

Vindictive Voronoi Games and Stabbing Delaunay Circles

Syed Ishtiaque Ahmed
Department of CSE, BUET
Dhaka-1000, Bangladesh
Email: ishtiaque@csebuuet.org

Masud Hasan*
School of Computer Science
University of Waterloo
Ontario, Canada N2L 3G1
Email: m2hasan@cs.uwaterloo.ca

Awalin Sopan
Department of Computer Science
University of Maryland
College Park, MD 20742, USA
Email: awal@cs.umd.edu

Abstract—In this paper we consider the following problem: Given a set of n Player1 sites in the plane and their Delaunay triangulation \mathcal{D} , place minimum possible Player2 sites such that in the resulting Delaunay triangulation \mathcal{D}' of the sites of both Players, the neighborhood between Player1 sites are as less as possible. We first consider placing minimum number of Player2 sites such that no two Player1 sites are neighbors in \mathcal{D}' . We show that to isolate a Player1 site p , two Player2 sites are both necessary and sufficient if p is in the convex hull of \mathcal{D} , otherwise three Player2 sites are both necessary and sufficient. This gives a linear time algorithm to individually isolate all Player1 sites by $3n - h$ Player2 sites, where h is the number of sites in the convex hull of \mathcal{D} . Then we give two more algorithms for this problem. The next algorithm runs also in linear time and uses $3(n-1) - h$ Player2 sites, but is much simpler. Our next algorithm uses $5|\mathcal{M}|$ sites, where $|\mathcal{M}|$ is the size of a maximum matching in \mathcal{D} , and runs in $O(\sqrt{nm})$ time, where m is the number of edges of \mathcal{D} .

Then we consider isolating sites by components, where a component in \mathcal{D}' is a maximal connected subset of sites of the same player. We show that it is possible to place n Player2 sites such that in \mathcal{D}' the number of components among Player1 sites is higher than that among Player2 sites.

The above problems are related to at least two existing well known research topics: *Voronoi games*, where the strategy of each of the two players is to place sites such that in the resulting Voronoi diagram some certain criteria is optimized for each player, and *proximity graphs*, where this problem is known as *minimum stabbing set of Delaunay circles*. Our bound of $5|\mathcal{M}|$ for the first problem would work better than known bound for the minimum stabbing set of Delaunay circles if \mathcal{M} has a smaller size.

Keywords—competitive facility location; Delaunay triangulation; matching in planar graphs; stabbing set; Voronoi games.

I. INTRODUCTION

Consider a city having multiple shops of a shopping chain, whose owner is called as *Player1*. It is common to assume that customers prefer to go to the shop which is geographically closest to them (provided that the

products are similar in quality). Thus, Voronoi diagram of the shops reasonably models the customers affiliation with a shop, where the customers within a Voronoi region are affiliated with the corresponding shop.

Now imagine that another owner, called *Player2*, of a shopping chain is planning to set up his shops (also called *sites*) in a city where at the moment he does not have any shops but his competitor's, i.e., *Player1*'s, shopping chain has already set up numerous shops. So, the objective of *Player2* would be to put the new shops in locations such that they can “interfere” the business of *Player1* as much as possible, sometimes using same number of shops or sometimes using minimum number of shops possible. There may be different criteria on how to interfere. Note that, knowing that some opponent may place shops in future, *Player1* also had placed his shops in “optimal” way. The problem of finding an optimal way of placing *Player2* shops is known as *Voronoi games* and falls in general under the *competitive facility location problem* [1], [2], [7], [8], [10], [11], [12], [13], [16]. (The survey papers [11], [12], [16] also discuss models other than Voronoi diagrams in competitive facility location problems.)

Voronoi games can be played more precisely in *one round* or in *n-round* with exactly n sites for both players. In one round, *Player1* first places all of his n sites, then *Player2* places his n sites. In an *n-round* game, *Player1* and *Player2* places one site at a time one after another. In both cases, the *winning criteria* is determined in the Voronoi diagram of $2n$ sites.

A well studied winning criterion for Voronoi game is that in the resulting Voronoi diagram, total Voronoi region of a player is maximized [1], [2], [7], [8], [10], [13]. A number of variations within this criterion have been studied so far. Ahn et.al. [1], [2] first studied this problem in one dimension, with the motivation of placing chain stores along highways. They showed that *Player2* can always win but the first player can keep the wining margin as small as possible. For a 2D playing area and for *n-round*, the problem of finding

*On leave from BUET, Bangladesh.

a winning strategy for Player2 is still open. However, several variations have been studied. Dehne, Klein and Seidel [10] studied how a single new site can be placed in an existing n -site Voronoi diagram so that the Voronoi area of the new site is maximized. They formulated the area of the new site as a function of its location. They proved that if the (wouldbe) neighbors of the new sites are in convex position then there can be exactly one maxima of the function and that is where the new site should be placed. However, the problem where the neighbors are in general positions is still open. Cheong, Efrat and Har-Peled [7] presented some approximate solutions to this problem. They gave an algorithm which approximates the Voronoi region of the new site to $(1 - \delta)$ times the optimal, for some $\delta > 0$, and runs in $O(n/\delta^2 + n \log n)$ time.

Cheong et.al. [8] addressed the problem in terms of one round Voronoi game. According to their formulation, two players place their sites inside a unit square and try to maximize the total Voronoi region of their own sites. They proved that for sufficiently large n the second player can always place his sites in such a way that the sum of the area of their Voronoi regions is at least $(1/2 + \alpha)$ times the total playing area, for any $\alpha > 0$, and thus found a winning strategy for the second player. Fekete and Meijer [13] extended the results of [8] to find winning strategies even when the playing area is not a unit square. They showed that the second player wins only if $n \geq 3$ and $\rho > \sqrt{2}/n$ or if $n = 2$ and $\rho > \sqrt{3}/2$, where ρ represents the aspect ratio of the playing area. They also proved that if the playing area is a polygon with holes, then finding a winning strategy for the second player is NP-hard.

Teramoto, Demaine and Uehara [15] studied discrete Voronoi games. They considered as the playing arena a given finite graph G of at least $2n$ vertices. Two players alternately occupy n vertices of G . Each vertex of G is assigned to a player based on the nearest neighbor rule. Finally, the player who dominates larger number of vertices wins. The authors in [15] show that when G is a d -ary tree it is possible to formulate a winning strategy or to show the tieness of the game, but it is NP hard for more general graphs.

Recently, Rasheed et. al. [14] have studied another criteria, which is that in the resulting Voronoi diagram, each player gets, as neighbours, as many opponent sites as possible using minimum possible sites of his own. Their motivation behind this criterion is that being neighbors of existing shopping chain, a new owner can gradually entice customer away from the existing sites by better service. They considered the game in one

round and considered several variation of this criterion. They showed that it is always possible to get all sites of Player1 as neighbors by using less than n sites, in particular, by using only $\frac{2n+2}{3}$ sites for $n \geq 12$. They also showed that Player2 can place his n sites such that, beside getting all sites of Player1 as neighbors, he can hide his own sites from self sites more than what Player1 can hide from his own sites. The motivation behind hiding self sites is avoiding self interference [14].

A. Our results

In this paper we consider yet another criterion of playing the Voronoi game. Our setting is in one round. Given a set of n Player1 sites in the plane and their Delaunay triangulation \mathcal{D} , we consider placing minimum possible Player2 sites such that in the resulting Delaunay triangulation \mathcal{D}' of the sites of both Players, the neighborhood between Player1 sites are as less as possible.

We first consider to place minimum number of Player2 sites such that each Player1 site is *individually isolated* in \mathcal{D}' . We call a Player1 site p to be individually isolated, or simply *isolated*, in \mathcal{D}' , if it has no Player1 site as neighbor. This version of the problem is more like an optimization problem, rather than games. We show that to isolate a Player1 site p , two Player2 sites are both necessary and sufficient if p is in the convex hull of \mathcal{D} , otherwise three Player2 sites are both necessary and sufficient. This gives a liner time algorithm to individually isolate all Player1 sites by $3n - h$ Player2 sites, where h is the number of sites in the convex hull of \mathcal{D} . Then we give two more algorithms for this problem. The next algorithm runs also in linear time and uses $3(n - 1) - h$ Player2 sites, but is much simpler. Our next algorithm uses $5|\mathcal{M}|$, where $|\mathcal{M}|$ is the size of a maximum matching \mathcal{M} in \mathcal{D} , and runs in $O(\sqrt{nm})$ time, where m is the number of edges of \mathcal{D} . See Fig. 1(a).

Then we consider isolating sites by *components*. This version of the problem is more like a game. A component is a maximal connected subgraph of \mathcal{D}' with all of its sites coming from a same player. We show that it is possible to place n Player2 sites such that in \mathcal{D}' the number of components among Player1 sites is higher than that among Player2 sites. See Fig. 1(b).

Note that we will *not* compute \mathcal{D}' . However, our placement of Player2 sites will confirm that if \mathcal{D}' is computed, then Player1 sites will be isolated as desired.

We call the above concept of isolating Player1 sites as *vindictive Voronoi games*. Vindictive Voronoi games is related to at least one other well known problems, which is called as *minimum stabbing set of geometric objects*

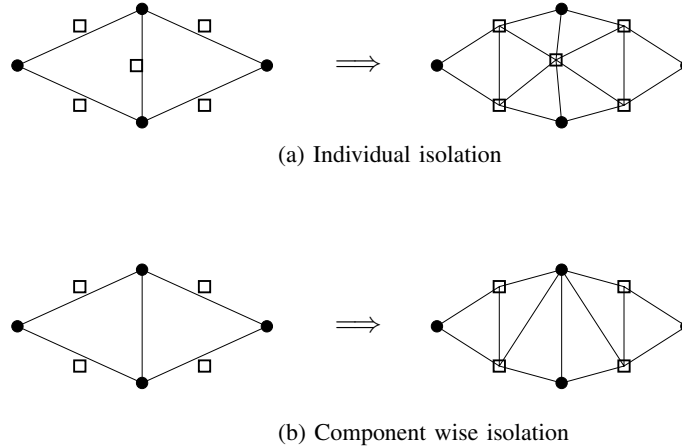


Figure 1. Player1 and Player2 sites are denoted by filled circles and empty squares respectively. (a) Individual isolation of four Player1 sites by five Player2 sites. (b) Four Player2 sites isolate same number of Player1 sites into three components, whereas Player2 himself remains in two components.

and falls under the broader classes of problems, called *witness Delaunay graphs* and *proximity graphs* [3]. Given a set of geometric objects, the problem of minimum stabbing set is to find a minimum set of points such that each object intersects at least one point. Aronov [3] studied the variation of this problem where the geometric objects are Delaunay circles of a given set of n points. Observe that this variation of the problem is same as our problem of isolating Player1 sites by minimum number of Player2 sites, since a circle through three Player1 sites is a Delaunay circle if and only if it does not contain any other site. Aronov et. al. [3] mentioned an upper bound of $2n - 2$ on the size of a minimum stabbing set. Observe that \mathcal{D} is triangulated plane graph. The best known lower bound on the size of a maximum matching of a triangulated plane graph is $\frac{n+4}{3}$. So, if \mathcal{D} has a smaller maximum matching \mathcal{M} having size close to that lower bound, then our bound of $5|\mathcal{M}|$ works better than that given by Aronov et. al. [3].

Apart from having connection to competitive facility location problems and proximity graphs, Vindictive Voronoi games might be applicable to some other domain where someone may want to disrupt existing facilities/services by cutting off links between existing sites in expense of placing his own sites.

B. Outline

We shall mostly work on the Delaunay triangulation of the sites. We shall follow incremental process: Player2 will place his sites one by one as necessary. It will require an incremental update of the Delaunay triangulation to see how the addition of a new site

changes the neighborhood among the sites.

In order to “efficiently” isolate Player1 sites by the sites of Player2, we shall select edges and vertices from \mathcal{D} and then insert Player2 sites at certain distance close to those selected edges and vertices. However, to optimize the strategy of Player2, those edges and vertices should be carefully selected so that the optimal isolation criterion holds. For that, we shall use edges of a maximum matching in \mathcal{D} .

We organize the rest of the paper as follows. We first study, in Section II, the basics of Voronoi diagram, Delaunay triangulation, their incremental update, including the update of the neighborhood among the sites, and matching in a graph (these mostly come from [14]). In Section III, we give our results on individual isolation. In Section IV, we give our results on component wise isolation. Finally, in Section V, we conclude the paper with some directions to future work.

II. PRELIMINARIES

Given a set of n points P (of Player1) in the plane, a *Voronoi diagram* [4], [9] \mathcal{V} of P is the subdivision of the plane into n regions, one for each point in P , such that any point x in the plane lies in the region corresponding to a point y in P if and only if the distance of x from y is smaller than its distance from any other point in P . The points of P are called the *Voronoi sites*, or simply *sites*, and their regions are called the *Voronoi regions* of \mathcal{V} . Voronoi regions meet at *Voronoi edges* and *Voronoi vertices*. Two Voronoi regions are called *neighbors* if they share a Voronoi edge. Note that the Voronoi diagram \mathcal{V} is a connected plane graph [4], [9].

Throughout this paper, we assume that the Voronoi sites, as the two players place them, are in general position, i.e., no three sites are collinear and no four sites are cocircular.

The *Delaunay triangulation* [4], [9] \mathcal{D} of P is the straight line dual of \mathcal{V} , where the vertices are the sites of P and two vertices are connected by a *Delaunay edge* if and only if their corresponding Voronoi regions are neighbors in \mathcal{V} . Two sites defining a Delaunay edge are also called *neighbors* to each other. Similar to \mathcal{V} , \mathcal{D} is also a plane graph [4], [9]. Note that the boundary of \mathcal{D} is the convex hull of the points of P , and we denote that convex hull by \mathcal{H} . Since the points of P are in general position, faces of \mathcal{D} , possibly except the outer face, are triangles and are called *Delaunay triangles*. The circumcircle of a Delaunay triangle is called its *Delaunay circle*. One important characterization of a Delaunay triangulation is that three sites u, v and w form a Delaunay triangle if and only if their circumcircle does not contain any other site [4], [9]. Observe that even if the outer face of \mathcal{D} is a triangle, by the above characterization it is not a Delaunay triangle when $n \geq 4$. An edge (a vertex) of \mathcal{D} that is not an edge (a vertex) of \mathcal{H} is called an *internal edge* (vertex) of \mathcal{D} . For $n \geq 4$, \mathcal{D} has at least one internal edge.

A. Inserting a new site

The phenomenon of placing a new site in a Delaunay triangulation and the resulting changes in the neighborhood of the sites is well studied in incremental construction of Delaunay triangulation [4], [9]. Let $\Delta(u, v, w)$ be a Delaunay triangle. We denote by $C(u, v, w)$ the corresponding Delaunay circle passing through u, v and w . We now present the following lemmas and corollary which will play central role in the rest of the paper. Also see Fig. 2.

Lemma 1. *If a new site q is placed inside a Delaunay circle $C(u, v, w)$, then in the resulting Delaunay triangulation, u, v and w become neighbors of q .*

A proof of this lemma can be found for example in [4] [9, Page194].

Lemma 2. [14] *If a new site q is placed outside a Delaunay circle $C(u, v, w)$, then in the resulting Delaunay triangulation, $\Delta(u, v, w)$ remains a Delaunay triangle.*

B. Maximum matching

A *matching* [17] \mathcal{M} in a graph G is a subset of the edges of G such that any vertex of G has at most one incident edge in \mathcal{M} . If a vertex has an incident

edge in \mathcal{M} , then it is called a *matched* vertex, otherwise it is called an *unmatched* vertex. A matching is *maximum* if it has maximum possible cardinality among all possible matchings. There exist efficient polynomial time algorithms to compute a maximum matching in a graph [17]. While $n/2$ is the upper bound of a maximum matching for general graphs with n vertices, not all classes of graphs hold that upper bound. One such graph class is the *triangulated plane graphs*, where each face, including the outer face, is a triangle. However, Biedl et.al. [5], [6] proved that any triangulated plane graph with $n \geq 10$ vertices has a matching of size at least $\frac{n+4}{3}$.

III. INDIVIDUAL ISOLATION

A. Linear Algorithms

We first show how to individually isolate n Player1 sites by using $3n - h$ Player2 sites in linear time. Our idea is to isolate each Player1 site in \mathcal{H} by two Player2 sites and each internal Player1 site by three Player2 sites.

As we place Player2 sites, the resulting Delaunay triangulation will change, which is denoted by \mathcal{D}' and its convex hull by \mathcal{H}' .

Lemma 3. *Two Player2 sites are necessary and sufficient to isolate a Player1 site if it is in the convex hull \mathcal{H} of \mathcal{D} . Three Player2 sites are necessary and sufficient to isolate any other Player1 site. Moreover, isolating a player1 site r_1 in such ways can be done in $O(d(r_1))$ time, where $d(r_1)$ is the degree of r_1 in \mathcal{D} .*

Proof: Proving the necessity is easy. Let r_1 be a Player1 site in \mathcal{D} where all of its neighbors are currently of Player1. When r_1 becomes isolated, all of its neighbors will be of Player2. After isolation, if r_1 is in \mathcal{H}' , then it has two or more Player2 sites as neighbors, otherwise it has three or more Player2 sites as neighbors. Therefore, if r_1 is in \mathcal{H} before isolation, and it remains in \mathcal{H}' after isolation, then it needs at least two Player2 sites to change its set of neighbors. On the other hand, if r_1 is an internal site in \mathcal{D}' after isolation, then it requires at least three Player2 sites as its neighbors.

For sufficiency, first consider that r_1 is in \mathcal{H} . The idea is to place two Player2 sites b_1 and b_2 sufficiently close to r_1 such that in \mathcal{D}' , $\Delta(r_1, b_1, b_2)$ becomes a Delaunay triangle and all of r_1, b_1 and b_2 remain in \mathcal{H}' . To be precise, let e_1 and e_2 be two adjacent edges of r_1 that are also edges of \mathcal{H} . Let the interior angle between these two edges at r_1 be α . Since \mathcal{D} is convex, we have $\alpha < 180^\circ$. We draw a circle c passing through

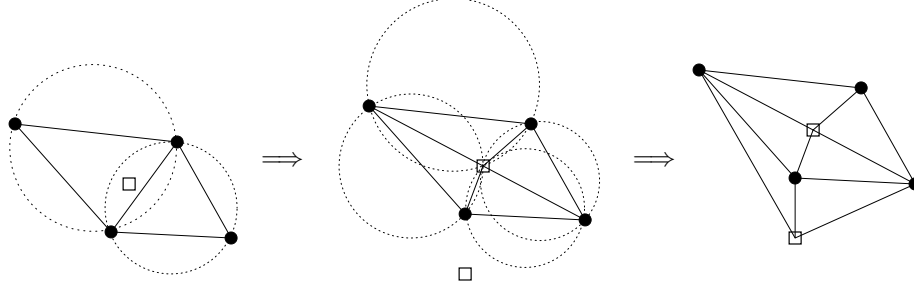


Figure 2. Illustration of inserting a new site.

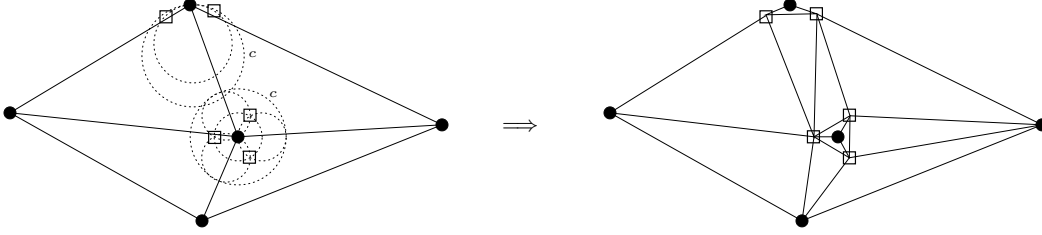


Figure 3. Isolating a convex hull site by two sites and an internal site by three sites (not all the labels are shown for clarity's sake).

r_1 such that c properly intersects \mathcal{D} , does not contain any other site of \mathcal{D} , and intersects both e_1 and e_2 in two other different points. Since $\alpha < 180^\circ$, it is always possible to draw such a circle with its center, say, on the bisector of α and by keeping the diameter smaller than the distance of r_1 from its nearest neighbor. (See Fig. 3). Let s_1 and s_2 be two circular segments created by the intersection of e_1 and e_2 with c . Place two Player2 sites b_1 and b_2 inside those two segments. Since b_1 and b_2 are inside the circle c , the circle $C(r_1, b_1, b_2)$ is a subset of c and therefore, is empty of any other site. So, in \mathcal{D}' , $\triangle(r_1, b_1, b_2)$ is a Delaunay triangle. Moreover, the edge (b_1, b_2) intersects e_1 , e_2 , and any other Delaunay edge of \mathcal{D} that is incident to r_1 . That means, b_1, b_2 , and any other neighbor of r_1 in \mathcal{D} are no more neighbors of r_1 in \mathcal{D}' . It implies that r_1 is isolated of all other Player1 sites.

Now consider that r_1 is an internal site in \mathcal{D} . Here, our idea is to place three Player2 sites b_1, b_2, b_3 sufficiently close to r_1 such that in \mathcal{D}' , we have that b_1, b_2, b_3 are neighbors of each other and r_1 is the only site inside the triangle formed by them. To be precise, we draw a circle c with center r_1 and diameter smaller than the distance of r_1 from its closest neighbor. c does not contain any other site. We draw another circle c' with center r_1 and radius half of that of c . We placed three Player2 sites b_1, b_2, b_3 on c' such that they are equally apart of each other. This will make the circle through r_1

and any two of b_1, b_2, b_3 a Delaunay circle in \mathcal{D}' , since that circle is a subset of c and therefore, is empty of any other site. (See Fig. 3). It implies that r_1 has only neighbors b_1, b_2, b_3 .

In both cases, finding the radius of c can be done by computing the distance of r_1 from its neighbors in $O(d(r_1))$ time. Other operations take constant time. ■

This gives our first linear time algorithm to individually isolate Player1 sites by Player2 sites.

Theorem 1. *Given n Player1 sites in the plane and their Delaunay triangulation \mathcal{D} , they can be individually isolated by $3n - h$ Player2 sites in $O(n)$ time, where h is the number of sites in \mathcal{H} .*

Proof: By Lemma 3, $2h$ sites are sufficient to isolate the sites in \mathcal{H} , and for the remaining $n - h$ internal sites, we need $3(n - h)$ sites, giving a total of $3n - h$ sites. Time required is $\sum_i d(r_i) = O(n)$, since \mathcal{D} is planar. ■

The above algorithm is adaptive to the number of sites in \mathcal{H} . In the worst case there can be only three sites in \mathcal{H} , and thus the above algorithm has a worst case bound of $3n - 3$. We next present another linear-time algorithm which has a slightly better worst case bound but is much simpler.

Theorem 2. *Given n Player1 sites in the plane and their Delaunay triangulation \mathcal{D} , they can be individually isolated by m sites of Player2 in $O(n)$ time, where m*

is the number of edges in \mathcal{D} .

Proof: Here the idea is to destroy all existing edges of \mathcal{D} so that no two Player1 sites become neighbor in \mathcal{D}' . Let $\triangle(u, v, w)$ be a Delaunay triangle of \mathcal{D} with $e = (u, v)$ being an edge of \mathcal{H} . Place a Player2 site b outside \mathcal{D} but close to e and inside the Delaunay circle $C(u, v, w)$. By Lemma 1, b will be a neighbor of w and thus ceasing the neighborhood between u and v .

With $e = (u, v)$ as an internal edge, let $\triangle(u, v, w)$ and $\triangle(u, v, x)$ be two adjacent Delaunay triangles of e . We place a Player2 site b inside the intersection of $C(u, v, w)$ and $C(u, v, x)$ and thus making both w and x neighbors of b by Lemma 1 and ceasing the neighborhood between u and v .

Since \mathcal{D} is planar, by Euler's theorem it holds that $m \leq 3(n-1) - h$, where h is the number of sites in \mathcal{H} . In worst case, when $h = 3$, we get $m \leq 3n - 6$. Since we place only one Player2 site for one edge of \mathcal{D} , we need at most $m \leq 3(n-1) - h \leq 3n - 6$ player2 sites. The running time is also constant per edge and is thus linear in total. ■

B. Improved Bound

We now show how to improve the bounds on number of Player2 sites required to isolate Player1 sites. Our idea is to use five Player2 sites to individually isolate two Player1 sites that are the end points of an edge of \mathcal{D} . We want to do that for as many edges of \mathcal{D} as possible, and for that we chooses the edges of a maximum matching of \mathcal{D} . To isolate the remaining Player2 sites in \mathcal{D} , we will use three sites for each as mentioned in Lemma 3.

Lemma 4. *Let $e = (u, v)$ be an internal edge of \mathcal{D} . Then it is possible to individually isolate u and v by using five sites of Player2.*

Proof: Let $\triangle(u, v, w)$ and $\triangle(u, v, x)$ be the two adjacent Delaunay triangles of e . Let $R(u, v) = C(u, v, w) \cap C(u, v, x)$. We place a Player2 site c somewhere very close to e but within $R(u, v)$. By Lemma 1, all of u, v, w and x become neighbors of c . We will place two more Player2 sites a and b to isolate u . The idea is to make a, b, c and u as the neighbors of each other with u being inside of the triangle formed by a, b and c . Consider the two Delaunay circles $C(c, u, w)$ and $C(c, u, x)$. They are empty of any other sites. We place a and b inside these two circles, respectively, and slightly "above" the edge (c, u) as shown in Fig. 4 such that the triangle formed by a, b and c keeps u inside of it. By Lemma 1, both of them become neighbors of a . Moreover, by keeping them close enough to u will

make themselves neighbors of each other. Similarly, we place two other sites, which together with c isolate v . See Fig. 4. ■

Theorem 3. *Given n Player1 sites in the plane and their Delaunay triangulation \mathcal{D} , for $n \geq 7$ it is possible for Player2 to individually isolate all n sites of Player1 by using $5|\mathcal{M}|$ sites in $O(\sqrt{nm})$ time, where $|\mathcal{M}|$ is the size of a maximum matching \mathcal{M} of \mathcal{D} .*

Proof: If \mathcal{H} is not a triangle, then we insert three sites, which do not belong either to Player1 or to Player2, far enough and outside all existing Delaunay circles of \mathcal{D} such that in the resulting Delaunay triangulation of $n' = n + 3$ sites, the convex hull is the triangle containing made by these three sites. Moreover, we place these three sites in such a way that they follow the general position of the sites. By Lemma 2, neighborhood among the n sites of Player1 do not change in the resulting Delaunay triangulation. We will individually isolate these n' sites, and w.l.o.g we use the symbols \mathcal{D} and \mathcal{H} for this updated Delaunay triangulation of $n' = n + 3$ sites. Observe that it is not very difficult to place these three sites and to update \mathcal{D} in linear time.

\mathcal{D} is now a maximally triangulated plane graph. We find a maximum matching \mathcal{M} of \mathcal{D} in $O(\sqrt{nm})$ time [17]. Let Z be the set of unmatched vertices in \mathcal{D} . There is no edge between two vertices of Z , since such an edge would be added to \mathcal{M} to get a larger matching.

Now for each edge $e \in \mathcal{M}$, if e is an internal edges of \mathcal{D} , then we isolate its two end points by using five Player2 sites as mentioned in Lemma 4. Otherwise, e is an edge of \mathcal{H} , and we isolate each of its two end points by using two Player2 sites as mentioned in Lemma 3. After we are done with all edges of \mathcal{M} , all matched vertices of \mathcal{D} are individually isolated. Since there is no edge between two vertices of Z , edges of \mathcal{D} no longer exist, and thus the vertices of Z are also individually isolated. The number of Player2 sites required by the above placement is at most $5|\mathcal{M}|$. ■

According to Aronov et.al. [3], the problem of *minimum stabbing set for Delaunay circles* is, given a set of n points P , to find another set of points Q with minimum cardinality such that every circle through two points of P contains a point of Q . A point of Q inside such a circle is called a *stabbing point* and the set Q is called a *stabbing set*. Remember that in a Delaunay triangulation, two sites form a Delaunay edge if and only if there is an empty circle through them. That means, according to our terminology, P is the set of Player1 sites, Q is the Player2 sites, and the problem of minimum stabbing set of Delaunay circles

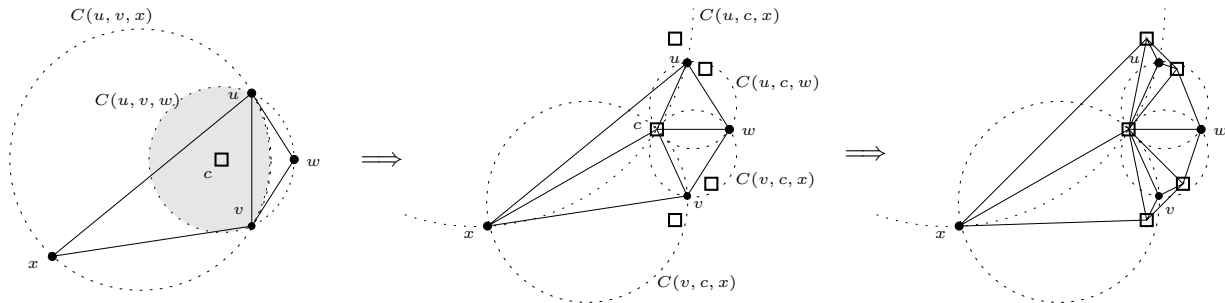


Figure 4. Isolating two end points of an edge by five sites.

is equivalent to finding the minimum number of Player2 sites such that Player1 sites are individually isolated in \mathcal{D}' . Remember that \mathcal{D} is triangulated plane graph. The best lower bound on the size of a maximum matching of a triangulated plane graph is $\frac{n+4}{3}$ [5], [6]. So, if \mathcal{D} has a smaller maximum matching close to that lower bound, then our bound of $5|\mathcal{M}| \approx \frac{5}{3}(n+4)$ in Theorem 3 would work better than the bound of $2n - 2$ given by Aronov et. al. [3]. In particular, for $\frac{n+4}{3} \leq |\mathcal{M}| \leq \frac{2n-2}{5}$, our algorithm would perform better.

Remark 1. *It is possible to achieve in $O(\sqrt{nm})$ time an upper bound of $5|\mathcal{M}|$ on the minimum stabbing set for Delaunay circles, where $|\mathcal{M}|$ is the size of a maximum matching in the given Delaunay triangulation.*

IV. COMPONENT WISE ISOLATION

In this section, we study the problem in approach of more like a Voronoi game. We play the game in one round. After Player1 has placed his n sites, Player2 place *all* of his n sites. We define the *score* of Player i , for $i = 1, 2$, as $s_i = n - c_i$, where c_i is the number of components of the sites of Player i in \mathcal{D}' . So, the lowest score of Player i is zero when all of its n sites are individually isolated and the highest score is $n - 1$ when all n sites are connected. The player whose score is higher *wins* the game. Based on this scoring strategy it is possible to devise a *winning strategy* for Player 2 in one round.

Theorem 4. *After Player1 has placed is n sites, Player2 can place his n sites such that in \mathcal{D}' the number Player1 components is higher than that of Player2, and thus Player2 wins.*

Proof: Take any $n/3$ Player1 sites from \mathcal{D} . Individually isolate each of them by three sites of Player2 by using Lemma 3. In the resulting Delaunay triangulation, each of these $n/3$ sites of Player1 is a single component and the remaining sites of Player1 make at least one

component. So, Player1 has $c_1 \geq n/3 + 1$ components. On the other hand, all n sites of Player2 have been engaged in isolating the sites in \mathcal{D} . As a whole, these n sites of Player2 may or may not be connected, but every three sites (we call them a *triplet*) that are used to isolate a single site of Player1 are connected. Moreover, triplets may themselves be connected. So, $c_2 \leq n/3$. ■

V. CONCLUSION

We think that the bound in Theorem 3 is not optimal, as Aronov [3] showed a far smaller lower bound of n to isolate all sites of Player1. Currently, the winning margin for Player2 in Theorem 4 is 1 in worst case, i.e., Player2 can have only one component less than that of Player2. We think that in future this can be improved so that the number of Player2 components is much smaller than that of Player1.

We also find it challenging to find results in n -round.

REFERENCES

- [1] H.-K. Ahn, S.-W. Cheng, O. Cheong, M. Golin, and R. van Oostrum. Competitive facility location: the Voronoi game. *Theoretical Computer Science*, 310(1-3):457–467, 2004.
- [2] H.-K. Ahn, S.-W. Cheng, O. Cheong, M. J. Golin, and R. van Oostrum. Competitive facility location along a highway. In *COCOON'01: Proc. 7th Annual International Conference on Computing and Combinatorics*, volume 2108 of *Lecture Notes in Computer Science*, pages 237–246. Springer, 2001.
- [3] B. Aronov, M. Dulieu, and F. Hurtado. Witness (de-launay) graphs. In *Proc. 7th Japan Conference on Computational Geometry and Graphs*, 2009. Available online at: http://www.jaist.ac.jp/~uehara/JCCGG09/short/paper_16.pdf.
- [4] F. Aurenhammer and R. Klein. Voronoi diagrams. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 201–290. North-Holland, 2000.

- [5] T. Biedl, E. D. Demaine, C. A. Duncan, R. Fleischer, and S. G. Kobourov. Tight bounds on maximal and maximum matchings. In *ISAAC'01: Proc. 12th International Symposium on Algorithms and Computation*, volume 2223 of *Lecture Notes in Computer Science*, pages 308–319. Springer, 2001.
- [6] T. C. Biedl, E. D. Demaine, C. A. Duncan, R. Fleischer, and S. G. Kobourov. Tight bounds on maximal and maximum matchings. *Discrete Mathematics*, 285(1-3):7–15, 2004.
- [7] O. Cheong, A. Efrat, and S. Har-Peled. Finding a guard that sees most and a shop that sells most. *Discrete & Computational Geometry*, 37(4):545–563, 2007.
- [8] O. Cheong, S. Har-Peled, N. Linial, and J. Matoušek. The one-round Voronoi game. In *SCG'02: Proc. 18th ACM Annual Symposium on Computational Geometry*, pages 97–101, 2002.
- [9] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer, 2nd edition, 2000.
- [10] F. K. H. A. Dehne, R. Klein, and R. Seidel. Maximizing a Voronoi region: The convex case. In *ISAAC'02: Proc. 13th International Symposium on Algorithms and Computation*, volume 2518 of *Lecture Notes in Computer Science*, pages 624–634. Springer, 2002.
- [11] H. Eiselt and G. Laporte. Competitive spatial models. *European Journal of Operational Research*, 39:231–242, 1989.
- [12] H. Eiselt, G. Laporte, and J.-F. Thisse. Competitive location models: A framework and bibliography. *Transportation Science*, 27:44–54, 1993.
- [13] S. P. Fekete and H. Meijer. The one-round Voronoi game replayed. *Computational Geometry*, 30(2):81–94, 2005.
- [14] M. M. Rasheed, M. Hasan, and M. S. Rahman. Maximum neighbour voronoi games. In *Proc. 3rd International Conference on Algorithms and Computation (WALCOM'09)*, volume 5431 of *Lecture Notes in Computer Science*, pages 93–104. Springer, 2009.
- [15] S. Teramoto, E. D. Demaine, and R. Uehara. Voronoi game on graphs and its complexity. In *CIG'06: Proc. IEEE Symposium on Computational Intelligence and Games*, pages 265–271, Reno, Nevada, May 2006. IEEE.
- [16] R. Tobin, T. Friesz, and T. Miller. Competitive location models: A framework and bibliography. *Annals of Operations Research*, 18:267–276, 1989.
- [17] D. B. West. *Introduction to Graph Theory*. Pearson Education, 2nd edition, 2002.