy of the modeled discrete game. The presence of non-unique Nash equilibria makes the estimation of preference parameters challenging. Existing methods address this challenge by introducing a large number of latent parameters causing problems such as over-fitting. To tackle the problem of non-unique Nash equilibrium without over-fitting the model, we use a network level parameter called airport presence [63]. The Metropolis-Hastings algorithm [64] is used to sample the posterior on preferences towards various route characteristics and the effect of competition, and to study the relative significance of each of the route characteristics in the route selection decision.

Section 6.2 reviews existing literature, Section 6.3 describes the theoretical model of airline competition and Section 6.4 provides an overview of the numerical solution procedure used to estimate the parameters in the model. Finally, Section 6.5 summarizes the results obtained such as estimates of the decision parameters, inferences drawn from the preference estimation and the effect of cost and demand fluctuation on the evolutionary characteristics of the network.

6.2 Literature Review

Various approaches have been used in the past to model airline route selection decisions using publicly available data on route characteristics such as market demand, operating costs, route length (examples include Jaillet et al. [65], and Balakrishnan et al. [66], Lohatepanont and Barnhart [67]). These approaches are either prescriptive or descriptive in nature. In prescriptive approaches the objective is to provide guidelines for airlines in making routing and fleet planning decisions to maximize profit. In descriptive approaches the aim is to model or understand how airlines make routing decisions. Dantzig [68] and Kushige [57] used a prescriptive approach based on linear programming for fleet assignment on routes to maximize profits of airlines. When the number of alternative routes is large, this approach breaks down. Mixed integer programming was used by de Lamotte et al. [58] to study airline route schedule planning. Balakrishnan et al. [66] developed a Lagrangian-based solution to this mixed integer formulation of long-haul aircraft routing problems which selects candidate routes from a large set of possible alternatives. A category of descriptive approaches to model routing decisions of airlines is using network theory. Some of the examples of efforts using network theory are multiplier model by Song et al. [69], and machine learning techniques such as random forests, and support vector machine by Kotegawa [59] to model network evolution from historical data.

None of the descriptive approaches mentioned above explicitly model the decisions made by airlines in response to changes in factors that policy-makers can influence. Decision-based models have been developed to estimate the importance of decision variables that can be influenced by policy-makers. For example, Boguslaski et al. [70] studied the entry pattern of Southwest Airlines using demand, distance, and cost as explanatory variables. Sha et al. [71] developed a decision-making model for a US-based airline using demand, operating costs, geographic distances between airports, and hub or non-hub nature of airports as the inputs variables. While Sha et al. [71] modeled demand as a continuously increasing input parameter, a later work by Moolchandani et al. [72] developed models of passengers' decision-making.

In addition to route characteristics, the potential competition from other airlines also influences these decisions. Berry [73] builds on the discrete games model developed by Bresnahan [61] to include competition between airlines using the logarithmic number of carriers operating service on a route as a measure of competition in the route. The main challenge while solving the aircraft routing problem using discrete games is the presence of non-unique Nash equilibria. Non-unique Nash equilibria makes it difficult to estimate the choice probabilities of strategy profile [74]. Ciliberto and Tamer [62] proposed a model based on Gibbs sampling procedure to overcome this challenge of non-uniqueness. A drawback of this model is that there are as many latent variables as there are routes in the network. The number of latent parameters grows in proportion to the square of the number of airports in the network leading to problems such as over-fitting and poor predictive accuracy.

In summary, there have been numerous studies on airline decision-making on route selection. To this literature we add consideration of competition and how it affects the airlines' choices. This model is an extension of the discrete choice model developed previously by the authors, see Ref. [60].

6.3 Theoretical Model

This section describes the game theoretical approach that models the interaction between airlines while making route selection decisions. Following the work of Ciliberto et al. [62] the entry decisions made by airlines on each route are modeled as a discrete game with complete information played between two players. The payoff achieved by a player on operating a service on a route depends on route characteristics, preferences for route characteristics and the presence of competitors in that route. Market factors are assumed to be known to the researcher. Here, the researcher is a stakeholder extraneous to the airlines who is trying to understand the route selection decision of the airlines and does not have access to all the private data that the airlines possess while making such decisions. The aim of the model is to estimate the airline preferences for route characteristics and the change in payoff due to the presence of the competitor, based on the available data of decisions made in the past. In this section, we present a theoretical model and an approach to estimate the parameters in the decision model. The section is structured as follows: Section 6.3.1 describes the game-theoretic model of the route-level interaction, Section 6.3.2 describes the forward model that characterizes the Nash Equilibria strategies of the players, and Section 6.4 is on the numerical procedure to obtain the posterior distribution on the target parameters based on the decisions made in the past.

6.3.1 Game Theoretic Model of Route Interaction

Airline interaction at the route level is modeled as a discrete game of perfect information. The payoff matrix of this game is summarized in Table 6.1. The payoff matrix describes the utility gained by each player for different strategy profiles. A strategy profile is a combination of strategies undertaken by the players. The strategy of a player k is denoted by the symbol s^k, and on a route r, it is either to operate (denoted as $(s^k(r) = 1)$) or not to operate (denoted as $(s^k(r) = 0)$) service on that route.

		(Player 2) $s^2 = 0$	$s^2 = 1$
(Player 1)	$s^1 = 0$	$0, 0$	$0, \vec{X}^2(r)\vec{\beta}^2 + \epsilon^2$
	$s^1 = 1$	$\vec{X}^1(r)\vec{\beta}^1 + \epsilon^1, 0$	$\vec{X}^1(r)\vec{\beta}^1 + l^1 + \epsilon^1, \vec{X}^2(r)\vec{\beta}^2 + l^2 + \epsilon^2$

Table 6.1.
Payoff matrix associated with the strategy profiles adopted by the players.

We assume that route decisions are independent, i.e., the decision on whether or not to operate service on a route is independent of the decisions made on other routes. However, it does depend on the connectivity and network-level influence of the route, which is accounted for in our model. From one time step to the next, airlines make decisions to add new routes or delete existing routes based on the utility attained.

The utility attained by adding or deleting a route is affected by the decisions made on other routes. The network properties of a route such as the degree of the terminal airports, overall connectivity, and eigenvector centrality are affected by the decisions made on other routes. The network structure, in turn, alters the demand and cost of operating service on the route. However, this effect on utility is small as the fraction of routes added or deleted in a year is normally under 10%, as observed from the BTS T-100 dataset [18].

The utility attained by player k on operating a service on the route is written as the sum of observed (V_{ij}^k) and unobserved (ϵ^k) components, as shown in Equation (6.1). The observed component accounts for all the factors observed by the researcher such as the market demand, route length, operating cost and presence of a competitor. The unobservable component is only from the perspective of the researcher. From the perspective of the players, this is a perfect information game.

$$U_{ij}^k(r) = V_{ij}^k(r) + \epsilon^k(r) \tag{6.1}$$

The observed component consists of two parts (Equation (6.2)): a) a route specific component that depends on the route characteristics ($\vec{X}^k(r)\vec{\beta^k}$), and b) a part that accounts for the effect of the presence of the competitor ($s^{-k}(r)l^k$).

$$V_{ij}^k(r) = \vec{X}^k(r)\vec{\beta}^k + s^{-k}(r)l^k \tag{6.2}$$

In the first component, ($\vec{X^k}$) denotes the route characteristics and ($\vec{\beta^k}$) denotes the preferences of player k towards them. Following from Sha et al. [71], the main route characteristics contributing to airline decisions are market demand, non-stop distance, and operating cost. Therefore, these three route characteristics are considered in the model. In the second component, $s^{-k}(r)$ denotes the strategy adopted by a competitor of player k on route r, and l^k denotes the change in the utility of player k in the presence of its competitor. The term $s^{-k}(r)l^k$, by formulation, vanishes for a route if the competitor decides not to operate service on that route. The parameter l^k captures the difference in utility due to the presence of a competitor from being a monopolist operator on a route. This parameter is different for different airlines depending on the characteristics of the airlines and the market conditions. For example, full-service airlines and low-cost carriers respond differently in the presence of competition. l^k is positive or negative depending on whether the presence of the competitor is beneficial or harmful. Sugawara and Omori [74] showed that for Japanese airline market JAL airlines favors those routes where ANA airlines have already been operating ($l^k > 0$ for JAL airlines) whereas entry of JAL airlines reduces the profit of ANA airlines ($l^k < 0$ for ANA airlines).

The strategy profiles of both the players and the utility associated with them are summarized as a payoff matrix in Table 6.1. In this matrix, each element has two values of payoff, where the first value corresponds to the utility of the first player, and the second value is the utility of the second player for the associated strategy profile. Without loss of generality, the utility of each player for not operating a route is assigned to be zero.

6.3.2 Nash Equilibria of the Game

The game described in Section 6.3.1 is played on each route. The rational strategy is to operate a service on the route if the payoffs are positive, and not operate otherwise. Following from Equations (6.1) and (6.2), the total utility attained by player A^k on operating service on a route is obtained as $\vec{X}^k(r)\vec{\beta}^k + \; s^{-k}l^k + \epsilon^k$. The best response strategy of a player A^k is to operate service on the route if the utility is positive as shown in Equation (6.3). $s^k(r) = 1$ denotes that player A^k operates a service on route r and $s^k(r) = 0$ denotes that player A^k does not operate a service on route r.

$$s^k(r) = \begin{cases} 0 & \text{if } \vec{X}^k(r)\vec{\beta}^k + s^{-k}l^k + \epsilon^k \leq 0 \\ 1 & \text{if } \vec{X}^k(r)\vec{\beta}^k + s^{-k}l^k + \epsilon^k > 0 \end{cases} \tag{6.3}$$

There are four possible pure Nash equilibria strategies corresponding to strategy profiles $\vec{S}(r) = (s^1(r), s^2(r)) = (0,0), (1,0), (0,1), and (1,1)$. For a given payoff matrix a unique Nash equilibrium strategy may or may not exist depending on the payoffs associated with the strategy profiles. Assuming that the presence of the competitor negatively affects the utility of operating service on the route for both the players $l^1 < 0, l^2 < 0$, the Nash Equilibrium of the game in each route is expressed as a function of unobserved variables, ϵ^1, ϵ^2 as shown in Figure 6.1. This is a reasonable assumption as the entry of an airline in a route decreases the profit of the other airlines operating in that route in the US domestic ATN [74]. The regions of Nash-equilibria are different if the presence of a competitor increases the utility attained for any one player. A detailed analysis of all such cases is provided by Bresnahan and Reiss [61].

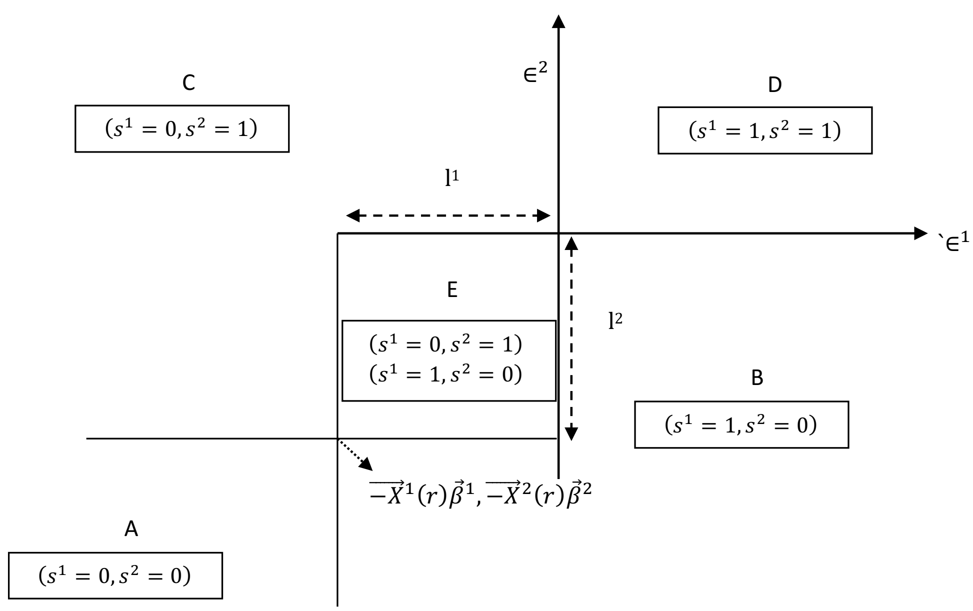


Figure 6.1. Nash Equilibria regions as functions of unobserved variables assuming $l^1 < 0, l^2 < 0$.

Figure 6.1 is a plot of Nash-equilibria of the game described in Table 6.1 with unobserved variables as axes (ϵ^1 as the horizontal axis and ϵ^2 as the vertical axis).

Regions A, B, C, and D have unique Nash equilibria corresponding to equilibria $\vec{S}(r) = (s^1(r), s^2(r)) = (0,0), (1,0), (0,1), (1,1)$ respectively. Region E has multiple Nash-equilibria.

The presence of regions with multiple Nash-equilibria makes the estimation of model parameters harder. This is because if we assign distributional assumptions on the unobserved variables (ϵ^1, ϵ^2) the choice probabilities on Nash equilibria are not well-defined in the regions of multiple equilibria. We propose an approach based on a network parameter called airport presence [63] to overcome this problem of regions of multiple Nash-equilibria. The airport presence of an airline at an airport is the ratio of the number of airports directly operated to by the airline to the total number of airports directly operated to by all airlines from that airport. We assume that the probability of a player entering the route conditional on the route falling in region E is a function of the airport presence of the player in the route. Several authors (e.g., Borenstein [75], and Levinine [76]) have argued that the level of operations in an airport has a significant impact on the competitive position on the route operated from or to the airport. Berry [63] investigated the effect of airport presence on the oligopoly product differentiation. Our examination of historic decisions indicates that the probability of a player entering is a non-linear function of airport presence. We observe from the past data [18] that if the airport presence of a route was more than 0.5 then the probability of retaining that route was quite high as observed from data. As an example, Figure 6.2 shows the entry decision of United Airlines (UA) as a function of its airport presence in the year 2012-13. A similar logistic functional relation between entry decision of UA and airport presence was observed for other years as well. Therefore, in region E, a logistic functional relation was assumed between the probability of playing equilibrium $(1,0)$ and airport presence, as shown in Equation (6.4).

$$p_r = (1 + \alpha_1 + \alpha_2 a^1(r))^{-1} \tag{6.4}$$

This probability of equilibrium $(1,0)$ being played in route r is denoted by p_r and airport presence of UA is denoted by $a^1(r)$.

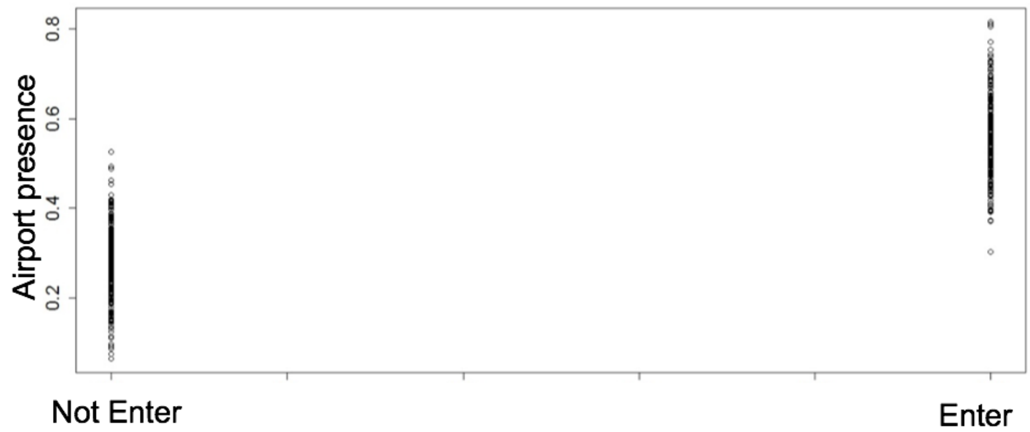


Figure 6.2. Entry decision of United Airlines (UA) as a function of airport presence in year 2013. All routes in a network formed by the top 132 US domestic airports [77] was considered in this plot.

6.4 Numerical Procedure to Estimate the Parameters in the Theoretical Model

To estimate the parameters of the decision model developed in Section 6.3, a Bayesian approach is used with priors obtained using Discrete Choice Analysis and likelihood obtained by data from historic decisions made by the airlines. The posterior distributions of parameters, so obtained, are sampled using the standard MCMC method based MH algorithm [64].

6.4.1 Priors using Discrete Choice Analysis (DCA)

Sha et al. [60] developed a model based on discrete choice random-utility theory to estimate decision-making preferences of airlines. In DCA, the utility (U_i) for alternative i consists of a component that is observed by the researcher (V_i) and unobserved component ϵ_i which is the uncertainty, as shown in Equation (6.5).

$$U_i = V_i + \epsilon_i \tag{6.5}$$

The observed component is deterministically estimated from the researcher's point of view through different techniques such as survey data and expertise. The unobserved component captures the uncertainty due to unobserved attributes, measurement errors, etc. For example, the airline route decision criteria not included in the model are captured in this term. A linear form of the observed component was used by Sha et al. as shown in Equation (6.6).

$$V_i = \vec{x^T}\vec{\beta_i} \tag{6.6}$$

where $\vec{x} = (x_1, ..., x_n)^T$ is a set of n explanatory variables for utility, and $\beta_i = (\beta_{i1}, ..., \beta_{ik})^T$ is a set of weights that quantify the preferences of decision maker i. Under random-utility maximization assumption, the decision maker prefers alternative i over j if $U_i \geq U_j$. The probability of a decision maker choosing alternative i is obtained as a function of the cumulative distribution of $(\epsilon_j - \epsilon_i)$ as shown in Equation (6.7) [78].

$$P_i = P(U_i \geq U_j) = P(V_i - V_j \geq \epsilon_j - \epsilon_i) \tag{6.7}$$

Sha et al. used a multinomial logit model [79] that assumes ϵ_i to be independent and identically distributed following a Gumbel distribution. The explanatory variables were market demand, direct operating cost, distance and whether a route connects hub airports or not. The preferences for the explanatory variables were then estimated using the discrete choice model.

In our model, we use the estimates of the preferences obtained by running the discrete choice model as parameters of the prior distribution. Discrete choice analysis gives point estimates for preference parameters of the airlines. For preference parameters of each player towards the explanatory variables, a normal prior is assigned with mean obtained using discrete choice analysis. For the parameter l^k, a normal prior is assigned truncated in the region $(-\infty, 0)$ since we assume that the presence of the competitor decreases the utility for both the players. α_1, α_2 are given a normal prior.

The parameters for prior on $\vec{\alpha}$ are obtained by fitting a logistic curve between airport presence and entry decision of Player A^1 in all routes where $(0,1),(1,0)$ equilibria were played.

6.4.2 Likelihood

After assigning the priors, the next step is to obtain likelihood function of the model parameters from the observed data. Assuming that the unobserved variables, ϵ^1 and ϵ^2, are independent and follow the standard normal distribution, the likelihood of each of the strategy profile regions in Figure 6.1 is expressed as a function of the vector of route characteristics $\vec{X}^k(r)$ and the preferences $\vec{\beta}^k$ towards them. Equations 6.8 through 6.12 are the likelihood values of falling in each of the five regions [74]. In these equations, ϕ denotes cumulative density of standard normal distribution and L^i denotes the likelihood of falling into region i in Figure 6.1.

$$L_A(\vec{\beta},\vec{X}_r) = \phi(-\vec{X}_r^1\vec{\beta}^1)\phi(-\vec{X}_r^2\vec{\beta}^2) \tag{6.8}$$

$$L_B(\vec{\beta},\vec{X}_r) = \{\phi(\vec{X}_r^1\vec{\beta}^1) - \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\}\phi(-\vec{X}_r^2\vec{\beta}^2) + \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\phi(-\vec{X}_r^2\vec{\beta}^2 - l^2) \tag{6.9}$$

$$L_C(\vec{\beta},\vec{X}_r) = \{\phi(\vec{X}_r^1\vec{\beta}^1) - \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\}\phi(\vec{X}_r^2\vec{\beta}^2 + l^2) + \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\phi(\vec{X}_r^2\vec{\beta}^2) \tag{6.10}$$

$$L_D(\vec{\beta},\vec{X}_r) = \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\phi(\vec{X}_r^2\vec{\beta}^2 + l^2) \tag{6.11}$$

$$L_E(\vec{\beta},\vec{X}_r) = \{\phi(\vec{X}_r^1\vec{\beta}^1) - \phi(\vec{X}_r^1\vec{\beta}^1 + l^1)\}\{\phi(\vec{X}_r^2\vec{\beta}^2) - \phi(\vec{X}_r^2\vec{\beta}^2 + l^2)\} \tag{6.12}$$

The joint likelihood function assuming that routes are independent is given by Equation 6.13.

$$L(\vec{X},\vec{S}|\vec{\beta},\vec{\alpha},l^1,l^2)=\prod_{r=1}^{R}\Bigg[\Bigg\{(L_A)^{I[\vec{S}(r)=1]}(L_B+\lambda_r L_E)^{I[\vec{S}(r)=2]}(L_C+L_E-\lambda_r L_E)^{I[\vec{S}(r)=3]}$$
$$(L_D)^{I[\vec{S}(r)=4]}\Bigg\}\Bigg\{\frac{1}{1+e^{\alpha_1+\alpha_2 a^k}}\Bigg\}^{I[\vec{S}(r)=2,3]}\Bigg] \quad (6.13)$$

In this equation, R is the total number of routes in the network and L_i are defined in Equations 6.8 through 6.12. λ_r is a Bernoulli variable with airport presence as the parameter (Equation 6.4.2).

$$\lambda_r \sim \; Bernoulli(p_r)$$

$\lambda_r = 1$ when $(1,0)$ equilibrium played and $\lambda_r = 0$ when $(0,1)$ is played.

6.4.3 Posterior

The joint posterior density function for the parameters $\vec{\alpha}$, and $\vec{\beta}$ is proportional to the product of prior and likelihood as shown in Equation 6.14.

$$Y(\vec{\beta},\vec{\alpha},l^1,l^2|\vec{X},\vec{S}) \propto \pi(\vec{\alpha})\pi(\vec{\beta})\pi(l^1)\pi(l^2)L(\vec{X},\vec{S}|\vec{\beta},\vec{\alpha},l^1,l^2) \quad (6.14)$$

where $\pi(\vec{\beta})$, $\pi(\vec{\alpha})$, and $\pi(\vec{l^k})$ denote the prior distribution on preference parameters on explanatory variables, airport presence, and interaction factor respectively, and L is the joint likelihood function.

6.4.4 Sampling posterior distribution

The posteriors on the parameters $\vec{\beta}$, l^1, and l^2 are sampled using the standard MCMC method based MH algorithm [64]. While sampling $\vec{\alpha}$ the range is limited so that it gives a valid probability sample. A distribution similar to the prior with the same mean and standard deviation of 5000 times lower than the prior was used as the jump function in the MH algorithm.

6.5 Estimation Results of the Empirical Study

This section shows the results of Bayesian estimation on preference parameters.

The results from the discrete choice analysis were used to obtain the parameters for the prior distribution. Discrete choice analysis provides separate preference parameter values for route addition and route deletion. A weighted average of this (based on the fraction of operating and non-operating routes) were used to obtain the prior parameter. The standard deviation was set high so that the prior remains flat and do not bias the posterior. Thus the resulting posterior distribution is inferred primarily from the real data. The parameters of the prior distribution for the three explanatory variables are tabulated in Table 6.2. The discrete choice analysis do not distinguish between airlines and therefore the same parameter values were used in prior for both the airlines.

Table 6.2.
Statistics of the posteriors of decision parameters (after burn-in period). Prior Mean (add) are the results obtained on preference parameters by running the discrete choice model for route addition. Prior Mean (del) are the same results for route deletion.

Parameter	**Prior mean**	**Prior stdev**
Demand	0.055	10
Demand	0.469	10
Distance	-0.044	10

A total of 1 million samples were used in the Metropolis-Hastings algorithm to estimate the posterior distribution of the preference parameters. For the proposal distribution a standard deviation 5000 times smaller than the prior was used. This resulted in a reasonable acceptance rate of 33.4% for the posterior samples when

Metropolis-Hastings MCMC was implemented. The raw MCMC posterior samples for the preference parameter towards the explanatory variables of both the airlines are shown in Figure 6.3.

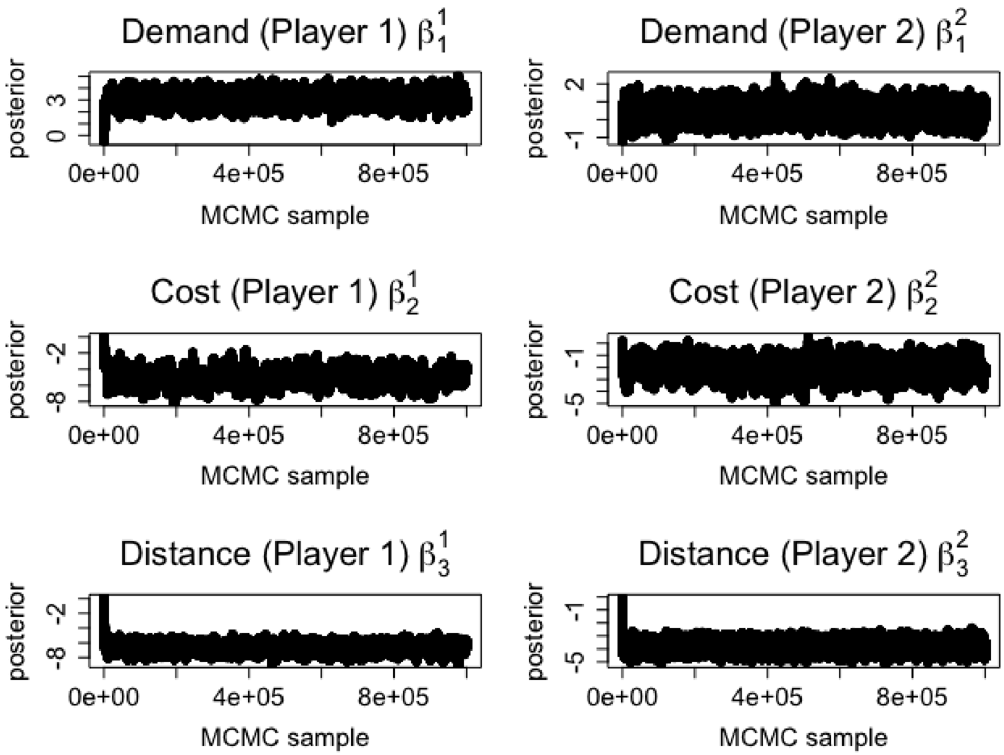


Figure 6.3. Raw MCMC posterior samples of the preference parameters towards cost, market demand, and distance of both the airlines.

2000000 samples were taken as the burn-in period. For the filtered samples, autocorrelation plots were made. Figure 6.4 shows the autocorrelation between samples that are adjacent to each other or spaced by a certain number *lag*. Based on the autocorrelation plot, one in every 10000 posterior sample is picked.

Figures 6.5 and 6.6 show the posterior samples after removing the samples in burn-in period and after accounting for autocorrelation.

Table 6.3 compares the statistics of the posterior on preference parameters with the prior. The prior mean corresponds to the estimates of the preference parameters

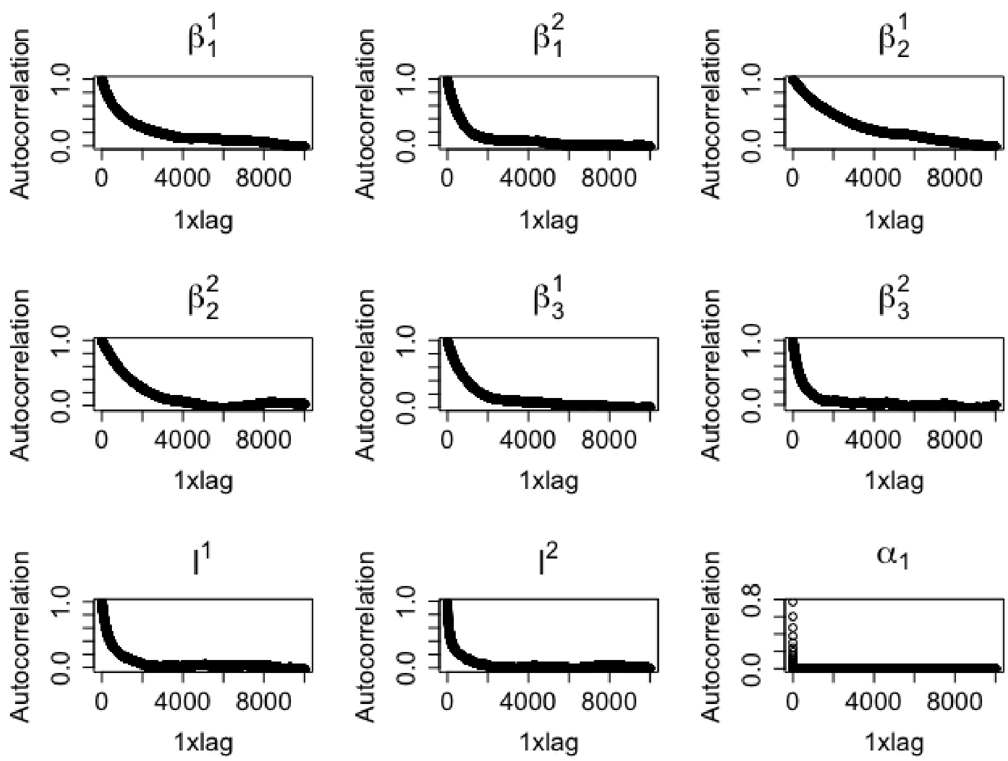


Figure 6.4. Plot showing average autocorrelation between samples that are *lag* spaced apart for each all the parameters.

using discrete choice analysis for the same year. Discrete choice analysis provides two sets of parameters, one for non-operating routes for route addition and the other for operating route for route deletion. The posterior samples on preference parameters quantify the preferences of the airline towards explanatory variables such as cost (unit: cent/nautical mile/seat), demand (unit: 1000 passengers), and distance (unit: 1000 nautical miles). For the interaction parameter the prior parameters correspond to the intercept of discrete choice analysis result. For the coefficient of airport presence the prior parameters were obtained by curve-fitting the parameters α_1 and α_2 on the data for the previous year.

From Table 6.3, it is observed that both the players under competition have a positive coefficient for demand and negative for the cost and distance. Higher the

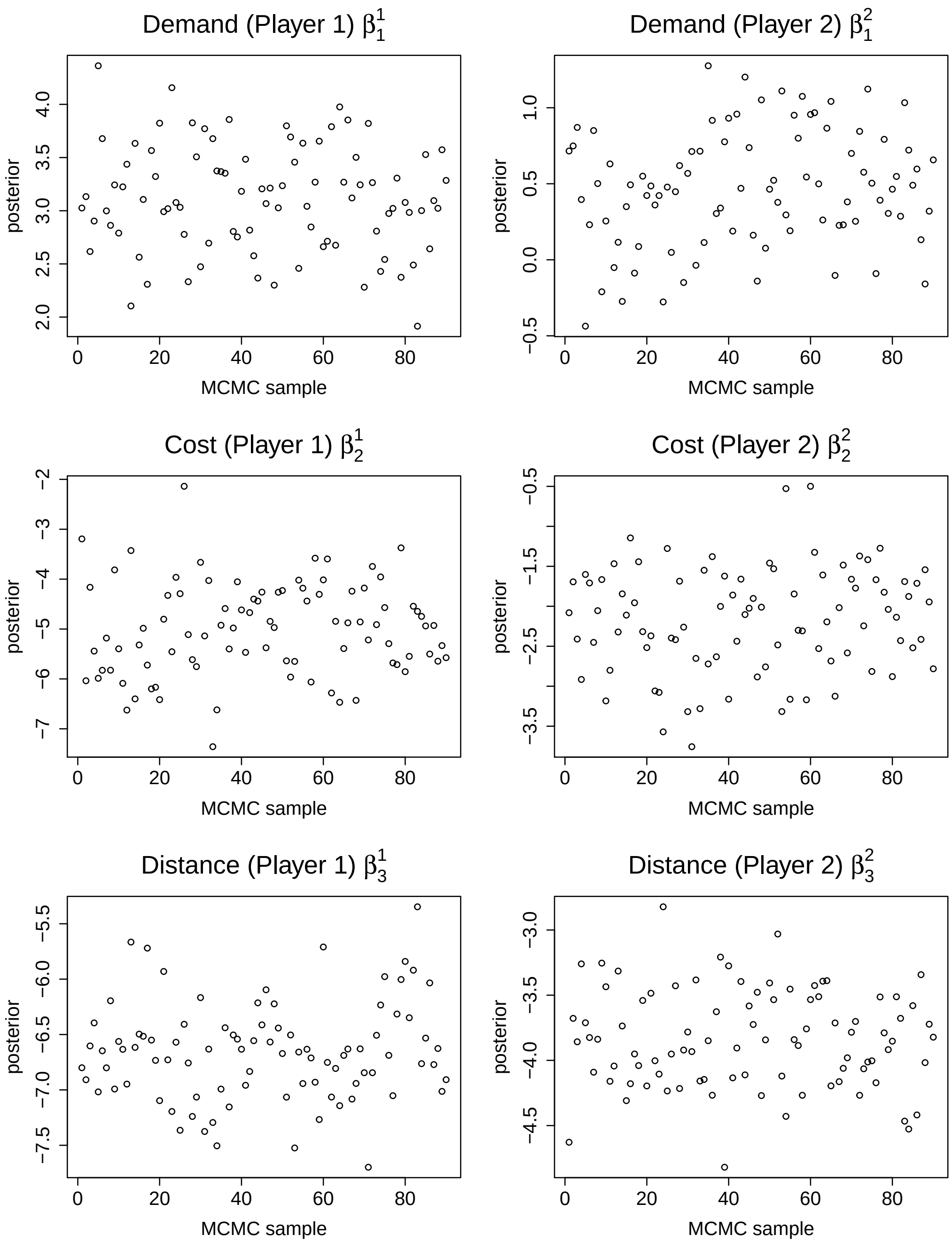


Figure 6.5. MCMC posterior samples for cost, demand, and distance preference parameters after processing (removing the burn-in period and accounting for autocorrelated samples).

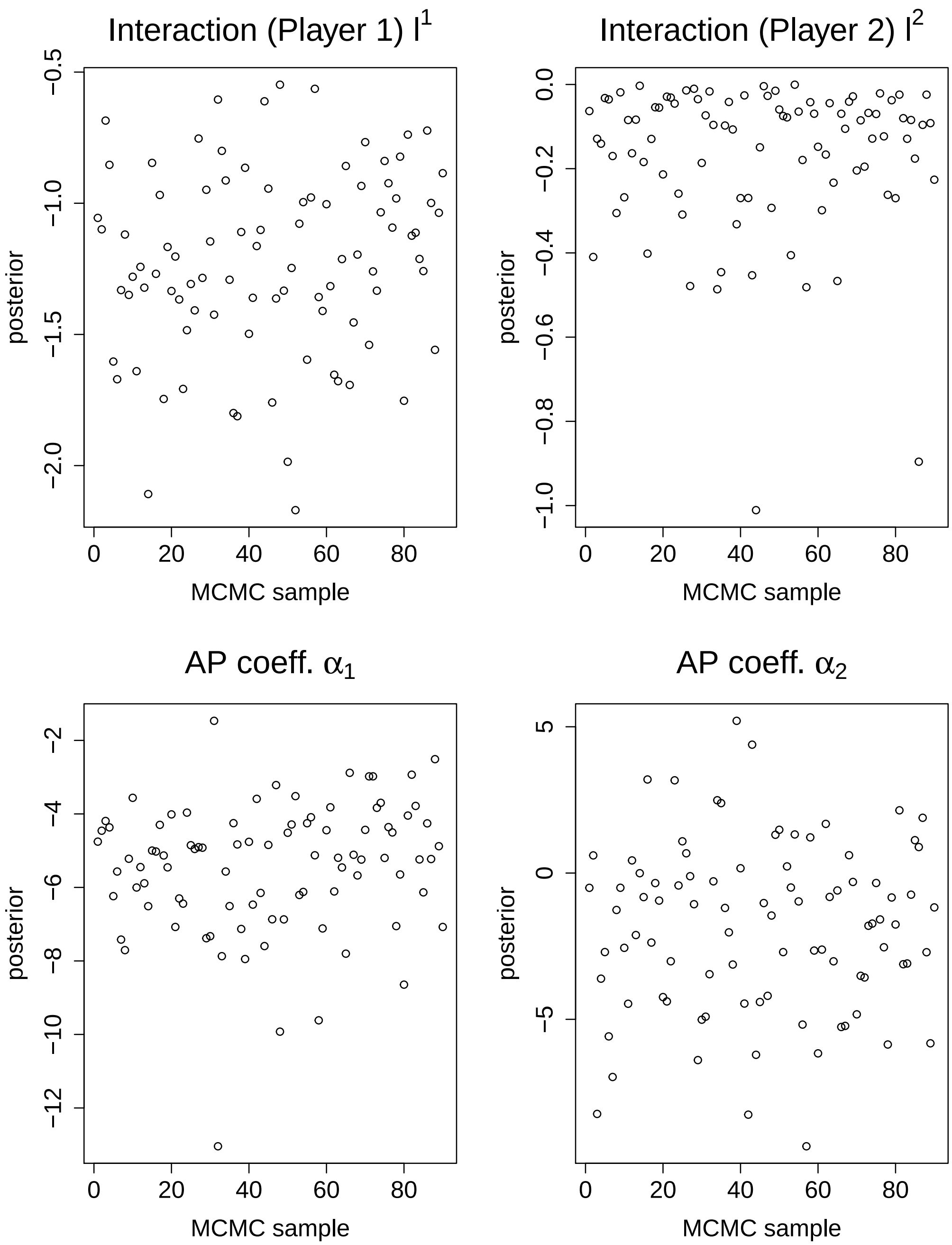


Figure 6.6. MCMC posterior samples for interaction between airlines and airport presence coefficient after processing (removing the burn-in period and accounting for autocorrelated samples).

Table 6.3.
Statistics of the filtered posterior samples. Prior Mean (add) are the results obtained on preference parameters by running the discrete choice model for route addition. Prior Mean (del) are the same results for route deletion.

Parameter	**DCA (add)**	**DCA (del)**	**Posterior mean (UA)**	**Posterior mean (DL)**	**Posterior std dev (UA)**	**Posterior std dev (DL)**
Demand (β_2)	0.087	-0.23	3.11	0.47	0.49	0.38
Cost (β_1)	0.74	-1.97	-5.01	-2.17	0.93	0.65
Distance (β_3)	-0.04	-0.17	-6.66	-3.84	0.44	0.37
Interaction	-1.91	1.49	-1.22	-0.17	0.35	0.18

demand that a route brings in and lower the operating cost involved in adding the route, higher is the utility of adding that route. The effect of increasing cost negatively affects the probability of the route getting added. Furthermore, the magnitude of preference parameter for cost was one order of magnitude higher than the parameter for demand which means that cost had a higher significance in the route selection decision. This is encouraging from policy designer standpoint as cost is the variable that they can directly control. Demand is governed more by market factors. UA had a larger decline in the payoff from the presence of the competitor as compared to DL.

6.6 Conclusions

The competition between airlines at route level is modeled as discrete games of perfect information and the parameters of the model are estimated using an approach based on airport presence. Although the results presented in this paper are for competition between UA and DL, the approach can be extended to study the competition between any two airlines. The approach can also be extended to include more than two airlines. In two player case there is only one region of non-unique Nash equilibrium. If there are more than two players, then there will be multiple regions of non-unique Nash equilibrium and likelihood functions of each of those regions have to be addressed separately. A limitation of the approach is that it does not take into account the effect of decision made on a route in the current period on the decision made on other routes. This is a reasonable assumption as addition or deletion happens only in less than 10 % of the routes from one period to the next.

7. POLICY DESIGN USING PREDICTION OF EVOLUTIONARY DYNAMICS

In Chapter 6 a model to understand how airlines make decisions was developed. The dataset of historic decisions and cost model was used for Bayesian estimation of model parameters. Now, having developed an airline route selection decision model we use it to predict the effects of cost and demand on the evolution of the network topology.

Bresnahan had developed discrete games approach; however, such approach relies on Gibbs sampling including large number of latent variables. Inclusion of such large number of variables leads to over-fitting. Our model on the other hand uses relatively only a small number of variables. In terms of accounting for airline decisions to understand evolution of Air Transportation Network there are other competing models such as discrete choice analysis. The proposed model is found to be more accurate than competing models that do not consider the effect of competition particularly on routes where airlines actually made a decision to add or delete a route.

7.1 Route prediction accuracy

To evaluate the discrete games model, route prediction accuracy was measured and compared with the discrete choice model. Route prediction accuracy of an airline is defined as the fraction of routes where the model accurately predicts the entry decision of the airline. To obtain this accuracy a route-by-route comparison of the model prediction and the actual decision was made. For the period 2006-07, the model was trained using decisions made by both the airlines on 500 routes (of the 8646) in the year 2006. These 500 routes were a random sample consisting of all the four Nash equilibria. The trained model was then used to predict the accuracy in the year 2007. A comparison of the prediction accuracy obtained using the current model with that of discrete choice is tabulated in Table 7.1. The overall accuracy of prediction by discrete choice analysis from years 2006-13 is 90.6% and the accuracy for the current model is 77.1%.

Table 7.1.
Comparison of overall prediction accuracy of discrete choice analysis and current model.

Period	**Discrete choice**	**UA (current)**	**Delta (current)**
2006-07	90.4%	89.1%	85.9%
2007-08	89.3%	89.3%	85.8%
2008-09	89.6%	90.3%	86.7%
2009-10	91.8%	89.8%	78.0%
2010-11	90.1%	90.4%	79.2%
2011-12	90.1%	88.0%	79.3%
2012-13	90.5%	87.2%	77.6%

Though the overall accuracy of discrete choice is better than the current model, in routes where an addition or deletion decision was made (we call this dynamic accuracy) the current model performs better than the discrete choice model. This is because discrete choice prediction bases itself on the current status of the route. It relies on two separate models – one for route addition and the other for route deletion. Implicitly, the information of the current status of the route increases the prediction accuracy achieved using discrete choice model. From one period to the next, addition or deletion happens only in less than 10% of the routes and the discrete choice model predicts that the route status is unchanged in a significant majority of the routes. For example, in the period 2006-07 among the 2591 existing routes, only 53 routes were selected for deletion and only 107 new routes were added to the network. Even a model that takes information about route status in the previous year and predicts no changes to happen would yield a 94% accuracy. But the real use of the model is to understand how airlines make routing decisions. Therefore the merit of the model is gauged by comparing the accuracy only on those routes where route status changed from one period to the next. We call this dynamic accuracy of the model. The model that predicts no changes in the route status will have a dynamic accuracy of 0%. The discrete choice model had an average dynamic accuracy of 49.1%, and the current discrete games model has an average dynamic accuracy of 71.1% for UA and 69.9% for DL from the years 2006-2013. The accuracy achieved in individual years is listed in Table 7.2. This increased accuracy justifies that including the effect of competition helps better understand how airlines make addition and deletion decisions.

Table 7.2.
Comparison of dynamic prediction accuracy of discrete choice analysis and current model.

Period	**Discrete choice**	**UA (current)**	**Delta (current)**
2006-07	50.5 %	70.9 %	74.9 %
2007-08	40.0 %	65.7 %	74.3 %
2008-09	55.3 %	75.3 %	76.5 %
2009-10	59.1 %	77.9 %	75.6 %
2010-11	43.0 %	62.3 %	59.4 %
2011-12	47.4 %	65.6 %	60.4%
2012-13	56.7 %	64.9 %	63.6 %

Now, between 80% and 90% of the routes are non-operational and therefore, prediction accuracy alone as a metric is not sufficient to assess the performance of the model. The model was further validated using ROC curves. Figure 7.1 show the ROC curves for prediction of route entry decision for both the airline. If either UA or DL is operating the route then the route is operational, and the route is non-operational in a period only if neither UA nor DL operate in that period. This ROC curve had an Area Under Curve (AUC) of 0.74. Figures 7.2and 7.3 show ROC curves for prediction of route entry decision made by Player 1 (UA) and Player 2 (DL) respectively. The predictive model for Player 1 had an AUC of 0.77, whereas Player 2 our model attained an AUC of 0.70. The AUC for the predictive models of each airline were between 0.7 and 0.8. It is neither an excellent model (AUC$\geq$ 0.9) nor a good model (0.9 >AUC$\geq$ 0.8). But it is neither poor(0.7 >AUC$\geq$ 0.6) nor fail model

(0.6 >AUC≥ 0.5). This means that the model established is able to 'fairly' capture the decision making behavior of the airlines.

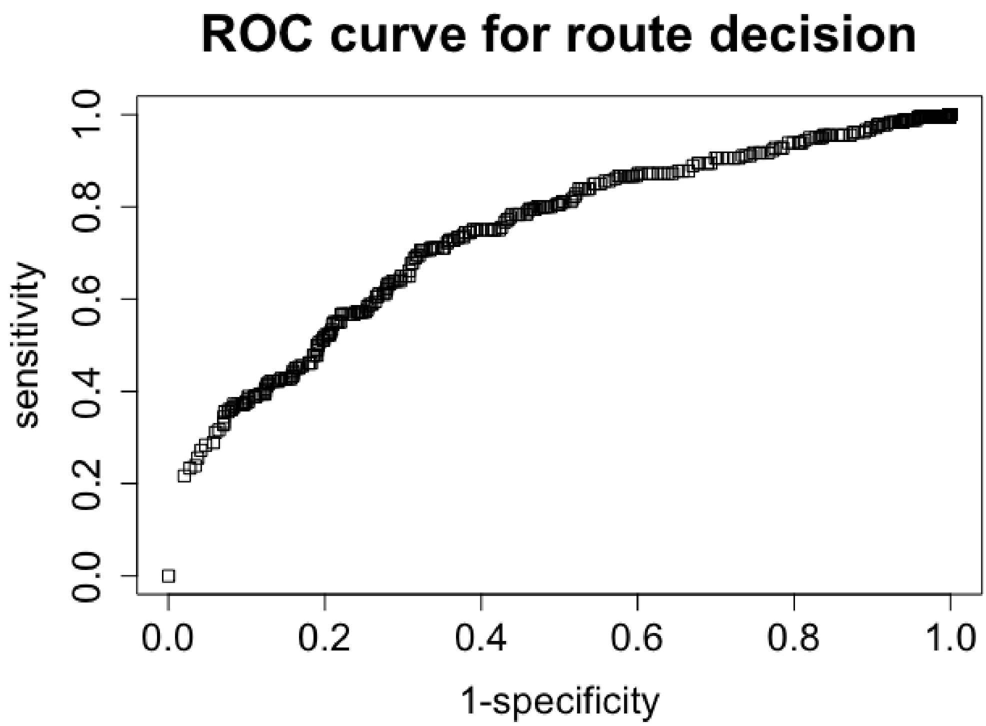


Figure 7.1. ROC curve for prediction of route entry decision for both the airlines.

7.2 Policy experimentation discussion

Based on the estimates for airlines' preferences of explanatory variables such as demand, cost, and distance the evolutionary behavior of the air transportation network is studied in this section. In particular, we focus on the effect of the explanatory variables on the Nash-equilibrium.

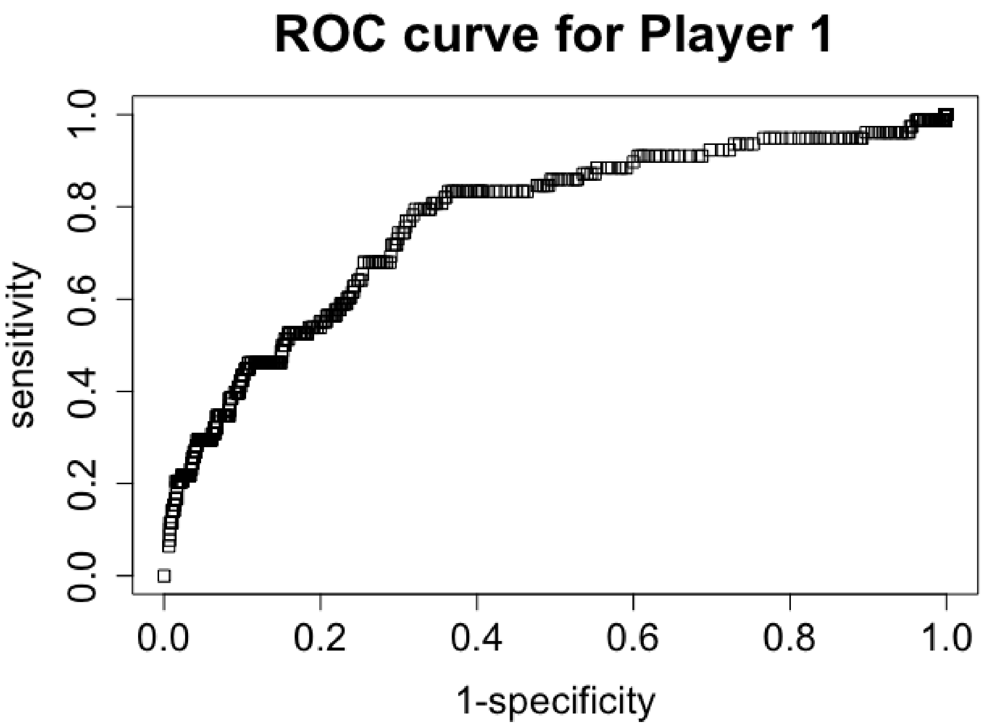


Figure 7.2. ROC curve for prediction of route entry decision made by Player 1 (UA).

7.2.1 Forward simulation

In Section 7.1 we validated the prediction accuracy of the model based on the estimated preferences towards explanatory variables. Since the model is able to predict the airline route decisions using only the explanatory variables with high accuracy we hypothesize that the model should capture the decision making behavior of the airlines on varying the explanatory variables. The three explanatory variables that were considered in our analysis were market demand, operating cost, and route distance.

We perform forward simulation by varying the explanatory variables: market demand and operating cost. Route distance is excluded from the analysis as it is fixed and neither the policymakers nor the market forces can alter the route distance from

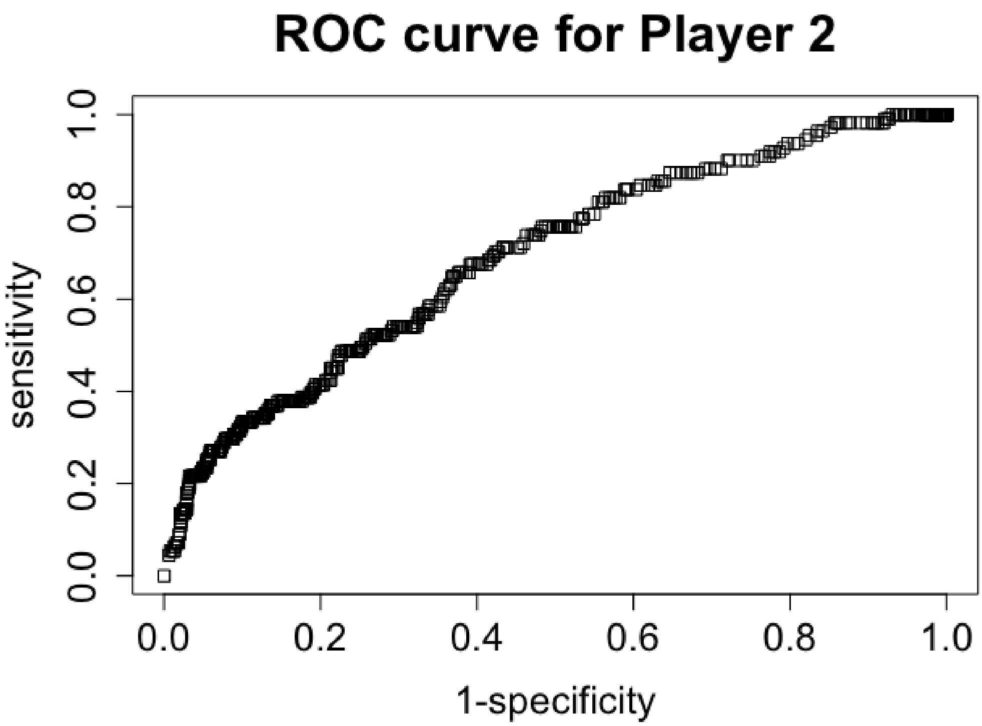


Figure 7.3. ROC curve for prediction of route entry decision made by Player 2 (DL).

one period to the next. While varying the explanatory variables, it was performed at a network level i.e. all the routes are simultaneously varied and network level aggregated effects are studied. While varying the operating cost the change was uniform for every route whereas for market demand the change for a route was in proportion to the existing demand in that route. This is because for passenger demand variations between period are usually in proportion to the congestion. For example, a busier airline route normally has more demand fluctuation between period in comparison to a less sought after route. On the other hand, for operating cost if airport charges or landing fees is increased then it is absolute for all routes and not in proportion to the existing operating cost for a route. It may be debatable as in practice the variation might be a combination of proportional and uniform; however, this assumption of relative or absolute does not fundamentally change the observation and conclusions drawn from varying the explanatory variables. The assumption of distinguishing proportional from absolute was made only to mimic the variations in practice while performing the forward simulation.

Varying operating cost

Based on the decisions made by the two airlines (whether to operate or not to operate), there are four possible outcomes or Nash equilibria strategies in each route. These are $(0,0)$, $(1,0)$, $(0,1)$, and $(1,1)$; where the first entry denote the strategy of Player 1 and second entry denote the strategy of Player 2. Figure 7.4 shows the number of routes where each of the four Nash equilibria strategies were played as a function of operating cost incurred by Player 1. The operating cost is increased uniformly for all the routes. In the figure, std dev indicate the standard deviation in the distribution of cost across all routes in the network the network. X-axis is the number of standard deviations by which the operating costs were increased for Player 1.

Through Figure 7.5 we explain the Nash equilibrium behavior with varying operating cost. This figure is constructed to analyze the outcome when operating cost of Player 2 is increased. From the posterior samples we obtained that the preference parameter towards cost for Player 2 had a mean value of -2.17. This means Player 2 had a negative preference towards higher cost. The solid lines partitions the Two-Dimensional space formed by the unknown variables ϵ_1 and ϵ_2 to five regions. Four of the regions have unique pure Nash equilibrium whereas the central region do not have a unique Nash equilibrium. The blue dotted line indicate the new boundaries demarcating the equilibrium regions after increasing the operating cost of Player 2. All the horizontal boundary lines shifts upwards in the vertical direction whereas all the vertical boundary lines are unchanged. This is because the coordinates $(-\vec{X}^1\vec{\beta}^1, -\vec{X}^2\vec{\beta}^2)$ and the lengths l^1, l^2 uniquely define the regions in the Two-Dimensional space. The

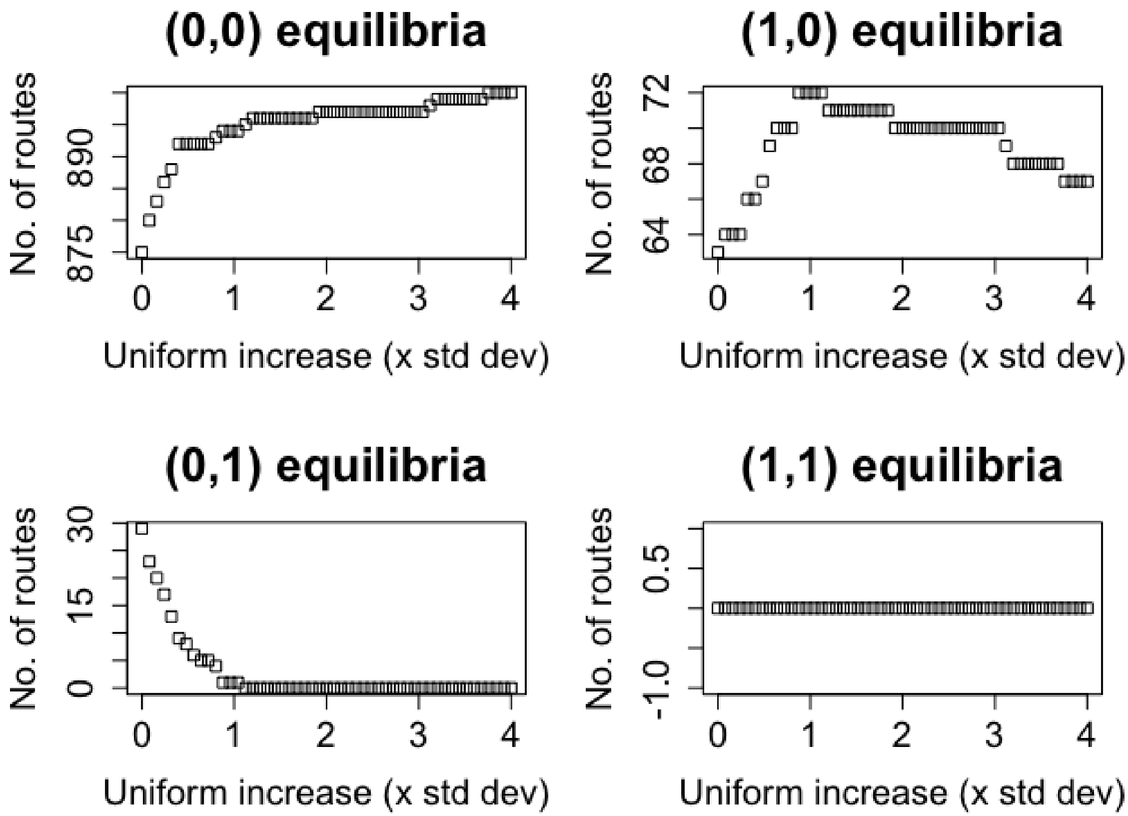


Figure 7.4. Comparing the number of routes in each Nash-equilibrium was predicted by increasing the operating cost of Player 2 (DL).

preference parameters for both the players ($\vec{\beta}^1$, $\vec{\beta}^2$) are unchanged as they are characteristic of the airlines. Since the operating cost of only Player 2 is changed, $\vec{X}^1$ also remains constant. Therefore, $\vec{X}^2$ is the only variable which defines the vertical position of the boundaries. Increasing operating cost shifts the boundary lines along the vertical axis (ϵ^2) in the positive direction as the coefficient associated with cost for Player 2 is negative. On increasing the operating cost of Player 2, area corresponding to region of $(0, 0)$ Nash equilibrium increases as indicated by area A_2. In Figure 7.4 we observe that the number of routes corresponding to $(0, 0)$ Nash equilibrium increases as expected. On the other hand, the area A_2 (along with some area swept by the central region) is removed from the likelihood region (region C in Figure 7.5)

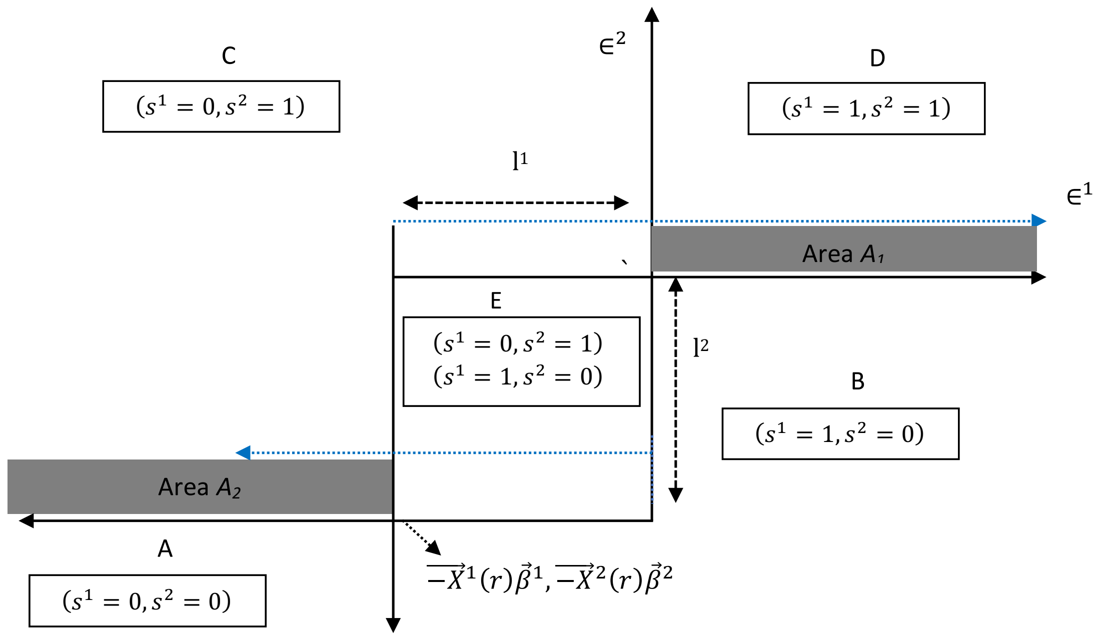


Figure 7.5. Effect on the likelihood regions of Nash-equilibria by increasing the operating cost of Player 2.

associated with equilibrium $(0, 1)$. Thus, the number of routes with $(0, 1)$ equilibrium decreases as observed from Figure 7.4. Region D corresponding to equilibrium $(1, 1)$ decreases in area by A_2 as shown in Ffigure 7.5. ϵ^1 and ϵ^2 are modeled as random variables drawn from standard normal distribution and the product cumulative density functions of the normal distribution corresponding to the area of the region gives the likelihood (and in turn probability) of that equilibrium being played as discussed in Equations 6.8 to 6.12. However, in the routes considered for prediction there were no routes where $(1, 1)$ equilibrium was played. The predictive model predicts this outcome with the actual values of explanatory variables. Increasing cost further will further diminish the probability of $(1, 1)$ equilibrium being played closer to zero. As a result the number of $(1, 1)$ equilibrium remains at zero on increasing cost as observed in Figure 7.4. The area under $(1, 0)$ equilibrium in region B of Figure 7.5 increases

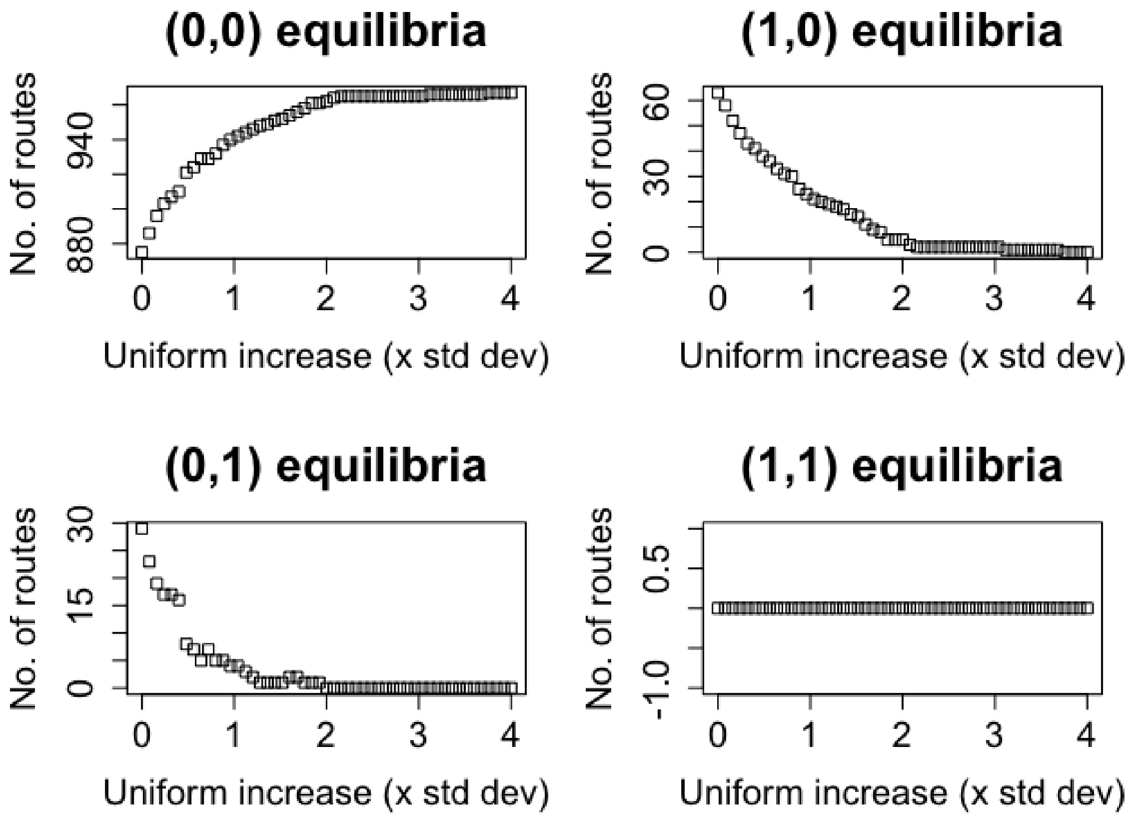


Figure 7.6. Comparing the number of routes in each Nash-equilibrium was predicted by increasing the operating cost of both the players.

and this is seen from an increased number of routes falling in $(1, 0)$ equilibrium in Figure 7.5.

Another interesting observation is that once the number routes with $(0, 1)$ equilibrium goes to zero (around 1.1 std dev in Figure 7.4 the number of routes where $(1, 0)$ equilibrium is being played decreases based on the assumptions made in our model. This is because once all the routes where $(0, 1)$ equilibrium was being played converts to $(0, 0)$ or $(1, 0)$, increasing cost further does not add any additional area into $(1, 0)$ equilibrium. However, the contribution of central region decreases on increasing the cost further. Area of the central region remains the same as parameters l^1 and l^2 are independent of the operating cost. However, the likelihood value associated with the same area decreases because of the nature of Gaussian density function. For example, $\phi(1.5) - \phi(1)$ is higher than $\phi(2.5) - \phi(2)$, where ϕ is cumulative density function of standard normal.

The interpretations through Figure 7.5 were provided on increasing the operating cost of Player 2 only. However, if the increase was made for operating cost of Player 1 instead the effects are very similar except that now the number of routes where $(1, 0)$ equilibrium is being played goes down to zero. Figure 7.6 shows the effect when the operating costs associated with both the players are increased. All equilibria except $(0, 0)$ goes down to zero eventually. Number of routes with $(0, 1)$ equilibrium drops faster compared to the routes with $(1, 0)$ equilibrium. This is because the preference parameter towards cost for Player 1 has a mean value of -5.01, whereas for Player 2 has a mean value of -2.17. Since Player 1 has a higher magnitude for the preference parameter, varying the cost has a bigger effect on the utility function of Player 1 as compared to Player 2. Therefore the number of routes by Player 1 drops more drastically towards zero as compared to Player 2.

Varying market demand

Figure 7.8 shows the number of routes where each of the four Nash equilibria strategies were played as a function of market demand for Player 1. The market demand for a route is increased in proportion to the original demand of that route. Figure 7.7 explains the effect of increasing the market demand of Player 1 on the five regions. As observed from Figure 7.7 increasing market demand for Player 1 shifts all the vertical lines demarcating the boundaries of the five regions along the negative ϵ^1-axis direction. The shift is in the negative direction as the preference parameters towards demand has a positive coefficient for Player 1 from the predictive distribution results. Similar to the operating cost, since the market demand for Player 2 is unchanged the horizontal lines are unchanged.

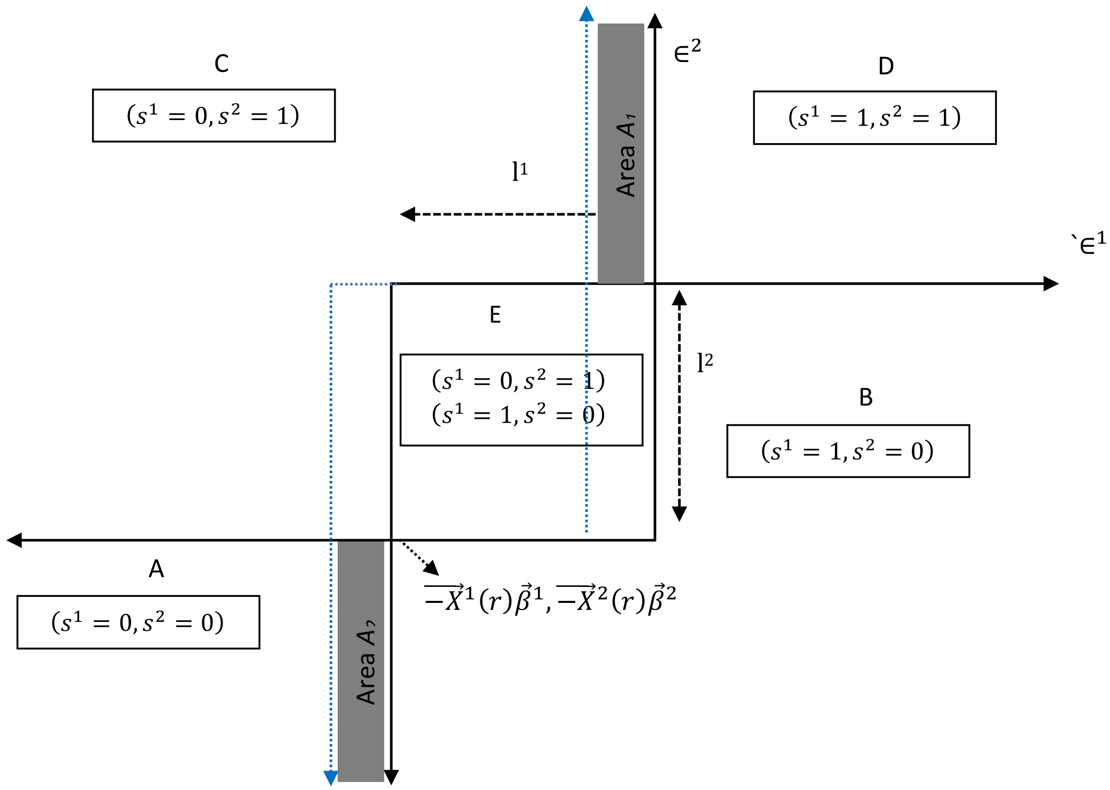


Figure 7.7. Effect on the likelihood regions of Nash-equilibria by increasing the operating cost of Player 2.

When market demand for Player 1 is increased area A_1 gets added to region D as observed in Figure 7.7. Unlike the operating cost where the number of routes with $(1,1)$ equilibrium being played remained zero, this results in a few routes with $(1,1)$ equilibrium. Figure 7.9 shows the number of routes where each of the four Nash equilibrium strategies was played as a function of market demand for Player 2. The effects are very symmetrical to the one where market demand of Player 1 was changed. Preference parameter for demand had a mean value of 3.11 for Player 1 and 0.47 for Player 2. Thus Player 1 has a relatively high preference coefficient for demand in comparison to Player 2. This result in an interesting phenomenon. Even though the number of routes where Player 1 exclusively operates or those with $(1,0)$

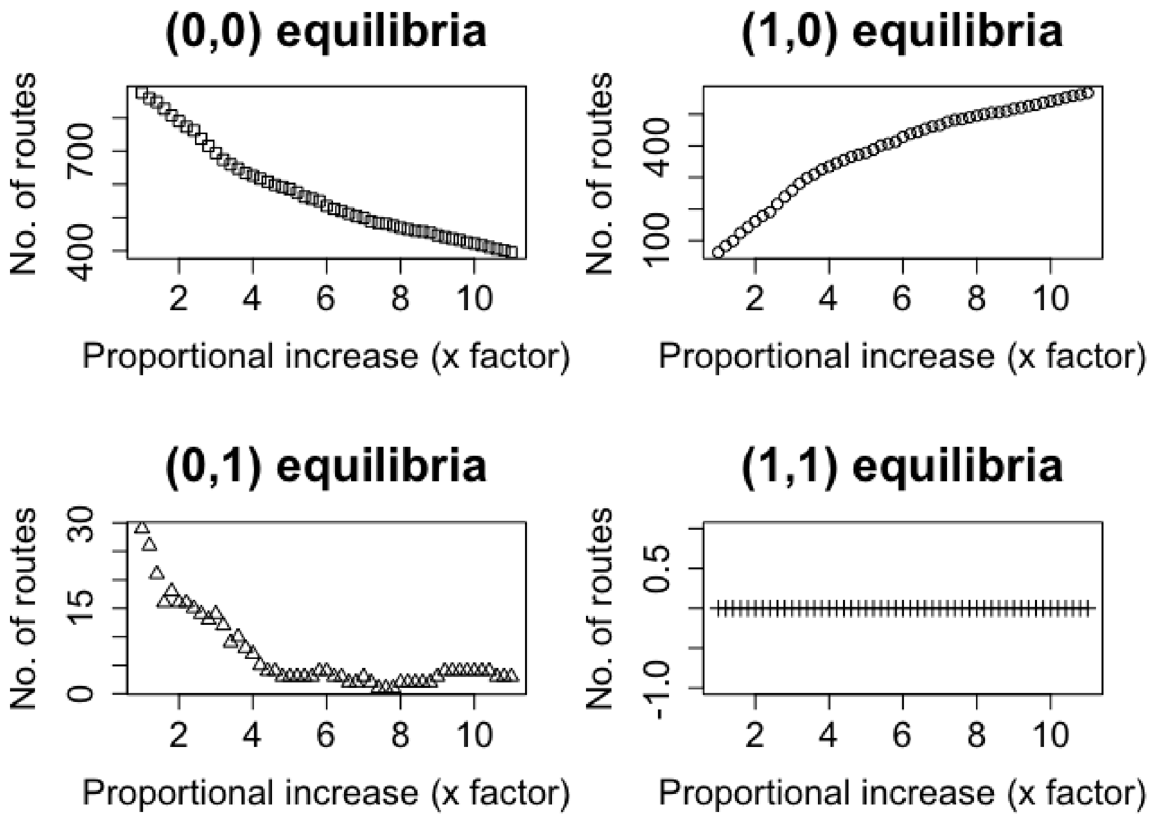


Figure 7.8. Number of routes in each Nash-equilibrium due to proportionally increasing the market demand of Player 1 (UA).

equilibrium decreases to zero as the market demand for Player 2 increases, Player 1 is able to compete and operate in a few routes i.e. there are a few routes with $(1, 1)$ equilibrium even when market demand associated with Player 1 is increased to very high values. This was not observed when market demand for Player 1 was increased. Player 1 unanimously dominated all the routes.

Figure 7.9 shows the number of routes where each of the four Nash equilibria strategies were played when the market demand of both the players are increased. Unlike all the previous cases, this result in a a large number of routes with $(1, 1)$ equilibrium. Number of routes with $(0, 0)$ equilibrium monotonically decreases with increasing market demand. An interesting observation is that, on increasing the mar-

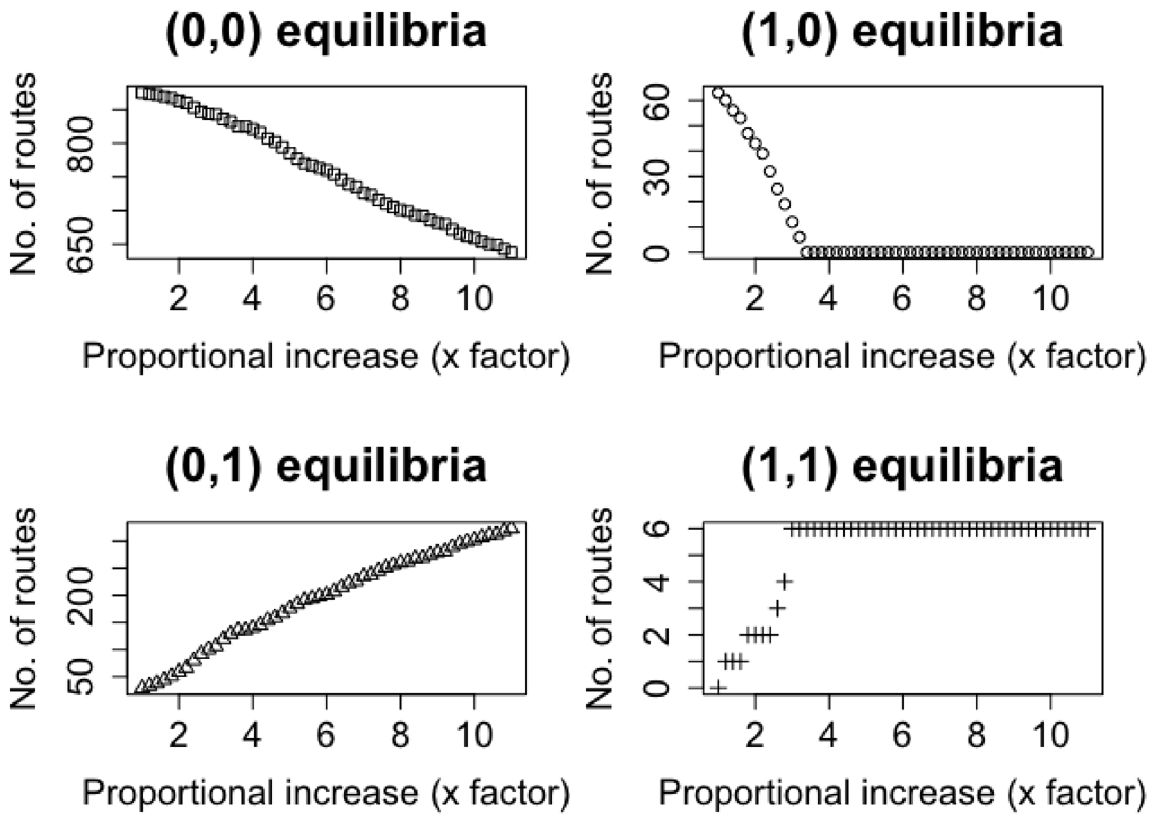


Figure 7.9. Number of routes in each Nash-equilibrium due to proportionally increasing the market demand of Player 2 (DL).

ket demand of both the player simultaneously number of routes with $(1, 0)$ equilibrium increases whereas those with $(0, 1)$ equilibrium decreases. This is because the Player 1 had a much higher preference coefficient for demand in comparison to Player 2 and therefore is able to attain higher gains from an increasing market demand in comparison to Player 2. This results in a large number of routes where only Player 1 operates.

From policy designers' standpoint, the only variable that can be directly influenced is operating cost. Non-stop distance is a geographic factor, which is fixed for every route. Demand is governed more by market factors and is difficult to be directly influenced. Through indirect approaches such as installing a cheap and fast alternate

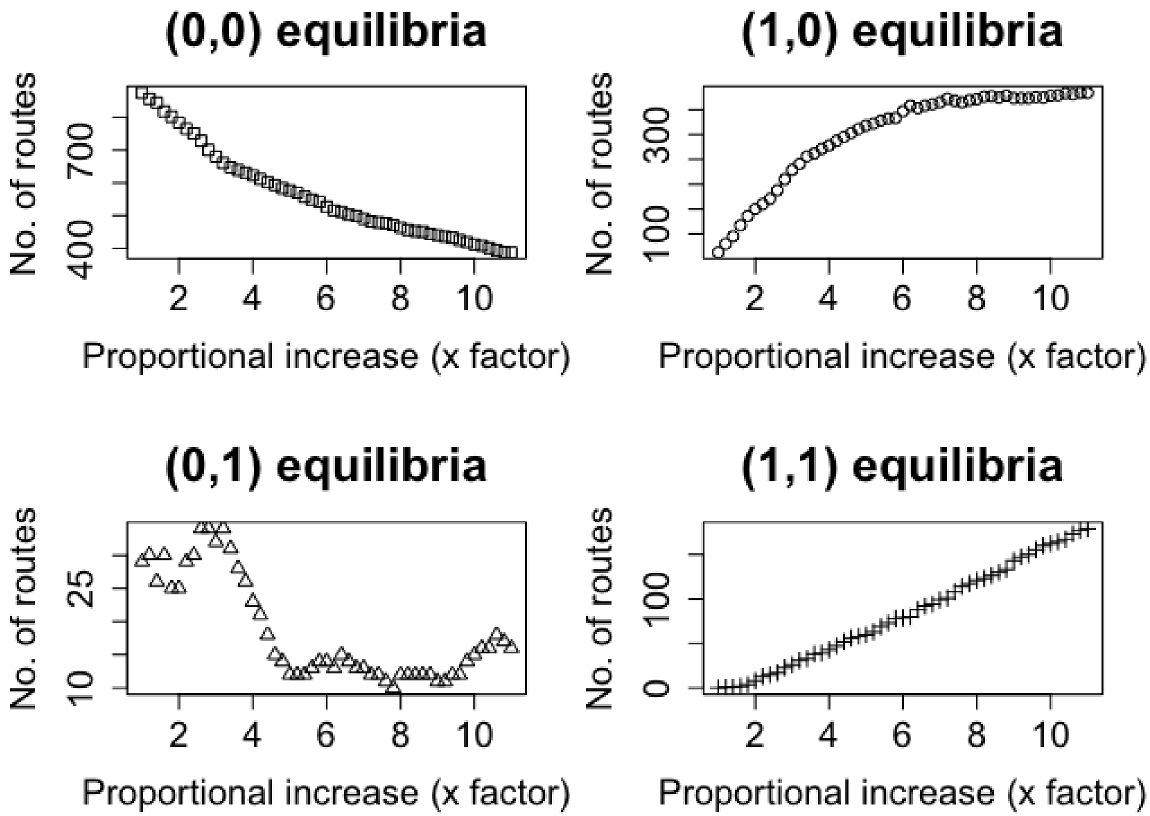


Figure 7.10. Number of routes in each Nash-equilibrium due to proportionally increasing the market demand of both the players.

mode of transportation between the origin and destination, existing demand for a route can be modified. However, it is much harder to directly influence. Cost is relatively easier to directly influence by levying taxes, imposing penalties for entering and operating service on congested routes, and airport regulation (such as single-till or dual-till [80]).

The evolutionary model developed here can be used to study how the network evolution can be influenced by influencing the decision parameters. We conducted simulation experiments by varying the cost and demand values of each route. The number of routes has direct implications on network properties such as robustness, resilience, and connectivity. A unit in x-axis indicates an increase in the operating cost or demand by one standard deviation of the network distribution. When the demand for UA (Player 1) is increased, fewer routes will have $(0, 0)$ as Nash-equilibrium. The number of routes where equilibria $(1, 0)$ is played increases whereas those where (0,1) is played decreases. Similarly, decreasing the cost of UA or decreasing the demand for DL (Player 2) decreases the number of routes where $(0, 0)$ and $(0, 1)$ equilibria are played. The number of routes was more sensitive to demand than cost as seen from these figures. Though the preference estimate for the cost (-0.74) had a larger magnitude than demand (0.05), varying demand by one-standard-deviation seemed to have a higher influence on the number of Nash-equilibria than cost. This is because the magnitude of demand is higher than cost.

7.3 Conclusions

Although the overall accuracy of the discrete choice model is higher, the discrete games-based approach is more accurate in predicting the airline decision routes where an addition or deletion was actually made. The latter accuracy is more important for the objective of the model which is to guide policymakers to understanding how airlines make route selection decisions.

8. CLOSURE

8.1 Summary of dissertation

In this dissertation, a mechanism design approach was established specifically for two different complex engineered systems: resource allocation in cloud-based design and manufacturing system and guide policymakers in air transportation system by understanding how airlines make decisions.

In CBDM, three different scenarios were analyzed from the viewpoint of strategic nature of the interacting agents. Based on the analysis and how well the requirements in the scenario match with the properties of the mechanism, best matching mechanisms were recommended for each of the three scenarios. The performance of these mechanisms differ in resource-scarce and resource-surplus conditions. Simulation studies were used to draw insights on the relative performance of the mechanisms under various resource conditions. A theoretical framework was established for optimal scheduling of the recommended matching mechanisms. The theoretical framework was made under certain simplifying assumptions such as deterministic arrival of service seekers and constant job-processing time. The effects of relaxing these assumptions on the optimal matching period were further studied.

In ATS, a discrete games based model was established to model the topological evolution of the network based on routing decisions made by the airlines. The parameters of the discrete games were estimated using MCMC numerical sampling techniques. While estimating those parameters the presence of non-unique Nash equilibrium creates additional challenges. An airport presence based tie-breaking rule was established to address this challenge. Prediction accuracies and ROC curves were used to validate the model. Based on the predictive model for network evolution, policy experiments were conducted to provide recommendations for policymakers.

8.2 Limitation and Opportunities for Future Work

Mechanism design was conventionally applied in economic systems. While extending the techniques and methods developed in one set of applications to another, this brings additional challenges and research gaps that needs to be addressed. While few of such challenges were addressed in this thesis there are still other research gaps that needs further investigation.

Resource allocation in CBDM offers a new set of challenges that need to be addressed. Most of these are mechanism design related specific issues that arises to inherent limitations or assumptions underlying existing mechanisms.

Firstly, existing bipartite matching mechanisms are based on the assumption that the alternatives of each participating agents are substitutes and not complements. In conventional economic applications this is a valid assumption. For example in kidney exchange two potential donors are substitutes from the standpoint of the receiver. Similarly, in matching students to school or residents to hospitals two different students are substitutes. Whereas in CBDM there are scenarios where the alternatives are complements instead of substitutes. For example the designer might be having multiple parts that needs to be printed and assembled to achieve the final objective. The utility of being matched to a machine-owner to get a sub-component in the assembly printed would depend on whether or not the designer was able to get matched to a suitable machine-owner to print other sub-components in the assembly. Here the alternatives are not clear substitutes. Another example, is that of a machine-owner trying to print multiple designs in the same run to save time and increase profit. There would be multiple designer alternatives offering designs that are complement to one another and could be printed simultaneously in the same run. In such a scenario, the designers are complements and not substitutes. Another research gap is to integrate the matching mechanisms along with the task of resource discovery.

Resource discovery is a challenging task in CBDM as it is impossible for all agents to provide an exhaustive list of their alternatives. One way of achieving this is to extract the general preference characteristics of individual agents towards attributes of their alternatives and based on how well the alternatives satisfy these requirement, use a utility-based approach to generate the preference rank ordering. However, some of the properties of existing bipartite matching mechanism breaks down when such an approach is used to elicit the preferences of the participants. For example, the deferred acceptance mechanism does not remain strategy-proof anymore when preferences are elicited in the form of preferences towards attributes instead of preferences towards alternatives. There are ways to game the system. There is a need to design strategy proof mechanisms when agents reveal attribute preferences instead of alternatives.

Another limitation of bipartite matching mechanisms is when there is exchange of transferable utility such as money. There are methods in which participating agents can collude to game the system. One such strategy is termed as ring-formation where a subset of agents exchange money between them for selfish gains [3].

A key assumption made in this thesis while modeling resource allocation in CBDM was that the resource are perfectly divisible. A designer or set of designers were matched to machine owners based on preferences and objectives of both the agents. However, there may be scenarios where resources are continuous. For example, if the resources are modeled as time-slots in the machines then this cannot be perfectly divided among the resource seekers. It would be interesting to explore extensions of techniques such as Knapsack auctions to account for this continuous nature of resources.

Another limitation of the approaches established in this thesis is in the optimal multi-period implementation of matching mechanisms. In the theoretical framework, we ignored the temporal variation in utility. We assumed that the utility attained by designers on being matched to a machine owner does not change by the time the designer is matched to a particular machine owner. For example, if the designer needed to get the part printed urgently and was not matched by a specific date then the utility drops to zero after the desired date. In other cases, the utility might exponentially decay with waiting time. We ignored this temporal fluctuation. This is a valid assumption in CBDM application as the time scales of implementation are small. However, if the framework that we established is extended into other applications such as kidney exchange program then it is important to analyze the effects of relaxing the assumption. One effect of such temporal variations in utility is the effect that several desirable properties of the mechanism breaks down. For example, individually rational mechanisms such as Top-Trading Cycle or Defferred Acceptance cease to be individually rational since the designer might be better off not being matched if the utility drops to zero. It will be an interesting to analyze the interplay between satisfaction of properties and optimality in matched objectives in applications where temporal variations are significant.

Addressing such limitations not only enables effective application of the mechanism design principles in engineering systems but also result in further advancement of the scope of existing mechanisms. For example, for resource allocation in CBDM the bipartite matching mechanisms needed to be implemented multiple times and the scheduling the period of interval at which the matching mechanism are implemented was important. However, there are no existing studies on optimal scheduling of multi-period implementation of matching mechanisms. In this thesis we established a theoretical framework for optimal scheduling of such mechanisms. The results and insights are highly generic and generalizes into applications even outside of CBDM if the mechanisms are implemented repeatedly in a multi-period fashion. Similarly, the policy recommendations for airlines provides valuable insights while developing network policies for other complex network that evolves based on decisions made by competing agents.

In ATS, a limitation on the discrete-games based predictive model that we developed is that we assume that the decision made on a given route in the current period is independent of the decisions made on other routes in the current period. However, this should not be confused with decisions made on other routes being ignored completely. The decisions on other routes is considered in our approach through the network level parameter called airport presence. Airport presence of a route indicates the average degree of the terminal airports of that route. This shows how well connected the route is and this has an influence on the route decision. But this effect is accounted for based on the state of the network in the previous period and not based on decisions on other routes in current period. This is a reasonable assumption as addition or deletion happens only in less than 10 % of the routes from one period to the next based on BTS data [18]. Therefore, the network properties like connectivity, centrality do not change drastically from one period to the next.

The discrete games model used in this thesis considers only competition between two players. This is not a limitation of the approach. There are theoretical frameworks for discrete games model for more than two players. It would be interesting to construct airport presence sampling techniques to solve for those parameters. The interaction effects between airlines when more than two airlines are involved is also a potential area for further investigation.

REFERENCES

REFERENCES

[1] Ali A. Minai. Chapter 1 complex engineered systems: A new paradigm.

[2] A. E. Roth, T. Sonmez, and M. U. Unver. Kidney exchange. *The Quarterly Journal of Economics*, 119(2):457–488, may 2004.

[3] Robert W Irving Dan Gusfield. The stable marriage problem: Structure and algorithms. *Concurrent Engineering*, 1989.

[4] Tim Roughgarden. Preface. In *Twenty Lectures on Algorithmic Game Theory*, pages xi–xiv. Cambridge University Press.

[5] Eric S Maskin. Mechanism design: How to implement social goals. *American Economic Review*, 98(3):567–576, may 2008.

[6] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[7] 3Dhubs. 3Dhubs kernel description. `https://www.3dhubs.com/`, 2015. Accessed: 2015-05-11.

[8] Adam. Recommended Hubs - How they are sorted. `https://www.3dhubs.com/talk/thread/recommended-hubs-how-they-are-sorted/`, 2016. [Online; accessed 28-July-2018].

[9] Dazhong Wu, David W. Rosen, Lihui Wang, and Dirk Schaefer. Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation. *Computer-Aided Design*, 59:1–14, feb 2015.

[10] Shapeways. Shapeways. `http://www.shapeways.com/`, 2015. Accessed: 2015-05-11.

[11] Filemon. WE JUST HIT 25,000 HUBS. `https://www.3dhubs.com/talk/thread/we-just-hit-25000-hubs/`, 2015. [Online; accessed 28-July-2018].

[12] Wayne E Smith. Various optimizers for single-stage production. *Naval Research Logistics (NRL)*, 3(1-2):59–66, 1956.

[13] Edward J Anderson and Chris N Potts. On-line scheduling of a single machine to minimize total weighted completion time. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 548–557. Society for Industrial and Applied Mathematics, 2002.

[14] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1-2):166–196, 2001.

[15] Birgit Heydenreich, Rudolf Müller, and Marc Uetz. Mechanism design for decentralized online machine scheduling. *Operations research*, 58(2):445–457, 2010.

[16] Atila Abdulkadiroğlu, Parag A Pathak, and Alvin E Roth. The new york city high school match. *American Economic Review*, 95(2):364–367, may 2005.

[17] Jiaoe Wang, Huihui Mo, Fahui Wang, and Fengjun Jin. Exploring the network structure and nodal centrality of china's air transport network: A complex network approach. *Journal of Transport Geography*, 19(4):712–721, jul 2011.

[18] Air carriers: T-100 domestic market (u.s. carriers). `http://www.transtats.bts.gov/Fields.asp?Table_ID=258`. Accessed: 2017-10-11.

[19] Downloadable Route maps. `https://www.delta.com/content/www/en_US/traveling-with-us/where-we-fly/routes/downloadable-route-maps.html/`, 2018. [Online; accessed 28-July-2018].

[20] Fred L Smith Jr and Braden Cox. Airline deregulation. *The Concise Encyclopedia of Economics. Liberty Fund, Inc. Available at http://www. econlib. org/library/Enc/AirlineDeregulation. html*, 2008.

[21] United Airlines, route map. `https://www.metabunk.org/united-airlines-1966-route-map-video.t6737/`, 2015. [Online; accessed 28-July-2018].

[22] Fei Tao, Yefa Hu, Dongming Zhao, and Zude Zhou. Study on resource service match and search in manufacturing grid system. *The International Journal of Advanced Manufacturing Technology*, 43(3-4):379–399, sep 2008.

[23] Fei Tao, Yuanjun LaiLi, Lida Xu, and Lin Zhang. FC-PACO-RM: A parallel method for service composition optimal-selection in cloud manufacturing system. *IEEE Transactions on Industrial Informatics*, 9(4):2023–2033, nov 2013.

[24] Fei Tao, Lin Zhang, and A.Y.C. Nee. A review of the application of grid technology in manufacturing. *International Journal of Production Research*, 49(13):4119–4155, jul 2011.

[25] Fei Tao, Ying Cheng, Li Da Xu, Lin Zhang, and Bo Hu Li. CCIoT-CMfg: Cloud computing and internet of things-based cloud manufacturing service system. *IEEE Transactions on Industrial Informatics*, 10(2):1435–1442, may 2014.

[26] Malte Gebler, Anton JM Schoot Uiterkamp, and Cindy Visser. A global sustainability perspective on 3d printing technologies. *Energy Policy*, 74:158–167, 2014.

[27] Anderson Chris. Makers: The new industrial revolution. *New York: Crown Business*, 2012.

[28] Ralph L. Keeney and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press, 1993.

[29] M. G. Fernandez, Carolyn Conner Seepersad, David W. Rosen, Janet K. Allen, and Farrokh Mistree. Decision support in concurrent engineering - the utility-based selection decision support problem. *Concurrent Engineering*, 13(1):13–27, mar 2005.

[30] Onur Kesten. On two competing mechanisms for priority-based allocation problems. *Journal of Economic Theory*, 127(1):155–171, mar 2006.

[31] Leonid Hurwicz and Stanley Reiter. *Designing Economic Mechanisms*. Cambridge University Press (CUP), 2006.

[32] D. Gale and L. S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9, jan 1962.

[33] D. G. McVitie and L. B. Wilson. The stable marriage problem. *Communications of the ACM*, 14(7):486–490, jul 1971.

[34] Vincent P. Crawford and Elsie Marie Knoer. Job matching with heterogeneous firms and workers. *Econometrica*, 49(2):437, mar 1981.

[35] Lloyd Shapley and Herbert Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1(1):23–37, mar 1974.

[36] Atila Abdulkadiroğlu and Tayfun Sönmez. School choice: A mechanism design approach. *American Economic Review*, 93(3):729–747, jun 2003.

[37] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, mar 1957.

[38] Thingiverse. http://www.thingiverse.com/, 2016. Accessed: 2016-03-27.

[39] iMaterialise. imaterialise. `http://i.materialise.com`, 2015. Accessed: 2016-03-27.

[40] Michel Balinski and Tayfun Sönmez. A tale of two mechanisms: Student placement. *Journal of Economic Theory*, 84(1):73–94, jan 1999.

[41] Haluk I. Ergin. Consistency in house allocation problems. *Journal of Mathematical Economics*, 34:77–97, 1999.

[42] Lars Ehlers and Bettina Klaus. Resource-monotonicity for house allocation problems. *International Journal of Games Theory*, 32(4), aug 2004.

[43] Lars Ehlers, Bettina Klaus, and Szilvia Pápai. Strategy-proofness and population-monotonicity for house allocation problems. *Journal of Mathematical Economics*, 38(3):329–339, nov 2002.

[44] Atila Abdulkadiroglu and Tayfun Sonmez. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689, may 1998.

[45] J. Von Neumann and O. Morgenstern. *The Theory of Games and Economic Behavior*. NJ: Princeton University Press, 1947.

[46] Fuhito Kojima. When can manipulations be avoided in two-sided matching markets? – maximal domain results. *The B.E. Journal of Theoretical Economics*, 7(1), jan 2007.

[47] N. Mandhan, Thekinen J., A. Lo, and Panchal J.H.Wu. Matching designers and 3d printing service providers using gale-shapley matching. In *Tools and Methods of Competitive Engineering (TMCE 2016)*, May 9-13, 2016.

[48] Joseph Thekinen and Jitesh H. Panchal. Resource allocation in cloud-based design and manufacturing: A mechanism design approach. *Journal of Manufacturing Systems*, 43:327–338, apr 2017.

[49] George Christodoulou, Elias Koutsoupias, and Annamária Kovács. Mechanism design for fractional scheduling on unrelated machines. In *International Colloquium on Automata, Languages, and Programming*, pages 40–52. Springer, 2007.

[50] Navendu Jain, Ishai Menache, Joseph Seffi Naor, and Jonathan Yaniv. A truthful mechanism for value-based scheduling in cloud computing. *Theory of Computing Systems*, 54(3):388–406, 2014.

[51] Nir Andelman, Yossi Azar, and Motti Sorani. Truthful approximation mechanisms for scheduling selfish related machines. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 69–82. Springer, 2005.

[52] Ilya M Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280, 2001.

[53] Edward G Thurber. Concerning the maximum number of stable matchings in the stable marriage problem. *Discrete Mathematics*, 248(1-3):195–219, 2002.

[54] Senvol. http://senvol.com/database/, 2018. Accessed: 2018-03-07.

[55] Ultimaker Cura. `https://ultimaker.com/en/products/ultimaker-cura-software/`, 2018. Accessed: 2018-08-02.

[56] Peter Belobaba. The airline planning process. *The Global Airline Industry*, pages 153–181, 2009.

[57] T Kushige. A solution of most profitable aircraft routing. In *AGIFORS Symposium Proceedings*, volume 3, page 1963, 1963.

[58] HD De Lamotte and DFX Mathaisel. Experience with mpsx-mip for olympic airways fleet assignment problems. *MIT Flight Transportation Laboratory Memorandum*, 1983.

[59] Tatsuya Kotegawa. *Analyzing the evolutionary mechanisms of the air transportation system-of-systems using Network Theory and machine learning algorithms.* PhD thesis, Purdue University, 2012.

[60] Zhenghui Sha, Kushal Moolchandani, Jitesh H. Panchal, and Daniel A. DeLaurentis. Modeling airlines' decisions on city-pair route selection using discrete choice models. *Journal of Air Transportation*, 24(3):63–73, jul 2016.

[61] Timothy F. Bresnahan and Peter C. Reiss. Empirical models of discrete games. *Journal of Econometrics*, 48(1-2):57–81, apr 1991.

[62] Federico Ciliberto and Elie Tamer. Market structure and multiple equilibria in airline markets. *Econometrica*, 77(6):1791–1828, 2009.

[63] Steven T Berry. Airport presence as product differentiation. *The American Economic Review*, 80(2):394–399, 1990.

[64] Siddhartha Chib and Edward Greenberg. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4):327–335, 1995.

[65] Patrick Jaillet, Gao Song, and Gang Yu. Airline network design and hub location problems. *Location science*, 4(3):195–212, 1996.

[66] Anantaram Balakrishnan, T William Chien, and Richard T Wong. Selecting aircraft routes for long-haul operations: a formulation and solution method. *Transportation Research Part B: Methodological*, 24(1):57–72, 1990.

[67] Manoj Lohatepanont and Cynthia Barnhart. Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science*, 38(1):19–32, 2004.

[68] George B Dantzig. Linear programming and extensions. princeton landmarks in mathematics and physics. *Princeton University Press*, 1963.

[69] Kisun Song, Jung-Ho Lewe, and Dimitri Mavris. A multi-tier evolution model of air transportation networks. In *14th AIAA Aviation Technology, Integration, and Operations Conference*, page 3267, 2014.

[70] Charles Boguslaski, Harumi Ito, and Darin Lee. Entry patterns in the southwest airlines route system. *Review of Industrial Organization*, 25(3):317–350, 2004.

[71] Zhenghui Sha, Kushal A. Moolchandani, Apoorv Maheshwari, Joseph Thekinen, Jitesh Panchal, and Daniel A. DeLaurentis. Modeling airline decisions on route planning using discrete choice models. In *15th AIAA Aviation Technology, Integration, and Operations Conference*. American Institute of Aeronautics and Astronautics, jun 2015.

[72] Kushal Moolchandani, Zhenghui Sha, Apoorv Maheshwari, Joseph Thekinen, Navindran Davendralingam, Jitesh Panchal, and Daniel A. DeLaurentis. Hierarchical decision-modeling framework for air transportation system. In *16th AIAA Aviation Technology, Integration, and Operations Conference*. American Institute of Aeronautics and Astronautics, jun 2016.

[73] Steven T Berry. Estimation of a model of entry in the airline industry. *Econometrica: Journal of the Econometric Society*, pages 889–917, 1992.

[74] Shinya Sugawara and Yasuhiro Omori. Duopoly in the japanese airline market: bayesian estimation for the entry game. *The Japanese Economic Review*, 63(3):310–332, 2012.

[75] Severin Borenstein. Hubs and high fares: dominance and market power in the us airline industry. *The RAND Journal of Economics*, pages 344–365, 1989.

[76] Michael E Levine. Airline competition in deregulated markets: theory, firm strategy, and public policy. *Yale J. on Reg.*, 4:393, 1986.

[77] Passenger boarding (enplanement) and all-cargo data for u.s. airports). `https://www.faa.gov/airports/planning_capacity/passenger_allcargo_stats/passenger/`. Accessed: 2017-10-11.

[78] Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

[79] Moshe E Ben-Akiva and Steven R Lerman. *Discrete choice analysis: theory and application to travel demand*, volume 9. MIT press, 1985.

[80] Achim I. Czerny. Price-cap regulation of airports: Single-till versus dual-till. *Journal of Regulatory Economics*, 30(1):85–97, jul 2006.

APPENDICES

VITA

VITA

Joseph D Thekinen was born in 1991 in Kochi- a beautiful city in a small state Kerala, India. He did his high school at Viswajyothi CMI Public School, Ankamaly. He received his senior secondary education from two different schools, St. Antony's Public School in Kerala (11^{th} grade) and Maharishi Vidya Mandir in Tamil Nadu (12^{th} grade). He went to Indian Institute of Technology, Kharagpur for his undergraduate and masters studies. Immediately, after he completed his undergraduate education he came to Purdue University for his PhD in Mechanical Engineering. He worked in the Design Engineering Lab at Purdue (DELP). His graduate research and studies were on designing mechanisms for various complex engineered systems. After PhD, Joseph would like to continue his research on mechanism design and other specific challenges in systems engineering in an academic position at a university.