Academic College of Tel-Aviv Yaffo

School of Computer Science

Partial matching on grids using Manhattan distances

Author
Hila Ohana-Brumer

Supervisors
Prof. Michal Stern
Dr. Nili Guttmann-Beck

May 4, 2022

Contents

1 Abstract

Let $G = (V, E)$ be a complete undirected graph where V is a set of vertices on a grid and E is the set of edges, such that every edge $e \in E$ has a non-negative length measured by Manhattan topology. The problem is to find a maximum perfect matching M, i.e., find vertex disjoint edges that cover all vertices, such that the sum of the length of the edges in M is maximized. In our work we will focus on the problem of perfect maximum matching. A maximum matching is perfect when it is a full coverage of the graph vertices. We will deal with finding a maximum matching on a non-directed graph whose vertices are arranged on a number of parallel lines called banks on three dimensions grid.

TODO - Note (hand writing) to add vertex in wrong places.

2 Introduction

Let $G = (V, E)$ be a complete undirected graph where V is a set of vertices on a grid and E is the set of edges, such that every edge $e \in E$ has a non-negative length measured by Manhattan topology. The problem is to find a maximum perfect matching M, i.e., find vertex disjoint edges that cover all vertices, such that the sum of the length of the edges in M is maximized. In our work we will focus on the problem of perfect maximum matching. Maximum matching is perfect when matching M is full coverage of the graph vertices. We will deal with finding a maximum matching on a non-directed graph whose vertices are arranged on a number of parallel lines called banks on three dimensions grid. Since we focus on a restricted case with a specific topology, our algorithms improve the complexity of the well known algorithms regarding maximum perfect matching.

Position each given vertex using coordinates (x, y, z). The distance function that defines the distance from one vertex to another is calculated based on the Manhattan distance, that is, given two vertices with coordinates $v_1 = (x_1, y_1, z_1)$ and $v_2 = (x_2, y_2, z_2)$, the distance is calculated as follows: $d(v_1, v_2) = |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2|$.

Our research introduces algorithms for three problems.
The first algorithm is an algorithm for finding a maximum matching on a graph with four balanced banks in the three-dimensional Manhattan space. Through our work we assume that $\mathcal{Y} \geq \mathcal{Z}$ where Y banks values can be 0, $\mathcal{Y}$, or $2\mathcal{Y}$ and the Z banks values can be 0 or $\mathcal{Z}$.
Balanced banks are defined to be a graph where the number of vertices is equal on each bank. The algorithm and proof presents a unique approach for solving the maximum matching problem. First the algorithm creates a matching which is optimal on the Y axis and on the Z axis and then improves the matching only if the value that can be achieved on the X axis is greater than the value that achieved on the Z axis, since we assume that $\mathcal{Y} \geq \mathcal{Z}$.
The second problem we introduce is an algorithm for finding a maximum matching on a graph with four unbalanced banks in the three-dimensional Manhattan space. Unbalanced banks is defined to be a graph where the number of vertices on each bank is not equal.
The third problem we introduce is an algorithm for finding a maximum matching on a graph with six balanced banks in the three-dimensional Manhattan space. The six balanced graph is a combination of two Z planes and three Y planes, such that $\mathcal{Y} \geq \mathcal{Z}$, and the number of vertices on each bank is equal. The algorithm finds a local maximum matching.

The problem of maximum matching is one of the classic problems of graph theory, and many studies deal with this problem. Edmonds in [2] finds a polynomial algorithm for solving the problem and thus proves that the problem is solvable in polynomial time. His work serves as a platform for many studies even these days. There is a special interest in solving the problem on different families of graphs, for which algorithms with better complexity have been found than the original Edmonds general algorithm. Beyond academic research, the problem of maximum matching has many applicative uses in the fields of engineering, such as image processing, design of printed circuits, and decoding of geographic information. Solving this problem is also relevant for another research problem, which is the problem of harmful facilities. This problem deals with the location of facilities as far away from each other as possible or from population centers, for example: determining the location of nuclear facilities.

Our research is a follow-up to three previous studies. Schwartz introduces in [9] a number of polynomial algorithms for finding a perfect maximum matching and a blocked maximum q-matching. Schwartz solves these problems on a straight-line graph, a graph in which all vertices are on one straight line, and on a two-dimensional two-bank Manhattan graph, a graph in which all vertices are on two parallel lines.
Kostina introduces in [7] a polynomial time algorithm for finding a perfect maximum matching on a two-dimensional Manhattan graph with n banks with equal distances, a graph in which all vertices are on parallel n banks with equal distances.
Third, Berkowitz presents in [1] a polynomial algorithm for finding a perfect maximum matching for the three banks on three dimensions with fixed distances and suggested some ideas towards solutions for four banks on three dimensions. The results of this study are then extended to four balanced banks and four unbalanced banks on the three-dimensional space. Furthermore, we present interesting and fundamental results concerning the case where the vertices are arranged on six balanced banks on three dimensions.

A significant and known result for a maximum matching on general graphs is by Edmonds. In [2], Edmonds describes the Blossom Algorithm for finding the perfect matching on a general graph. The main idea of the algorithm is to shrink odd-length circles to a single vertex while looking for improving paths in the graph. Gabow in [3] improves the efficiency in the algorithm of Edmonds. Hassin, Rubinstein and Tamir in [5] use the algorithm of Edmonds to find a maximum block q-matching on a graph. They propose adding new vertices to the graph, connecting the new vertices to each other in weightless edges, connecting the new vertices to existing vertices in "heavy" edges weighing $K = W(V) + 1$, i.e., a certain maximum weight in the graph. Next, run the algorithm presented by Edmonds. The algorithm selects all the heavy edges we added and the maximum edges in the original graph. Karp, Upfal and Wigderson in [6] present a random algorithm with a polynomial run-time that uses a polynomial number of processors. Glover in [4] present an efficient serial algorithm for finding maximum matching on a two-sided convex graph. This algorithm has been improved in the work of Lipski and Preparata [8], in which they present an even more efficient algorithm for finding maximum matching in these graphs.

Schwartz [9] presents in his work a number of algorithms for finding a perfect maximum matching and a maximum q-matching. In his work, Schwartz deals with a straight line graph and with only two banks with Manhattan topology. Schwartz proves that given a straight line graph, the matching of the extreme vertices is a perfect maximum matching. The $SLMA$ Straight Line Matching Algorithm introduced by Schwartz finds a perfect maximum matching on a straight line graph with time efficiency of $O(|V|log|V|)$. The algorithm works so that in each iteration, a matching is performed between the right extreme vertex, that has not yet been paired on the straight line graph, with the left extreme vertex, that has not yet been paired. Further in his research, Schwartz shows that the heaviest q edges in the matching of the extreme vertices, performed by the above algorithm, gives a maximum q-matching. In addition, Schwartz presents a solution for maximum matching on two-banks Manhattan topology. The algorithm iteratively pairs the heaviest edges of three options:

1. The edge that connects the left extreme vertex in the lower bank with the right extreme vertex in the upper bank.
2. The edge that connects the left extreme vertex at the upper bank with the right extreme vertex at the lower bank.
3. The edge that connects extreme vertices in the dense bank. In a graph where the number of vertices is not equal between the banks, the dense bank is the bank with the largest number of vertices.

The efficiency of all the algorithms presented by Schwartz in his work is $O(|V|log|V|)$. Kostina in [7] has extended the work, presenting a general algorithm for finding a perfect maximum matching for parallel banks on a Manhattan graph with time efficiency of $O(|V|log|V|)$. The core idea of the algorithm is that in order to solve the problem, the vertices can be compressed to one straight line, i.e., to one dimension, and use the algorithm presented by Schwartz. After finding a maximum matching on the straight line, the matching on the original graph can be updated by opening the contruction, so that the sum of the distance values on the X-axis is maximum, and the distances of the Y-axis are not effected. Kostina proved that in this way, a maximum matching for n banks on two dimensions can be found. Berkowitz in [1], presents an algorithm for finding a perfect matching for three banks. In his work, Berkowitz proves that for three banks located in the shape of a "right-angled triangle," the graph can be converted to a two-dimensional graph using Manhattan distances and solved using the work of Kostina. In addition, Berkowitz in his work suggested idea for an algorithm for perfect matching on the three-dimensional case, where there are four banks at equal distances, and the banks are balanced, i.e., there is an equal number of vertices on each bank.

The work is organized as follows:
Section 3 introduces definitions that are crucial for our research.
Section 4 introduces basic lemmas and algorithms, focusing on algorithms for matching edges on one and two banks.
Section 5 introduces general proofs for matching on four banks graph on three dimensions.
Section 6 introduces an algorithm and a proof for perfect maximum matching on four balanced banks graph on three dimensions.
Section 7 introduces an algorithm and a proof for perfect maximum matching on four unbalanced banks graph on three dimensions.
Section 8 introduces an algorithm and a proof for local maximum matching on six balanced banks graph on three dimensions.
Section 9 summarizes our results and gives possible further directions of research.

3 Definitions

Throughout our work, we assume that the vertices of the graph are arranged on lines called banks. For each bank, Y and Z coordinates are constants, and the vertices along the banks diverse according to their X-coordinate. Also, assume that there are two possible values of Z, and without loss of generality, the values are 0 and $\mathcal{Z}$. The values of Y are $0, \mathcal{Y}, 2\mathcal{Y}, \ldots, n\mathcal{Y}$.

Notation 3.1 *In the following discussions we always find prefect matching. For simplicity of writing we will refer to them as matching.*

Notation 3.2 *All edges are ordered according to their $X - value$. So the edge between the vertices v, u will be marked $\boldsymbol{(v, u)}$ if $\boldsymbol{X[v] \leq X[u]}$ and $\boldsymbol{(u, v)}$ otherwise.*

Definition 3.3 *Given a set of edges E on a set of vertices V that are arranged according to their X-coordinate. v_i is one of the vertices of edge e_i.*

Definition 3.4 *For a vertex $v = (v_x, v_y, v_z)$ define $\boldsymbol{X[v]} = v_x$, $\boldsymbol{Y[v]} = v_y$ and $\boldsymbol{Z[v]} = v_z$ to be the X-coordinate, Y-coordinate and Z-coordinate, respectively. Furthermore, for an edge $e = (v_1, v_2)$ define $X[e] = |X[v_1] - X[v_2]|$, $Y[e] = |Y[v_1] - Y[v_2]|$ and $Z[e] = |Z[v_1] - Z[v_2]|$.*

Notation 3.5 *Let $\boldsymbol{B_{(\mathcal{Y},\mathcal{Z})}}$ be the bank whose Y-coordinate is $\boldsymbol{\mathcal{Y}}$ and Z-coordinate is $\boldsymbol{\mathcal{Z}}$. For every $\boldsymbol{v \in B_{(\mathcal{Y},\mathcal{Z})}}$, $\boldsymbol{Y[v] = \mathcal{Y}}$ and $\boldsymbol{Z[v] = \mathcal{Z}}$.*

Definition 3.6 *For a set of edges $F = e_1, \ldots, e_m$, define the X-value of F, $X[F] = \sum_{i=1}^{m} X[e_i]$, the Y-value of F, $Y[F] = \sum_{i=1}^{m} Y[e_i]$ and the Z-value of F, $Z[F] = \sum_{i=1}^{m} Z[e_i]$. Denote and define the value of F as $val(F) = X[F] + Y[F] + Z[F]$.*

Definition 3.7 *Given a graph $G = (V, E)$. Define:*
$\boldsymbol{Xopt} = \max\{X[M] |\ M \text{ is a matching on } G\}$, *and let $\boldsymbol{X - OPT}$ be the set of matchings such that for each matching $M' \in X - OPT$, $X[M'] = Xopt$.*
$\boldsymbol{Yopt} = \max\{Y[M] |\ M \text{ is a matching on } G\}$, *and let $\boldsymbol{Y - OPT}$ be the set of matchings such that for each matching $M' \in Y - OPT$, $Y[M'] = Yopt$.*
$\boldsymbol{Zopt} = \max\{Z[M] |\ M \text{ is a matching on } G\}$, *and let $\boldsymbol{Z - OPT}$ be the set of matchings such that for each matching $M' \in Z - OPT$, $Z[M'] = Zopt$.*
*Define **opt** to be the value of a maximum matching on a graph G that is,*
$opt = max\{val(M) | M \text{ is a matching on } G\}$.

Definition 3.8 *Given a graph $G = (V, E)$. Define a maximum matchings value for each Y and Z values:*
For every $0 \leq i \leq p$, $\boldsymbol{opt_i} = \max\{val(M) \mid M \text{ is a matching on } G,\ Y[M] = Yopt, Z[M] = Zopt - 2i\mathcal{Z}\}$.
In addition, let $\boldsymbol{OPT_i}$ be the corresponding set of matchings, that is
$OPT_i = \{M |\ M \text{ is a matching on } G,\ val(M) = opt_i,\ Y[M] = Yopt, Z[M] = Zopt - 2i\mathcal{Z}\}$.
For example,
$\boldsymbol{opt_0} = \max\{val(M) \mid M \text{ is a matching on } G,\ Y[M] = Yopt,\ Z[M] = Zopt\}$.
$\boldsymbol{opt_1} = \max\{val(M) \mid M \text{ is a matching on } G,\ Y[M] = Yopt,\ Z[M] = Zopt - 2\mathcal{Z}\}$.
...
$\boldsymbol{opt_j} = \max\{val(M) \mid M \text{ is a matching on } G,\ Y[M] = Yopt,\ Z[M] = Zopt - 2j\mathcal{Z}\}$, *for $0 \leq j \leq p$.*

Definition 3.9 *Two edges $e = (v_1, v_2)$ and $f = (u_1, u_2)$ are $\boldsymbol{X - disjoint}$ if $\boldsymbol{X[v_2] < X[u_1]}$ or $\boldsymbol{X[v_1] > X[u_2]}$. For example see Figure 1.*

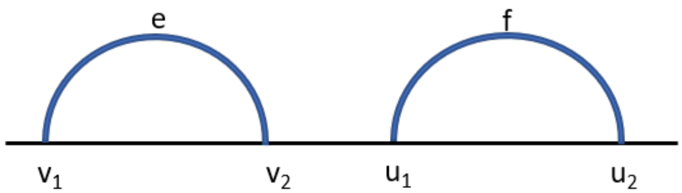

Figure 1: X-disjoint edges

Definition 3.10 *An edge $e = (v_1, v_2)$ is $\boldsymbol{X - contains}$ an edge $\boldsymbol{f = (u_1, u_2)}$ if $\boldsymbol{X[v_1] \leq X[u_1]}$ and $\boldsymbol{X[v_2] \geq X[u_2]}$. In this case we also say that edge f is $\boldsymbol{X - contained}$ in edge e. See Figure 2.*

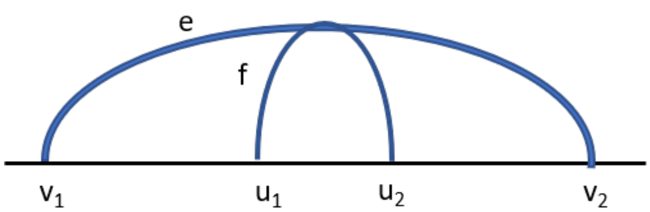

Figure 2: $X - contained$ edges

Definition 3.11 *Two edges $e = (v_1, v_2)$ and $f = (u_1, u_2)$ are $\boldsymbol{X - intersect}$ if $\boldsymbol{X[v_1] < X[u_1] \leq X[v_2] < X[u_2]}$. See Figure 3.*

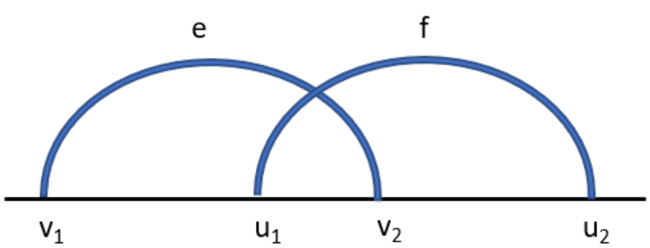

Figure 3: $X - intersect$ edges

Definition 3.12 *We define* **X-distance** *between two vertices $v^1 = (v_x^1, v_y^1, v_z^1)$ and $v^2 = (v_x^2, v_y^2, v_z^2)$ to be $|v_x^1 - v_x^2|$. The X-distance between two $X - disjoint$ edges $e_1 = (v_1^1, v_1^2)$ and $e_2 = (v_2^1, v_2^2)$ is equal to* $\min\{|X[v_1^1] - X[v_2^1]|, |X[v_1^1] - X[v_2^2]|, |X[v_1^2] - X[v_2^1]|, |X[v_1^2] - X[v_2^2]|\}$.

Definition 3.13 *Given a matching M, define $\mathbf{\Delta_M}$ to be the X-distance between the two farthest $X-disjoint$ edges in M, and denote e_{Δ_M} and e'_{Δ_M} to be the corresponding edges, that is, $\Delta_M = max\{X[e,e'] |\ e, e' \in M,\ e, e'$ are $X-disjoint\}$.*

Definition 3.14 *Define $\boldsymbol{\alpha}$ to be the minimal index of opt_i that provides a matching M with $Y[M] = Yopt$, $Z[M] = Zopt - 2\alpha\mathcal{Z}$ and $\Delta_M < \mathcal{Z}$. That is, $\alpha = min\{i \mid \exists M$ matching on G, $Y[M] = Yopt, Z[M] = Zopt - 2i\mathcal{Z}, \Delta_M < \mathcal{Z}\}$.*

Definition 3.15 *Define a **swap** between two edges $e_1 = (v_1, v_2)$ and $e_2 = (u_1, u_2)$ to be one of the options of creating two new edges f_1, f_2 or g_1, g_2, where $f_1 = (v_1, u_1)$ and $f_2 = (v_2, u_2)$, or $g_1 = (v_1, u_2)$ and $g_2 = (u_1, v_2)$.*

Definition 3.16 *Define $\boldsymbol{X - Preserving}$ swap to be a swap between two edges, such that $X-value$ is preserved or increased. That is, if f_1, f_2 are the edges created by an $X - Preserving$ swap on e_1, e_2, then $X[e_1, e_2] \leq X[f_1, f_2]$.*
$\boldsymbol{Y - Preserving}$ swap defined as a swap between two edges such that $Y - value$ is preserved or increased.
$\boldsymbol{Z - Preserving}$ swap defined as a swap between two edges such that $Z - value$ is preserved or increased.

Definition 3.17 *Define $\boldsymbol{X - Improving}$ **and** $\boldsymbol{Y - Preserving}$ swap to be a swap between two edges such that $X-value$ is increased and $Y-value$ is preserved or increased. That is, if f_1, f_2 are the edges created by an $X - Improving\ Y - Preserving$ swap on e_1, e_2, then $X[e_1, e_2] < X[f_1, f_2]$, $Y[e_1, e_2] = Y[f_1, f_2]$.*

Definition 3.18 *For U a set of vertices $u_1, \ldots, u_n$, arranged according to their X-coordinate, we define $\boldsymbol{X_{mid}}$ to be the X-value which satisfies $X[u_1] \leq X[u_2] \leq \cdots \leq X[u_{\frac{n}{2}}] \leq X_{mid} \leq X[u_{\frac{n}{2}+1}] \leq \cdots \leq u_n$*

Definition 3.19 *Define* **X containment matching** *as a matching M which satisfies that, for every two edges $e, e' \in M$, either e is $X-contained$ in e' or e' is $X-contained$ in e. See figure 4.*

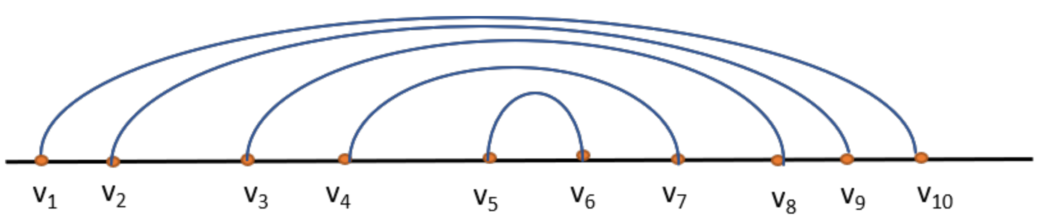

Figure 4: A matching which contains only $X - contained$ edges

Definition 3.20 *Given a matching M on a set of vertices V and $U \subseteq V$, define $\boldsymbol{M[U]}$ to contain all the edges in M such that both vertices are in U. That is,*
$M[U] = \{(u_1, u_2) | u_1, u_2 \in U, (u_1, u_2) \in M\}$.

Definition 3.21 *An edge* (v, u) $\boldsymbol{X - crosses}$ *a vertex* w *if* $X[v] \leq X[w] \leq X[u]$*, see Figure 5. Otherwise, define the edge to be* $\boldsymbol{X - one - sided}$ *with respect to* w*. If* (v, u) *is* $X - one - sided$ *with respect to* w*, then either* $X[v] \leq X[u] \leq X[w]$*, in this case we say that* u *is* $X - closer$ *to* w*, or* $X[w] \leq X[v] \leq X[u]$*, in this case we say that* v *is* $\boldsymbol{X - closer}$ *to* w*, see Figure 6.*

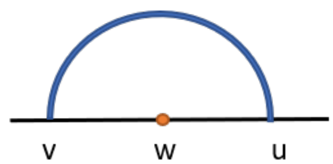

Figure 5: An edge which $X - crosses\ w$

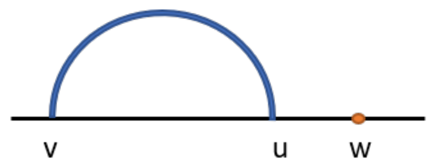

Figure 6: An edge which is $X - one - sided$ with respect to w, where u is $X - closer$ to w

Definition 3.22 *Denote* $\boldsymbol{S'(M)}$ *to be the set of vertices which are* $X - closer$ *to* X_{mid} *where every* $v \in S'(M)$*,* $X[v] < X_{mid}$*.*
Denote $\boldsymbol{S''(M)}$ *to be the set of vertices which are* $X - closer$ *to* X_{mid} *where every* $v \in S'(M)$*,* $X[v] > X_{mid}$*.*
Denote $\boldsymbol{S(M)} = S'(M) \cup S''(M)$*. See Figure 7.*

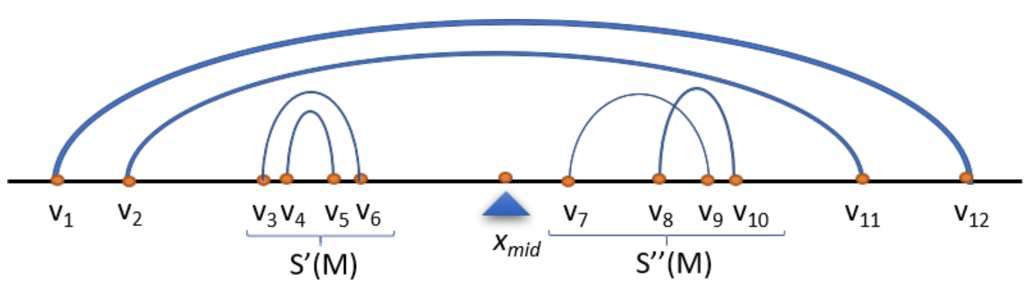

Figure 7: S'(M) and S"(M) of a matching M

4 Basic matching

In this section we consider matchings on one bank and two banks which contain t vertices on each one of the banks. This section introduces basic lemmas and algorithm for maximum matching. In addition this section contains basic lemmas and algorithm for match with respect to a point on one bank.

4.1 Basic properties

Lemma 4.1 *Every swap on two $X-disjoint$ edges $e = (v_1, v_2)$ and $e' = (u_1, u_2)$, with $X[v_2] < X[u_1]$ provides two edges f and f' with $X[f, f'] = X[e, e'] + 2(X[u_1] - X[v_2])$.*

Proof: Denote $a = X[v_1, v_2]$, $b = X[v_2, u_1]$, $c = X[u_1, v_2]$, see Figure 8. Since the edges are $X - disjoint$ $b > 0$. Using these notations, $X[(v_1, v_2), (u_1, u_2)] = a + c$, $X[(v_1, u_1), (v_2, u_2)] = a+2b+c$ and $X[(v_1, u_2), (v_2, u_1)] = a+2b+c$. Therefore, $X[f, f'] = X[e, e']+2(X[u_1]-X[v_2])$. ■

Corollary 4.2 *Every swap on two $X - disjoint$ edges e, e' provides two edges f and f' with $X[e, e'] < X[f, f']$.*

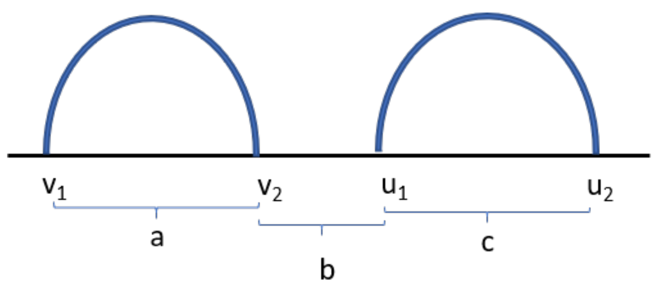

Figure 8: $X - disjoint$ edges

Lemma 4.3 *If edges $(v_1, v_2), (u_1, u_2)$ are $X - intersect$ such that $X[v_1] < X[u_1]$. Then a swap resulting $X - contained$ edges (u_1, v_2) and (v_1, u_2) is an $X - Preserving$ swap.*

Proof: Since $(v_1, v_2), (u_1, u_2)$ are $X - intersect$ then $X[v_1] \leq X[u_1] \leq X[v_2] \leq X[u_2]$, see Figure 9. The offered swap $(v_1, u_2), (u_1, v_2)$ yields two $X - contained$ edges, see Figure 10.
Denote $a = X[v_1, u_1]$, $b = X[u_1, v_2]$, $c = X[v_2, u_2]$. Using these notations, $X[(v_1, v_2), (u_1, u_2)] = a+2b+c$ and $X[(v_1, u_2), (u_1, v_2)] = a+2b+c$. Therefore we get $X[(v_1, v_2), (u_1, u_2)] = X[(v_1, u_2), (u_1, v_2)]$ and the resulting $X - contained$ edges is an $X - Preserving$ swap. ■

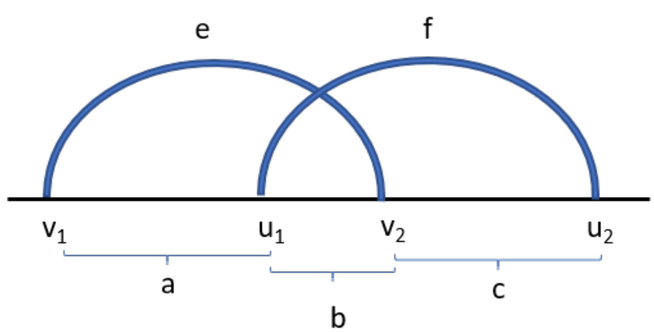

Figure 9: $X - intersect$ edges

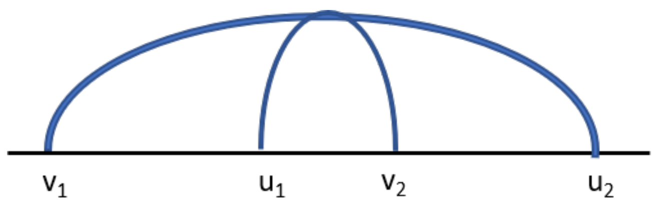

Figure 10: $X - contained$ edges

Lemma 4.4 *Every swap on two $X - disjoint$ edges $e = (v_1, v_2)$ and $e' = (u_1, u_2)$, with $X[v_2] < X[u_1]$ provides two edges which are not $X - disjoint$.*

Proof: Consider two $X - disjoint$ edges e and e', see Figure 8. There are two possible swaps for e and e'. One possible swap $f_1 = (v_1, u_1)$ and $f_1' = (v_2, u_2)$, which are $X - intersect$. Another possible swap $f_2 = (v_1, u_2)$ and $f_2' = (v_2, u_1)$ which are $X - contained$. Therefore, every swap provides edges which are not $X - disjoint$. ■

Corollary 4.5 *If a matching M contains two $X - disjoint$ edges then it is not $Xopt$.*

Proof: Suppose by a contradiction that M is $Xopt$ and contains two $X-disjoint$ edges, $e = (v_1, v_2)$ and $f = (u_1, u_2)$. According to Corollary 4.2, a swap between those edges yields $X[(v_1, u_1), (v_2, u_2)] > X[(v_1, v_2), (u_1, u_2)]$ or $X[(v_1, u_2), (v_2, u_1)] > X[(v_1, v_2), (u_1, u_2)]$, contradicting the assumption that M is $Xopt$. ■

Lemma 4.6 *Given a matching M on $v_1, \ldots, v_n$, for an even n, arranged according to their X-coordinate. If M does not contain any $X - disjoint$ edges, then for $1 \leq i, j \leq n$, for every edge $e = (v_i, v_j) \in M$, $X[v_i] \leq X_{mid} \leq X[v_j]$.*

Proof: Suppose by contradiction that M contains an edge $f_1 = (u_1, u_2)$ such that $X[u_1] \leq X[u_2] < X_{mid}$. There are $\frac{n}{2}$ vertices with $X - value \leq X_{mid}$, and $\frac{n}{2}$ vertices with $X - value \geq X_{mid}$. Therefore, by enumeration argument, M contains at least one edge $f_2 = (w_1, w_2)$ with $X_{mid} \leq X[w_1] \leq X[w_2]$. In this case, f_1 and f_2 are $X-disjoint$, see Figure 11, contradicting the assumption of the lemma.
Similarly, M does not contain an edge $e = (u_1, u_2)$ such that $X_{mid} \leq X[u_1] \leq X[u_2]$. ■

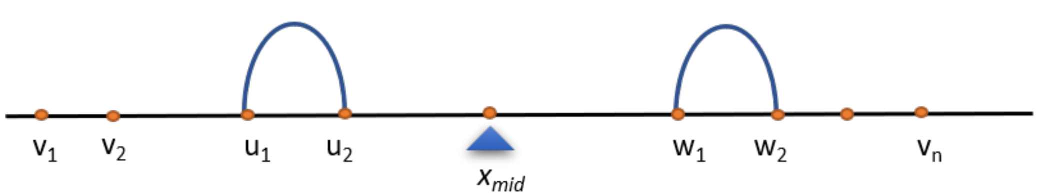

Figure 11: $X - disjoint$ edges

Lemma 4.7 *If there exist two edges e, f which $X - cross$ X_{mid}, then they are not $X - disjoint$.*

Proof: Let edge $e = (v, v')$ which $X - crosses$ X_{mid} then $X[v_1] \leq X_{mid} \leq X[v_2]$. Let edge $f = (u, u')$ which $X - crosses$ X_{mid}, then $X[u_1] \leq X_{mid} \leq X[u_2]$. Therefore, edges e and f are not $X - disjoint$. ∎

Corollary 4.8 *If two edges are $X - disjoint$, then both cannot be $X - cross$ X_{mid}.*

Lemma 4.9 *Given a matching M on $v_1, \ldots, v_n$, arranged according to their X-coordinate. If M is an $X - containment$ matching, then M contains edge (v_1, v_n).*

Proof: Suppose by contradiction that M is $X - containment$ matching that does not contain edge $e = (v_1, v_n)$. Therefore, in M there are two edges $f = (v_1, v_i), f' = (v_j, v_n)$. There are two options, either $v_1 < v_i < v_j < v_n$ or $v_1 < v_j < v_i < v_n$. Then edges f, f' are either $X - disjoint$ or $X - intersect$, contradicting the assumption that M is an $X - containment$ matchings. ∎

Lemma 4.10 *If M and M' are two $X - containment$ matchings on the same set of vertices $V = \{v_1, \ldots, v_n\}$, for an even n, arranged according to their X-coordinate, then $M \equiv M'$.*

Proof: We prove this lemma by induction on n.
For $n = 2$, the correctness of the lemma is trivial.
Assume the lemma is correct for all $k < n$ and prove it for n.
According to Lemma 4.9 M and M' contain edge (v_1, v_n).
According to the induction assumption, for $V' = V[v_2, \ldots, v_{n-1}]$ the $X - containment$ matchings $M[V'] \equiv M'[V']$.
Therefore, every $X - containment$ matching on the same set of vertices satisfies $M \equiv M'$. ∎

Lemma 4.11 *If a matching M on $v_1, \ldots, v_n$, arranged according to their X-coordinate, does not contain edge (v_1, v_n), then there is a matching M' with $X[M'] \geq X[M]$ such that M' contains edge (v_1, v_n).*

Proof: Since matching M does not contain edge (v_1, v_n), M contains two edges, $e = (v_1, v_i),$ *and* e'= (v_j, v_n), where e and e' are $X - intersect$ edges or $X - disjoint$ edges.
Denote $M' = M \setminus \{((v_1, v_i), (v_j, v_n))\} \cup \{(v_1, v_n), (v_i, v_j)\}$.
If edges e, e' are $X - disjoint$ edges, then according to lemma 4.1, $X[M'] > X[M]$.
If edges e, e' are $X - intersect$ edges, then according to Lemma 4.3, M' satisfies $X[M'] = X[M]$.
Therefore, in any case, $X[M'] \geq X[M]$. ∎

Lemma 4.12 *Given an $X - containment$ matching M on $V = \{v_1, \ldots, v_n\}$, for an even n, arranged according to their X-coordinate, M is $Xopt$.*

Proof: We prove this lemma by induction on n.
For $n = 2$, the correctness of the lemma is trivial.
Assume the lemma is correct for all even $k < n$ and prove it for n.
According to Lemma 4.9 M contains edge (v_1, v_n).
Next we prove that M is $Xopt$. Suppose by contradiction there exists a matching M' with $X[M'] > X[M]$. If M' contains edge (v_1, v_n) then let $M'' \equiv M'$. Otherwise, according to Lemma 4.11,

there exist a matching M'' with $X[M''] \geq X[M']$ such that M'' contains edge (v_1, v_n). Therefore, both M'' and M contain edge (v_1, v_n) and $X[M''] > X[M]$. In this case, $X[M''[v_2, \ldots, v_{n-1}]] > X[M[v_2, \ldots, v_{n-1}]]$.
Thus contradicting the induction hypothesis which states that $M[v_2, \ldots, v_{n-1}]$ is $Xopt$ since it is an $X - containment$ matching on $n-2$ edges. ∎

Lemma 4.13 *Given a matching M on $V = \{v_1, \ldots, v_n\}$, for an even n, arranged according to their X-coordinate. If M does not contain $X - disjoint$ edges, then M is $Xopt$.*

Proof: Since the matching does not contain $X - disjoint$ edges, then every two edges are in M, e and e', e is $X - contained$ in e' or e' is $X - contained$ in e or e and e' $X - intersect$.
If M is an $X - containment$ matching, then according to Lemma 4.12, M is $Xopt$.
Suppose M is not an $X - containment$ matching. Assume M contains edges e and e' which $X - intersect$. According to Lemma 4.3, there is $X - Preserving$ swap which creates f, f' with either f is $X - contained$ in f' or f' is $X - contained$ in f, such that $X[e, e'] = X[f, f']$. Denote $M' = M \setminus \{e, e'\} \cup \{f, f'\}$. This matching satisfies $X[M'] = X[M]$.
Continue in this manner until there are no $X - intersect$ edges. Denote the matching created by M''.
M'' does not contain $X - disjoint$ edges or $X - intersect$ edges and is therefore an $X - containment$ matching. According to Lemma 4.12, M'' is $Xopt$. Since $X[M] \equiv X[M'']$, therefore M is $Xopt$. ∎

Remark 4.14 *Let $V = \{v_1, \ldots, v_n\}$ be a set of vertices, arranged according to their X-coordinate and M a matching on V. For every $2 \leq i, j \leq n-1$ edge (v_i, v_j) is $X - contained$ in edge (v_1, v_n).*

Lemma 4.15 *For two sets of vertices $V = \{v_1, \ldots, v_{n_1}\}$ and $U = \{u_1, \ldots, u_{n_2}\}$, arranged according to theirs X-coordinate and M a matching on $V \cup U$, edges $(v_1, u_{n_2}), (u_1, v_{n_1})$ are not $X - disjoint$.*

Proof: Without loss of generality, assume that $X[v_1] \leq X[u_1]$.
Edge (v_1, u_{n_2}) satisfy $X[v_1] < X[u_1] \leq X[u_{n_2}]$, and therefore edges $(v_1, u_{n_2}), (u_1, v_{n_1})$ are not $X - disjoint$. See Figure 12.
∎

Lemma 4.16 *For two sets of vertices, $V = \{v_1, \ldots, v_{n_1}\}$ and $U = \{u_1, \ldots, u_{n_2}\}$, arranged according to their X-coordinate. For every $2 \leq i_1, j_1 \leq n_1 - 1$, $2 \leq i_2, j_2 \leq n_2 - 1$, edges $(v_1, u_{n_2}), (u_1, v_{n_1})$ are not $X - disjoint$ with any edge of the form (v_{i_1}, u_{j_2}) and (u_{i_2}, v_{j_1}).*

Proof: Without loss of generality, assume that $X[v_1] \leq X[u_1]$.
For every $2 \leq i_1, j_1 \leq n_1 - 1$ $2 \leq i_2, j_2 \leq n_2 - 1$, edge (v_{i_1}, u_{j_2}) is $X - contained$ in edge (v_1, u_{n_2}) since $X[v_{i_1}] > X[v_1]$, $X[u_{j_2}] < X[u_n]$. Therefore these edges are not $X - disjoint$.
Similarly, edge (u_{i_2}, v_{j_1}) is $X - contained$ in edge (u_1, v_{n_1}), since $X[u_{i_2}] > X[u_1]$, $X[v_{j_1}] < X[v_n]$. Therefore these edges are not $X - disjoint$. See Figure 12. ∎

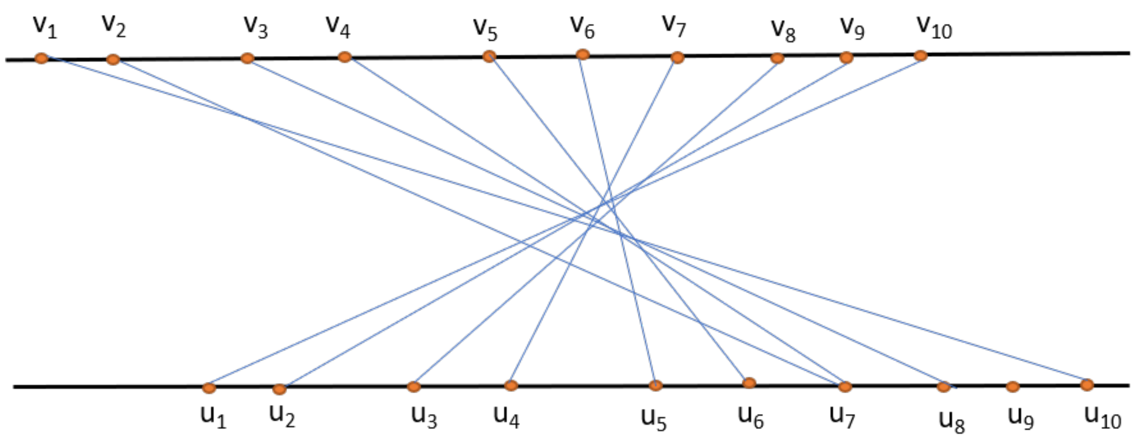

Figure 12: Two banks perfect matching

4.2 One bank matching

Consider the following two equivalent algorithms:

Algorithm 1 One Bank Match:

A matching procedure for a set of vertices on one bank

function ONE-BANK-MATCH

Input:

Vertices $v_1, \ldots, v_n$, with an even n, which reside on one bank and are ordered according to their X-coordinate.

Output:

A matching M on $v_1, \ldots, v_n$

begin

Initialize an empty match M

for $i = 1, \ldots, \frac{n}{2}$:

$M = M \cup \{(v_i, v_{n+1-i})\}$

end for

return M

end function

Lemma 4.17 *For a set of vertices $V = \{v_1, \ldots, v_n\}$, with an even n, arranged according to their X-coordinate on one bank, Algorithm One Bank Match Recursive provides a matching M on V that is an $X - containment$ matching.*

Proof: We prove this lemma by induction on n.
For $n = 2$, the correctness of the lemma is trivial.
Assume the lemma is correct for all $k < n$ and prove it for n.
The algorithm is recursive. In every iteration of the recursive, we first call One Bank Match Recursively algorithm on $v_2, \ldots, v_{n-1}$.

Algorithm 1.1 One Bank Match Recursive:

A matching procedure for a set of vertices on one bank

function One-Bank-Match-Recursive

Input:

Vertices $v_1, \dots, v_n$, with an even n, which reside on one bank and are ordered according to their X-coordinate.

Output:

A matching M on $v_1, \dots, v_n$

begin

  if $n == 0$:

    return

  end if

  return $[OneBankMatchRecursive(v_2, \dots, v_{n-1})] \cup (v_1, v_n)$

end function

According to the induction assumption, all edges in $M[v_2, \dots, v_{n-1}]$ are $X - contained$. Edge (v_1, v_n) contains all those edges. Therefore, all the edges in the matching are $X - contained$. ■

Corollary 4.18 *For a set of vertices $V = \{v_1, \dots, v_n\}$, with an even n, arranged according to their X-coordinate on one bank, according to Lemma 4.12, Algorithm One Bank Match provides a matching M on V that is $Xopt$.*

4.3 Two banks matching

Consider the following two equivalent algorithms

Algorithm 2 X-Diagonals:

A matching procedure for two sets of vertices on two different banks

function X-Diagonals

Input:

1. A bank where the vertices are ordered according to their X-coordinate $V = \{v_1, \dots, v_{k_1}\}$.
2. A bank where the vertices are ordered according to their X-coordinate $U = \{u_1, \dots, u_{k_2}\}$.

Assumptions $k_1 \geq k_2$, and k_1, k_2 are even.

Output:

A matching M_D

begin

  Initialize an empty match M_D

  for $i = 1, \dots, \frac{k_2}{2}$:

    $M_D = M_D \cup (v_i, u_{k_2+1-i})$

    $M_D = M_D \cup (v_{k_1+1-i}, u_i)$

  end for

  $M_D = M_D \cup One - Bank - Match(V[v_{\frac{k_2}{2}+1}, \dots, v_{k_1 - \frac{k_2}{2}}])$

  return M_D

end function

Algorithm 2.1 ***X*-Diagonals-Recursive:**

A matching procedure for two sets of vertices on two different banks

function X-Diagonals

Input:

1. A bank where the vertices are ordered according to their X-coordinate $V = v_1, \ldots, v_{k_1}$.
2. A bank where the vertices are ordered according to their X-coordinate $U = u_1, \ldots, u_{k_2}$.

Assumptions $k_1 \geq k_2$, and k_1, k_2 are even.

Output:

A matching M_D

begin

if $k_2 == 0$ *and* $k_1 == 0$: **return** $\emptyset$

end if

if $k_2 == 0$: **return** One-Bank-Match(V)

end if

return $X - Diagonals - Recursive(V[v_2, \ldots, v_{k_1-1}], U[u_2, \ldots, u_{k_2-1}]) \cup \{(v_1, u_{k_2}), (v_{k_1}, u_1)\}$

end function

Lemma 4.19 *Given two sets of vertices $V = \{v_1, \ldots, v_{k_1}\}$ and $U = \{u_1, \ldots, u_{k_2}\}$, for an even k_1, k_2, arranged according to their X-coordinate. Algorithm $X - Diagonals - Recursive$ returns a matching M_D which does not contain $X - disjoint$ edges.*

Proof: Without the loss of generality, we assume that $k_1 \geq k_2$ and we prove this lemma by induction on k_1.

For $k_1 = 0$, since $k_1 \geq k_2$ then $k_2 = 0$, the correctness of the lemma is trivial.

For $k_1 > 0$ *and* $k_2 = 0$, the algorithm returns the output of One-Bank-Match, denoted by M_D. According to Lemma 4.17, M is an $X - containment$ matching and therefore does not contain $X - disjoint$ edges.

For $k_1 > 0$ and $k_2 > 0$, assume the lemma is correct for all even $k < k_1$ and prove it for k_1.

Let M' be the matching returned by Algorithm $X - Diagonals - Recursive$ for $k = k_1 - 2$, such that $M' = X - Diagonals - Recursive[v_2, \ldots, v_{k_1-1}][u_2, \ldots, u_{k_2-1}]$. Let M_D be the matching returned by the algorithm, $M_D = M' \cup (v_1, u_{k_2}), (v_{k_1}, u_1)$.

M' is a matching on $k_1 - 2$ vertices in one bank and $k_2 - 2$ vertices on the other bank. Since $k_1 \geq k_2$ it is also true that $k_1 - 2 \geq k_2 - 2$ and according to the induction hypothesis M' does not contain $X - disjoint$ edges.

According to the induction assumption, in M' there are no $X - disjoint$ edges. According to Lemma 4.15, $(v_1, u_{k_2}), (v_{k_1}, u_1)$ are not $X - disjoint$. Furthermore, according to Lemma 4.16, (v_1, u_{k_2}) and (v_{k_1}, u_1) are not $X - disjoint$ with any of the edges in M'.

Therefore, the matching provided by the algorithm M_D does not contain $X - disjoint$ edges. ■

Theorem 4.20 *Given two sets of vertices $V = \{v_1, \ldots, v_{k_1}\}$ and $U = \{u_1, \ldots, u_{k_2}\}$, arranged according to their X-coordinate, Algorithm $X - Diagonals - Recursive$ returns a matching M_D which is $Xopt$.*

Proof: According to Lemma 4.19, M_D which is the output of Algorithm $X - Diagonals$, does not contain $X - disjoint$ edges. Therefore, according to Lemma 4.13, M_D is $Xopt$. ■

Lemma 4.21 *Given two sets of vertices $V = \{v_1, \ldots, v_{k_1}\}$ and $U = \{u_1, \ldots, u_{k_2}\}$, arranged according to their X-coordinate. For M_D which is the output of Algorithm $X-Diagonals-Recursive$, if $k_1 = k_2$ then every edge contains one vertex from V and one vertex from U.*

Proof: For the case where $k_1 = k_2$, the algorithm does not call One-Bank-Match. Therefore, for this case, in every iteration of Algorithm $X - Diagonals$, we match two vertices, one vertex from V one vertex from U. Therefore, every edge in M_D contains one vertex from V and one vertex from U. ■

4.4 Match with respect to a point

This subsection introduces lemmas for edges and swaps of edges with respect to a point.

Lemma 4.22 *Given e, e' two edges which are $X - intersect$ and let w be a point. Let f, f' be two edges which are the result from a swap on e, e', such that f, f' are $X - contained$. Then the swap does not change the number of edges which $X - cross$ w.*

Proof: Let $e = (v_1, v_2)$ and $e' = (u_1, u_2)$ be $X - intersect$ edges. Therefore, $X[v_1] \leq X[u_1] \leq X[v_2] \leq X[u_2]$, see Figure 13.
Since f, f' are a result from a swap, such that f and f' are $X - contained$, then $f = (v_1, u_2), f' = (u_1, v_2)$, see Figure 14.
If both e, e' $X - cross$ w, then $X[u_1] \leq X[w] \leq X[v_2]$, and both f, f' $X - cross$ w.
If e $X - crosses$ w and e' does not $X - cross$ w, then $X[v_1] \leq X[w] \leq X[u_1]$. In that case, f $X - crosses$ w and f' does not $X - cross$ w.
If neither of e and e' $X - cross$ w then either $w < X[v_1]$ or $X[u_2] < w$ and f, f' does not $X - cross$ w. Thus, the swap does not change the number of edges which $X - cross$ w. ■

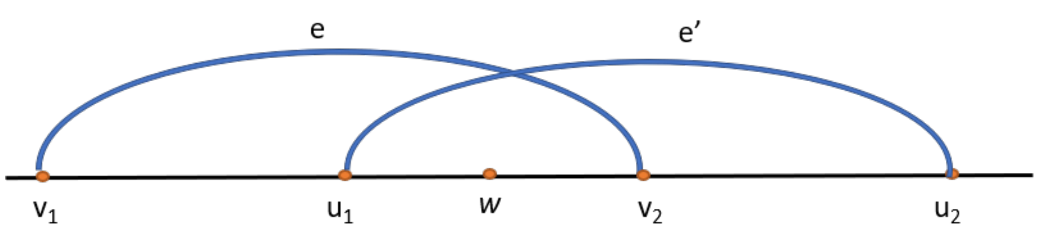

Figure 13: $X - intersect$ edges

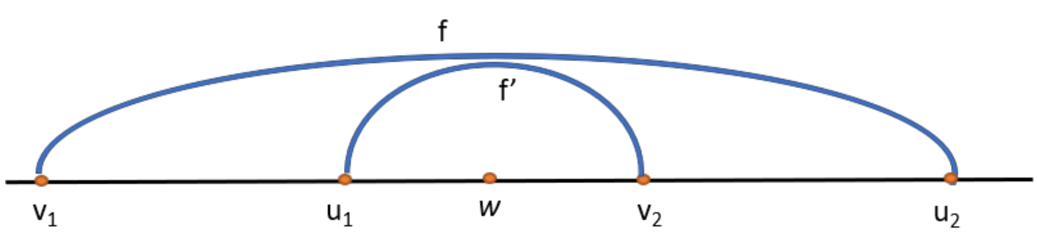

Figure 14: $X - contained$ edges

Corollary 4.23 *Given $e,\ e'$ two edges which are $X - one - sided$ edges with respect to a point w. If e and e' are not $X - disjoint$, then they are on the same side of w.*

Proof: Since $e,\ e'$ are not $X - disjoint$ then they are $X - intersect$ or $X - contained$. According to Lemma 4.22, then number of edges which $X - cross\ w$ is the same if $e,\ e'$ are $X - intersect$ or $X - contained$. Therefore, since $e,\ e'$ are $X - one - sided$ edges with respect to w then they are on the same side of w. ∎

Lemma 4.24 *Let M_0 and M_1 be two matchings which are $Xopt$ on the same set of vertices, and let w be a point. Then the number of edges in M_0 which $X - cross\ w$ is equal to the number of edges in M_1 which $X - cross\ w$.*

Proof: Since M_0 and M_1 are $Xopt$ then, according to Corollary 4.5, M_0 and M_1 do not contain $X - disjoint$ edges. Therefore, M_0 and M_1 contain only $X - intersect$ and $X - contained$ edges. Denote $e, e' \in M_0$ such that $e,\ e'$ are $X - intersect$ edges. According to Lemma 4.3, there is an $X - Preserving$ swap which creates $f,\ f'$ such that $X[e, e'] = X[f, f']$ and $f,\ f'$ are $X - contained$. Denote $\widehat{M_0} = M_0 \setminus \{e, e'\} \cup \{f, f'\}$, matching M_0 satisfies $X[M_0] = X[\widehat{M_0}]$.
According to Lemma 4.22, this swap does not change the number of edges which $X - cross\ w$ in M_0.
Continue in this manner until there are no $X - intersect$ edges. Denote the matching created by M_0^*. M_0^* is an $X - containment$ matching that contains the same number of edges which $X - cross\ w$ as M_0 and satisfies $X[M_0] = X[M_0^*]$.
Perform the same for M_1, resulting in M_1^*, such that $X[M_1^*] = X[M_1]$ and M_1^* is an $X - containment$ matching that contains the same number of edges that $X - cross\ w$ as M_1.
Since M_0^* and M_1^* are on the same set of vertices and both include only $X - contained$ edges, then according to Lemma 4.10, $M_0^* \equiv M_1^*$. Therefore, the number of edges which $X - cross\ w$ in M_0^* and in M_1^* is equal to the number of edges which $X - cross\ w$ in M_0 and equal to the number of edges which $X - cross\ w$ in M_1. ∎

Lemma 4.25 *Given M a matching which is $Xopt$, and let w be a point. All the edges in M that are $X - one - sided$ with respect to w are on the same side of w, such that this side is determined by comparing the $X - value$ of w in respect to X_{mid}.*

Proof: Since M is $Xopt$ then, according to Lemma 4.5, M does not contain $X - disjoint$ edges. Denote $e, e' \in M$ such that $e,\ e'$ are $X - one - sided$ with respect to w. According to Corollary 4.23, e and e' are on the same side of w. Therefore, all the edges in M that are $X - one - sided$ with respect to w are on the same side of w.
Assume, without loss of generality, that all edges e in M are $X - one - sided$ with respect to w satisfy $X[e] < X[w]$.
Let k_1 be the number of edges in M which are $X - one - sided$ with respect to w. There are $k_2 = \frac{n}{2} - k_1$ edges which $X - cross\ w$ with $\frac{k_2}{2}$ vertices with X-coordinate smaller than $X[w]$ and $\frac{k_2}{2}$ vertices with X-coordinate bigger than $X[w]$. Therefore $X_{mid} < X[w]$, see Figure 15.
Similarly, if all edges in M that are $X - one - sided$ with respect to w satisfy $X[e] > X[w]$, then $X_{mid} > X[w]$. ∎

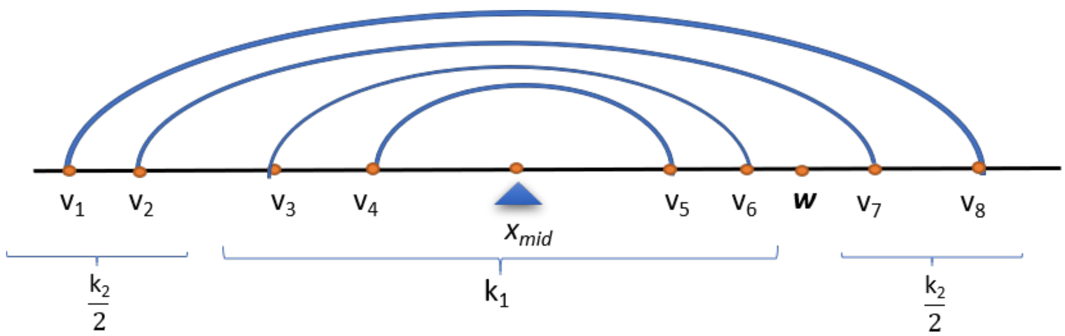

Figure 15: $Xopt$ matching with $X - one - sided$ edges with respect to w

Corollary 4.26 *Let M_0 and M_1 be two matchings which are $Xopt$ on the same set of vertices, and let w be a point. The number of edges which are $X - one - sided$ with respect to w is equal in M_0 and M_1, and they are all on the same side of w.*

Lemma 4.27 *Given two $X - intersect$ edges $e,\ e'$ and two $X - contained$ edges $f,\ f'$, on the set of vertices $V = \{v_1, v_2, v_3, v_4\}$, arranged according to their X-coordinate, and a point w. The set of vertices which are $X - closer$ to w is the same set in $e,\ e'$ and in $f,\ f'$.*

Proof: Without loss of generality, $e = (v_1, v_3)$ and $e' = (v_2, v_4)$, $f = (v_1, v_4)$ and $f' = (v_2, v_3)$.
If $X[w] < X[v_1]$, see Figure 16, then v_1, v_2 are the $X - closer$ vertices to w in $e,\ e'$ and in $f,\ f'$.
If $X[v_4] < X[w]$, see Figure 17, then v_3, v_4 are the $X - closer$ vertices to w in $e,\ e'$ and in $f,\ f'$.
If e is $X - one - sided$ with respect to w and e' $X - crosses$ w, see Figure 18, then for both $e,\ e'$ and $f,\ f'$, vertex v_3 is $X - closer$ to w. Otherwise, e' is $X - one - sided$ to w and e is $X - crosses$ w, then for both $e,\ e'$ and $f,\ f'$, vertex v_2 is $X - closer$ to w, see Figure 19 and Figure 20.
If both e and e' are $X - cross$ w, see Figure 21, then there are no $X - closer$ vertices to w for both $e,\ e'$ and $f,\ f'$. ■

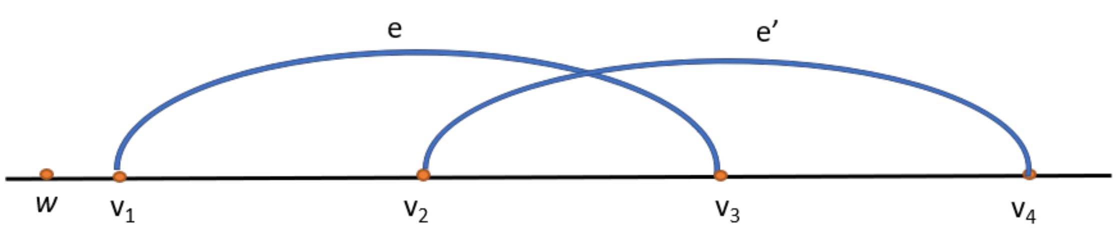

Figure 16: $X - intersect$ edges where $X[w] < X[v_1]$

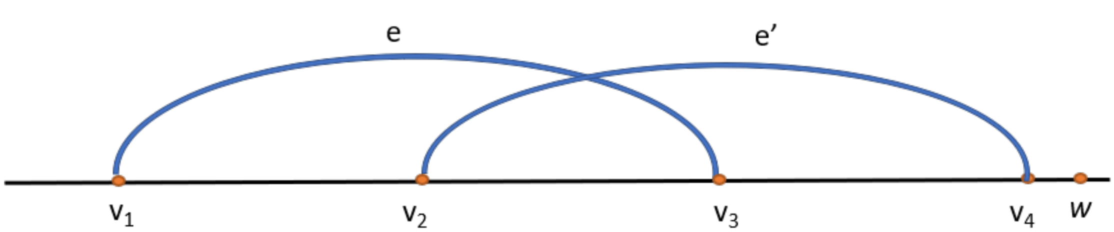

Figure 17: $X - intersect$ edges where $X[v_4] < X[w]$

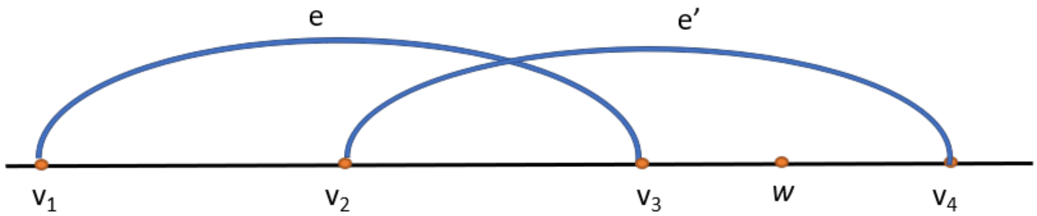

Figure 18: $X - intersect$ edges where e is $X - one - sided$ with respect to w and e' $X - crosses$ w

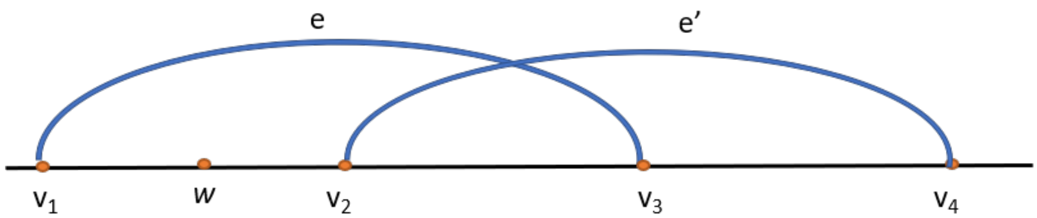

Figure 19: $X - intersect$ edges where e' is $X - one - sided$ to w and e is $X - crosses$ w

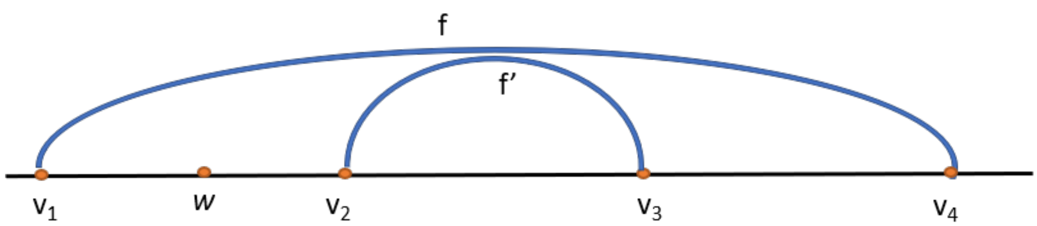

Figure 20: $X - contained$ edges with $X - one - sided$ edges with respect to w

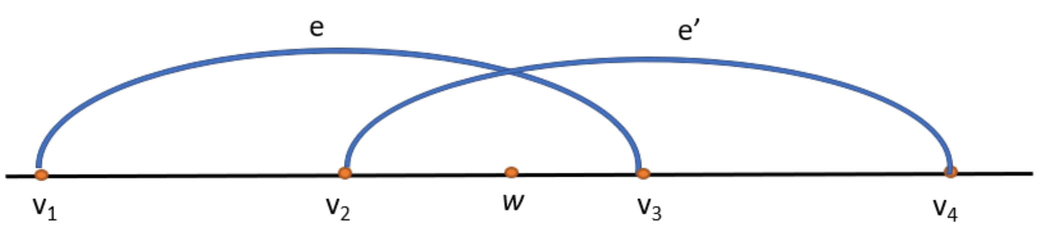

Figure 21: $X - intersect$ edges where both e and e' are $X - cross$ w

Lemma 4.28 *Let M_0 and M_1 be two matchings which are $Xopt$ on the same set of vertices, and let w be a point. Then the sets of points which are $X - closer$ to w in all the $X - one - sided$ edges is the same set in M_0 and in M_1.*

Proof: Since M_0 and M_1 are $Xopt$ then, according to Lemma 4.5, M_0 and M_1 does not contain $X - disjoint$ edges. Therefore, M_0 and M_1 contain only $X - intersect$ and $X - contained$ edges.

Denote $e, e' \in M_0$ such that $e,\ e'$ are $X-intersect$ edges. According to Lemma 4.3, there is an $X-Preserving$ swap which creates $f,\ f'$ such that $X[e,e'] = X[f,f']$ and $f,\ f'$ are $X-contained$. Denote $\widehat{M_0} = M_0 \setminus \{e,e'\} \cup \{f,f'\}$. Matching $\widehat{M_0}$ satisfies $X[M_0] = X[\widehat{M_0}]$.
According to Lemma 4.22, this swap does not change the number of edges which $X - cross\ w$ in M_0 and does not change the number of edges which are $X - one - sided$ to w in M_0. According to Lemma 4.27, the same vertices are $X - closer$ to w in $e,\ e'$ and in $f,\ f'$.
Continue until there are no $X - intersect$ edges. Denote the matching created by M_0^*. M_0^* is an $X - containment$ matching and M_0^* contains the same number of edges which $X - cross\ w$ as M_0. Furthermore, the same set of vertices of M_0 and M_0^* are $X - closer$ to w.
Perform the same for M_1, resulting in M_1^* such that $X[M_1^*] = X[M_1]$, with the same number of edges which $X - cross\ w$ in M_1 and M_1^*. Furthermore, the same set of vertices which are $X - closer$ to w in M_1 and in M_1^*.
Since M_0^* and M_1^* are on the same set of vertices and both include only $X - contained$ edges, then according to Lemma 4.10, $M_0^* \equiv M_1^*$. Therefore, the sets of points which are $X - closer$ to w in all the $X - one - sided$ edges is the same set in M_0^* and in M_1^*. since the set is equal in M_0^* and in M_0 and is equal in M_1^* and in M_1 the claim of Lemma is proven. ■

5 Four banks properties

This section introduces general proofs for matching on four banks graph.

Definition 5.1 *$G_4^{\triangledown} = (V,E)$ is a four banks complete graph such that $V = \{v_1^1, \ldots, v_{t_1}^1, v_1^2, \ldots, v_{t_2}^2, v_1^3, \ldots, v_{t_3}^3, v_1^4, \ldots, v_{t_4}^4\}$. The vertices on bank i for $i = 1,2,3,4$, are $v_1^i, \ldots, v_{t_i}^i$ and are ordered according to their X-coordinate.*

Definition 5.2 *Define $\boldsymbol{P_{Y=0}}$ to be the plane that contains all the banks with $Y - value = 0$. For four banks graph the plane contains $B_{(0,0)}$ and $B_{(0,\mathcal{Z})}$*

Definition 5.3 *Define $\boldsymbol{P_{Y=\mathcal{Y}}}$ to be the plane that contains all the banks with $Y - value = \mathcal{Y}$. For four banks graph the plane contains $B_{(\mathcal{Y},0)}$ and $B_{(\mathcal{Y},\mathcal{Z})}$*

Definition 5.4 *Define $\boldsymbol{P_{Z=0}}$ to be the plane that contains all the banks with $Z - value = 0$. For four banks graph the plane contains $B_{(0,0)}$ and $B_{(\mathcal{Y},0)}$*

Definition 5.5 *Define $\boldsymbol{P_{Z=\mathcal{Z}}}$ to be the plane that contains all the banks with $Z - value = \mathcal{Z}$. For four banks graph the plane contains $B_{(0,\mathcal{Z})}$ and $B_{(\mathcal{Y},\mathcal{Z})}$*

Definition 5.6 *Define edge e to be* **Diagonal Edge** *if one endpoint of e is in $B_{(0,0)}$ and the other is in $B_{(\mathcal{Y},\mathcal{Z})}$ or if one endpoint of e is in $B_{(\mathcal{Y},0)}$ and the other is in $B_{(0,\mathcal{Z})}$.*

Definition 5.7 *Define edges whose both endpoints are on the same $Y - Plane$ or whose both endpoints are on the same $Z - Plane$ to be* **One Plane Edges**.

Definition 5.8 *Define edges whose both endpoints are on the same bank to be* **One Bank Edges**.

In the following we prove important properties regarding four banks.

Lemma 5.9 *Given $G_4^{\triangledown}$ a four unbalanced graph. Let M be a matching on the graph containing two edges e, e', such that e is a Diagonal Edge and e' is a One Plane Edge. Then there exists a swap on e, e' resulted in f, f', which yields $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$.*

Proof: Since $G_4^{\triangledown}$ is a four banks graph, then $e = (v_1, v_2)$ and $e' = (u_1, u_2)$ satisfy that one endpoint of e and one endpoint of e' are on the same bank. Without the loss of generality, suppose these endpoints are v_1 and u_1, see Figure 22.
Swap $f = (u_1, v_2)$ and $f' = (v_1, u_2)$ satisfy that f is a Diagonal Edge on the same Diagonal as e and f' is a One Plane Edge on the same plane as e'. Therefore, $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$, see Figure 23. ∎

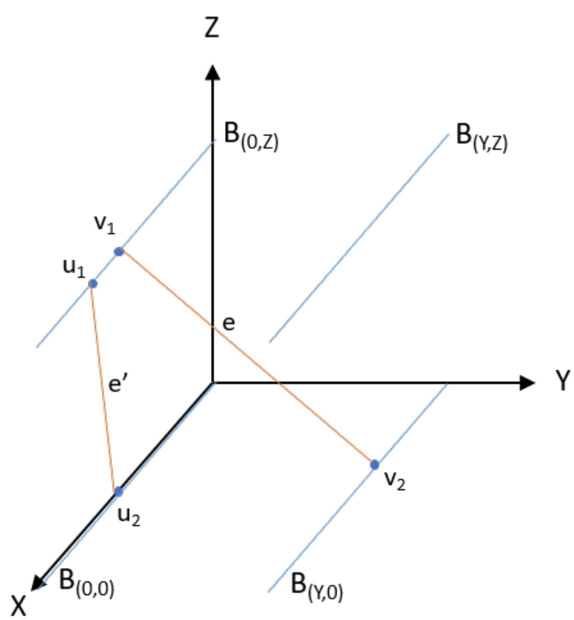

Figure 22: A Diagonal Edge and a One Plane Edge

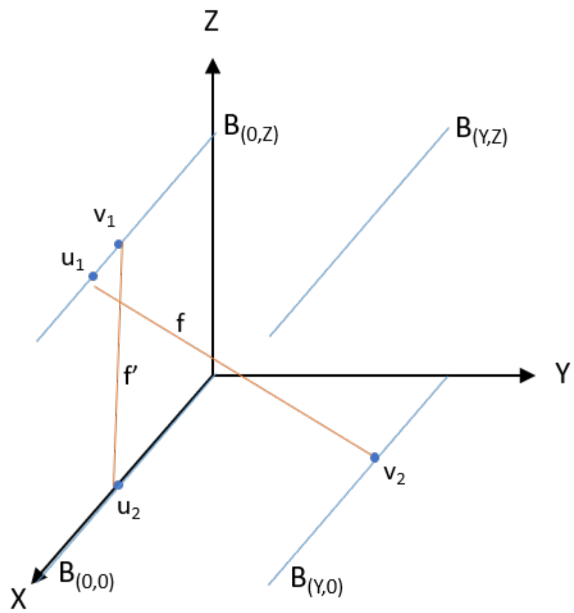

Figure 23: A Diagonal Edge and a One Plane Edge after a swap

Lemma 5.10 *Given $G_4^{\bigtriangledown}$ a four unbalanced graph. Let M be a matching on the graph containing two edges e, e', such that e is a Diagonal Edge and e' is a One Bank Edge. Then there exists a swap on e, e' resulted in f, f', which yields $Y[e, e'] = Y[f, f']$ and $Z[e, e'] = Z[f, f']$.*

Proof: Let $e = (v_1, v_2)$ be a Diagonal Edge and $e' = (u_1, u_2)$ be a One Bank Edge. Then there are two possibilities. One possibility is for the case where the endpoints of e are on the same bank as one of the endpoints of e, see Figure 24. Without the loss of generality, a swap $f = (u_1, v_2)$ and $f' = (v_1, u_2)$ satisfies that f is a Diagonal Edge on the same Diagonal as e and f' is a One Bank Edge on the same bank as e'. Therefore, $Y[e, e'] = Y[f, f']$ and $Z[e, e'] = Z[f, f']$, see Figure 25. Second possibility is for the case where the endpoints of e' are not on the same bank as one of the endpoints of e, see Figure 26. Then any swap to f, f' satisfy that one of the edges is a One Plane Edge on the Y-Plane and the other is a One Plane Edge on the Z-Plane. Therefore, $Y[e, e'] = \mathcal{Y} = Y[f, f']$ and $Z[e, e'] = \mathcal{Z} = Z[f, f']$. See Figure 27. ∎

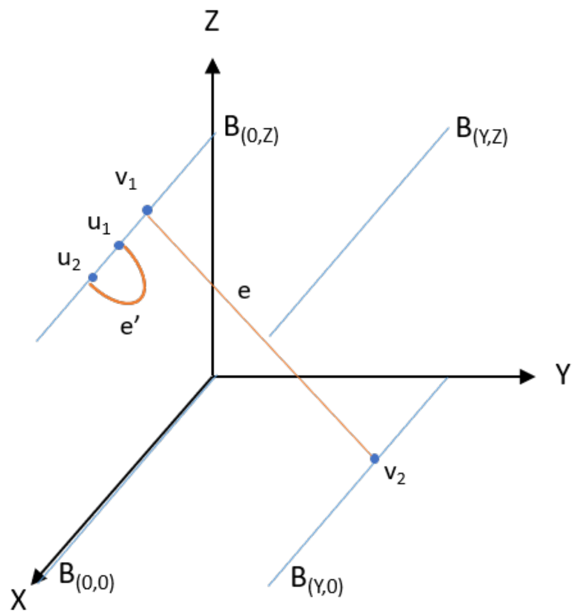

Figure 24: A Diagonal Edge and a One Bank Edge

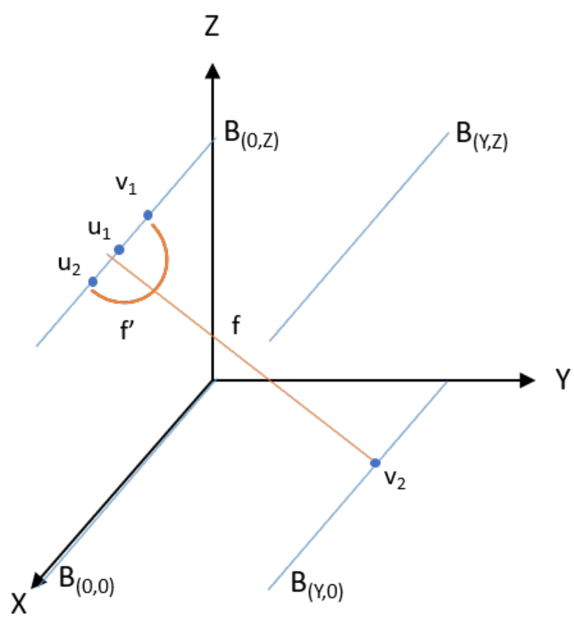

Figure 25: A Diagonal Edge and a One Bank Edge after a swap

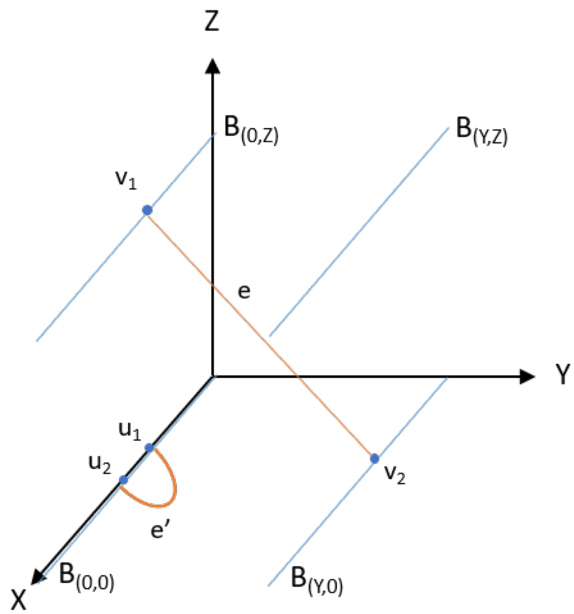

Figure 26: A Diagonal Edge and a One Bank Edge

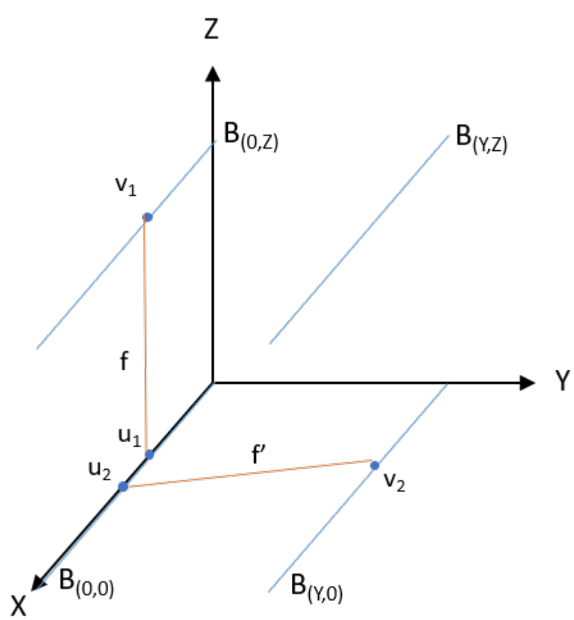

Figure 27: A Diagonal Edge and a One Bank Edge after a swap

Lemma 5.11 *Given $G_4^{\triangledown}$ a four unbalanced graph. If e, e' satisfy that e is a One Plane Edge and e' is a One Bank Edge and one endpoint of e is on the same bank as the endpoints of e', then any swap resulted in f, f' yields $Y[e, e'] = Y[f, f']$ and $Z[e, e'] = Z[f, f']$.*

Proof: Let $e = (v_1, v_2)$ and $e' = (u_1, u_2)$, such that, without the loss of generality, v_1 is on the same bank as u_1 and u_2, see Figure 28. Swap $f = (u_1, v_2)$ and $f' = (v_1, u_2)$ satisfy that f is a One Plane Edge on the same plane as e and f' is a One Bank Edge on the same bank as e'. Therefore, $Y[e, e'] = Y[f, f']$ and $Z[e, e'] = Z[f, f']$, see Figure 29. ∎

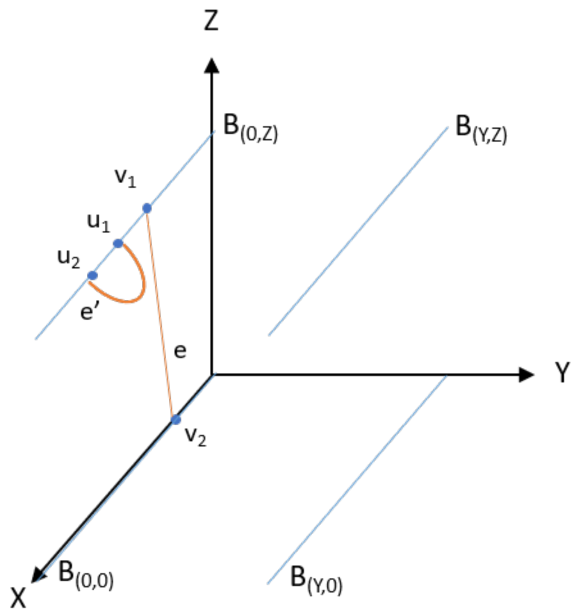

Figure 28: A One Plane Edge and a One Bank Edge

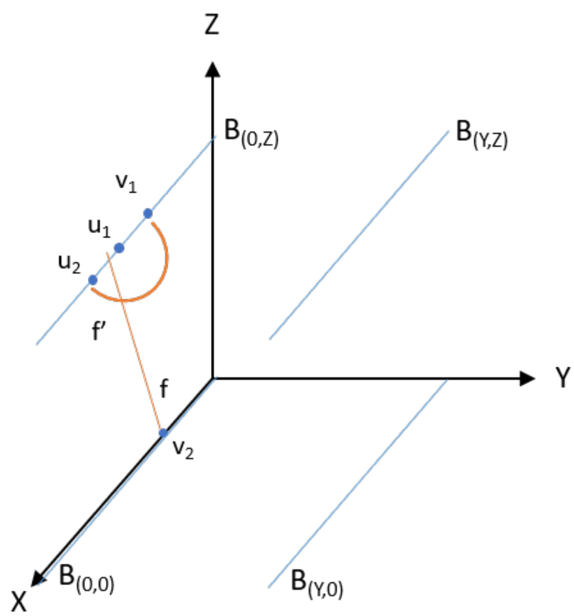

Figure 29: A One Plane Edge and a One Bank Edge after a swap

Lemma 5.12 *Given $G_4^{\triangledown}$ a four unbalanced graph. If e, e' satisfy that e and e' are both One*

Plane Edges on the same plane, then any swap resulted in f, f' yields $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$.

Proof: Let $e = (v_1, v_2)$ and $e' = (u_1, u_2)$. Without the loss of generality, since e, e' are on the same plane, v_1 is on the same bank as u_1 and v_2 is on the same bank as u_2, see Figure 30. A swap $f = (u_1, v_2)$ and $f' = (v_1, u_2)$ satisfy that f and f' are both One Plane Edges on the same plane as e, e', see Figure 31. Therefore, $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$. ■

Lemma 5.13 *Given $G_4^{\triangledown}$ a four unbalanced graph. If e, e' are both One Plane Edges then any swap resulted in f, f' yields $Y[e,e'] \leq Y[f,f']$ and $Z[e,e'] \leq Z[f,f']$.*

Proof: Let $e = (v_1, v_2)$ and $e' = (u_1, u_2)$.
For case e and e' are both on the same plane, follows from Lemma 5.12.
For case e and e' are on different Y-planes, see Figure 32. Then swap $f = (v_1, u_2)$ and $f' = (v_2, u_1)$ yields $Y[e,e'] = Y[f,f']$, $Z[e,e'] < Z[f,f']$, see Figure 33.
For case e and e' are on different Z-planes, then swap $f = (v_1, u_2)$ and $f' = (v_2, u_1)$ yields $Y[e,e'] < Y[f,f']$, $Z[e,e'] = Z[f,f']$.
For case e is on Y-plane and e' is on Z-plane, see Figure 34. Then swap $f = (v_1, u_2)$ and $f' = (v_2, u_1)$ yields $Y[e,e'] = Y[f,f']$, $Z[e,e'] = Z[f,f']$, see Figure 35.
■

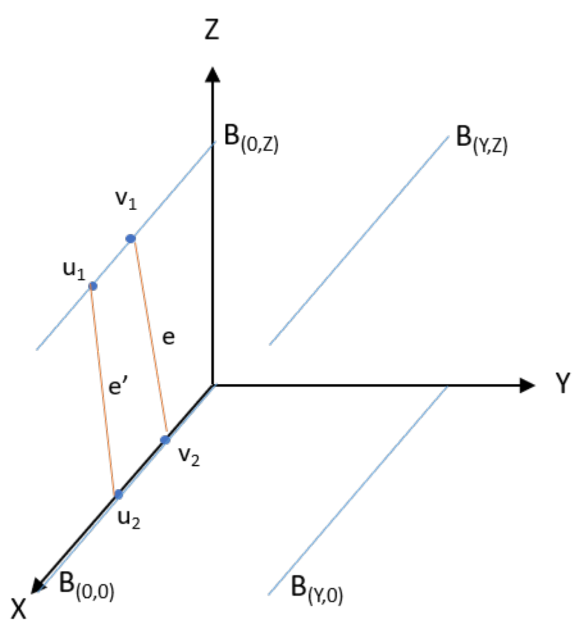

Figure 30: Two One Plane Edges on the same plane

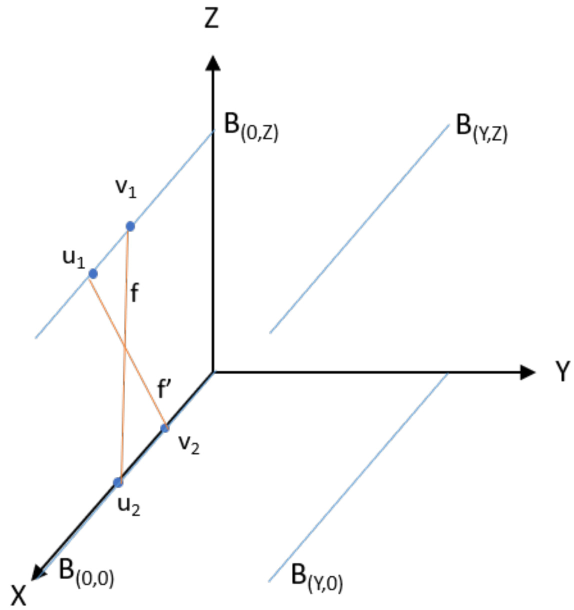

Figure 31: Two One Plane Edges on the same plane after a swap

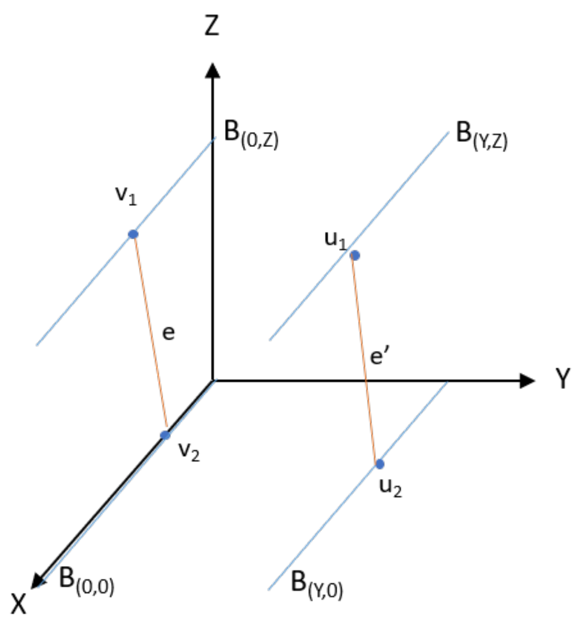

Figure 32: Two One Plane Edges on different Y planes

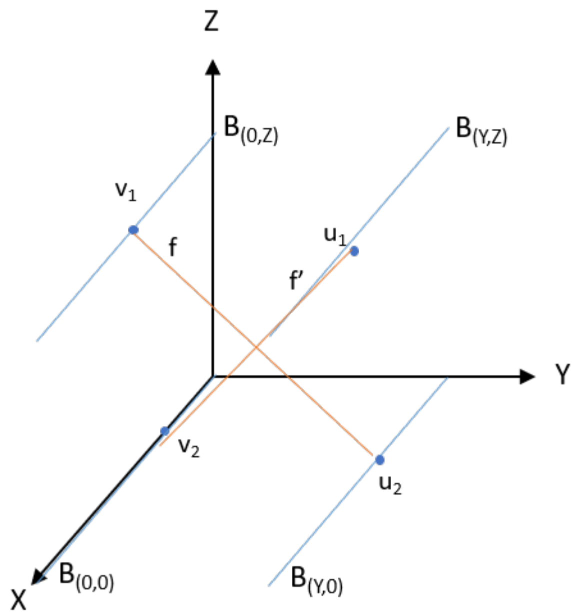

Figure 33: Two One Plane Edges on different Y planes after a swap

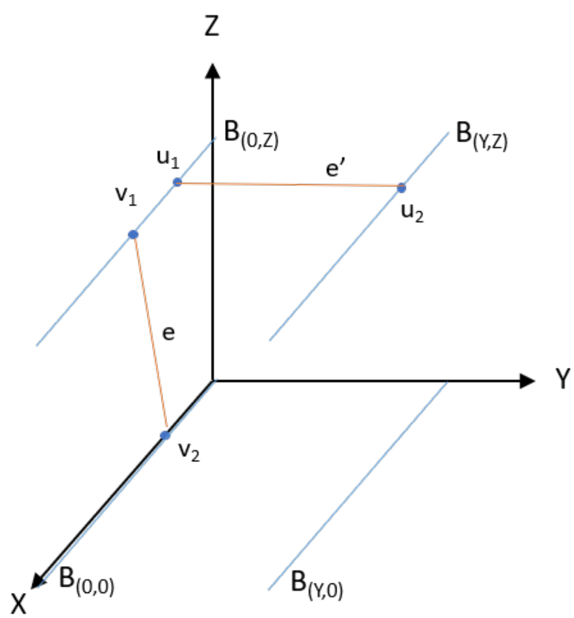

Figure 34: Two One Plane Edges one on Y plane and the other on Z plane

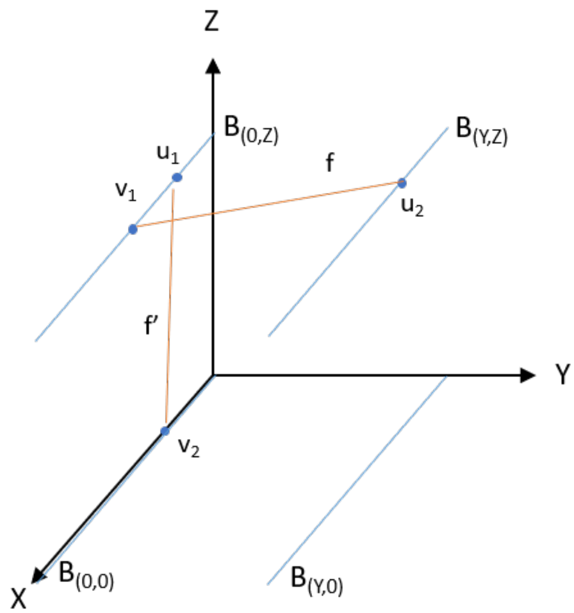

Figure 35: Two One Plane Edges one on Y plane and the other on Z plane after a swap

Lemma 5.14 *Given $G_4^{\triangledown}$ a four unbalanced graph. If e, e' are both One Bank Edges then there exists a swap resulted in f, f' which yields $Y[e,e'] \leq Y[f,f']$ and $Z[e,e'] \leq Z[f,f']$.*

Proof: Let $e = (v_1, v_2)$ and $e' = (u_1, u_2)$.
For case e, e' are on the same bank, see Figure 36. Then a swap $f = (v_1, u_1)$ and $f' = (v_2, u_2)$ yields $Y[e,e'] = Y[f,f']$, $Z[e,e'] = Z[f,f']$, see Figure 37.
For case e, e' are on different banks, see Figure 38. Then a swap $f = (v_1, u_1)$ and $f' = (v_2, u_2)$ and the swap is $Y - Improving$ or $Z - Improving$ or both, and therefore, $Y[e,e'] \leq Y[f,f']$ and $Z[e,e'] \leq Z[f,f']$, see Figure 39. ∎

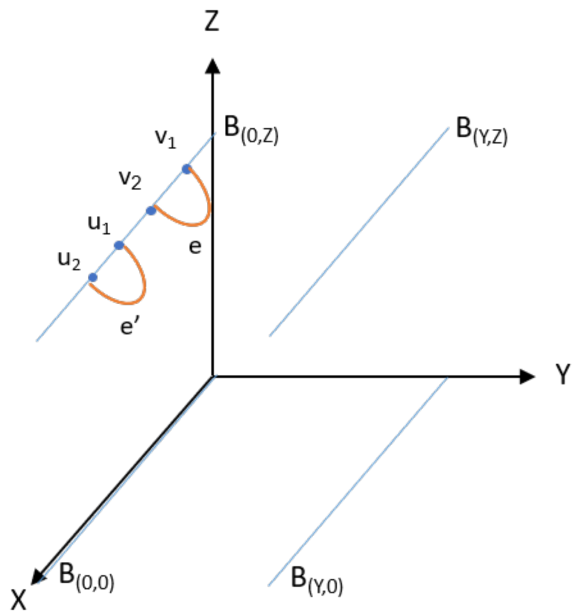

Figure 36: Two One Bank Edges on the same bank

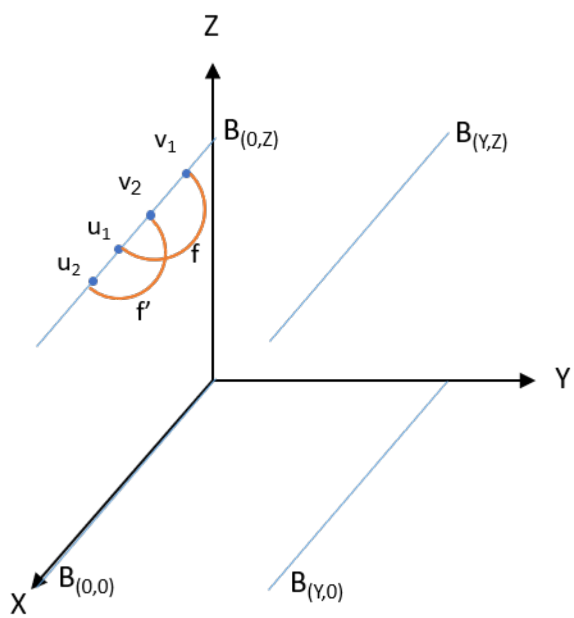

Figure 37: Two One Bank Edges on the same bank after a swap

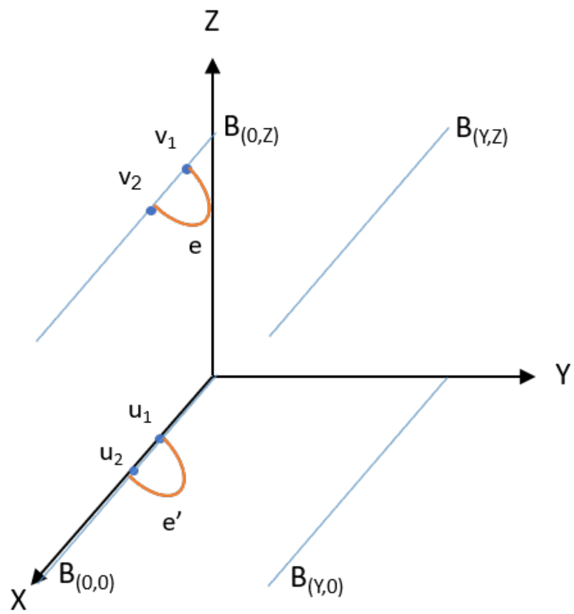

Figure 38: Two One Bank Edges on different banks

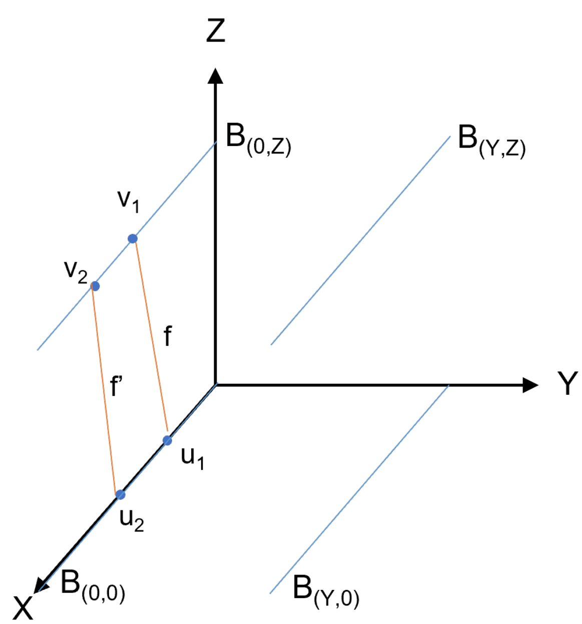

Figure 39: Two One Bank Edges on different banks after a swap

6 Four Balanced Banks Matching

In this section we consider matchings on a graph which contains t vertices on each one of the four banks. In the beginning of this section we introduce basic properties concerning this case, followed by an algorithms which finds a maximum matching. The section continues by a set of definitions regarding the algorithm. Finally we prove the correctness of the algorithm.

In all the algorithms we assume that $\mathcal{Y} \geq \mathcal{Z}$, otherwise, change the roles of Y axis and Z axis.

Definition 6.1 *$G_4 = (V, E)$ is a four balanced graph with t vertices, therefore, $V = \{v_1^1, \ldots, v_t^1, v_1^2, \ldots, v_t^2, v_1^3, \ldots, v_t^3, v$*

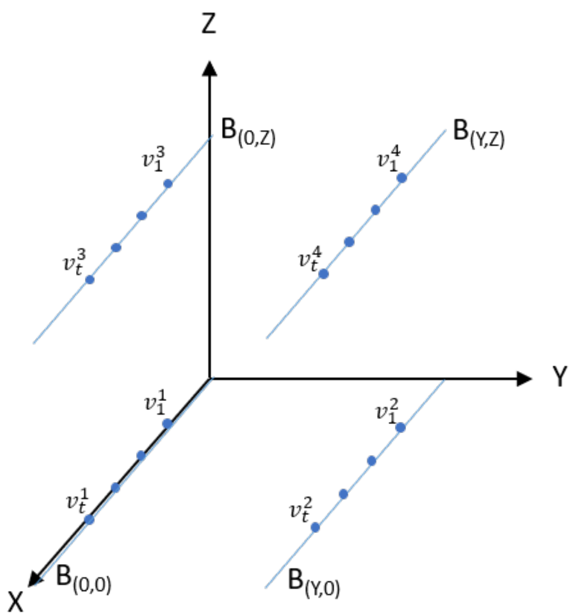

Figure 40: Four Balanced Banks

Lemma 6.2 *Given G_4 a four balanced graph with t vertices on each bank. A matching M on G_4 is $Yopt$ if $Y[M] = 2\mathcal{Y}t$. Furthermore, if M is $Yopt$, then M does not contain any edge e with $Y[e] = 0$.*

Proof: In G_4, the maximal $Y - value$ for each edge is $\mathcal{Y}$. In G_4, there are $4t$ vertices. Therefore, a full matching contains 2t edges. For M a $Yopt$ matching, which contains $2t$ edges with the maximal $Y - value = \mathcal{Y}$, then $Y[M] = 2\mathcal{Y}t$. therefore, for every edge $e \in M$, $Y[e] = \mathcal{Y}$ and M does not contain edge e with $Y[e] = 0$. ■

Notation 6.3 *For every $M_0^o \in OPT_0$, according to Definition 3.8, and Lemma 6.2, $Y[M_0^o] = Yopt = 2t\mathcal{Y}$, and $Z[M_0^o] = Zopt = 2t\mathcal{Z}$.*
Therefore, all the edges in OPT_0 are on the diagonals, so M_0^o is composed from two matchings $\boldsymbol{M_0^{o\prime}}$ containing only edges with one vertex in $B_{(0,0)}$ and the other vertex in $B_{(\mathcal{Y},\mathcal{Z})}$. And $\boldsymbol{M_0^{o\prime\prime}}$ containing only edges with one vertex in $B_{(\mathcal{Y},0)}$ and the other vertex in $B_{(0,\mathcal{Z})}$.

Lemma 6.4 *Given G_4 a four balanced graph with t vertices on each bank. A matching M on G_4 is $Zopt$ if $Z[M] = 2\mathcal{Z}t$. Furthermore, if M is $Zopt$, then M does not contain any edge e with $Z[e] = 0$.*

Proof: In G_4, the maximal $Z - value$ for each edge is $\mathcal{Z}$. In G_4, there are $4t$ vertices. Therefore a full matching contains 2t edges. For M a $Zopt$ matching, which contains $2t$ edges with the maximal Z-value= $\mathcal{Z}$, then $Z[M] = 2\mathcal{Z}t$. therefore, for every edge $e \in M$, $Z[e] = \mathcal{Z}$ and M does not contain edge with $Z - value = 0$. ■

Corollary 6.5 *Given G_4 a four balanced graph with t vertices on each bank and let M be a matching on G_4. The number of edges with $Z - value = 0$ in M is even.*

Lemma 6.6 *Given G_4 a four balanced graph with t vertices on each bank and let M be a matching on G_4. If M contains edge e with $Y[e] = 0$ on a Y-plane, then it contains another edge e' such that $Y[e'] = 0$ on the other Y plane.*

Proof: Since in G_4 there are two banks on each plane, there are $2t$ vertices on each Y plane with $P_{Y=0}, P_{Y=\mathcal{Y}}$.
For each edge f, if $Y[f] = \mathcal{Y}$ then f connects vertices on two different planes and if $Y[f] = 0$ then f connects vertices on the same plane.
Given an edge e with $Y[e] = 0$, without loss of generality, $e \in P_{Y=0}$, then there are at most $2t - 2$ vertices on $P_{Y=0}$ which are matched to vertices on $P_{Y=\mathcal{Y}}$.
Since there are $2t$ vertices on each plane, there are at least two vertices on plane $P_{Y=\mathcal{Y}}$, which are matched together. Therefore, there is another edge e' such that $Y[e'] = 0$ on the other Y plane. ■

Lemma 6.7 *Given G_4 a four balanced graph. If $\mathcal{Y} > \mathcal{Z}$ then every maximum matching $M \in OPT$ is $Yopt$.*

Proof: Suppose by contradiction that M is a maximum matching on a balanced graph which is not $Yopt$. Then according to Lemma 6.2, M contains at least one edge e such that $Y[e] = 0$. Furthermore, according to Lemma 6.6, M contains another edge e' such that $Y[e'] = 0$.
Let f_1, f_1' and f_2, f_2' be the two possible sets of edges created by $Y - Improving$ swap on e and e'. At least one pair of edges f_1, f_1' or f_2, f_2' does not contain $X - disjoint$ edges, denote those edges by $f,\ f'$.
Let $M' = M \setminus \{e, e'\} \cup \{f, f'\}$. For matching M', since edges $f,\ f'$ are crossing Y-plane, $Y[M'] = Y[M] + 2Y$, $Z[M'] \geq Z[M] - 2Z$. Since f, f' are not $X - disjoint$, then according to Lemma 4.2, $X[M'] \geq X[M]$. Since $\mathcal{Y} > \mathcal{Z}$, then $opt = val(M) < val(M')$, contradicting the assumption that M is a maximum matching. ■

Lemma 6.8 *Given G_4 a four balanced graph. There exists $0 \leq i^* \leq p$ such that $opt = opt_{i^*}$.*

Proof: According to Lemma 6.7, every $M \in OPT$ on G_4 is $Yopt$. Since G_4 is a balanced graph, then the number of edges with $Z - value = 0$ in M is even, denote this number by $2i^*$. Giving that, $Y[M] = Yopt$, $Z[M] = Zopt - 2i^*\mathcal{Z}$. Therefore, $M \in OPT_{i^*}$ and $val(M) = opt_{i^*} = opt$. ■

Algorithm 3 $XDZ4$**:**

Finds a maximum perfect matching.

function XDZ4()

Input:

A graph $G = (V, E)$ with four balanced banks.

On each bank i there are t vertices $v_1^i, \ldots, v_t^i$ ordered according to their X-coordinate

Assumptions $\mathcal{Y} > \mathcal{Z}$.

Output:

A maximum perfect matching M

begin

Phase 1:

Initialize an empty match M

$M = M \cup X - Diagonals(B_{(0,0)}, B_{(\mathcal{Y},\mathcal{Z})})$

$M = M \cup X - Diagonals(B_{(\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$

Phase 2:

while M contains $X - disjoint$ edges :

Find $e_{\Delta_M}, e'_{\Delta_M}$ (see Definition 3.13)

while $\Delta_M > \mathcal{Z}$:

Let f, f' be the edges created by $Y - Preserving$ and $X - Improving$ swap on $e_{\Delta_M}, e'_{\Delta_M}$

$M = M \setminus \{e_{\Delta_M}, e'_{\Delta_M}\} \cup \{f, f'\}$

Find $e_{\Delta_M}, e'_{\Delta_M}$ (see Definition 3.13)

end while

end while

return M

end function

Example 6.9 *Consider a graph $G = (V, E)$ with four vertices on each bank, see Figure 41. Assume that the order of all vertices in V is demonstrated in Figure 42.*
For example, $X[v_1] < X[v_2] < X[w_1]$. Also assume that $\mathcal{Y} > \mathcal{Z}$.
Phase 1 of Algorithm $XDZ4$ yields an M which is a perfect matching satisfying that all vertices of bank $B_{(0,0)}$ are matched to all vertices of bank $B_{(\mathcal{Y},\mathcal{Z})}$, and all vertices of bank $B_{(\mathcal{Y},0)}$ are matched to all vertices of bank $B_{(0,\mathcal{Z})}$, each according to $X - Diagonal$ Algorithm, see Figure 43 and Figure 44.
As can be seen in Figure 45, the matching contains $X - disjoint$ edges $e = (s_2, v_3)$ and $e' = (u_1, w_4)$. If we assume that $X[u_1] - X[v_3] > \mathcal{Z}$, a $Y - Preserving$ and $X - Improving$ swap on e, e' yields $f = (v_3, u_1)$ and $f' = (s_2, w_4)$.
Thus, after one iteration of phase 2, there are no more $X - disjoint$ edges and the matching M, which is the result of the algorithm is a perfect matching. See Figure 46 and Figure 47, which demonstrate this matching.

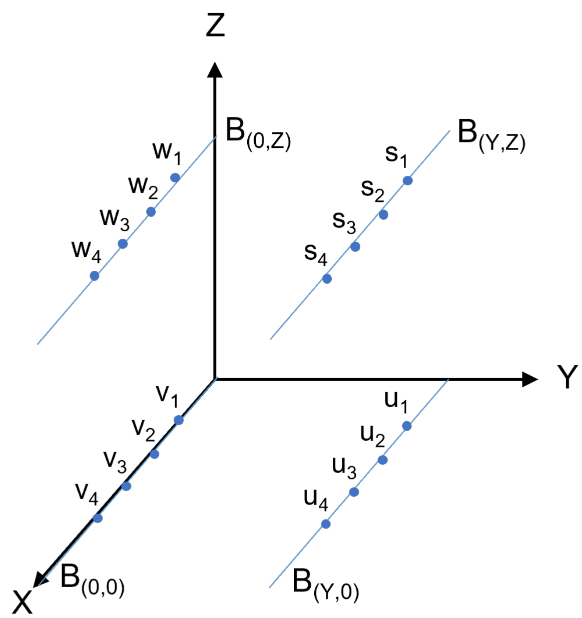

Figure 41: An Example of a graph G

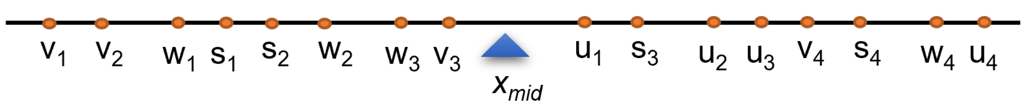

Figure 42: An Example of a graph G, a view on the X-coordinate

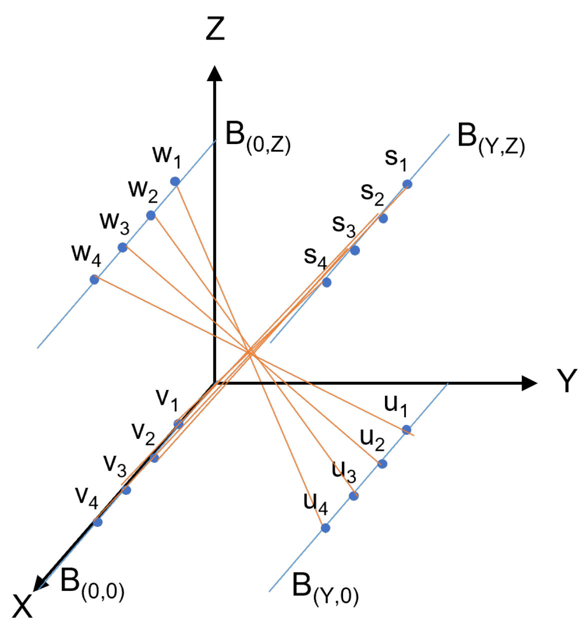

Figure 43: G after phase 1 of $XDZ4$ Algorithm

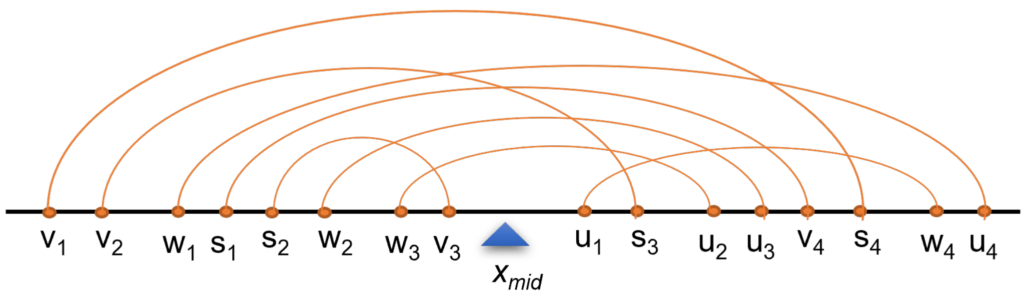

Figure 44: G after phase 1 of $XDZ4$ Algorithm, a view on the X-coordinate

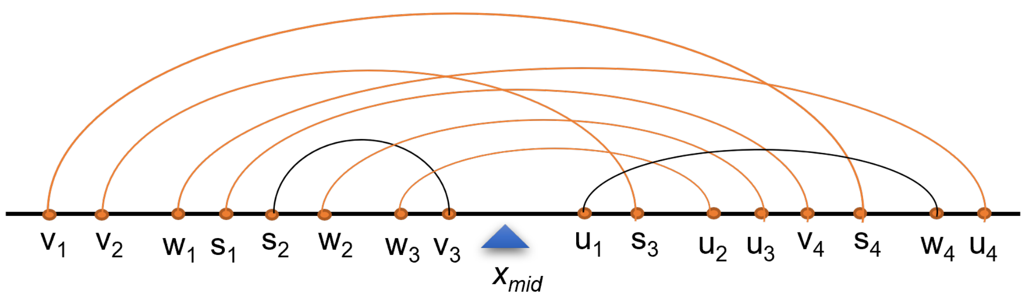

Figure 45: X-disjoint edges on G

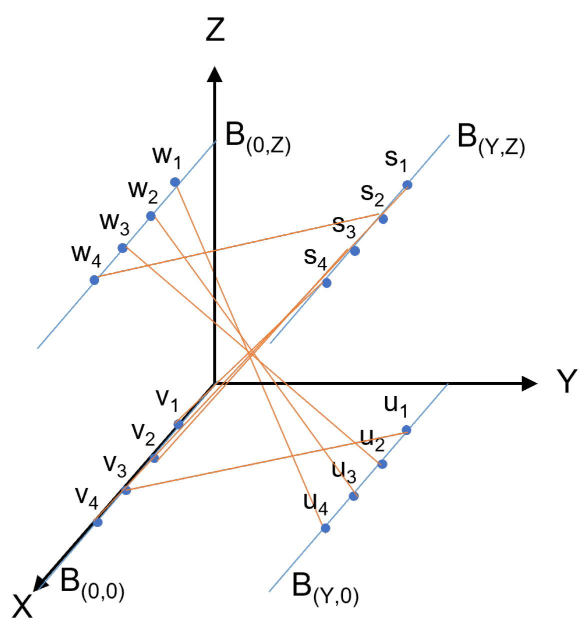

Figure 46: G after phase 2 of $XDZ4$ Algorithm

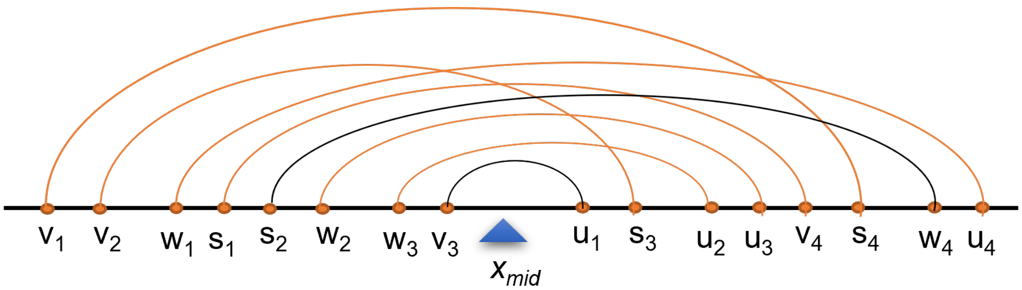

Figure 47: G after phase 2 of $XDZ4$ Algorithm, a view on the X-coordinate

Definition 6.10 *For Algorithm $XDZ4$, define $\boldsymbol{XDZ4-No-Delta}$ to be the Algorithm in the case that Algorithm $XDZ4$ would not stop according to the stopping condition $\Delta_M > \mathcal{Z}$, and finitely determined when M does not contain $X-disjoint$ edges.*

Remark 6.11 *Given G_4 a four balanced graph. All the matchings created by Algorithm $XDZ4$ and the match created by Algorithm $XDZ4-No-Delta$ before the stopping condition $\Delta_M > \mathcal{Z}$ is activated, are the same matches.*

Definition 6.12 *Given G_4 a four balanced graph and let M be the matching created by Algorithm $XDZ4$. Denote $\boldsymbol{\alpha}$ to be the number of swaps performed in Phase 2 of Algorithm $XDZ4$, and $\boldsymbol{p}$ be the number of iterations in Algorithm $XDZ4-No-Delta$.*

Definition 6.13 *For $0 \le j \le p$, denote $\boldsymbol{M_j}$ as the matching provided by Algorithm $XDZ4-No-Delta$ in the j iteration of Phase 2 of Algorithm $XDZ4$. Furthermore, define:*
For every $0 \le i \le p$, $\boldsymbol{xdz4_i} = val(M_i)$.
For example:
$\boldsymbol{xdz4_0} = val(M_0)$.
$\boldsymbol{xdz4_1} = val(M_1)$.
...
$\boldsymbol{xdz4_p} = val(M_p)$.

Definition 6.14 *Given G_4 a four balanced graph, define $\boldsymbol{\Delta_{M_i}}$ to be Δ_M, where M is the matching in Algorithm $XDZ4$ in the i iteration, $0 \le i \le p$.*

Remark 6.15 *Given G_4 a four balanced graph. For $0 \le i \le p$, if $i \le \alpha$, then $\Delta_{M_i} > \mathcal{Z}$.*
If $i > \alpha$, then $\Delta_{M_i} \le \mathcal{Z}$.

Definition 6.16 *Given G_4 a four balanced graph. For M_0 a matching resulted after Phase 1 of Algorithm $XDZ4$, denote:*
$\boldsymbol{M_0'} = X-Diagonals(B_{(0,0)}, B_{(\mathcal{Y},\mathcal{Z})})$,
$\boldsymbol{M_0''} = X-Diagonals(B_{(\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$.
Therefore, $M_0 = M_0' \cup M_0''$.

Definition 6.17 *Given G_4 a four balanced graph. For every matching M on G_4,*
Denote by $\boldsymbol{M'}$ the set of edges on the diagonal $B_{(0,0)}, B_{(\mathcal{Y},\mathcal{Z})}$.
Denote by $\boldsymbol{M''}$ the set of edges on the diagonal $B_{(\mathcal{Y},0)}, B_{(0,\mathcal{Z})}$.

Lemma 6.18 *Given G_4 a four balanced graph and let M be the matching which is a result of Algorithm $XDZ4$. In every iteration of Algorithm $XDZ4$, edges $e_{\Delta_M}, e'_{\Delta_M}$ are on different diagonals.*

Proof: By definition, edges $e_{\Delta_{M_0}}, e'_{\Delta_{M_0}}$ are $X-disjoint$. According to Lemma 4.19, Since M'_0 and M''_0 are all Diagonal Edges created by algorithm $X-Diagonals$ then M'_0 and M''_0 do not contain $X-disjoint$ edges. Therefore, one of them is in M'_0 and the other in M''_0, and thus are on different diagonals. ∎

Lemma 6.19 *Given G_4 a four balanced graph. Let M be a matching obtained during the process of Algorithm $XDZ4$. Every two edges $e_{\Delta_M}, e'_{\Delta_M}$ that are being swapped to f, f' in Phase 2 of Algorithm $XDZ4$ satisfy that $X[f,f'] = X[e_{\Delta_M}, e'_{\Delta_M}] + 2\Delta_M,\ Y[e_{\Delta_M}, e'_{\Delta_M}] = Y[f,f'] = 2\mathcal{Y},\ Z[e_{\Delta_M}, e'_{\Delta_M}] = 2\mathcal{Z},\ Z[f,f'] = 0$.*

Proof: Let f_1, f'_1 and f_2, f'_2 be the two possible sets of edges created by a $Y-Preserving$ and $X-Improving$ swap on e_{Δ_M} and e'_{Δ_M}.
According to Lemma 6.18, $e_{\Delta_M}, e'_{\Delta_M}$ are on different diagonals, then either $Y[f_1, f'_1] = 2\mathcal{Y}$, $Z[f_1, f'_1] = 0$ and $Y[f_2, f'_2] = 0$, $Z[f_2, f'_2] = 2\mathcal{Z}$ or the opposite.
The algorithm chooses the swap that preserve Y, denote these edges by $f,\ f'$.
By definition, $e_{\Delta_M}, e'_{\Delta_M}$ are the farthest $X-disjoint$ edges. Then according to Lemma 4.1, the swapped edges satisfy $X[f,f'] = X[e_{\Delta_M}, e'_{\Delta_M}] + 2\Delta_M$.
Therefore, by Algorithm $XDZ4$ the swap results $X[f,f'] = X[e_{\Delta_M}, e'_{\Delta_M}] + 2\Delta_M,\ Y[e_{\Delta_M}, e'_{\Delta_M}] = Y[f,f']$,
$Z[e_{\Delta_M}, e'_{\Delta_M}] = 2\mathcal{Z},\ Z[f,f'] = 0$. ∎

Lemma 6.20 *Given G_4 a four balanced graph with t vertices on each bank. M_0 is $Yopt$ and $Zopt$.*

Proof: For every edge $e \in M_0$ either $e \in M'_0$ or $e \in M''_0$. In both cases $Y[e] = \mathcal{Y}, Z[e] = \mathcal{Z}$.
Since there are t vertices on each bank, then there are 2t vertices in M'_0 and 2t vertices in M''_0. Therefore, $Y[M_0] = 2t\mathcal{Y}, Z[M_0] = 2t\mathcal{Z}$. According to Lemma 6.2, $Y[M_0] = Yopt$, and according to Lemma 6.4, $Z[M_0] = Zopt$. ∎

Lemma 6.21 *For every $0 \le j \le p$:*
$X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$.
$Y[M_{j+1}] = Y[M_j]$.
$Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$.

Proof: In every iteration in Phase 2 of Algorithm $XDZ4-No-Delta$, the algorithm swaps two $X-disjoint$ edges $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ using a $Y-Preserving$ and $X-Improving$ swap such that f, f' are the edges created by the swap. According to Lemma 6.19, these edges satisfy $X[f,f'] = X[e,e'] + 2\Delta_M,\ Y[e,e'] = Y[f,f'] = 2\mathcal{Y},\ Z[e,e'] = 2\mathcal{Z},\ Z[f,f'] = 0$,.
Therefore, $X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$, $Y[M_{j+1}] = Y[M_j]$, $Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$. ∎

Corollary 6.22 *For every* $0 \leq j \leq p$*:*
$X[M_j] = X[M_0] + \sum_{k=0}^{j-1} 2\Delta_{M_k}$.
$Y[M_j] = Y[M_0]$.
$Z[M_j] = Z[M_0] - 2j\mathcal{Z}$.
Therefore, $xdz4_{j+1} = xdz4_j + 2\Delta_{M_j} - 2j\mathcal{Z}$.
$xdz4_p = xdz4_0 + \sum_{k=0}^{p-1} 2\Delta_{M_k} - 2p\mathcal{Z}$.

Lemma 6.23 $xdz4_p$ *=* $Xopt$ *+* $Yopt$ *+* $Zopt$ *-* $2p\mathcal{Z}$*.*

Proof: According to Lemma 6.20, $Y[M_0] = Yopt$ and $Z[M_0] = Zopt$. According to Definition 6.12, p is the number of swaps until there is no $X - disjoint$ edges in the matching. Since in M_p there are no $X - disjoint$ edges, then, according to Lemma 4.13, $X[M_p] = Xopt$.
According to Corollary 6.22, $Y[M_p] = Y[M_0] = Yopt$, and $Z[M_p] = Z[M_0] - 2p\mathcal{Z} = Zopt - 2p\mathcal{Z}$.
Therefore, $xdz4_p$ = $Xopt$ + $Yopt$ + $Zopt$ - $2p\mathcal{Z}$. ∎

Lemma 6.24 $xdz4_\alpha = max\{xdz4_0, \ldots, xdz4_p\}$*.*

Proof: According to Corollary 6.22, for $0 \leq j \leq p$, $xdz4_j = xdz4_{j+1} - 2\Delta_{M_j} + 2\mathcal{Z}$.
Therefore, for $i \leq \alpha$, it holds that $xdz4_{i+1} \geq xdz4_i$.
For $i > \alpha$, it holds that $xdz4_{i+1} < xdz4_i$.
Hence, $xdz4_\alpha = max\{xdz4_0, \ldots, xdz4_p\}$. ∎

Lemma 6.25 *Let* M *be a matching obtained during the process of Algorithm* $XDZ4$*. For every* e*,* e' *that are* $X - disjoint$ *edges in* M*, without loss of generality, one of the vertices of* e *is in* $S'(M)$ *and one of the vertices of* e' *is in* $S''(M)$*.*

Proof: By Definition 3.22, all the vertices in $S'(M)$ are in M', and all the vertices in $S''(M)$ are in M''.
According to Lemma 6.18, e and e' are $X - disjoint$ edges and on different diagonals. Therefore, without loss of generality, e is in $S'(M)$ and e' is in $S''(M)$. ∎

Lemma 6.26 *For every* $0 \leq j \leq p$*, the number of vertices in* $S'(M_j)$ *is equal to the number of vertices in* $S''(M_j)$*.*

Proof: There must be an equal number of $X - one - sided$ edges on both sides of X_{mid}, and by definition, $S'(M_j)$ and $S''(M_j)$ are on different sides of X_{mid}. Therefore, the number of vertices in $S'(M_j)$ is equal to the number of vertices in $S''(M_j)$. ∎

Lemma 6.27 *Given* G_4 *a four balanced graph. For* M_j*, the matching resulting in Phase 2 of Algorithm* $XDZ4$ *in the* j *iteration,* $e_{\Delta_{M_j}}$ *and* $e'_{\Delta_{M_j}}$ *contains the minimal* $X - value$ *vertex in* $S'(M)$ *and the maximal* $X - value$ *vertex in* $S''(M)$*.*

Proof: Since $e_{\Delta_{M_j}}$ and $e'_{\Delta_{M_j}}$ are $X - disjoint$, then according to Lemma 6.25, without loss of generality, one of the vertices of $e_{\Delta_{M_j}}$ is in $S'(M_j)$ and one of the vertices of $e'_{\Delta_{M_j}}$ is in $S''(M_j)$.
Since Δ_{M_j} is the maximum distance between two vertices, one in $S'(M_j)$ and the other in $S''(M_j)$, then those vertices must be the minimal $X - value$ vertex in $S'(M)$ and the maximal $X - value$ vertex in $S''(M)$, see Figure 64. ∎

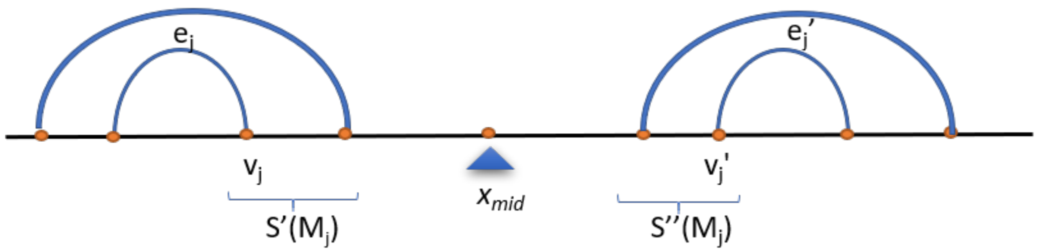

Figure 48: A matching with the farthest $X - disjoint$ e and e'

Lemma 6.28 *Given G_4 a four balanced graph. For M_j, the matching resulting in Phase 2 of Algorithm $XDZ4$ in the j iteration, a $Y - Preserving$ and $X - Improving$ swap performed on $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ results in $f,\ f'$, that are not $X - disjoint$ with any other edge in M_j.*

Proof: Denote $e_{\Delta_{M_j}} = (v_1, v_2), e'_{\Delta_{M_j}} = (u_1, u_2)$. According to Lemma 6.27, without loss of generality, v_2 is the minimal vertex in $S'(M_j)$ and u_1 is the maximal vertex in $S''(M_j)$.
The possible swaps are $f_1, f_1' = (v_1, u_2), (v_2, u_1)$ or $f_2, f_2' = (v_1, u_1), (v_2, u_2)$. Since both options are $X - Improving$, choose $f,\ f'$ to be the $X - Improving$ and $Y - Preserving$ swap, see Figure 49. Hence edges $f,\ f'$ satisfy that $X - cross$ all the vertices in $S'(M_j)$ and $S''(M_j)$, and are not $X - disjoint$ with any edge which contains vertices from $S(M_j)$. Since $f,\ f'$ $X - cross$ X_{mid} and thus, according to Lemma 4.7, $f,\ f'$ are not $X - disjoint$ with any edge that $X - cross$ X_{mid}. Therefore, $f,\ f'$ are not $X - disjoint$ with any edge in M_j. ∎

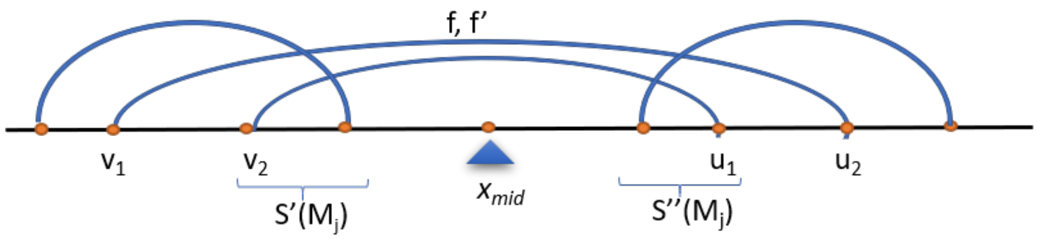

Figure 49: A matching after swapping to f and f'

Lemma 6.29 *For every two matchings M and M' on G_4 a four balanced graph, if $S'(M) = S'(M')$ and $S''(M) = S''(M')$, the distance between the farthest $X - disjoint$ edges satisfies $\Delta_M = \Delta_{M'}$.*

Proof: Δ_M and $\Delta_{M'}$ are the distances on the X-axis between the farthest $X - disjoint$ edges. Therefore, Δ_M is the distance between the minimal $X - value$ vertex in $S'(M)$ and the maximal $X - value$ vertex in $S''(M)$ and $\Delta_{M'}$ is the distance between the minimal $X - value$ vertex in $S'(M')$ and the maximal $X - value$ vertex in $S''(M')$. Since $S'(M) = S'(M')$ and $S''(M) = S''(M')$, it follows $\Delta_M = \Delta_{M'}$ ∎

Lemma 6.30 *For every $0 \le j \le p$ and for every $M_j^o \in OPT_j$, $Y[M_j^o] = Y[M_j]$ and $Z[M_j^o] = Z[M_j]$.*

Proof: For every j, M_j^o is a matching with the maximum value with exactly $2j$ edges with $Z-value = 0$, and zero edges with $Y-value = 0$. Therefore, $Z[M_j^o] = Zopt - 2j\mathcal{Z}$ and $Y[M_j^o] = Yopt$. According to Corollary 6.22, $Z[M_j] = Z[M_0] - 2j\mathcal{Z}$, and $Y[M_j] = Y[M_0]$. According to Lemma 6.20, $Z[M_0] = Zopt$, and $Y[M_0] = Yopt$.
Therefore, $Z[M_j] = Zopt - 2j\mathcal{Z} = Z[M_j^o]$ and $Y[M_j] = Yopt = Y[M_j^o]$. ■

Lemma 6.31 *For every $M_0^o \in OPT_0$, $X[M_0^o] = X[M_0]$ and $opt_0 = xdz4_0$.*

Proof: X-Diagonals Algorithm returns, according to Lemma 4.20, a matching which is optimal with respect to X-coordinate. Therefore, $X[M_0']$ is optimal on the X-coordinate for the set of vertices $B_{(0,0)} \cup B_{(\mathcal{Y},\mathcal{Z})}$, and $X[M_0'']$ is optimal on the X-coordinate for the set of vertices $B_{(\mathcal{Y},0)} \cup B_{(0,\mathcal{Z})}$. Therefore, $X[M_0^{o\prime}] \leq X[M_0']$ and $X[M_0^{o\prime\prime}] \leq X[M_0'']$. Furthermore, since $M_0^o \in OPT_0$, then $X[M_0] \leq X[M_0^o]$. Summing these inequalities yields that $X[M_0^o] = X[M_0]$ and $opt_0 = xdz4_0$. ■

Lemma 6.32 *For every matching $M_0^o \in OPT_0$, $S'(M_0) = S'(M_0^o)$ and $S''(M_0) = S''(M_0^o)$.*

Proof: According to Definition 6.16 and Notation 6.3, matchings M_0' and $M_0^{o\prime}$ are $Xopt$ on the diagonals. Therefore, according to Lemma 4.28, $S'(M_0) = S'(M_0^o)$. Similarly, $S''(M_0) = S''(M_0^o)$. ■

Corollary 6.33 *If e and e' are $X-disjoint$ edges in M_0 or in $M_0^o \in OPT_0$, then the vertices of e and e' which are $X-closer$ to X_{mid} are in $S'(M_0) = S'(M_0^o)$ and in $S''(M_0) = S''(M_0^o)$.*

Lemma 6.34 *Any matching $M_1^o \in OPT_1$ satisfies $S'(M_1) = S'(M_1^o)$, $S''(M_1) = S''(M_1^o)$.*

Proof: According to Definition 3.8, M_1^o contains exactly two edges e and e' whose $Z-value$ is zero with $Y[e] = Y[e'] = \mathcal{Y}$. Perform a $Y-Preserving\ Z-Improving$ swap on e and e', and let f and f' be the resulting edges. Since $Z[f] = Z[f'] = \mathcal{Z}$ and $Y[f] = Y[f'] = \mathcal{Y}$ then f and f' are Diagonal Edges.
Denote $\widehat{M_1^o} = M_1^o \setminus \{e, e'\} \cup \{f, f'\}$. In $\widehat{M_1^o}$ there are zero edges with $Z-value = 0$ and zero edges with $Y-value = 0$.
Suppose that $\widehat{M_1^o} \notin OPT_0$. Perform a swap which improves $X-value$ without changing $Y-value$ or $Z-value$. But, in this case, we could have performed a similar swap in M_1^o and improve $X-value$, contradicting the assumption that $M_1^o \in OPT_1$. Hence $\widehat{M_1^o} \in OPT_0$.
According to Lemma 6.32, $S'(M_0) = S'(\widehat{M_1^o})$ and $S''(M_0) = S''(\widehat{M_1^o})$. Therefore, Δ_{M_0} is between the farthest vertices in $S'(M_0) = S'(\widehat{M_1^o})$ and $S''(M_0) = S''(\widehat{M_1^o})$.
$\Delta_{\widehat{M_1^o}}$ is between f and f' and is also between the farthest vertices in $S'(\widehat{M_1^o})$ and $S''(\widehat{M_1^o})$. Since $S'(M_0) = S'(\widehat{M_1^o})$ and $S''(M_0) = S''(\widehat{M_1^o})$ these are the same vertices, denote these vertices by v' and v''.
According to Algorithm $XDZ4$, $S'(M_1) = S'(M_0) \setminus v'$ and $S''(M_1) = S''(M_0) \setminus v''$.
Since M_1^o is achieved by replacing f and f' in $\widehat{M_1^o}$ by e and e', then $S'(M_1^o) = S'(\widehat{M_1^o}) \setminus v'$ and $S''(M_1^o) = S''(\widehat{M_1^o}) \setminus v''$.
Therefore, $S'(M_1) = S'(M_1^o)$ and $S''(M_1) = S''(M_1^o)$. ■

Lemma 6.35 *For every $0 \leq j \leq p$, any matching $M_j^o \in OPT_j$ provides that $S'(M_j) = S'(M_j^o)$ and $S''(M_j) = S''(M_j^o)$.*

Proof: We prove this lemma by induction on j.
For $j = 0$, according to Lemma 6.32, $S'(M_0) = S'(M_0^o)$ and $S''(M_0) = S''(M_0^o)$.
For $j = 1$, according to Lemma 6.34, $S'(M_1) = S'(M_1^o)$ and $S''(M_1) = S''(M_1^o)$.
Assume the lemma is correct for all $k < j$ and prove it for j.
According to Definition 3.8, M_j^o contains exactly $2j$ edges whose $Z-value$ is zero with $Y-value = \mathcal{Y}$. Choose $e,\ e'$ such that their X distance is the minimal.
Perform a $Z-Improving$ and $Y-Preserving$ swap on e and e', and let $f,\ f'$ be the resulting edges. Since $Z[f] = Z[f'] = \mathcal{Z}$ and $Y[f] = Y[f'] = \mathcal{Y}$ then f and f' are Diagonal Edges.
Denote $\widehat{M_j^o} = M_j^o \setminus \{e, e'\} \cup \{f, f'\}$. In $\widehat{M_j^o}$ there are $2(j-1)$ edges with $Z-value = 0$ and zero edges with $Y-value = 0$.
Suppose that $\widehat{M_j^o} \notin OPT_{j-1}$. Perform a swap which improves $X-value$ without changing $Y-value$ or $Z-value$. But, in this case, we could have performed a similar swap in M_j^o and improve $X-value$, contradicting the assumption that $M_j^o \in OPT_j$. Hence $\widehat{M_j^o} \in OPT_{j-1}$.
According to the induction assumption, $S'(\widehat{M_j^o}) = S'(M_{j-1})$ and $S''(\widehat{M_j^o}) = S''(M_{j-1})$. Therefore, $\Delta_{M_{j-1}}$ is between the farthest vertices in $S'(\widehat{M_j^o}) = S'(M_{j-1})$ and $S''(\widehat{M_j^o}) = S''(M_{j-1})$.
$\Delta_{\widehat{M_j^o}}$ is between f and f' and is also between the farthest vertices in $S'(\widehat{M_j^o})$ and $S''(\widehat{M_j^o})$.
Since $S'(\widehat{M_j^o}) = S'(M_{j-1})$ and $S''(\widehat{M_j^o}) = S''(M_{j-1})$ these are the same vertices, denote these vertices as v' and v''.
According to Algorithm $XDZ4$, $S'(M_j) = S'(M_{j-1}) \setminus v'$ and $S''(M_j) = S''(M_{j-1}) \setminus v''$.
Since M_j^o is achieved by replacing f and f' in $\widehat{M_j^o}$ by e and e', then $S'(M_j^o) = S'(\widehat{M_j^o}) \setminus v'$ and $S''(M_j^o) = S''(\widehat{M_j^o}) \setminus v''$.
Therefore, $S'(M_j) = S'(M_j^o)$ and $S''(M_j) = S''(M_j^o)$. ■

Lemma 6.36 *For every index $0 \leq j \leq p$, $opt_j = xdz4_j$.*

Proof: We prove this lemma by induction on j.
For j=0, according to Lemma 6.31, $opt_0 = xdz4_0$.
Assume the lemma is correct for all $k < j$ and prove it for j.
For any matching $M_j^o \in OPT_j$, chose the two edges e and e' with $Z-value = 0$ such that the $X-value$ of those edges is minimal. Let $f,\ f'$ be the resulting edges. Since $Z[f] = Z[f'] = \mathcal{Z}$ and $Y[f] = Y[f'] = \mathcal{Y}$, then f and f' are Diagonal Edges.
Denote $\widehat{M_j^o} = M_j \setminus \{e, e'\} \cup \{f, f'\}$. In M_j^o there are $2(j-1)$ edges with $Z-value = 0$ and zero edges with $Y-value = 0$.
Suppose that $\widehat{M_j^o} \notin OPT_{j-1}$. Perform a swap which improves $X-value$ without changing $Y-value$ or $Z-value$. But, in this case, we could have performed a similar swap in M_j^o and improve $X-value$, contradicting the assumption that $M_j^o \in OPT_j$. Hence $\widehat{M_j^o} \in OPT_{j-1}$.
According to Lemma 6.35, $S'(M_{j-1}) = S'(\widehat{M_j^o})$ and $S''(M_{j-1}) = S''(\widehat{M_j^o})$. Therefore, according to Lemma 6.29, $\Delta_{M_{j-1}} = \Delta_{\widehat{M_j^o}}$, and $X[M_j^o] = X[\widehat{M_j^o}] + 2\Delta_{M_{j-1}}$.
According to Lemma 6.21, $X[M_j] = X[M_{j-1}] + 2\Delta_{M_{j-1}}$. Therefore, $X[M_j^o] = X[M_j]$.
According to Lemma 6.30, $Y[M_j] = Y[M_j^o]$ and $Z[M_j] = Z[M_j^o]$.
Therefore, $xdz4_j = opt_j$ and $opt_j = opt_{j-1} + 2\Delta_{M_{j-1}} - 2(j-1)\mathcal{Z}$. ■

Lemma 6.37 $opt_\alpha = max\{opt_0, \ldots, opt_p\}$.

Proof: According to Lemma 6.36 for every index $0 \le j \le p$, $opt_j = xdz4_j$. According to Lemma 6.24, $xdz4_\alpha = max\{xdz4_0, \ldots, xdz4_p\}$. Therefore, $opt_\alpha = max\{opt_0, \ldots, opt_p\}$. ∎

Theorem 6.38 *Given G_4, a four balanced graph with t vertices on each bank. Algorithm $XDZ4$ on G_4 is finitely determined and returns a maximum matching.*

Proof: According to Definition 6.10, Algorithm $XDZ4$ finitely determined, and the result of the algorithm is a matching M which satisfies $val(M) = xdz4_\alpha$.
According to Lemma 6.36, $xdz_\alpha = opt_\alpha$ and according to Lemma 6.37, $opt = opt_\alpha$.
Hence Algorithm $XDZ4$ returns a maximum value matching. ∎

7 Four Unbalanced Banks Matching

In this section we consider matchings on a graph with four unbalanced banks where the core idea of the algorithm is similar to the four balanced graph. In the beginning of this section we introduce basic properties concerning this case, followed by an algorithms which finds a maximum matching. The section continues by a set of definitions regarding the algorithm. Finally we prove the correctness of the algorithm.

In all our discussion and algorithms we assume that $\mathcal{Y} \ge \mathcal{Z}$. Otherwise, change the roles of Y axis and Z axis.

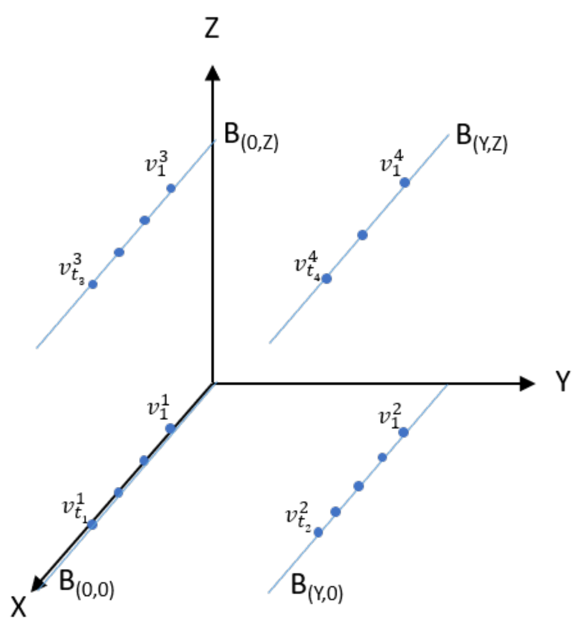

Figure 50: Four Unbalanced Banks

7.1 Definitions

Property 7.1 *Given $G_4^{\bigtriangledown}$, a four unbalanced banks graph. For $0 \leq j \leq p$, the number of edges with $Y - value = 0$ in opt_j is l_Y and the number of edges with $Z - value = 0$ is $l_Z + 2j$.*

Remark 7.2 *For every $M_0^o \in OPT_0$, according to Definition 7.1, $Y[M_0^o] = Yopt$, and $Z[M_0^o] = Zopt$.*

Property 7.3 *In order for an edge $e = (v, u)$ to have a vertex in $S'(M)$, both sides of the edge must be on the same side of X_{mid}. In addition, if two edges e, f are $X - disjoint$, then each is on a different side of X_{mid} and therefore one* **touches** *$S'(M)$ and the other* **touches** *$S''(M)$.*

Definition 7.4 *Given $G_4^{\bigtriangledown} = (V, E)$, a four unbalanced banks graph, and $V' \in V$. Define $\boldsymbol{B_{(y,z)}[V']}$ to be the set of vertices in V' on $B_{(y,z)}$ bank.*

Definition 7.5 *Given $G_4^{\bigtriangledown}$ a four unbalanced banks graph. For $1 \leq i \leq n$, for $\boldsymbol{e_i}$ **on** $\boldsymbol{S(M)}$ where there are n vertices in $S(M)$ that are arranged according to their X-coordinate where one of the vertices of e_i is in the i-th place.*

7.2 Basic claims

Lemma 7.6 *Given $G_4^{\bigtriangledown}$, a four unbalanced banks graph with $t_{(0,0)}, t_{(\mathcal{Y},0)}, t_{(0,\mathcal{Z})}, t_{(\mathcal{Y},\mathcal{Z})}$ vertices on four banks. $Yopt = min\{(t_{(0,0)} + t_{(0,\mathcal{Z})}), (t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Y}$.*

Proof: The maximum $Y - value$ for an edge e is $\mathcal{Y}$, which happens if one vertex of e is on $P_{Y=0}$ and the other vertex of e is on $P_{Y=\mathcal{Y}}$. Since on $P_{Y=0}$ there are $t_{(0,0)} + t_{(0,\mathcal{Z})}$ vertices, and on $P_{Y=\mathcal{Y}}$ there are $t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})}$ vertices, then the maximum number of edges with $Y - value = \mathcal{Y}$ is $min\{t_{(0,0)} + t_{(0,\mathcal{Z})}, t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})}\}$.
Therefore, $Yopt = min\{(t_{(0,0)} + t_{(0,\mathcal{Z})}), (t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Y}$. ■

Lemma 7.7 *Given $G_4^{\bigtriangledown}$, a four unbalanced banks graph with $t_{(0,0)}, t_{(\mathcal{Y},0)}, t_{(0,\mathcal{Z})}, t_{(\mathcal{Y},\mathcal{Z})}$ vertices on four banks. $Zopt = min\{(t_{(0,0)} + t_{(\mathcal{Y},0)}), (t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Z}$.*

Proof: The maximum $Z - value$ for an edge e is $\mathcal{Z}$, which happens if one vertex of e is on $P_{Z=0}$ and the other vertex of e is on $P_{Z=\mathcal{Z}}$. Since on $P_{Z=0}$ there are $t_{(0,0)} + t_{(\mathcal{Y},0)}$ vertices, and on $P_{Z=\mathcal{Z}}$ there are $t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})}$ vertices, then the maximum number of edges with $Z - value = \mathcal{Z}$ is $min\{t_{(0,0)} + t_{(\mathcal{Y},0)}, t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})}\}$.
Therefore, $Zopt = min\{(t_{(0,0)} + t_{(\mathcal{Y},0)}), (t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Z}$. ■

Definition 7.8 *Define $\boldsymbol{l_Y}$ to be the minimum number of edges with $\boldsymbol{Y - value = 0}$ in a $\boldsymbol{Yopt}$ matching.*
$\boldsymbol{l_Y = |(t_{(0,0)} + t_{(0,\mathcal{Z})}) - (t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})})|}$.
Define $\boldsymbol{l_Z}$ to be the minimum number of edges with $\boldsymbol{Z - value = 0}$ in a $\boldsymbol{Zopt}$ matching.
$\boldsymbol{l_Z = |(t_{(0,0)} + t_{(\mathcal{Y},0)}) - (t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})})|}$.

Lemma 7.9 *Given $G_4^{\triangledown}$, a four unbalanced banks graph. If a matching M is not $Yopt$, then there exists $i \geq 1$ such that M contains $l_Y + 2i$ edges with $Y - value = 0$ with at least one edge on $P_{Y=0}$ and at least one edge on $P_{Y=\mathcal{Y}}$.*

Proof: Denote by l_M the number of edges whose $Y - value = 0$.
For each edge e, $Y[e] = \mathcal{Y}$ if and only if e connects vertices on two different planes. Assume, without loss of generality, that in $G_4^{\triangledown}$ the number of vertices on plane $P_{Y=0}$ is bigger than the number of vertices on plane $P_{Y=\mathcal{Y}}$. In that case, $t_{(0,0)} + t_{(0,\mathcal{Z})} > t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})}$. Since M is not $Yopt$, the number of edges whose $Y - value = \mathcal{Y}$ is smaller than both $t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})}$ and $t_{(0,0)} + t_{(0,\mathcal{Z})}$. Therefore, there is at least one edge e on plane $P_{Y=0}$ and one edge e' on plane $P_{Y=\mathcal{Y}}$. Furthermore, $Y[e, e'] = 0$.
A $Y - Improving$ swap on e and e' yields $f,\ f'$ such that $Y[f, f'] = 2\mathcal{Y}$ and in the new matching the number of edges whose $Y - value = 0$ is $l_M - 2$. We can continue in this manner until we reach l_Y edges with $Y - value = 0$. If we perform i swaps we resulted in $l_M - 2i = l_Y$ edges with $Y - value = 0$. Hence $l_M = l_Y + 2i$. ■

Corollary 7.10 *Given $G_4^{\triangledown}$, a four unbalanced banks graph. If a matching M is not $Zopt$, for $i \geq 1$, then M contains $l_Z + 2i$ edges with $Z - value = 0$.*

Proof: Similar to the proof of 7.9. ■

Lemma 7.11 *Given $G_4^{\triangledown}$, a four unbalanced banks graph. If $\mathcal{Y} > \mathcal{Z}$, then every maximum matching $M \in OPT$ is $Yopt$.*

Proof: Suppose by contradiction that M is a maximum matching on an unbalanced graph which is not $Yopt$. According to Lemma 7.9, it contains at least $l_Y + 2$ edges with $Y - value = 0$. Furthermore, there exist e and e' such that $Y[e] = Y[e'] = 0$ with $e \in P_{Y=0}$ and $e' \in P_{Y=\mathcal{Y}}$.
Let $f_1,\ f_1'$ and $f_2,\ f_2'$ be the two possible sets of edges created by $Y - Improving$ swap on e and e'. At least one pair of edges from f_1, f_1' or $f_2,\ f_2'$ does not contain $X - disjoint$ edges, denote those edges by $f,\ f'$.
Let $M' = M \setminus \{e, e'\} \cup \{f, f'\}$. $Y[M'] = Y[M] + 2Y$, $Z[M'] \geq Z[M] - 2Z$. Since f, f' are not $X - disjoint$ then according to Lemma 4.2, $X[M'] \geq X[M]$. Since $\mathcal{Y} > \mathcal{Z}$, the value of the matching satisfies $val(M) < val(M')$, contradicting the assumption that M is a maximum matching. ■

Lemma 7.12 *Given $G_4^{\triangledown}$, a four unbalanced banks graph. If $\mathcal{Y} > \mathcal{Z}$, then there exists $0 \leq i^* \leq p$ such that $opt = opt_{i^*}$.*

Proof: According to Lemma 7.11, every $M \in OPT$ on $G_4^{\triangledown}$ is $Yopt$. According to Corollary 7.10, there is an index i^* such that there are $l_Z + 2i^*$ edges with $Z - value = 0$. Therefore, $Y[M] = Yopt$, $Z[M] = Zopt - 2(i^*)\mathcal{Z}$. Hence, $M \in OPT_{i^*}$ and $val(M) = opt_{i^*} = opt$. ■

Lemma 7.13 *Given $G_4^{\triangledown}$, a four unbalanced banks graph, and let M be a matching on the graph. The number of vertices in $S'(M)$ is equal to the number of vertices in $S''(M)$.*

Proof: According to Definition 3.18, there is an equal number of $X - one - sided$ edges on both sides of X_{mid}. By Definition 3.22, $S'(M)$ and $S''(M)$ are on different sides of X_{mid}. Therefore, the number of vertices in $S'(M)$ is equal to the number of vertices in $S''(M)$. ■

7.3 The Algorithm

Algorithm 4 $XDZ4 - UN$**:** Finds a maximum perfect matching,

function XDZ4-UN()

Input:

A graph $G = (V, E)$ with four unbalanced banks graph with $t_{(0,0)}, t_{(\mathcal{Y},0)}, t_{(0,\mathcal{Z})}, t_{(\mathcal{Y},\mathcal{Z})}$ vertices on the banks. for each bank the vertices are ordered according to the X-coordinate $v_1, \ldots, v_{ti}$

Output:

Return matching M

begin

Phase 1:

Initialize an empty match M

$F^1_{(0,0),(\mathcal{Y},\mathcal{Z})} = X - Diagonals(B_{(0,0)}, B_{(\mathcal{Y},\mathcal{Z})}),\ F^1_{(\mathcal{Y},0),(0,\mathcal{Z})} = X - Diagonals(B_{(\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$

$F_1 = F^1_{(0,0),(\mathcal{Y},\mathcal{Z})} \cup F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$

$M = M \cup F_1$

Phase 2:

For $V' = V \setminus V(M)$, perform:

$F^2_{(0,0),(\mathcal{Y},0)} =$ X-Diagonals$(B_{(0,0)}[V'], B_{(\mathcal{Y},0)}[V'])$

$F^2_{(0,\mathcal{Z}),(\mathcal{Y},\mathcal{Z})} =$ X-Diagonals$(B_{(0,\mathcal{Z})}[V'], B_{(\mathcal{Y},\mathcal{Z})}[V'])$

$F^2_{(0,0),(0,\mathcal{Z})} =$ X-Diagonals$(B_{(0,0)}[V'], B_{(0,\mathcal{Z})}[V'])$

$F^2_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})} =$ X-Diagonals$(B_{(\mathcal{Y},0)}[V'], B_{(\mathcal{Y},\mathcal{Z})}[V'])$

$F_2 = F^2_{(0,0),(\mathcal{Y},0)} \cup F^2_{(0,\mathcal{Z}),(\mathcal{Y},\mathcal{Z})} \cup F^2_{(0,0),(0,\mathcal{Z})} \cup F^2_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$

$M = M \cup F_2$

Phase 3:

For $V' = V \setminus V(M)$, perform:

$F^3_{(0,0)} =$ One-Bank-Match$(B_{(0,0)}[V'])$, $F^3_{(\mathcal{Y},0)} =$ One-Bank-Match$(B_{(\mathcal{Y},0)}[V'])$

$F^3_{(0,\mathcal{Z})} =$ One-Bank-Match$(B_{(0,\mathcal{Z})}[V'])$, $F^3_{(\mathcal{Y},\mathcal{Z})} =$ One-Bank-Match$(B_{(\mathcal{Y},\mathcal{Z})}[V'])$

$F_3 = F^3_{(0,0)} \cup F^3_{(\mathcal{Y},0)} \cup F^3_{(0,\mathcal{Z})} \cup F^3_{(\mathcal{Y},\mathcal{Z})}$

$M = M \cup F_3$

Phase 4:

$L =$ *All the edges touching* $S(M)$ (*see Definition* 3.22). $n = |L|$

for $i = 1, \ldots, \frac{n}{2}$:

e_i and e_{n-i+1} defined on $S(M)$ (according to Definition 7.5)

if e_i and e_{n-i+1} are not both Diagonal Edges :

Let f, f' be the edges created by $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap on e_i, e_{n-i+1} (Such a swap exists according to Lemma 7.26)

$M = M \setminus \{e_i, e_{n-i+1}\} \cup \{f, f'\}$

end if

end for

Phase 5:

while M contains $X - disjoint$ edges :

Find $e_{\Delta_M},\ e'_{\Delta_M}$ (see Definition 3.13)

while $\Delta_M > \mathcal{Z}$:

Let f, f' be the edges created by $Y - Preserving$ and $X - Improving$ swap on $e_{\Delta_M},\ e'_{\Delta_M}$

$M = M \setminus \{e_{\Delta_M}, e'_{\Delta_M}\} \cup \{f, f'\}$

Find $\{e_{\Delta_M}, e'_{\Delta_M}\}$ (see Definition 3.13)

end while

end while

Definition 7.14 *Define* $\mathbf{XDZ4 - UN - No - Delta}$ *to be a variation of Algorithm* $XDZ4-UN$ *which does not stop according to the stopping condition* $\Delta_M > \mathcal{Z}$*, and stops when* M *does not contain* $X - disjoint$ *edges.*

Remark 7.15 *Given* $G_4^{\triangledown}$ *a four unbalanced banks graph. All matchings created by Algorithm* $XDZ4 - UN$ *and the matchings created by Algorithm* $XDZ4 - UN - No - Delta$*, before the stopping condition* $\Delta_M > \mathcal{Z}$ *is activated, are the same matchings.*

Definition 7.16 *Given* $G_4^{\triangledown}$ *a four unbalanced banks graph. Let* M *be the matching created by Algorithm* $XDZ4 - UN$*. Denote* $\boldsymbol{\alpha}$ *to be the number of swaps performed in Phase 5 of Algorithm* $XDZ4-UN$ *and* $\boldsymbol{p}$ *to be the number of iterations in Phase 5 of Algorithm* $XDZ4-UN-No-Delta$*.*

Definition 7.17 *Given* $G_4^{\triangledown}$ *a four unbalanced banks graph. Suppose this graph is the input to Algorithm* $XDZ4 - UN$*. Denote:*
Let $\boldsymbol{M_0^1}$ *be the matching after Phase 1.*
Let $\boldsymbol{M_0^2}$ *be the matching after Phase 2.*
Let $\boldsymbol{M_0^3}$ *be the matching after Phase 3.*
Let $\boldsymbol{M_0^4}$ *be the matching after Phase 4.*

Definition 7.18 *For* $0 \leq j \leq p$*, Denote* M_j *as the matching provided by Algorithm* $XDZ4-UN-No-Delta$ *in the* j *iteration of Phase 5. Define:*
For every $0 \leq j \leq p$*,* $\boldsymbol{xdz4_j^{\triangledown}} = val(M_j)$*.*
For example:
$\boldsymbol{xdz4_0^{\triangledown}} = val(M_0^4)$*.*
$\boldsymbol{xdz4_1^{\triangledown}} = val(M_1)$*.*
...
$\boldsymbol{xdz4_p^{\triangledown}} = val(M_p)$*.*

Definition 7.19 *Given* $G_4^{\triangledown}$ *a four unbalanced banks graph. Define* $\mathbf{\Delta_{M_j}}$ *to be* Δ_M *where* M *is the matching at the* j *iteration of Phase 5 of Algorithm* $XDZ4 - UN - No - Delta$*,* $0 \leq j \leq p$*.*

Remark 7.20 M_0^1 *is combined from two sets of edges, the Diagonal Edges in* $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ *and the Diagonal Edges in* $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$*.* $M_0^1 = F_1$*.*
Since $G_4^{\triangledown}$ *is a four unbalanced banks graph, by enumeration argument, in* $G_4^{\triangledown}$ *there are two banks where all vertices on the bank are matched in* M_0^1 *and two banks which are on the same plane that may contain vertices that are not matched. Denote these banks as* B', B''*.*
For example, if $t_{(0,0)} > t_{(\mathcal{Y},\mathcal{Z})}$ *and* $t_{(\mathcal{Y},0)} > t_{(0,\mathcal{Z})}$*, then all vertices in* $B_{(0,\mathcal{Z})}$ *and* $B_{(\mathcal{Y},\mathcal{Z})}$ *are matched, but there are* $t_{(0,0)} - t_{(\mathcal{Y},\mathcal{Z})}$ *vertices on* $B_{(0,0)}$ *and* $t_{(\mathcal{Y},0)} - t_{(0,\mathcal{Z})}$ *vertices on* $B_{(\mathcal{Y},0)}$ *which are not matched in Phase 1.*
M_0^2 *is the union of edges in* M_0^1 *and edges returned by* $F_2 = X - Diagonals(B', B'')$*. In Phase 3, the matching is performed either on vertices from* $B_{(0,0)}$ *or on vertices from* $B_{(\mathcal{Y},0)}$*. Denote this bank as* B*.*
M_0^3 *is the union of edges in* M_0^2 *and* $F_3 = One - Bank - Match(B)$*.* $M_0^3 = F_1 \cup F_2 \cup F_3$*.*

7.4 Phase 3

Lemma 7.21 *Given $G_4^{\triangledown}$ a four unbalanced banks graph.* $Y[M_0^3] = Yopt$.

Proof: First we note that, in Phase 3 of Algorithm $XDZ4 - UN$, all the edges in F_3 are on the same bank, therefore $Y[F_3] = 0$.
Consider the first two stages of the algorithm, there are two options:

1. Suppose $t_{(0,0)} < t_{(\mathcal{Y},\mathcal{Z})}$ and $t_{(0,\mathcal{Z})} < t_{(\mathcal{Y},0)}$,
(A similar proof holds for the case $t_{(0,0)} > t_{(\mathcal{Y},\mathcal{Z})}$ and $t_{(0,\mathcal{Z})} > t_{(\mathcal{Y},0)}$).
According to Lemma 7.6, $Yopt = (t_{(0,0)} + t_{(0,\mathcal{Z})})\mathcal{Y}$. In Phase 1, $Y[F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}] = \mathcal{Y}t_{(0,0)}$ and $Y[F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}] = \mathcal{Y}t_{(0,\mathcal{Z})}$.
In Phase 2, all the vertices of the edges in F_2 are on $P_{Y=\mathcal{Y}}$. Therefore, $Y[F_2] = 0$.
Therefore, $Y[M_0^3] = Y[F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}] + Y[F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}] = \mathcal{Y}(t_{(0,0)} + t_{(0,\mathcal{Z})}) = Yopt$.

2. Suppose $t_{(0,0)} < t_{(\mathcal{Y},\mathcal{Z})}$ and $t_{(\mathcal{Y},0)} < t_{(0,\mathcal{Z})}$,
(A similar proof holds for the case $t_{(0,0)} > t_{(\mathcal{Y},\mathcal{Z})}$ and $t_{(\mathcal{Y},0)} > t_{(0,\mathcal{Z})}$).
In Phase 1, $Y[F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}] = \mathcal{Y}t_{(0,0)}$, $Y[F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}] = \mathcal{Y}t_{(\mathcal{Y},0)}$.
In Phase 2, all the vertices of the edges in F_2 are on $B_{(\mathcal{Y},\mathcal{Z})}$ and on $B_{(0,\mathcal{Z})}$. Therefore, $Y[F_2] = min\{t_{(\mathcal{Y},\mathcal{Z})} - t_{(0,0)}, t_{(0,\mathcal{Z})} - t_{(\mathcal{Y},0)}\}$.

   - If $t_{(\mathcal{Y},\mathcal{Z})} - t_{(0,0)} < t_{(0,\mathcal{Z})} - t_{(\mathcal{Y},0)}$
   then $t_{(\mathcal{Y},\mathcal{Z})} + t_{(\mathcal{Y},0)} < t_{(0,\mathcal{Z})} + t_{(0,0)}$.
   According to Lemma 7.6, $Yopt = \mathcal{Y}(t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})})$.
   In this case, it is also true that $Y[M_0^3] = Y[F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}] + Y[F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}] + Y[F_2] = \mathcal{Y}(t_{(0,0)} + t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})} - t_{(0,0)}) = \mathcal{Y}(t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})}) = Yopt$.

   - If $t_{(\mathcal{Y},\mathcal{Z})} - t_{(0,0)} > t_{(0,\mathcal{Z})} - t_{(\mathcal{Y},0)}$
   then $t_{(\mathcal{Y},\mathcal{Z})} + t_{(\mathcal{Y},0)} > t_{(0,\mathcal{Z})} + t_{(0,0)}$.
   According to Lemma 7.6, $Yopt = \mathcal{Y}(t_{(0,0)} + t_{(0,\mathcal{Z})})$.
   In this case, it is also true that $Y[M_0^3] = Y[F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}] + Y[F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}] + Y[F_2] = \mathcal{Y}(t_{(0,0)} + t_{(\mathcal{Y},0)} + t_{(0,\mathcal{Z})} - t_{(\mathcal{Y},0)}) = \mathcal{Y}(t_{(0,0)} + t_{(0,\mathcal{Z})}) = Yopt$.

■

Lemma 7.22 *Given $G_4^{\triangledown}$ a four unbalanced banks graph.* $Z[M_0^3] = Zopt$.

Proof: Similar to the proof of Lemma 7.21. ■

Corollary 7.23 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. $Y[M_0^3] = Yopt$ and $Z[M_0^3] = Zopt$.*

Property 7.24 *There exists a four unbalanced banks graph $G_4^{\triangledown}$ which satisfies that $val(M_0^3) < opt_0$.*

Proof: We prove that by presenting an example.
Consider the graph presented in Figure 51. According to Algorithm $XDZ4-UN$, using the notations of the Algorithm, $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})} = (v^1_1, v^4_4), (v^1_2, v^4_3), (v^1_3, v^4_2), (v^1_4, v^4_1)$.
$F^1_{(\mathcal{Y},0),(0,\mathcal{Z})} = (v^2_1, v^3_4), (v^2_2, v^3_1)$.
$F_2 = $.
And $F_3 = (v^3_2, v^3_3)$.
In that case, $M^3_0 = F^1_{(0,0),(\mathcal{Y},\mathcal{Z})} \cup F^1_{(\mathcal{Y},0),(0,\mathcal{Z})} \cup F_2 \cup F_3 = (v^1_1, v^4_4), (v^1_2, v^4_3), (v^1_3, v^4_2), (v^1_4, v^4_1), (v^2_1, v^3_4), (v^2_2, v^3_1), (v^3_2, v^3_3)$.
Therefore, $Y[M^3_0] = 6\mathcal{Y}$, $Z[M^3_0] = 6\mathcal{Z}$. According to Lemma 7.6, $Yopt = min\{(t_{(0,0)} + t_{(0,\mathcal{Z})}), (t_{(\mathcal{Y},0)} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Y}$. Then $Yopt = 6\mathcal{Y}$. Furthermore, according to Lemma 7.7, $Zopt = min\{(t_{(0,0)} + t_{(\mathcal{Y},0)}), (t_{(0,\mathcal{Z})} + t_{(\mathcal{Y},\mathcal{Z})})\}\mathcal{Z}$. Then $Zopt = 6\mathcal{Z}$. Therefore, $Y[M^3_0] = Yopt$ and $Z[M^3_0] = Zopt$.
If edge $e = (v^3_2, v^3_3)$ is $X - disjoint$ with edge $e' = (v^1_3, v^4_2)$, there are two possible swaps $f,\ f'$ where $f = (v^3_2, v^4_2)$, $f' = (v^1_3, v^3_3)$ or $f = (v^3_2, v^1_3)$, $f' = (v^3_3, v^4_2)$. In both cases, $X[e,e'] < X[f,f']$, $Y[e,e'] = Y[f,f'] = \mathcal{Y}$, $Z[e,e'] = Z[f,f'] = \mathcal{Z}$.
Therefore M^3_0 is not the maximum matching although it has the optimal Y and Z values. ∎

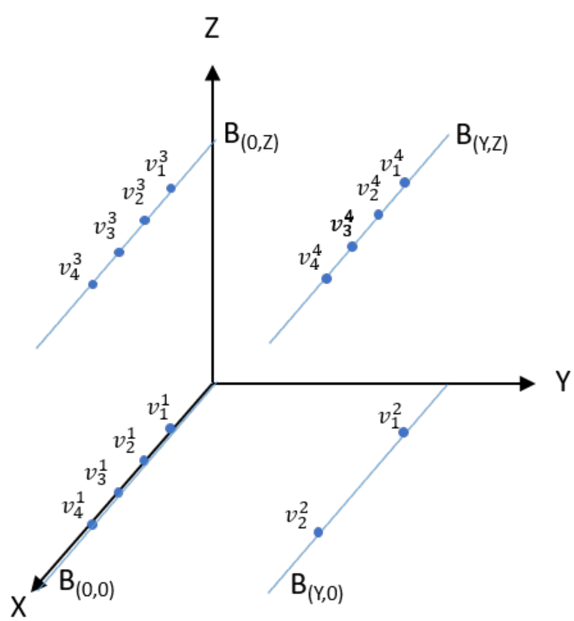

Figure 51: $G^{\triangledown}_4$ for example

Corollary 7.25 *Given $G^{\triangledown}_4$ a four unbalanced banks graph. $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}, F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}, F_2, F_3$ are each $Xopt$ on the corresponding sets of vertices, and therefore do not contain a set of two edges which are $X - disjoint$ and are not on the same bank.*

7.5 Phase 4

Lemma 7.26 *Given $G^{\triangledown}_4$ a four unbalanced banks graph. In Phase 4 of Algorithm $XDZ4-UN$. Let $e,\ e'$ be a $X - disjoint$ edges. Every $f,\ f'$ created from a swapped $e,\ e'$ satisfy $X[e,e'] < X[f,f']$, $Y[e,e'] = Y[f,f']$, $Z[e,e'] = Z[f,f']$. Thus this is a $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap.*

Proof: According to Lemma 4.19, $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ and $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ do not contain $X-disjoint$ edges.
According to Remark 7.20, F_2 is combined of a set of edges which is a result of Algorithm $X-Diagonals$ on two banks on one plane. Then, according to Lemma 4.19, F_2 does not contain $X-disjoint$ edges.
Furthermore, according to Remark 7.20, F_3 is combined from a set of edges which is a result of Algorithm One-Bank-Match on one bank. Therefore, according to Lemma 4.18 and Lemma 4.13, F_3 does not contain $X-disjoint$ edges.
Therefore, $X-disjoint$ edges may exist only between edges from two different sets of edges among $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}, F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}, F_2$ and F_3.
Let $e = e_i,\ e' = e_{n-i+1}$. Since $e,\ e'$ are $X-disjoint$, then for any swap $X[e,e'] < X[f,f']$.
In Phase 4 of Algorithm $XDZ4-UN$, every $e,\ e'$ which are being swapped are not both of the big diagonals. Therefore $e,\ e'$ satisfy one of the followings:

- e is in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ or $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ and edge e' is in F_2. In this case, according to Lemma 5.9, there exists a swap $f,\ f'$ such that $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$.

- e is in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ or $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ and edge e' is in F_3. In this case, according to Lemma 5.10, there exists a swap $f,\ f'$ such that $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$.

- e is in F_2 and edge e' is in F_3. In this case, according to Lemma 5.11, there exists a swap $f,\ f'$ such that $Y[e,e'] = Y[f,f']$ and $Z[e,e'] = Z[f,f']$.

Therefore, in Phase 4 of Algorithm $XDZ4-UN$, every $f,\ f'$ created from a swap on $e,\ e'$, satisfy $X[e,e'] < X[f,f']$, $Y[e,e'] = Y[f,f']$, $Z[e,e'] = Z[f,f']$. ■

Corollary 7.27 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. $Y[M_0^4] = Yopt$.*

Proof: According to Lemma 7.21, $Y[M_0^3] = Yopt$. Since in Phase 4 of Algorithm $XDZ4-UN$, all the swaps are $Y-Preserving$ swaps, then for each $f,\ f'$ a resulted edges of such a swap $Y[f,f'] = 0$. Therefore, $Y[M_0^4] = Yopt$. ■

Corollary 7.28 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. $Z[M_0^4] = Zopt$.*

Corollary 7.29 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. $Y[M_0^4] = Yopt$ and $Z[M_0^4] = Zopt$.*

7.6 Phase 5

Lemma 7.30 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. Let M be the matching which is a result of Algorithm $XDZ4-UN$. In every iteration $0 \le j \le p$ of Phase 5 of Algorithm $XDZ4-UN$, edges $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ are on different diagonals. The vertices of one of the edges are on $B_{(0,0)}$ and $B_{(\mathcal{Y},\mathcal{Z})}$, and the vertices of the second edge are on $B_{(\mathcal{Y},0)}$ and $B_{(0,\mathcal{Z})}$.*

Proof: In Phase 4 of Algorithm $XDZ4-UN$, all the edges which are $X-disjoint$ and not both on the diagonals are swapped. Therefore, in Phase 5 of Algorithm $XDZ4-UN$, all the edges which are $X-disjoint$ are both on different diagonals. ■

Lemma 7.31 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. Let M be a matching obtained during the process of Algorithm $XDZ4-UN$. Every two edges e_{Δ_M}, e'_{Δ_M} that are being swapped to f, f' in Phase 5 of Algorithm $XDZ4-UN$ satisfy that both are big-Diagonal edges, otherwise the edges have been swapped in Phase 4.*
Furthermore, e, e' satisfy $Y[e,e'] = Y[f,f'] = 2\mathcal{Y}$, $Z[e,e'] = 2\mathcal{Z}$, $Z[f,f'] = 0$, and $X[f,f'] = X[e,e'] + 2\Delta_M$.

Proof: Let f_1, f_1' and f_2, f_2' be the two possible sets of edges created by an $X-Improving$ swap on e_{Δ_M}, e'_{Δ_M}. Therefore, one of the two possible swaps, denoted by f, f', yields $Y[f_1, f_1'] = 2\mathcal{Y}$ and $Z[f_1, f_1'] = 0$.
By definition, e_{Δ_M}, e'_{Δ_M} are the farthest $X-disjoint$ edges and according to Lemma 4.1, the swapped edges satisfy $X[f,f] = X[e_{\Delta_M}, e'_{\Delta_M}] + 2\Delta_M$.
Therefore, the $Y-Preserving$ swap satisfies that $Y[e,e'] = Y[f,f'] = 2\mathcal{Y}$, $Z[e,e'] = 2\mathcal{Z}$, $Z[f,f'] = 0$, and $X[f,f'] = X[e,e'] + 2\Delta_M$. ■

Lemma 7.32 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every $0 \le j \le p$,*
$X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$.
$Y[M_{j+1}] = Y[M_j]$.
$Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$.

Proof: In every iteration in Phase 5 of Algorithm $XDZ4-UN-No-Delta$, the algorithm swaps two $X-disjoint$ edges $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ using a $Y-Preserving$ and $X-Improving$ swap, Denote that f, f' the edges created by the swap. These edges satisfy, according to Lemma 7.31, $Y[e,e'] = Y[f,f'] = 2\mathcal{Y}$, $Z[e,e'] = 2\mathcal{Z}$, $Z[f,f'] = 0$, $X[f,f'] = X[e,e'] + 2\Delta_M$.
Therefore, considering iteration $j+1$, $X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$, $Y[M_{j+1}] = Y[M_j]$, $Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$. ■

Corollary 7.33 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every $0 \le j \le p$,*
$X[M_j] = X[M_0^4] + \sum_{k=0}^{j-1} 2\Delta_{M_k}$.
$Y[M_j] = Y[M_0^4]$.
$Z[M_j] = Z[M_0^4] - 2j\mathcal{Z}$.

Corollary 7.34 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every $0 \le j \le p$,*
$xdz4_{j+1}^{\triangledown} = xdz4_j^{\triangledown} + 2\Delta_{M_j} - 2j\mathcal{Z}$.
$xdz4_p^{\triangledown} = xdz4_0^{\triangledown} + \sum_{k=0}^{p-1} 2\Delta_{M_k} - 2p\mathcal{Z}$.

Lemma 7.35 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. $xdz4_p^{\triangledown} = Xopt + Yopt + Zopt$ - $2p\mathcal{Z}$.*

Proof: According to Corollary 7.29, $Y[M_0^4] = Yopt$ and $Z[M_0^4] = Zopt$. According to Definition 7.16, in M_p there are no $X-disjoint$ edges. Therefore, according to Lemma 4.13, $X[M_p] = Xopt$. According to Corollary 7.33, $Y[M_p] = Y[M_0^4] = Yopt$, and $Z[M_p] = Z[M_0^4] - 2p\mathcal{Z} = Zopt - 2p\mathcal{Z}$. Therefore, $xdz4_p^{\triangledown} = Xopt + Yopt + Zopt$ - $2p\mathcal{Z}$. ■

Remark 7.36 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For $0 \le j \le p$,*
If $j \le \alpha$, then $\Delta_{M_j} > \mathcal{Z}$.
If $j > \alpha$, then $\Delta_{M_j} \le \mathcal{Z}$.

Lemma 7.37 $xdz4_{\alpha}^{\triangledown} = max\{xdz4_0^{\triangledown}, \dots, xdz4_p^{\triangledown}\}$.

Proof: According to Corollary 7.34, for $0 \le j \le p$, $xdz4_{j+1}^{\triangledown} = xdz4_j^{\triangledown} + 2\Delta_{M_j} - 2j\mathcal{Z}$. T herefore, $xdz4_j^{\triangledown} = xdz4_{j+1}^{\triangledown} - 2\Delta_{M_j} + 2j\mathcal{Z}$.
Therefore, for $j \le \alpha$, since by Remark 7.36, $\Delta_{M_j} > \mathcal{Z}$, it holds that $xdz4_{j+1}^{\triangledown} > xdz4_j^{\triangledown}$.
And for $j > \alpha$, since $\Delta_{M_j} \le \mathcal{Z}$, it holds that $xdz4_{j+1}^{\triangledown} \le xdz4_j^{\triangledown}$.
Hence, $xdz4_{\alpha}^{\triangledown} = max\{xdz4_0^{\triangledown}, \dots, xdz4_p^{\triangledown}\}$. ■

Lemma 7.38 *For every $0 \le j \le p$, and for every $M_j^o \in OPT_j$, $Y[M_j^o] = Y[M_j]$ and $Z[M_j^o] = Z[M_j]$.*

Proof: For every j, M_j^o is a maximum matching with exactly $l_Z + 2j$ edges with $Z - value = 0$, and l_Y edges with $Y - value = 0$. Therefore, $Z[M_j^o] = Zopt - 2j\mathcal{Z}$ and $Y[M_j^o] = Yopt$.
According to Corollary 7.33, $Z[M_j] = Z[M_0^4] - 2j\mathcal{Z}$, and $Y[M_j] = Y[M_0^4]$. Furthermore, according to Corollary 7.29, $Z[M_0^4] = Zopt$, and $Y[M_0^4] = Yopt$.
Therefore, $Z[M_j] = Zopt - 2j\mathcal{Z} = Z[M_j^o]$ and $Y[M_j] = Yopt = Y[M_j^o]$. ■

Lemma 7.39 *For every e, e', $X - disjoint$ edges in M_0^4, without loss of generality, one of the vertices of e is on one diagonal and in $S'(M_0^4)$ and one of the vertices of e' is on the other diagonal and in $S''(M_0^4)$.*

Proof: According to Lemma 7.30, e and e' are $X - disjoint$ edges on different diagonals. All the edges on the diagonals created by Algorithm $XDZ4 - UN$ are in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ and $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$. According to Corollary 7.25, there are no $X - disjoint$ edges in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ and there are no $X - disjoint$ edges in $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$. Therefore, all the edges with one vertex in $S'(M_0^4)$ are either in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ or in $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ and all the edges with one vertex in $S''(M_0^4)$ are in the matching. Without loss of generality, e is in $S'(M_0^4)$ and on one diagonal, e' is in $S''(M_0^4)$ and on the other diagonal. ■

Lemma 7.40 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every $0 \le j \le p$, edges $e_{\Delta_{M_j}}$ and $e'_{\Delta_{M_j}}$ contain the minimum $X - value$ vertex in $S'(M_j)$ and the maximum $X - value$ vertex in $S''(M_j)$.*

Proof: Since $e_{\Delta_{M_j}}$ and $e'_{\Delta_{M_j}}$ are $X - disjoint$, then according to Lemma 7.39, without loss of generality, v_j one of the vertices of $e_{\Delta_{M_j}}$ is in $S'(M_j)$ and v'_j one of the vertices of $e'_{\Delta_{M_j}}$ is in $S''(M_j)$. Since Δ_{M_j} is the maximum distance between two vertices, one in $S'(M_j)$ and the other in $S''(M_j)$, v_j and v'_j must be the vertices with the minimum $X - value$ vertex in $S'(M_j)$ and the maximum $X - value$ vertex in $S''(M_j)$. For example, see Figure 52. ■

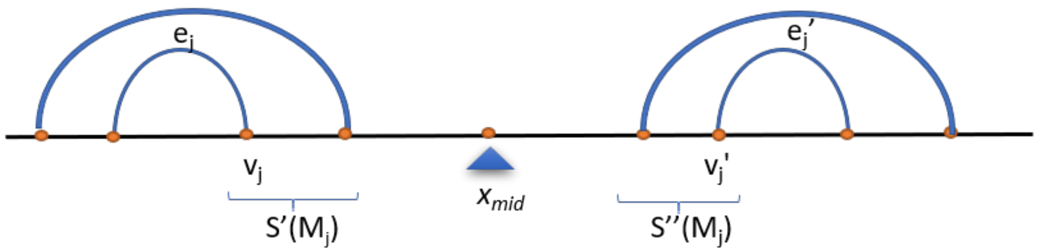

Figure 52: A matching with the farthest $X - disjoint$ e and e'

Lemma 7.41 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. Let M be the matching returned by Algorithm $XDZ4 - UN - No - Delta$. Let M' be the matching returned by the algorithm if we skip Phase 4 in Algorithm $XDZ4 - UN - No - Delta$. Then $M \equiv M'$.*

Proof: Denote by n the number of vertices in $S(M_0^3)$ for both Algorithm $XDZ4 - UN - No - Delta$ and Algorithm $XDZ4 - UN$.
Denote by $E_1 \in M_0^3$ the set of pairs of edges $(e_i, e_{n-i+1}), 1 \leq i \leq n$ on $S(M_0^3)$. According to Definition 7.5, not both (e_i, e_{n-i+1}) are Diagonal Edges. Denote E_2 the rest of edges in M_0^3, which are both Diagonal Edges.
For M returned by Algorithm $XDZ4 - UN - No - Delta$, the algorithm swaps first the edges in E_1 and in the next phase swaps all the edges in E_2.
For M' returned by Algorithm $XDZ4 - UN - No - Delta$ where skipping Phase 4, since according to Lemma 7.40, $e_{\Delta_{M_0^3}}$ and $e'_{\Delta_{M_0^3}}$ contain the minimum $X - value$ vertex in $S'(M_0^3)$ and the maximum $X - value$ vertex in $S''(M_0^3)$, then the algorithm swaps every two edges e_i, e_{n-i+1}.
Therefore, the exact same edges are swapped in E_1 and in E_2, although in different order.
Therefore, $M \equiv M'$, such that all edges of M and M' are the same edges. ∎

Lemma 7.42 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. During Phase 4 and 5 of Algorithm $XDZ4 - UN$, every pair of edges e, e' that are being swapped in these phases , the edges after the swap cross X_{mid}.*

Proof: Consider edges $e = (v_1, v_2)$ with $X[v_1] \leq X[v_2]$ and $e' = (u_1, u_2)$ with $X[u_1] \leq X[u_2]$. In every iteration of Phase 4 of Algorithm $XDZ4 - UN$, it follows that $v_2 \in S'(M)$ and $u_1 \in S''(M)$. Since v_1 and v_2 are on one side of X_{mid} and u_1 and u_2 are on the other side, then the two possible swaps $f_1 = (v_1, u_1), f_1' = (v_2, u_2)$ and $f_2 = (v_1, u_2), f_2' = (v_2, u_1)$ cross X_{mid}. ∎

Lemma 7.43 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every two matchings M and M' on $G_4^{\triangledown}$, with $S'(M) = S'(M')$ and $S''(M) = S''(M')$, the distance between the farthest $X - disjoint$ edges satisfies $\Delta_M = \Delta_{M'}$.*

Proof: Δ_M and $\Delta_{M'}$ are the distances on the X axis between the farthest $X - disjoint$ edges. Therefore, Δ_M is the distance between the minimum $X - value$ vertex in $S'(M)$ and the maximum $X - value$ vertex in $S''(M)$. Furthermore, $\Delta_{M'}$ is the distance between the minimum $X - value$ vertex in $S'(M')$ and the maximum $X - value$ vertex in $S''(M')$. Since $S'(M) = S'(M')$ and $S''(M) = S''(M')$, then $\Delta_M = \Delta_{M'}$ ∎

Lemma 7.44 *Given $G_4^{\triangledown}$ a four unbalanced banks graph. For every $0 \le j \le p$, for matching M_j, a $Y-Preserving$ and $X-Improving$ swap performed on $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$, results in edges that are not $X-disjoint$ with any other edge in M_j.*

Proof: Denote $e_{\Delta_{M_j}} = (v_1, v_2)$, $e'_{\Delta_{M_j}} = (u_1, u_2)$. According to Lemma 7.40, without loss of generality, we can assume that v_2 is the minimum $X-value$ vertex in $S'(M_j)$ and u_1 is the maximum $X-value$ vertex in $S''(M_j)$.
The possible swaps are $f_1 = (v_1, u_1)$ and $f'_1 = (v_2, u_2)$ or $f_2 = (v_1, u_2)$ and $f'_2 = (v_2, u_1)$. Since both options are $X-Improving$, choose $f,\ f'$ to be the $X-Improving$ and $Y-Preserving$ swap, see Figure 53.
Hence, edges $f,\ f'$ $X-cross$ all the vertices in $S'(M_j)$ and $S''(M_j)$, and thus they are not $X-disjoint$ with any edge which contains vertices from $S(M_j)$. Since edges $f,\ f'$ $X-cross$ X_{mid}, according to Lemma 4.7, edges $f,\ f'$ are not $X-disjoint$ with any edge that $X-cross$ X_{mid}. Therefore, edges $f,\ f'$ are not $X-disjoint$ with any other edge in M_j. ■

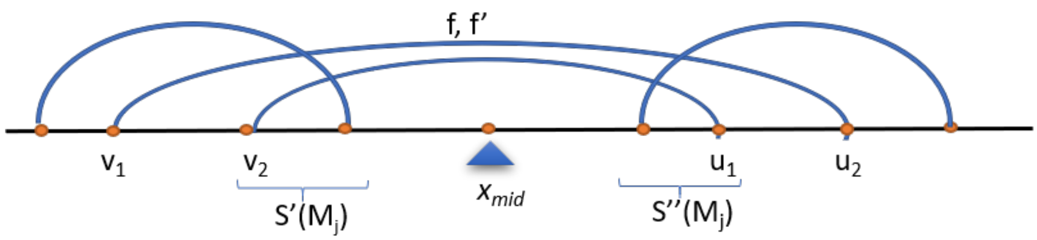

Figure 53: A matching after swapping to f and f'

Lemma 7.45 *For every matching $M_0^o \in OPT_0$, every pair of $X-disjoint$ edges reside on two Diagonal Edges on different diagonals.*

Proof: Since $M_0^o \in OPT_0$, M_0^o has the maximum $X-value$, where $Y[M_0^o] = Yopt$ and $Z[M_0^o] = Zopt$.
On a four banks graph there are Diagonal Edges, One Plane Edges and One Bank Edges. Assume by contradiction, that there are two $X-disjoint$ edges $f,\ f' \in M_0^o$, where f and f' are not both Diagonal Edges. The possible locations for f and f' are:

1. Edge f is a Diagonal Edge and edge f' is a One Plane Edge. According to Lemma 5.9, there exits a swap between Diagonal Edge and One Plane Edge that preserves $Y-value$ and $Z-value$. Since edges $f,\ f'$ are $X-disjoint$, according to Lemma 4.4, any swap on these edges improves $X-value$ and yields two edges which are not $X-disjoint$.

2. Edge f is a Diagonal Edge and edge f' is One Bank Edge. According to Lemma 5.10, there exits a swap between Diagonal Edge and One Bank Edge that preserves $Y-value$ and $Z-value$. Since edges $f,\ f'$ are $X-disjoint$, any swap on these edges improves $X-value$ and yields two edges which are not $X-disjoint$.

3. Edge f and edge f' are both One Plane Edges. According to Lemma 5.13, there exits a swap between two One Plane Edges that preserves or improves $Y-value$ and $Z-value$. Since

edges $f,\ f'$ are $X - disjoint$, according to Lemma 4.4, any swap on these edges improves $X - value$ and yields two edges which are not $X - disjoint$.

4. Edge f is a One Plane Edge and edge f' is One Bank Edge. According to Lemma 5.11, there exits a swap between One Plane Edge and One Bank Edge that preserves $Y - value$ and $Z - value$. Since edges $f,\ f'$ are $X - disjoint$, according to Lemma 4.4, any swap on these edges improves $X - value$ and yields two edges which are not $X - disjoint$.

5. Edge f and edge f' are both One Bank Edges. According to Lemma 5.14, there exits a swap between two One Bank Edges that preserves or improves $Y - value$ and $Z - value$. Since edges $f,\ f'$ are $X - disjoint$, according to Lemma 4.4, any swap on these edges improves $X - value$ and yields two edges which are not $X - disjoint$.

For all possible locations for f and f', after the swap we have a matching with a $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap, contradicting the assumption that $M_0^o \in OPT_0$.
Therefore, in M_0^o there are only $X - disjoint$ edges between Diagonal Edges on different diagonals.
■

Definition 7.46 *Given G_4^{∇} a four unbalanced banks graph, and a matching M on G_4^{∇}. For $S'(M) = v_1, \ldots, v_p$, $S''(M) = u_1, \ldots, u_p$, define $\Delta(S'(M), S''(M)) = \sum_{i=1}^{p} 2(X[u_{p-i+1}] - X[v_i])$.*

Lemma 7.47 *Given G_4^{∇} a four unbalanced banks graph, and a matching M on G_4^{∇}. $X[M] = Xopt - \Delta(S'(M), S''(M))$.*

Proof: $S(M)$ contains all $X - disjoint$ edges in M. All $X - disjoint$ edges in M are with one vertex in $S'(M)$ or in $S''(M)$. Performing all the swaps for all $X - disjoint$ edges in M, will result M' with no $X - disjoint$ edges and $X[M'] = Xopt$.
Each swap of two $X - disjoint$ edges in M, one with a vertex in $S'(M)$ and the other with a vertex in $S''(M)$, results in two edges which are both crossing X_{mid}. Therefore, according to Lemma 4.7, they are not $X - disjoint$. Each swap provides, for $1 \leq i, j \leq p$, $X - value$ of $2(X[u_j] - X[v_i])$.
Therefore, $X[M] = Xopt - \sum_{i=1}^{p} 2(X[u_{p-i+1}] - X[v_i]) = Xopt - \Delta(S'(M), S''(M))$. ■

Lemma 7.48 *For every matching $M_0^o \in OPT_0$, $S'(M_0^4) = S'(M_0^o)$ and $S''(M_0^4) = S''(M_0^o)$.*

Proof: Since $M_0^o \in OPT_0$, M_0^o has the maximum $X - value$ such that $Y[M_0^o] = Yopt$ and $Z[M_0^o] = Zopt$. According to Lemma 7.45, in M_0^o every pair of $X - disjoint$ edges reside on Diagonal Edges and on different diagonals. Therefore, one vertex is in $S'(M_0^o)$ and on one diagonal and the other vertex is in $S''(M_0^o)$ and on the other diagonal.
For M_0^4, according to Corollary 7.25, $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ and $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ are $Xopt$. According to Lemma 7.39, every pair of $X - disjoint$ edges has one edge in $F^1_{(0,0),(\mathcal{Y},\mathcal{Z})}$ and the other edge in $F^1_{(\mathcal{Y},0),(0,\mathcal{Z})}$ with one vertex in $S'(M_0^4)$ and on one diagonal and the other vertex is in $S''(M_0^4)$ and on the other diagonal.
According to Lemma 7.47, $X[M_0^o] = Xopt - \Delta(S'(M_0^o), S''(M_0^o))$ and $X[M_0^4] = Xopt - \Delta(S'(M_0^4), S''(M_0^4))$.
By the way Algorithm $XDZ4-UN$ is performed, M_0^4 has a minimum value of $\Delta(S'(M_0^4), S''(M_0^4))$, since the $X - disjoint$ edges are the inner edges on the diagonals.
Since $X[M_0^o]$ has the maximum $X - value$ for matching with $Y - value = Yopt$ and $Z - value = Zopt$, then $\Delta(S'(M_0^o), S''(M_0^o)) = \Delta(S'(M_0^4), S''(M_0^4))$ and therefore $S'(M_0^4) = S'(M_0^o)$ and $S''(M_0^4) =$

$S''(M_0^o)$.
■

Lemma 7.49 *For every* $M_0^o \in OPT_0$, $X[M_0^o] = X[M_0^4]$ *and* $xdz4_0^{\triangledown} = opt_0$.

Proof: According to Lemma 7.47, $X[M_0^o] = Xopt - \Delta(S'(M_0^o), S''(M_0^o))$ and $X[M_0^4] = Xopt - \Delta(S'(M_0^4), S''(M_0^4))$. According to Lemma 7.48, $S'(M_0^4) = S'(M_0^o)$ and $S''(M_0^4) = S''(M_0^o)$ and therefore $X[M_0^o] = X[M_0^4]$.
According to Lemma 7.27, $Y[M_0^4] = Yopt$ and according to Lemma 7.28, $Z[M_0^4] = Zopt$.
Therefore, $xdz4_0^{\triangledown} = opt_0$. ■

Lemma 7.50 *For every* $0 \leq j \leq p$, *any matching* $M_j^o \in OPT_j$ *satisfies* $S'(M_j) = S'(M_j^o)$ *and* $S''(M_j) = S''(M_j^o)$.

Proof: The proof of this lemma is by induction on j.
For $j = 0$, according to Lemma 7.48, $S'(M_0^4) = S'(M_0^o)$ and $S''(M_0^4) = S''(M_0^o)$.
Assume the lemma is correct for all $k < j$ and prove it for j.
Consider $M_j^o \in OPT_j$. According to Definition 7.1, M_j^o contains exactly $l_Z + 2j$ edges with $Z - value = 0$, and those $2j$ edges are the swapped edges in the algorithm.
Since there are $l_Z + 2j$ edges with $Z - value = 0$, then by enumeration argument there exist edges with $Z - value = 0$ on both Z-planes. Choose from the edges with $Z - value = 0$ two edges e, e', such that their X-distance is the minimum and e, e' are on different Z-planes.
Perform a $Z - Improving$ and $Y - Preserving$ swap on e and e', and let f, f' be the resulting edges. Denote the vertices of $f = (v, v'), f' = (u, u')$. Since f, f' are $X - disjoint$ edges then $X[v] < X[v'] \leq X[u] < X[u']$.
Since $Z[f] = Z[f'] = \mathcal{Z}$ and $Y[f] = Y[f'] = \mathcal{Y}$ then f and f' are Diagonal Edges.
Denote $\widehat{M_j^o} = M_j \setminus \{e, e'\} \cup \{f, f'\}$. Therefore, $S'(\widehat{M_j^o}) = S'(M_j) \setminus v'$ and $S''(\widehat{M_j^o}) = S''(M_j) \setminus u$.
In $\widehat{M_j^o}$ there are $l_Z + 2(j-1)$ edges with $Z - value = 0$ and l_Y edges with $Y - value = 0$.
$\widehat{M_j^o}$ is in OPT_{j-1}, otherwise we can perform a swap which improves its $X - value$ without changing its $Y - value$ or $Z - value$. But in this case, performing a similar swap in M_j^o which improves its $X - value$ without changing its $Y - value$ or $Z - value$, contradicts the assumption that $M_j^o \in OPT_j$.
Hence $\widehat{M_j^o} \in OPT_{j-1}$.
According to the induction assumption, $S'(\widehat{M_j^o}) = S'(M_{j-1})$ and $S''(\widehat{M_j^o}) = S''(M_{j-1})$. Therefore, according to Lemma 7.43, $\Delta_{M_{j-1}} = \Delta_{\widehat{M_j^o}}$. By definition, $\Delta_{M_{j-1}} = \Delta_{\widehat{M_j^o}}$ is the X-distance between f and f' and is also the X-distance between the vertices in $S'(M_{j-1})$ and $S''(M_{j-1})$ which are farthest from X_{mid}. According to Algorithm $XDZ4 - UN$, $S'(M_j) = S'(M_{j-1}) \setminus v'$ and $S''(M_j) = S''(M_{j-1}) \setminus u$. Therefore, $S'(M_j) = S'(M_j^o)$ and $S''(M_j) = S''(M_j^o)$. ■

Lemma 7.51 *For every index* $0 \leq j \leq p$, $opt_j = xdz4_j^{\triangledown}$.

Proof: Let $M_j^o \in OPT_j$. According to Lemma 7.50, $S'(M_j) = S'(M_j^o)$ and $S''(M_j) = S''(M_j^o)$. According to Lemma 7.47, $X[M_j] = Xopt - \Delta(S'(M_j), S''(M_j))$ and $X[M_j^o] = Xopt - \Delta(S'(M_j^o), S''(M_j^o))$.
Therefore, $X[M_j] = X[M_j^o]$.
According to Lemma 7.38, $Y[M_j] = Y[M_j^o]$ and $Z[M_j] = Z[M_j^o]$.
Therefore, $X[M_j] + Y[M_j] + Z[M_j] = X[M_j^o] + Y[M_j^o] + Z[M_j^o]$, and thus $xdz4_j^{\triangledown} = opt_j$. ■

Lemma 7.52 $opt_\alpha = max\{opt_0, \ldots, opt_p\}$.

Proof: According to Lemma 7.51, for every index $0 \leq j \leq p$, $opt_j = xdz4^{\triangledown}_j$. According to Lemma 7.37, $xdz4^{\triangledown}_\alpha = max\{xdz4^{\triangledown}_0, \ldots, xdz4^{\triangledown}_p\}$. Therefore, $opt_\alpha = max\{opt_0, \ldots, opt_p\}$. ■

Lemma 7.53 *Given $G^{\triangledown}_4$ a four unbalanced banks graph. Algorithm $XDZ4-UN$ on $G^{\triangledown}_4$ is finitely determined.*

Proof: Since Algorithm $XDZ4-UN-No-Delta$ is finitely determined after p iterations, and according to Remark 7.15, all the operations of Algorithm $XDZ4-UN$ are contained in the first operations of Algorithm $XDZ4-UN-No-Delta$, then Algorithm $XDZ4-UN$ is finitely determined. ■

Theorem 7.54 *Given $G^{\triangledown}_4$ a four unbalanced banks graph. Algorithm $XDZ4-UN$ on $G^{\triangledown}_4$ is finitely determined and returns a maximum matching.*

Proof: According to Lemma 7.53, Algorithm $XDZ4-UN$ finitely determined. According to Definition 7.16, the result of the algorithm is a matching M which satisfies $val(M) = xdz4^{\triangledown}_\alpha$. According to Lemma 7.51, $xdz4^{\triangledown}_\alpha = opt_\alpha$ and according to Lemma 7.52, $opt = opt_\alpha$. Hence, the algorithm returns a maximum value matching. ■

8 Six Balanced Banks Matching

In this section we consider matchings on a graph which contains t vertices on each one of the six banks. We introduce basic properties concerning this case, followed by an algorithm which finds a local maximum matching so that the algorithm is finitely determined and there is no simple swap for two edges that can improve its value.

In all the algorithms assume the graph has two Z planes, $P_{Z=0}, P_{Z=\mathcal{Z}}$, and three Y planes, $P_{Y=0}, P_{Y=\mathcal{Y}}, P_{Y=2\mathcal{Y}}$, and $\mathcal{Y} \geq \mathcal{Z}$. Otherwise, change the roles of Y axis and Z axis.

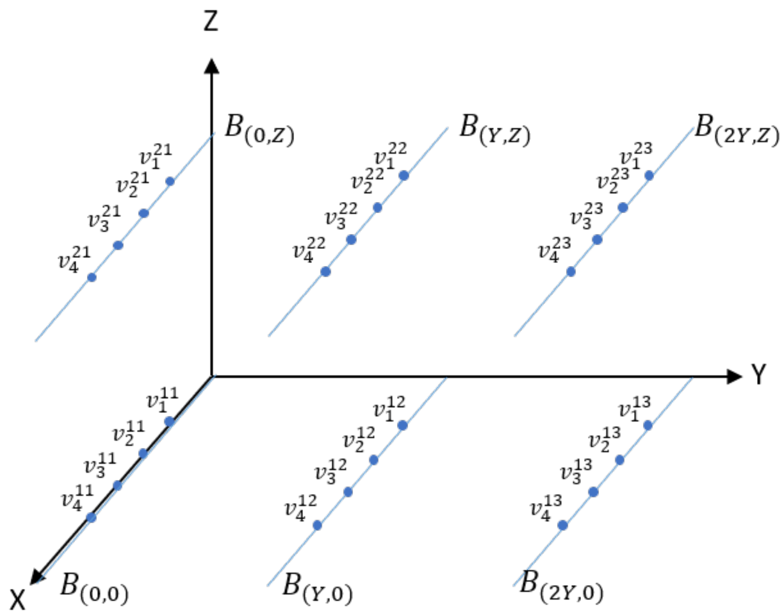

Figure 54: six balanced banks graph

Definition 8.1 *$G_6 = (V, E)$ is a* **six balanced banks graph** *with t vertices on each bank.* $V = \{v_1^{11}, \ldots, v_t^{11}, v_1^{12}, \ldots, v_t^{12}, v_1^{13}, \ldots, v_t^{13}, v_1^{21}, \ldots, v_t^{21}, v_1^{22}, \ldots, v_t^{22}, v_1^{23}, \ldots, v_t^{23}\}$.
The vertices on bank $\boldsymbol{B_{ij}}$ are $v_1^{ij}, \ldots, v_t^{ij}$ and ordered according to their X-coordinate.

Definition 8.2 *Define $\boldsymbol{P_{Y=0}}$ to be the plane that contains all the banks with $Y - value = 0$. For six balanced banks graph the plane $P_{Y=0}$ contains $B_{(0,0)}$ and $B_{(0,\mathcal{Z})}$*

Definition 8.3 *Define $\boldsymbol{P_{Y=\mathcal{Y}}}$ to be the plane that contains all the banks with $Y - value = \mathcal{Y}$. For six balanced banks graph the plane $P_{Y=\mathcal{Y}}$ contains $B_{(\mathcal{Y},0)}$ and $B_{(\mathcal{Y},\mathcal{Z})}$*

Definition 8.4 *Define $\boldsymbol{P_{Y=2\mathcal{Y}}}$ to be the plane that contains all the banks with $Y - value = 2\mathcal{Y}$. For six balanced banks graph the plane $P_{Y=2\mathcal{Y}}$ contains $B_{(2\mathcal{Y},0)}$ and $B_{(2\mathcal{Y},\mathcal{Z})}$.*

Definition 8.5 *Define $\boldsymbol{P_{Z=0}}$ to be the plane that contains all the banks with $Z - value = 0$. For six balanced banks graph the plane $P_{Z=0}$ contains $B_{(0,0)}$, $B_{(\mathcal{Y},0)}$ and $B_{(2\mathcal{Y},0)}$.*

Definition 8.6 *Define $\boldsymbol{P_{Z=\mathcal{Z}}}$ to be the plane that contains all the banks with $Z - value = \mathcal{Z}$. For six balanced banks graph the plane $P_{Z=\mathcal{Z}}$ contains $B_{(0,\mathcal{Z})}$, $B_{(\mathcal{Y},\mathcal{Z})}$ and $B_{(2\mathcal{Y},\mathcal{Z})}$.*

Definition 8.7 *Define edge e to be a* **Big-Diagonal** *Edge if one vertex of e is in $B_{(0,0)}$ and the other is in $B_{(2\mathcal{Y},\mathcal{Z})}$ or if one vertex of e is in $B_{(2\mathcal{Y},0)}$ and the other is in $B_{(0,\mathcal{Z})}$. For example, see Figure 55.*

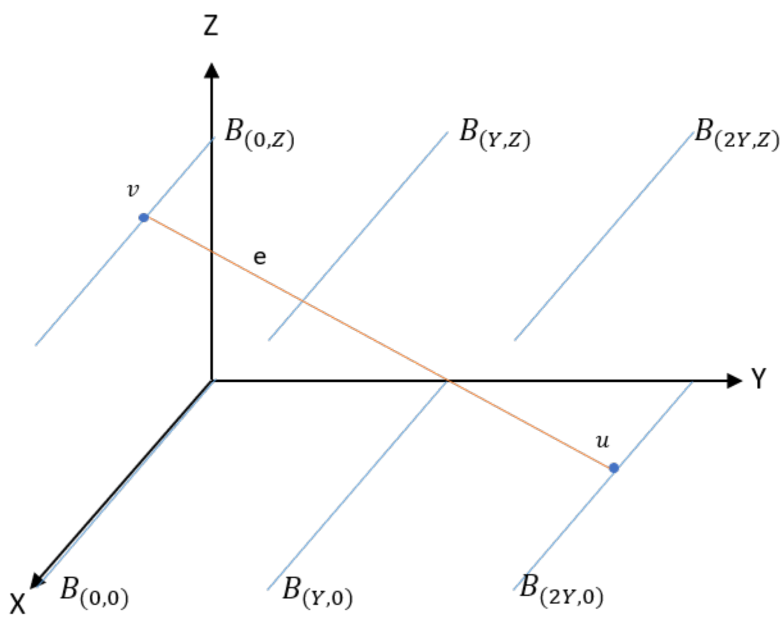

Figure 55: Big Diagonal edge example

Definition 8.8 *For an edge $e = (v, u)$ and a set of vertices U, define that e* ***touches*** *U if $v \in U$ or $u \in U$.*

8.1 Good Swaps

Definition 8.9 *Define a* **conjoint-edges** *to be a set of four edges* (e_1, e_2, e_3, e_4) *such that*
$e_1 = (v_1, v'_1)$, $v_1 \in B_{(\mathcal{Y},0)}$, $v'_1 \in B_{(\mathcal{Y},\mathcal{Z})}$, *and satisfies* $v_1 \leq X_{mid} \leq v'_1$.
$e_2 = (v_2, v'_2)$, $v_2 \in B_{(\mathcal{Y},0)}$, $v'_2 \in B_{(\mathcal{Y},\mathcal{Z})}$, *and satisfies* $v'_2 \leq X_{mid} \leq v_2$.
$e_3 = (v_3, v'_3)$, $v_3 \in B_{(0,0)}$, $v'_3 \in B_{(2\mathcal{Y},\mathcal{Z})}$.
$e_4 = (v_4, v'_4)$, $v_4 \in B_{(2\mathcal{Y},0)}$, $v'_4 \in B_{(0,\mathcal{Z})}$.
Such that e_3 and e_4 are X-disjoint and on different sides in respect of X_{mid}. see Figures 56, 57.

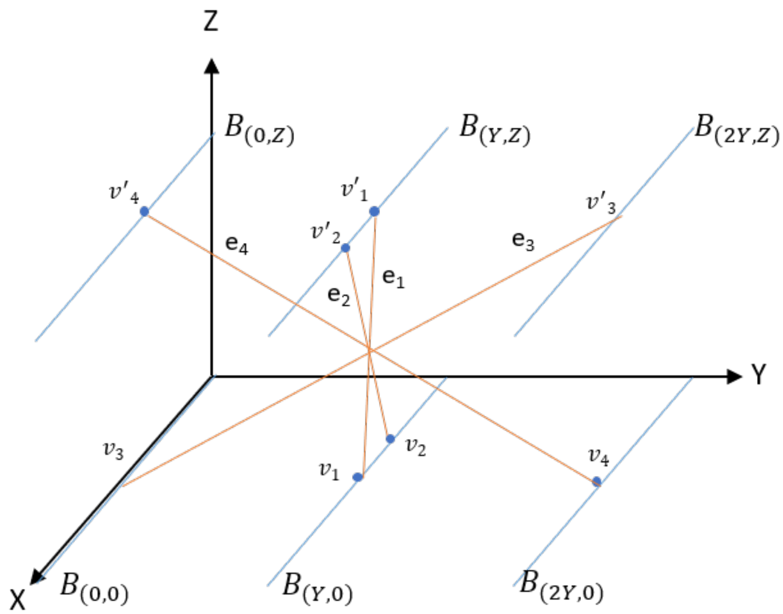

Figure 56: $e_1,\ e_2,\ e_3,\ e_4$

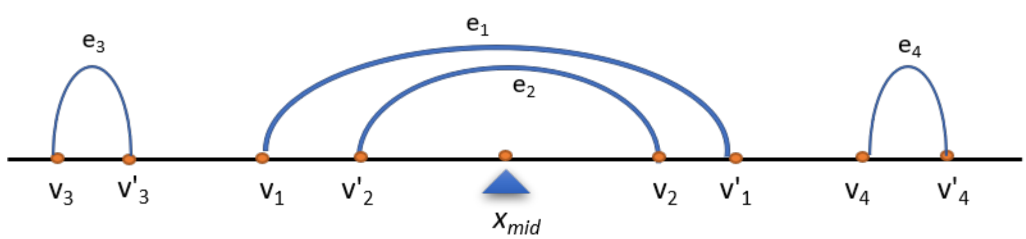

Figure 57: $e_1,\ e_2,\ e_3,\ e_4$ on X-axis

Definition 8.10 *Define a* **conjoint-swap** *to be a swap of four conjoint-edges* $e_1 = (v_1, v_1'),\ e_2 = (v_2, v_2'),\ e_3 = (v_3, v_3'),\ e_4 = (v_4, v_4')$ *using the notation of Definition 8.9, the edges are defined to be* $f_1 = (v_3, v_1'),\ f_2 = (v_2, v_3'),\ f_3 = (v_3, v_4'),\ f_4 = (v_4, v_2')$*. See Figures 58, 59.*

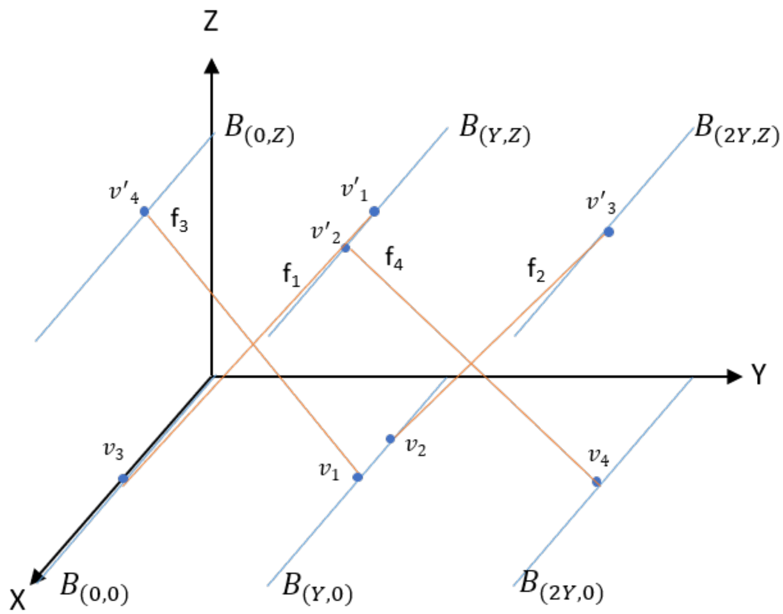

Figure 58: $f_1,\ f_2,\ f_3,\ f_4$

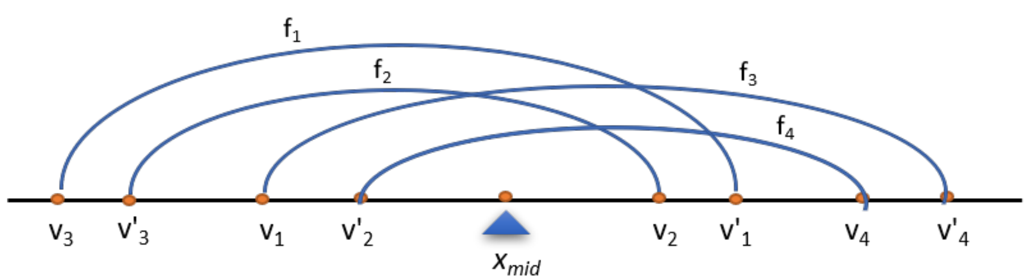

Figure 59: $f_1,\ f_2,\ f_3,\ f_4$ on X-axis

Lemma 8.11 *A conjoint-swap exists such that the swap is $Y-Preserving$, $Z-Preserving$ and $X-Improving$.*

Proof: The set of the conjoint-edges e_1, e_2, e_3, e_4 and the set of the swapped edges f_1, f_2, f_3, f_4 provide the values such that:
The $Y-value$ of e_1, e_2, e_3, e_4 is $Y[e_1] = 0$, $Y[e_2] = 0$, $Y[e_3] = 2\mathcal{Y}$, $Y[e_4] = 2\mathcal{Y}$ and in total, $Y[e_1, e_2, e_3, e_4] = 4\mathcal{Y}$. The $Y-value$ of the swapped edges is $Y[f_1] = \mathcal{Y}$, $Y[f_2] = \mathcal{Y}$, $Y[f_3] = \mathcal{Y}$, $Y[f_4] = \mathcal{Y}$ and in total $Y[f_1, f_2, f_3, f_4] = 4\mathcal{Y}$. Therefore, the conjoint-swap which is $Y-preserving$.
The $Z-value$ of e_1, e_2, e_3, e_4 is $Z[e_1] = \mathcal{Z}$, $Z[e_2] = \mathcal{Z}$, $Z[e_3] = \mathcal{Z}$, $Z[e_4] = \mathcal{Z}$ and in total, $Z[e_1, e_2, e_3, e_4] = 4\mathcal{Z}$. The $Z-value$ of the swapped edges is $Z[f_1] = \mathcal{Z}$, $Z[f_2] = \mathcal{Z}$, $Z[f_3] = \mathcal{Z}$, $Z[f_4] = \mathcal{Z}$ and in total $Z[f_1, f_2, f_3, f_4] = 4\mathcal{Z}$. Therefore, there exist a conjoint-swap which is $Z-Preserving$.
Concerning the $X-value$, since e_1, e_2 cross X_{mid} and e_3, e_4 are X-disjoint and on different sides in respect to X_{mid} and since f_1, f_2, f_3, f_4 are all crossing X_{mid} then the swap is $X-Improving$. ∎

Lemma 8.12 *Given G_6 a six balanced banks graph and a set of four conjoint-edges e_1, e_2, e_3, e_4. There exists a conjoint-swap which yields $f_1,\ f_2,\ f_3,\ f_4$ such that $Y[e_1\ , e_2\ , e_3\ , e_4] = Y[f_1,\ f_2,\ f_3,\ f_4]$ and $Z[e_1, e_2, e_3, e_4] = Z[f_1,\ f_2,\ f_3,\ f_4]$.*

Proof: Given conjoint-edges $e_1,\ e_2,\ e_3,\ e_4$, see Figure 56, a conjoint-swap can be perform on all four edges such that $f_1 = (v_1', v_3)$, $f_2 = (v_2, v_3')$, $f_3 = (v_1, v_4')$, $f_4 = (v_2', v_4)$, see Figure 58.
In that case, $Y[e_1\ , e_2\ , e_3\ , e_4] = 4\mathcal{Y} = Y[f_1,\ f_2,\ f_3,\ f_4]$ and $Z[e_1\ , e_2\ , e_3\ , e_4] = 4\mathcal{Z} = Z[f_1,\ f_2,\ f_3,\ f_4]$. ∎

Lemma 8.13 *Given G_6 a six balanced banks graph. For a set of four conjoint-edges e_1, e_2, e_3, e_4. A conjoint-swap yields $f_1,\ f_2,\ f_3,\ f_4$ such that $f_1,\ f_2,\ f_3,\ f_4$ $X-cross\ X_{mid}$.*

Proof: Given the conjoint-edges $e_1,\ e_2,\ e_3,\ e_4$, see Figure 56, a conjoint-swap performed on all four edges satisfies $f_1 = (v_1', v_3)$, $f_2 = (v_2, v_3')$, $f_3 = (v_1, v_4')$, $f_4 = (v_2', v_4)$.
In that case, the edges $f_1,\ f_2,\ f_3,\ f_4$ all $X - cross\ X_{mid}$, see Figure 59. ∎

Lemma 8.14 *Given G_6 a six balanced banks graph. For two edges $e = (v_1, v_2)$, $e' = (u_1, u_2)$ that satisfy $v_1 \in B_{(\mathcal{Y},0)}$, $v_2 \in B_{(\mathcal{Y},\mathcal{Z})}$, $u_1 \in B_{(0,0)}$, $u_2 \in B_{(2\mathcal{Y},\mathcal{Z})}$. There exists a swap which yields edges $f,\ f'$ such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.*

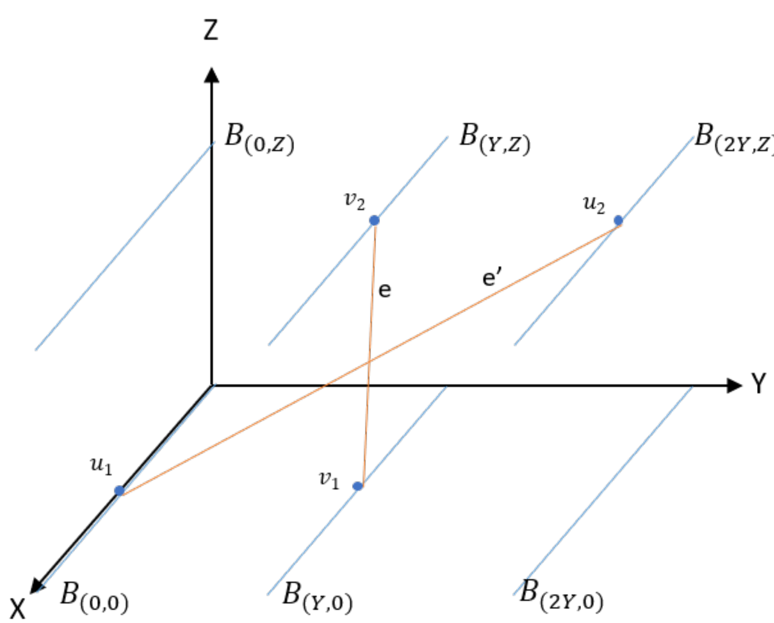

Figure 60: e and e' of Lemma 8.14

Proof: Choose $f = (v_1, u_2)$ and $f' = (u_1, v_2)$. These edges satisfy $Y[f,\ f'] = Y[e,\ e'] = 2\mathcal{Y}$, $Z[f,\ f'] = Z[e,\ e'] = 2\mathcal{Z}$, see Figure 61. ∎

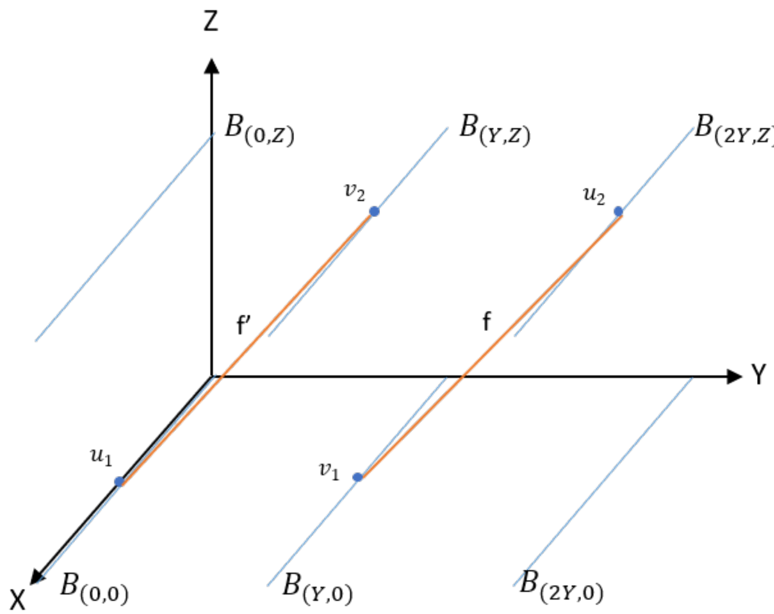

Figure 61: f and f'

Corollary 8.15 *Given G_6 a six balanced banks graph. For two edges $e = (v_1, v_2)$, $e' = (u_1, u_2)$ that satisfy $v_1 \in B_{(\mathcal{Y},0)}$, $v_2 \in B_{(\mathcal{Y},\mathcal{Z})}$, $u_1 \in B_{(2\mathcal{Y},0)}$, $u_2 \in B_{(0,\mathcal{Z})}$. There exists a swap which yields $f,\ f'$ such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.*

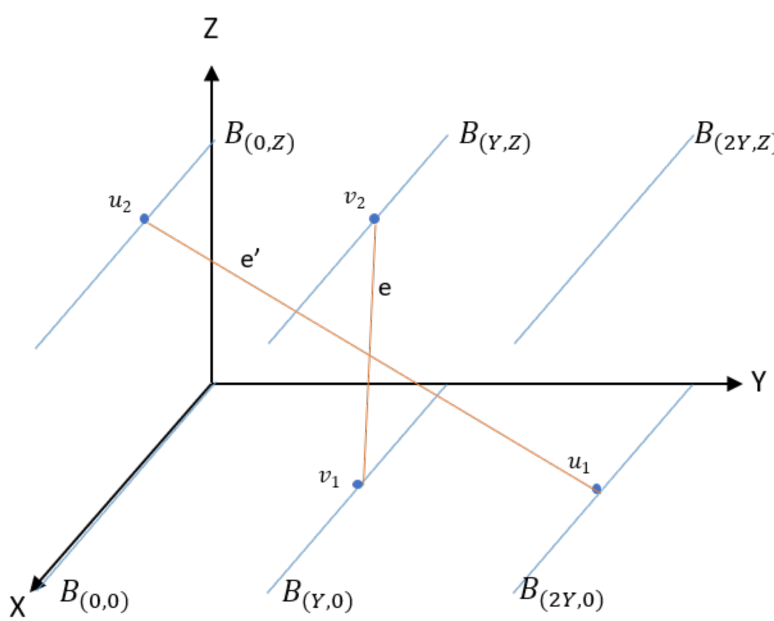

Figure 62: e and e' of Corollary 8.15

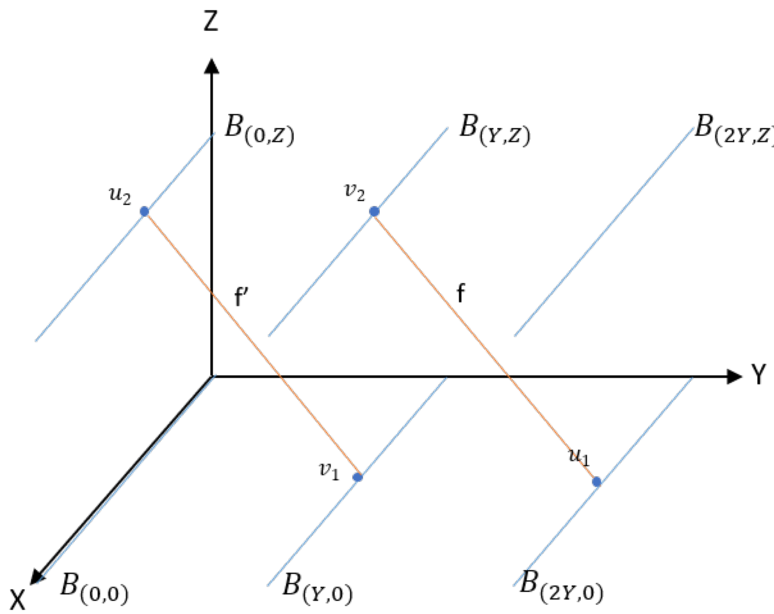

Figure 63: f and f' of Corollary 8.15

Corollary 8.16 *Given G_6 a six balanced banks graph. For two edges $e = (v_1, v_2)$, $e' = (u_1, u_2)$ that satisfy $v_1 \in B_{(\mathcal{Y},0)}$, $v_2 \in B_{(2\mathcal{Y},\mathcal{Z})}$, $u_1 \in B_{(0,0)}$, $u_2 \in B_{(\mathcal{Y},\mathcal{Z})}$. There exists a swap which yields $f,\ f'$ such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.*

Corollary 8.17 *Given G_6 a six balanced banks graph. For two edges $e = (v_1, v_2)$, $e' = (u_1, u_2)$ that satisfy $v_1 \in B_{(\mathcal{Y},0)}$, $v_2 \in B_{(0,\mathcal{Z})}$, $u_1 \in B_{(2\mathcal{Y},0)}$, $u_2 \in B_{(\mathcal{Y},\mathcal{Z})}$. There exists a swap which yields $f,\ f'$ such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.*

Corollary 8.18 *Given G_6 a six balanced banks graph. For two $X-disjoint$ edges $e,\ e'$ such that $e \in B_{(\mathcal{Y},0)}, B_{(\mathcal{Y},\mathcal{Z})}$ and e' is either in $B_{(0,0)}, B_{(2\mathcal{Y},\mathcal{Z})}$ or in $B_{(2\mathcal{Y},0)}, B_{(0,\mathcal{Z})}$, there exists a $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap.*

Corollary 8.19 *Given G_6 a six balanced banks graph. For two $X - disjoint$ edges $e,\ e'$, if e, e' are not both Big-Diagonals Edges on different diagonals, then there exists a $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap.*

8.2 Optimal Values

Lemma 8.20 *Given G_6 a six balanced banks graph with t vertices on each bank. If a matching M on G_6 is $Yopt$ then $Y[M] = 4\mathcal{Y}t$. Furthermore, if M is $Yopt$ then M contains at most t edges e, such that $Y[e] = 0$ and all those edges are on $P_{Y=\mathcal{Y}}$.*

Proof: Suppose by contradiction there is a matching M which is $Yopt$ and contains an edge e with $Y[e] = 0$ on a plane which is not $P_{Y=\mathcal{Y}}$, without loss of generality, suppose edge e is on plane $P_{Y=0}$. By enumeration argument, there is at least one other edge e' not on plane $P_{Y=0}$.
There are three options for edge e':

1. $e' \in P_{Y=\mathcal{Y}}$.
2. $e' \in P_{Y=2\mathcal{Y}}$
3. e' is connecting a vertex from $P_{Y=\mathcal{Y}}$ and a vertex from $P_{Y=2\mathcal{Y}}$

In all three cases, there is a swap of $e,\ e'$ that improves the $Y-value$, contradicting the assumption that M is $Yopt$. Therefore all the edges with $Y-value=0$ are on plane $P_{Y=\mathcal{Y}}$.
Since plane $P_{Y=\mathcal{Y}}$ contains $2t$ vertices there are at most t edges with $Y-value=0$.
Let l_1 to be the number of edges with $Y-value=0$ on plane $P_{Y=\mathcal{Y}}$.
Let l_2 to be the number of edges with $Y-value=\mathcal{Y}$ with one vertex on plane $P_{Y=0}$ and one vertex on plane $P_{Y=\mathcal{Y}}$.
Let l_3 to be the number of edges with $Y-value=\mathcal{Y}$ with one vertex on plane $P_{Y=\mathcal{Y}}$ and one vertex on plane $P_{Y=2\mathcal{Y}}$.
The rest of the edges are with one vertex on $P_{Y=0}$ and the other vertex on $P_{Y=2\mathcal{Y}}$, denote their number as l_4.
Since there are $2t$ vertices on $P_{Y=2\mathcal{Y}}$, $2t=l_3+l_4$.
Since there are $2t$ vertices on $P_{Y=0}$, $2t=l_2+l_4$.
Therefore, $l_2=l_3$ and $l_4=2t-l_2$.
Hence, $Y[M]=0l_1+\mathcal{Y}(l_2+l_3)+2\mathcal{Y}(2t-l_2)=2l_2\mathcal{Y}+4t\mathcal{Y}-2l_2\mathcal{Y}=4t\mathcal{Y}$. ■

Corollary 8.21 *Given G_6 a six balanced banks graph with t vertices on each bank. A matching M with $Y[M]=4t\mathcal{Y}$ is $Yopt$.*

Lemma 8.22 *Given G_6 a six balanced banks graph with t vertices on each bank and let M be a matching on G_6. If a matching M is **not** $Yopt$ then there is at least one edge e with $Y[e]=0$ on $P_{Y=0}$ or on $P_{Y=2\mathcal{Y}}$.*

Proof: In G_6 there are two banks on each Y plane, and therefore, there are $2t$ vertices on each Y plane $P_{Y=0}, P_{Y=\mathcal{Y}}$, and $P_{Y=2\mathcal{Y}}$.
Let l_1 be the number of edges with $Y-value=0$ on plane $P_{Y=\mathcal{Y}}$.
Let l_2 be the number of edges with $Y-value=\mathcal{Y}$ with one vertex on plane $P_{Y=0}$ and one vertex on plane $P_{Y=\mathcal{Y}}$.
Let l_3 be the number of edges with $Y-value=2\mathcal{Y}$ with one vertex on plane $P_{Y=0}$ and one vertex on plane $P_{Y=2\mathcal{Y}}$.
Let l_4 be the number of edges with $Y-value=\mathcal{Y}$ with one vertex on plane $P_{Y=Y}$ and one vertex on plane $P_{Y=2\mathcal{Y}}$.
Without loss of generality suppose that all the edges in M whose $Y-value=0$ are only on plane $P_{Y=\mathcal{Y}}$.
Since there are $2t$ vertices on plane $P_{Y=2\mathcal{Y}}$, then there are $2t-l_2$ edges with one vertex on plane $P_{Y=2\mathcal{Y}}$ and the other vertex on plane $P_{Y=\mathcal{Y}}$ and since $l_1+l_2=2t$ then $Y[M]=l_1\cdot 0+2l_2\mathcal{Y}+(2t-l_2)2\mathcal{Y}=4t\mathcal{Y}$.
According to Lemma 8.20, the $Y-value$ of a $Yopt$ matching is $4t\mathcal{Y}$, in contradiction to the assumption of the lemma that M is not $Yopt$.
■

Lemma 8.23 *Given G_6 a six balanced banks graph with t vertices on each bank. If a matching M on G_6 is $Zopt$ then $Z[M]=3\mathcal{Z}t$. Furthermore, if M is $Zopt$ then M does not contain any edge e with $Z[e]=0$.*

Proof: In G_6, the maximal $Z - value$ for each edge is $\mathcal{Z}$. Since G_6 contains $6t$ vertices, a full matching contains $3t$ edges. Therefore, $Zopt$ matching contains $3t$ edges even with the maximum $Z - value = \mathcal{Z}$, connecting a vertex from $P_{Z=0}$ and a vertex from $P_{Z=\mathcal{Z}}$, giving that $Z[M] = 3\mathcal{Z}t$.
■

Corollary 8.24 *Given G_6 a six balanced banks graph with t vertices on each bank. A matching M with $Z[M] = 3t\mathcal{Z}$ is $Zopt$.*

Lemma 8.25 *Given G_6 a six balanced banks graph with t vertices on each bank and let M be a matching on G_6. The number of edges with $Z - value = 0$ in M is even.*

Proof: In G_6 there are two Z-planes, with $3t$ vertices on each Z-plane.
For each edge f, if $Z[f] = \mathcal{Z}$ then f connects vertices on two different Z-planes and if $Z[f] = 0$ then f connects vertices on the same Z-plane.
Suppose there are l edges with $Z - value = 0$, on plane $P_{Z=0}$. In this case, there are $3t - 2l$ vertices on $P_{Z=0}$ which are matched to vertices on $P_{Z=\mathcal{Z}}$.
Since there are $3t$ vertices on each Z-plane, there are $2l$ vertices on plane $P_{Z=\mathcal{Z}}$, which are matched together, creating l edges on plane $P_{Z=\mathcal{Z}}$ with $Z - value = 0$.
All together, there are $2l$ edges with $Z - value = 0$. l on plane $P_{Z=0}$ and l on plane $P_{Z=\mathcal{Z}}$. ■

Lemma 8.26 *Given G_6 a six balanced banks graph. If $\mathcal{Y} > \mathcal{Z}$ then every maximum matching $M \in OPT$ is $Yopt$.*

Proof: Suppose by contradiction that M is a maximum matching on a six balanced banks graph which is not $Yopt$, then according to Lemma 8.22, it contains at least one edge e such that $Y[e] = 0$ and e is on plane $P_{Y=0}$ or on plane $P_{Y=2\mathcal{Y}}$, without loss of generality, suppose that $e \in P_{Y=0}$.
By enumeration argument, there is another edge e' which satisfies exactly one of the following possibilities: e' is on $P_{Y=\mathcal{Y}}$ or e' is on $P_{Y=2\mathcal{Y}}$ or e' has one vertex on $P_{Y=\mathcal{Y}}$ and another vertex on $P_{Y=2\mathcal{Y}}$.
There is a swap on $e,\ e'$ which yields edges $f,\ f'$ which are not $X - disjoint$. Giving that, $X[f,\ f'] \geq X[e,\ e']$. According to their location on the Y-planes, $Y[f,\ f'] \geq Y[e,\ e'] + 2\mathcal{Y}$.
Note that it is possible that $Z[f,\ f'] = Z[e,\ e'] - 2\mathcal{Z}$.
Let $M' = M \setminus \{e,\ e'\} \cup \{f,\ f'\}$. $Y[M'] \geq Y[M] + 2Y$, $Z[M'] \geq Z[M] - 2Z$ and $X[M'] \geq X[M]$.
Since $\mathcal{Y} > \mathcal{Z}$, $val(M) < val(M')$, contradicting the assumption that M is a maximum matching.
■

Lemma 8.27 *Given G_6 a six balanced banks graph. There exists $0 \leq i^* \leq p$ such that $opt = opt_{i^*}$.*

Proof: According to Lemma 8.26, every $M \in OPT$ on G_6 is $Yopt$. According to Corollary 8.25, the number of edges with $Z - value = 0$ in M is even, denote this number by $2i^*$.
Giving that, $Y[M] = Yopt$, $Z[M] = Zopt{-}2i^*\mathcal{Z}$. Therefore, $M \in OPT_{i^*}$ and $val(M) = opt_{i^*} = opt$.
■

Algorithm 5 XDZ6:

Finds a maximum perfect matching.

function XDZ6()

Input:

A graph $G = (V, E)$ with six balanced banks graph.

On bank i there are t vertices $v_1^i, \ldots, v_t^i$ ordered according to their X-coordinate

Assumptions $\mathcal{Y} > \mathcal{Z}$.

Output:

A maximum perfect matching M

begin

Phase 1:

$F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})} = X - Diagonals(B_{(0,0)}, B_{(2\mathcal{Y},\mathcal{Z})})$

$F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})} = X - Diagonals(B_{(2\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$

$F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})} = X - Diagonals(B_{(\mathcal{Y},0)}, B_{(\mathcal{Y},\mathcal{Z})})$

Initialize $M = F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})} \cup F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})} \cup F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$

Phase 2:

$L =$ *All the edges which touch* $S(M)$

$L' = L \cap F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ [Either all the edges in L' touch $S'(M)$ or all touch $S''(M)$, (See Corollary 8.40)].

Order the edges in L' $e_1, \ldots, e_n$, according to the order of the vertices in $S'(M)$ or $S''(M)$.

for $i = 1 \ldots n$:

Let e' be the farthest $X - disjoint$ edge from e_i.

Let $f,\ f'$ be the edges created by a $Y - Preserving$, $Z - Preserving$ and $X - Improving$ swap on $e_i,\ e'$ [According to Corollary 8.18 such a swap exists, and according to Corollary 8.19, e, e' are not both X-Diagonals Edges].

$M = M \setminus \{e_i,\ e'\} \cup \{f,\ f'\}$

end for

Phase 3:

while M contains four conjoint-edges $e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}$ such that Δ_M is the maximum in M [see Definitions 8.9, 3.13]. :

Let $f_1,\ f_2,\ f_3,\ f_4$ be the edges created by a conjoint-swap on $e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}$, [see Definition 8.10].

$M = M \setminus \{e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}\} \cup \{f_1,\ f_2,\ f_3,\ f_4\}$

end while

Phase 4:

while M contains $X - disjoint$ edges :

Find $e_{\Delta_M},\ e'_{\Delta_M}$ (see Definition 3.13)

while $\Delta_M > \mathcal{Z}$:

Let $f,\ f'$ be the edges created by $Y - Preserving$ and $X - Improving$ swap on $e_{\Delta_M},\ e'_{\Delta_M}$

$M = M \setminus \{e_{\Delta_M},\ e'_{\Delta_M}\} \cup \{f,\ f'\}$

Find $\{e_{\Delta_M},\ e'_{\Delta_M}\}$ [see Definition 3.13].

end while

end while return M

end function

Definition 8.28 *For Algorithm $XDZ6$, define* **XDZ6-No-Delta** *to be the algorithm in the case that Algorithm $XDZ6$ would not stop according to the stopping condition $\Delta_M > \mathcal{Z}$ in Phase 4, and finitely determined when M does not contain $X - disjoint$ edges.*

Remark 8.29 *Given G_6 a six balanced banks graph. All the matchings created by Algorithm $XDZ6$ and the matching created by Algorithm $XDZ6-No-Delta$ before the stopping condition $\Delta_M > \mathcal{Z}$ in Phase 4 is activated, are the same matches.*

Definition 8.30 *Given G_6 a six balanced banks graph and let M be the matching created by Algorithm $XDZ6$. Denote $\boldsymbol{\alpha}$ to be the number of swaps performed in Phase 4 of Algorithm $XDZ6$, and $\boldsymbol{p}$ be the number of iterations in Algorithm $XDZ6 - No - Delta$.*

Definition 8.31 *Given G_6 a six balanced banks graph. Denote:*
Let $\boldsymbol{M_0^1}$ be the matching after Phase 1 of Algorithm $XDZ6$ when the input is G_6.
Let $\boldsymbol{M_0^2}$ be the matching after Phase 2 of Algorithm $XDZ6$ when the input is G_6.
Let $\boldsymbol{M_0^3}$ be the matching after Phase 3 of Algorithm $XDZ6$ when the input is G_6.

Definition 8.32 *For $0 \leq j \leq p$, denote $\boldsymbol{M_j}$ as the matching provided by Algorithm $XDZ6-No-Delta$ in the j iteration of Phase 4 of the algorithm. Furthermore, define:*
For every $0 \leq j \leq p$, $\boldsymbol{xdz6_j} = val(M_j)$.
For example:
$\boldsymbol{xdz6_0} = val(M_0^3)$.
$\boldsymbol{xdz6_1} = val(M_1)$.
...
$\boldsymbol{xdz6_p} = val(M_p)$.

Definition 8.33 *Given G_6 a six balanced banks graph, define $\boldsymbol{\Delta_{M_i}}$ to be Δ_M where M is the matching in Phase 4 of Algorithm $XDZ6 - No - Delta$ in the i iteration, $0 \leq i \leq p$.*

Remark 8.34 *Given G_6 a six balanced banks graph. For $0 \leq i \leq p$, if $i \leq \alpha$, then $\Delta_{M_i} > \mathcal{Z}$.*
If $i > \alpha$, then $\Delta_{M_i} \leq \mathcal{Z}$.

Lemma 8.35 *Given G_6 a six balanced banks graph, Algorithm $XDZ6$ on G_6 is finitely determined.*

Proof: Since Algorithm $XDZ6 - No - Delta$ is finitely determined after p swaps, and according to Remark 8.29, all the operations of Algorithm $XDZ6$ are contained in Algorithm $XDZ6-No-Delta$, Algorithm $XDZ6$ is finitely determined. ∎

8.3 Phase 1

Lemma 8.36 *Given G_6 a six balanced banks graph. M_0^1, the matching created by Phase 1 of Algorithm $XDZ6$ on G_6 satisfies $Y[M_0^1] = Yopt$.*

Proof: M_0^1 is combined of three sets of edges which are $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$, $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ and $F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$. Which satisfy $Y[F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}] = 2t\mathcal{Y}$, since it contains t edges, each with value $Y - value = 2\mathcal{Y}$.
$Y[F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}] = 2t\mathcal{Y}$, since it contains t edges, each with value $Y - value = 2\mathcal{Y}$.
and $Y[F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}] = 0$, since it contains t edges, each with value $Y - value = 0$.
Therefore, $Y[M_0^1] = Y[F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}] + Y[F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}] + Y[F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}] = 4t\mathcal{Y}$ and according to Lemma 8.21, $Y[M_0^1] = 4t\mathcal{Y} = Yopt$. ∎

Lemma 8.37 *Given G_6 a six balanced banks graph. M_0^1, the matching created by Phase 1 of Algorithm $XDZ6$ on G_6 satisfies that $Z[M_0^1] = Zopt$.*

Proof: M_0^1 is combined of three sets of edges which are $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$, $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ and $F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$. Which satisfy $Z[F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}] = t\mathcal{Z}$, since it contains t edges with $Z - value = \mathcal{Z}$.
$Z[F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}] = t\mathcal{Z}$, , since it contains t edges with $Z - value = \mathcal{Z}$.
$Z[F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}] = t\mathcal{Z}$, since it contains t edges with $Z - value = \mathcal{Z}$.
Therefore, $Z[M_0^1] = Z[F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}] + Z[F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}] + Z[F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}] = 3t\mathcal{Y}$ then according to Lemma 8.24, $Z[M_0^1] = 3t\mathcal{Y} = Zopt$. ∎

8.4 Phase 2

Lemma 8.38 *If there exists an edge $e \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ which touches $S'(M_0^1)$, then every edge $f \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ either touches $S'(M)$ or $X - crosses$ X_{mid}.*

Proof: Since $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ is a result of Algorithm $X - Diagonals$ on two banks, according to Lemma 4.20, $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ is $Xopt$ on the vertices of $B_{(0,0)}$ and $B_{(2\mathcal{Y},\mathcal{Z})}$ and does not contain $X - disjoint$ edges.
Assume by contradiction that there exists an edge which touches $S''(M_0^1)$. In this case, e and f are $X - disjoint$. Contradicting the assumption that $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ does not contain $X - disjoint$ edges. ∎

Corollary 8.39 *If there exist an edge $e \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ which touches $S'(M)$, then every edge $f \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ either touches $S'(M)$ or $X - crosses$ X_{mid}.*

Corollary 8.40 *If there exist an edge $e \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ which touches $S'(M)$, then every edge $f \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ either touches $S'(M)$ or $X - crosses$ X_{mid}.*

Lemma 8.41 *Given G_6 a six balanced banks graph. Every two edges e_i, e' that are swapped in Phase 2 of Algorithm $XDZ6$, satisfy that $e_i \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ and $e' \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ or $e' \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.*

Proof: In Phase 2 of Algorithm $XDZ6$, we swap edge $e_i \in L'$ and e' which are $X - disjoint$. According to the way L' was defined, e_i either touches $S'(M)$ or $S''(M)$. without loss of generality suppose that e touches $S'(M)$. Since e' and e_i are $X - disjoint$ then e' touches $S''(M)$.
According to Corollary 8.40, if e touches $S'(M)$ then every edge in $F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ either touches $S'(M)$

or $X - crosses\ X_{mid}$.
Therefore $e' \notin F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ and it must satisfy $e' \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ or $e' \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.
■

Corollary 8.42 *Given G_6 a six balanced banks graph. Every two edges e_i, e' that are swapped in Phase 2 of Algorithm $XDZ6$ satisfy that one of them touches $S'(M)$ and the other touches $S''(M)$.*

Lemma 8.43 *Given G_6 a six balanced banks graph. Every two edges f, f', which are the result of a swap in Phase 2 of Algorithm $XDZ6$, $X - cross\ X_{mid}$.*

Proof: Let e and e' be two edges which are swapped in Phase 2 of Algorithm $XDZ6$. According to Corollary 8.42, one of them touches $S'(M)$ and the other touches $S''(M)$. Therefore, e and e' are $X - disjoint$ and each on different side of X_{mid}.
The result of the swap between such edges yields two edges which are not $X - disjoint$ and both $X - cross\ X_{mid}$. ■

Corollary 8.44 *Given G_6 a six balanced banks graph. In M_0^2, the result of Phase 2 of Algorithm $XDZ6$, all the edges on plane $P_{Y=\mathcal{Y}}$ are $X - cross\ X_{mid}$.*

Lemma 8.45 *Given G_6 a six balanced banks graph. M_0^2, the matching created at the end of Phase 2 of Algorithm $XDZ6$ on G_6, satisfies that $Y[M_0^2] = Yopt$ and $Z[M_0^2] = Zopt$.*

Proof: According to Lemma 8.41, in Phase 2 of Algorithm $XDZ6$, every two edges e, e' that are swapped in Phase 2 of Algorithm $XDZ6$ satisfy that $e \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ and $e' \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ or $e' \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.
If $e' \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$, according to Lemma 8.14, there exists a swap which yields f, f' such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.
If $e' \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$, according to Corollary 8.15, there exists a swap which yields f, f' such that $Y[e,\ e'] = Y[f,\ f']$ and $Z[e,\ e'] = Z[f,\ f']$.
Since e and e' are $X - disjoint$ edges then any swap is $X - Improving$.
Therefore, $Y[M_0^2] = Y[M_0^1]$ and $Z[M_0^2] = Z[M_0^1]$. According to Lemma 8.36, $Y[M_0^2] = Y[M_0^1] = Yopt$ and according to Lemma 8.37, $Z[M_0^2] = Z[M_0^1] = Zopt$. ■

8.5 Phase 3

Lemma 8.46 *Given G_6 a six balanced banks graph. Every four edges e, e', e_{Δ_M}, e'_{Δ_M} that are swapped in Phase 3 of Algorithm $XDZ6$, satisfy that $e,\ e' \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ and either $e_{\Delta_M} \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and $e'_{\Delta_M} \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ or $e'_{\Delta_M} \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and $e_{\Delta_M} \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.*

Proof: In Phase 2 of Algorithm $XDZ6$ we swapped every edge in L' with another edge which is $X - disjoint$.
According to Lemma 8.43, the result of this swap are two edges f and f' which $X - cross\ X_{mid}$.
Therefore, in the end of Phase 2, there are no edges in $F^1_{(0,\mathcal{Z}),(\mathcal{Y},\mathcal{Z})}$ that are $X - disjoint$, and therefore there are no edges in $F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ which touches $S(M)$.

Consequently, in the beginning of Phase 3 of Algorithm $XDZ6$, all the edges which touches $S'(M)$ and $S''(M)$ are from $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ or $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.
According to Lemmas 8.38, 8.39, either all the edges which touches $S'(M)$ are from $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and all the edges which touch $S''(M)$ are from $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ or vise versa.
Hence, either e_{Δ_M} touches $S'(M)$ and e'_{Δ_M} touches $S''(M)$ and in this case $e_{\Delta_M} \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and $e'_{\Delta_M} \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ or e_{Δ_M} touches $S''(M)$ and e'_{Δ_M} touches $S'(M)$ and in this case $e_{\Delta_M} \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$ and $e'_{\Delta_M} \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ ■

Corollary 8.47 *Given G_6 a six balanced banks graph. Every two edges e_{Δ_M}, e'_{Δ_M}, that are swapped in Phase 4 of Algorithm $XDZ6$, satisfy that one of them is in $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and the other is in $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.*

Lemma 8.48 *Given G_6 a six balanced banks graph. M_0^3, the matching created by Phase 3 of Algorithm $XDZ6$ on G_6, satisfies that $Y[M_0^3] = Yopt$ and $Z[M_0^3] = Zopt$.*

Proof: According to Lemma 8.46, in Phase 3 of Algorithm $XDZ6$, every four edges satisfy that $e,\ e' \in F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$, and without loss of generality, $e_{\Delta_M} \in F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and $e'_{\Delta_M} \in F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.
According to Lemma 8.12, the edges created by the swap in Phase 3 of Algorithm $XDZ6$ $f_1,\ f_2,\ f_3,\ f_4$ satisfy that $Y[e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}] = Y[f_1,\ f_2,\ f_3,\ f_4]$ and $Z[e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}] = Z[f_1,\ f_2,\ f_3,\ f_4]$.
Therefore, $Y[M_0^3] = Y[M_0^2]$ and $Z[M_0^3] = Z[M_0^2]$. According to Lemma 8.45, $Y[M_0^3] = Y[M_0^2] = Yopt$ and $Z[M_0^3] = Z[M_0^2] = Zopt$. ■

Lemma 8.49 *Given G_6 a six balanced banks graph. Every four conjoint-edges $e,\ e',\ e_{\Delta_M},\ e'_{\Delta_M}$, that are swapped in Phase 3 of Algorithm $XDZ6$ to $f_1,\ f_2,\ f_3,\ f_4$ satisfy that $f_1,\ f_2,\ f_3,\ f_4$, $X - cross\ X_{mid}$.*

Proof: According to Lemma 8.13, there exists a conjoint swap on a set of four conjoint-edges that satisfies $f_1,\ f_2,\ f_3,\ f_4$ $X - cross\ X_{mid}$. ■

Lemma 8.50 *M_0^3 contains only $X - disjoint$ edges, such that one is on $(B_{(0,0)}, B_{(2\mathcal{Y},\mathcal{Z})})$ and the other is on $(B_{(2\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$.*

Proof: M_0^1 is combined from three sets of edges $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})} \cup F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})} \cup F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ such that every set is $Xopt$.
According to Lemma 8.43, Every two edges $e,\ e'$ that are swapped in Phase 2 of Algorithm $XDZ6$ to $f,\ f'$ satisfy that f and f' $X - cross\ X_{mid}$. According to Lemma 8.49, Every four edges $e,\ e', e_{\Delta_M},\ e'_{\Delta_M}$ that are swapped in Phase 3 of Algorithm $XDZ6$ to $f_1,\ f_2,\ f_3,\ f_4$ satisfy that $f_1,\ f_2,\ f_3,\ f_4$ $X - cross\ X_{mid}$.
Therefore, in M_0^3, $X - disjoint$ edges exist only if the edges are big-Diagonal Edges. That means that all the rest of the edges are not $X - disjoint$ and therefore they are $Xopt$. ■

Lemma 8.51 *For every matching $M_0^o \in OPT_0$, $S'(M_0^3) = S'(M_0^o)$ and $S''(M_0^3) = S''(M_0^o)$.*

Proof: Since $M_0^o \in OPT_0$, M_0^o has the maximal $X-value$ such that $Y[M_0^o] = Yopt$ and $Z[M_0^o] = Zopt$.
According to Lemma 7.45, in M_0^o every pair of $X - disjoint$ edges $e,\ e'$ satisfy one of the three possibilities:

1. $e \in (B_{(0,0)}, B_{(2\mathcal{Y},\mathcal{Z})})$ and $e' \in (B_{(2\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$
2. $e \in (B_{(0,0)}, B_{(\mathcal{Y},\mathcal{Z})})$ and $e' \in (B_{(\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$
3. $e \in (B_{(\mathcal{Y},0)}, B_{(2\mathcal{Y},\mathcal{Z})})$ and $e' \in (B_{(2\mathcal{Y},0)}, B_{(\mathcal{Y},\mathcal{Z})})$

According to Lemma 8.50, M_0^3 contains only $X-disjoint$ edges such that one is on $(B_{(0,0)}, B_{(2\mathcal{Y},\mathcal{Z})})$ and the other is on $(B_{(2\mathcal{Y},0)}, B_{(0,\mathcal{Z})})$.
According to Lemma 8.64, $X[M_0^o] = Xopt - \Delta(S'(M_0^o), S''(M_0^o))$ and $X[M_0^3] = Xopt - \Delta(S'(M_0^3), S''(M_0^3))$.
By the way Algorithm $XDZ6$ works, M_0^3 has a minimum value of $\Delta(S'(M_0^3), S''(M_0^3))$, since the $X - disjoint$ edges are the inner edges on the big-diagonals.
Since $X[M_0^o]$ has the higher $X - value$ for matching with $Y - value = Yopt$ and $Z - value = Zopt$, $\Delta(S'(M_0^o), S''(M_0^o)) = \Delta(S'(M_0^3), S''(M_0^3))$ and therefore $S'(M_0^3) = S'(M_0^o)$ and $S''(M_0^3) = S''(M_0^o)$.
■

Lemma 8.52 *For every $M_0^o \in OPT_0$, $X[M_0^o] = X[M_0^3]$ and $opt_0 = xdz6_0$.*

Proof: M_0^1 is combined from three sets of edges $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})} \cup F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})} \cup F^1_{(\mathcal{Y},0),(\mathcal{Y},\mathcal{Z})}$ such that every set is $Xopt$.
According to Lemma 8.43, Every two edges $e,\ e'$ that are swapped in Phase 2 of Algorithm $XDZ6$ to f_1, f_2 satisfy that f and f' $X - cross$ X_{mid}. According to Lemma 8.49, Every four edges $e,\ e', e_{\Delta_M},\ e'_{\Delta_M}$ that are swapped in Phase 3 of Algorithm $XDZ6$ to $f_1,\ f_2,\ f_3,\ f_4$ satisfy that $f_1,\ f_2,\ f_3,\ f_4$ $X - cross$ X_{mid}.
Therefore, in M_0^3, $X - disjoint$ edges exist only if the edges are big-Diagonal Edges. That means that all the rest of the edges are not $X - disjoint$ and therefore they are $Xopt$.
According to Definition 8.30, there are p vertices in $S'(M_0^o)$ and p vertices in $S''(M_0^o)$, therefore according to Lemma 8.64, $X[M_0^o] = Xopt - \Delta(S'(M_0^o), S''(M_0^o))$.
According to Lemma 8.51, $S'(M_0^3) = S'(M_0^o)$ and $S''(M_0^3) = S''(M_0^o)$ and therefore $X[M_0^o] = X[M_0^3]$.
According to Lemma 8.48, $Y[M_0^3] = Yopt$ and $Z[M_0^3] = Zopt$.
Therefore, $xdz6_0 = opt_0$. ■

8.6 Phase 4

Lemma 8.53 *Given G_6 a six balanced banks graph. Let M be a matching obtained during the process of Algorithm $XDZ6$. Every two edges $e_{\Delta_M},\ e'_{\Delta_M}$, that are swapped in Phase 4 of algorithm $XDZ6$ to $f,\ f'$, satisfy that $X[f,\ f'] = X[e_{\Delta_M},\ e'_{\Delta_M}] + 2\Delta_M$, $Y[e_{\Delta_M},\ e'_{\Delta_M}] = Y[f,\ f'] = 4\mathcal{Y}$, $Z[e_{\Delta_M},\ e'_{\Delta_M}] = 2\mathcal{Z}$, $Z[f,\ f'] = 0$.*

Proof: Let f_1, f_1' and f_2, f_2' be the two possible sets of edges created by a $Y - Preserving$ and $X - Improving$ swap on e_{Δ_M} and e'_{Δ_M}.

According to Lemma 8.47, $e_{\Delta_M},\ e'_{\Delta_M}$ satisfy that one of then is in $F^1_{(0,0),(2\mathcal{Y},\mathcal{Z})}$ and the other in $F^1_{(2\mathcal{Y},0),(0,\mathcal{Z})}$.
Therefore, a $Y-Preserving$ swap yields $Y-value = 4\mathcal{Y}$ and $Z-value = 0$.
The Algorithm chooses the swap that preserve Y, denote these edges by $f,\ f'$.
By definition, $e_{\Delta_M},\ e'_{\Delta_M}$ are the farthest $X-disjoint$ edges. According to Lemma 4.1, the swapped edges satisfy $X[f,\ f'] = X[e_{\Delta_M},\ e'_{\Delta_M}] + 2\Delta_M$.
Therefore, Algorithm $XDZ6$ satisfy that $X[f,\ f'] = X[e_{\Delta_M},\ e'_{\Delta_M}] + 2\Delta_M,\ Y[e_{\Delta_M},\ e'_{\Delta_M}] = Y[f,\ f'],\ Z[e_{\Delta_M},\ e'_{\Delta_M}] = 2\mathcal{Z},\ Z[f,\ f'] = Z[e_{\Delta_M},\ e'_{\Delta_M}] - 2\mathcal{Z} = 0$. ∎

Lemma 8.54 *For every* $0 \le j \le p$*:*
$X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$.
$Y[M_{j+1}] = Y[M_j]$.
$Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$.

Proof: In every iteration in Phase 4 of Algorithm $XDZ6-No-Delta$, the algorithm swaps two $X-disjoint$ edges $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ using a $Y-Preserving$ and $X-Improving$ swap such that $f,\ f'$ are the edges created by the swap. According to Lemma 8.53, these edges satisfy $Y[e,\ e'] = Y[f,\ f'] = 4\mathcal{Y},\ Z[e,\ e'] = 2\mathcal{Z},\ Z[f,\ f'] = 0,\ X[f,\ f'] = X[e,\ e'] + 2\Delta_M$.
Therefore, $X[M_{j+1}] = X[M_j] + 2\Delta_{M_j}$, $Y[M_{j+1}] = Y[M_j]$, $Z[M_{j+1}] = Z[M_j] - 2\mathcal{Z}$. ∎

Corollary 8.55 *For every* $0 \le j \le p$*:*
$X[M_j] = X[M_0^3] + \sum_{k=0}^{j-1} 2\Delta_{M_k}$.
$Y[M_j] = Y[M_0^3]$.
$Z[M_j] = Z[M_0^3] - 2j\mathcal{Z}$.
Therefore, $xdz6_{j+1} = xdz6_j + 2\Delta_{M_j} - 2j\mathcal{Z}$.
$xdz6_p = xdz6_0 + \sum_{k=0}^{p-1} 2\Delta_{M_k} - 2p\mathcal{Z}$.

Lemma 8.56 $xdz6_\alpha = max\{xdz6_0, \ldots, xdz6_p\}$.

Proof: According to Corollary 8.55, for $0 \le j \le p$, $xdz6_j = xdz6_{j+1} - 2\Delta_{M_j} + 2\mathcal{Z}$.
Therefore, for $i \le \alpha$, it holds that $xdz6_{i+1} \ge xdz6_i$.
For $i > \alpha$, it holds that $xdz6_{i+1} < xdz6_i$.
Hence, $xdz6_\alpha = max\{xdz6_0, \ldots, xdz6_p\}$. ∎

Lemma 8.57 *Given* G_6 *a six balanced banks graph. For* M_j*, the matching obtained in the* j *iteration of Phase 4 of Algorithm* $XDZ6$*,* $e_{\Delta_{M_j}}$ *and* $e'_{\Delta_{M_j}}$ *contains the minimal* $X-value$ *vertex in* $S'(M)$ *and the maximal* $X-value$ *vertex in* $S''(M)$*.*

Proof: Since $e_{\Delta_{M_j}}$ and $e'_{\Delta_{M_j}}$ are $X-disjoint$, then without loss of generality, one of the vertices of $e_{\Delta_{M_j}}$ is in $S'(M_j)$ and one of the vertices of $e'_{\Delta_{M_j}}$ is in $S''(M_j)$.
Since Δ_{M_j} is the maximum distance between two vertices, such that one in $S'(M_j)$ and the other in $S''(M_j)$, and these vertices must be the minimal $X-value$ vertex in $S'(M)$ and the maximal $X-value$ vertex in $S''(M)$, see Figure 64. ∎

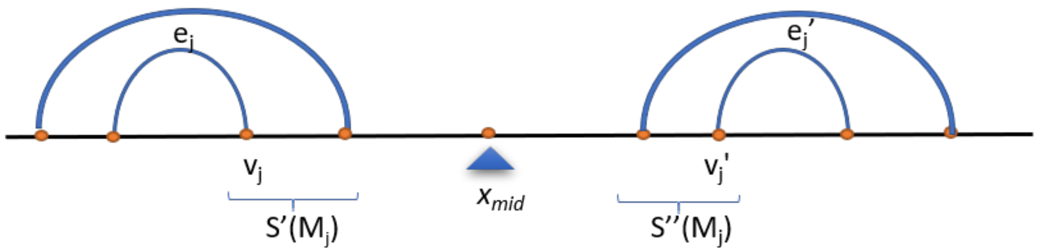

Figure 64: A matching with the farthest $X - disjoint$ e and e'

Lemma 8.58 *Given G_6 a six balanced banks graph. For M_j, the matching obtained in the j iteration of Phase 4 of Algorithm $XDZ6$, a $Y - Preserving$ and $X - Improving$ swap performed on $e_{\Delta_{M_j}}, e'_{\Delta_{M_j}}$ results in $f,\ f'$, that are not $X - disjoint$ with any other edge in M_j.*

Proof: Denote $e_{\Delta_{M_j}} = (v_2, v_3), e'_{\Delta_{M_j}} = (u_2, u_4)$. According to Lemma 8.57, without loss of generality, v_2 is the minimal vertex in $S'(M_j)$ and u_4 is the maximal vertex in $S''(M_j)$.
The possible swaps are $f_1, f_1' = (v_2, u_2), (v_3, u_4)$ or $f_2, f_2' = (v_2, u_4), (v_3, u_2)$. Since both options are $X - Improving$, choose $f,\ f'$ to be the $X - Improving$ and $Y - Preserving$ swap, see Figure 65. Hence $f,\ f'$ $X - cross$ all the vertices in $S'(M_j)$ and $S''(M_j)$, and are not $X - disjoint$ with any edge which contains vertices from $S(M_j)$. Since $f,\ f'$ $X - cross$ X_{mid}, according to Lemma 4.7, $f,\ f'$ is not $X - disjoint$ with any edge that $X - cross$ X_{mid}. All the edges before the swap crossed X_{mid}, therefore, $f,\ f'$ are not $X - disjoint$ with any edge in M_j. ■

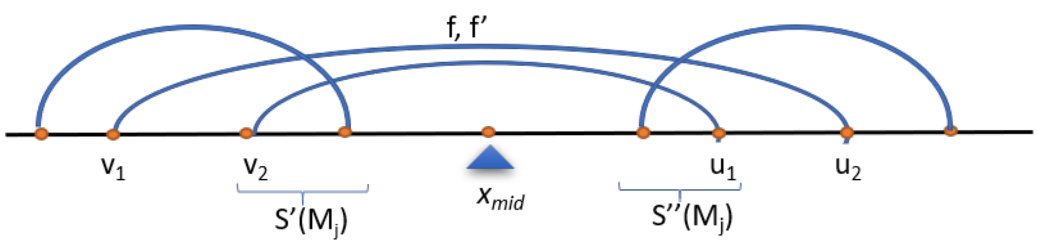

Figure 65: A matching after swapping to f and f'

Lemma 8.59 *For every two matchings M and M' on G_6 a six balanced banks graph, if $S'(M) = S'(M')$ and $S''(M) = S''(M')$, the distance between the farthest $X - disjoint$ edges satisfies $\Delta_M = \Delta_{M'}$.*

Proof: Δ_M and $\Delta_{M'}$ are the distances on the X axis between the farthest $X - disjoint$ edges. Therefore, Δ_M is the distance between the minimal $X - value$ vertex in $S'(M)$ and the maximal $X - value$ vertex in $S''(M)$ since $S'(M) = S'(M')$ and $S''(M) = S''(M')$, $\Delta_M = \Delta_{M'}$. $\Delta_{M'}$ is also the distance between these two vertices. ■

Lemma 8.60 *The number of vertices in $S'(M_j)$ is equal to the number of vertices in $S''(M_j)$.*

Proof: By enumeration argument, there must be an equal number of $X - one - sided$ edges on both sides of X_{mid}, and by definition, $S'(M_j)$ and $S''(M_j)$ are on different sides of X_{mid}, then the number of vertices in $S'(M_j)$ is equal to the number of vertices in $S''(M_j)$. ∎

Lemma 8.61 *For every $0 \leq j \leq p$, for every $M_j^o \in OPT_j$, $Y[M_j^o] = Y[M_j]$ and $Z[M_j^o] = Z[M_j]$.*

Proof: For every j, M_j^o is a matching with the maximum value with exactly $2j$ edges with $Z - value = 0$, and t edges with $Y - value = 0$ therefore, $Z[M_j^o] = Zopt - 2j\mathcal{Z}$ and $Y[M_j^o] = Yopt$.
According to Corollary 8.55, $Z[M_j] = Z[M_0^3] - 2j\mathcal{Z}$, and $Y[M_j] = Y[M_0^3]$. And, according to Lemma 8.48, $Z[M_0^3] = Zopt$, and $Y[M_0^3] = Yopt$.
Therefore, $Z[M_j] = Zopt - 2j\mathcal{Z} = Z[M_j^o]$ and $Y[M_j] = Yopt = Y[M_j^o]$. ∎

Definition 8.62 *Given G_6 a six balanced banks graph, and a matching M on G_6.*
Let $v_1, \ldots, v_p$ be the vertices in $S'(M)$ ordered according to their $X - value$.
Let $u_1, \ldots, u_p$ be the vertices in $S''(M)$ ordered according to their $X - value$.
Denote $\boldsymbol{\Delta(S'(M), S''(M))} = \sum_{i=1}^{p} 2(X[u_{p-i+1}] - X[v_i])$.

Lemma 8.63 *Given G_6 a six balanced banks graph, and a matching M on G_6. $X[M] = Xopt - \Delta(S'(M), S''(M))$.*

Proof: Each swap of two $X - disjoint$ edges in M, one with a vertex in $S'(M)$ and the other with a vertex in $S''(M)$, results in two edges which both $X - cross$ X_{mid}, and therefore according to Lemma 4.7, they are not $X - disjoint$.
$S(M)$ contains all the $X - disjoint$ edges in M. Performing all the swaps for all the $X - disjoint$ edges in M, will result in M' with no $X - disjoint$ edges and $X[M'] = Xopt$.
Each swap the algorithm matches the edges that touches v_1, u_p that will add to the matching the $X - value$ $X[u_p] - X[v_1]$ and in the next iteration swaps the vertices v_2, u_{p-1} and so on.
Therefore, for $1 \leq i, j \leq p$, $X - value$ of $2(X[u_j] - X[v_i])$. Therefore, $X[M] = Xopt - \sum_{i=1}^{p} 2(X[u_{p-i+1}] - X[v_i])$. ∎

Lemma 8.64 *Given G_6 a six balanced banks graph, and a matching M on G_6. All $X - disjoint$ edges in M satisfy that one touches $S'(M)$ and other touches $S''(M)$. Then $X[M] = Xopt - \Delta(S'(M), S''(M))$.*

Proof: $S(M)$ contains all the $X - disjoint$ edges in M. Performing all the swaps for all the $X - disjoint$ edges in M, results in a matching M' with no $X - disjoint$ edges and $X[M'] = Xopt$.
Each swap of two $X - disjoint$ edges in M, one touches $S'(M)$ and the other touches $S''(M)$, results in two edges which both cross X_{mid}, therefore according to Lemma 4.7, they are not $X - disjoint$.
Each swap provides for $1 \leq i, j \leq p$, an $X - value$ of $2(X[u_j] - X[v_i])$.
Therefore, $X[M] = Xopt - \sum_{i=1}^{p} 2(X[u_{p-i+1}] - X[v_i])$. ∎

Lemma 8.65 *Given G_6 a six balanced banks graph with t vertices on each bank. Algorithm $XDZ6$ on G_6 is finitely determined and returns a matching M, such that if $e,\ e' \in M$ are $X - disjoint$ edges then both are Big-Diagonal Edges.*

Proof: In phase 2 of Algorithm $XDZ6$ all the $X - disjoint$ edges that are not both are Big-Diagonals Edges are swapped. Therefore, $e,\ e'$ are both Big-Diagonal Edges. ∎

Lemma 8.66 *Given G_6 a six balanced banks graph with t vertices on each bank. Algorithm $XDZ6$ on G_6 is finitely determined and returns a matching M, such that there is no possible swap for two edges that improves the matching.*

Proof: According to Lemma 8.35, Algorithm $XDZ6$ finitely determined, and according to Definition 8.30, the result of the algorithm is a matching M which satisfies $val(M) = xdz6_\alpha$.
For $e, e' \in M\ X - disjoint$ edges, according to Lemma 8.65, since $e,\ e'$ are $X - disjoint$ edges then both Big-Diagonal Edges. In Phase 4 of Algorithm $XDZ6$, $e,\ e'$ have not been swapped. Therefore, $\Delta_M < \mathcal{Z}$ then a swap provides M' such that $val(M) > val(M')$.
Since all the other edges are not $X - disjoint$ then there is no possible swap for two edges that improves the $X - value$ and preserve $Y - value$ and $Z - value$. Therefore, there is no possible swap for two edges that improves the matching. ∎

9 Summary and further research

In this paper we introduce algorithms for solving maximum matching on four balanced graph, maximum matching on four unbalanced graph, and local maximum matching on six balanced graph, we used Manhattan topology. Our proofs are based on matching pairs with maximum Y and Z values, then performing swaps to improve the overall sum of the matching, when the X-value is greater than the Y or Z values.

Further research should be to generalize our algorithms for solving maximum matching for $2n$ balanced graph and $2n$ unbalanced graph.

References

[1] Y. Berkovitz. Maximum matching on dimensional graphs using manhattan distances. Technical report, The Academic College of Tel Aviv Yaffo, 2019.

[2] J. Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards Section B Mathematics and Mathematical Physics*, page 125, 1965.

[3] H. N. Gabow. An efficient implementation of edmonds' algorithm for maximum matching on graphs. *J. ACM*, 23(2):221—234, April 1976.

[4] F. Glover. Maximum matching in a convex bipartite graph. *Naval Research Logistics Quarterly*, 14(3):313–316, 1967.

[5] R. Hassin, S. Rubinstein, and A. Tamir. Approximation algorithms for maximum dispersion. *Operations Research Letters*, 21(3):133–137, 1997.

[6] R. M. Karp, E. Upfal, and A. Wigderson. Constructing a perfect matching is in Random NC. *Combinatorica*, 6(1):35–48, 1986.

[7] Y. Kostina. Maximal matching on multilevel banks river. Technical report, The Academic College of Tel Aviv Yaffo, 2017.

[8] W. Lipski and F. Preparata. Efficient algorithms for finding maximum matchings in convex bipartite graphs and related problems. *Acta Informatica*, 15:329–346, 2004.

[9] E. Shvartz. Obnoxious points optimal matching. final project. Technical report, The Academic College of Tel Aviv Yaffo, 2014.