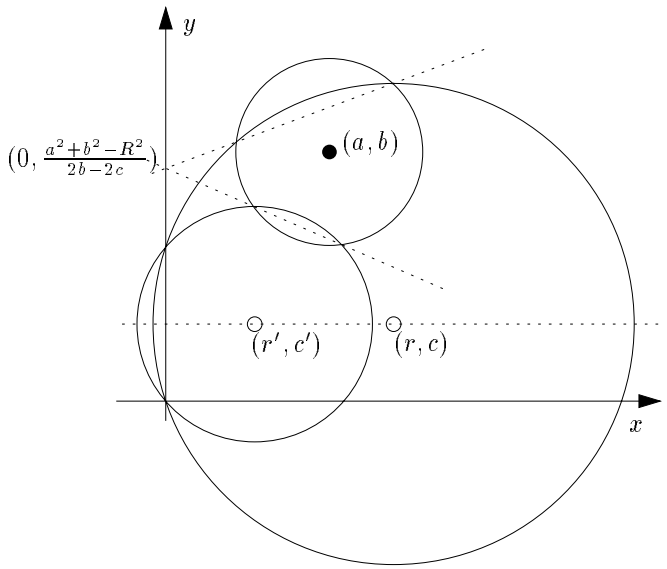

Figure 17: Locus of intersection points of two circles, one fixed and one variable. The fixed circle is the one with center (a, b); the variable one is with center (r, c) and radius $(r^2 + c^2)^{1/2}$ and r is allowed to vary (that is all circles in this family pass through the same two points on the y-axis). The family of lines through the intersection points of the two circles are concurrent.

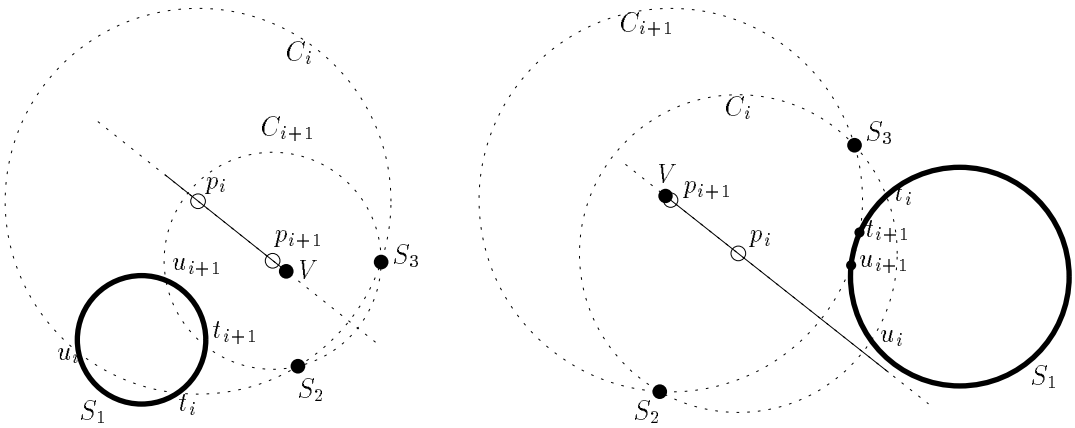

Figure 16: The case of two points and a circle. The Voronoi vertex V can lie at the end-point of $\mathcal{VC}(S_1, S_2, S_3)$ (solid line in the picture) which is nearest to S_2, S_3 (left), or at the one which is farthest from S_2, S_3 (right).

Consider the point p_1 obtained in the first iteration, which by definition lies on the line segment $\mathcal{VC}(S_1, S_2, S_3)$. To show $\{p_i\} \to V$, it suffices to show that p_2 lies in the segment $\overline{p_1 V}$ (if $p_2 = p_1$, then $p_1 = V$).

We show this with the following argument. Let C_i and r_i, as before, denote the Voronoi circle and its radius in the ith iteration (see Figure 16). Let C_i intersect S_1 at t_i, u_i (one of these is the point on S_1 closest to p_{i-1}). Let t_i, u_i divide S_1 into two arcs termed the *interior arc wrt* C_i and *exterior arc wrt* C_i.

Let $\rho_C(t)$ denote the polar angle of point t on circle C. In the following ρ stands for ρ_{S_1}. Below, we first show that the signs of the two quantities $\rho(t_i) - \rho(t_{i+1})$ and $\rho(u_i) - \rho(u_{i+1})$ are opposite for all i (and converging to the limit they both converge to zero). This implies that both t_{i+1}, u_{i+1} lie in the interior arc wrt C_i. To complete the argument simply observe that the interior arc wrt C_i has to properly include that wrt C_{i+1} (this is because the p_i are all on a line segment and therefore to "one side" of the circle).

Finally, we show that the signs of $\rho(t_i) - \rho(t_{i+1})$ and $\rho(u_i) - \rho(u_{i+1})$ are opposite. Consider the intersections of a generic circle with center (a, b) and radius R and another with center (r, c) and radius $\sqrt{r^2 + c^2}$, for varying r (see Figure 17). It is easily seen that these intersection points $t(r), u(r)$ (if they exist) lie on the line (for a given r)

$$y = \frac{r - a}{b - c} x + \frac{a^2 + b^2 - R^2}{2b - 2c}.$$

Now as r varies, notice that the family of lines passes through the fixed point $(0, \frac{a^2+b^2-R^2}{2b-2c})$. Moreover, the slope of lines $(\frac{r-a}{b-c})$ varies linearly with r. By considering the intersections of lines passing through a fixed point with a circle, the desired property on the polar angles follows at once. □

Remark: The proof goes through roughly as already presented for S_1 being any finite convex object instead of being restricted to a circle.

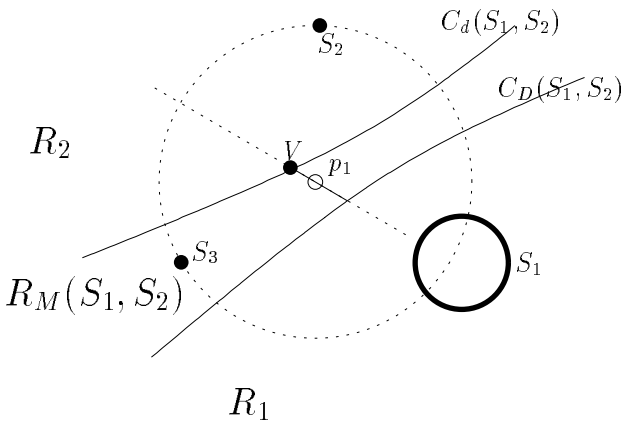

Figure 15: S_1 is a circle and S_2, S_3 are two point objects. The locus of the circle centers $\mathcal{VC}(S_1, S_2, S_3)$ is a segment taken from the perpendicular bisector of points S_2, S_3. At one end point of this segment lies the Voronoi vertex V. The non-intersecting curves $C_d(S_1, S_2)$ and $C_D(S_1, S_2)$ partition the plane into the sets R_1, R_2 including S_1, S_2, respectively, and $R_M(S_1, S_2)$, the "middle" region.

Let $\mathcal{VC}(S_1, S_2, S_3)$ denote the *locus of circle centers*, i.e., the centers of circles that pass through all possible triplets of points taken from S_1, S_2, S_3.

In the case of two points and a line (in general position), the locus of circle centers $\mathcal{VC}(S_1, S_2, S_3)$ coincides with $bis(S_2, S_3) \setminus \overline{V_+V_-}$. However, in the present case in which S_1 is a circle, we first show that $\mathcal{VC}(S_1, S_2, S_3)$ is a segment (a subset of $bis(S_2, S_3)$) and that V lies at one end-point of this segment.

Towards this end, define $\text{dist}(p, S)$ to be the (Euclidean) distance from point p to the nearest point on S; and $\text{Dist}(p, S)$, the distance from p to the *farthest* point on S. Notice that these two distance measures coincide if (and only if) S is a single point.

Now consider the locus of points p that satisfy $\text{dist}(p, S_1) = \text{dist}(p, S_2)$; and likewise the locus of points for $\text{Dist}(p, S_1) = \text{Dist}(p, S_2)$. Call the two curves formed be $C_d(S_1, S_2), C_D(S_1, S_2)$, respectively. $C_d(S_1, S_2)$ is the same as $bis(S_1, S_2)$. See Figure 15 for an example.

The open curves $C_d(S_1, S_2), C_D(S_1, S_2)$ clearly do not intersect (unless both S_1, S_2 are points, in which case they are coincident) and therefore separate the plane into three unbounded regions $R_1, R_2, R_M(S_1, S_2)$, where R_2 is the region farthest from S_1, R_1, the region farthest from S_2; and $R_M(S_1, S_2)$, the "middle" region. The property of any point in R_1 is that it is closer to *every* point (or the farthest) in S_1 than to *any* (or the nearest) point in S_2. Likewise for R_2. On the other hand, points q in $R_M(S_1, S_2)$ satisfy the property that there exist points $q_1 \in S_1$, $q_2 \in S_2$ such that $\text{dist}(p, q_1) = \text{dist}(q, q_2)$.

From this last property of points in R_M, it is clear that

$$\mathcal{VC}(S_1, S_2, S_3) \subset \cap_{i,j \in \{1,2,3\}, i \neq j} R_M(S_i, S_j). \tag{6}$$

For our case of a circle and two points, $R_M(S_2, S_3) = bis(S_2, S_3)$, and from (6) $\mathcal{VC}(S_1, S_2, S_3) \subset R_M(S_2, S_3) \cap R_M(S_1, S_2)$. Therefore, $\mathcal{VC}(S_1, S_2, S_3)$ is a line segment. To see that V lies at one end-point of this line segment $R_M(S_2, S_3) \cap R_M(S_1, S_2)$, notice that V is the intersection point of $R_M(S_2, S_3) = bis(S_2, S_3)$ and $C_d(S_1, S_2) = bis(S_1, S_2)$ which is a bounding curve of $R_M(S_1, S_2)$.

Unfortunately, in extending this to the case $c \neq 0$, we get one factor that appears quite unwieldy. Therefore we choose a different approach. Let the x-coordinates of the Voronoi vertices, V_-, V_+, be X_-, X_+ respectively. We show below that the sequence $\{x_i\}$ converges to one of X_+, X_- depending on x_1. This is sufficient because the nearest point from p_i on S_1 is $(x_i, 0)$, fully defined by p_i's x-coordinate.

Let $X_M = \max(X_+, X_-)$ and $X_m = \min(X_+, X_-)$. We show that $\{x_i\} \to X_M$ if $x_1 > \frac{X_+ + X_-}{2} = ac/(a-b)$. A similar proof holds for showing $\{x_i\} \to X_m$ if $x_1 < ac/(a-b)$. (The case of $x_1 = ac/(a-b)$ leads to x_2 being $\pm\infty$ and then onwards x_2 may replace x_1 in the following analysis).

The proof that $\{x_i\} \to X_M$ if $x_1 > ac/(a-b)$ follows from the following propositions:

1. $x_2 > X_M$.
2. If $x_1 > X_M$ then $x_2 < x_1$.

Proof. First consider x_2 in terms of x_1:

$$x_2 = \frac{a^2 b - ac^2 - ab^2 + a x_1^2 - b x_1^2}{-2\,ca + 2\,x_1 a - 2\,x_1 b}. \tag{5}$$

It can be easily verified that the only solutions (for x_1) to $x_2 = x_1$ occur at the x-coordinates of the Voronoi vertices, i.e., at $x_1 = X_M, X_m$. Further simple algebra shows that $x_2 < x_1$ in the intervals $x_1 \in (X_m, ac/(a-b)) \cup (X_M, +\infty)$ and $x_2 > x_1$ when $x_1 \in (-\infty, X_m) \cup (ac/(a-b), X_M)$ (as mentioned before, $x_1 = ac/(a-b)$ is a singularity).

This proves 2. To see 1, first assume $X_M = X_+$ (this is exactly when $a > b$) and consider solutions to $x_2 = X_M$ for x_1. Working out the algebra shows that this quadratic equation has a double root at $x_1 = X_M$ and that for all $x_1 \neq X_M$, $x_2 > X_M$. A similar proof holds for $X_M = X_-$. □

Remark: The key to this solution was the fact that it was sufficient to consider x-coordinates alone and the simple expression for x_2 in terms of x_1 (or in general any x_{i+1} in terms of x_i). Unfortunately, such expressions do not exist for other simple situations such as a circle and two points. However, this proof can be extended to the case of a line *segment* and two points by further case analysis.

A.2 Case of Two Points and One Circle

In this section we consider S_1 being a circle and S_2, S_3 being points (as before) outside the circle. As long as the two points are visible from each other, a Voronoi vertex exists and is unique. Assume S_2, S_3 are visible from each other and let the Voronoi vertex be V.

For the unit circle, and two point objects $(a, 0)$ and (c, b), if (x_n, y_n) denotes the current point, then

$$x_{n+1} = \frac{bD(a^2-1) + y_n(c^2+b^2-a^2)}{2\,(-x_n b + (c-a)y_n + abD)}$$

$$y_{n+1} = \frac{cD(1-a^2) + aD(c^2+b^2-1) + x_n(a^2-b^2-c^2)}{2\,(-x_n b + (c-a)y_n + abD)}$$

where $D = \sqrt{x_n^2 + y_n^2}$. Since the explicit algebraic mapping is quite involved,[3] we resort to a geometric proof.

[3] The algebraic expressions for x_{n+1} and y_{n+1} get far more complicated for say two circles and a point, two circles and a line, or three circles.

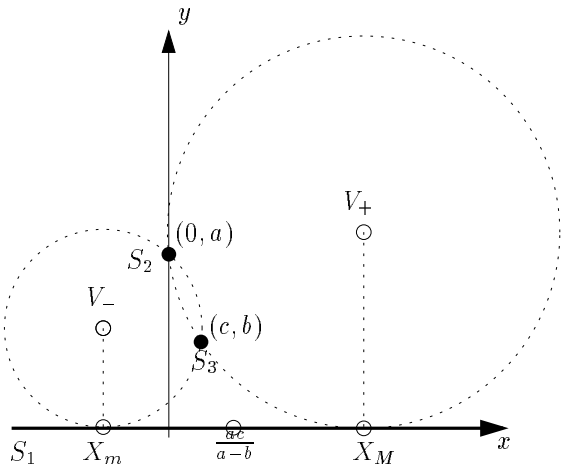

Figure 14: Proving convergence for the case of two point objects and a line. Two Voronoi vertices, shown as V_+ and V_-, can exist with definite regions of convergence. The proof is simplified by considering x-coordinates alone. If the initial seed point p_0 has its x-coordinate greater (less) than $ac/(a-b)$ then the convergence is to V_+ (V_-). If the x-coordinate is exactly $ac/(a-b)$, p_1 is one of two infinities in opposite directions; the convergence depends on the direction chosen.

2. If $a \neq b$ (and none equals zero)[2] then two Voronoi vertices exist:

$$V_- = \left(\frac{ac - \sqrt{\Delta}}{a-b}, \frac{\mathrm{Poly}(a,b,c) - \sqrt{\Delta}}{(a-b)^2} \right);$$

$$V_+ = \left(\frac{ac + \sqrt{\Delta}}{a-b}, \frac{\mathrm{Poly}(a,b,c) + \sqrt{\Delta}}{(a-b)^2} \right).$$

where $\Delta = ab(c^2 + (a-b)^2)$ (and $\mathrm{Poly}(a,b,c)$ is a polynomial in (a,b,c)).

Observe that the solutions are real if and only if $ab > 0$ (that is, if and only if S_2 is 'visible' from S_3).

The second case of $a \neq b$ is more general and is therefore considered in the rest of this section. We also assume $ab > 0$. The proof is algebraic in nature and we are indebted to the Computer Algebra system *Maple V* for saving us many frustrating hours.

Let point p_i in the sequence $\{p_i\}$ have coordinates (x_i, y_i), define a Voronoi circle C_i of radius r_i.

We first give a simple "one-line" proof for the case $c = 0$. Consider the difference $r_1^2 - r_2^2$ and its factorization:

$$r_1^2 - r_2^2 = \frac{(3\,ab + x_1^2)\,(ab + 3\,x_1^2)\,(ab - x_1^2)^2}{16\,x_1^2\,(ab + x_1^2)^2}.$$

Suffices to observe that this is strictly > 0 (since $ab > 0$).

[2] none equal zero by the non-intersecting obstacles assumption. If exactly one of them equals zero, then exactly one (real) Voronoi vertex exists.

[19] D. Leven and M. Sharir. Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams. *Discrete Comput. Geom.*, 2:9–31, 1987.

[20] E. Mazer, J. M. Ahuactzin, E.-G. Talbi, and P. Bessiere. The Ariadne's clew algorithm. Technical report, Laboratoire d'Informatique Fondamentale et d'Intelligence Artificielle, Grenoble, France, July 1992.

[21] C. Ó'Dúnlaing, M. Sharir, and C. K. Yap. Retraction: a new approach to motion-planning. In *Proc. 15th Annu. ACM Sympos. Theory Comput.*, pages 207–220, 1983.

[22] C. Ó'Dúnlaing, M. Sharir, and C. K. Yap. Generalized Voronoi diagrams for moving a ladder, I: topological analysis. *Commun. Pure Appl. Math.*, 39:423–483, 1986.

[23] Atsuyuki Okabe, Barry Boots, and Kokichki Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams.* John Wiley & Sons, Chichester, England, 1992.

[24] M. Overmars and P. Švestka. A probabilistic learning approach to motion planning. In *The First Workshop on the Algorithmic Foundations of Robotics.* A. K. Peters, Boston, MA, 1994.

[25] H-O Peitgen, H. Jürgens, and D. Saupe. *Chaos and Fractals.* Springer-Verlag, New York, 1992.

[26] S. Stifter. An axiomatic approach to Voronoi-diagrams in 3D. *J. Comput. Syst. Sci.*, 43:361–374, 1991.

[27] J. Vleugels, J. Kok, and M. Overmars. A self-organizing neural network for robot motion planning. In *Proc. Internat. Conf. Artific. Neural Networks*, pages 281–284, 1993.

[28] J. Vleugels and M. Overmars. Approximating Voronoi diagrams. Technical report, Dept. Comput. Sci., Utrecht Univ., Utrecht, the Netherlands, 1994 (to appear).

[29] C. K. Yap. An $O(n \log n)$ algorithm for the Voronoi diagram of a set of simple curve segments. *Discrete Comput. Geom.*, 2:365–393, 1987.

A Special cases

In this section we consider the case in which two of the objects are points. Although this is the simplest deviation from the trivial case of three point objects, we find that the proofs of convergence are not trivial. We consider two points and one line first in Section A.1 and then two points and a circle in Section A.2. The latter can be extended to proving the general case of two points and any third object.

A.1 Case of Two Points and One Line

In this section we prove convergence to a Voronoi vertex for the case of S_1 being an infinite line and S_2, S_3 being single points (see Figure 14).

Without loss of generality, we assume S_1 to be the x-axis and $S_2 = (0, a)$, $S_3 = (c, b)$, and S_2, S_3 different (actually, this follows from our earlier assumption that no two of the obstacles intersect). The following may be readily verified and require no proof.

1. If $a = b \neq 0$ then only one Voronoi vertex exists: $(c/2, (c^2 + 4a^2)/(8a))$.

References

[1] H. Alt and C. K. Yap. Algorithmic aspect of motion planning: a tutorial, part 2. *Algorithms Rev.*, 1(2):61–77, 1990.

[2] F. Aurenhammer. Voronoi diagrams: a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23:345–405, 1991.

[3] D. Avis and B. K. Bhattacharya. Algorithms for computing d-dimensional Voronoi diagrams and their duals. In F. P. Preparata, editor, *Computational Geometry*, volume 1 of *Advances in Computing Research*, pages 159–180. JAI Press, London, England, 1983.

[4] P. Berge, Y. Pomeau, and C. Vidal. *Order within Chaos: Towards a Deterministic Approach to Turbulence*. Wiley, New York, 1984.

[5] J. Canny. *The Complexity of Robot Motion Planning*. MIT Press, Cambridge, MA, 1987.

[6] J. Canny and B. R. Donald. Simplified Voronoi diagrams. *Discrete Comput. Geom.*, 3:219–236, 1988.

[7] N. Carriero and D. Gelernter. *How to Write Parallel Programs*. MIT Press, Cambridge, MA, 1990.

[8] H. Choset and J. Burdick. Sensor based planning and nonsmooth analysis. In *Intl. Conf. on Robotics and Automation*, pages 3034–3041, San Diego, May 1994. IEEE.

[9] G. Foux, M. Heymann, and A. Bruckstein. Two-dimensional robot navigation among unknown stationary polygonal obstacles. *IEEE J. Robot. Autom.*, 9:96–102, 1993.

[10] G.-J. Giezeman. *PlaGeo—A Library for Planar Geometry*. Dept. Comput. Sci., Utrecht Univ., Utrecht, the Netherlands, August 1993.

[11] E. G. Gilbert and D. W. Johnson. Distance functions and their application to robot path planning in the presence of obstacles. *IEEE J. Robot. Autom.*, RA-1:21–30, 1985.

[12] B. Grünbaum. *Convex Polytopes*. Wiley, New York, NY, 1967.

[13] D. R. Hofstadter. *Metamagical Themas*. Penguin Books, London, 1986. (Chapter 16: Mathematical Chaos and Strange Attractors).

[14] J. E. Hopcroft, J. T. Schwartz, and M. Sharir. *Planning, Geometry, and Complexity of Robot Motion*. Ablex Publishing, Norwood, NJ, 1987.

[15] D. Kirkpatrick and J. Snoeyink. Tentative prune-and-search for computing Voronoi vertices. In *Proc. 9th Annu. ACM Sympos. Comput. Geom.*, pages 133–142, 1993.

[16] D. G. Kirkpatrick. Efficient computation of continuous skeletons. In *Proc. 20th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 18–27, 1979.

[17] K. Kuratowski and A. Mostowski. *Set Theory*. North-Holland, Amsterdam, the Netherlands, 1976.

[18] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, 1991.

vertex, for some sufficiently small ε. However, in practice we observed that the region of convergence is much larger that this conservative ε-neighborhood and in a relatively few number of trials (between one and four), one always secured a hit.

The second part of our work dealt with implementing our ideas into an efficient motion planner for a planar robot with two degrees of freedom. We introduced suitable strategies to pick up triples of obstacles in a complex scene that are likely to define a Voronoi vertex, and discussed how to capture the connectivity of the diagram and use it for path planning and navigation purposes. We have assumed that the obstacles are modeled as polygonal objects for the implementations. This is only because we do not yet have routines that return closest points on curved objects; a problem that we hope to overcome in the future. Finally, we believe our implementation can be efficiently parallelized. Rather than sequentially running a fixed number of iterations from a single seed point, we can pick a sufficiently large (depending on the required probability of success) number of seed points and execute iterations on them in parallel. This also has an advantage that we do not have to detect oscillations necessarily. If convergence isn't achieved from a seed point, we simply ignore it. The Linda computational framework [7] seems to be the natural testbed to implement a parallel version of our technique in which the search for vertices is executed by a number of processes in parallel.

We plan to extend these ideas to configuration spaces in higher dimensions. In doing this, we expect other complications and interesting problems. In all our experiments with objects in the plane, we experienced only the cases of convergence to a point (possibly at infinity) or oscillation of some finite period, that is, well behaved and predictable behavior of the sequence $\{p_i\}$. Chaotic behavior or curves with fractal dimension and strange attractors such as the *attractor of Hénon*[1] were never observed. A general proof that chaos is impossible for our scheme in the planar case would be difficult; the map $p_i \mapsto p_{i+1}$ appears quite complicated except for the simplest of cases. We analyze some special cases in the appendix.

Apart from the obvious reasons (curiosity, extension, more application to robotics) for studying the behavior of our mapping in higher dimensions, there is another one: it might be easier to find an example of chaotic behavior in higher dimensions. In d dimensions a Voronoi vertex is defined by $d+1$ objects; the new point will be the center of the circumhypersphere of the $d+1$ points on the objects nearest to the current point. Chaotic behavior and strange attractors are more the rule than the exception in anything but the simplest of systems and it would be very interesting to detect the presence or prove the absence of chaos in our system.

Acknowledgements

We thank Mark de Berg, Ken Goldberg, Rachel Kuske, Govindan Rajeev, Otfried Schwarzkopf, Ferdinand Verhulst, and Mahendra Verma for interesting discussions. We are indebted to Otfried for the drawing editor *Ipe* with which the pictures in this document were drawn, and to Geert-Jan Giezeman for his *PlaGeo* library [10], a set of geometric and graphic routines which greatly eased our implementations and led to the efficient timings.

[1] Chaotic behavior may be observed by considering iterations in the one dimensional map $x_{n+1} = ax_n(1-x_n)$, for $a \in (3.6, 4), x_0 = 1/25$. The attractor of Hénon can be given by $(x_{n+1}, y_{n+1}) = (y_n - ax_n^2 - 1, bx_n), a = 7/5, b = 3/10, x_0 = y_0 = 0$. For a gentle introduction, see [13]. For more than that, see [25].

earlier trials; only 4% of the trials add to the existing knowledge of the environment. This could be improved by changing the way in which starting points for the sequences $\{p_i\}$ are chosen, e.g., we could 'trace' the diagram instead of picking random points. It is our feeling that this would dramatically increase the performance of our algorithm.

5 Discussion and Future Work

In this paper we have presented a simple iterative technique to seek Voronoi vertices of a set of planar objects while being restricted to performing only nearest point queries on the objects. Beginning at a random seed point, we query for the three nearest points on different obstacles and jump to the center of their circumcircle and repeat. Within a few iterations, this process usually converges towards the desired Voronoi vertex. The spirit of this type of solution – make a random initial guess, perform some computation which gives a refined guess, and reiterate – will be familiar to puzzle solvers. A well-known puzzle is the following. We are given a self-referential sentence of the form

"In this sentence, the number of occurrences of 'a' is __ , of 'b' is __ , . . , of 'z' is __."

and the desired solution is a 26-tuple of positive integers that makes the sentence true. A simple, intuitive, and robust method to solve these genre of puzzles (works for most sentences in any language) is to simply *guess at random* a 26-tuple, say twenty six ones, and plug it in to get the (false) sentence: "In this sentence, the number of occurrences of 'a' is one, of 'b' is one, ..., of 'z' is one." Now count the number of a's, b's, and so on in this false sentence; this becomes the refined 26-tuple. Repeat this for a few iterations, and more often than not, you will arrive at a stable solution (there may be several): a self-documenting sentence. Hofstadter calls this process *Robinsonizing* after the logician Raphael Robinson [13].

The underlying space of 26-tuples being discrete, the *exact* solution is achieved in the case of the puzzle. This is sometimes true in our Voronoi vertex-finding technique as well; for instance, when the Voronoi vertex is defined by points on convex objects where the tangent is not defined (like vertices of polygons) we achieve it exactly. Otherwise, when the Voronoi vertex is defined by smooth object boundaries, we achieve convergence: we get closer and closer to the Voronoi vertex without actually reaching it. As an example, the Voronoi vertex in Figure 5, is achieved exactly from a suitable seed point in its vicinity; however, for that in Figure 1, we achieve only convergence (the Voronoi vertex is defined by interiors of the line segments). An open problem in this context is to study rates of convergence.

We do not always achieve convergence to a Voronoi vertex from any seed point. We defined the region of convergence of a Voronoi vertex to be the set of seed points that lead to convergence to that vertex. When the initial random guess is outside the region of convergence, our technique was observed to lead to oscillations with some finite period. We present some geometric conditions when each of convergence/oscillation occurs. We do not yet have simultaneously necessary *and* sufficient conditions for convergence. In other words, we have not yet been able to precisely characterize regions of convergence which we believe to be an interesting open problem.

In case the initial seed point misses the region of convergence (sequence oscillates), a different seed point is required for convergence. We showed that the region of convergence is of non-zero measure proving the probabilistic completeness of our technique: the probability of missing the region of convergence in N random tries decreases exponentially with N, if a Voronoi vertex exists. We proved this by showing that the region of convergence included an ε-ball around the Voronoi

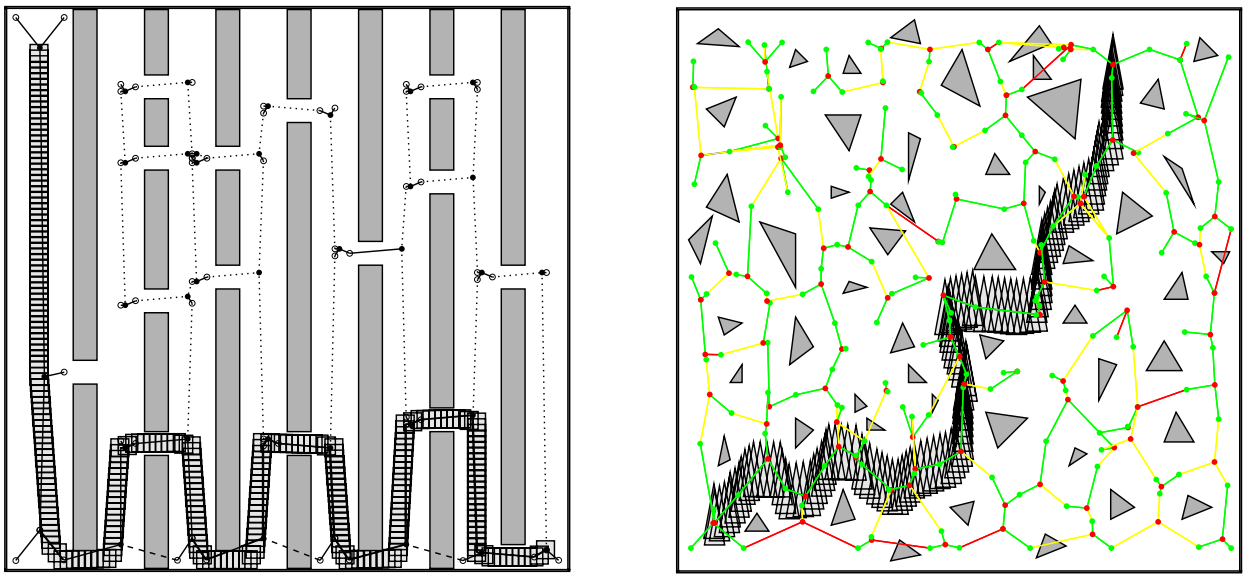

Figure 13: (a) There are a number of possible paths from the top left to the bottom right, of which our algorithm finds a relatively short one. The data structure consists of only 37 vertices and 62 via-points and is computed in 2.54 seconds on average. (b) This scene is relatively difficult for Voronoi-based approaches because of the large number of obstacles. The data structure is again very compact and comprises only 105 vertices and 168 via-points. Due to the inefficient way of picking starting points however, our algorithm takes 24.14 seconds to solve this scene. We believe that this can be improved dramatically by choosing a better strategy.

rectangular robot having to pass through a narrow corridor in order to reach the goal configuration. The scene, together with our network representation of (part of) the Voronoi diagram and a path for the robot, is shown in Figure 12. The safe edges in the network are drawn as solid lines, whereas the unknown edges are dotted. Notice that only part of the Voronoi diagram, which is sufficient to solve the posed problem, is built up. The resulting path for the robot was computed in an average time of 1.21 seconds, taken over 20 runs of the program.

We compared our method with the others mentioned above and observed a comparable performance in timings. Our approach also usually resulted in a more compact network. We discuss a couple of more scenes now.

The second consists of a number of parallel rectangles with little room for the robot to move between them; see Figure 13(a). The experiments again indicate that the data structure from our approach is smaller, while the running time is competitive to the other approaches.

In the third and last scene, a triangular robot moves from the bottom left to the top right amidst a large number of small obstacles. This scene is relatively difficult for Voronoi-based approaches because of the complexity of the Voronoi diagram. The running time of our algorithm reflects this complexity, by taking 24.14 seconds on average to compute a network of 105 vertices and 168 via-points, while the random motion planner and the approximated diagram performed in about a third of that time. Our approach still produces a more succinct network; the degrade of performance is because of repetitive labor: we observed from experiments that in cluttered scenes almost 96% of the trials (starting from a random point) result in Voronoi vertices which were already found in

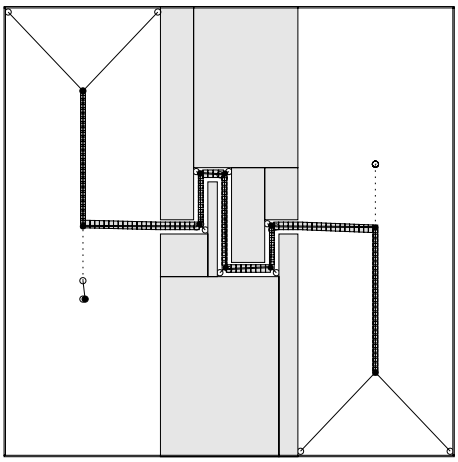

Figure 12: The robot (a tiny square) has to pass through a narrow corridor to move from the left side of the scene to the right side. Our algorithm solves this problem through a network consisting of 11 vertices and 20 via-points, which is built up in 1.21 seconds on average.

implies that collision detection must be performed only for those "unknown" edges which actually participate in a candidate path. Inferring the presence of an edge while deferring its evaluation is very convenient, as the latter requires the computation of the via point: the impact on the global performance is relevant.

4.3 Experimental Results

A motion planner for planar robots based on the method described in the previous sections has been implemented in C++ on a Silicon Graphics Indigo workstation, which is based on an R3000 processor clocked at 33 MHz and rated with 24.2 SPECfp92 and 22.4 SPECint92. The program implements the computation of Voronoi vertices, the incremental construction of the graph $\mathcal{G}$, and the search for a collision-free path on the diagram. The source code consists of approximately 2000 lines, 25% of which implement our algorithm, while the rest is devoted to managing the graphical interface and handling user interaction.

Testing has been based on a set of several representative scenes, each of which have their own peculiarities. We compared the performance of our algorithm to other methods by running two existing motion planning algorithms, which we now briefly outline, on the same test scenes. The first approach, called the *approximated Voronoi diagram approach* [28], constructs a network of cells which approximates the configuration space Voronoi diagram. It subdivides the configuration space into primitive cells, and recursively subdivides the cells that are intersected by the Voronoi diagram. This method combines the advantages of a Voronoi-based approach (completeness, high clearance) with fast execution speed. The second algorithm to which we compare our method is the motion planning approach developed by Overmars [24] called the *random motion planner*, which builds up a network of nodes in the configuration space by connecting randomly chosen configurations of the robot using a simple local planner. The resulting algorithm turns out to be generally applicable and very efficient in terms of running time. Both these methods have been implemented on the same architecture as our method, which facilitates a fair comparison.

We now present experimental results for three of the test scenes. The first scene consists of a

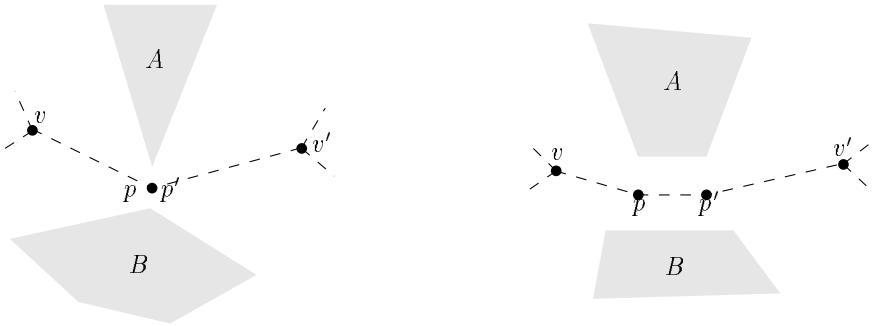

Figure 11: Connecting Voronoi vertices through their via points.

to the via point of obstacles A, B, we begin the iteration with $q_0 = v$. When at q_i, we obtain the points on A, B closest to q_i, and set q_{i+1} as their midpoint. It can be shown easily that this sequence converges to the via point of A, B.

Via points help to precisely delimit the portion of the Voronoi diagram which is influenced by a given vertex, and this makes it possible to incrementally assemble the complete diagram by appropriately joining suitable sub-diagrams.

Definition 4.1 *Let the Voronoi vertex v be given, and p_1, p_2, p_3 denote the associated via points. The portion of the Voronoi diagram $\mathcal{V}$ relative to v, denoted as $\mathcal{V}_v$, is defined as*

$$\mathcal{V}_v = \bigcup_{i=1,2,3} \widehat{vw_i}$$

where $\widehat{vw_i}$ denotes the portion of the Voronoi diagram between points v, w_i that lie on the Voronoi diagram, and $w_i = p_i$ if no other Voronoi vertex v' lies on $\widehat{vp_i}$, $w_i = v'$ else.

Consider now the two Voronoi vertices v and v' which are adjacent in $\mathcal{V}$, and let $\widehat{vv'} = bis(A, B) \cap \mathcal{V}$ be the Voronoi edge connecting them. Consider the set S of all via points which belong to $\widehat{vv'}$. Assume that S is not empty, and let p and p' be the via points associated to v and v', respectively, and lying on $\widehat{vv'}$. Only two cases are then possible, namely either S is a singleton and therefore $p = p'$, or S contains an infinite number of elements, the points of the line segment $\overline{pp'} \subset \widehat{vv'}$. The two cases are depicted in Figure 11, left and right, respectively.

The complete diagram can be incrementally obtained by appropriately joining the sub-diagrams corresponding to the portions relative to different vertices. If part of the Voronoi diagram V has been computed, and the portion $\mathcal{V}_v$ relative to a new vertex v must be added, it is then sufficient to keep track of the via points found so far and of their defining obstacle pairs to correctly join $\mathcal{V}_v$ to $\mathcal{V}$.

Our implementation builds a graph $\mathcal{G}$ which captures the connectivity of the Voronoi diagram, but differs from $\mathcal{V}$ in some regards for computational efficiency. Nodes in $\mathcal{G}$ (vertices or via points) correspond to points of $\mathcal{V}$, but edges in $\mathcal{G}$ do not always represent edges in $\mathcal{V}$: The presence of an edge in $\mathcal{G}$ only indicates a *possible* motion for the robot between its end nodes. This deviation from edges in $\mathcal{V}$ is motivated by the attempt to minimize the number of edges which must be checked for collision although they might not appear in the path which is returned as result of the computation.

It is desirable to be able to infer the presence of as many edges as possible, delaying the check for the safety of the corresponding path segments ("edge evaluation") as long as possible. This

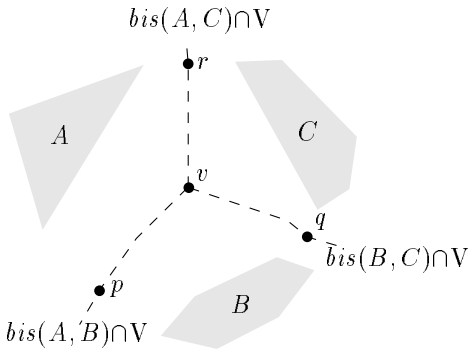

Figure 10: A Voronoi vertex and the three associated via points.

the triple-seed pair $(\mathcal{T}', v)$ in place of $(\mathcal{T}, p)$. However, this could lead to triple-seed cycles: $(\mathcal{T}', v)$ could lead back to $(\mathcal{T}, p)$ eventually. If cycles are detected, a random p is chosen again.

All computations involved in determining a Voronoi vertex for a given candidate triple, except for the occasional computation of nearest triples (twice for a random p, once otherwise), is performed with local information only (independent of n) making the technique very efficient. The only geometric computations required throughout are that of nearest point queries (function ϕ) and their Voronoi vertex (function γ).

The approximation of the Voronoi vertex for triple $\mathcal{T}$ and with seed $p_0 = p$ as $p_i = (\gamma \circ \phi)^i(p_0)$, $i = 0, 1, \ldots$, can be terminated when p_i and p_{i+1} are "sufficiently close," i.e., are at most some chosen parameter ϵ apart. It has been observed in practice that when convergence occurs, it is very fast: distances $d(p_i, p_{i+1})$ start decreasing after a couple of iterations (five at most) and between ten and fifteen iterations, $d(p_i, p_{i+1})$ is less than $\epsilon = 10^{-6}$. This observation can be used to detect oscillations of the sequence $\{p_i\}$; if it does not converge for a relatively large number of iterations (50 in our implementation just playing it very safe) we assume that oscillations have set in. However, even if we falsely conclude oscillations (in case of extremely slow convergence), the completeness of our method (Theorem 3.1) is not affected.

4.2 Incremental Construction of the Diagram

In this section we show how to incrementally infer the topology of the Voronoi diagram of the obstacles.

To perform path planning, we build up a connectivity structure (graph) $\mathcal{G}$ around the Voronoi vertices. The graph is computed by incremental construction: for every Voronoi vertex v found for a triple $\mathcal{T}$, the portion of the Voronoi diagram local to v and $\mathcal{T}$ is computed as follows. Consider the portion of the Voronoi diagram given by $\mathcal{V}_{A,B} = bis(A, B) \cap \mathcal{V}$. A *via point* defined by the obstacle pair A, B is a point $p \in \mathcal{V}_{A,B}$ such that $\forall q \in \mathcal{V}_{A,B} : d(p, A) \leq d(q, A)$. We can uniquely associate three via points to every Voronoi vertex as follows. Suppose that v is defined by the obstacle triple $\mathcal{T} = \{A, B, C\}$; notice that v is an endpoint of $\mathcal{V}_{A,B}$ by definition. Now follow $\mathcal{V}_{A,B}$ while walking away from v until a via point is encountered. This uniquely associates a via point to $\mathcal{V}_{A,B}$. Analogously we can associate via points to $\mathcal{V}_{A,C}$ and $\mathcal{V}_{B,C}$; see also Figure 10.

To determine the via points associated to a given Voronoi vertex v, we perform an iterative technique much like that for Voronoi vertices. The difference is we are dealing now with two obstacles instead of three. Suppose we are at the Voronoi vertex v for obstacles A, B, C. To get

4 Path Planning and Experiments

In previous sections we considered the problem of determining a Voronoi vertex for three planar objects via a sequence of closest-point queries on the three objects. In this section we apply this vertex-finding technique to planning a path for a planar robot with two degrees of freedom that avoids a set of n static planar obstacles in its workspace. We have currently assumed polygonal obstacles for the implementation because of the lack of availability of software for handling curved objects. Non-convex polygons are decomposed into convex pieces. Our method is based only on nearest point computations performed in the workspace; replacing configuration space operations by equivalent workspace ones has also been considered in [24, 27].

Going from three obstacles to a scene with several obstacles raises the following key issue: not every triple of obstacles defines a vertex in the Voronoi diagram of all obstacles. For example, for convex obstacles in the plane there exists only a linear number of Voronoi vertices for the cubic number of triples [23]. Therefore, we first need to devise a selection strategy that can efficiently suggest candidate Voronoi triples.

Applying techniques from previous sections, we compute the Voronoi vertices for these triples of obstacles. Notice that this could lead to oscillation for a given triple of obstacles even if they define a vertex (see Section 2.1); we have to decide how to detect and deal with this case. Next, to perform path planning, we build up a connectivity structure (graph) $\mathcal{G}$ around the Voronoi vertices determined. This graph does not represent the exact topology of the Voronoi diagram, but for this application it is a sufficient approximation thereof. Finally, a path may be searched in $\mathcal{G}$ between given configurations using familiar methods.

Given start and goal configurations, $\mathcal{G}$ is incrementally built up until there is a path from start to goal. Additional Voronoi triples are sought as and when necessary. When $\mathcal{G}$ is completely determined (no new Voronoi triples are detected) but no path exists in $\mathcal{G}$, we report failure. We elaborate in Section 4.2.

4.1 Determining Voronoi Triples

As indicated in the previous section, we desire a strategy that can suggest candidate Voronoi triples in a computationally efficient manner and yet with a high "hit ratio". A candidate triple can then be subjected to the iterative procedure described in previous sections.

The first step is the selection of a point p in configuration space according to some criterion. Borrowing from Overmars and Švestka [24], we noticed that simply choosing p uniformly at random over the configuration space (not necessarily in free space) is a viable choice. Next, the nearest three obstacles to p are determined by closest-point queries. These are denoted as $\mathit{triple}(p)$ and considered a candidate Voronoi triple.

This strategy provides a triple $\mathcal{T}$ of obstacles and a point p such that $\mathit{triple}(p) = \mathcal{T}$. The results of the previous section are applied here, and the sequence $(\gamma \circ \phi)^i(p_i)$, $i = 0, 1, \ldots$, relative to $\mathcal{T}$ and with $p_0 = p$ is used to compute a Voronoi vertex for the given three obstacles. Notice that while constructing the terms p_i of the sequence, it is possible that for some p_j, $\mathit{triple}(p_j) \neq \mathcal{T}$. To avoid the repeated computation of the three nearest obstacles to p_i (which is an expensive operation) we initially disregard this possibility and assume that $\mathit{triple}(p_i) = \mathcal{T}$. Only when a Voronoi vertex v for triple $\mathcal{T}$ is achieved, the triple $\mathcal{T}'$ of three nearest obstacles is recomputed for v. If $\mathcal{T}' = \mathcal{T}$, then v is a Voronoi vertex for the entire scene. Therefore, we add v to the set of Voronoi vertices discovered and continue with another random seed point. On the other hand, if $\mathcal{T}' \neq \mathcal{T}$, then v is not a Voronoi vertex for the entire scene (although it is one for the triple $\mathcal{T}$) and we begin with

Assuming that ε satisfies (3), the further condition we are looking for can be obtained by simplifying $Y < \varepsilon/6$ which gives

$$\varepsilon < \frac{\sin\phi - \sin\theta}{3 + |\cos\phi|}. \tag{4}$$

Since any ε that satisfies (4) also satisfies (3), the required choice of ε can be made from (4).

Remark: If the initial circle is not a unit circle but has radius $r > 0$, then the right hand side of (4) should be multiplied by a factor of $1/r$. □

Lemma 3.2 *Let v be a Voronoi vertex for three convex objects S_1, S_2, S_3. Consider a seed point p_0 at distance ε from v. Then, for sufficiently small ε, the next point $p_1 = \rho(p_0)$ in the iterative scheme, is at distance at most $\varepsilon/2$ away from v.*

Proof. Recall that $near(S, p)$ denotes the closest point on S from p. Since the S_i are convex and p_0 is ε away from v, $s_i = near(S_i, v)$ is at distance at most ε from $near(S_i, p_0)$. Further, since the Voronoi circle of v is tangent to S_1 (by definition of a Voronoi circle), $near(S_i, p_0)$ can be approximated, for ϵ sufficiently small, by the projection of p_0 on this common tangent. This projected point s_i' is clearly at a distance of at most ε from s_i.

Lemma 3.1 shows that the center of the circumcircle of the points $\{s_1, s_2, s_3'\}$ is at most $\varepsilon/6$ away from v for sufficiently small positive ε. Repeating the argument, we get that the center of the circumcircle of $\{s_1', s_2', s_3'\}$ is at most

$$3 \cdot \frac{\varepsilon}{6} = \frac{\varepsilon}{2}$$

away from v as required. □

The following theorem shows probabilistic completeness of our iterative technique.

Theorem 3.1 *Let v be a Voronoi vertex for three convex objects. The probability that n seed points, chosen uniformly at random from a set including v and of finite measure* A, *all fail to converge to v falls exponentially.*

Proof. From Lemma 3.2, it follows that there is an ε neighborhood of convergence for v. Let the measure of this neighborhood be α; $\alpha > 0$. The probability of hitting the neighborhood of convergence with a (uniformly distributed) random seed point is α/A. Thus the probability that n random tries all fail to converge is $(1 - (\alpha/\mathrm{A}))^n$. □

Remarks: As stated, the theorem above holds for convex objects. Even if the objects are not convex but can be decomposed into a finite number of convex components, then a region of convergence of non-zero measure can still be shown to exist for every Voronoi vertex. For small enough ε, the closest points from anywhere in the region of convergence to any object will lie in the same convex component of that object. In particular, the theorem holds for polygonal objects since they are thus decomposable.

Also, the assumption that A be of finite 2-D measure is also not really necessary. We can generate seed points from points on the boundary of obstacles and directions in $[0, 2\pi)$. These sets are of finite 1-D measure. A set of non-zero 2-D measure in the plane will then correspond to sets of non-zero 1D measure. The details are omitted.

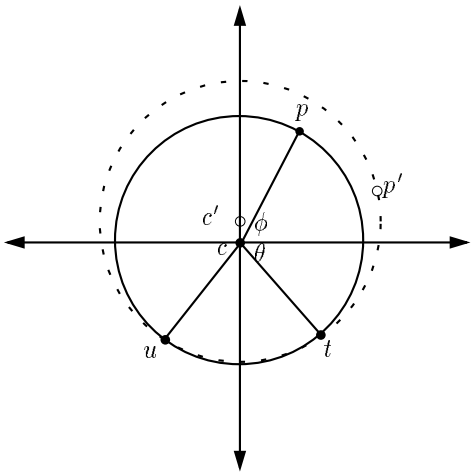

Figure 9: Proving a bound on how much the center c of a circle defined by three points u, t, p can move when one of the points p moves by a distance ε. The center of the circle defined by u, t, p' is c'.

the seed point initially misses the region of convergence. Will a sufficiently large number of trials guarantee a "hit" with high probability? This is the subject of this section; we answer the question in the affirmative in Theorem 3.1.

Lemma 3.1 *Consider a circle defined by three points on its boundary. There exists a sufficiently small ε such that if one of the three points moves by ε tangentially, then the center of the circle shifts by at most $\varepsilon/6$.*

Proof. See Figure 9. Wlog consider a unit circle centered at the origin. Rotate the coordinate axes such that the fixed points t and u have polar angles θ and $\pi - \theta$, respectively; the perpendicular bisector of the two fixed points is the y axis. Let $p = (\cos\phi, \sin\phi)$ be the third point as shown, $\sin\phi > \sin\theta$.

Let p' be the point obtained by moving p by a distance ε along direction z. Simple geometry shows that the center of the circle defined by u, t, p' has coordinates:

$$\left(0\,,\,\frac{\varepsilon^2 + 2\varepsilon\cos(\phi - z)}{2(\sin\phi - \sin\theta + \varepsilon\sin z)}\right).$$

If p' is moved tangentially, then $z = \phi \pm \pi/2$. In such a case, $\cos(\phi - z) = 0$ and the y-coordinate of the new circle center becomes

$$Y = \frac{\varepsilon^2}{2(\sin\phi - \sin\theta \pm \varepsilon\cos\phi)}.$$

It now suffices to show that $|Y| < \varepsilon/6$ for sufficiently small ε. Note that $|Y| = Y$ if $(\sin\phi - \sin\theta \pm \varepsilon\cos\phi) > 0$ which is true for (remember $\sin\phi > \sin\theta$)

$$\varepsilon < \frac{\sin\phi - \sin\theta}{|\cos\phi|}. \tag{3}$$

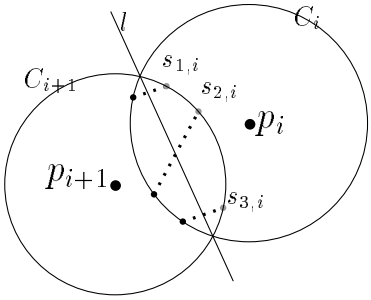

Figure 8: Establishing that the sequence $\{p_i\}$ cannot have oscillations of period two if the objects are completely visible from each other. Let oscillations of period two between points p_i and p_{i+1} have set in after some number of iterations. $\{s_{1,i}, s_{2,i}, s_{3,i}\}$ denotes the triple of nearest points from p_i on the objects; C_{i+1} is the circle through these points and from the iterative scheme, p_{i+1} is its center. The closest points on the objects from p_{i+1} are also shown and they define a circle which coincides with C_i. The line l through the intersections of the circles then has to intersect all three objects contradicting the premise of complete visibility.

may be labelled "towards S_2" and "away from S_2". Take the latter direction and travel on it. At some sufficiently distant point on this bisector, because the line though ab separates S_2 from the current point, a or c clearly become nearer than point b and in fact nearer than the entire object S_2. This contradicts the statement that $bis(S_3, S_1)$ lies entirely in $Vor(S_2)$.

To see that C1 $\Rightarrow$ C3, observe that convergence to a finite point is impossible because of lack of any Voronoi vertex (Lemma 2.1) and so is divergence to infinity because that would imply there be three collinear points on the three objects nearest to a point at infinity (which is not possible because these three points would have to be on their respective boundaries. This in turn is ruled out because one object completely hides the other two from each other). Finally, C3 clearly implies C1 because Voronoi vertices are the only stable points for the iterative scheme. □

If convergence and oscillations are the only outcomes possible, then the previous theorem gives an easy geometric criterion to conclude oscillations from any seed point. The reader might expect that there exists a dual geometric criterion which guaranteed convergence, i.e., if three objects are completely visible, then the sequence converges to a Voronoi vertex (which exists due to the equivalence of C1 and C2 in Theorem 2.2). However, the example in Figure 1 counters this claim. Nevertheless, we can show that, if S_1, S_2, S_3 are completely visible from each other, oscillations of period two are impossible (See Figure 8). Note that the example in Figure 1 has oscillations of period three from points outside the region of convergence.

3 Probabilistic Completeness: Multiple Trials

In Section 2, we studied some convergence properties of the iterative scheme on a single trial. We noticed that there is a region of convergence for every Voronoi vertex; a seed point taken from within this region converges to the vertex. The region of convergence, however, may not include the entire plane; seed points taken from outside this region can lead to oscillations after some iterations. Therefore, there is clearly a need for multiple trials of the iterative scheme in case

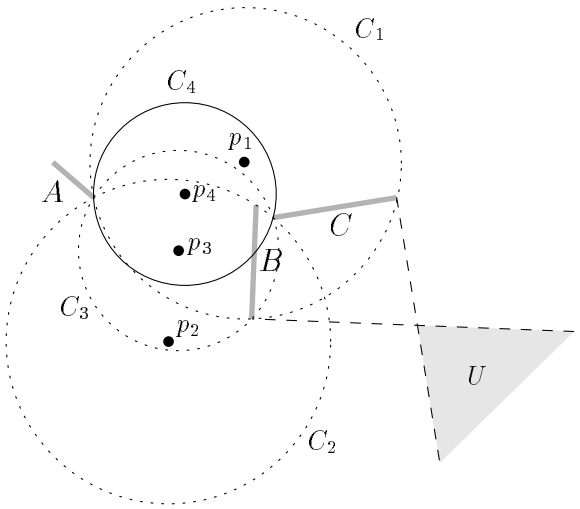

Figure 6: A more complicated example in which oscillation occurs. A, B, C are the three line-segment objects. For $p_0 \in U$ shown, $p_1, \ldots, p_4$ result and thereafter $p_1, \ldots, p_4$ repeat. The circumcircle of which p_i is the center is denoted as C_i; i.e., the closest points on the objects from p_{i-1} lie on C_i. The difference between this example and the previous one, other than the fact that they are of different periodicity, is that the initial closest points on objects are "visible" from each other in this example but in the last figure, object S_2 hides the initial closest points on S_1 and S_3. This rules out any prediction of convergence/oscillation in terms of visibility between the initial closest points alone. However, there are connections between visibility and convergence; we explore some in Section 2.2.

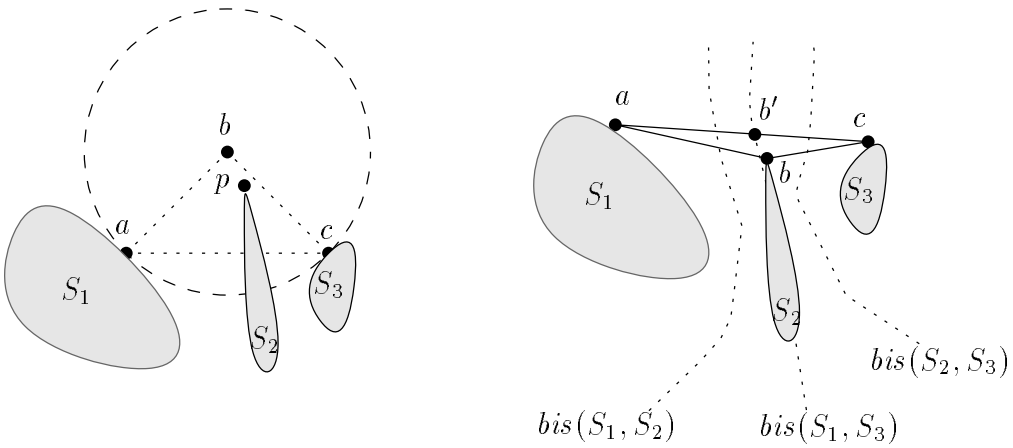

Figure 7: Illustrations to the proof of Theorem 2.2. On the left, S_2 (completely) hides S_1 from S_3. Then any point (like b shown) equidistant from S_1, S_3 will be closer to S_2. Therefore no Voronoi vertices can exist for the three objects. On the right, no object completely hides the second from the third. The proof works by showing that the three Voronoi edges have to intersect.

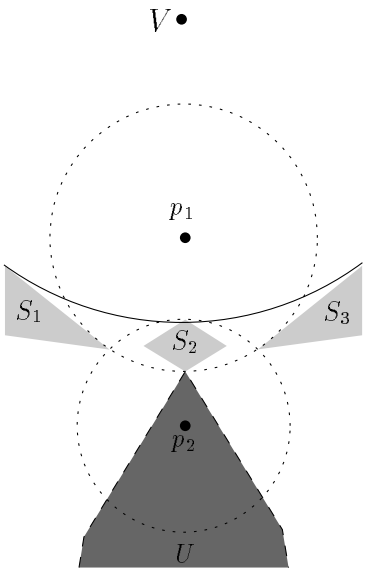

Figure 5: A case of three objects and seed points leading to oscillations in the sequence $\{p_i\}$. The S_i are the objects and U is the dark region bounded by segments perpendicular to the edges of the objects. For any initial point from U, p_1 is as shown. It is easy to verify that $p_{2i} = p_0$ and $p_{2i+1} = p_1$. The three objects have a Voronoi vertex V; the relevant arc from the Voronoi circle is indicated.

C3 *The iterative procedure from any seed point on objects S_1, S_2, S_3 produces a sequence $\{p_i\}$ that does not converge.*

Proof. We first show the equivalence of C1 and C2 and then that of C1 and C3.

To see that C2 $\Rightarrow$ C1, consider the left of Figure 7. S_1, S_2, S_3 are three objects such that S_2 hides S_1 from S_3. To prove that no Voronoi vertex can exist, we show that any point equidistant from S_1, S_3 will be strictly nearer to S_2.

First a simple claim: in an isosceles triangle, the apex is strictly closer to any interior point than to any of the two vertices of the base (this is true because the circle centered at the apex and with radius one of the non-base edges includes all interior points of the triangle in its interior).

Consider a point b equidistant from S_1, S_3. Let a, c be the points on S_1, S_3 closest to b. Therefore $\triangle abc$ is isosceles with apex b. From the hypothesis that S_2 hides a from S_3, we get that the edge ac intersects the interior of S_2. This clearly implies (from the claim made above) that there exists a point $p \in \partial S_2$ such that $d(b,p) < d(b,a) = d(c,a)$.

Now we prove $\neg$(C2) $\Rightarrow \neg$(C1). Notice that $\neg$(C2) readily implies that (and is implied by) there exists a triangle abc with $a \in S_1, b \in S_2, c \in S_3$ such that each edge with end-vertices on two objects does not intersect the third. See the right of Figure 7. Consider the three bisectors $bis(S_1, S_2)$, $bis(S_2, S_3)$, and $bis(S_3, S_1)$ between the three objects. Suppose there still exists no Voronoi vertex then no two of these bisectors intersect and so one of the bisectors lies entirely in the Voronoi region of the third. Without loss of generality, let $bis(S_3, S_1)$ lie in $Vor(S_2)$. Modify a, b locally so that the entire line through ab does not intersect S_2. Since the bisector between two objects always intersects any line segment between the two objects, let $bis(S_3, S_1)$ intersect ac at b'. Now since $ab \cap S_2 = \emptyset$, $b' \cap S_2 = \emptyset$. Therefore there are well defined directions from b' along $bis(S_3, S_1)$ which

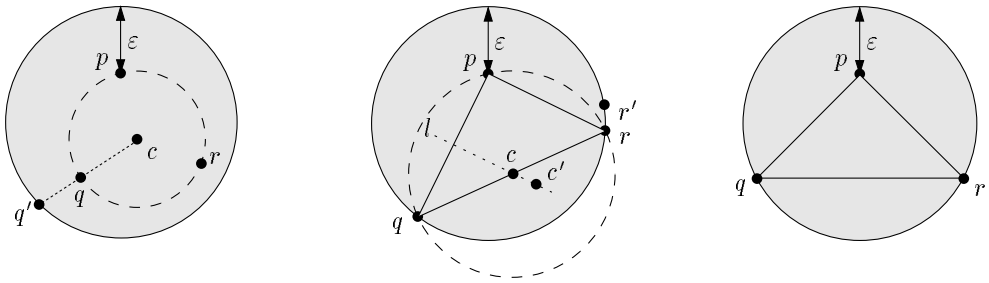

Figure 4: Illustration to the proof of Corollary 2.1 to Lemma 2.3. The corollary gives a upper bound on the radius of circle C in terms of that of disc D where C is defined by three points (p, q, r) that lie in D and include C's center in their convex hull. The proof is by showing that the lower bound is achieved by the case of p, q, r being in the right most configuration. The quantity ε is maximum of the distances of p, q, r from the the boundary of D. Obviously if $\varepsilon = 0$, $C = D$.

2.1 Examples of oscillations in $\{p_i\}$

We now give an example of a sequence $\{p_i\}$ that oscillates although the objects define a Voronoi vertex. Consider three sets S_1, S_2, S_3 as depicted in Figure 5, and let U denote the region shown. For any $p_0 \in U$, $\phi(p_0)$, the set of closest points on objects, is the same. Therefore $p_1 = (\gamma \circ \phi)(p_0)$ is also the same for any $p_0 \in U$. Now, $p_2 = (\gamma \circ \phi)(p_1)$ is back in U and therefore the sequence $\{p_i\}$ will oscillate between the positions of p_1, p_2 for any seed point from U. The three objects, however, do have a Voronoi vertex (V in the figure). The region of convergence of V therefore definitely does not include U.

A more complicated example of an oscillation with period three is now presented. Consider the segments A, B, C as shown in Figure 6, and any point $p_0 \in U$. Points p_1, p_2, p_3, p_4 follow as shown and thereafter $p_2, \ldots, p_4$ repeat. This example also indicates that periods of oscillation need not be bounded. By varying the lengths of the segments, any finite periodicity can be obtained.

2.2 Visibility and Convergence

In this section we present some results relating the concept of visibility between objects to the question of convergence or oscillation of the sequence $\{p_i\}$. This is motivated by the oscillation examples shown previously.

An object S_1 is said to *(completely) hide* S_2 from S_3 (or S_3 from S_2) if the set

$$CH(S_2 \cup S_3) \setminus int(S_1)$$

is disconnected. On the other extreme, in a scene consisting of objects (S_1, S_2, S_3), objects S_2, S_3 are said to be *(completely) visible* from each other if $CH(S_2 \cup S_3) \cap S_1 = \emptyset$. Three objects are completely visible from each other if they are pairwise completely visible.

Theorem 2.2 *The following three statements are equivalent.*

C1 *S_1, S_2, S_3 have no Voronoi vertex.*

C2 *One of S_1, S_2, S_3 hides the second from the third.*

Proof. Without loss of generality, let $\varepsilon_p = \varepsilon$. See Figure 4. p is ε away from the top most point of D. Let us select q, r such that

1. $q, r \in D$ (let T refer to $\triangle pqr$, and C the circumcircle of T),
2. T is non-obtuse (this is equivalent to the condition that the $center(C) \in CH(p, q, r)$), and
3. $rad(C)$ is maximized.

The first claim is that p, q have to be on ∂D. To see this, let pqr form a non-obtuse triangle and assume that q is not on the ∂D. See the left part of Figure 4. Let c denote the center of C shown in dotted lines. Push q outwardly from c onto ∂D, and let the new point be q'. The operation does not destroy the property of non-obtuseness. The circumcircle of $\triangle pq'r$ with center c' clearly has larger radius than does T. Similarly, r can be pushed back to a point r' along ∂D.

The next claim is that given q (on ∂D), the r such that $rad(C)$ is maximized is a point (also on ∂D) such that T is right-angled at p. See the middle part of Figure 4. This follows from the following argument. Consider the perpendicular bisector l of pq. Since T is right-angled at p, l intersects qr (the hypotenuse) at its midpoint c, the center of the circumcircle of T. Let, if possible, there exist an non-obtuse triangle pqr' with center of circumcircle c' with larger radius of circumcircle. Then, c' has to be more distant from pq than is c implying $rc' < rc$. Thus, to make $c'r' = c'q = c'r$, r' has to be away from r as shown. This implies that $\triangle pqr'$ is obtuse.

Therefore, the problem reduces to fitting a triangle right-angled at p with largest hypotenuse (the radius of the circumcircle of a right-triangle is half the length of the hypotenuse) with the two other vertices being on the circumference of D. It should come as no surprise (and this can be verified by elementary calculus) that this extremal right-triangle is isosceles. See the right part of Figure 4. It can be shown that the radius r of the circumcircle of this triangle satisfies

$$r^2 = \frac{R^2}{2} + \frac{(R-\varepsilon)}{2}\sqrt{R^2 + 2R\varepsilon - \varepsilon^2} \leq R^2 - \frac{\varepsilon^2}{2}$$

Thus,

$$\frac{r}{R} \leq \sqrt{1 - \frac{1}{2}\left(\frac{\varepsilon}{R}\right)^2} \leq 1 - \left(\frac{\varepsilon}{2R}\right)^2 .$$

□

The following theorem gives a sufficient condition for convergence based on Lemma 2.3.

Theorem 2.1 *If in every iteration, the new point p_{i+1} lies in the convex hull of the three closest points from p_i (that is, every triple of closest points defines a non-obtuse triangle), then the sequence $\{p_i\}$ converges to a Voronoi vertex of S_1, S_2, S_3.*

Proof. Let the disk circumscribing the elements of $\phi(p_i)$ be denoted by D_i of radius R_i. The center of D_i is p_{i+1}. The hypothesis along with Lemma 2.3 imply that R_i is a non-increasing sequence, $R_{i+1} \leq R_i$ and if $R_{i+1} = R_i$, $D_i = D_{i+1}$. (The corollary to Lemma 2.3 gives an upper bound on R_{i+1}/R_i in terms of R_i.)

This implies that the sequence $\{D_i\}$ converges to a disk D^*. Therefore the sequence of corresponding centers $\{p_i\}$ converges to a point p^*. Lemma 2.1 implies $p^* = Vor(S_1, S_2, S_3)$. □

The above proof is quite general; it only relies on the definition of a Euclidean distance metric. Notice that this proof also holds for non-regular or non-compact sets S_1, S_2, S_3. It is also extendible to objects in higher dimensions (with appropriate modifications to the number of objects forming a Voronoi vertex *etc.*).

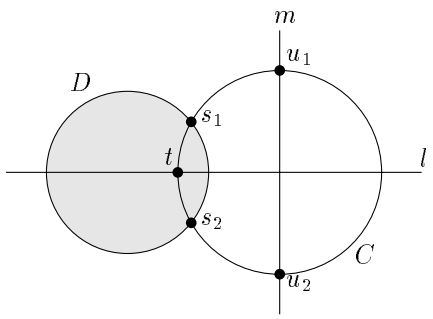

Figure 3: Illustration to the proof of Lemma 2.3 which shows that the radius of C is upper-bounded by that of D if three points defining C belong to D (an arbitrary disc) and include the center of C in their convex hull. The proof is by contradiction: if the radius of C is larger or equal, then the center of C cannot lie in the convex hull of the three points defining it.

Lemma 2.3 *Given a circular disc $D \subset R^2$ and three non-collinear points $p, q, r \in D$, let C be the unique circle through p, q, r. If $center(C) \in CH(p, q, r)$ then $rad(C) \leq rad(D)$. Equality implies C coincides with ∂D.*

Proof. Given the hypothesis, towards a contradiction, suppose that $rad(C) > rad(D)$. Since C passes through points $p, q, r \in D$, $C \cap D \neq \emptyset$ and D cannot exist in C's interior. Denote the intersection points of C with ∂D as s_1, s_2, respectively (see Figure 3). Now let l be the line through the centers of C and D, t the unique intersection point of l with $C \cap D$, m the line through $center(C)$ perpendicular to l, and u_1, u_2 the intersection points of m with C. Since it is supposed that $rad(C) > rad(D)$ it follows that $d(t, center(C)) > d(t, center(D))$, and hence that points u_1, u_2 are outside D. Thus any line through the open (and disjoint) arcs s_1u_1 and s_2u_2 of C separates p, q, r from $center(C)$. From Lemma 2.2 it now follows that $center(C) \notin CH(p, q, r)$ which contradicts the hypothesis.

To prove the second claim, assume $rad(C) = rad(D)$. Towards a contradiction, further suppose that C is different from ∂D. These imply that the centers of C, D cannot be coincident. Also C, D intersect in an arc as before and $s_1, s_2, l, t, m, u_1, u_2$ can be uniquely defined. Via a similar argument, it again follows that $d(t, center(C)) > d(t, center(D))$ and any line through the open arcs s_1u_1, s_2u_2 separates p, q, r from $center(C)$ contradicting $center(C)$ being in $CH(p, q, r)$. □

In the following corollary we give a lower bound on the ratio $rad(C)/rad(D)$. Refer to the proof of the previous lemma. Let $\varepsilon_p, \varepsilon_q, \varepsilon_r$ denote the distances of p, q, r, respectively, from the circumference of D (that is, p is at distance $r - \varepsilon_p$ from $center(D)$ and likewise for q, r), and let $\varepsilon = \max(\varepsilon_p, \varepsilon_q, \varepsilon_r)$. Let $r = rad(C), R = rad(D)$. Note that $R > \varepsilon = 0$ if and only if p, q, r lie on the circumference of D.

Corollary 2.1 *Let r, R be the radii of C and D, respectively. Then,*

$$\frac{r}{R} \leq 1 - \left(\frac{\varepsilon}{2R}\right)^2.$$

maps a triplet of points taken from the three sets to a point equidistant from the triplet (which can be at infinity, if the points are collinear). In other words,

$$\gamma(s_1, s_2, s_3) = q \text{ such that } d(q, s_1) = d(q, s_2) = d(q, s_3) \tag{1}$$

(notice that γ is just *Vor* when restricted to point objects.) Our first goal in this paper is to study the behavior of the composition of γ with ϕ which we denote by $\rho : R^2 \to R^2$.

$$\rho = \gamma \circ \phi. \tag{2}$$

Specifically, we wish to investigate the relationship between $Vor(S_1, S_2, S_3)$ and $p_i = \rho^i(p_0)$ for sufficiently large i while varying the initial point p_0 over R^2. In the next section we study geometric conditions for convergence of sequence $\{p_i\}$ to $Vor(S_1, S_2, S_3)$.

Remark: Note that a Voronoi vertex could exist at infinity. However, we still choose to use the term "convergence" to the Voronoi vertex rather than "divergence" if the Voronoi vertex is finally attained.

2 Convergence of a Single Trial

In this section we study conditions for the convergence or the oscillation for the sequence $\{p_i\}$. We begin with a sufficient condition under which the sequence converges to a Voronoi vertex. Then we introduce the concept of oscillations in the sequence with some examples (Section 2.1). Conditions linking inter-object visibility with oscillations are presented in Section 2.2; some special cases of convergence are treated in the Appendix.

Lemma 2.1 *Let objects S_1, S_2, S_3 and a seed point p_0 be given, and define $p_i = \rho^i(p_0)$. If $\lim_{i\to\infty} p_i$ exists, then it is a Voronoi vertex for S_1, S_2, S_3.*

Proof. The limit could be finite or infinite. If the limit is finite, let it be point p^*. Then there exists an N such that for all $i > N$ and any chosen ε independent of i, $d(p_i, p^*) < \varepsilon$. This implies that there exists a constant c (independent of ε) such that the maximum distance between any of the corresponding elements of the point sets $\phi(p^*)$ and $\phi(p_i)$ is at most $c\varepsilon$. Since p_{i+1} is simply the Voronoi vertex for the three points (treated as point objects) in $\phi(p_i)$, p^* is the Voronoi vertex of the three points on the objects it is closest to. Therefore it is a Voronoi vertex for the three objects.

The proof in the infinite case is similar. The sequence diverges to infinity in a particular direction if and only if the closest points on the objects from infinity in that direction are collinear in a perpendicular direction. □

If the limit does not exist, the sequence can oscillate with some finite period or can exhibit chaotic behavior. The latter phenomenon was not observed in our computer experiments. While we do not rule out this possibility, we proceed to present conditions under which convergence or oscillations are guaranteed. The following lemma is a well-known result from [12].

Lemma 2.2 *Given a point $p \in R^d$ and a convex set $S \in R^d$, then $p \notin S$ iff there exists a hyperplane which separates p from S.*

For a circle or circular disk (a circle together with its interior) C, let $rad(C)$ denote its radius and $center(C)$ its center.

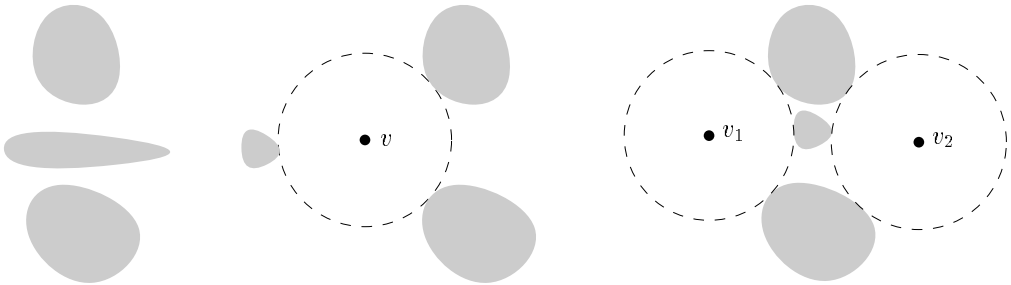

Figure 2: Three non-intersecting convex sets can define no Voronoi vertex (left), one vertex v (middle), or two vertices v_1 and v_2 (right).

This paper has been submitted to a special issue of the journal *Computational Geometry: Theory and Applications* devoted to papers selected from the sixth Canadian Conference on Computational Geometry.

1.1 Preliminaries

Let R denote the set of reals, and R^2 the plane. We include all points at infinity in R^2. The boundary of a set S is denoted as ∂S and its interior as $int(S)$, and its convex hull as $CH(\ldots)$. Unless otherwise specified, we work in R^2 and therefore a point refers to an element of R^2. The Euclidean distance between two points p and q is denoted as $d(p,q)$. Extend the notation to include distances between points and sets: the distance between point p and the set S is defined as $d(p,S) = \inf\{d(p,s) \mid s \in S\}$. It is clear that if $p \notin S$, $d(p,S)$ is achieved at a point $s \in \partial S$.

For a set S_i, the *Voronoi region* $Vor(S_i)$ for S_i is the set of points $\{p \mid \forall j : d(p,S_i) \leq d(p,S_j)\}$. For a pair of sets S_i, S_j, their bisector $bis(S_i,S_j)$ is the locus of points equidistant from both. Now let three disjoint and regular sets S_1, S_2, S_3 be given in the plane (a set is regular if it coincides with the closure of its interior. Formal definitions can be found in Kuratowski and Mostowski [17]). A *Voronoi vertex* for S_1, S_2, S_3 is a point p such that the three distances $d(p,S_i)$ are all equal. In such a case the distance $d(p,S_i)$ may be termed the Voronoi distance. The circle centered at a Voronoi vertex and radius equal to the Voronoi distance is called a *Voronoi circle*. If the sets are convex and a Voronoi vertex exists, the Voronoi circle intersects them in exactly one point.

While Voronoi regions and bisectors always exist and are uniquely defined, the set of Voronoi vertices for three sets could be empty. On the other hand, more than one Voronoi vertex could exist for three given sets. However, for three (possibly non-intersecting) convex sets, at most two Voronoi vertices can exist (See Fig. 2). Whenever our attention is focused on one Voronoi vertex for three sets, we refer to it as $Vor(S_1,S_2,S_3)$.

We define two functions ϕ and γ, and their composition ρ as follows. The function

$$\phi : R^2 \rightarrow S_1 \times S_2 \times S_3$$

maps a point $p \in R^2$ to the respective closest points in the three sets from p, i.e., $\phi(p) = (s_1, s_2, s_3)$ where $d(p,S_i)$ is achieved at $s_i \in \partial S_i$. The function

$$\gamma : S_1 \times S_2 \times S_3 \rightarrow R^2$$

(finite) period > 1. More complicated outcomes such as infinite period oscillations and chaos are possible and are in fact quite the norm in complex systems in nature. In this paper we will use the intuitive terms of convergence and oscillations (instead of orbits of particular period); we never observed the phenomena of chaos in our system.

After describing notation and defining the problem and the iteration sequence formally in Section 1.1, we study the behavior of this sequence in Section 2 with the goal of obtaining geometric conditions on the placement of the objects and the choice of the seed point under which convergence (or oscillation) occurs.

We do not yet have conditions that are both necessary and sufficient for convergence; regions of convergence appear difficult to compute in general. However, in Section 3 we show that the region of convergence for a Voronoi vertex defined by convex objects (or objects that can be decomposed into convex components) is of non-zero (two dimensional) measure. Thus, a randomly chosen seed point (assuming bounded space to make the random choices) will eventually hit the region of convergence with high probability. This proves probabilistic completeness of our algorithm: the probability that we have not detected the Voronoi vertex decreases exponentially with the number of random seed point selections.

In tracking down Voronoi vertices from randomly chosen seed points, we also build an approximation to the Voronoi edges connecting up Voronoi vertices to maintain the topology of the Voronoi diagram. This is motivated by applications in robot motion planning: the basic problem is to determine a collision-free motion from a start configuration to a goal configuration for a robot moving amidst but avoiding a set of obstacles. A well-known, natural, and intuitively appealing approach is to try and plan a motion that keeps the robot as far away from the obstacles as possible; this approach is often referred to as *retraction motion planning* [22, 5, 19, 6, 26] (we refer to Latombe [18] for an overview of other existing approaches). The Voronoi diagram is central to the idea of retraction motion planning. Given the Voronoi diagram in the planar configuration space of a robot, retraction motion planning works by retracting the start and goal configurations onto the diagram and then connecting them via edges and vertices of the diagram [21, 14, 1]. Whenever there exists a path, this approach is guaranteed to find one which maximizes the clearance of the robot.

In Section 4, we discuss the implementation of a planar robot path planner based on this iterative idea for computing Voronoi vertices. Since we only require answers to nearest point queries and do not assume an exact shape description for the obstacles, the method seems better suited for real robotics applications than traditional Voronoi-based approaches. This approach is similar to the sensor-based planning in the robotics literature [11, 20, 9]; the environment is partially or fully unknown and the robot incrementally learns it by using its sensor.

Choset and Burdick have recently investigated a similar idea [8] in a method for tracing the Voronoi diagram along with an account of the analytical properties of the Euclidian distance function between a point and a convex set. They show that the distance function is non-smooth, but describe how a generalized gradient can be defined. Based on this, they propose a method of tracing the Voronoi diagram by following bisectors, keeping track of the distance to the currently nearest obstacles, and proceeding recursively when a Voronoi vertex is encountered. Our incremental construction of the diagram is somewhat similar, although there are some substantial differences. First, the size of the steps that the robot takes while following bisectors is adaptive in our method and not fixed to some small positive number. Secondly, we try to infer the topology of the diagram, and to delay collision detection (edge evaluation) up to the path-search phase. By tracing the entire diagram in small steps as Choset and Burdick do, a large number of distance measurements are required to construct parts of the diagram which may not be eventually needed for finding a path.

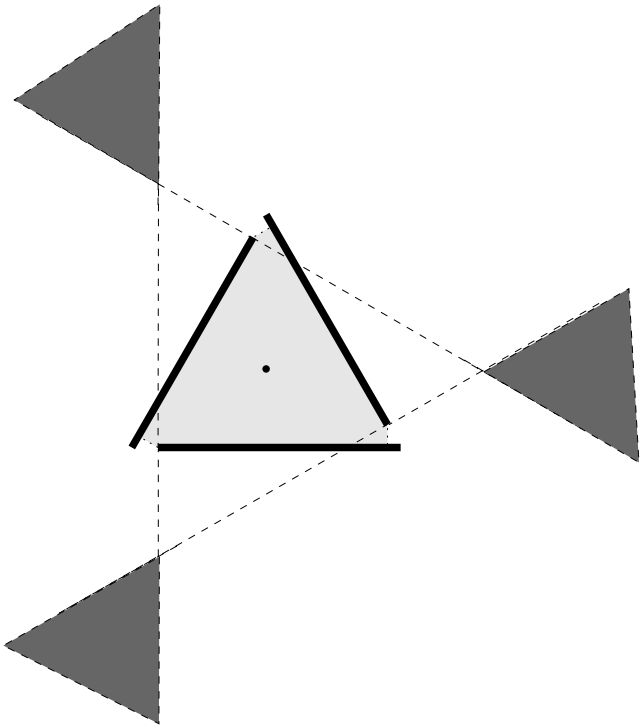

Figure 1: Three line segments, shown thick, with their Voronoi vertex marked and a subset of its region of convergence shown lightly shaded. The heavily shaded regions, intersection of half spaces bounded by perpendiculars to the line segments, may be termed "regions of oscillation": the reader may verify that any seed point from them immediately locks into oscillations of period 3. These are closed regions. Other regions appear difficult to classify as converging or oscillating.

from the three points s_1, s_2, s_3 (in other words, q is the Voronoi vertex for the three point objects s_i). Now reiterate beginning from q. Clearly, if p was (accidently) chosen as the Voronoi vertex to begin with, $q = p$ and any further iterations remain at the vertex; the Voronoi vertex is a "stable point" under this iterative scheme. This directly follows from the definition of the Voronoi vertex.

However, what does not follow directly from the definition of a Voronoi vertex, and which we will prove in this paper, is that the Voronoi vertex is not merely a stable *point* under the iterative procedure but is also a stable *attractor*, that is *every Voronoi vertex is associated with a* region of convergence *such that a seed point taken from anywhere inside this region converges to the vertex under the iterative scheme.* See Figure 1.

Contrary to optimistic expectations, however, the region of convergence does not always include the entire plane. For seed points taken from outside the region of convergence, the sequence of points can cycle between some finite set of points none of which are the Voronoi vertex (see Figure 1). Also, when three objects do not define a Voronoi vertex (when one object "hides" the second from the third), the sequence of points obtained from any seed point oscillate with finite period.

Dynamical Systems and Chaos Theory have studied such problems which involve mappings from $\Re^d \mapsto \Re^d$ [4, 25]. The simplest outcomes of such mappings are the two that we observed above: convergence and cycles of finite period and the technical terms for these phenomena are *orbits*. Convergence to a point is referred as an *orbit of period 1* while oscillations are *orbits of*

Hunting Voronoi Vertices*

Vincenzo Ferrucci† Mark Overmars‡ Anil Rao‡ Jules Vleugels‡

Abstract

Given three objects in the plane, a Voronoi vertex is a point that is equidistant simultaneously from each. In this paper, we consider the problem of computing Voronoi vertices for planar objects of fixed but possibly unknown shape; we only require the ability to query the closest point on an object from a given point. Our technique is simple, robust and iterative in nature: beginning from some initial (seed) point, it computes a sequence of points based on intermediate closest-point queries. This technique is observed to either converge to a Voronoi vertex or oscillate with some finite period. We study geometric conditions on shape/placement of the objects and choice of the initial point that guarantee convergence or oscillation. We show that our technique is probabilistically complete; selecting seed points at random will eventually guarantee convergence to a Voronoi vertex, if one exists.

Our motivation for seeking Voronoi vertices comes from robot motion planning: a Voronoi vertex is a natural haven for mobile robots avoiding obstacles. We conclude by briefly describing an efficient implementation of a retraction-like path planner for a planar robot based on our iterative strategy for seeking Voronoi vertices.

1 Introduction

A familiar notion in computational geometry is that of the *Voronoi diagram* [2, 16] which can informally be defined as follows. Given a set of sites, the *Voronoi region* of a site is the set of points closer (under the Euclidean metric) to that site than to any other. The Voronoi diagram is the network formed by the boundaries of the individual Voronoi regions. In the plane, this network is one-dimensional and is made up of *Voronoi edges* and *Voronoi vertices*; Voronoi vertices are points equidistant from three nearest sites while Voronoi edges are subsets of the locus of points equidistant from two sites. If the space is bounded, the Voronoi diagram is connected and preserves the connectivity of the space.

The problem of computing the Voronoi diagram for a given set of sites is a familiar one in the field of computational geometry and has been extensively studied for polygonal and simple curved sites [3, 15, 16, 29]. However, not much is known in regard to arbitrarily curved objects. Towards this end, we propose a new technique, the novelty of which is that the (possibly complex) shapes of the objects are not required exactly; instead we only need the ability to be able to answer queries of the form: "What is the nearest point from point p on object S?"

Let three disjoint regular sets S_1, S_2, S_3 be given in the plane; choose a (seed) point p. Determine three points $s_i \in S_i$ which achieve minimum distance from p. Next, compute the point q equidistant

*This research was partially supported by ESPRIT Basic Research Action No. 6546 (project PROMotion) and by the Netherlands Organization for Scientific Research (NWO).

†Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Via Buonarroti 12, 00185 Roma, Italy.

‡Department of Computer Science, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, the Netherlands.