Monochromatic Plane Matchings in Bicolored Point Set

A. Karim Abu-Affash* Sujoy Bhore† Paz Carmi‡

Abstract

Motivated by networks interplay, we study the problem of computing monochromatic plane matchings in bicolored point set. Given a bicolored set P of n red and m blue points in the plane, where n and m are even, the goal is to compute a plane matching M_R of the red points and a plane matching M_B of the blue points that minimize the number of crossing between M_R and M_B as well as the longest edge in $M_R \cup M_B$. In this paper, we give asymptotically tight bound on the number of crossings between M_R and M_B when the points of P are in convex position. Moreover, we present an algorithm that computes bottleneck plane matchings M_R and M_B, such that there are no crossings between M_R and M_B, if such matchings exist. For points in general position, we present a polynomial-time approximation algorithm that computes two plane matchings with linear number of crossings between them.

1 Introduction

Let P be a set of n points in the plane, where n is even. A *perfect matching* of P is a perfect matching in the complete Euclidean graph induced by P. A *bottleneck matching* M of P is a perfect matching of P that minimizes the length of the longest edge of M. Let λ_M denote the bottleneck of M, i.e., the length of the longest edge in M. A *plane matching* is a matching with non-crossing edges. Matching problems have been studied extensively, see, e.g., [3, 5, 6, 9, 12, 13].

In this paper, we study the bottleneck plane matching problem in bicolored point set. Let P be a bicolored set consisting of n red and m blue points, such that n and m are even. Let R denote the set of the red points of P, and let B denote the set of the blue points of P. An *L-monochromatic matching* M of P is a matching of P, such that (i) $M = M_R \cup M_B$, where M_R and M_B are plane perfect matchings of R and B, respectively, and (ii) $\lambda_M \leq L$. Let $cr(P, L)$ denote the minimum number of intersection present in any L-monochromatic matching of P. In this paper, we investigate the value $cr(P, L)$ for different bicolored point sets, and study the problem of computing an L-monochromatic matching.

1.1 Related work

The bottleneck plane matching problem for general point set has been first studied by Abu-Affash et al. [2]. They showed that the problem is NP-hard and presented a $2\sqrt{10}$-approximation algorithm. In [1], the authors showed how to compute a plane matching of size at least $n/5$, whose edges have length at most λ^* in $O(n \log^2 n)$ time, and a plane matching of size at least $2n/5$, whose edges have length at most $(\sqrt{2} + \sqrt{3}) \cdot \lambda^*$ in $O(n \log n)$ time, where λ^* is the length of the longest edge of a bottleneck plane matching. Carlsson and Armbruster [4] proved that the bipartite (red/blue) version of the bottleneck plane matching problem is NP-hard, and gave an $O(n^3 \log n)$-time algorithm that solves the problem when the points are on convex position, and an $O(n^4 \log n)$-time algorithm that solves the problem when the red points lie on a line l and the blue points lie above (or below) l.

For bicolored inputs, Merino et al. [10] obtained a tight bound on the number of intersections in monochromatic minimum weight matchings for bicolored point sets. Tokunaga [11] examined non-crossing spanning trees of the red points and the blue points and found a tight bound on the minimum number of intersections between the red and blue spanning trees. Joeris et al. [7] studied the number of intersections for monochromatic planar spanning cycles. In each of the above works, points could have only one of two colors and number of intersection points were proved to be asymptotically linear on total number of points. Kano et al. [8] considered the case of more than two colors and studied the number of intersections for monochromatic spanning trees.

1.2 Our results

In Section 2, we consider the case when the points of P are in convex position. We give a tight bound on $cr(P, L)$ in this case. Moreover, we give an algorithm that computes in $O(|P|^3 + |P|)$ time an L-monochromatic matching of P, where L is the minimum real number such that the edges of the matching do not cross each other, if such a matching exists. In Section 3, we give a polynomial-time approximation algorithm for points in general position.

*Software Engineering Department, Shamoon College of Engineering, Beer-Sheva 84100, Israel, abuaa1@sce.ac.il.

†Department of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel, sujoy.bhore@gmail.com.

‡Department of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel, carmip@cs.bgu.ac.il. The research is partially supported by the Lynn and William Frankel Center for Computer Science.

2 Monochromatic Matching of Points in Convex Position

In this section, we consider the case where the points of $P = R \cup B$ are in convex position, i.e., the points of P form vertices of a convex polygon. Let M_R and M_B denote the bottleneck plane matchings of the sets R and B, respectively. Let λ_R and λ_B denote the bottlenecks of M_R and M_B, respectively, and let $\lambda = \max\{\lambda_R, \lambda_B\}$. Notice that $M_R \cup M_B$ is not necessarily a plane matching. Notice also that in any L-monochromatic matching of P, the length of the longest edge is at least λ. Let $cr(P, \lambda)$ be the minimum number of intersection present in any L-monochromatic matching of P with $L = \lambda$. In the following, we give a tight bound of $cr(P, \lambda)$.

2.1 Lower bound on $cr(P, \lambda)$

In Figure 1, we show a set of bicolored points P in convex position, such that the number of crossings between M_R and M_B cannot be better than linear.

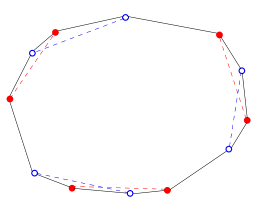

Figure 1: Any L-monochromatic matching of P has linear number of crossings between the red and the blue matchings.

2.2 Upper bound on $cr(P, \lambda)$

For an edge (a, b) and a point w, let $\|(a, b) - w\|$ denote the minimum Euclidean distance between w and any point on (a, b); see Figure 2. For two non-intersecting edges (a, b) and (c, d), such that $\{a, c, d, b\}$ are in convex position and (a, c) does not intersect (b, d), let $\|(a, b) - (c, d)\|$ denote $\max\{|ca|, |db|\}$; see Figure 3. In the following theorem, we give an upper bound on $cr(P, \lambda)$.

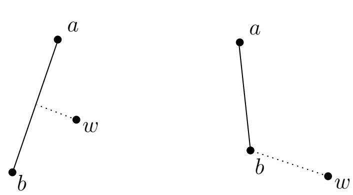

Figure 2: $\|(a, b) - w\|$ is the minimum distance between w and any point on (a, b).

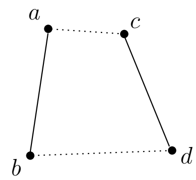

Figure 3: $\|(a, b) - (c, d)\|$ is $\max\{|ca|, |db|\}$.

Theorem 1 *Let $P = R \cup B$ be a set of points in convex position. Then $cr(P, \lambda) \leq \frac{9k}{2}$, where $k = \min\{n, m\}$, $n = |R|$, and $m = |B|$.*

Proof. Assume, w.l.o.g., that $n \leq m$, i.e., $k = n$. For the sake of contradiction, suppose that $cr(P, \lambda) \geq \frac{9n}{2} + 1$. Let M_P be a λ-monochromatic matching of P with minimum number of intersections and $M_P = M_R \cup M_B$. Thus, the number of intersection in M_P is at least $\frac{9n}{2} + 1$. Since $|M_R| = \frac{n}{2}$, by pigeonhole principle there is at least one edge $(x, y) \in M_R$ intersected by at least 10 edges from M_B. Assume, w.l.o.g., that x and y are on the X-axis, and notice that $|xy| \leq \lambda$. We define the region $U = \{w \in \mathbb{R}^2 : \|(x, y) - w\| \leq \lambda\}$; see Figure 4. Observe that the total perimeter of the region U is at most $2(\pi + 1)\lambda \approx 8.29\lambda$.

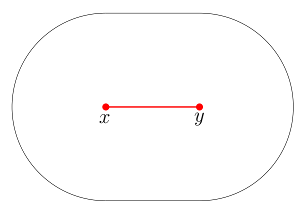

Figure 4: The set of all points of distance at most λ from (x, y).

Let $E = \{(p_1, q_1), (p_2, q_2), \ldots, (p_j, q_j)\}$, where $j \geq 10$, be the set of edges of M_B that intersect (x, y), such that $p_1, p_2, \ldots, p_j$ are above (x, y), $q_1, q_2, \ldots, q_j$ are below (x, y), and, for each $1 \leq i < j$, the intersection point of (p_i, q_i) with (x, y) is to the left of the intersection point of (p_{i+1}, q_{i+1}) with (x, y).

Lemma 2 *There is at least two edges (p_i, q_i), (p_{i+1}, q_{i+1}) in E, such that $\|(p_i, q_i) - (p_{i+1}, q_{i+1})\| \leq \lambda$.*

Proof. For the sake of contradiction, suppose that, for each $1 \leq i < j$, we have $\|(p_i, q_i) - (p_{i+1}, q_{i+1})\| > \lambda$. Thus, for each $1 \leq i < j$, either $|p_i p_{i+1}| > \lambda$ or $|q_i q_{i+1}| > \lambda$. Therefore, the total perimeter of the convex polygon S formed by $\{p_1, p_2, \ldots p_j, q_j, q_{j-1}, \ldots, q_1\}$ is at least $\sum_{i=1}^{j-1} |p_i p_{i+1}| + |q_i q_{i+1}| > 9\lambda$, since $j \geq 10$. This contradicts the fact that S is contained inside the convex region U whose perimeter is at most 8.29λ. $\square$

By Lemma 2, there are two adjacent edges $(p_i, q_i), (p_{i+1}, q_{i+1}) \in E$, such that $|p_i p_{i+1}| \leq \lambda$ and $|q_i q_{i+1}| \leq \lambda$. We replace the edges (p_i, q_i) and (p_{i+1}, q_{i+1}) in M_B by the edges (p_i, p_{i+1}) and (q_i, q_{i+1}) to obtain a new bottleneck plane matching M'_B of B of bottleneck at most λ. Let $M'_P = M_R \cup M'_B$. Clearly, M'_P is a λ-monochromatic matching of P. In the following lemma, we show that the new edges (p_i, p_{i+1}) and (q_i, q_{i+1}) do not increase the number of intersections in M'_P.

Lemma 3 *If (p_i, p_{i+1}) or (q_i, q_{i+1}) intersects an edge (u, v) in M_R, then either (p_i, q_i) or (p_{i+1}, q_{i+1}) intersects (u, v) in M_P.*

Proof. Assume, w.l.o.g, that (p_i, p_{i+1}) intersects an edge (u, v) in M_R. Let Q be the quadrangle obtained by points x, p_i, p_{i+1} and y. Clearly, Q is empty of points of P, since P in convex position. Moreover, since the edges of M_R do not intersect each other, (u, v) and (x, y) do not intersect. Thus, (u, v) intersects (p_i, q_i) or (p_{i+1}, q_{i+1}) ; see Figure 5. □

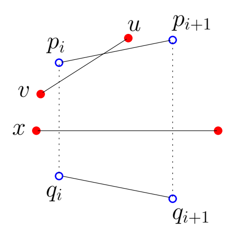

Figure 5: An illustration of Lemma 3.

By Lemma 3, the edges (p_i, p_{i+1}) and (q_i, q_{i+1}) do not increase the number of intersections in M'_P. However, (p_i, p_{i+1}) and (q_i, q_{i+1}) decrease the number of intersections in M'_P, since they do not intersect (x, y). This implies that the number of intersections in M'_P is less than that in M_P, which contradicts the assumption that M_P is a λ-monochromatic matching with minimum number of intersections. This completes the proof of Theorem 1. □

Remark. Abu-affash et al. [1] computed a bottleneck plane matching of a set of points in convex position in $O(|P|^3)$ time. We use their algorithm to compute bottleneck plane matchings of R and of B, separately. Then in additional $O(|P|)$ time, we compute a λ-monochromatic matching M_P of P by considering each edge (x, y) of M_R separately, and if there are two adjacent edges $(p_i, q_i), (p_{i+1}, q_{i+1}) \in M_B$ intersecting (x, y), such that $|p_i p_{i+1}| \leq \lambda$ and $|q_i q_{i+1}| \leq \lambda$, then we replace them by the edges (p_i, p_{i+1}) and (q_i, q_{i+1}). By the above lemmas, the number of intersections in M_P is at most $\frac{9k}{2}$, where $k = \min\{n, m\}$.

2.3 Monochromatic plane matching

Let L be the minimum value such that there exists an L-monochromatic matching of P with no crossings, if such a matching exists. That is, L is the minimum value such that $cr(P, L) = 0$, if such a value exists. An *optimal plane matching* of P is an L-monochromatic matching of P with no crossings. In this section, we present a polynomial-time algorithm that computes an optimal plane matching of P, if such a matching exists,

Let $P = R \cup B$ be a set of points in convex position, where R contains n even number of red points and B contains m even number of blue points. Recall that M_R (resp., M_B) is a bottleneck plane matching of R (resp., B) of bottleneck λ_R (resp., λ_B) and $\lambda = \max\{\lambda_R, \lambda_B\}$. Thus, the length of the longest edge in any L-monochromatic matching of P is at least λ. Our algorithm computes the minimum value L, such that $cr(P, L) = 0$ (if exists), and constructs an optimal plane matching of P.

Let us assume that $R = \{r_1, \ldots, r_n\}$, $B = \{b_1, \ldots, b_m\}$, and $P = \{p_1, \ldots, p_{n+m}\}$, such that $p_i \in R \cup B$. Let $P = R \cup B$ denote the vertices of the convex polygon, and are ordered in clockwise order; see Figure 6. Assume, w.l.o.g., that $p_1 = r_1$. Notice that, a bottleneck plane matching M_R of R and a bottleneck plane matching M_B of B can be computed separately in polynomial time [2]. However, the edges of M_R may cross the edges of M_B. Let M_P denote an optimal plane matching of P. We first state the following observation.

Observation 1 *For each edge (p_i, p_j) in M_P,*
(i) p_i and p_j have the same color,
(ii) (p_i, p_j) partitions the points of P into two disjoint point sets, such that the number of red points and blue points contained in each set is even, and
(iii) $i + j$ is odd; see Figure 6.

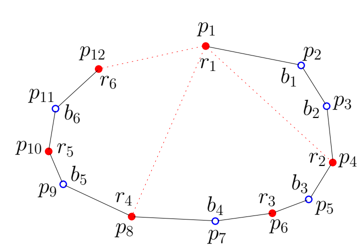

Figure 6: The convex polygon that is obtained from P. p_1 can be matched to the points p_4, p_8, p_{12}.

Based on this observation, we define a weight function for each pair of points in P as follows. For each $1 \leq i < j \leq n+m$, we define $w_{i,j} = |p_i p_j|$, if p_i and p_j are of the same color, $i + j$ is odd, and the edge (p_i, p_j) partitions the points in P into two disjoint point sets such that the number of red points and blue points contained in each set is even. Otherwise, $w_{i,j} = \infty$. Let $P[i,j]$ denote the set of points $\{p_i, p_{i+1}, \dots, p_j\}$. Let $\ell_{i,j}$ be the minimum value, such that $cr(P[i,j], \ell_{i,j}) = 0$. Hence, $L = \ell_{1,n+m}$.

Let p_k be the point matched to p_1 in M_P. Thus, $L = \ell_{1,n+m} = \max\{w_{1,k}, \ell_{2,k-1}, \ell_{k+1,n+m}\}$. Therefore, to compute $\ell_{1,n+m}$, we compute $\max\{w_{1,k}, \ell_{2,k-1}, \ell_{k+1,n+m}\}$, for each even k between 2 and $n+m$, and take the minimum over these values. In general, for each $1 \leq i < j \leq n+m$, such that $i+j$ is odd, we compute $\ell_{i,j}$ by

$$\min_{k=i+1,i+3,\cdots,j} \begin{cases} w_{i,k} & \text{, if } k = i+1 = j \\ \max\{w_{i,k}, \ell_{k+1,j}\} & \text{, if } k = i+1 \\ \max\{w_{i,k}, \ell_{i+1,k-1}\} & \text{, if } k = j \\ \max\{w_{i,k}, \ell_{i+1,k-1}, \ell_{k+1,j}\} & \text{, otherwise.} \end{cases}$$

We compute $L = \ell_{1,n+m}$ using dynamic programming. The dynamic programming table T contains $(n+m)$ rows and $(n+m)$ columns and the entry $T[i,j]$ corresponds to a solution of the problem for the set $P[i,j]$. Notice that, each entry $T[i,j]$ is computed by processing $O(n+m)$ entries that are already computed. Hence, we can compute $L = \ell_{i,j} = T[1, n+m]$ in $O((n+m)^3)$ time. Thus, we have the following theorem.

Theorem 4 *Given a set P of n red and m blue points in convex position, where n and m are even, one can compute in $O((n+m)^3)$ time the minimum value L, such that $cr(P, L) = 0$, and if L is finite, then an optimal plane matching can be computed in $O((n+m)^3)$ time.*

3 Monochromatic Matching of Points in General Position

Let $P = R \cup B$ be a set of bicolored points in the plane, such that R contains n red points, B contains m blue points, and n and m are even. In this section, we present a polynomial-time approximation algorithm that computes an L-monochromatic matching of P with at most $O(|P|)$ crossings.

Let M_R and M_B be bottleneck plane matchings of R and B, respectively. Let λ_R and λ_B be the bottlenecks of M_R and M_B, respectively, and let $\lambda = \max\{\lambda_R, \lambda_B\}$. Recall that, in any L-monochromatic matching of P, the length of the longest edge is at least λ. In [2], Abu-affash et al. proved that computing a bottleneck plane matching of a set of points in general position in the plane is NP-Hard. This implies that it is NP-Hard to compute a λ-monochromatic matching of P since $\lambda = \max\{\lambda_R, \lambda_B\}$. However, in this section we give a polynomial time approximation algorithm that computes a $(2\sqrt{10}\lambda)$-monochromatic matching of P with linear number of intersections. That is, we compute an L-monochromatic matching M_P with $L = 2\sqrt{10}\lambda$, such that total number of intersections between the red and the blue edges in M_P is $O(|P|)$.

Let M_R^* and M_B^* be bottleneck matchings of R and B that may have crossings, respectively. Let λ_R^* and λ_B^* be the bottlenecks of M_R^* and M_B^*, respectively. Since M_R^* and M_B^* can be computed in polynomial-time [5], we assume, w.l.o.g., that $\lambda_B^* \leq \lambda_R^*$. Thus, since $\lambda_B^* \leq \lambda_B$ and $\lambda_R^* \leq \lambda_R$, we have $\lambda_R^* \leq \lambda$. In the rest of this section, we prove the following theorem.

Theorem 5 *Let $P = R \cup B$ be a set of bicolored points in the plane, such that contains n red points, B contains m blue points, and n and m are even. Then, there is a polynomial-time approximation algorithm that computes a $(2\sqrt{10}\lambda_R^*)$-monochromatic matching M_P of P, such that the number of intersections in M_P is $O(|P|)$.*

Proof. Let M_P^* be $M_R^* \cup M_B^*$. We begin by laying a grid of side length $2\sqrt{2}\lambda_R^*$. Assume, w.l.o.g. that no point of P lies on the boundary of a grid cell. Each edge of M_P^* is either contained in a grid cell or connects two points from two adjacent cells (i.e., two cells sharing a side or corner). For an edge e in M_P^*, we say that e is an *internal* edge if it is contained in a grid cell, and an *external* edge otherwise. An external edge can be of two types: *straight* external edge (s-edge for short) connects between two points in two grid cells that share a side, and *diagonal* external edge (d-edge for short) connects between two points in two grid cells that share a corner. Let C be a grid cell. The degree of C is denoted by $deg(C)$ and it is equal to the number of external edges of M_P^* with an endpoint in C.

Our algorithm consists of two stages. In Stage 1, we convert M_P^* into a new matching M_P' of P, such that $deg(C) \leq 8$, for each grid cell C, and, in Stage 2, we construct $(2\sqrt{10}\lambda_R^*)$-monochromatic matching M_P based on M_P'.

Stage 1

In this stage, which is taken from [2], we convert M_R^* (resp., M_B^*) to a new perfect matching M_R' (resp., M_B') of R (resp., of B). The conversion is done by applying a sequence of rules on M_R^* and on M_B^* separately. Each rule is applied as long as there is an instance in the current matching to which it can be applied. When there are no more such instances, we move to the next rule in the sequence.

Rule 1: If there are two d-edges (a,b) and (c,d) associated with the same corner, such that a and c in the same cell and b and d in the same cell, then these edges are replaced by two internal edges (a,c) and (b,d); see Figure 7(a).

Rule 2: If there are two d-edges (a,b) and (c,d) associated with the same corner, such that a, b, c, and d are in different cells, then these edges are replaced by two s-edges (a,c) and (b,d); see Figure 7(b).

For a d-edge (a,b) that connects between two cells C and C' and associated with a corner r, we define a

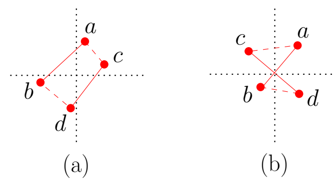

Figure 7: (a) Two d-edges (a, b) and (c, d) are replaced by two internal edges (a, c) and (b, d), and (b) two d-edges (a, b) and (c, d) are replaced by two s-edges.

a *danger zone* of (a, b) in each of the two other cells sharing r as an isosceles right triangle; see Figure 8(a). The length of its sides is $\sqrt{2}\lambda^*_R$ and it is semi-open (i.e., it does not include the hypotenuse of length $2\lambda^*_R$).

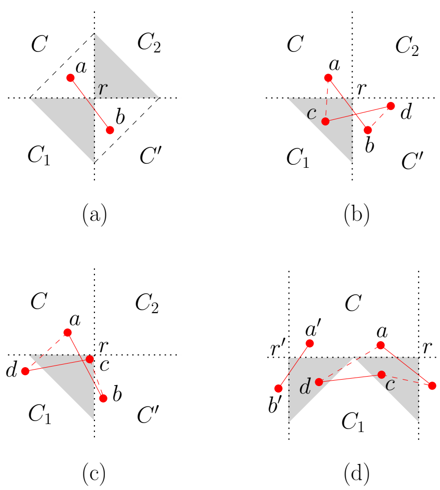

Figure 8: (a) The danger zone defined by the edge (a, b) in C_1 and C_2, (b) (c, d) is an s-edge, (c) (c, d) is an internal edge and d is not in another danger zone in C_1, and (d) (c, d) is an internal edge and d is in another danger zone in C_1.

Rule 3: This rule is applied on a d-edge (a, b) and an edge (c, d) with an endpoint c inside a danger zone defined by (a, b); see Figure 8(b–d).

- If (c, d) is an s-edge, then its other endpoint d is in one of the cells C_1 or C_2; see Figure 8(b). In this case, we replace (a, b) and (c, d) by an internal edge (b, d) and an s-edge (a, c).
- If (c, d) is an internal edge, then consider the other endpoint d of (c, d). If d is not in a danger zone in C_1 defined by another d-edge, then we replace (a, b) and (c, d) by two s-edges (a, d) and (b, c); see Figure 8(c). If d is in a danger zone in C_1 defined by another d-edge (a', b'), then (a', b') is associated with one of the two corners of C_1 adjacent to the corner r; see Figure 8(d). Therefore, either C or C' contains an endpoint of both (a, b) and (a', b'). We replace (a, b) and (c, d) by two s-edges (a, d) and (b, c).

Rule 4: This rule is applied on a d-edge (a, b) and an s-edge (c, d), such that a and c are in the same cell, and b and d are in two adjacent cells that share a side; see Figure 9(a). We replace (a, b) and (c, d) by an internal edge (a, c) and an s-edge (b, d).
Rule 5: This rule is applied on two s-edges (a, b) and (c, d), such that a and c are in the same cell, and b and d are in the same cell; see Figure 9(b). We replace (a, b) and (c, d) by two internal edge (a, c) and (b, d).

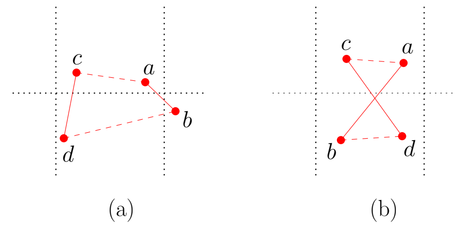

Figure 9: (a) Rule 4, and (b) Rule 5.

Lemma 6 (Lemma 3.2 in [2]) *Let M'_P be $M'_R \cup M'_B$. Then, M'_P has the following properties.*

1. *An edge is either contained in a single cell, or connects between a pair of points in two adjacent cells.*
2. *A corner of the grid has at most one d-edge of M'_R and at most one d-edge of M'_B associated with it.*
3. *A d-edge in M'_P is of length at most λ^*_R.*
4. *The two danger zones defined by a d-edge of M'_R (resp., M'_B) are empty of points of R (resp., of B).*
5. *Each cell C contains at most 4 external edges of M'_R and 4 external edges of M'_B, and $deg(C) \leq 8$.*
6. *If a d-edge in M'_R (resp., in M'_B) connecting between cells C_1 and C_2, and C is a cell sharing a side with both C_1 and C_2, then there is no s-edge in M'_R (resp., in M'_B) connecting between C and either C_1 or C_2.*

Stage 2

In this stage, we construct a $(2\sqrt{10}\lambda^*_R)$-monochromatic matching M_P of P based on $M'_P = M'_R \cup M'_B$. We consider each cell separately. Let C be a non-empty grid cell and let R_C (resp., B_C) be the set of points of R (resp., B) lying in C. Recall that there are at most 4 external edges of M'_R and at most 4 external edges of M'_B associated with C. We use the same procedure of [2] to select the points in R_C (resp., in B_C) that will serve as endpoints of external edges of M'_R (resp., of M'_B), such that the external edges of M'_R (resp., of M'_B) that will be connected to these point do not cross each other. For each external edge e of M'_R (resp., of

M'_B) connecting between two cells C_1 and C_2, let a and b be the points that were chosen as the endpoint of e in C_1 and C_2, respectively. We add the edge (a,b) to M_R (resp., to M_B). It has be proved in [2] that the length of (a,b) is at most $2\sqrt{10}\lambda^*_R$.

Let $R^E_C \subseteq R_C$ (resp., $B^E_C \subseteq B_C$) be the set of points of R_C (resp., of B_C) that were chosen as endpoints of external edges, and let $R^I_C = R_C \setminus R^E_C$ (resp., $B^I_C = B_C \setminus B^E_C$). By the way we select the points of R^E_C (resp., of B^E_C), the points in R^I_C (resp., B^I_C) are contained in a convex polygon $X_R \subseteq C$ (resp., $X_B \subseteq C$), such that any external edge of M'_R (resp., M'_B) with endpoint in R^E_C (resp., in R^E_C) does not intersect the interior of X_R (resp., X_B). For each cell C, we compute a minimum weight matching $M(R^I_C)$ (resp., $M(R^I_C)$) of the points in R^I_C (resp., in B^I_C) and we add it to M_R (resp., to M_B). Clearly, the edges of $M(R^I_C)$ do not cross each other and do not cross the external edges that are connected to points from R^E_C, and the edges of $M(B^I_C)$ do not cross each other and do not cross the external edges that are connected to points from B^E_C. Moreover, the length of each edge in $M(R^I_C)$ and in $M(B^I_C)$ is at most $4\lambda^*_R$.

Let M_P be $M_R \cup M_B$. Since M_R (resp., M_B) is a plane matching and each edge in M_R (resp., in M_B) is of length at most $2\sqrt{10}\lambda^*_R$, M_P is a $(2\sqrt{10}\lambda^*_R)$-monochromatic matching of P. We now show that the number of intersections in M_P, i.e., between the edges of M_R and the edges of M_B, is $O(|P|)$.

We bound the number of intersections for each cell separately. In each cell C, there would be three types of intersections: intersection between an external edge of M_R and an external edge of M_B, intersection between an external edge of M_R and an internal edge of M_B (and vice versa), and intersection between an internal edge of M_R and an internal edge of M_B. We bound the number of intersection of each type separately.

Recall that the number of external edges of M_R (resp., of M_B) that have one endpoint in R^E_C (resp., in B^E_C) is at most 4. This implies that each external edge of M_R can be intersected by at most 4 external edges of M_B, and thus the total number of intersections between the external edges of M_P that have endpoint in C is at most $4|R^E_C|$. Moreover, the total number of intersections between the external edges of M_P that have endpoint in C and the internal edges of $M(R^I_C) \cup M(B^I_C)$ is at most $2(|R^I_C| + |B^I_C|)$. Finally, the total number of intersections between the internal edges of $M(R^I_C)$ and the internal edges of $M(B^I_C)$ is at most $(|R^I_C| + |B^I_C|)/2$, by Theorem 1 in [10]. Thus, the number of intersections produced by the points in C is at most $4(|R_C| + |B_C|)$. Therefore, the number of intersections in M_P is at most $4(|R| + |B|) = 4|P|$. This completes the proof. $\square$

References

[1] A. K. Abu-Affash, A. Biniaz, P. Carmi, A. Maheshwari, and M. H. M. Smid. Approximating the bottleneck plane perfect matching of a point set. *Comput. Geom.*, 48(9):718–731, 2015.

[2] A. K. Abu-Affash, P. Carmi, M. J. Katz, and Y. Trabelsi. Bottleneck non-crossing matching in the plane. *Comput. Geom.*, 47(3):447–457, 2014.

[3] G. Aloupis, J. Cardinal, S. Collette, E. D. Demaine, M. L. Demaine, M. Dulieu, R. Fabila-Monroy, V. Hart, F. Hurtado, S. Langerman, M. Saumell, C. Seara, and P. Taslakian. Matching points with things. In *LATIN, volume 6034 of LNCS*, pages 456–467, 2010.

[4] J. G. Carlsson, B. Armbruster, S. Rahul, and H. Bellam. A bottleneck matching problem with edge-crossing constraints. *Int. J. Comput. Geometry Appl.*, 25(4):245–262, 2015.

[5] M. S. Chang, C. Y. Tang, and R. C. T. Lee. Solving the euclidean bottleneck matching problem by k-relative neighborhood graphs. *Algorithmica*, 8(1–6):177–194, 1992.

[6] A. Efrat, A. Itai, and M. J. Katz. Geometry helps in bottleneck matching and related problems. *Algorithmica*, 31(1):1–28, 2001.

[7] B. Joeris, I. Urrutia, and J. Urrutia. Geometric spanning cycles in bichromatic point sets. *Graphs and Combinatorics*, 31(2):453–465, 2015.

[8] M. Kano, C. Merino, and J. Urrutia. On plane spanning trees and cycles of multicolored point sets with few intersections. *Inf. Process. Lett.*, 93(6):301–306, 2005.

[9] L. Lovász and M. D. Plummer. *Matching Theory.* Elsevier Science Ltd, 1986.

[10] C. Merino, G. Salazar, and J. Urrutia. On the intersection number of matchings and minimum weight perfect matchings of multicolored point sets. *Graphs and Combinatorics*, 21(3):333–341, 2005.

[11] S. Tokunaga. Intersection number of two connected geometric graphs. *Inf. Process. Lett.*, 59(6):331–333, 1996.

[12] P. M. Vaidya. Geometry helps in matching. *SIAM J. Comput.*, 18(6):1201–1225, 1989.

[13] K. R. Varadarajan. A divide-and-conquer algorithm for min-cost perfect matching in the plane. In *FOCS*, pages 320–331, 1998.