Homework 9

Due by the start of class on Thursday, May 7. (Submissions will be through Gradescope.) Late homeworks are not accepted (unless an extension has been prearranged) so please turn in whatever you have completed by the due date. Unless otherwise specified, you may assume that all inputs are given in *general position.*

Problem 1. Let us consider a motion planning problem in the plane involving a set of rectangular obstacles and a translating "L"-shaped robot (see Fig. 1(a)). The robot $\mathcal{R}$ is an "L"-shaped polygon as shown in Fig. 1(b). Its reference point is its lower-left corner. The obstacles consist of a collection of non-intersecting rectangles $\{O_1, \ldots, O_n\}$, where O_i has lower-left corner p_i and horizontal length ℓ_i and vertical height h_i (see Fig. 1(c)).

You are given a start point s and target point t. Assuming the robot's reference point starts at s, is there an obstacle-avoiding motion that terminates with the endpoint at t? In this problem will explore a solution to this problem.

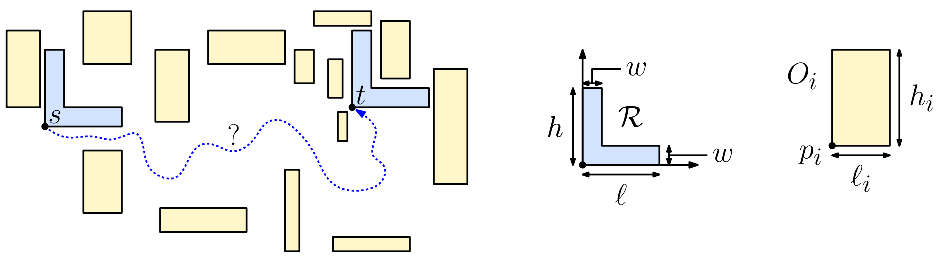


Figure 1: L-shaped robot motion planning.

(a) Given an obstacle O_i, provide a clear description and a picture of the associated C-obstacle (its shape, side lengths, and location).

(b) Given the starting and target points s and t, sketch an algorithm for determining whether there is a collision-free translational motion of the robot with its reference point starting at s and ending at t. A high-level sketch of the algorithm is sufficient. You may express the running time in terms of n and the total complexity m of the union of configuration obstacles. (See part (c).)

(**Hint:** Plane sweep and trapezoidal maps may be helpful.)

(c) What is the worst-case combinatorial complexity of the boundary of the union of all the C-obstacles? (Note that the boundary of the union need not be simply connected.) You may select any values you like for the robot's dimensions (h, ℓ, and w), and you may place the n rectangles anywhere you like. Justify your answer. (**Hint:** It is not linear in n.)

Problem 2. Recall that a collection of pseudodisks is a set of convex bodies $\{o_1, \ldots, o_n\}$ in $\mathbb{R}^2$ such that for all $i \neq j$, $o_i \backslash o_j$ and $o_j \backslash o_i$ are both connected sets. In class we gave an $O(n)$

upper bound on the complexity of the union of a collection of polygonal pseudodisks with n vertices in total.)The objective of this problem is to get more precise bounds.

(a) Assume that the union boundary contains m original vertices of the polygons. Show that the complexity of the union boundary (which includes both these vertices and the points where the boundaries intersect) is at most $2n - m$. (**Note:** Clearly, $m \geq 3$, and so this implies an immediate upper bound of $2n - 3$.)

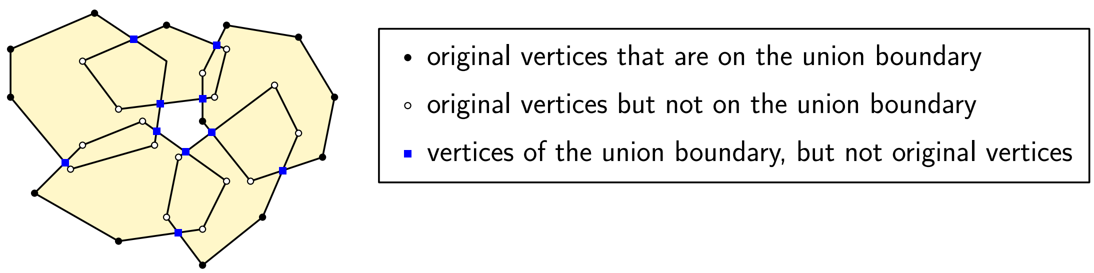


Figure 2: Complexity of the union of a collection of pseudodisks. (Black vertices are counted by m. The combination of black and white vertices are counted by n. The combination of black and blue points are counted in the union boundary.)

(b) Prove a lower bound of at least $2n - 6$ on the worst-case complexity of the number of vertices on the union boundary by constructing an example that has this complexity. Formally, if $f(n)$ denotes the maximum number of boundary vertices that can arise in the union of any collection of polygons having n total vertices, then give an example of a collection of showing that $f(n) \geq 2n - 6$. (Your example should be sufficiently generic that it is clear that it applies for arbitrarily large values of n. We will give partial credit for an answer of the form $2n - c$, where $c > 6$ is a constant.)

Problem 3. In this problem, we will consider a geometric problem analogous to finding the greatest common divisor of two numbers. You are given two convex polygons, P and Q, which we assume are the results of Minkowski sums of two convex polygons, denoted P' and Q', with a common convex polygon R, that is, $P = P' \oplus R$ and $Q = Q' \oplus R$ (see Fig. 3).

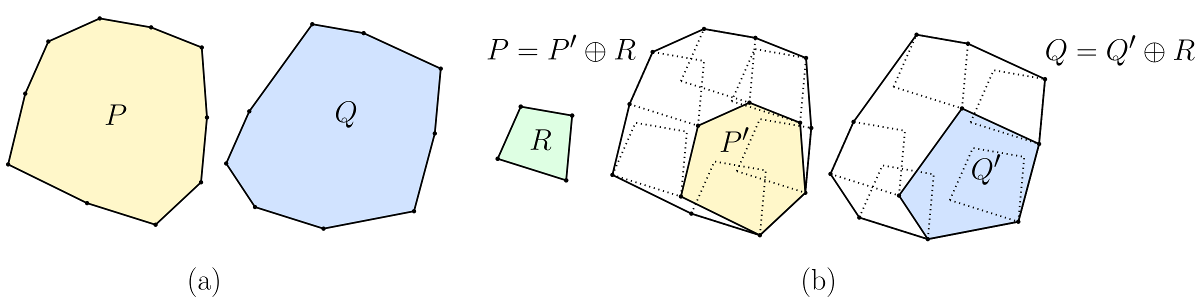


Figure 3: Maximal common summands.

Such a polygon R is called a *common summand* of P and Q. (Observe that there is always a trivial solution, where $P' = P$, $Q' = Q$ and $R = \{O\}$, where O denotes the origin, but this is not allowed since R is not a proper polygon.)

(a) Prove that if R is a common summand of P and Q and there exist convex polygons R_1 and R_2 such that $R = R_1 \oplus R_2$, then both R_1 and R_2 are common summands of P and Q. **Hint:** It may be useful to use the fact that Minkowski addition is both commutative and associative.

(b) Prove that for any convex polygon R and any α, where $0 \leq \alpha \leq R$, $R = \alpha R + (1-\alpha)R$. (This implies that if P and Q have a common summand, then they have an infinite number of common summands.)

We say that a common summand R of P and Q is *maximal* if for any convex polygon $R' \neq \{O\}$, $R \oplus R'$ is not a common summand of P and Q.

(c) Present an efficient algorithm which, given P and Q, determines whether they have a common summand, and if so, computes a maximal common summand. It will simplify the algorithm to assume the *general-position case*, where P', Q', and R are in general position, implying that there are no parallel edges between any two of them. (The general case will be considered in the challenge problem.)

Justify your algorithm's correctness and derive its running time. Aim for a running time of $O(n+m)$, where n and m are the number of sizes of P and Q, respectively.

Hint: The following fact may be useful. Consider any set of vectors $\{v_1, \ldots, v_n\}$ such that $\sum_{i=1}^{n} v_i = O$ (the zero vector). Such a set of vectors, if joined tail to head in angular order, define the edges of a convex polygon (see Fig. 4).

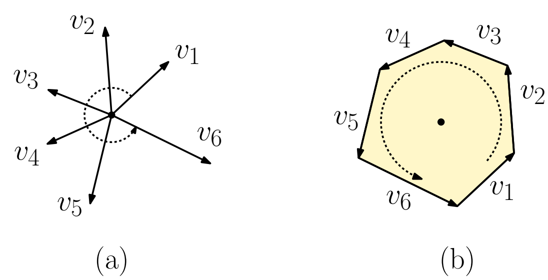


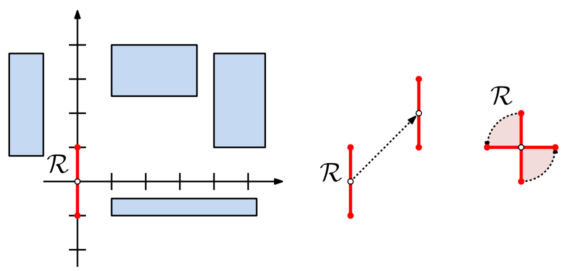
Figure 4: Maximal common summands.

Problem 4. (Optional–Ungraded) In this problem, we will be planning the motion of a line-segment robot $\mathcal{R}$ in the plane amidst a collection of obstacles consisting of n disjoint obstacles $\mathcal{P} = \{P_1, \ldots, P_n\}$. Each obstacle is an axis-parallel rectangle. In particular $P_i = [x_i^-, x_i^+] \times [y_i^-, y_i^+]$. The robot is 2-units long, and its reference position oriented vertically with its midpoint at the origin (see Fig. 5(a)). The robot is restricted to two types of motion:

Translation: It can translate through a vector $t = (t_x, t_y)$, moving its reference point from its current position p to $p + t$.

Rotation: The robot can rotate about its reference point by either $+90°$ (that is, counter-clockwise) or $-90°$ (that is, clockwise). When a rotation is performed, the entire circular arc swept out by the any point on the segment must be free of any obstacles. (Think of the segment as spinning in the plane—not picking it up, rotating it, and putting it down.)

Therefore, the robot's *configuration* consists of a point (x, y) where its center point is located and its orientation $o \in \{V, H\}$, where "V" indicates that the robot is parallel with the y-axis and "H" indicates that it is parallel with the x-axis.



(a) (b)

Figure 5: Motion planning for a rotating/translating segment.

A motion plan consists of a sequence of translations and rotations. (Note that the robot either translates or rotates. It cannot translate while rotating.)

(a) For each of the possible orientations of the robot ("H" or "V"), describe the shape of the corresponding collision obstacle $\mathcal{C}_{\mathcal{R}}(P_i)$ (in terms of the parameters $x_i^-, x_i^+, y_i^-, y_i^+$).

(b) For each of the possible orientations of the robot ("H" or "V"), describe the shape of the corresponding collision obstacle $\mathcal{C}_{\mathcal{R}}(P_i)$ in the cases of $+90°$ (counterclockwise) and $-90°$ (clockwise) rotation (in terms of the parameters $x_i^-, x_i^+, y_i^-, y_i^+$). There are four shapes in all and their boundaries will involve circular arcs.

(c) Given your answers to (a) and (b), present an algorithm to determine whether there exists a motion plan from an arbitrary starting placement configuration $s = (x_s, y_s, o_s)$ to a given target $t = (x_t, y_t, o_t)$. You may assume that both s and t are collision-free.

Hint: I'm looking for a high level description of how to combine reachability among the various collision obstacles. Efficiency is not a huge consideration, but your solution should run in polynomial time in n.

Note: Challenge problems are not graded as part of the homework. The grades are recorded separately. After final grades have been computed, I may "bump-up" a grade that is slightly below a cutoff threshold based on these extra points. (But there is no formal rule for this.)

Challenge Problem: Solve Problem 3(c) in the general case, where no assumptions are made about general position. (This implies that P and Q may have parallel edges of different lengths. **Hint:** I do not know how to solve this problem efficiently for general R. It suffices to solve the problem under the assumption that P and Q share only a constant number of parallel edges.