COT5520: COMPUTATIONAL GEOMETRY

Homework # 4

Due date: Nov 21, 2006, Tuesday (beginning of the class)

Your solutions should be concise, but complete, and typed (or handwritten clearly). Feel free to consult textbooks, journal and conference papers and also each other, but write the solutions yourself and cite your sources. Answer **only five** of the following six questions. Each problem is worth 20 pts.

1. Let P be a finite set of points in the plane and H the dual set of lines. Translate the following statements about P to statements about H.
   - (i) No three points of P lie on a common line.
   - (ii) The point $p \in P$ is a vertex of the convex hull.
   - (iii) P has a subset of k points in convex position.
   - (iv) P is contained in a strip bounded by two parallel lines at unit distance from each other.

2. A k-*coloring* of a line arrangement is a map χ from the set of faces to $\{1, 2, \ldots, k\}$ such that $\chi(f) \neq \chi(g)$ if $f \neq g$ share a common edge.

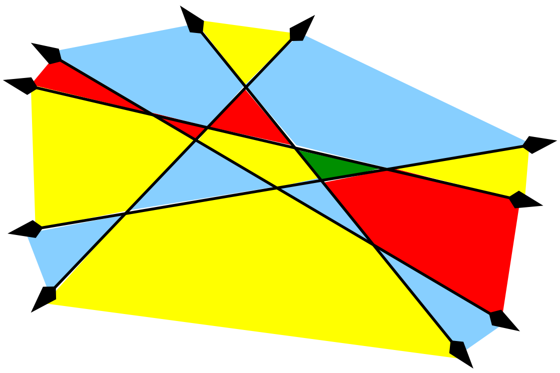

Figure 1: A 4-coloring of an arrangement of 5 lines.

   - (i) What is the smallest k such that every line arrangement has a k-coloring?
   - (ii) Which line arrangements are 2-colorable?
   - (iii) Draw a 2-colorable line arrangement for which the ratio of the number of faces of one color over the number of faces of the other color is as large as you can manage. What ratio do you get?

3. Let H be a set of n lines in general position. Consider the graph whose nodes are the vertices and whose arcs are the bounded edges in the arrangement $\mathcal{A}(H)$. Describe an algorithm that takes time $O(n^2)$ and space $O(n)$ to compute the x-monotone path with the largest number of arcs.

4. Let S be a set of n line segments in the plane. A *stabber* is a line h that intersects all segments of S.

   (i) Give an $O(n^2)$ algorithm to decide whether or not S has a stabber.

   (ii) Now assume that all line segments are vertical. Give a randomized algorithm with $O(n)$ expected running time that decides whether S has a stabber.

5. Consider variants of the metric TSP problem in which the objective is to find a simple path containing all the vertices of the graph. Three different problems arise, depending on the number k of endpoints of the path that are specified.

   - Design a 3/2 factor approximation algorithm for $k = 0$ or $k = 1$.
   - Design a 5/3 factor approximation algorithm for $k = 2$.

6. Study the rotational plane sweep algorithm for computing visibility graphs (described in the textbook and in class). For each vertex, we sort all the other points around it. This leads to $O(n^2 \log n)$ total time for all sortings. Show that this can be improved to $O(n^2)$ time. (Hint: You can use the quadratic time algorithm described in class for computing arrangements.)