1. Suppose that in the line-sweep algorithm for the line-segment intersection problem, some lines are allowed to be vertical. Explain how you can handle a "vertical segment" event.

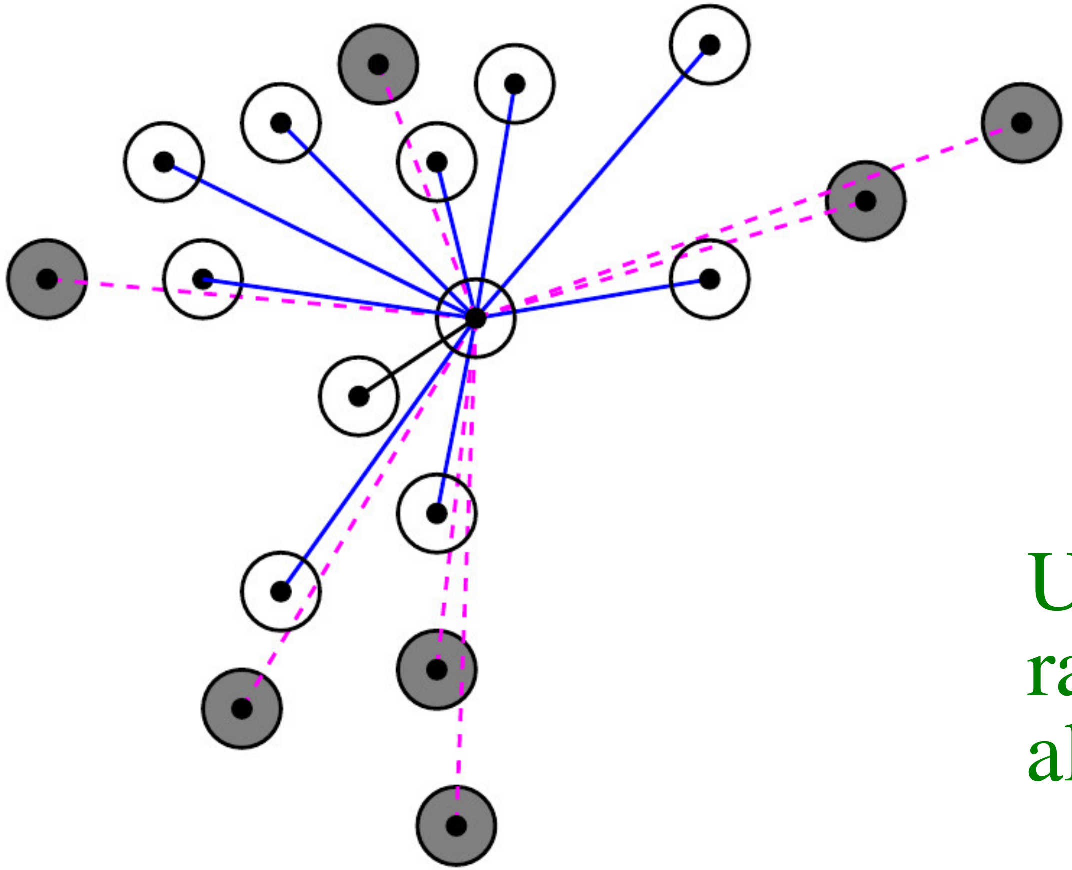
(a) The event queue Q contains a single event for each vertical segment L_i. When this event occurs, we delete the event from Q. Suppose that the sweep line information S is maintained as a height-balanced binary search tree with top-to-bottom ordering of active segments. We locate the insertion position of the upper end point of L_i in S. From this position, we keep on looking at successors in S, until the successor lies below the bottom end point of L_i. Let all these successors be $L_{j_1}, L_{j_2}, \ldots, L_{j_r}$. We report the intersection of L_i with each of $L_{j_1}, L_{j_2}, \ldots, L_{j_r}$.

(b) Suppose that there are h_1 intersection points among non-vertical segments, and there are h_2 intersection points involving vertical segments. The effort spent to identify the former h_1 intersection points remains $O((n+h_1)\log n)$ as in the original algorithm. For a vertical segment L_i, the determination of r intersection points (with $L_{j_1}, L_{j_2}, \ldots, L_{j_r}$) takes $O(r \log n)$ time, since the size of S is always $\leqslant n$. So the total effort associated with handling all "Vertical Segment" events is $O(h_2 \log n)$. Consequently, the total running time is $O((n+h_1+h_2)\log n)$, that is, $O((n+h)\log n)$.

Note: All the r intersections with L_i can be found in $O(r + \log n)$ time.

2.

There are n cell-phone towers in the plane. The towers are located at the points (x_i, y_i) for $i = 1, 2, 3, \ldots, n$. Around each tower, there is an interference zone of a fixed radius r (the same for all the towers). Assume that the interference zones do not overlap with each other. Two towers can communicate without interference if the line segment joining them does not intersect with the interference zone of any other tower. Your task is to determine all the towers with which the first tower (located at (x_1, y_1)) can communicate without interference. The following figure gives an example. Propose an $O(n \log n)$-time algorithm to solve this problem. Clearly mention the data structures that your algorithm uses. Also justify that your algorithm actually achieves the given running time.



Use a ray-sweep algorithm.

(Sketch) Use a ray-sweep algorithm. The ray emanates from the position (x_1, y_1) of the first tower.

Active towers: Those for which the interference zones intersect with the ray. These are kept sorted with respect to their distances from (x_1, y_1).

Events:

- **Enter circle:** Insert the tower in the list of active towers.
- **Leave circle:** Delete the tower from the list of active towers.
- **Center of circle:** Report the tower as supporting interference-free communication if and only if it is the closest active tower from (x_1, y_1).

The list of active towers should support arbitrary insertion and deletion, and should therefore be implemented as a height-balanced tree.

The event queue may be implemented as a heap. There are $3(n-1)$ events. All can be initially put in the heap, and a makeheap operation is performed on the heap. The heap ordering is with respect to the angles with the horizontal.

The event queue may be a sorted array as well.
Running time is O(n log n).

3. Use a circle-sweep algorithm to solve Exercise 2. Here, a circle centered at Tower 1 grows from radius 0 to radius ∞.

The shadow arcs cast by the towers already encountered by the sweeping circle are maintained as a sorted list of disjoint subintervals of $[0, 2\pi)$. We only handle "Meet Tower" events. During such an event, we check whether the tower lies inside a shadow arc, or outside all such arcs. The decision of interference-free communication is made accordingly.

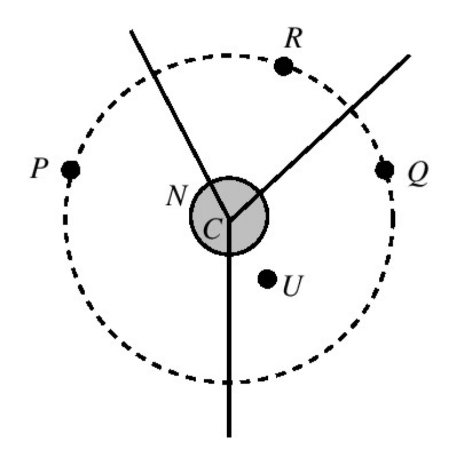
Moreover, this new tower adds to a shadow arc to the list. This new interval may need to be merged with existing intervals. Since the arc sizes decrease monotonically with distance from the center of the circle, merging (if needed at all) is done with only the predecessor and/or the successor intervals.

We may maintain the shadow list as a height-balanced BST. There are at most n - 1 arcs, so the height of the tree is O(log n). Searching takes O(log n) time. Merging requires finding only one successor and only one predecessor, followed possibly by a constant number of insert and delete operations in the tree. So each merging can be completed in O(log n) time.

Since there are n - 1 events, the total running time is O(n log n).

4. Let S be a set of n sites. Prove the following facts about Vor(S).

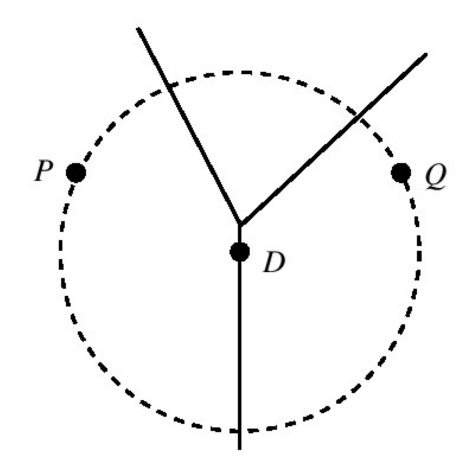
(a) Let C be a vertex of Vor(S). Then C is equidistant from three sites P,Q,R. Prove that the circle with center at C and passing through P,Q,R contains no other sites. Conversely, if C is the center of a circle containing three sites P,Q,R but no other sites (on or inside), then C is a vertex in Vor(S).



(a) Suppose that U is a point in S lying in the closed disk defined by the circle of the question (see Figure 134(a)). If U lies on the boundary of the disk, then the four points P, Q, R, U lie on the same circle, violating the assumption that the sites in S are in general position. So U must lie in the interior of the disk. But then U is closer to C than each of P, Q, R. Therefore there exists a neighborhood N of C (may be a disk) containing C in the interior such that each point of N is closer to U than to each of P, Q, R. This contradicts the fact that the Voronoi cells of P, Q, R intersect at the center C.

For proving the converse, note that C is equidistant from P, Q, R but is more distant to any other site. Therefore a neighborhood N of C has the same property, that is, each point of N is closer to one or more of P, Q, R than to any other site.

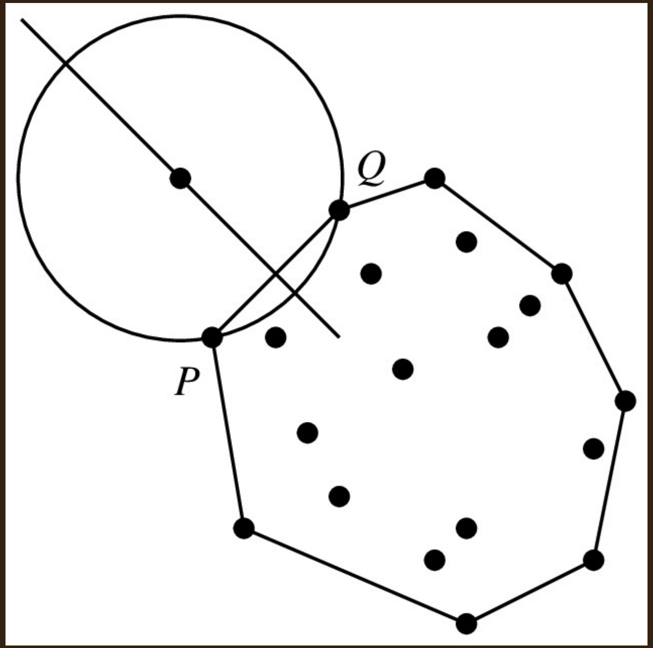
(b) Let P,Q be two sites. Prove that the Voronoi cells of P and Q share an edge if and only if there exists a circle passing through P and Q and containing no other site.



(b) Let there be a circle through P and Q containing no other points of S on or inside it (see Figure 134(b)). The center D of this circle lies on the perpendicular bisector of PQ. The point D is equidistant from P and Q but is more distant to any other point of S. Therefore D belongs to the Voronoi diagram $\mathrm{Vor}(S)$. If D is a vertex in $\mathrm{Vor}(S)$, then by Part (a) the given circle contains a third point R on it, a contradiction. Thus D must belong to an edge of $\mathrm{Vor}(S)$, that is, a part of the perpendicular bisector of PQ is in $\mathrm{Vor}(S)$, that is, the Voronoi cells of P and Q share an edge.

Conversely, let the Voronoi cells of P and Q share an edge. Choose any point D in the interior of this edge. If the circle contains a site R inside it, then a neighborhood of D containing D in its interior is closer to R than to P and Q, so D does not belong to $\mathrm{Vor}(S)$. If there is a third site R on the circle, then by Part (a), D is a vertex of $\mathrm{Vor}(S)$, that is, not in the interior of an edge of $\mathrm{Vor}(S)$.

(c) Prove that two sites P and Q share a semi-infinite edge if and only if PQ is an edge of CH(S).



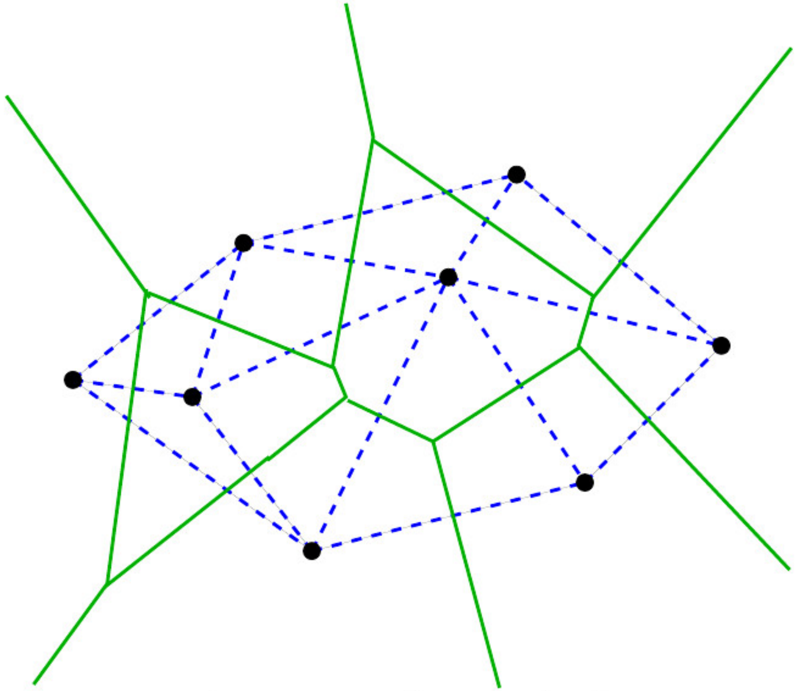
(c) First suppose that PQ is an edge of $\mathrm{CH}(S)$. Then, all other sites are on one side of PQ. Consider a circle passing through P and Q and having center on the other side and on the perpendicular bisector of PQ. Since no three points are collinear, the segment PQ cannot contain another site. This means that when the center of the circle is sufficiently far away from PQ, it does not contain any site other than P and Q. As the center moves farther away, the circle becomes flatter and flatter, and will never contain other sites. Therefore all these centers belong to the boundary between $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ (see Part (c) of Figure 134).

Conversely, suppose that $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ share a semi-infinite edge. A circle passing through P and Q and having its center on any point on the semi-infinite edge does not contain any other site. Since the edge is semi-infinite, we can take the center arbitrarily far away from PQ. In the limit, the circle approaches the line PQ. This implies that all sites are on one side of PQ, so PQ is an edge of $\mathrm{CH}(S)$. Indeed, if there is a site on the other side of PQ, the circle would eventually become flat enough to include that site, contradicting the hypothesis that $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ share a semi-infinite edge.

5. Prove that any algorithm for computing the Voronoi diagram of n sites must take O(n log n) time in the worst case.

Use reduction CH $\leqslant$ VOR. Pass S for CH to S for VOR (O(n) time). Given Vor(S), CH(S) can be identified from the semi-infinite edges. Since the size of Vor(S) is linear in n, this can be done in O(n) time. Therefore an o(n log n) algorithm for Vor(S) gives a o(n log n)-time algorithm for CH(S).

6. [Delaunay triangulation] Let S be a set of sites in general position. Join P and Q if and only if VCell(P) and VCell(Q) share an edge. Prove that this construct produces a triangulation of S (or CH(S)).

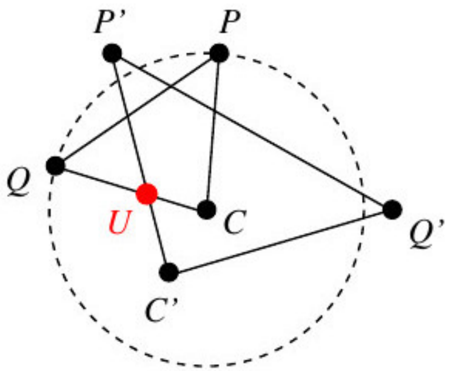

Delaunay triangulation

Since the points in S are in general position, each vertex of $\mathrm{Vor}(S)$ belongs to exactly three Voronoi cells, implying that the faces of the given construction are triangles. It therefore suffices to prove that the edges of the construction do not intersect each other in their interiors.

Let P, Q, P', Q' be four points of S such that P, Q share an edge in $\mathrm{Vor}(S)$, and P', Q' share an edge in $\mathrm{Vor}(S)$. Assume that the edges PQ and $P'Q'$ intersect. By Exercise 5.240(b), there exists a circle with center C such that P, Q lie on the circle but no other sites (including P', Q') are enclosed by the circle. It then follows that the segment CP lies completely inside the Voronoi cell of P, whereas the segment CQ lies completely inside the Voronoi cell of Q. Let T denote the triangle CPQ.

Likewise, there is a circle with center C' such that P', Q' lie on the circle, but P, Q lie outside the circle. Let T' denote the triangle $C'P'Q'$. We have $C'P'$ (resp. $C'Q'$) lying entirely inside the Voronoi cell of P' (resp. Q').

Figure 135: Figure explaining the contradiction in the proof of Exercise 5.243



Since the segments PQ and $P'Q'$ intersect, the triangles T and T' must also intersect (see Figure 135). That is, one of the edges CP and CQ must intersect with one of the edges $C'P'$ and $C'Q'$ (in Figure 135 CQ intersects with $C'P'$). This intersection point U belongs to two different Voronoi cells, a contradiction.