Pairwise Stability in Two-Sided Matching Market with General Increasing Functions

by

Asma Javaid

School of Natural Sciences

National University of Sciences and Technology

Islamabad, Pakistan

June 2014

Pairwise Stability in Two-Sided Matching Market with General Increasing Functions

Thesis submitted towards the degree of

Master of Philosophy in Mathematics

by

Asma Javaid

This research was carried out in

School of Natural Sciences

National University of Sciences and Technology

Islamabad, Pakistan.

Supervised by

Dr. Yasir Ali

June 2014

Dedicated to my mother

Abstract

Two-sided matching problem is well known in mathematical economics. The primary objective of two-sided matching problem is the formation of stable partnership between set of participants. The marriage model and the assignment game are considered two standard models in the theory of two-sided matching problems. Both models are given as two disjoint sets of participants, M and W, and have to match or assign the participants of M to the participants of W. The goal is to find a stable matching between the participants of M and the participants of W. The exact definition of what constitutes a stable matching depends on the model under consideration. Participants in the marriage model are rigid because no participant will negotiate on side payments. While, in assignment game participants are flexible because side payment are permitted between participants of both sets. In this thesis, we generalize a one-to-one matching model from linear valuation functions to non-linear valuation functions and bounded side payments. We use algorithmic approach to show the existence of pairwise stable outcome for our one-to-one matching model. Our model includes discrete one-to-one matching model of Ali and Farooq.

Acknowledgement

First of all, I humbly bow my head before ALMIGHTY ALLAH. Then, I would like to express my sincere gratitude to my supervisor Dr. Yasir Ali for his continuous support, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. Without his continuous support, encouragement and guidance the completion of this thesis was just a dream. I really acknowledge him for all of his help and patience.
Besides my supervisor, I would like to thank the rest of my thesis committee: Dr. Rashid Farooq, Dr. Quanita Kiran, Dr. Mujeeb-ur-Rahman for their encouragement, insightful comments, and hard questions.
Lastly, I would like to thank my mother for supporting and providing me a full opportunity of devoting myself for M.Phil studies. In the end I will definitely thanks my fellow mates specially my friend Mehwish for her manual support, strength, and help.

Asma Javaid

Preface

Over the last few decades, a lot of research has been done in the two-sided matching problem by a large number of scholars. In a two-sided matching problem, the set of participants are divided into two disjoint sets, say M and W. The primary objective of two-sided matching problem is the formation of stable partnership between the participants of M and W. A matching X, is a one-to-one correspondence between the participant of one set to the participant of other set. Main requirement in a two-sided matching problems is that of stability of matchings.

The concept of finding two-sided stable matching was first given by Gale and Shapley [7] in their paper *"College Admissions and the Stability of Marriage"*. In the course of presenting an algorithm for matching applicants to college places, they introduced and solved stable marriage problem. This problem consists two disjoint sets of participants M and W. Each participant ranks a subset of other set of participants in order of preferences. The aim is to form a one-to-one matching X of the participants such that no two participants would prefer each other to their partner in X. The authors used their solution to this problem as a basis for solving the extended problem where one of the sets consists of college applicants, and the other consists of colleges, each of which has a quota of places to fill. The monetary transfer is not permitted in their model. For this reason, participants in this model are called rigid. Many additional variants of the stable marriage problem have been discussed in the literature. Guessfield and Iriving [8] published a book that covers many variants of original stable marriage problem such as if the preferences of participants may include ties, incomplete preferences and roommate problem.

Shapley and Shubik [14] presented the one-to-one buyer-seller model known as *"assignment game"*. In their model participants are flexible, because exchange of money is permitted among participants of both sets. Each participant on one side can supply exactly one unit of some indivisible good and exchange it for money with a participant from the other side whose demand is also one unit. Shapley and Shubik [14] showed that the core of the game is always non-empty and can be identified with the set of stable outcomes, which is a solution set based upon a linear programming formulation of the model.

After this, two-sided matchings have been studied extensively. Different approaches have been made by many researchers in which they generalize the marriage model of Gale and Shapley [7] and assignment game of Shapley and Shubik [14]. Main aim of these researchers was to find common result for both of [7] and [14] models in a more general way. Erikson and Karlinder [5] and Sotomayor [15] presented the hybrid models. These models are the generalization of the discrete marriage model [7] and the continuous assignment game [14]. Existence of stable outcome and the core is discussed in [5, 15]. Farooq [6] presented a one-to-one matching model in which he identified the preferences of participants by strictly increasing linear functions. He proposed an algorithm to show the existence of pairwise stable outcome in his model by taking money as a continuous variable. His model includes the marriage model of Gale and Shapely [7], assignment game of Shapely and Shubik [14] and Erikson and Karlander [5] hybrid model as special cases. The motivation of our work from the stable matching literature is the model of Ali and Farooq [1]. Ali and Farooq [1] presented a one-to-one matching model by taking money as a discrete variable in linear increasing function. They designed an algorithm to show that pairwise stable outcome always exists. The complexity of Ali and Farooq's [1] algorithm depends on the size of those intervals where prices fall. Our model is the generalized form of Ali and Farooq [1] model. We consider the preferences of participants by general increasing functions that are non-linear in the whole real line and showed that a pairwise stable outcome also exists in our model.

In Chapter 1, we give some basic introduction of two-sided matching problems and complexity of algorithms. The second chapter constitutes a brief review of the two-sided matching problems: the marriage model due to Gale and Shapley [7], the assignment game due to Shapley and Shubik [14] and Ali and Farooq [1] model. An example is also included to understand these models and working of algorithm.

In Chapter 3, we extend the model of Ali and Farooq [1] by generalizing valuation functions. We also discuss correctness and termination of the algorithm.

Contents

Chapter 1

Preliminaries

1.1 Introduction

Stable matching theory in two-sided market, the term "two-sided" refers to the fact that participants in such markets belong to two disjoint sets, has been extensively employed across various disciplines to deal with numerous economical problems. These economical problems include formation of marriages among unmatched individuals, interaction between unemployed workers and vacancies opened by organizations or dealing between buyers and sellers in public sale. The matching markets have as primary object; the formation of partnerships. Stable matching algorithms are also a hotspot for research. These algorithms connect the two major branches of studies that are algorithmic theory and combinatorial optimization. Graph theory is one of the most important tools to analyze these algorithms which already mark an extensive level of use in combinatorial optimization. In this chapter, our focus is to accommodate two conflicting goals. On one side, we believe that an introductory text should be lean and concentrate on the essential, so as to offer guidance to those who are new in this field. On the other hand, it has been my particular concern to write with sufficient detail to make the text easy to read and understand. Brief explanation of two-sided stable matching and basic definitions related to it will be given in section 1.2. In section 1.3, we will discuss the algorithms and their complexities.

1.2 Matching and Stability

In this section, we will discuss stability in a two-sided matching market. In a two-sided matching market, we have to find stable matching between two disjoint sets of participants M and W, in which each participant of M has strict preferences over each participant of other set. *Preferences* of participants represent the choice that he or she would make when faced with different alternatives. A participant $m \in M$ is said to *prefer* $w \in W$ to $w' \in W$ if he chooses w when faced with a choice between the two. Thus, we can say that stability in a two-sided matching market depends upon the preferences of the participants.
A matching in a two-sided market is said to be stable when the following two conditions are not satisfied:

(i) some given participant A of a matched pair prefers a participant B of the other matched pair over the participant to which A is already matched and,

(ii) B also prefers A over the participant to which B is already matched.

In other words, a matching X is said to be *stable* if there does not exist any pair (A, B) in which A and B are not matched to each other under the matching X, but both A and B prefers one another to their current partner. Such a pair is called *blocking pair*. In order to have a stable matching between the participants of two disjoint sets we have to avoid the formation of blocking pair. In other words, we can say that a criteria of stability in a two-sided matching is that there exists no blocking pair in a matching. Here, we will discuss some properties of preferences that will be helpful in the subsequent parts of this thesis.
Consider two disjoint and finite sets of participants, say men M and women W, where

$$M = \{m_1, m_2, \ldots, m_r\},$$

and

$$W = \{w_1, w_2, \ldots, w_s\}.$$

Participants of these two sets have a preference list over each other. Preference list shows that the participant of one set has a choice to select the most preferable one when faced with different alternatives. Generally, the preference list of each $m \in M$ is denoted by $Pl(m)$, and can be represented as rank order lists of the form $Pl(m) = w_3, w_2, m, \ldots, w_4$, denoting that $m's$ first choice is w_3, second choice is w_2, third is to remain unmatched. Participants that appear after m in the preference list of m are not acceptable for m. Hence m likes to marry his first choice w_3, if w_3 is not available for m then m will select w_2 if possible otherwise he will like to remain unmatched. Thus, we can say that in the preferences list of m, woman w is said to be *acceptable* for m if m likes her at least as well as remaining single. If m prefers to remain single than being matched to women w then w is said to be unacceptable for m, and $m's$ preference only include the acceptable partners. We use $>_i$ to denote the ordering relationship of participants (on either side of the market),that is, if $w >_m w^{'}$ then it means that in the preference list of m participant m prefers w to $w^{'}$, and $w \geq w^{'}$ means that m prefers w at least as well as $w^{'}$ or is *indifferent* between them. Now, if the preference list of $m \in M$ is given by

$$Pl(m) = w_3, [w_2, w_1], m, \ldots, w_4.$$

Then this list shows that $m's$ likes to match with w_3 first and likes both w_2 and w_1 but is indifferent between them or in other words he has tied preferences over w_2 and w_1. Third preference of m shows that man m prefers to remain single. If a participant is not indifferent between any two acceptable mates, then we will say that he/she has *strict preferences*. Similar assumptions are for $Pl(w)$.

Since we have two disjoint set of participants each participant of these two sets have complete and transitive preference ordering over each other. *Transitive preferences* means that if man m likes w_1 at least as well as w_2, and m likes w_2 at least as well as w_3 then m likes w_1 at least as well as w_3. *Complete ordering* means that any two participants can be compared but the participant is never confronted with the choice that he is unable to make. When the preferences of the participants form a compete ordering and are transitive then these participants are called *rational*.

A *matching* $X : M \cup W \rightarrow M \cup W$ is a one-to-one correspondence between set of participants such that $m = X(w)$ if and only if $w = X(m)$, read as under matching X if w is matched to m then m is matched to w. Or we can say that m and w are matched with each other under matching X. If $m = X(m)$ then m is said to be unmatched or self-matched under the matching X. Here w and m are said to be *mutually acceptable* if both are acceptable for each other. It is not always possible to match all the participants. If the individual does not get the participant of his or her choice then he or she are likely to remain unmatched instead of getting the participant that he or she doesn't like.

A *matching* X is blocked by a pair of participants (m, w) if they prefer each other to the partner they receive under the matching X, that is,

$$w >_m X(m) \quad \text{and} \quad m >_w X(w) \quad \text{but} \quad (m, w) \notin X.$$

Such kind of pairs is called a blocking set in general. If there is a blocking set in the matching, the participants involved have an incentive to break up and form new marriages. Therefore such an "unstable" matching is not desirable.

1.2.1 One-Sided and Two-Sided Matching Market

In the above section, we had discussed that in two-sided matching participants of both sets have a list of preferences over each other and each participant has a consent to match his self with a participant of other set. Two-sided matching market usually includes labor market in which the participants of one-side (men, worker, and buyer) need to be matched with participants on the other side (women, firm, and seller) and participants of both sides have preferences over each other. This two-sided structure allows us to draw strong conclusions about the matching mechanism and properties of matching.

There also exists matching markets that are one-sided in which our goal is to allocate n items to n participants in such a manner that each participant have a complete preference list and a unit demand over the items. For example, when people need to be assigned rooms in a dormitory, or places in a public school that does not itself has preferences or strategic actions. These markets matches people to places, but only on one side people are active participants in this market. Since the preferences are complete, so there exists truthful (strategy proof) mechanisms in which participants do not have an incentive to misrepresent their preferences. Sometimes one-sided matching markets have such allocation mechanisms that do not involve monetary compensations/payments among participants.

Some markets can also be hybrid, with both two and one sided properties. *Hybrid markets* are those that contain both rigid and flexible participants. Rigid in the sense that monetary transfer is not allowed between participants and in flexible monetary transfer is allowed between participants of both sets. Now, we will give an example of stable matching in which we have two sets, M and W, such that the participants are mutually acceptable and they have strict preference list.

Example 1.2.1. Consider two disjoint sets of participants

$$M = \{a, b, c, d\},$$

and

$$W = \{\alpha, \beta, \gamma, \delta\}.$$

The preference list of these participants is defined in Table 1.1.

According to this preference list we have three matchings in which two are stable

Men's list	*Women's list*
$Pl(a) : \beta, \delta, \alpha, \gamma$	$Pl(\alpha) : b, a, d, c$
$Pl(b) : \gamma, \alpha, \delta, \beta$	$Pl(\beta) : d, c, a, b$
$Pl(c) : \beta, \gamma, \alpha, \delta$	$Pl(\gamma) : a, d, c, b$
$Pl(d) : \delta, \alpha, \gamma, \beta$	$Pl(\delta) : b, a, d, c$

Table 1.1: An instance of size $k = 4$

and one is not stable. We denote these three matchings as X, X^* and X^{**} where

$$X = \{(a, \delta), (b, \gamma), (c, \beta), (d, \alpha)\},$$

$$X^* = \{(a, \alpha), (b, \gamma), (c, \beta), (d, \delta)\},$$

$$X^{**} = \{(a, \delta), (b, \alpha), (c, \beta), (d, \gamma)\}.$$

The matching X is stable because it contains no blocking pair. However a could form a blocking pair if a is matched with β because β prefers c to a. Matching X^*

is not stable because $X^*(a) = \alpha$ but in the preference list of a we have $\delta > \alpha$ this implies $\delta > X^*(a)$. Also $X^*(\delta) = d$ in matching X^* but in the preference list of δ we have $a > d$ this implies $a > X^*(\delta)$. This is a contradiction so (a, δ) forms a blocking pair. *Matching* X^{**} is also stable. Thus, we conclude that X and X^{**} are only stable matchings with this preference list but there may be many unstable matchings other than X^*.

Remark: Since in this example we have same number of men and women so we have total $4! = 24$ matchings in which only two are stable and all of the rest are unstable. Since each men and women appear only once in these matchings so it is called marriage.

1.2.2 The Optimal Assignment Game

Assignment game is related to graph matchings particularly bipartite graphs. If in a bipartite graph (U_1, U_2, E) each edge is assigned a non-negative weight $w(e)$, here $w(e)$ is the weight of each edge in a bipartite graph, then the optimal assignment problem is the problem of finding a matching X such that the sum of the weight of the matching X is as large as possible. Here, we have assumed that the weight of an edge is a non-negative integer, that is, $w(e) > 0$. The weight matrix of the bipartite graph $K_{n,n} = (U_1, U_2, E)$ is denoted as $A = [a_{ij}]$ where a_{ij} represents the weight of the edge joining vertex i in U_1 to vertex j in U_2. Thus, an optimal assignment problem consists of the solution of n such elements from the matrix such that

(i) no two selected elements lie in the same row or same column,

(ii) the sum of the n selected entries is as large as possible.

The selection of n such elements then defines an optimum matching X, also known as an optimal assignment .

The core of the game between two disjoint set of participants is an optimal assignment. A matching X is called *Pareto Optimal* if there exists no matching X' such that some participant prefers X' to X and no participant prefers X to X'.

1.3 Complexity Theory

In this section, we will analyze algorithms in graphs. We will concentrate more on graph-theoretical aspects of these algorithms than on their actual implementation on a computer. Sometimes we will not be able to prove the best possible complexity of an algorithm. So, in most such cases we will provide reference to a better complexity. Also, some very basic notation of data structure and algorithms are required and will be given below.

1.3.1 Algorithm

There are many general classes of problems that arise in game theory and discrete mathematics. To solve such general classes of problems we first need to construct the model that translates any given problem into a mathematical context. Moreover, to find solution of such problems a method is needed that solves the general problem using the defined model. This method is generally called algorithm because it contains a sequence of steps that leads us to the desired answer. Thus, a procedure that contains finite set of precise instructions for performing computational tasks or for solving a problem is called *algorithm*. There exists several ways of defining algorithm but simplest one is to use the English language to describe the sequence of steps. In this thesis we will also define the algorithms in this way.

1.3.2 Complexity of Algorithm

Every algorithm has its complexity. Complexity of an algorithm is the efficiency of algorithm or the running time of algorithm. There exists many methods that check the running time of algorithm and one of them is to check how much time computer is taking to solve a particular problem when input values are of specified size. An analysis of the time required to solve a problem of particular size involves the time complexity of an algorithm that depends on the size of its input. The time complexity of an algorithm is the maximum number of computational steps used by the algorithm when the input has a particular size. Time complexity is often expressed as $O(f)$ where f is a function of the given size of input and $O(f)$

denotes the set of functions g such that $|g(x)| \leqslant c|f(x)|$ here $|x| \geqslant a$ and a, c are constants. When function f is quadratic polynomial we typically abuse notation by writing $O(n^2)$ instead of $O(f)$ to describe function that grows at most quadratically in terms of n. Here big O notation estimates the number of operations an algorithm uses when size of input grows. Moreover, by using big O notation we can compare two algorithms to determine which one is more efficient if the input size grows. Also big O notation often describes the growth of function, but it has limitations. In particular, when we are saying the growth of $f(x)$ is $O(g(x))$, then $f(x)$ has upper bound in terms of $g(x)$, for the size of $f(x)$ for large value of x. However, for large value of x big O notation does not provide lower bound for the size of $f(x)$. In that case, we use big Ω notation and when we want to give both an upper bound and lower bound on the size of function $f(x)$, relative to reference function $g(x)$ we use big Θ notation. Since we do not know how long a particular operation may take time on a particular computer, so constant factor in running time have little meaning. Hence big O notation $O(f)$ is convenient to use. Donald Knuth in 1970 observed that big O notation is often used by careless writers and speakers as if it had the same meaning as big Θ notation. For more details about this section reader is advised to review the book of H. Rosen [13].

Chapter 2

A Selective Review of Stable Matching Literature

In this chapter, we will review some well-known models of two-sided matching problem. These models involve two disjoint sets of participants, say M and W, each of whom ranks a subset of the other set of participants in order of preference. First of all we will review very famous model introduced by Gale and Shapely [7] in 1962. This model is known as *marriage model*. The second model we will review in this chapter is due to Shapley and Shubik [14] which was published in 1972 and is known as *assignment game*. The third model which we will completely review is the model of Ali and Farooq [1]. This model is also the motivation of our work.

2.1 The Classical Stable Marriage Problem

The concept of stable marriage problem was first introduced by Gale and Shapely [7] in the paper *"College Admissions and the Stability of Marriage"*. This paper is considered to be one of the most interesting abstractions in the matching market. In this paper, the stability of marriage problem is a game in which we have equal number of men and women as participants. The aim is to find a one-to-one matching between men and women such that there is no unmatched couple each of whom prefers the other to their partner in the matching. Marriage problem is the most common example of one-to-one two-sided matching market. The authors used their solution to this problem as a basis for solving the extended problem where one of the sets consists of college applicants, and the other consists of colleges, each of which has a quota of places to fill. We will first discuss the marriage model and then college admissions problem.

2.1.1 One-to-One Matchings: The Marriage Model

In the marriage model, there are two disjoint sets of participants, men M and women W of equal size n, each of whom has complete and transitive preferences over the participants of the opposite set. The task is to find a stable matching, that is, a set of pairs of acceptable participants such that no two participant of opposite set who would both rather be matched with each other than their current partners. If there are no such participants, all the marriages are said to be stable. In [7], Gale and Shapley proved that it is always possible to find a matching that makes all marriages stable, and provided a polynomial time algorithm widely known as *Deferred Acceptance Algorithm*. This algorithm can be used to find one of two extreme stable marriages, the so-called men-optimal or women-optimal marriages.
A matching in the marriage model is called *men-optimal* if every man likes his partner to any other partner that he could possibly have in the matching. Also, a matching is called *women-optimal* if every women likes her partner to any other partner that she could possibly have in matching.

2.1.2 Deferred Acceptance Algorithm: (with men proposing)

At the start of the algorithm, each person is free and becomes engaged during the execution of the algorithm.

Step 1: **(a)** Each man who is not engaged proposes to the most preferred woman in his preference list.

(b) Each woman engages with the most preferred proposal and rejects all other if she receives more than one proposal.

Step 2: If all the men are matched or rejected by all the women then stop. Else go to Step 3.

Step 3: Each rejected man deletes the woman who has rejected him. Go to Step 1.

This algorithm is said to be men-oriented algorithm or men-optimal because sequence of proposals is from men to women. If we reverse the sequence of proposals from women to men then the algorithm is said women-oriented algorithm that will produce the women optimal matching. Thus, Gale and Shapley [7] algorithm has only two possible versions, that is, men-oriented algorithm and women-oriented algorithm. The complexity of above algorithm is $O(n^2)$. Gale and Shapley [7] guarantee that two important results arise from this algorithm:

1. This algorithm always produces a stable set of marriages.

2. This algorithm produces an optimal set of marriages which implies that in each matching, every man is at least as well under the assignment given by *Deferred Acceptance Algorithm* as he would be under any other assignment.

In the men oriented algorithm, each man proposes to an acceptable woman in his preference list and no woman will ever be engaged with an unacceptable man. If there exists some man who is matched but prefers some other woman, but by the above algorithm, he must have already proposed to her and she has rejected him. This means, she has already a man to whom she strictly prefers than this man. Thus, they do not form a blocking pair.

Theorem 2.1.1 (Gale and Shapley [7])**.** *The men proposing Deferred Acceptance Algorithm always gives a stable marriage for each marriage problem.*

To prove this theorem Gale and Shapley [7] devised the above *deferred acceptance algorithm*. This theorem guarantees that if the number of men and women coincide, and all participants express a strict order over all the participants of the other set, everyone gets married, and the returned matching is stable.

Now, we will discuss the following examples to implement the above algorithm first taking men as proposers and then women as proposers.

Example 2.1.1. Consider two disjoint sets of participants, set of men M and set of women W. Let $M = \{A, B, C, D\}$ and $W = \{\alpha, \beta, \gamma, \delta\}$. The symbol $\longrightarrow$ is the sign of proposing. Preference list of each participant is showed in Table 2.1

Men's list	*Women's list*
$Pl(A) : \alpha, \beta, \gamma, \delta$	$Pl(\alpha) : C, D, A, B$
$Pl(B) : \alpha, \beta, \gamma, \delta$	$Pl(\beta) : D, A, B, C$
$Pl(C) : \beta, \gamma, \alpha, \delta$	$Pl(\gamma) : A, B, C, D$
$Pl(D) : \gamma, \alpha, \beta, \delta$	$Pl(\delta) : C, D, A, B$

Table 2.1: An instance of size $k = 4$

First, we find the matching X when men are proposers. Initially all men and women are unmatched and matching X is empty.
At Step **1(a)** of **First iteration**, $A \longrightarrow \alpha$, $B \longrightarrow \alpha$, $C \longrightarrow \beta$, $D \longrightarrow \gamma$. Now, α has two proposals A and B and $A >_\alpha B$ in the preference list of α therefore B is rejected at Step **1(b)** and $X = \{(A, \alpha), (C, \beta), (D, \gamma)\}$. Since B is rejected, the algorithm will not stop. We will move to Step **3**. At Step **3** B deletes α and goes to Step **1(a)**. At Step **1(a)** of **Second iteration**, $B \longrightarrow \beta$ and $B >_\beta C$ in the preference list of β so β switches to B and $X = \{(A, \alpha), (B, \beta), (D, \gamma)\}$. In **Third iteration**, $C \longrightarrow \gamma$ and $C >_\gamma D$, so γ switches to C and $X = \{(A, \alpha), (B, \beta), (C, \gamma)\}$. In **Fourth iteration**, $D \longrightarrow \alpha$ and $D >_\alpha A$, so α switches to D and $X = \{(D, \alpha), (B, \beta), (C, \gamma)\}$. In **Fifth iteration**, $A \longrightarrow \beta$ and $A >_\beta B$, so β switches to A and $X = \{(D, \alpha), (A, \beta), (C, \gamma)\}$. In **Sixth iteration**, $B \longrightarrow \gamma$ and $B >_\gamma C$, so γ switches to B and $X = \{(D, \alpha), (A, \beta), (B, \gamma)\}$. In **Seventh iteration**, $C \longrightarrow \alpha$ and $C >_\alpha D$, so α switches to C and $X = \{(C, \alpha), (A, \beta), (B, \gamma)\}$. In **Eighth iteration**, $D \longrightarrow \beta$ and $D >_\beta A$, so β switches to D and $X = \{(D, \beta), (B, \gamma), (C, \alpha)\}$. In **Ninth iteration**, $A \longrightarrow \gamma$ and $A >_\gamma B$, so γ switches to A and $X = \{(D, \beta), (A, \gamma), (C, \alpha)\}$. In **Tenth iteration**, $B \longrightarrow \delta$ and $\delta \longrightarrow B$ so $X = \{(A, \gamma), (B, \delta), (C, \alpha), (D, \beta)\}$. Thus, when the men do the proposing, the stable pairings is $X = \{(A, \gamma), (B, \delta), (C, \alpha), (D, \beta)\}$.
Now, if we apply the algorithm when the women do the proposing using the same

preference lists then we can find the stable pairing X^* obtained by following the same Steps as above.

At Step **1(a)** of **First iteration**, $\alpha \longrightarrow C$, $\beta \longrightarrow D$, $\gamma \longrightarrow A$, $\delta \longrightarrow C$. Now C has two proposals α and δ. Since $\alpha >_C \delta$ therefore δ is rejected at Step **1(b)** and $X^* = \{(\alpha, C), (\beta, D), (\gamma, A)\}$. Since δ is rejected the algorithm will not stop. We will move to Step **3**. At Step **3**, δ deletes C and goes to Step **1(a)**. At Step **1(a)** of **Second iteration**, $\delta \longrightarrow D$ and $\beta >_D \delta$ so β stays with D and δ is rejected again and $X^* = \{(\alpha, C), (\beta, D), (\gamma, A)\}$. In **Third iteration**, $\delta \longrightarrow A$ and $\gamma >_A \delta$ so γ stays with A and δ is rejected and $X^* = \{(\alpha, C), (\beta, D), (\gamma, A)\}$. In **Fourth iteration**, $\delta \longrightarrow B$ and $B \longrightarrow \delta$ so $X^* = \{(\alpha, C), (\beta, D), (\gamma, A), (\delta, B)\}$. Thus, when the women do the proposing, the stable pairings is $X^* = \{(\alpha, C), (\beta, D), (\gamma, A), (\delta, B)\}$.

As we have demonstrated with our above example, using the Gale-Shapely algorithm we sometimes find more than one stable solution: one when men are proposers or one when the women are proposers. However, this algorithm sometimes produces the same result. Thus, when we get the same result where the members of either set are proposing, this is called unique stable pairing. Now, we give an example in which men-optimal and women-optimal matching is not same as above just to show that the case that has come in above example will not be true always.

Example 2.1.2. Consider two sets, set of men and set of women denoted by M and W, respectively. Let $M = \{A, B, C, D\}$ and $W = \{\alpha, \beta, \gamma, \delta\}$. Preference list of each participant is showed in Table 2.2.

Men's list	*Women's list*
$Pl(A) : \gamma, \alpha, \beta, \delta$	$Pl(\alpha) : A, C, D, B$
$Pl(B) : \delta, \beta, \gamma, \alpha$	$Pl(\beta) : D, C, A, B$
$Pl(C) : \alpha, \beta, \gamma, \delta$	$Pl(\gamma) : D, A, B, C$
$Pl(D) : \delta, \alpha, \beta, \gamma$	$Pl(\delta) : C, A, B, D$

Table 2.2: An instance of size $k = 4$

If men do the proposing the stable matching is $X = \{(A, \gamma), (B, \delta), (C, \alpha), (D, \beta)\}$. If women do the proposing the stable matching is $X^* = \{(\alpha, C), (\beta, B), (\gamma, A), (\delta, D)\}$.

Clearly, we can see that men-optimal and women-optimal stable matchings X and X^* are not the same using the preference list defined in Table 2.2. The above discussion depends on two-sided nature of problem. Gale and Shapley [7] also showed in contrast a one-sided nature that is roommate problem with even cardinality. Roommate problem is also rephrased as same sex stable marriage problem.
In this problem, there is only one group of participants, each participant has strict preferences over all other participants, and each participant would make their preference list from 1 to $n-1$. Set of matchings in this problem is also unstable if there exists any blocking pair.
The following lemma of Guesfield and Iriving [8] shows that roommate problem is the generalization of marriage problem.

Lemma 2.1.2 (Guess and Iriving [8])**.** *Given an instance of stable marriage problem with same no of participants in two disjoint sets, there is an insistence of stable roommate problem with even cardinality of participants such that the stable roommate matchings are precisely the stable matchings for original stable marriage instance.*

Now, we present an example which shows that it is not possible that stable marriage always exists.

Example 2.1.3. Consider an even number of potential roommates $\{a, b, c, d\}$ such that a prefers b first in his or her preferences, b prefer c first and c prefer a first but all a, b, and c prefers d last, that is,

$$Pl(a) = b, c, d$$
$$Pl(b) = c, b, d$$
$$Pl(c) = a, b, d$$
$$Pl(d) = b, a, c.$$

In this case, with such a preference list, we cannot find a stable roommate pair, because a prefers b, b prefers c and c prefers a, and whoever is matched with d would prefer to be paired with one of the other two who would also prefer to be paired with that person than the person they are paired with.

The above examples of one-to-one matching are helpful to start many-to-one matching. Because many-to-one (college admission problem) matching is closely related to it. Although there is only a slight difference between college admission problem and marriage problem, in the next section we will categorize this difference.

2.1.3 Many to One Matching: The College Problem

The college problem is widely studied by economists and game theorists and it is the most well-known application of stable marriage problem. The mechanism that is used to solve a marriage problem and a college admission problem share some properties. This mechanism guarantees that resulting outcome is optimal stable for one of the party but not optimal stable for both parties. Here, this point is worth noting that in marriage problem men-optimal stable mechanism and women-optimal stable mechanism are symmetric of each other. While in the college admission problem, student optimal stable matching and college optimal stable matching, are not. Like in the marriage problem, the college admission problem also has two disjoint sets of colleges and students. Suppose there are n students, that is, $S = \{s_1, s_2, \ldots, s_n\}$ who are applying to get admission in k colleges, that is, $C = \{c_1, c_2, \ldots, c_k\}$. Each student has a strict order preferences over colleges. It is not possible for each college to give admission to all students who are applying because of limited resources which means that each $c_j \in C$ has a specific quota of enrollment in each academic year, that is, $q_{(c_j)} \geq 1$. Here $q_{(c_j)}$ represents the maximum number of students that a college c_j can enroll. This is known as quota of college c_j. Each college has some specific criteria of giving admission. The criteria is to enrol best q qualified students but this criteria is not generally satisfied because it is not possible that all of the q students who are accepted will accept the offer of college because they may also have applied to some other college. Since each college has strict preference list so if the college offers admission to only q students then it may be the case that some of these q students had applied to more than one college. Consequently, these other colleges might also offer them admission. So, in order to fulfill the admission quota colleges introduce the concept of waiting lists. The purpose of giving the concept of waiting list is that if more than q students applied then college will give the admission to the best q students and place rest of them in the waiting list such that they may not be admitted yet but may be admitted later if some vacancy occurs. This introduction of waiting list creates new problems for students as well as for colleges. Suppose student is getting admission offer from the college that he prefers least and the college that he prefers most placed his name in the waiting list. Now suppose if the student gets admission in the college that he prefers least but later the college that he prefers most, on not fulfilling its quota, offers him admission. If he accepts the offer and withdraws his admission from least preferred college to enroll in most preferred then this action of student will not please the least preferred college. On the other side if the student restricts himself to be enrolled in most preferred college then it will hurt other students because the least preferred college has already fulfilled its quota of best q students and has rejected the students who were in waiting list. To solve these kind of difficulties Gale and Shapley [7] introduced the mechanism known as Assignment Game. They observed that the same *deferred acceptance algorithm* (college c_i offered admission at each point to its q_i most preferred students who hadn't yet rejected it in the college-proposing version, or rejecting all but the q_i most preferred applications it had received at any point of the student-proposing version) would produce a stable matching defined in the above section. Basically the mechanism is to assign students to colleges which should satisfy both groups and remove all uncertainties and difficulties. The outcome produced by the algorithm would not admit any student-college blocking pairs defined precisely as for the marriage model.

2.1.4 Assignment Criteria

Consider a set of k colleges and a set of n students. Each student has strict order preferences (there are no ties) over the colleges, has eliminated all those colleges that he would never accept at any cost and finally each student can accept only one offer of admission at a time. Each college also has strict preferences over students who have applied to it and eliminated all those students that he would never have enrolled in any case even its admission quota remains unfilled. The goal is to find such an assignment between colleges and students that it satisfies criteria of fairness.

To find such an assignment is not simple in actual because complications may arise. For example: we have two colleges A and B and two students a and b, a prefers A to B and b prefers B to A. But A prefers b to a and B prefers a to b. In such a case there exists no assignment that satisfies all preferences. But, by philosophical principle, colleges exist to serve the students so a should get admission in A and b in B that is each student should get consideration over colleges. Whatever the assignment will be eventually, the following situation should not occur.

We have two students a, b and two colleges A, B. Student a is assigned to college A and b is assigned to college B but college A prefers student b to student a and college B prefers student a to student b. If this kind of situation occurs in the assignment then such an assignment is called *unstable*.

Hence, when we try to find any assignment between two disjoint sets our first criteria will be that this assignment should be stable, which means that there exists no blocking pair. So a question arises here will it always be possible to find such a stable assignment? And if such an assignment exist we still have to decide which stable solution is to be preferred among many stable solutions.

Suppose we have set of stable assignments. There is an assignment in which each applicant is at least as well off as compared to any other assignment. Then that assignment is called *optimal assignment*. But the existence of stable assignment doesn't guarantee that they are optimal stable assignment. However if there exists an optimal assignment then it must be unique.

2.2 Assignment Game by Shapely and Shubik

In the above section, we have discussed the marriage model of Gale and Shapely [7] that has at least three assumptions: (**1**) monetary transfer between participants is not allowed (**2**) no participant of same set can make a partnership (**3**) every participant can have at most one partner. Now, if we relax the condition **(1)** and allow monetary transfer between participants then we get a problem that is called Assignment Problem designed by Shapely and Shubik [14]. Special feature of Shapely and Shubik [14] model is that the participant set is separated into two different type of sets, the set of sellers and the set of buyers, monetary transfer is allowed between participants of both sets. Involvement of money in the assignment game of Shapely and Shubik [14] makes it different from the marriage model of Gale and Shapely [7]. Each seller in assignment game of Shapely and Shubik [14] has a unit of indivisible goods and each buyer needs exactly one unit of indivisible goods. Differentiations on the units of indivisible goods are allowed and therefore buyer might place different valuation on the unit of different sellers. The main economic assumption of Shapely and Shubik [14] model are:

- identification of utility with money,
- side payments are permitted between participants,
- the objects of trade between participants are indivisible,
- supply and demand functions are inflexible.

2.2.1 Model Description

Let we have two disjoint sets, N_1 and N_2, where N_1 denotes the set of homeowners and N_2 denotes the set of prospective purchasers. Here homeowners are sellers and purchasers are buyers. The i^{th} seller values his house at c_i dollars and the j^{th} buyer value the same house at h_{ij} dollars. If $h_{ij} > c_i$ then there exists a favorable price for both of the parties buyer and seller but it cannot be assumed that this inequality always holds in all cases. The possible move in the game includes that a house is transferred from its owner to any buyer and the transfer of money from buyer to seller. Let i^{th} seller sells his house to j^{th} buyer for price p_i, and if no one is ready to deal with third party, then i^{th} seller final profit is

$$p_i - c_i$$

and the j^{th} buyer final profit is

$$h_{ij} - p_i.$$

2.2.2 Coalition Form

Let $N = \{1, 2, \ldots, n\}$ denote the set of players in a game and $n \geq 2$ denote the number of players in a game. Any subset of set N is called a *coalition* or *alliance* and the set of all coalitions is denoted by 2^N. The empty set ϕ is also a coalition called *Empty Coalition* and the set N is called *Grand Coalition*. A pareto optimal matching X has the property that there exists no coalition of participants who want to improve their allocations(say by exchanging items with one another), without requiring some other participant to be worse off.

Example 2.2.1. In the set N we have three participants $N = \{a, b, c\}$ that is $n = 3$ then the total number of coalitions are eight

- three one participant coalition $\{a, b, c\}$,
- three two participant coalition $\{(a, b), (a, c), (b, c)\}$,
- grand coalition N,
- empty coalition ϕ.

In two-sided matching market the set of players is partitioned into two disjoint sets say N_1 and N_2. A *coalition* S, in two-sided matching market, is defined as the subset of $N_1 \times N_2$ where $N_1 \times N_2$ denotes the set of all possible players.

Definition 2.2.1. The *core* of the game is the set of efficient and individual rational payoff vectors under which no coalition can receive a greater value then the sum of its members payoffs.

2.2.3 The Characteristic Function

The maximum value for the coalition S is called the characteristic function of S and is denoted as $\nu(S)$. If the coalition S consists of less then two players then $\nu(S) = 0$. In the coalition it is not possible to have a profitable deal between the players that are from the same set.
The *characteristic function* $\nu : 2^N \to \mathbb{R}$ maps every subset S of N to a non-negative real number. $\nu(S)$ is called the worth of coalition S, denoted this worth by α_{ij} and

$a_{ij} = \nu(\{i,j\})$ for $i, j \in S$, is defined by

$$\nu(\{i,j\}) = \begin{cases} \max\{0, h_{ij} - c_i\} & \text{if} \quad i \in N_1 \quad \textit{and} \quad j \in N_2 \\ 0 & \text{if} \quad \{i,j\} \in N_1 \quad \textit{or} \quad \{i,j\} \in N_2\ . \end{cases}$$

If the coalition S consists of more than two players then the value of such larger mixed coalition is obtained by maximizing the coalition total gain. Mathematically we write it as

$$\nu(S) = \max[a_{i_1 j_1} + a_{i_2 j_2} + \ldots + a_{i_k j_k}]. \tag{2.2.1}$$

where $k > 2$ and $\{i_1, i_2, \ldots, i_k\} \in S \cap N_1$ and $j_1, j_2, \ldots, j_k \in S \cap N_2$ with $k = \min\{|S \cap N_1|, |S \cap N_2|\}$.

The expression given by (2.2.1) is called *optimal assignment problem* for the assignment game. Now, we give few examples for an optimal assignment game.

Example 2.2.2. Consider a matrix with entries a_{ij} where ($i \in S \cap N_1$ and $j \in S \cap N_2$) for an assignment game:

$$\begin{pmatrix} 0 & 6 & 0 \\ 6 & 0 & 6 \\ 0 & 6 & 0 \end{pmatrix}.$$

The bold face entries in the below matrix represent the four optimal assignments with a total profit of 12 units, that is, $\nu(S) = 12$:

$$\begin{pmatrix} 0 & \mathbf{6} & 0 \\ 6 & 0 & \mathbf{6} \\ \mathbf{0} & 6 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & \mathbf{6} & 0 \\ \mathbf{6} & 0 & 6 \\ 0 & 6 & \mathbf{0} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{0} & 6 & 0 \\ 6 & 0 & \mathbf{6} \\ 0 & \mathbf{6} & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 6 & \mathbf{0} \\ \mathbf{6} & 0 & 6 \\ 0 & \mathbf{6} & 0 \end{pmatrix}.$$

Example 2.2.3. Consider the matrix with entries (a_{ij}), where $i \in S \cap N_1$ and $j \in S \cap N_2$) for an assignment game:

$$\begin{pmatrix} 5 & 8 & 2 \\ 7 & 9 & 6 \\ 2 & 3 & 2 \end{pmatrix}.$$

In this example we have only one optimal assignment, that is

$$\begin{pmatrix} 5 & \mathbf{8} & 2 \\ \mathbf{7} & 9 & 6 \\ 2 & 3 & \mathbf{2} \end{pmatrix}.$$

The bold face entries represent the optimal assignment with a total profit of 17 units, that is, $\nu(S) = 17$.

2.2.4 Standard Linear Programming Terminology

Shapely and Shubik [14] used the linear programming theory developed by Dantzig [4] to solve the assignment problem given in (2.2.1). Consider the coalition of all players of assignment game. Our aim is to determine the worth of $N_1 \cup N_2$. For optimal assignment game, we introduce mn non-negative real variables x_{ij} and impose $m+n$ constraints on them. These non-negative real variables represent the quantity of good i allocated to buyer j where $i \in N_1$ and $j \in N_2$. Then the linear programming is used to find an assignment of indivisible goods to buyers to maximize the total value achieved such that no buyers get more than one unit of indivisible good and no seller sell more than one unit of indivisible good. So the linear programming is:

$$\begin{aligned} \text{maximize} \quad & p = \sum_{i\in N_1}\sum_{j\in N_2} a_{ij}x_{ij} \\ \text{subject to} \quad & \sum_{i\in N_1} x_{ij} \leq 1 \qquad \forall\, j \in N_2, \\ & \sum_{j\in N_2} x_{ij} \leq 1 \qquad \forall\, i \in N_1, \\ & x_{ij} \geq 0, \quad \forall\, i \in N_1,\, j \in N_2. \end{aligned} \tag{2.2.2}$$

The first constraint of above linear programming ensures that no seller supplies more than they have. Second constraint shows that no buyer is interested in acquiring more than one unit in total. Maximum value of $\sum_{i\in N_1}\sum_{j\in N_2} a_{ij}x_{ij}$ is attained when all the variables x_{ij} are either 0 or 1. So that the continuous linear programming problem effectively coincides with discrete assignment game. Now we have

$$p_{max} = \nu(N_1 \cup N_2).$$

Each linear programming problem is associated with another unique linear programming problem called dual linear programming having the form

$$\begin{aligned} &\text{minimize} \quad d = \sum_{i \in N_1} u_i + \sum_{j \in N_2} v_j \\ &\text{subject to} \quad u_i + v_j \geq a_{ij} \quad (\forall\, i \in N_1,\, j \in N_2). \end{aligned} \tag{2.2.3}$$

Here u_i be the dual variable associated with each constraint $\sum_{i \in N_1} u_i$ and v_j be the dual variable associated with each constraint $\sum_{j \in N_2} v_j$. The above linear program in (2.2.2) is sometimes called the primal problem and the program in (2.2.3) is called the dual problem of (2.2.2). Primal problem and its dual form a fundamental relationship between their solutions. This relationship states that '*If the primal or dual problem contain a feasible solution then the other problem also acquires a feasible solution and the optimum solution of the objective function of both the problems coincides*', that is,

$$p_{max} = d_{min}.$$

Let the vector $(u, v) = (u_1, \ldots, u_m, v_1, \ldots, v_n)$ minimizes the dual problem defined in (2.2.3). Then we have

$$\sum_{i \in N_1} u_i + \sum_{j \in N_2} v_j = d_{min} = p_{max} = \nu(N_1 \cup N_2). \tag{2.2.4}$$

The constraints of the dual problem defined in (2.2.3) show that for each pair (i, j) with $i \in N_1$ and $j \in N_2$ we have

$$u_i + v_j \geq a_{ij} = \nu(i, j).$$

Thus for coalition S we have

$$\sum_{i \in S \cap N_1} u_i + \sum_{j \in S \cap N_2} v_j \geq \nu(S). \tag{2.2.5}$$

Equation (2.2.4) guarantees the feasibility of payoff vector (u, v) and equation (2.2.5) guarantees its optimality by an alliance S. Thus, the feasibility of payoff vector (u, v) in (2.2.4) and optimality of the solution in (2.2.5) ensures that it is the *core* of the game .

In other sense, we can say that any payoff vector in the core that satisfies equation (2.2.4) and equation(2.2.5) clearly fulfills the conditions for a solution to the dual LP problem. The following theorem of Shapely and Shubik [14] summarizes the above whole discussion.

Theorem 2.2.1 (Shapely and Shubik [14]). *The set of solutions of the dual LP assignment problem of the associated assignment problem and core of the assignment game coincide with each other.*

2.3 A Review of Two-Sided Matching Models with Linear Valuations and Discrete Side Payments

In 2008, Farooq [6] presented a one-to-one matching model that includes the marriage model of Gale and Shapely [7], assignment game of Shapely and Shubik [14] and Erikson and Karlander [5] hybrid model as special cases. Important features of Farooq model [6] are:

- set of participants is partitioned into two disjoint sets,
- each participant has at most one partner of opposite set,
- monetary transfers are allowed and are bounded by lower and upper bounds,
- the preferences of participants are identified by strict increasing linear function of money,
- money is considered as a continuous variable.

Farooq [6] designed an algorithm to show that there always exists a pairwise stable outcome for his matching model with linear valuation and bounded side payments.

A careful analysis of his algorithm reveals that a proper implementation solves the problem in $O(n^7)$.

Theorem 2.3.1 (Farooq [6])**.** *The complexity of* STABLE-OUTCOME *is* $O(n^7)$ *where* n *denotes the number of participants.*

Motivated by his work, Ali and Farooq [1] presented a one-to-one matching model with linear valuations and bounded side payments. The work done in this thesis is motivated by the work of Ali and Farooq [1], so in the remaining part of this chapter, we will review the model of Ali and Farooq [1]. Following are the main features of Ali and Farooq model [1]:

- set of participants is partitioned in to two disjoint sets,
- each participant has at most one partner of opposite set,
- monetary transfer are allowed and are bounded by lower and upper bounds,
- the preferences of participants are identified by strict increasing linear function of money,
- money is considered as a discrete variable.

Consider a finite set of buyers and sellers. Set of all possible buyer-seller pairs is given by $E = P \times Q$, where P and Q denotes the respective set of buyers and sellers. For each buyer-seller pairs upper and lower bound of prices is defined by two vectors $\underline{\pi}, \overline{\pi} \in \mathbb{Z}^E$ where lower bound is always less than or equal to upper bound. Initially price vector $p = (p_{ij} \in \mathbb{Z} \mid (i,j) \in E)$ for each pair $(i,j) \in E$[1], must be feasible and is defined by[2]:

$$p_{ij} = \begin{cases} \overline{\pi}_{ij} & \text{if } \nu_{ji}(-\overline{\pi}_{ij}) \geq 0 \\ \max\left\{\underline{\pi}_{ij}, \left\lfloor \frac{\beta_{ji}}{\alpha_{ji}} \right\rfloor\right\} & \text{otherwise.} \end{cases} \tag{2.3.1}$$

The preferences of participants are given by the linear increasing function of money, that is,

$$\nu_{ij}(x) = \alpha_{ij}x + \beta_{ij}, \quad \nu_{ji}(-x) = -\alpha_{ji}x + \beta_{ji}, \tag{2.3.2}$$

[1]The notation $\mathbb{Z}$ stand for set of integers and notation $\mathbb{R}$ stand for set of real numbers. The notation $\mathbb{Z}^E$ stands for integer lattice whose points are indexed by E.

[2]$\lfloor x \rfloor = \sup\{n \in \mathbb{Z} \mid x \geq n\}$.

where $\beta_{ij}, \beta_{ji} \in \mathbb{R}$ and $\alpha_{ij}, \alpha_{ji} \in \mathbb{R}^+$ and $x \in \mathbb{Z}$.

Ali and Farooq [1] specified an outcome by a 4 tuple (X, p, q, r) and described that an outcome is pairwise stable if it satisfies the following two conditions $(ps1)$ and $(ps2)$ given below:

(ps1) $q \geq \mathbf{0}$ and $r \geq \mathbf{0}$,

(ps2) $\nu_{ij}(c) \leq q_i$ or $\nu_{ji}(-c) \leq r_j$ for all $c \in [\underline{\pi}_{ij}, \overline{\pi}_{ij}]_{\mathbb{Z}}$ and for all $(i, j) \in E$.[3]

Here $(q, r) \in \mathbb{R}^P \times \mathbb{R}^Q$ is defined by

$$q_i = \begin{cases} \nu_{ij}(p_{ij}) & \text{if } (i,j) \in X \text{ for some } j \in Q \\ 0 & \text{otherwise} \end{cases} \qquad (i \in P), \tag{2.3.3}$$

$$r_j = \begin{cases} \nu_{ji}(-p_{ij}) & \text{if } (i,j) \in X \text{ for some } i \in P \\ 0 & \text{otherwise} \end{cases} \qquad (j \in Q). \tag{2.3.4}$$

Before describing the algorithm mathematically, exclude all those buyer-seller pairs that are not mutually acceptable from set E and such excluded pairs are defined by:

$$L_0 = \{(i, j) \in E \mid \nu_{ji}(-p_{ij}) < 0\}, \tag{2.3.5}$$

$$E_0 = \{(i, j) \in E \mid \nu_{ij}(p_{ij}) < 0\}. \tag{2.3.6}$$

The set of mutually acceptable buyer-seller pairs is given by

$$\widetilde{E} = E \setminus \{E_0 \cup L_0\}. \tag{2.3.7}$$

Define $\tilde{q}_i$ for each $i \in P$, and $\widetilde{E}_P$ by (2.3.8) and (2.3.9)

$$\tilde{q}_i = \max\{\nu_{ij}(p_{ij}) \mid (i, j) \in \widetilde{E}\}, \tag{2.3.8}$$

and

$$\widetilde{E}_P = \{(i, j) \in \widetilde{E} \mid \nu_{ij}(p_{ij}) = \tilde{q}_i\}. \tag{2.3.9}$$

The maximum over an empty set is taken to be zero by definition. Here the set $\widetilde{E}_P$ contains those buyer-seller pairs which are mutually acceptable and the buyer is

[3]For any $x, y \in \mathbb{Z}$, we define $[x, y]_{\mathbb{Z}} = \{a \in \mathbb{Z} \mid \ x \leq a \leq y\}$.

most preferred for seller out of all acceptable buyers. Initially, consider vector $r = \mathbf{0}$ and the subset $\widehat{E}_P$ of $\widetilde{E}_P$ is defined by

$$\widehat{E}_P = \{(i,j) \in \widetilde{E}_P \mid \nu_{ji}(-p_{ij}) \geq r_j\}. \tag{2.3.10}$$

Initially, $\widehat{E}_P$ will coincide with $\widetilde{E}_P$. However, in the further iterations of the algorithm $\widehat{E}_P$ may be a proper subset of $\widetilde{E}_P$. Since we have no matching X at the start of the algorithm, so consider $\widetilde{Q} = \emptyset$, where $\widetilde{Q}$ denotes the set of matched buyers in X, that is,

$$\widetilde{Q} = \{j \in Q \mid j \text{ is matched in } X\}. \tag{2.3.11}$$

Ali and Farooq [1] defined that at each step in the algorithm the matching X in the bipartite graph $(P, Q; \widehat{E}_P)$ must satisfy the following two conditions:

(a1) X matches all members of $\widetilde{Q}$,

(a2) X maximizes $\sum_{(i,j)\in X} \nu_{ji}(-p_{ij})$ among the matchings having (a1).

Since initially $\widetilde{Q} = \emptyset$, so any matching satisfies $(a1)$. Upto these steps an outcome (X, p, q, r) obviously satisfies the condition (ps1). To satisfy the condition (ps2), define the set U by

$$U = \{(i,j) \in \widetilde{E}_P \mid i \text{ is unmatched in } X\}. \tag{2.3.12}$$

Here U is the set of all those seller-buyer pairs which are mutually acceptable and the buyer is most preferred for the seller but the seller is unmatched in X. If $U = \emptyset$ then there is no need to modify price vector p. But if $U \neq \emptyset$ then find an integer n_{ij} by[4] (2.3.13), to modify price vector p.

$$n_{ij} = \max\left\{1, \left\lceil \frac{r_j - \nu_{ji}(-p_{ij})}{\alpha_{ji}} \right\rceil \right\}. \tag{2.3.13}$$

The modified price vector is denoted by $\tilde{p}$ and is defined by

$$\tilde{p}_{ij} := \begin{cases} \max\{\underline{\pi}_{ij}, p_{ij} - n_{ij}\} & \text{if } (i,j) \in U \\ p_{ij} & \text{otherwise} \end{cases} \qquad (i,j) \in E. \tag{2.3.14}$$

[4] $\lceil y \rceil = \inf\{n \in \mathbb{Z} | y \leq n\}$.

Modified price vector must also be feasible and the condition $(ps1)$ remain preserved. Since, initially the price vector p is as large as possible so modified price vector $\tilde{p}$ defined in (2.3.14) monotonically decreases p, and hence preserves $\nu_{ji}(-p_{ij}) \geq 0$ and $r \geq \mathbf{0}$ for all $(i,j) \in U$. Also define the subset L and $\widetilde{E}_0$ of U by

$$L = \{(i,j) \in U \mid p_{ij} - n_{ij} < \underline{\pi}_{ij}\}, \tag{2.3.15}$$

and

$$\widetilde{E}_0 := \{(i,j) \in U \mid \nu_{ij}(\tilde{p}_{ij}) \leq 0\}. \tag{2.3.16}$$

Ali and Farooq [1] considered money as a discrete variable in their model and developed an algorithm to show the existence of pairwise stable outcome. The reason for taking money as a discrete variable in their model is useful in auction market because each bid should increase the price of the item by, say, 1 USD. Also, in Ali and Farooq [1] model when upper and lower bounds are set to zero, it coincides with Gale and Shapley [7] model.

Algorithm for finding a pairwise stable outcome:

Step 0: Set $r = \mathbf{0}$ and $\widetilde{Q} = \emptyset$. Initially define (X, p, q, r), L_0, E_0, $\widetilde{E}$, $\tilde{q}$, $\widetilde{E}_P$ and $\widehat{E}_P$ by using (2.3.5)−(2.3.10), respectively. Find a matching X in the bipartite graph $(P, Q; \widehat{E}_P)$ that satisfies the condition (a1) and (a2). Again define r, $\widetilde{Q}$ and U by (2.3.4), (2.3.11) and (2.3.12), respectively

Step 1: If $U = \emptyset$ then, all sellers are matched in matching X, define q by (2.3.3) and stop. Otherwise go to Step 2.

Step 2: For each $(i,j) \in U$ calculate n_{ij} by (2.3.13) and new price vector $\tilde{p}$ by (2.3.14). Define L and $\widetilde{E}_0$ by (2.3.15) and (2.3.16), respectively and update E_0 by $E_0 := E_0 \cup \widetilde{E}_0$ and L_0 by $L_0 := L_0 \cup L$.

Step 3: Replace price vector p by $\tilde{p}$ and modify $\widetilde{E}$ by

$$\widetilde{E} := \widetilde{E} \setminus \{L_0 \cup E_0\}. \tag{2.3.17}$$

Again define $\tilde{q}$ by (2.3.8) and modify $\widetilde{E}_P$, $\widehat{E}_P$ by (2.3.9) and (2.3.10) respectively, for the updated p and $\widetilde{E}$. Find a matching X in the bipartite graph $(P, Q; \widehat{E}_P)$ that satisfies the conditions (a1) and (a2). Again define r, $\widetilde{Q}$ and U by (2.3.4), (2.3.11) and (2.3.12), respectively. Go to Step 1.

The complexity of this algorithm depends on the size of those intervals where price falls.

Theorem 2.3.2 (Ali and Farooq [1])**.** *After a finite number of iterations the algorithm always terminates.*

In the given example we followed the algorithmic approach of Ali and Farooq [1] model and find a stable matching X that satisfies the conditions (ps1) and (ps2).

Example 2.3.1. We have two sets of participants, that is, set of buyers P and set of sellers Q where $P = \{i_o, i_1, i_2, i_3\}$ and $Q = \{j_o, j_1, j_2, j_3\}$. Let $E = P \times Q$ denotes the set of all buyers and sellers and lower and upper bounds for each $(i, j) \in E$ is defined as

$$\begin{aligned} \underline{\pi}_{ij} &= -2 \qquad \forall (i,j) \in E, \\ \overline{\pi}_{ij_0} &= 3 = \overline{\pi}_{ij_1} \quad \forall i \in P, \\ \overline{\pi}_{ij_2} &= 2 = \overline{\pi}_{ij_3} \quad \forall i \in P. \end{aligned}$$

To find the values of valuation function $\nu_{ij}(x)$ and $\nu_{ji}(-x)$ define in (2.3.2), we fix the values of α_{ij} and α_{ji} as $\alpha_{ij} = 3$, $\alpha_{ji} = 4$, and the values of β_{ij} and β_{ji} for each $(i, j) \in E$ is shown in Table (2.3), (2.4). Now we apply the algorithm starting with

β_{ij}	j_0	j_1	j_2	j_3
i_0	2	-2.5	5	6
i_1	7	1	3	9
i_2	3	8	10	6
i_3	6	4	2	-4

Table 2.3: β_{ij} for $(i, j) \in E$

β_{ji}	i_0	i_1	i_2	i_3
j_0	6	-15	4	9
j_1	1	10	11	12
j_2	7	6	12	-3
j_3	11	0	-1	14

Table 2.4: β_{ji} for $(i, j) \in E$

Step 0. At Step 0, we have $r = \mathbf{0}$ and $\widetilde{Q} = \emptyset$. By (2.3.1), corresponding components of price vector is

$$p = (1, 0, 1, 2, -2, 2, 1, 0, 1, 2, 2, -1, 2, 3, -1, 2).$$

Now by using the values of α_{ij}, α_{ji}, β_{ij}, β_{ji} and the price vectors obtained at Step 0 we find the values of $\nu_{ij}(p_{ij})$ and $\nu_{ji}(-p_{ij})$ for each $(i,j) \in E$ in the Table (2.5), (2.6).

$\nu_{ij}(p_{ij})$	j_0	j_1	j_2	j_3
i_0	5	-2.5	8	12
i_1	1	7	6	9
i_2	6	14	16	3
i_3	12	13	-1	2

Table 2.5: $\nu_{ij}(p_{ij})$ for $(i,j) \in E$

$\nu_{ji}(-p_{ij})$	i_0	i_1	i_2	i_3
j_0	2	-7	0	1
j_1	1	2	3	0
j_2	3	2	4	1
j_3	3	0	3	6

Table 2.6: $\nu_{ji}(-p_{ij})$ for $(i,j) \in E$

From the values given in Table (2.5), (2.6) we obtain $\nu_{i_0j_1}(-p_{i_0j_1})$, $\nu_{i_3j_2}(-p_{i_3j_2})$ and $\nu_{j_0i_1}(-p_{i_1j_0})$ are negative, therefore we get $L_0 = \{i_1, j_0\}$ and $E_0 = \{(i_0, j_1), (i_3, j_2)\}$. Exclude these pairs from set E and set of mutually acceptable buyer-seller pairs is:

$$\widetilde{E} = E \setminus \{(i_1, j_0), (i_0, j_1), (i_3, j_2)\}.$$

Using (2.3.8), we obtain

$$\tilde{q}_{i_0} = 12, \quad \tilde{q}_{i_1} = 9, \quad \tilde{q}_{i_2} = 16, \quad \tilde{q}_{i_3} = 13.$$

Also by (2.3.9), we find

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_3), (i_2, j_2), (i_3, j_1)\}.$$

As the subset $\widehat{E}_P$ of $\widetilde{E}_P$ is defined in (2.3.10) and initially $r = \mathbf{0}$, so we have $\widehat{E}_P = \widetilde{E}_P$. From the bipartite graph $(P, Q, \widehat{E}_P)$, find a matching X that satisfies the condition (a1) and (a2). Thus, the required matching that satisfies both conditions is $X = \{(i_0, j_3), (i_2, j_2), (i_3, j_1)\}$. From the matching X we find r by (2.3.4) and get

$$r_{j_0} = 0, \quad r_{j_1} = 0, \quad r_{j_2} = 4, \quad r_{j_3} = 3.$$

Also from the matching X updating $\widetilde{Q}$, that contains matched buyers in X, thus

$$\widetilde{Q} = \{j_1, j_2, j_3\}.$$

Since set U, defined in (2.3.12), contain such pairs in which we have $\nu_{ij}(-p_{ij}) \leq r_j$. Thus $U = \{(i_1, j_3)\}$. As $U \neq \emptyset$, so we move to Step 2 of algorithm and find $n_{i_1 j_3}$ by (2.3.13) we get $n_{i_1 j_3} = 1$.

Modifying the corresponding components of price vector $\tilde{p}$ for the pairs that are in U by (2.3.14). All the components of price vector $\tilde{p}$ remain unchanged except $\tilde{p}_{i_1 j_3}$ so:

$$\tilde{p} = (1, 0, 1, 2, -2, 2, 1, -1, 1, 2, 2, -1, 2, 3, -1, 2).$$

Also all the valuations in Table (2.5), (2.6) remain unchanged except $\nu_{i_1 j_3}(\tilde{p}_{i_1 j_3})$ and $\nu_{j_3 i_1}(\tilde{p}_{i_1 j_3})$. Thus modified value of these valuations are $\nu_{i_1 j_3}(-1) = 6$ and $\nu_{j_3 i_1}(1) = 4$. Both sets L_0 and $\widetilde{E}_0$ are empty. So L_0 and E_0 remain unchanged for the modified price vector $\tilde{p}$, therefore modified $\widetilde{E}$ remain unchanged at Step 3. Again by (2.3.8) we have

$$\tilde{q}_{i_0} = 12, \quad \tilde{q}_{i_1} = 7, \quad \tilde{q}_{i_2} = 16, \quad \tilde{q}_{i_3} = 13.$$

Updating $\widetilde{E}_P$ and $\widehat{E}_P$ we get

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_1)\}.$$

and $\widetilde{E}_P = \widehat{E}_P$. The matching X in the bipartite graph $(P, Q, \widehat{E}_P)$ that satisfies (a1) and (a2) condition is

$$X = \{(i_0, j_3), (i_1, j_1), (i_2, j_2)\}.$$

Again update r by (2.3.4), we have

$$r_{j_0} = 0, \quad r_{j_1} = 2, \quad r_{j_2} = 4, \quad r_{j_3} = 3.$$

Updated $\widetilde{Q}$ is

$$\widetilde{Q} = \{j_0, j_1, j_3\}.$$

And updated U is

$$U = \{(i_3, j_1)\}.$$

Since $U \neq \emptyset$, we move to Step 2 and calculate the value of $n_{i_3 j_1}$ by (2.3.13) which is $n_{i_3 j_1} = 1$. All the components of modified price vector $\tilde{p}$ remain unchanged except $\tilde{p}_{i_3 j_1}$ so:

$$\tilde{p} = (1, 0, 1, 2, -2, 2, 1, -1, 1, 2, 2, -1, 2, 2, -1, 2).$$

Again all the valuations in Table (2.5), (2.6) remain unchanged except $\nu_{i_3 j_1}(\tilde{p}_{i_3 j_1})$ and $\nu_{j_1 i_3}(\tilde{p}_{i_3 j_1})$. Thus modified value of these valuations are $\nu_{i_3 j_1}(2) = 10$ and $\nu_{j_1 i_3}(-2) = 4$.

Again both sets L_0 and $\widetilde{E}_0$ are empty. So L_0 and E_0 remain unchanged for the modified price vector $\tilde{p}$. Thus $\widetilde{E}$ remains unchanged. Again by (2.3.8) we have

$$\tilde{q}_{i_0} = 12, \quad \tilde{q}_{i_1} = 7, \quad \tilde{q}_{i_2} = 16, \quad \tilde{q}_{i_3} = 12.$$

Updated $\widetilde{E}_P$ and $\widehat{E}_P$ is

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_0)\},$$

$$\widehat{E}_P = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_0)\}.$$

The matching X in the above bipartite graph is

$$X = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_0)\}.$$

Here $U = \emptyset$ that is all the buyers and sellers are matched so algorithm terminates. Matching X obviously satisfies (ps1) and (ps2) conditions. So X is pairwise stable outcome.

In the next section we will discuss some important results of Ali and Farooq [1] model. The proofs of these results can be seen in the original paper of Ali and Farooq [1]. So we will not prove them here. In section (2.3.1), we will just give the statements of lemma's and theorem's of Ali and Farooq model [1].

2.3.1 Main Results

Lemma 2.3.3 (Ali and Farooq [1]). *There exists a matching X in the bipartite graph $(P, Q, \widehat{E}_P)$ that satisfies condition (a1) and (a2) in each iteration of the algorithm at Step 3.*

The next lemma of Ali and Farooq [1] helps in proving the subsequent lemmas because it describes the important features of algorithm.

Lemma 2.3.4 (Ali and Farooq [1])**.** *In each iteration of the algorithm, following holds:*

(i) *The price vector p decreases or remains the same. In particular, if $U \setminus \{L \cup \widetilde{E}_0\} \neq \emptyset$ at Step 2 then p_{ij} decreases at Step 3 for all $(i,j) \in U \setminus \{L \cup \widetilde{E}_0\}$.*

(ii) *$\widetilde{E}$ reduces or remains the same. In particular, if $L \neq \emptyset$ or $\widetilde{E}_0 \neq \emptyset$ at Step 2 then $\widetilde{E}$ reduces at Step 3.*

(iii) *The vector r increases or remains same.*

The following lemma of Ali and Farooq [1] illustrates that in each iteration of algorithm at *Step 3* for each $(i,j) \in L$ we have $(new)p_{ij} = \underline{\pi}_{ij}$ and $\nu_{ji}(-(new)p_{ij}) \leq (old)r_j$.

Lemma 2.3.5 (Ali and Farooq [1])**.** *In each iteration of the algorithm at Step 3, we have $(new)p_{ij} = \underline{\pi}_{ij}$ and $\nu_{ji}(-(new)p_{ij}) \leq (old)r_j$ for each $(i,j) \in L$, where L is defined at Step 2.*

The next two lemmas of Ali and Farooq [1] must hold in each iteration of algorithm at *Step 3*.

Lemma 2.3.6 (Ali and Farooq [1])**.** *In each iteration of the algorithm at Step 3, we have $\nu_{ji}(-((old)p_{ij} - n_{ij})) \geq (old)r_j$ for each $(i,j) \in (old)U$, where n_{ij} is calculated at Step 2. Furthermore, if $\nu_{ji}\ (-((old)p_{ij} - n_{ij})) > (old)r_j$ for some $(i,j) \in (old)U$ then $(old)p_{ij} - n_{ij}$ is the maximum integer for which this inequality holds.*

Lemma 2.3.7 (Ali and Farooq [1])**.** *In each iteration of the algorithm at Step 3, we have $\nu_{ji}(-(new)p_{ij}) \geq (old)r_j$ for each $(i,j) \in (old)U \setminus L$, where L is defined at Step 2. Furthermore, if $\nu_{ji}(-(new)p_{ij}) > (old)r_j$ for some $(i,j) \in (old)U \setminus L$ then $(new)p_{ij}$ is the maximum integer in $[\underline{\pi}_{ij}, \overline{\pi}_{ij}]_{\mathbb{Z}}$ for which this inequality holds.*

The given theorem of Ali and Farooq [1] states that the four tuples (X, p, q, r) must satisfy conditions $(ps1)$ and $(ps2)$ when algorithm terminates.

Theorem 2.3.8 (Ali and Farooq [1]). *If the algorithm terminates then* $(X; p, q, r)$ *must satisfy* $(ps1)$ *and* $(ps2)$.

Chapter 3

Pairwise Stability in Two-sided Discrete Matching Market with General Increasing Valuation Function

Since the matching problems are associated with outcomes that need to be evaluated in terms of utility and the utility of a matching is the function of productivity of a pair of participants trading together. The primary objective of two-sided matching problem is the formulation of stable partnership between participants. In two-sided matching market, the matching X is said to be stable if there does not exist any alternative pair in X that is individually better off or like some other participant then the one to whom he or she is currently matched.
The motivation of our work from the stable matching literature is the matching model of Ali and Farooq [1]. Their model was arguably the simplest exchange of money one could think of. Preferences of participants in their model are given by strictly increasing linear function of prices. The objective of their model was to show that there indeed exists a pairwise stable outcome with linear valuations and bounded side payments, and to achieve this objective they proposed an algorithm. Money is considered as a discrete variable in their model. Motivated by their work,

we also consider a one-to-one matching model. In our model, preferences of participants are given by general increasing functions that are nonlinear in the whole real line. Because, during the recent years, the inflation rate has been very fast that is not increasing linearly. This hyper-inflation has endangered the stability of national and individual economy. So, in the current scenario linear valuations with discrete prices are not so much effective. Thus, it is natural to generalized a strictly increasing linear function of price from $\nu_{ij}(x)$ and $\nu_{ji}(-x)$ to strictly increasing nonlinear function of price $f_{ij}(x)$ and $f_{ji}(-x)$ for each $(i,j) \in E$. Thus, in this chapter we extend the idea of Ali and Farooq [1] paper by generalizing the utility functions.
We organize this chapter as fellows. We first give the description of our model briefly, in section 3.1. In section 3.2 and 3.3, we describe the buyer-seller sequential mechanism and supply and demand characterization of stable matchings. An algorithm for finding a pairwise stability is discussed in section 3.4. In section 3.5, we will discuss the main results of our model.

3.1 The Model Description

Throughout in this chapter, we model matching markets as trading platforms where buyers and sellers interact. Moreover, each buyer as well as seller can trade with at most one participant on the other side of the market at a particular time. This matching market consists two types of participants one type of participants is sellers U and second type of participants is buyers V. The negotiation and side payments between participants of both sides are allowed. Naturally, each participant wants to gain as much profit as possible from his/her partner. Let $E = U \times V$ denotes the set of all possible pairs of seller-buyer. Also when buyer and seller interact with each other in auction market they have some upper and lower bounds of prices. We express these bounds by vector $\underline{\pi}, \overline{\pi} \in \mathbb{Z}^E$ where always $\underline{\pi}_{ij} \leq \overline{\pi}_{ij}$ for each $(i,j) \in E$. The price vector is denoted by p and define as $p = (p_{ij} \in \mathbb{Z} | (i,j) \in E)$ always satisfies $\underline{\pi}_{ij} \leq p_{ij} \leq \overline{\pi}_{ij}$.

Since each participant has preferences over the participants of the other set, so the preferences of sellers over buyers and buyers over sellers is given by the utility function $f_{ij}(x)$ and $f_{ji}(-x)$ for each $(i,j) \in E$. Here $f_{ij}(x)$ denotes the utility to seller $i \in U$, when he or she trade with buyer $j \in V$, and get an amount x from buyer $j \in V$. Similarly, $f_{ji}(-x)$ represents the utility to buyer $j \in V$, when he or she trade with seller $i \in U$, and pays an amount x of money. Here the preferences of participants are not strict because it is based on monetary transfer. So, if we decrease money monotonically then the participants will change their preferences according to the new price.

3.2 The Buyer Seller Sequential Mechanism

Since preferences of participants are given by general increasing valuation functions $f_{ij}(x)$ and $f_{ji}(-x)$ so if $f_{ij}(x) \geq 0$, then we shall say that seller i is ready to make a partnership with buyer j at amount x. This means that buyer j is acceptable to seller i at amount x. Also, if $f_{ji}(-x) \geq 0$ then we say that buyer j is ready to make a partnership with seller i at amount x of money. If $f_{i_0 j_0}(x_1) > f_{i_0 j_1}(x_1)$, then we can say that seller i_0 *prefers* buyer j_0 to buyer j_1 at money x_1 where $i_0 \in U$ and $j_0, j_1 \in V$ and $x_1 \in \mathbb{Z}$. If $f_{j_0 i_0}(-x_1) > f_{j_0 i_1}(-x_1)$, then we can say that j_0 *prefers* i_0 to i_1 at money x_1 where $i_0, i_1 \in U$ and $j_0 \in V$ and $x_1 \in \mathbb{Z}$. If $f_{i_0 j_0}(x_1) = f_{i_0 j_1}(x_1)$, then seller i_0 is *indifferent* between j_0 and j_1 at money x_1. Also, if $f_{j_0 i_0}(-x_1) = f_{j_0 i_1}(-x_1)$, then buyer j_0 is said to be *indifferent* between i_0 and i_1 at money x_1. If $f_{ij}(x) = 0$, then seller i is indifferent between the buyer j and himself at x. If $f_{ji}(-x) = 0$ for some $x \in \mathbb{Z}$, then buyer j is indifferent between the seller i and himself at x. Preferences of participants are not strict in our model because the monetary transfer is allowed between participants of both sets.

3.3 The Supply and Demand Characterization of Stable Matchings

In this section, we describe the characteristic of an outcome for which it would be stable.

Let E is the set of all possible buyer-seller pairs. A subset X, of a set E, is called matching if every agent appear at most once in X. A matching X is said to be pairwise stable if it is individually rational and is not blocked by any buyer-seller pair. A 4-tuple $(X; p; q; r)$ of a matching X and a feasible price vector p is said to be a *pairwise-stable outcome* if the following two conditions are satisfied:

(p1) $q \geq \mathbf{0}$ and $r \geq \mathbf{0}$,

(p2) $f_{ij}(c) \leq q_i$ or $f_{ji}(-c) \leq r_j$ for all $c \in [\underline{\pi}_{ij}, \overline{\pi}_{ij}]_{\mathbb{Z}}$ and for all $(i, j) \in E$.

Where $(q, r) \in \mathbb{R}^U \times \mathbb{R}^V$ is defined by

$$q_i = \begin{cases} f_{ij}(p_{ij}) & \text{if } (i,j) \in X \text{ for some } j \in V \\ 0 & \text{otherwise} \end{cases} \qquad (i \in U), \tag{3.3.1}$$

$$r_j = \begin{cases} f_{ji}(-p_{ij}) & \text{if } (i,j) \in X \text{ for some } i \in U \\ 0 & \text{otherwise} \end{cases} \qquad (j \in V). \tag{3.3.2}$$

Condition (p1) says that the matching X is individually rational. Condition (p2) means $(X; p, q, r)$ is not blocked by any buyer-seller pair. A matching X is said to be *pairwise-stable* if $(X; p, q, r)$ is pairwise-stable.

To show the existence of pairwise-stable outcome in the model defined in section (3.1), we first need to calculate price vector p for each buyer-seller pairs. Since prices should be feasible and $p_{ij} \in \mathbb{Z}$ for each $(i, j) \in E$, so initially we define it by

$$p_{ij} = \begin{cases} \overline{\pi}_{ij} & \text{if } f_{ji}(-\overline{\pi}_{ij}) \geq 0 \\ \max\left\{\underline{\pi}_{ij}, \left\lfloor -f_{ji}^{-1}(0) \right\rfloor\right\} & \text{otherwise.} \end{cases} \tag{3.3.3}$$

Before describing the algorithm mathematically, we define few subsets of set E that help us to find a matching X satisfying condition (p1). Firstly, we define the subset K_0 and T_0 of set E, that contain those buyer-seller pairs from the set E that are not mutually acceptable, as:

$$K_0 = \{(i, j) \in E \mid f_{ji}(-p_{ij}) < 0\}, \tag{3.3.4}$$

$$T_0 = \{(i, j) \in E \mid f_{ij}(-p_{ij}) < 0\}. \tag{3.3.5}$$

K_0 is the set of all those pairs where buyer is not ready to trade with seller and T_0 is the set of all those pairs where seller is not ready to trade with buyer. Now the set of mutually acceptable buyer-seller pairs is defined as:

$$\widetilde{E} = E \setminus \{K_0 \cup T_0\}. \tag{3.3.6}$$

Define $\tilde{q}_i$ for each $i \in U$, and $\widetilde{E}_P$ by (3.3.7) and (3.3.8)

$$\tilde{q}_i = \max\{f_{ij}(p_{ij}) \mid (i,j) \in \widetilde{E}\}, \tag{3.3.7}$$

and

$$\widetilde{E}_P = \{(i,j) \in \widetilde{E} \mid f_{ij}(p_{ij}) = \tilde{q}_i\}. \tag{3.3.8}$$

The maximum over an empty set is taken to be zero by definition. Here the set $\widetilde{E}_P$ contains those buyer-seller pairs which are mutually acceptable and the buyer is most preferred for seller out of all acceptable buyers. Initially consider vector $r = \mathbf{0}$ and the subset $\widehat{E}_P$ of $\widetilde{E}_P$ is defined by:

$$\widehat{E}_P = \{(i,j) \in \widetilde{E}_P \mid f_{ji}(-p_{ij}) \geq r_j\}. \tag{3.3.9}$$

Initially $\widehat{E}_P$ will coincide with $\widetilde{E}_P$. However, in the further iterations of the algorithm $\widehat{E}_P$ may be a proper subset of $\widetilde{E}_P$.

Since we have no matching X at the start of the algorithm, so consider $\widetilde{V} = \emptyset$, where $\widetilde{V}$ denotes the set of matched buyers in X, that is,

$$\widetilde{V} = \{j \in V \mid j \text{ is matched in } X\}. \tag{3.3.10}$$

If $\widetilde{V} = \emptyset$, then there is no matched buyer in matching X. At each step in the algorithm, the matching X in the bipartite graph $(U, V; \widehat{E}_P)$ must satisfies the following conditions:

(s1) X matches all members of $\widetilde{V}$,

(s2) X maximizes $\sum_{(i,j) \in X} f_{ji}(-p_{ij})$ among the matchings that satisfy (s1).

Initially $\widetilde{V} = \emptyset$, so any matching satisfy (s1). Upto these Steps the outcome (X, p, q, r) obviously satisfies the condition (p1). To satisfy the condition (p2) we define the set K of all those buyer-seller pairs that are mutually acceptable and the buyer is most preferred for seller but the seller is unmatched in X by

$$K = \{(i,j) \in \widetilde{E}_P \mid i \text{ is unmatched in } X\}. \tag{3.3.11}$$

Since the set $\widetilde{E}_P$ contains such pairs that are mutually acceptable and the buyer is most preferred for seller out of all acceptable buyers. So there may exists a case in

$\widetilde{E}_P$ when seller i is indifferent between two distinct buyers. Thus it is totally based on the seller i to which particular buyer he wants to trade and the rest of unmatched pairs will go in set K.

If $K = \emptyset$, then there is no need to modify price vector p and define further sets but if K is not empty then we will modify price vector at each iteration of Step 3 by keeping condition (p1) preserved. The new price vector must also be feasible, that is, $\underline{\pi}_{ij} \leq \tilde{p}_{ij} \leq \overline{\pi}_{ij}$. Since we are considering increasing functions therefore we can find a real number $m^*_{ij} \in \mathbb{R}^{++}$ for each $(i,j) \in K$, to modify price vector p, such that

$$f_{ji}(-(p_{ij} - m^*_{ij})) = r_j.$$

Since we are dealing with discrete prices so we will define an integer m_{ij} as follows:

$$m_{ij} = \max\left\{1, \lceil m^*_{ij} \rceil\right\}. \tag{3.3.12}$$

Now, we have

$$f_{ji}(-(p_{ij} - m_{ij})) \geq r_j,$$

where $p_{ij} - m_{ij}$ is an integer and m_{ij} is the minimum positive integer that satisfies the above condition. This means that

$$f_{ji}(-(p_{ij} - (m_{ij} - 1))) \leq r_j.$$

Here the integer m_{ij} for each $(i,j) \in K$ helps us in finding the new price vector such that condition (p2) also satisfies. Now we define a subset L of K that contain those pairs from the set K for which modified price does not remain feasible.

$$L = \{(i,j) \in K \mid p_{ij} - m_{ij} < \underline{\pi}_{ij}\}. \tag{3.3.13}$$

The modified price vector $\tilde{p}$ must also be feasible and is defined by:

$$\tilde{p}_{ij} := \begin{cases} \max\{\underline{\pi}_{ij}, p_{ij} - m_{ij}\} & \text{if } (i,j) \in K \\ p_{ij} & \text{otherwise} \end{cases} \qquad (i,j) \in E. \tag{3.3.14}$$

We also define a subset $\widetilde{T}_0$ of K by:

$$\widetilde{T}_0 := \{(i,j) \in K \mid f_{ij}(\tilde{p}_{ij}) < 0\}. \tag{3.3.15}$$

Remark Throughout in the algorithm our modified price vector will be always decreasing and the size of matching X will be increasing. Also, the participants will change their preferences according to new price vector.

3.4 An Algorithm for Finding a Pairwise Stability

In this section, we propose an algorithm for finding a pairwise stable outcome for the model described in Section 3.2.

Input: Two disjoint and finite sets U and V, the set of ordered pairs $E = U \times V$, price vector $p \in \mathbb{Z}^E$, two vectors $\underline{\pi} \in \mathbb{Z}^E$ and $\overline{\pi} \in \mathbb{Z}^E$ where $\underline{\pi} \leq \overline{\pi}$, general increasing functions .

Output: Vectors $(q, r) \in \mathbb{R}^U \times \mathbb{R}^V$, and $p \in \mathbb{Z}^E$ must satisfy $(p1)$ and $(p2)$.

Step 0: Firstly put $\widetilde{V} = \emptyset$ and $r = \mathbf{0}$. Initially define p, K_0, T_0, $\widetilde{E}$, $\tilde{q}$, $\widetilde{E}_P$ and $\widehat{E}_P$ by (3.3.3)$-$(3.3.9), respectively and find a matching X in the bipartite graph $(U, V; \widehat{E}_P)$ satisfying (s1) and (s2). Define r, $\widetilde{V}$ and K by (3.3.2), (3.3.10) and (3.3.11), respectively.

Step 1: If $K = \emptyset$ then define q by (3.3.1) and stop. Otherwise go to Step 2.

Step 2: For each $(i, j) \in K$ calculate m_{ij} by (3.3.12) and new price vector $\tilde{p}$ by (3.3.14). Define L and $\widetilde{T}_0$ by (3.3.13) and (3.3.15), respectively and update T_0 by $T_0 := T_0 \cup \widetilde{T}_0$ and K_0 by $K_0 := K_0 \cup L$.

Step 3: Replace price vector p by $\tilde{p}$ and modify $\widetilde{E}$ by:

$$\widetilde{E} := \widetilde{E} \setminus \{K_0 \cup T_0\}. \tag{3.4.1}$$

Again define $\tilde{q}$ by (3.3.7) and modify $\widetilde{E}_P$, $\widehat{E}_P$ by (3.3.8) and (3.3.9) respectively, for the updated p and $\widetilde{E}$. Find a matching X in the bipartite graph $(U, V; \widehat{E}_P)$ that satisfies the conditions (s1) and (s2). Again define r, $\widetilde{V}$ and K by (3.3.2), (3.3.10) and (3.3.11), respectively. Go to Step 1.

We illustrate with an example the mechanism of the above algorithm and introduce briefly our results.

Example 3.4.1. Assume $n = 4$ and $U = \{i_o, i_1, i_2, i_3\}$, $V = \{j_o, j_1, j_2, j_3\}$ where U and V are set of buyers and sellers. Let $E = U \times V$ denote the set of all possible buyer-seller pairs. Lower and upper bounds for each $(i,j) \in E$ is defined as

$$\begin{aligned} \underline{\pi}_{ij} &= -2 \qquad \forall (i,j) \in E, \\ \overline{\pi}_{ij_0} &= 3 = \overline{\pi}_{ij_1} \quad \forall i \in U, \\ \overline{\pi}_{ij_2} &= 2 = \overline{\pi}_{ij_3} \quad \forall i \in U. \end{aligned}$$

To find the valuations of $f_{ij}(x) = \alpha_{ij}e^x + \beta_{ij}$ and $f_{ji}(-x) = \alpha_{ji}e^{-x} + \beta_{ji}$, we fixed the values of α_{ij} and α_{ji} as $\alpha_{ij} = 3$, $\alpha_{ji} = 4$ for each $(i,j) \in E$ and the values of β_{ij}, β_{ji} for each $(i,j) \in E$ is shown in Table (3.1), (3.2):

Now we apply the algorithm starting with Step 0. At Step 0 we have $r = \mathbf{0}$ and $\widetilde{V} = \emptyset$. By (3.3.3), corresponding components of price vector is

β_{ij}	j_0	j_1	j_2	j_3
i_0	2	-61	5	42
i_1	0	1	3	40
i_2	30	8	47	6
i_3	6	7	-5	-4

Table 3.1: β_{ij} for $(i,j) \in E$

β_{ji}	i_0	i_1	i_2	i_3
j_0	6	12	-15	9
j_1	1	10	8	12
j_2	7	6	12	-3
j_3	11	0	-1	14

Table 3.2: β_{ji} for $(i,j) \in E$

$$p = (3, 3, 2, 2, 3, 3, 2, 2, 1, 3, 2, -1, 3, 3, -1, 2).$$

Now by using the values of α_{ij}, α_{ji}, β_{ij}, β_{ji} and the price vectors obtained at Step 0 we find the values of $f_{ij}(p_{ij})$ and $f_{ji}(-p_{ij})$ for each $(i,j) \in E$ in the Table (3.3), (3.4).

$f_{ij}(p_{ij})$	j_0	j_1	j_2	j_3
i_0	62.25	-0.74	27.16	64.17
i_1	60.25	61.25	25.17	62.17
i_2	38.15	68.25	69.16	7.10
i_3	66.25	67.25	-3.89	18.16

Table 3.3: $f_{ij}(p_{ij})$ for $(i,j) \in E$

$f_{ji}(-p_{ij})$	i_0	i_1	i_2	i_3
j_0	6.19	12.19	-13.52	9.20
j_1	1.20	10.20	8.20	12.19
j_2	7.54	6.54	12.54	7.87
j_3	11.54	0.54	9.87	14.54

Table 3.4: $f_{ji}(-p_{ij})$ for $(i,j) \in E$

From the values given in Table (3.3), (3.4) we find that $f_{i_0 j_1}(-p_{i_0 j_1})$, $f_{i_3 j_2}(-p_{i_3 j_2})$ and $f_{j_0 i_2}(-p_{i_2 j_0})$ are negative, therefore we get $K_0 = \{i_2, j_0\}$ and $T_0 = \{(i_0, j_1), (i_3, j_2)\}$. Exclude these pairs from set E and set of mutually acceptable buyer-seller pairs is:

$$\widetilde{E} = E \setminus \{(i_2, j_0), (i_0, j_1), (i_3, j_2)\}.$$

Using (3.3.7), we obtain

$$\tilde{q}_{i_0} = 64.17, \quad \tilde{q}_{i_1} = 62.17, \quad \tilde{q}_{i_2} = 69.16, \quad \tilde{q}_{i_3} = 67.25.$$

Also by (3.3.8), we find

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_3), (i_2, j_2), (i_3, j_1)\}.$$

As the subset $\widehat{E}_P$ of $\widetilde{E}_P$ is defined in (3.3.9) and initially $r = \mathbf{0}$, so we have $\widehat{E}_P = \widetilde{E}_P$. From the bipartite graph $(U, V, \widehat{E}_P)$, find a matching X that satisfies the condition (s1) and (s2). Thus the required matching that satisfies both conditions is $X = \{(i_0, j_3), (i_2, j_2), (i_3, j_1)\}$. From the matching X we find r by (3.3.2) and get:

$$r_{j_0} = 0, \quad r_{j_1} = 12.19, \quad r_{j_2} = 12.54, \quad r_{j_3} = 11.54.$$

Also from the matching X updating $\widetilde{V}$, that contains matched buyers in X, Thus

$$\widetilde{V} = \{j_1, j_2, j_3\}.$$

Since set K, defined in (3.3.11), contain such pairs in which we have $f_{ij}(-p_{ij}) \leq r_j$. Thus $K = \{(i_1, j_3)\}$. Here the set K contain such pair in which buyer is most preferred for seller but seller is unmatched in X. As $K \neq \emptyset$, so we move to Step 2 of algorithm and find the integer $m_{i_1 j_3}$ by (3.3.12). Since we are taking the general function particularly exponential function. There must exist a number $m_{i_1 j_3}$ such that $f_{j_3 i_1}(-(p_{i_1 j_3} - m_{i_1 j_3})) \geq r_{j_3}$. So $m_{i_1 j_3} = 1$. To find matching X that satisfies condition (s2), define a subset L of K by (3.3.13). Here $L = \emptyset$. Modify the corresponding components of price vector $\tilde{p}$ for the pairs that are in K by (3.3.14). All the components of price vector $\tilde{p}$ remain unchanged except $\tilde{p}_{i_1 j_3}$ so :

$$\tilde{p} = (3, 3, 2, 2, 3, 3, 2, 1, 1, 3, 2, -1, 3, 3, -1, 2).$$

Also all the valuations in Table (3.3), (3.4) remain unchanged except $f_{i_1j_3}(\tilde{p}_{i_1j_3})$ and $f_{j_3i_1}(\tilde{p}_{i_1j_3})$. Thus modified value of these valuations is $f_{i_1j_3}(1) = 48.15$ and $f_{j_3i_1}(-1) = 1.47$.

For the pairs in K we define the set $\widetilde{T}_0$ by (3.3.15). Since $f_{i_1j_3}(-1)$ is not less than 0 so $\widetilde{T}_0 = \emptyset$. As both sets L_0 and $\widetilde{T}_0$ are empty. So L_0 and E_0 remain unchanged for the modified price vector $\tilde{p}$, therefore modified $\widetilde{E}$ remain unchanged in Step 3. Again by (3.3.7) we have

$$\tilde{q}_{i_0} = 64.17, \quad \tilde{q}_{i_1} = 61.25, \quad \tilde{q}_{i_2} = 69.16, \quad \tilde{q}_{i_3} = 67.25.$$

Updating $\widetilde{E}_P$ and $\widehat{E}_P$ by (3.3.8) and (3.3.9), respectively, we get

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_1)\},$$

and

$$\widehat{E}_P = \{(i_0, j_3), (i_1, j_1), (i_2, j_2), (i_3, j_1)\}.$$

The matching X in the above bipartite graph $(U, V, \widehat{E}_P)$ that satisfies both (s1) and (s2) condition is:

$$X = \{(i_0, j_3), (i_3, j_1), (i_2, j_2)\}.$$

Again update r by (3.3.2) we have

$$r_{j_0} = 0, \quad r_{j_1} = 12.19, \quad r_{j_2} = 12.54, \quad r_{j_3} = 11.54.$$

Updated $\widetilde{V}$ is

$$\widetilde{V} = \{j_1, j_2, j_3\}.$$

Also updated K is

$$K = \{(i_1, j_1)\}.$$

Since $K \neq \emptyset$, we move to Step 2 and calculate the value of $m_{i_1j_1}$ by (3.3.12) we get $m_{i_1j_1} = 4$ by equation (3.3.12). All the components of modified price vector remain unchanged except $p_{i_1j_1}$ so:

$$\tilde{p} = (3, 3, 2, 2, 3, -1, 2, 2, 1, 3, 2, -1, 3, 3, -1, 2).$$

Again all the valuations in Table (3.3), (3.4) remain unchanged except $f_{i_1j_1}(\tilde{p}_{i_1j_1})$ and $f_{j_1i_1}(\tilde{p}_{i_1j_1})$. Thus modified value of these valuations are $f_{i_1j_1}(-1) = 2.10$ and

$f_{j_1 i_1}(1) = 20.87$. Both sets L and $\widetilde{T}_0$ are empty. So L_0 and E_0 remain unchanged for the modified price vector $\tilde{p}$. At Step 3, Since both E_0 and L_0 were unchanged therefore modified $\widetilde{E}$ remain unchanged. Again by (3.3.7) we have

$$\tilde{q}_{i_0} = 64.17, \quad \tilde{q}_{i_1} = 60.25, \quad \tilde{q}_{i_2} = 69.16, \quad \tilde{q}_{i_3} = 67.25.$$

Updating $\widetilde{E}_P$ and $\widehat{E}_P$ by (3.3.8) and (3.3.9), respectively, we get

$$\widetilde{E}_P = \{(i_0, j_3), (i_1, j_0), (i_2, j_2), (i_3, j_1)\},$$

and $\widetilde{E}_P = \widehat{E}_P$. The matching X in the bipartite graph $(U, V, \widehat{E}_P)$ is

$$X = \{(i_0, j_3), (i_1, j_0), (i_2, j_2), (i_3, j_1)\}.$$

Here $K = \emptyset$, that is all the buyers and sellers are matched so algorithm terminates. The matching X obviously satisfy (p1), (p2) conditions. So matching X is pairwise stable.

3.5 The Main Results

In this section we will show that the core is non-empty by replacing the preferences of participants from linear increasing function to general increasing function. The common argument can be expressed in the same language as expressed in Ali and Farooq [1]. It turns out that the proof of the lemma's and theorems for their discrete model are compatible to the model treated here up to some small changes. The small changes lie essentially in the fact that the preferences of participants are replaced from linear increasing function to general increasing function. We will add prefixes $(old)*$ and $(new)*$ to sets/vectors/integers before and after update, respectively, in any iteration of the algorithm. The key result is Lemma 3.5.1 which will be proved here using the assumption defined in equation 3.3.12. The proof of Lemma 3.5.2 is the direct consequence of Lemma 3.5.1 that can be proof by following the same steps of Ali and Farooq [1] Lemma 3.4 but uses general increasing function instead of linear function. The remaining results follow from it. The proof of Theorem 3.5.5 can be seen in the paper of Ali and Farooq [1] in the original version. So we will not prove them here.

Lemma 3.5.1. *In each iteration of the algorithm at step 3 we have $f_{ji}(-(p_{ij} - m_{ij})) \geq r_j$ for each $(i,j) \in K$. Furthermore, if $f_{ji}(-(p_{ij} - m_{ij})) > r_j$ for some $(i,j) \in K$ then $p_{ij} - m_{ij}$ is the maximum integer for which this inequality holds.*

Proof. Let at Step 2 there exist pair $(i,j) \in K$ such that $f_{ji}(-(old)p_{ij}) \leq r_j$. At Step 2 we calculated an integer m_{ij} by (3.3.12) with the property for each $(i,j) \in K$ at Step 3 we have

$$f_{ji}(-((old)p_{ij} - m_{ij}) \geq r_j.$$

Next, we prove the second part of theorem that if $f_{ji}(-((old)p_{ij} - m_{ij})) > r_j$ then $(old)p_{ij} - m_{ij}$ is the maximum integer for which this holds. Since by (3.3.12) we have $m_{ij} \geq 1$ so we first consider the case when $m^*_{ij} < 1$, that is $m_{ij} = 1$ by (3.3.12). By (3.3.11) we have

$$f_{ji}(-(old)p_{ij}) \leq r_j$$

but

$$r_j < f_{ji}(-(old)p_{ij} - 1).$$

So for this case results holds. Now, consider when $m^*_{ij} > 1$ then $m_{ij} = \lceil m^*_{ij} \rceil \geq m^*_{ij}$ by (3.3.12). Since m^*_{ij} is a real number for which we have

$$r_j = f_{ji}(-(old)p_{ij} + m^*_{ij}).$$

Also for $\delta > 0$, we have

$$f_{ji}(-((old)p_{ij} - (m^*_{ij} + \delta))) > r_j > f_{ji}(-((old)p_{ij} - (m^*_{ij} - \delta))). \tag{3.5.1}$$

Since

$$m_{ij} = \lceil m^*_{ij} \rceil \geq m^*_{ij}. \tag{3.5.2}$$

Therefore, by (3.5.2) we can say that $(old)p_{ij} - m_{ij}$ is the maximum integer for which (3.5.1) holds.

□

Lemma 3.5.1 is the main result of our model because it permits us to create a new price vector by keeping the condition of mutually acceptability and feasibility of price vector.

Lemma 3.5.2. *For each $(i,j) \in L$ we have $(new)p_{ij} = \underline{\pi}_{ij}$ and $f_{ji}(-(new)p_{ij}) \leq (old)r_j$, where L is defined at Step 2.*

Proof. Since the set L contain those pairs from set K where price is not feasible. So, by (3.3.13), for each $(i,j) \in L$ we get the first part of the assertion that is $(new)p_{ij} = \underline{\pi}_{ij}$. Next, we prove the second part of lemma. Since $(new)p_{ij} > (old)p_{ij} - m_{ij}$, therefore, $f_{ji}(-(new)p_{ij}) < f_{ji}(-((old)p_{ij} - m_{ij}))$. By lemma 3.5.1, we have $f_{ji}(-(p_{ij} - m_{ij})) \geq r_j$. If $f_{ji}(-(p_{ij} - m_{ij})) = r_j$ then the result trivially holds. If $f_{ji}(-(p_{ij} - m_{ij})) > r_j$ then by lemma 3.5.1, $p_{ij} - m_{ij}$ is the maximum integer for which this inequality holds. Since $(new)p_{ij} > p_{ij} - m_{ij}$, therefore, $f_{ji}(-(new)p_{ij}) \leq r_j$. This completes the proof.

The next result shows that in the bipartite graph $(U, V; \widehat{E}_P)$ we can find a matching X that satisfies the condition (s1) and (s2).

Lemma 3.5.3. *There exists a matching X in the bipartite graph $(U, V; \widehat{E}_P)$ that satisfy condition (s1) and (s2) in each iteration of the algorithm at Step 3.*

Proof. If we show that $(old)X \subseteq (new)\widehat{E}_P$ in each iteration at Step 3 then lemma is proof. Since at Step 0 we find the set T_0 by (3.3.5) and at Step 2 we updated T_0 by $T_0 := T_0 \cup \widetilde{T}_0$ if $\widetilde{T}_0$ is nonempty then use updated value of T_0 otherwise it will remain the same. Also $L, \widetilde{T}_0 \subseteq K$ and $K \cap (old)X = \emptyset$ at Step 2. Therefore, (3.3.14) and (3.4.1) imply that $(old)X \subseteq (new)\widetilde{E}_P$ at Step 3. By (3.3.9), $(old)r_j$ is the lower bound of $f_{ji}(-(new)p_{ij})$ for each $(i,j) \in (new)\widehat{E}_P$, therefore, $(old)X \subseteq (new)\widehat{E}_P$. □

Lemma 3.5.4. *In each iteration of the algorithm, following hold:*

(i) *The price vector p decreases or remains same. In particular, if $K \setminus \{L \cup \widetilde{T}_0\} \neq \emptyset$ at Step 2 then p_{ij} decreases at Step 3 for all $(i,j) \in K \setminus \{L \cup \widetilde{T}_0\}$.*

(ii) *$\widetilde{E}$ reduces or remains same. In particular, if $L \neq \emptyset$ or $\widetilde{T}_0 \neq \emptyset$ at Step 2 then $\widetilde{E}$ reduces at Step 3.*

(iii) *The vector r increases or remains same.*

Proof. (i) Initially the price vector p is defined by (3.3.3) and in each iteration it is modified by (3.3.14). From (3.3.14), one can easily see that p decreases or remains same at Step 3. If $K \neq \emptyset$ then we find $\tilde{p}$ by (3.3.14) at Step 2. For each $(i, j) \in K$, m_{ij} is a positive integer. Now, if $K \setminus \{L \cup \widetilde{T}_0\} \neq \emptyset$ then one can easily see from (3.3.14) that $\tilde{p}_{ij} = (old)p_{ij} - m_{ij}$ for all $(i, j) \in K \setminus \{L \cup \widetilde{T}_0\}$ at Step 2. This proves the assertion.

(ii) Initially $\widetilde{E}$ is defined by (3.3.6) at Step 0 and it is modified by (3.4.1) at Step 3 in each iteration. If $L = \widetilde{T}_0 = \emptyset$ at Step 2 then $\widetilde{E}$ remains same at Step 3. If $L \neq \emptyset$ at Step 2 then $\widetilde{E}$ reduces at Step 3. If $\widetilde{T}_0 \neq \emptyset$ at Step 2 then T_0 enlarges at Step 2 and consequently, $\widetilde{E}$ reduces at Step 3.

(iii) By Lemma 3.5.3, we have $(old)X \subseteq (new)\widehat{E}_P$ in each iteration at Step 3 and from the above lemma we see that for each $(i, j) \in (new)\hat{E}_P$, we have $f_{ji}(-(new)p_{ij}) \geq (old)r_j$ and $(old)r_j = 0$ for each $j \in V \setminus (old)\widetilde{V}$ by (3.3.2). Since $(new)X \subseteq (new)\widehat{E}_P$ and by (s1), we have $(old)\widetilde{V} \subseteq (new)\widetilde{V}$ therefore, $(new)r_j = f_{ji}(-(new)p_{ij}) \geq (old)r_j$ for each $(i, j) \in (new)X$. Further, $(new)r_j = (old)r_j = 0$ for each $j \in V \setminus (new)\widetilde{V}$. Hence, the vector r increases or remains same.

□

Theorem 3.5.5. *The outcome (X, p, q, r) must satisfies the condition (p1) and (p2) if algorithm terminates.*

Theorem 3.5.6. *The algorithm terminates after finite number of iterations.*

Proof. In each iteration of the algorithm at Step 2, set L and $\widetilde{T}_0$ both are empty or at least one of them is non-empty. First consider the case when both sets $L = \widetilde{T}_0 = \emptyset$. Since p is discrete and bounded so by Lemma 3.5.4 (i), for each $(i, j) \in K$ price vector p_{ij} decreases or remain the same. Therefore, the price vector p can be decreased finite number of times.

Next, consider the case when $L \neq \emptyset$ or $\widetilde{T}_0 \neq \emptyset$. Now by Lemma 3.5.4 (ii) either $\widetilde{E}$ reduces or remain the same in each iteration of the algorithm at Step 3. Therefore, this case is possible at most $|E|$ times. This completes the proof. □

3.6 Open Problem

- We have generalized a one-to-one matching model of Ali and Farooq from linear valuations to non linear valuations. In our generalized model, we assume that the feasible price vector p is an integer vector, that is, $p \in \mathbb{Z}^E$ and showed the existence of pairwise stable outcome in our generalized model.

- It is an open problem to find a pairwise stable outcome in our model by assuming $p \in \mathbb{R}^E$. In such a case Farooq and Ali and Farooq models become a special case for our generalized model.

Bibliography

[1] Y. Ali and R. Farooq: Pairwise stability in a two-sided matching market with indivisible goods and money. *Journal of the Operations Research Society of Japan*, **54** (2011), 1-11.

[2] J. A. Bondy and U. S. R. Murty: Graph theory with applications. *Macmillan London*, Canada, (1982).

[3] V. P. Crawford and E. M. Knoer: Job matching with heterogeneous firms and workers. *Econometrica*, **49** (1981), 437–450.

[4] G. B. Dantzig: Linear programming and extensions. *Princeton University Press*, Princeton, 1963.

[5] K. Eriksson and J. Karlander: Stable matching in a common generalization of the marriage and assignment Models. *Discrete Mathematics*. **217** (2000), 135 − 156.

[6] R. Farooq: A polynomial-time algorithm for a stable matching problem with linear valuations and bounded side payments. *Japan Journal of Industrial and Applied Mathematics*, **25** (2008), 83-98.

[7] D. Gale and L.S. Shapley: College admissions and the stability of marriage. *American Mathematical Monthly*, **69** (1962), 9-15.

[8] D. Gusfield and R. W. Iriving: The stable marriage problem structure and algorithms. *MIT Press*, 1989.

[9] P. Hall: On representatives of subsets. *Journal of the London Mathematical Society*, **10** (1935), 26-30.

[10] A. M. Natarajan, P. Balasubramani and A. Tamilarasi: Operation reasearch. *PEARSON Education*.

[11] A. Roth and M. Sotomayor: Two-sided matching. A study in game-theoretic modeling and analysis. *Econometric Society Monographs*, **18**, Cambridge University Press.

[12] A. E. Roth and M. A. O. Sotomayor: Stable outcomes in discrete and continuous models of two-sided matching. A unified treatment. *The Brazilian Review of Econometrics*, **16** (1996), 1-24.

[13] K. H. Rosen: Discrete mathematics and its applications with combinatorics and graph theory seven edition.

[14] L. S. Shapley and M. Shubik: The assignment game I: The Core. *International Journal of Game Theory*, **1** (1972), 111-130.

[15] M. Sotomayor: Existence of stable outcomes and the lattice property for a unified matching market. *Mathematical Social Sciences*, **39** (2000), 119-132.