The Min-Max Voronoi Diagram of Polygons and Applications in VLSI Manufacturing

Evanthia Papadopoulou[1] and D. T. Lee[2]

[1] IBM TJ Watson Research Center, Yorktown Heights, NY 10598, USA
evanthia@watson.ibm.com
[2] Institute of Information Science, Academia Sinica, Nankang, Taipei, Taiwan
dtlee@iis.sinica.edu.tw

Abstract. We study the *min-max Voronoi diagram* of a set S of polygonal objects, a generalization of Voronoi diagrams based on the *maximum* distance between a point and a polygon. We show that the min-max Voronoi diagram is equivalent to the Voronoi diagram under the Hausdorff distance function. We investigate the combinatorial properties of this diagram and give improved combinatorial bounds and algorithms. As a byproduct we introduce the *min-max hull* which relates to the min-max Voronoi diagram in the way a convex hull relates to the ordinary Voronoi diagram.

1 Introduction

Given a set S of polygonal objects in the plane their *min-max* Voronoi diagram is a subdivision of the plane into regions such that the Voronoi region of a polygon $P \in S$ is the locus of points t whose maximum distance from (any point in) P is less than the maximum distance from any other object in S. The min-max Voronoi region of P is subdivided into finer regions by the farthest point Voronoi diagram of the vertex set of P. This structure generalizes both the ordinary Voronoi diagram of points and the farthest-point Voronoi diagram. The ordinary Voronoi diagram is derived if shapes degenerate to points and the farthest-point Voronoi diagram appears in the case where the set of points is the set of vertices of a single polygon ($|S| = 1$). The *min-max* Voronoi diagram can be defined equivalently on a collection S of sets of points instead of polygonal objects. It is equivalent to the Voronoi diagram of sets of points or polygonal objects under the Hausdorff metric (see Section 2).

The min-max Voronoi diagram problem was formulated in [11] where the *critical area* computation problem for *via-blocks* in VLSI designs was addressed via the L_∞ min-max Voronoi diagram of disjoint rectilinear shapes. This diagram had also been considered in [4] where it was termed the *Voronoi diagram of point clusters*, and in [1] where it was termed the *closest covered set diagram* for the case of disjoint convex shapes and arbitrary convex distance functions. In [4] general combinatorial bounds regarding this diagram were derived by means of envelopes in three dimensions. It was shown that the size of this diagram is

$O(n^2\alpha(n))$ in general and linear in the case of clusters of points with disjoint convex hulls. The latter was also shown in [1] for disjoint convex shapes and arbitrary convex distances. Using a divide and conquer algorithm for computing envelopes in three dimensions, [4] concluded that the cluster Voronoi diagram can be constructed in $O(n^2\alpha(n))$ time. In [1] the problem for disjoint convex sets was reduced to abstract Voronoi diagrams and the randomized incremental construction of [6] was proposed for its computation. This approach results in an $O(kn\log n)$ -time algorithm, where k is the time to construct the bisector of two convex polygons.

In this paper we provide tighter combinatorial bounds and algorithms that improve in an output sensitive fashion the time complexity bounds given in [4] and [1]. Specifically we show that the size of the diagram is $O(m)$ where m is the size of the *intersection graph* of S which reflects the number of relevant intersections among the shapes of S in addition to the number of convex hull edges of each shape in S (see Def. 7). In the worst case m is $O(n^2)$. Furthermore we show that in case of *non-crossing* polygons (not necessarily disjoint or convex, see Def. 5) the min-max Voronoi regions are connected and the size of the min-max Voronoi diagram is linear in the number of vertices on the convex hulls of the polygons in S. Thus, the connectivity and linearity of the diagram is maintained for a more general class of polygons than the ones shown in [4, 1]. We present a divide and conquer algorithm of time complexity $O((n+M+N+K)\log m)$ where n is the total number of points in S, M is the total number of vertices of *crossing* pairs of shapes (see Def. 5), K is the total number of vertices of *interacting*[3] pairs of shapes and $N = O(M\log m)$ is the sum for all dividing lines of the size of pairs of crossing shapes that intersect a dividing line in the divide and conquer scheme. As a byproduct we introduce the *min-max hull* of a set of shapes which relates to the min-max Voronoi diagram in the way a convex hull relates to the ordinary Voronoi diagram (see Def. 9). The special properties of the min-max Voronoi diagram are heavily exploited to maintain efficiency as the *merge curve* of the standard divide-and-conquer technique contains cycles. The algorithm simplifies to $O((n+N+K)\log n)$ for *non-crossing* polygons which is the case of interest to our VLSI manufacturing application. In our application, VLSI contact shapes tend to be small in size and well spaced in which case N, K are both negligible compared to n. A plane sweep approach of time complexity $O((n+K')\log n)$ for a non-crossing S is presented in a companion paper [10].

Our motivation for studying the min-max Voronoi diagram comes from an application in VLSI *yield* prediction, in particular the estimation of *critical area*, a measure reflecting the sensitivity of a VLSI design to spot defects during manufacturing [7–9, 11, 13]. In [12, 11] the critical area computation problem for *shorts*, *opens*, and *via-blocks* was reduced to variations of L_∞ Voronoi diagrams of segments. In particular, the L_∞ min-max Voronoi diagram of disjoint rectilinear shapes was introduced as a solution to the critical area computation problem for via-blocks [11]. The L_∞ metric corresponds to a square defect model. The

[3] A shape $Q \in S$ is *interacting* with $P \in S$ if Q is entirely enclosed in the minimum enclosing circle of P (Def. 11).

square defect model has been criticized [8] for the case of via-blocks and thus the Euclidean version of the problem needs to be investigated. The construction of the Euclidean min-max Voronoi diagram remains realistic as robustness issues are similar to the construction of Voronoi diagrams of points and not to those of segments.

2 Preliminaries

Let $d(p,q)$ denote the ordinary distance between two points in the plane p and q. The ordinary bisector between p and q, denoted as $b(p,q)$, is the locus of points equidistant from p and q. The bisector $b(p,q)$ partitions the plane into two *half-planes* $H(p,q)$ and $H(q,p)$, where $H(p,q)$ denotes the half plane associated with p. The *farthest distance* of a point p from a shape Q is $d_f(p,Q) = \max\{d(p,q), \forall q \in Q\}$. It is well known that $d_f(p,Q) = d(p,q)$ for some vertex q on the convex hull of Q. The convex hull of Q is denoted as $CH(Q)$. It is also well known that $d_f(p,Q)$ can be determined by the *farthest point* Voronoi diagram of the vertex set of Q, denoted as *f-Vor(Q)*. Let *freg*(q) denote the farthest Voronoi region of $q \in CH(Q)$, where q is on $CH(Q)$. The farthest distance between two polygons P and Q is $d_f(P,Q) = \max\{d(p,q), \forall (p,q), p \in P, q \in Q\}$. Clearly $d_f(P,Q) = d(p,q)$ where p and q are vertices of $CH(P)$ and $CH(Q)$ respectively. For two arbitrary sets of points A, B $d_f(A,B)$ is defined equivalently.

Definition 1. *The farthest bisector, denoted $b_f(P,Q)$, between P and Q is the locus of points equidistant from P and Q according to $d_f(P,Q)$, i.e., $b_f(P,Q) = \{y \mid d_f(y,P) = d_f(y,Q)\}$. Any point r such that $d_f(r,P) < d_f(r,Q)$ is said to be closer to P than to Q. The locus of points closer to P than to Q is denoted by $H(P,Q)$.*

$H(P,Q)$ need not be connected in general. We now show that $b_f(P,Q)$ is equivalent to the bisector of P,Q under the Hausdorff metric. The (directed) Hausdorff distance from P to Q is $d_h(P,Q) = \max_{p\in P}\min_{q\in Q} d(p,q)$. The (undirected) Hausdorff distance between P and Q is $D_h(P,Q) = \max\{d_h(P,Q), d_h(Q,P)\}$. The Hausdorff bisector between P and Q is $b_h(P,Q) = \{y \mid D_h(y,P) = D_h(y,Q)\}$. But for any point y, $d_h(y,P) = d(y,P)$, where $d(y,P) = \min_{p\in P}\{d(y,p)\}$, and $d_h(P,y) = d_f(y,P)$. But $d_f(y,P) \geq d(y,P)$. We thus conclude that $b_f(P,Q)$ and $b_h(P,Q)$ are equivalent.

Definition 2. *The* min-max *Voronoi diagram of S, denoted as mM-Vor(S), is a subdivision of the plane into regions such that the min-max Voronoi region of a polygon P, denoted as mM-$reg(P)$, is the locus of points closer to P according to d_f, than to any other shape in S i.e., $mM\text{-}reg(P) = \{y \mid d_f(y,P) \leq d_f(y,Q), \forall Q \in S, Q \neq P\}$. The min-max Voronoi region of P is subdivided into finer regions by the farthest point Voronoi diagram of the vertex set of P. That is, $mM\text{-}reg(p) = mM\text{-}reg(P) \cap freg(p)$ i.e., $mM\text{-}reg(p) = \{y \mid d(y,p) = d_f(y,P) \leq d_f(y,Q), \forall Q \in S, Q \neq P\}$. The Voronoi edges on the boundary of mM-$reg(P)$ are called inter-bisectors. The bisectors in the interior of mM-$reg(P)$ are called intra-bisectors. Since $b_f(P,Q)$ is equivalent to $b_h(P,Q)$ the min-max Voronoi diagram is also called the Hausdorff Voronoi diagram.*

Figure 1 illustrates the min-max Voronoi diagram of $S = \{P_1, P_2, P_3\}$. The shaded regions depict $mM\text{-}reg(P_1)$ and $mM\text{-}reg(P_3)$; the unshaded portion corresponds to $mM\text{-}reg(P_2)$. Inter-bisectors are shown in solid lines and intra-bisectors are depicted in dashed lines. Figure 2 illustrates the min-max Voronoi diagram of two intersecting segments P, Q; $mM\text{-}reg(P)$ is illustrated shaded. Both an intra- and an inter-bisector correspond to an ordinary bisector $b(p, q)$ between two points p, q. For an inter-bisector, p and q belong to different shapes, $H(p, q)$ consists of all points closer to p than q, and p is located in $H(p, q)$. For an intra-bisector, p, q are points of the same shape, $H(p, q)$ consists of points farther from p than q and p is not located in $H(p, q)$. Similarly we distinguish between three types of vertices: *inter-vertices* where at least three inter-bisectors meet, *intra-vertices* where at least three intra-bisectors meet, and *mixed-vertices* where one intra-bisector and two inter-bisectors meet. The term *intra-bisector* is used to denote any bisector in *f-Vor(P)*, the farthest point Voronoi diagram of the vertex set of P.

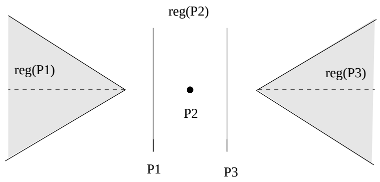

Fig. 1. The min-max Voronoi diagram of $S = \{P_1, P_2, P_3\}$.

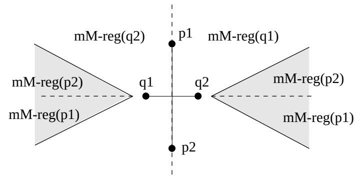

Fig. 2. The min-max Voronoi diagram of two intersecting segments.

Definition 3. *The circle $\mathcal{K}_y$, centered at an intra-bisector point y of radius $d_f(y, P)$ is called a P-*circle*. A P-circle is* empty *if it contains no shape other than P in its interior.*

Definition 4. *A* supporting line *of a convex polygon P is a straight line l passing through a vertex v of P such that the interior of P lies entirely on one side of l. Vertex v is called a* supporting vertex. *l and v are called* left *(resp.* right*)* supporting*) if P lies to the right (resp. left) of l. The portion of the common supporting line between the supporting vertices of two convex polygons such that both polygons lie on the same side of l is called a* supporting segment.

Definition 5. *Two polygons are called* non-crossing *if their convex hulls admit at most two supporting segments. Otherwise they are called* crossing.

An example of non-crossing and crossing polygons can be seen in Figure 3. In Figures 3(a) and 3(b) all polygons are non-crossing; in Figure 3(c) polygons C and D are crossing. In our application, VLSI via-shapes are disjoint and thus, their convex hulls must be non-crossing.

Definition 6. *A* chord *$\overline{p_i p_j} \in P$ is a diagonal or convex hull edge of P that induces an intra-bisector of f-Vor(P). Two chords $\overline{p_i p_j} \in P$ and $\overline{q_i q_j} \in Q$, are called* non-crossing *if they do not intersect or if they intersect but at least one* *of their endpoints lies strictly in the interior of $CH(P \cup Q)$. Otherwise they are called* crossing. *Shape Q is called* crossing *with chord $\overline{p_i p_j} \in P$ if there is a chord $\overline{q_i q_j} \in Q$ that is crossing with $\overline{p_i p_j}$. Otherwise it is called* non-crossing.

Definition 7. *The* intersection graph *of S, $G(S)$, has a vertex for every chord of a shape $P \in S$ and an edge for any two crossing chords $\overline{p_i p_j} \in P, \overline{q_i q_j} \in Q$. The size of the intersection graph is denoted by m. The total number of vertices of all crossing pairs of shapes is denoted by M.*

Definition 8. *A chain is a simple polygonal line. A chain C is monotone with respect to a line l if every line orthogonal to l intersects C in at most one point.*

3 Structure and Properties

In this section we list the structural properties of *mM-Vor(S)*. Unless we explicitly mention otherwise these properties have not been identified in [4, 1].

Property 1. *The farthest bisector $b_f(P, Q)$ is a subgraph of* f-Vor$(P \cup Q)$ *consisting of edge disjoint monotone chains. If a chain has just one edge, this is a straight line; otherwise its two extreme edges are semi-infinite rays.*

Corollary 1. *Any semi-infinite ray of $b_f(P, Q)$ corresponds to the perpendicular bisector induced by a distinct supporting segment between $CH(P)$ and $CH(Q)$. The number of chains constituting $b_f(P, Q)$ is derived by the number of supporting segments between $CH(P)$ and $CH(Q)$.*

Property 2. *For any point $x \in mM\text{-}reg(p)$ the segment $\overline{px} \cap freg(p)$ lies entirely in $mM\text{-}reg(p)$. $mM\text{-}reg(p)$ is said to be essentially star-shaped.*

Property 3. *The boundary of any connected component of $mM\text{-}reg(p)$, $p \in P$, $p \neq P$, consists of a sequence of outward convex chains, each one corresponding to an inter-bisector $b_f(p, Q_i)$, $Q_i \in S, Q_i \neq P$, and an inward convex chain corresponding to the intra-bisector $b_f(p, P)$. (Convexity is characterized as seen from the interior of $mM\text{-}reg(p)$).*

Let's now define the *min-max hull*, or *mM-hull* for short, of S, denoted as $mMH(S)$. The mM-hull is related to the *mM-Vor(S)* as the ordinary convex hull is related to the ordinary Voronoi diagram. An example is depicted in Figure 3.

Definition 9. *A shape $P \in S$, in particular a vertex $p \in P$, is said to be on the mM-hull of S, if and only if p admits a supporting line ℓ such that $CH(P)$ lies totally on one side of ℓ and none of the rest of shapes in S lie totally on the same side, except possibly from a shape having its boundary common to ℓ and its interior lying on the same side of ℓ as $CH(P)$. A segment $\overline{pq}$ joining two mM-hull vertices $p \in P, q \in Q$ such that $\overline{pq}$ is a supporting segment of P, Q is called an mM-hull* supporting segment. *An mm-hull edge is either a convex hull edge joining mM-hull vertices of one shape or an mM-hull supporting segment joining mM-hull vertices of different shapes.*

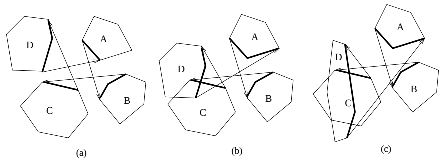

Fig. 3. The min-max hull.

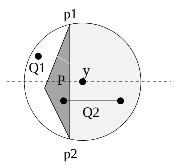

Fig. 4. $Q_1 \in \mathcal{K}_y^r$ and $Q_2 \in \mathcal{K}_y^f$.

By Definition 9, a supporting segment $\overline{pq}$, $p \in P$ and $q \in Q$, is an edge of $mMH(S)$ if and only if $CH(P)$ and $CH(Q)$ lie totally on one side of the line $\ell_{\overline{pq}}$ passing through $\overline{pq}$ and no other shape in S lies totally on the same side of $\ell_{\overline{pq}}$ (except possibly from a shape having its boundary common to $\ell_{\overline{pq}}$). Furthermore, a convex hull edge $\overline{pr}$, $p, r \in P$, is an mM-hull edge if and only if $CH(P)$ is the only shape in S lying entirely on one side of the underlying line $\ell_{\overline{pr}}$. Thus, the boundary of the mM-hull consists of a sequence of supporting segments, interleaved with a subsequence of convex chains. The order of traversal, say clockwise, of the mM-hull edges satisfies the property that each edge defines a supporting line of a shape P on the mM-hull: a right supporting line for mM-hull supporting segments and a left supporting line for convex hull chains. Figure 3 illustrates the mM-hulls of (a) disjoint, (b) non-crossing, and (c) crossing shapes respectively. Supporting segments are illustrated in arrows according to a clockwise traversal.

Property 4. *Region mM-reg(p) is unbounded if and only if vertex p lies on the mM-hull of S.*

Corollary 2. *All unbounded bisectors of mM-Vor(S) (both inter- and intra-bisectors) are cyclically ordered in the same way as the edges are ordered on the boundary of $mMH(S)$.*

Let $T(P)$ denote the tree of intra-bisectors of P i.e., the tree induced by *f-Vor(P)*. $T(P)$ is assumed to be rooted at the point of minimum weight i.e., the center of the minimum enclosing circle of P. Let $\overline{y_j y_k} \in T(P)$ be the intra-bisector segment of chord $\overline{p_i p_j}$, where y_j is the parent of y_k, and let y_i be a point on $\overline{y_j y_k}$. Point y_i partitions $T(P)$ in at least two parts. Let $T(y_i)$ denote the part containing the descendents of y_i in the rooted $T(P)$ i.e., the subtree of $T(P)$ rooted at y_i that contains segment $\overline{y_i y_k}$, and let $T_c(y_i)$ denote the complement of $T(y_i)$. $T(y_i)$ is referred to as the *subtree of $T(P)$ rooted at y_i*. Let $\mathcal{K}_{y_i}$ be the P-circle centered at y_i. Chord $\overline{p_i p_j}$ partitions $\mathcal{K}_{y_i}$ in two parts $\mathcal{K}_{y_i}^f$ and $\mathcal{K}_{y_i}^r$, where $\mathcal{K}_{y_i}^r$ is the part enclosing the portion of $CH(P)$ inducing $T(y_i)$ and $\mathcal{K}_{y_i}^f$ is the part enclosing the portion of $CH(P)$ inducing $T_c(y_i)$. Figure 4 depicts $\mathcal{K}_{y_i}^f$ shaded.

Definition 10. *A shape $Q \in S$ is called* limiting *with respect to chord $\overline{p_i p_j} \in P$, if Q is enclosed within a P-circle $\mathcal{K}$ passing through p_i, p_j, and Q is non-crossing with $\overline{p_i p_j}$. (Note that Q may be limiting but still be crossing with P). Q is called* forward limiting *if $Q \in \mathcal{K}_{y_i}^f \cup CH(P)$ or* rear limiting *if $Q \in \mathcal{K}_{y_i}^r \cup CH(P)$.*

Note that any shape enclosed in a P-circle must be forward limiting, rear limiting or crossing with P. In Figure 4 shapes Q_1 and Q_2 are rear and forward limiting respectively with respect to y and chord $\overline{p_1p_2}$.

Lemma 1. *Let $Q \in S$ be a limiting shape with respect to chord $\overline{p_ip_j} \in P$ and let $y \in T(P)$ be the center of the P-circle enclosing Q. If Q is forward (resp. rear) limiting then the whole $T(y)$ (resp. $T_c(y)$) is closer to Q than to P.*

Proof. Sketch. It is not hard to see that $\mathcal{K}_y^f \cup CH(P) \subset \mathcal{K}_{y_k}$ for any $y_k \in T(y)$, and $\mathcal{K}_y^r \cup CH(P) \subset \mathcal{K}_{y_j}$ for any $y_j \in T_c(y)$.

The following property gives a sufficient condition for a shape to have an empty Voronoi region and it is directly derived from Lemma 1. In the case of non-crossing shapes the condition is also necessary. For the subset of disjoint convex shapes property 5 had also been identified in [1].

Property 5. *$mM\text{-}Vor(P) = \emptyset$ if there exist two limiting shapes Q, R, enclosed in the same P-circle $\mathcal{K}_y$, $y \in b(p_i, p_j)$, $\overline{p_ip_j} \in P$, such that $Q \in \mathcal{K}_y^f \cup CH(P)$ and $R \in \mathcal{K}_y^r \cup CH(P)$. In case of a non-crossing S the condition is also necessary (except from the trivial case where $CH(P)$ entirely contains another shape).*

Theorem 1. *The min-max Voronoi diagram of an arbitrary set of polygons S has size $O(m)$, where m is the size of the intersection graph of S. In case of a non-crossing S, there is at most one connected Voronoi region for each $P \in S$ and mM-Vor(S) has size $O(n)$, where n is the total number of vertices in S.*

Proof. Sketch. It is enough to show that the number of mixed Voronoi vertices is $O(m)$. For any P at most two mixed Voronoi vertices can be induced by limiting shapes. Any other mixed Voronoi vertex on $b(p_i, p_j), p_i, p_j \in P$ must be induced by some $Q \in S$ that is crossing $\overline{p_ip_j}$. But for $\overline{p_ip_j}$ at most two mixed Voronoi vertices can be induced by Q. Thus, the bound is derived.

For a non-crossing S any shape enclosed in a P-circle must be limiting. By property 3 the boundary of any connected component of $mM\text{-}reg(p)$ for any $p \in P$ must contain a single intra-bisector chain of $T(P)$. Thus, by lemma 1 and the non-crossing property, $T(P) \cap mM\text{-}reg(P)$ must consist of a single connected component (if not empty). Hence, $mM\text{-}reg(P)$ must be connected.

In the case of disjoint convex shapes the connectivity and linearity of *mM-Vor(S)* had also been established in [4] and [1]. Our proof however is much simpler and extends the linearity property to the more general class of non-crossing polygons.

4 A Divide and Conquer Algorithm

Let S_l and S_r be the sets of shapes in S to the left and to the right respectively of a vertical dividing line $\mathcal{L}$, where a shape P is said to be to the *left* (resp. *right*) of $\mathcal{L}$ if the leftmost x-coordinate of P is to the left (resp. right) of $\mathcal{L}$. Assume that *mM-Vor*(S_l) and *mM-Vor*(S_r) have been computed. We shall compute *mM-Vor*(S) by merging *mM-Vor*(S_l) and *mM-Vor*(S_r). Let $\sigma(S_l, S_r)$ denote the *merge curve* between $mM\text{-}Vor(S_l)$ and $mM\text{-}Vor(S_r)$ i.e., the collection of bisectors $b(p_l, p_r)$ in $mM\text{-}Vor(S)$ such that $p_l \in S_l$ and $p_r \in S_r$. Let $S_{\mathcal{L}}$ consist of the shapes in S_l that intersect the dividing line and the shapes in S_r that are crossing with shapes in S_l. In case of non-crossing shapes $S_{\mathcal{L}} \subseteq S_l$.

Lemma 2. *The merge curve $\sigma(S_l, S_r)$ is a collection of edge disjoint unbounded chains and cycles. The cycles can only enclose regions of shapes in $S_{\mathcal{L}}$ that is, regions of shapes in S_l that intersect the dividing line and regions of shapes in S_r that are crossing with shapes in S_l (if any).*

Figure 5 shows *mM-Vor(S)*, $S = \{P_1, P_2, Q_1, Q_2\}$ with $mM\text{-}reg(P_2)$ depicted shaded. Dividing S as $S_l = \{P_1, P_2\}$ and $S_r = \{Q_1, Q_2\}$ shows that $\sigma(S_l, S_r)$ can have cycles enclosing regions of shapes in S_l. In Figure 6 the shaded regions depict $mM\text{-}reg(Q)$. Dividing S as $S_l = \{P_1, P_2\}$ and $S_r = \{Q\}$ shows that, in case of crossing shapes, a cycle in $\sigma(S_l, S_r)$ can enclose regions of shapes in S_r.

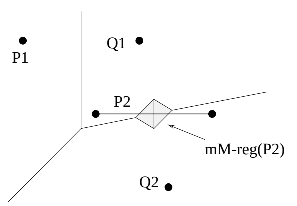

Fig. 5. *mM-Vor(S)*, $S = \{P_1, P_2, Q_1, Q_2\}$.

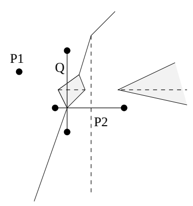

Fig. 6. *mM-Vor(S)*, $S = \{P_1, P_2, Q\}$.

To trace the merge curve $\sigma(S_l, S_r)$ we need to identify a starting point on every component. Then each component can be traced as in the ordinary Voronoi diagram case [14]. By Corollary 2 the unbounded portions of $\sigma(S_l, S_r)$ are induced by the mM-hull supporting segments between vertices of S_l and S_r. We first concentrate on identifying those mM-hull supporting segments. We then show how to identify a point on every cycle of $\sigma(S_l, S_r)$.

4.1 Merging two mM-hulls

The mM-hull of S is represented by an ordered (e.g. clockwise) list of its vertices. An alternative representation can be obtained by the *G*aussian Map [3] of S, for short *GMap(S)*, onto the unit circle. In the Gaussian Map every mM-hull edge e_i is mapped to a point $\nu(e_i)$ on the circumference of a unit circle $\mathcal{K}_o$ as obtained by the outward pointing unit normal. That is, point $\nu(e_i)$ represents a normal vector pointing in the half-plane bordered by the supporting line ℓ_{e_i} away from the mM-hull i.e., towards the interior of the shape(s) where the endpoints of e_i belong. We will use the same notation to denote both the vector and the corresponding point in the GMap. Figure 7 illustrates two mM-hulls and their respective Gaussian maps. In the Gaussian maps the supporting segments of the mM-hull are represented by longer arrows and are marked by the names of the two shapes they support.

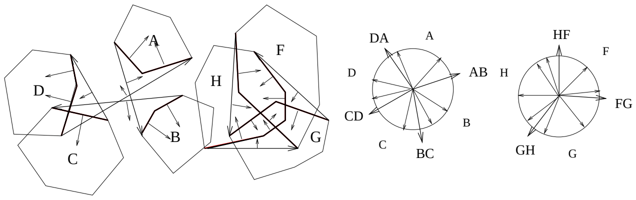

Fig. 7. The Gaussian Map of *mM-hull*($\{A, B, C, D\}$) and *mM-hull*($\{F, H, G\}$).

Lemma 3. *The normal vectors in GMap(S) appear in the same cyclic order (e.g. clockwise) as the respective cyclic traversal of the boundary of mM-hull(S).*

By Lemma 3 merging two mM-hulls corresponds to merging their Gaussian Maps. An edge e of $mMH(S_l)$ or $mMH(S_r)$ is called *valid* if e remains on the $mMH(S)$, otherwise e is called *invalid*. To merge $GMap(S_l)$ and $GMap(S_r)$ we merge their valid portions in cyclic order, and add vectors of the new supporting segments between $mMH(S_l)$ or $mMH(S_r)$. We will only provide the self-explanatory Figure 8 which illustrates the merging process of the two mM-hulls appearing in Figure 7. In particular, Figure 8(a) illustrates $mMH(S), S = S_l \cup S_r, S_l = \{A, B, C, D\}, S_r = \{G, F, H\}$, Figure 8(b) illustrates $GMap(S)$, and Figures 8(c) and 8(d) illustrate $GMap(S_l)$ and $GMap(S_r)$ respectively. In Figures 8(c) and 8(d) the invalid vectors are shown crossed out. The new vectors corresponding to the new supporting segments between $mMH(S_l)$ and $nMH(S_r)$ are indicated by thicker arrows.

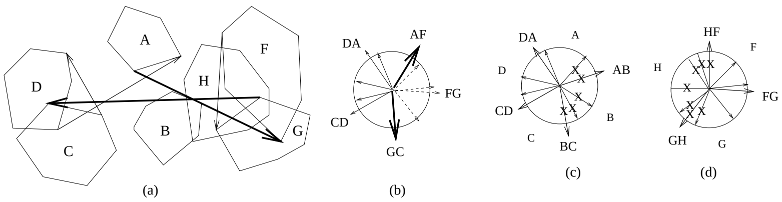

Fig. 8. Merging $mMH(S_l)$ and $mMH(S_r)$.

Theorem 2. *Merging* $mMH(S_l)$ *and* $mMH(S_r)$ *into* $mMH(S)$ *can be computed in linear time.*

4.2 Tracing the cycles of $\sigma(S_l, S_r)$

Our goal is to identify a starting point on every cycle of $\sigma(S_l, S_r)$. Let $S_P = S_l$ and $S_Q = S_r$ (resp. $S_P = S_r, S_Q = S_l$) for any $P \in S_l \cap S_{\mathcal{L}}$ (resp. $P \in S_r \cap S_{\mathcal{L}}$). Let $mM\text{-}reg'(p)$, $p \in P$, denote the region of p in *mM-Vor*(S_P) or *mM-Vor*(S_Q) and $mM\text{-}reg(p)$ denotes the region of p in *mM-Vor*(S). Consider a connected component σ of $\sigma(S_l, S_r)$. It partitions the plane into two parts such that one side of $\sigma(S_l, S_r)$ borders regions of S_l, denoted as $H(S_l, S_r)$, and the other side borders regions of S_r, denoted as $H(S_r, S_l)$. A region of *mM-Vor*(S_l) (resp. *mM-Vor*(S_r)) that falls in $H(S_r, S_l)$ (resp. $H(S_l, S_r)$) is said to be on the *wrong* side of σ and it may be partially enclosed by a cycle of $\sigma(S_l, S_r)$.

Let $P \in S_{\mathcal{L}}$ have a region in *mM-Vor*(S_P) on the wrong side of the components of $\sigma(S_l, S_r)$ computed so far. Let $T'(P) \subseteq T(P)$ be the intra-bisector tree derived by one connected component of $mM\text{-}reg'(P)$ in $H(S_Q, S_P)$. We assume that $T'(P)$ is rooted at the point of minimum weight. Note that in the general crossing case there may be more than one connected component of $mM\text{-}reg'(P)$ but each one is treated independently.

Consider an intra-bisector segment $\overline{y_iy_j} \in T'(P)$ where y_i is an ancestor of y_j in $T'(P)$ (i.e., $d_f(y_i, P) < d_f(y_j, P)$). Let p_i and p_j be the vertices inducing $\overline{y_iy_j}$ (i.e.,$\overline{y_iy_j} \in b(p_i, p_j)$). Let $q_i \in Q_i$ and $q_j \in Q_j$ be the owners of y_i and y_j respectively in *mM-Vor*(S_Q) (i.e., $y_i \in mM\text{-}reg'(q_i)$ and $y_j \in mM\text{-}reg'(q_j)$). Q_i and Q_j may or may not be the same shape. Comparing $d(y_k, p_k)$ and $d(y_k, q_k)$ for $k = i, j$ we can easily determine whether y_k is closer to P or Q_k (i.e., whether $y_k \in mM\text{-}reg(P)$ or $y_k \in mM\text{-}reg(q_k)$ in *mM-Vor*(S)). The following lemma can be derived from lemma 1 and property 5.

Lemma 4. *For an intra-bisector segment $\overline{y_iy_j} \in T'(P)$, and a merge curve cycle τ enclosing a portion of $\overline{y_iy_j}$ (if any) we have the following:*

1. *If $y_i \in mM\text{-}reg(P)$ and $y_j \in mM\text{-}reg(P)$, then τ exists but if S is non-crossing then τ cannot intersect $\overline{y_iy_j}$.*
2. *If exactly one of y_i, y_j belongs in $mM\text{-}reg(P)$ then τ must intersect $\overline{y_iy_j}$.*
3. *If $y_i \in mM\text{-}reg(Q_i)$ and $y_j \in mM\text{-}reg(Q_j)$ and if $Q_i = Q_j$, or if Q_i is forward limiting or if Q_j is rear limiting, then τ cannot intersect $\overline{y_iy_j}$.*
4. *If $y_i \in mM\text{-}reg(Q_i)$ and $y_j \in mM\text{-}reg(Q_j)$ but Q_i is rear limiting and Q_j is forward limiting then either τ intersects $\overline{y_iy_j}$ twice or $mM\text{-}reg(P) = \emptyset$.*

Lemma 5. *Given any point $y_i \in T'(P)$ such that $y_i \in mM\text{-}reg(P)$, a starting point of τ can be determined in time $O((N_\tau + B_\tau) \log n)$ where N_τ denotes the number of regions in S_P enclosed by τ and B_τ denotes the number of intra-bisectors of shapes in S_Q that get eliminated by τ.*

Proof. Sketch. Consider segment $\overline{q_iy_i}$ ($y_i \in mM\text{-}reg'(q_i)$). Let x be the intersection point of $\overline{q_iy_i}$ with the chain B of intra-bisectors bounding $reg'(q_i)$. If x is closer to q_i than the owner of x in *mM-Vor*(S_P) then τ must intersect segment $\overline{y_ix}$. Determine the intersection point by walking on segment $\overline{y_ix}$, starting at y_i, and considering intersections with the regions of *mM-Vor*(S_P) (if any). Otherwise, if x is closer to its owner in *mM-Vor*(S_P), walk along chain B until either the whole chain is determined to be closer to S_P or a point equidistant from q_i i.e., a point on τ, is found. In the former case this connected component of $mM\text{-}reg(q_i)$ must be empty; update y_i to the intersection point, if any, of $\overline{y_iy_j}$ and $mM\text{-}reg(q_i)$ and repeat the process. The method maintains time complexity.

Lemma 6. *For any segment $\overline{y_iy_j} \in T'(P)$ induced by chord $\overline{p_ip_j}$ such that $y_i \in mM\text{-}reg(Q_i)$, $y_j \in mM\text{-}reg(Q_j)$ and Q_i is rear limiting or crossing with $\overline{p_ip_j}$*

and Q_j is forward limiting or crossing with $\overline{p_i p_j}$, the first intersection x of τ with $\overline{y_i y_j}$ (if any) can be determined in time $O(M_\tau \log n)$ where M_τ denotes the total size of rear limiting or crossing shapes shapes enclosed in the P-circles centered along $\overline{y_i x}$.

Proof. Sketch. We traverse segment $\overline{y_i y_j}$, starting at y_i, considering the intersection points of $\overline{y_i y_j}$ with $mM\text{-}Vor(S_Q)$ until either a point closer to P is encountered, or y_j is reached, or a point t on some inter-bisector $b(q_k, q_l)$ $(q_k \in Q_k, q_l \in Q_l)$ of $mM\text{-}Vor(S_Q)$ is reached such that Q_l is forward limiting. In the first case we can easily determine the starting point of a cycle τ. In the second and third case we conclude that τ does not intersect $\overline{y_i y_j}$. For non-crossing shapes the latter observation implies $reg(P) = \emptyset$.

A starting point of τ in the non-crossing case can be found as follows:
Non-crossing case: Identify $S_{\mathcal{L}} \subset S_l$. For every $P \in S_{\mathcal{L}}$ such that $mM\text{-}reg'(P)$ is on the *wrong* side of the unbounded portion of $\sigma(S_l, S_r)$ identify $T'(P)$. Locate the root of $T'(P)$ in $mM\text{-}Vor(S_r)$. If the root remains in $mM\text{-}reg(P)$ identify a starting point of τ as explained in lemma 5. Otherwise visit the vertices of $T'(P)$ in order of increasing weight until either Case 2 or Case 4 of Lemma 4 is encountered. In case 2 follow lemma 5 and in case 4 follow lemma 6.

We now generalize to arbitrarily crossing shapes. Let $T(S_l)$ (resp. $T(S_r)$) denote the collection of all intra-bisector trees induced by the regions of shapes in $S_{\mathcal{L}} \cap S_l$ (resp. $S_{\mathcal{L}} \cap S_r$) that fall on the *wrong side* of the merge curves computed so far.
General case: Locate the vertices of $T(S_l)$ and $T(S_r)$ into $mM\text{-}Vor(S_r)$ and $mM\text{-}Vor(S_l)$ respectively. Process first $T(S_l)$ until empty and then repeat for $T(S_r)$. For each tree $T'(P)$ in $T(S_l)$ or $T(S_r)$ do:

1. For any vertex $y_i \in T(P)$ such that $y_i \in mM\text{-}reg(Q_i)$ and Q_i is a forward (resp. rear) limiting shape, eliminate $T(y_i)$ (resp. $T_c(y_i)$).
2. For any adjacent vertices $y_i, y_j \in T(P)$ such that $y_i \in mM\text{-}reg(Q_i)$ and $y_j \in mM\text{-}reg(Q_j)$ but $Q_i = Q_j$ eliminate segment $\overline{y_i y_j} \in T(P)$.
3. For any vertex $y_i \in T(P)$ such that $y_i \in mM\text{-}reg(P)$, determine a starting point of the cycle enclosing y_i following lemma 5.
4. For any adjacent vertices $y_i, y_j \in T(P)$ such that $y_i \in mM\text{-}reg(Q_i)$ and $y_j \in mM\text{-}reg(Q_j)$ determine the first intersection of $\overline{y_i y_j}$ and a merge cycle (if any) as explained in lemma 6. If a forward (resp. rear) limiting shape is identified during the traversal eliminate $T(y_i)$ (resp. $T_c(y_i)$). Eliminate any portion of $\overline{y_i y_j}$ encountered during the traversal.
5. For any starting point identified in Steps 3 and 4 trace the merge cycle.
6. Update $T(S_P)$ and $T(S_Q)$.
7. Repeat for every tree in $T(S_l)$ until $T(S_l) = \emptyset$. Repeat for every tree in $T(S_r)$. Repeat until both $T(S_l)$ and $T(S_r)$ are empty. (During the processing of $T(S_r)$, $T(S_l)$ may become non-empty and vice versa).

By lemma 1 all rear limiting shapes with respect to a chord of P are enclosed within the minimum enclosing circle of P.

Definition 11. *A shape $Q \in S$ is said to be* interacting *with $P \in S$ if Q is entirely enclosed in the minimum enclosing circle of P i.e., the minimum radius P-circle. Pair (P, Q) is called an* interacting pair. *The total number of vertices of interacting pairs of shapes is denoted by K.*

Theorem 3. *The min-max Voronoi diagram $mM\text{-}Vor(S)$ can be computed by divide and conquer in time $O((n + M + N + K) \log m)$ where n is the number of input points, M is the number of vertices of crossing pairs of shapes, m is the size of the intersection graph $G(S)$, K is the number of vertices of interacting pairs of shapes, and $N = \Sigma_{\mathcal{L}} |S_{\mathcal{L}}|$ for all dividing lines $\mathcal{L}$ (N is $O(M \log m)$). For a non-crossing S, the algorithm simplifies to $O((n + N + K) \log n)$.*

References

1. M. Abellanas, G. Hernandez, R. Klein,V. Neumann-Lara, and J. Urrutia, *Discrete Computat. Geometry* 17, 1997, 307-318.
2. F. Aurenhammer, "Voronoi diagrams: A survey of a fundamental geometric data structure," *ACM Comput. Survey,* 23, 345-405, 1991.
3. L.L. Chen, S.Y. Chou, and T.C. Woo, 1993, "Optimal parting directions for mold and die design," *Computer-Aided Design*, Vol. 26, No. 12, pp.762-768.
4. H. Edelsbrunner, L.J. Guibas, and M. Sharir, "The upper envelope of piecewise linear functions: algorithms and applications", *Discrete Computat. Geometry* 4, 1989, 311-336.
5. R. Klein, "Concrete and Abstract Voronoi Diagrams", vol. 400, *Lecture Notes in Computer Science*, Springer-Verlag, 1989.
6. R. Klein,, K. Melhorn, S. Meiser, "Randomized Incremental Construction of Abstract Voronoi diagrams", *Computational geometry: Theory and Aplications* 3,1993, 157-184
7. W. Maly, "Computer Aided Design for VLSI Circuit Manufacturability," *Proc. IEEE*, vol.78, no.2, 356-392, Feb. 90.
8. C.H. Ouyang, W.A. Pleskacz, W. Maly, "Extraction of Critical Areas for Opens in Large VLSI Circuits", *IEEE Trans. on Computer-Aided Design*, vol. 18, no 2, 151-162, February 1999.
9. B. R. Mandava, "Critical Area for Yield Models", IBM Technical Report TR22.2436, East Fishkill, NY, 12 Jan 1982.
10. E. Papadopoulou, "Plane sweep construction for the min-Max (Hausdorff) Voronoi diagram", Manuscript in preparation.
11. E. Papadopoulou, "Critical Area Computation for Missing Material Defects in VLSI Circuits", *IEEE Transactions on Computer-Aided Design*, vol. 20, no.5, May 2001, 583-597.
12. E. Papadopoulou and D.T. Lee, "Critical Area Computation via Voronoi Diagrams", *IEEE Trans. on Computer-Aided Design*, vol. 18, no.4, April 1999,463-474.
13. E. Papadopoulou and D.T. Lee, "The L_∞ Voronoi Diagram of Segments and VLSI Applications", *International Journal of Computational Geometry and Applications*, Vol. 11, No. 5, 2001, 503-528.
14. Preparata, F. P. and M. I. Shamos, *Computational Geometry: an Introduction*, Springer-Verlag, New York, NY 1985.