Introduction to Motion Planning

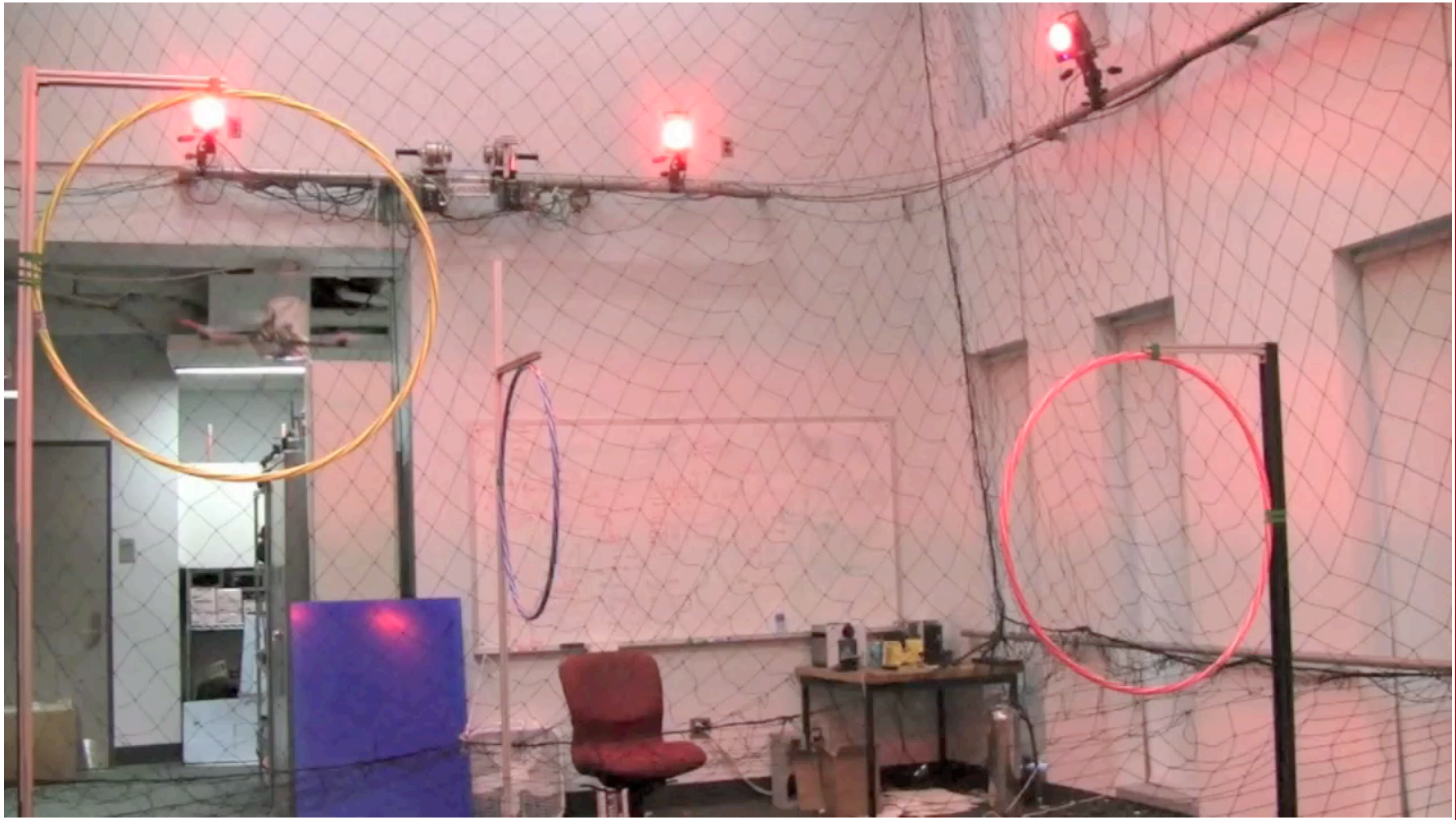
Lesson 9.1

Central Question

How to plan paths/trajectories/motion in an known environment with obstacles?

The motion planning problem

Path planning – geometry
Trajectory planning – time parameterized
Motion planning – forces, actuators, constraints



D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," *Proc. IEEE International Conference on Robotics and Automation*. Shanghai, China, May, 2011.

Penn Engineering

The Basic Problem

- Euclidean world, R^N, N=2 or 3
- Obstacles O_1, O_2, … O_p, all closet subsets of R^N
- Rigid Body (robot) A

Reference

- Planning algorithms, Lavalle (Section 4.3), http://lavalle.pl/planning/

Rigid Body *A*

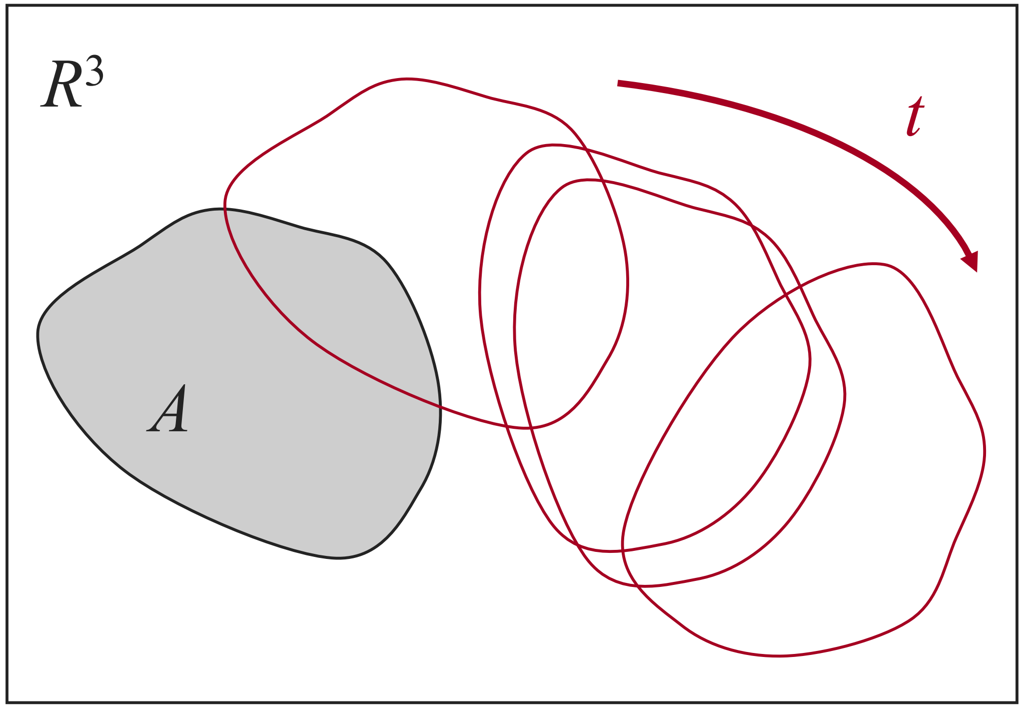
Body

$$A \subset R^3$$

Rigid Body Displacement

Map

$$g : A \to R^3$$



Rigid Body Motion

Continuous family of maps

$$g(t) : A \to R^3$$

Each displacement is a new pose (position + orientation)

The Basic Problem

- Euclidean world, R^N
- Obstacles O_1, O_2, … O_p, all closet subsets of R^N
- Robot (Rigid Body) A *or collection of rigid bodies*
- Given initial and final position/orientation (pose) of A, find a continuous (and legal) sequence of poses

Given

$$g(t_I): A \to R^3$$

$$g(t_F): A \to R^3$$

Find a safe, continuous family of maps

$$g(t): A \to R^3$$

Configuration Space (C-space)

The motion planning problem is best formulated in configuration space (denote by C)

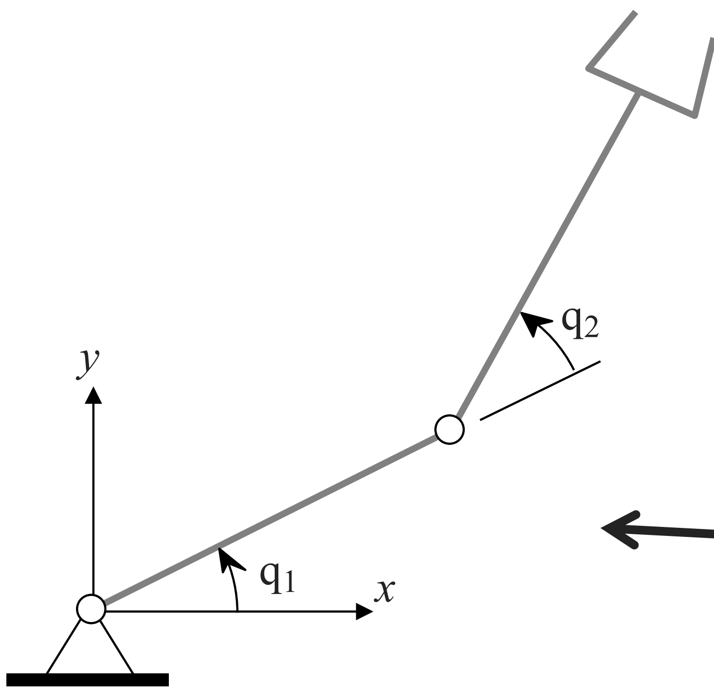
- $C \neq R^N$
- C is the set of all position/orientations
  - all possible maps $g : A \rightarrow R^3$

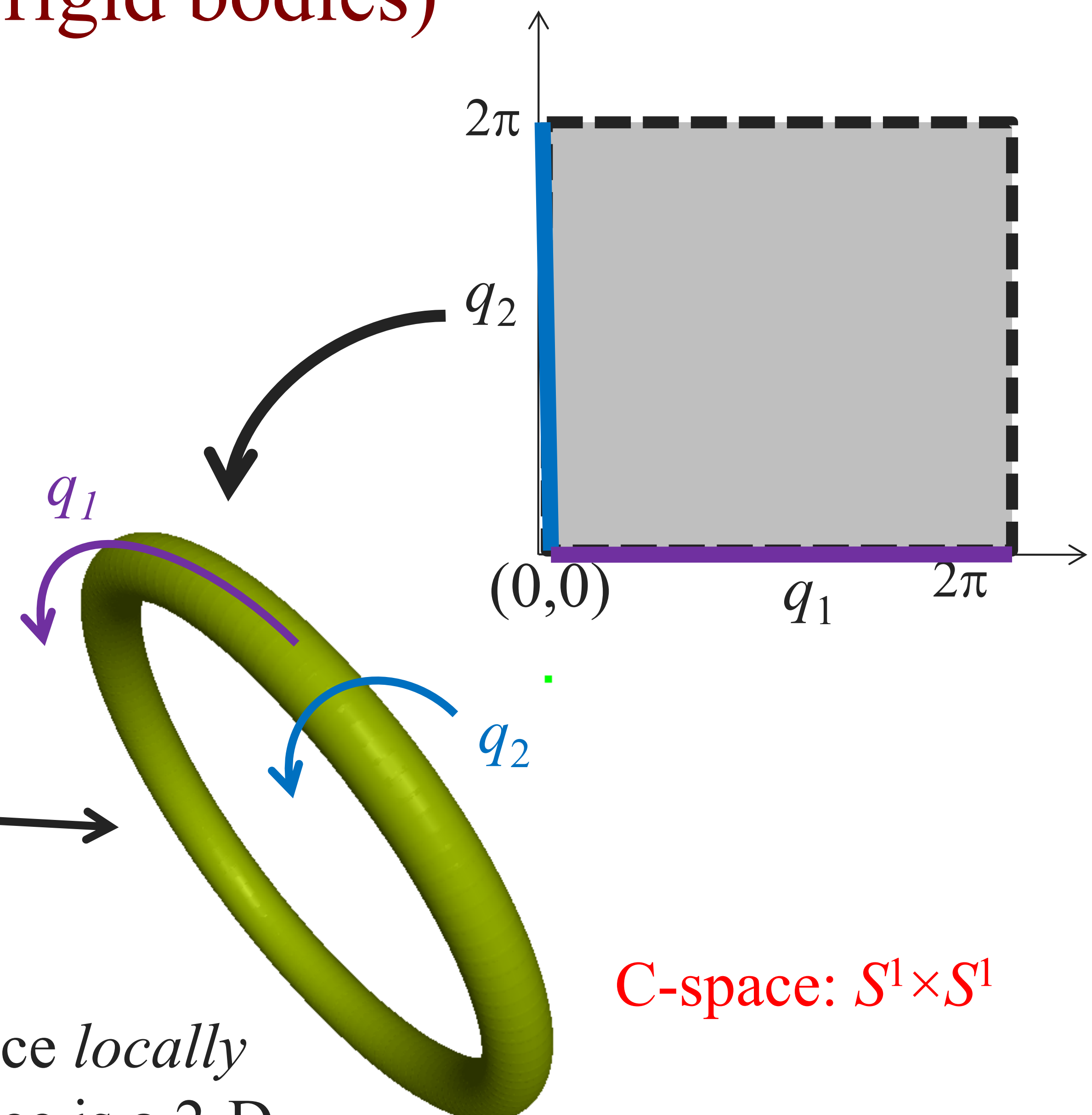
Examples of configuration space

- Point robot in N-dimensional space R^N Ans: R^N
- Rectangular robot in R^2 Ans: $SE(2)$
- Fixed robot arm with 2 revolute joints

Penn Engineering

C-space for a 2-R arm
(2 rigid bodies)





C-space: $S^1 \times S^1$

The configuration space *locally* looks like R^2, and hence is a 2-D *manifold*

Configuration Space (C-space)

The motion planning problem is best formulated in configuration space (denote by C)

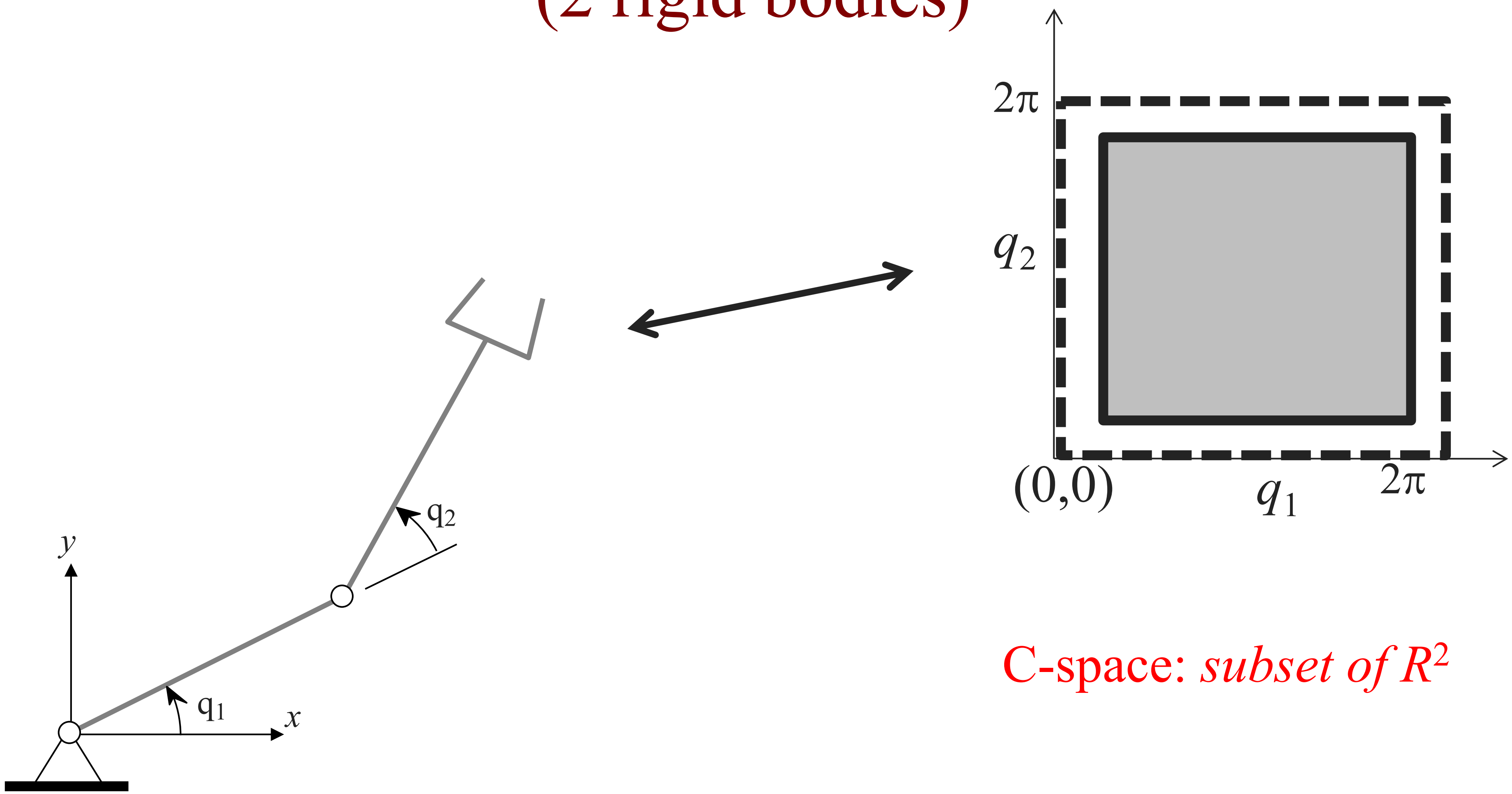
- $C \neq R^N$
- C is the set of all position/orientations
  - all possible maps $g : A \rightarrow R^3$

Examples of configuration space

- Point robot in N-dimensional space R^N Ans: R^N
- Rectangular robot in R^2 Ans: $SE(2)$
- Fixed robot arm with 2 revolute joints Ans: $S^1 \times S^1$
- Fixed robot arm with 2 revolute joints each with limits

Penn Engineering

C-space for a 2-R arm with two rigid bodies (2 rigid bodies)

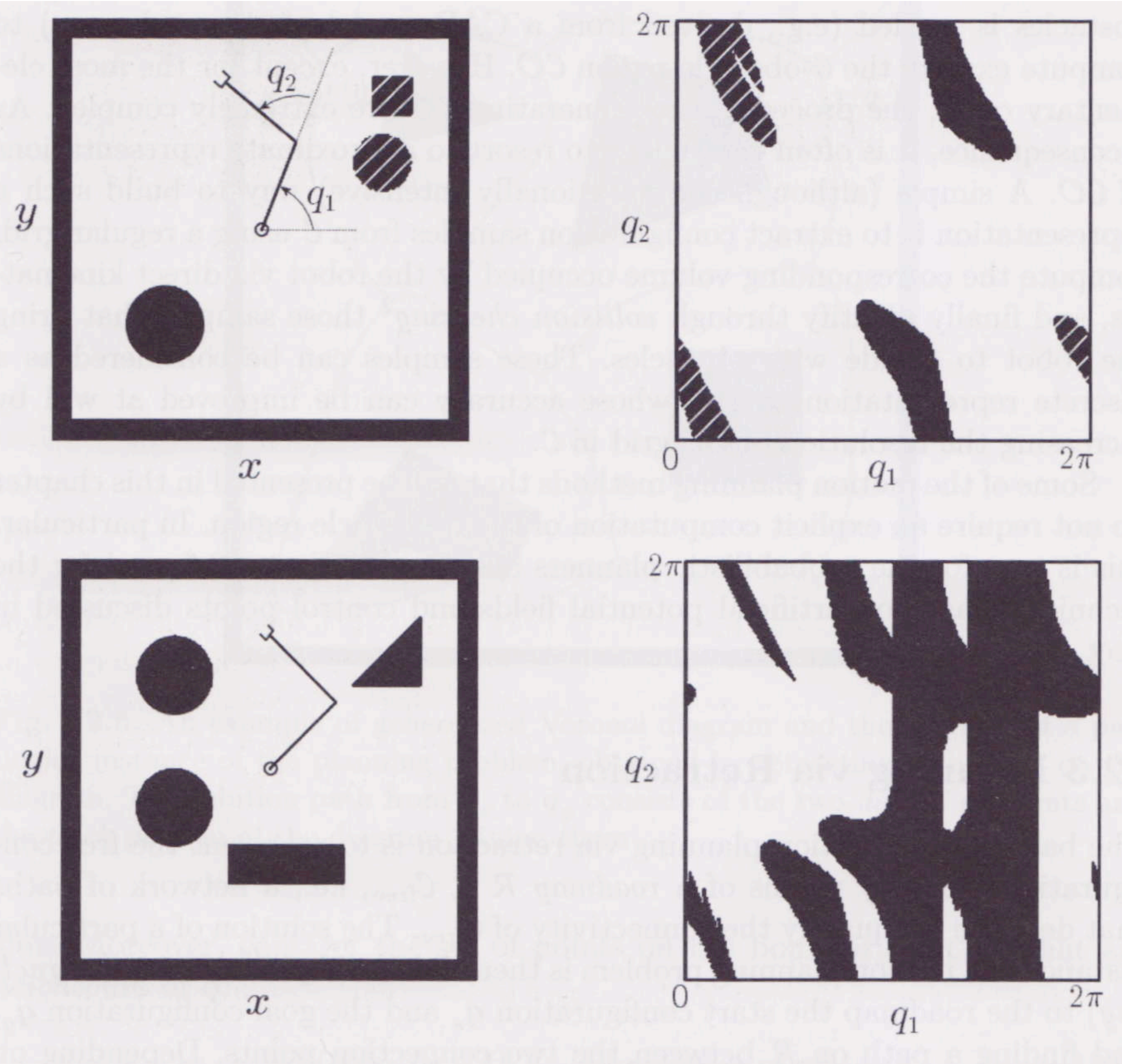


C-space: *subset of R^2*

The configuration space *locally* looks like R^2, and hence is a 2-D *manifold*

Fixed robot arm with 2 revolute joints but with obstacles

Is there a solution to the motion planning problem for any pair of initial and goal configurations?



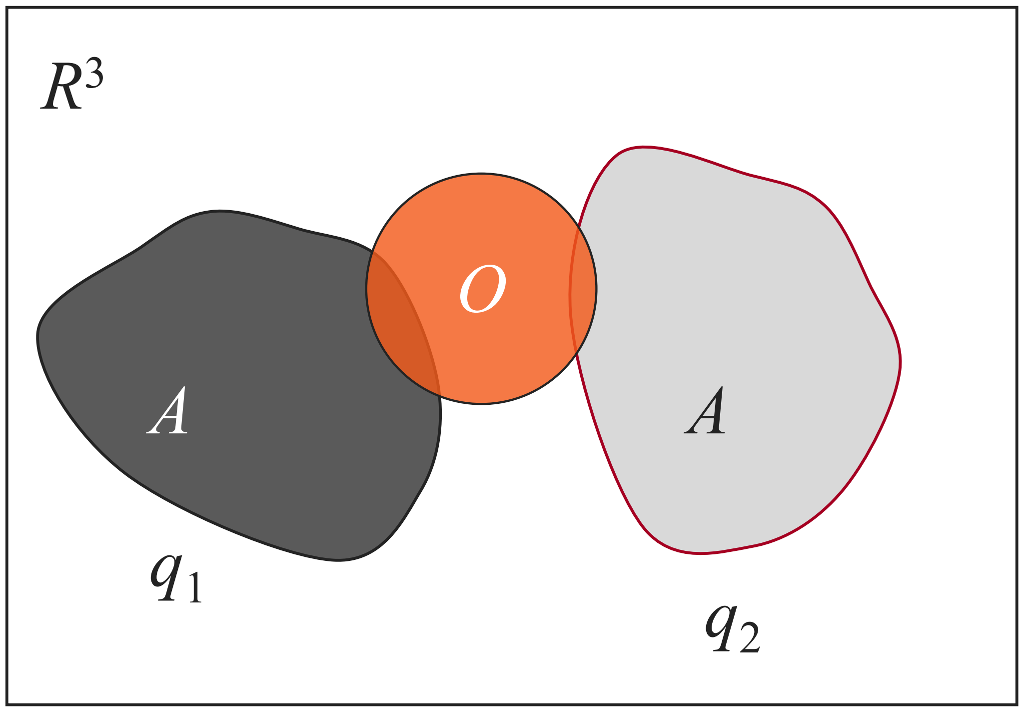
[Siciliano, Sciavicco, Villani, Oriolo, 09]

Obstacles in C-space

Obstacle, O, possibly the union of many disjoint subsets of R^N

The *obstacle region*, $\mathcal{C}_{obs} \subseteq \mathcal{C}$, is defined as

$$\mathcal{C}_{obs} = \{q \in \mathcal{C} \mid \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\}$$



Lavalle, Section 4.3.1

Obstacles in C-space

Obstacle, O, possibly the union of many disjoint subsets of R^N

The *obstacle region*, $\mathcal{C}_{obs} \subseteq \mathcal{C}$, is defined as

$$\mathcal{C}_{obs} = \{q \in \mathcal{C} \mid \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\}$$

Robot with m rigid bodies A_i

$$\mathcal{C}_{obs} = \left(\bigcup_{i=1}^{m} \{q \in \mathcal{C} \mid \mathcal{A}_i(q) \cap \mathcal{O} \neq \emptyset\} \right) \bigcup \left(\bigcup_{[i,j] \in P} \{q \in \mathcal{C} \mid \mathcal{A}_i(q) \cap \mathcal{A}_j(q) \neq \emptyset\} \right)$$

pairs of colliding rigid bodies

e.g.,

upper arm intersecting with forearm
robot 1 colliding with robot 2

Obstacles in C-space

Obstacle, O, possibly the union of many disjoint subsets of R^N

The *obstacle region*, $\mathcal{C}_{obs} \subseteq \mathcal{C}$, is defined as

$$\mathcal{C}_{obs} = \{q \in \mathcal{C} \mid \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\}$$

Robot with m rigid bodies A_i

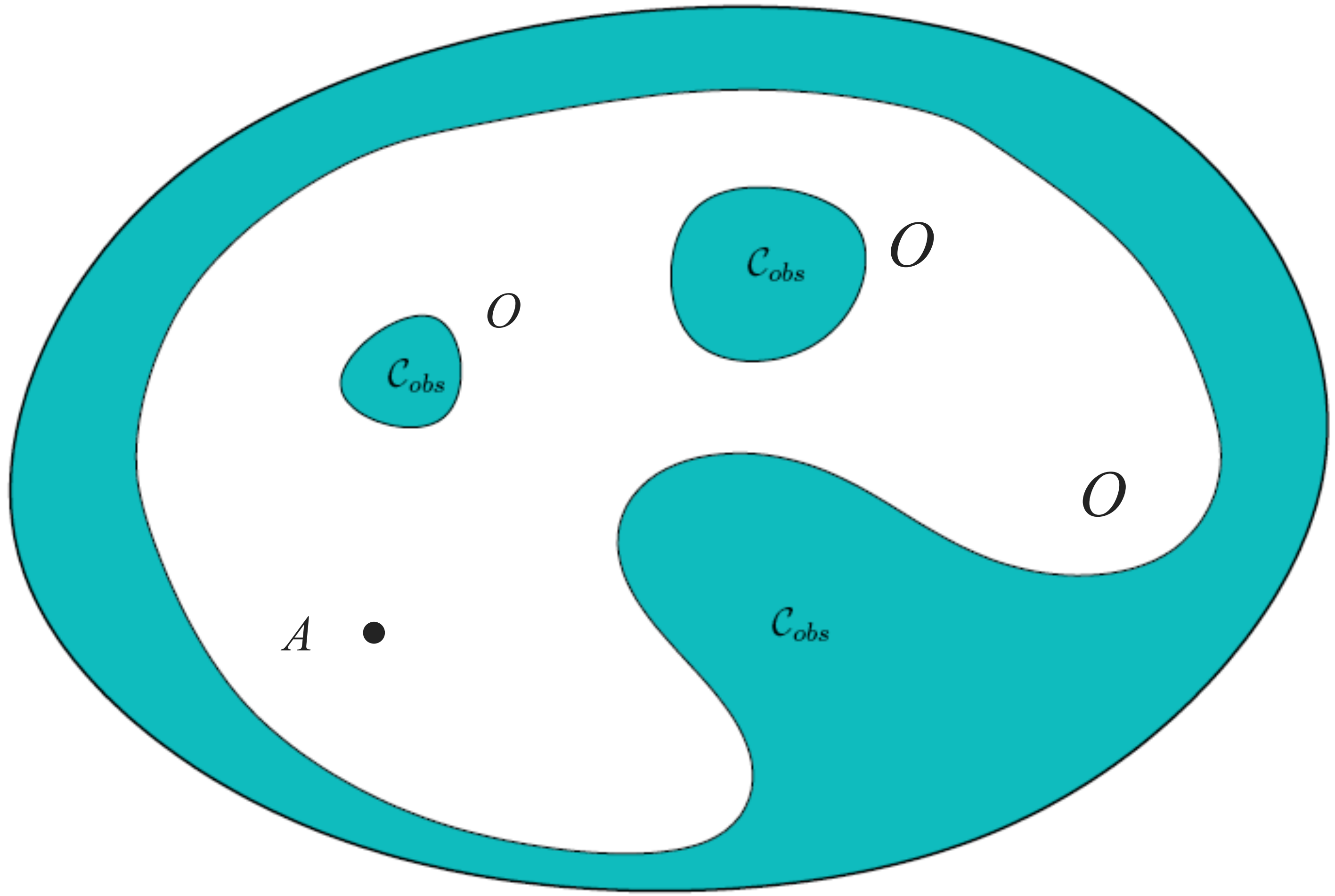
$$\mathcal{C}_{obs} = \left(\bigcup_{i=1}^{m} \{q \in \mathcal{C} \mid \mathcal{A}_i(q) \cap \mathcal{O} \neq \emptyset\} \right) \bigcup \left(\bigcup_{[i,j] \in P} \{q \in \mathcal{C} \mid \mathcal{A}_i(q) \cap \mathcal{A}_j(q) \neq \emptyset\} \right)$$

pairs of colliding rigid bodies

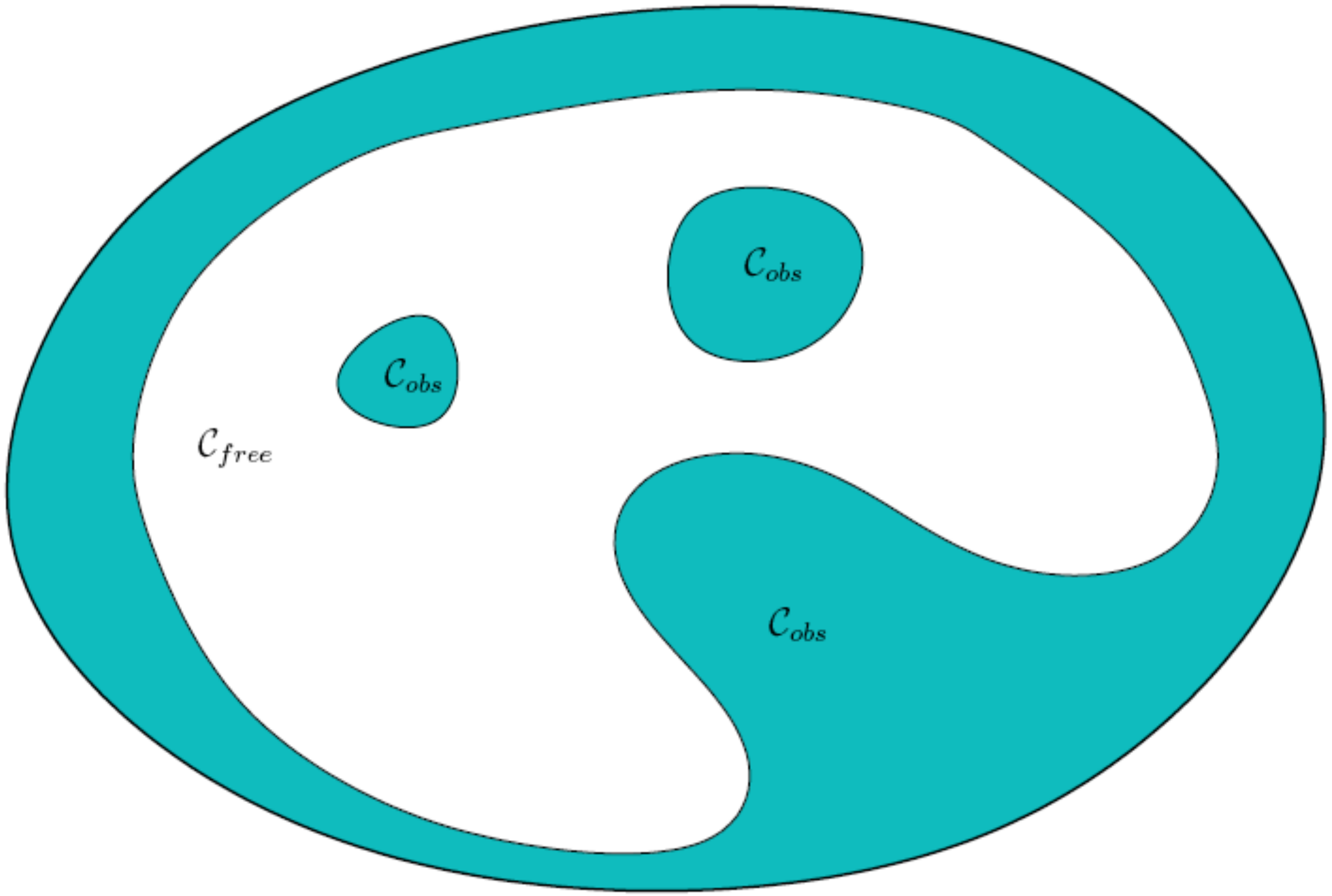
The free space, C_{free}, is an open set in R^N

$$\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obs}$$

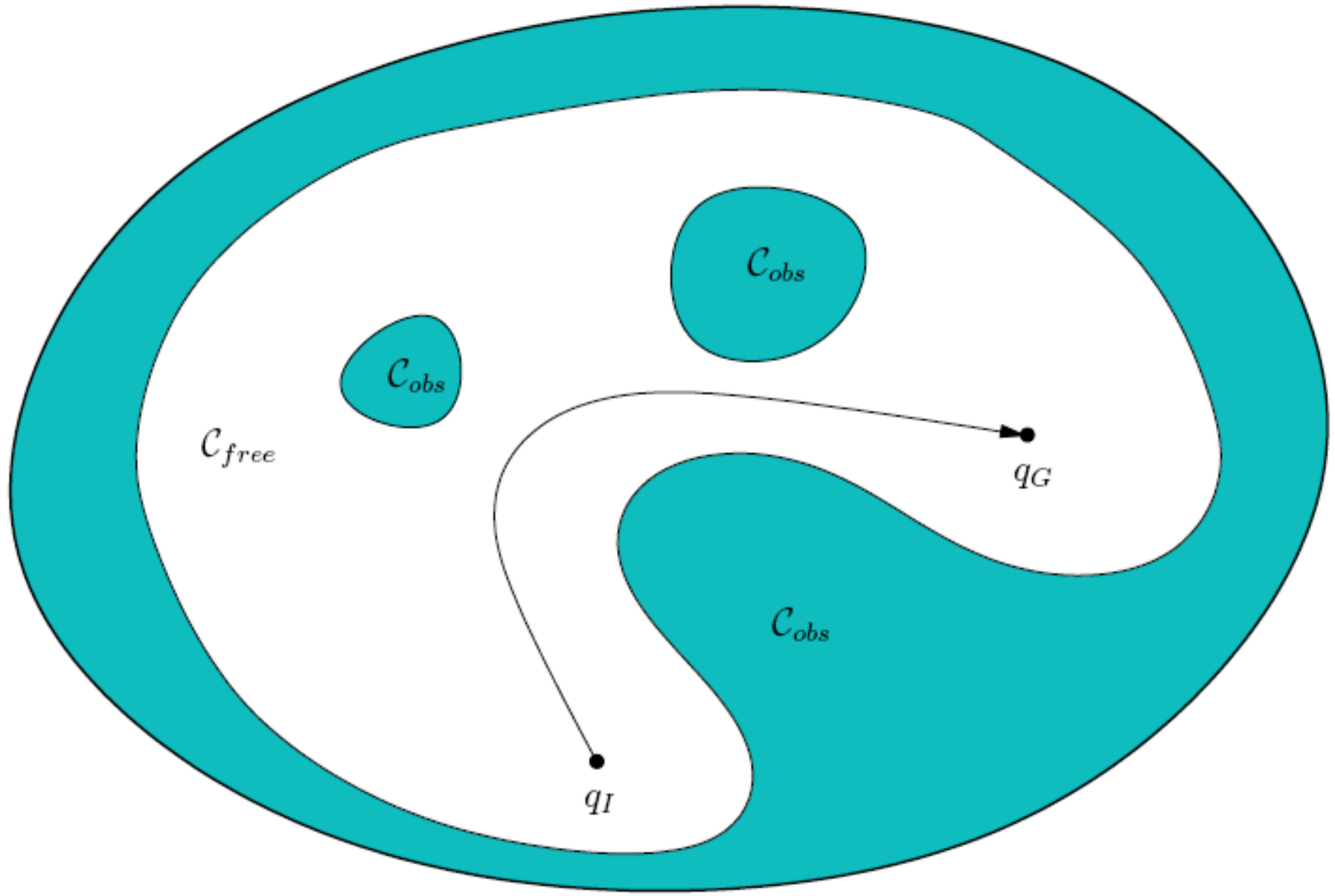
Example: Point Robot in the Plane



Obstacle Region and Free Space



The Basic Motion Planning Problem



There exists a motion plan from q_I to q_G *iff* q_I and q_G belong to the same connected component of C_{free}

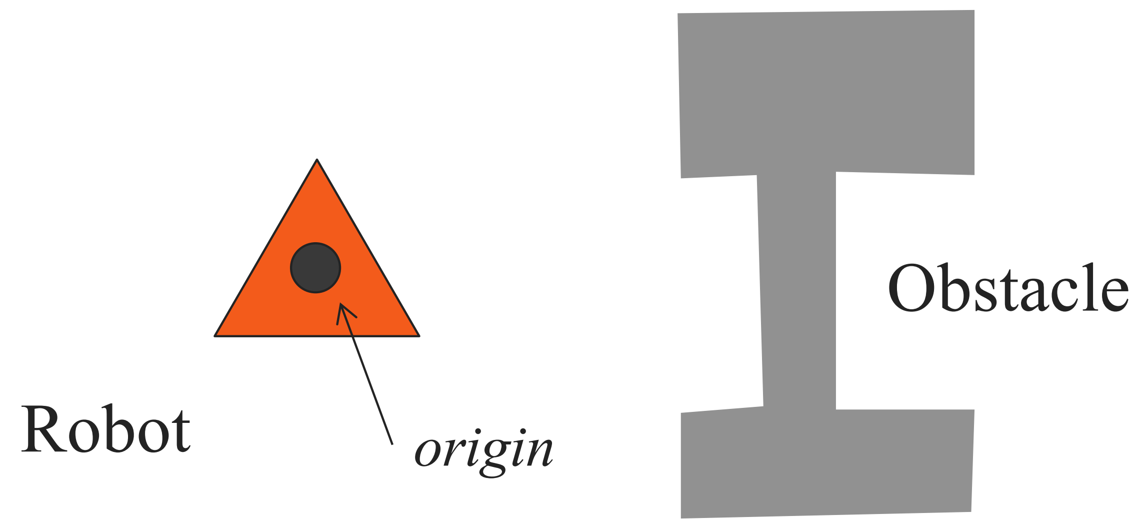
Lavalle, 4.3.1

Penn Engineering

Modeling Obstacle Regions and Free Space for a Robot with Finite Extent

Example: A single-rigid-body robot that can only translate in R^2

(configuration space is R^2)



The *obstacle region*, $\mathcal{C}_{obs} \subseteq \mathcal{C}$, is defined as

$$\mathcal{C}_{obs} = \{q \in \mathcal{C} \mid \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\}$$

Modeling Obstacle Regions and Free Space for a Robot with Finite Extent

Example: A single-rigid-body robot that can only translate in R^2

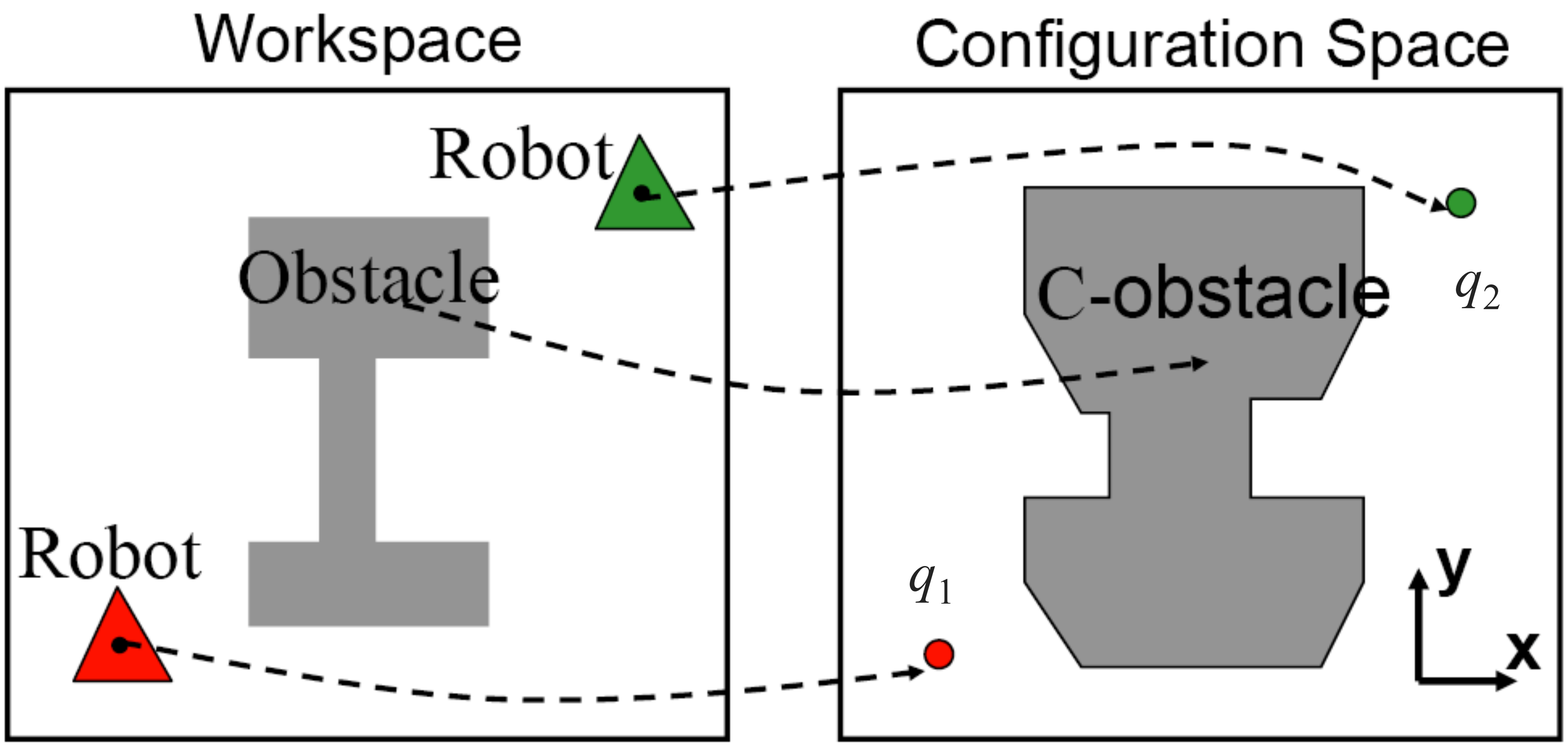


Figure by Dinesh Manocha, UNC

Key Idea: Minkowski Sum

The *Minkowski* sum of two sets A and B

$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$

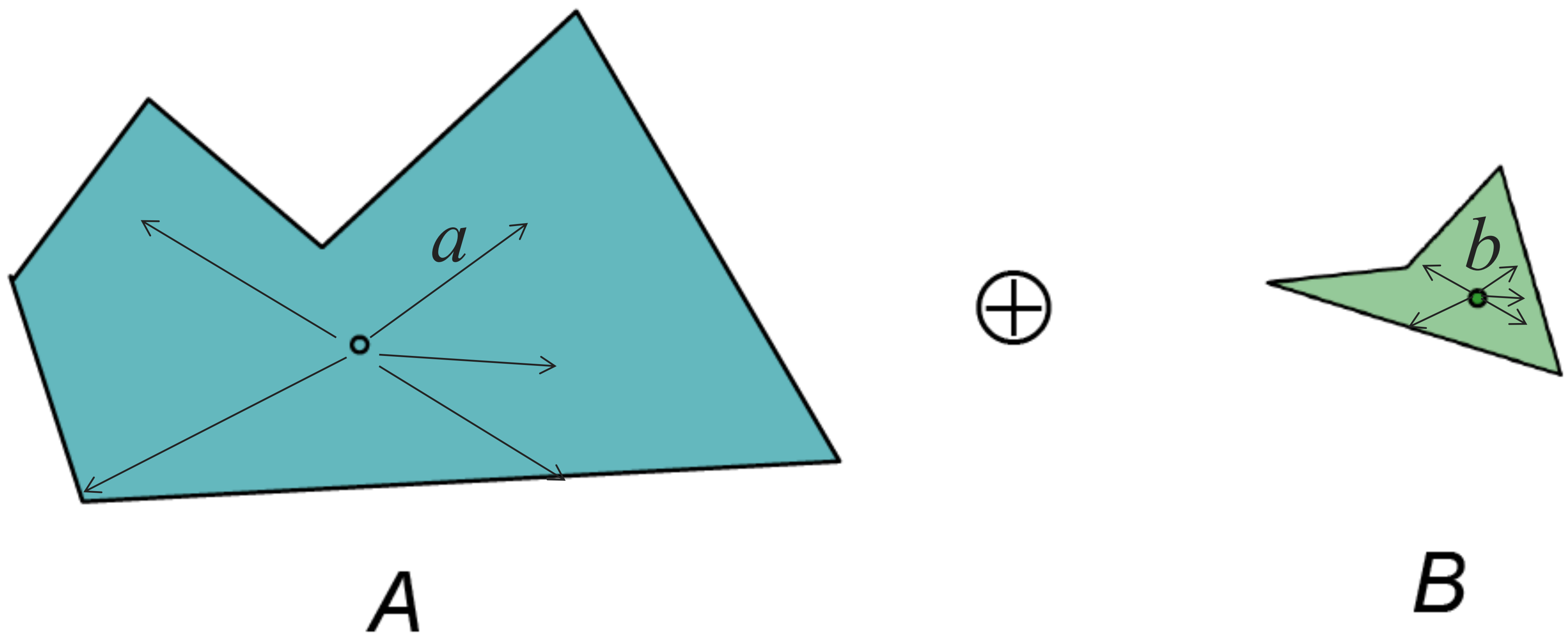


Figure by Dinesh Manocha, UNC

$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$

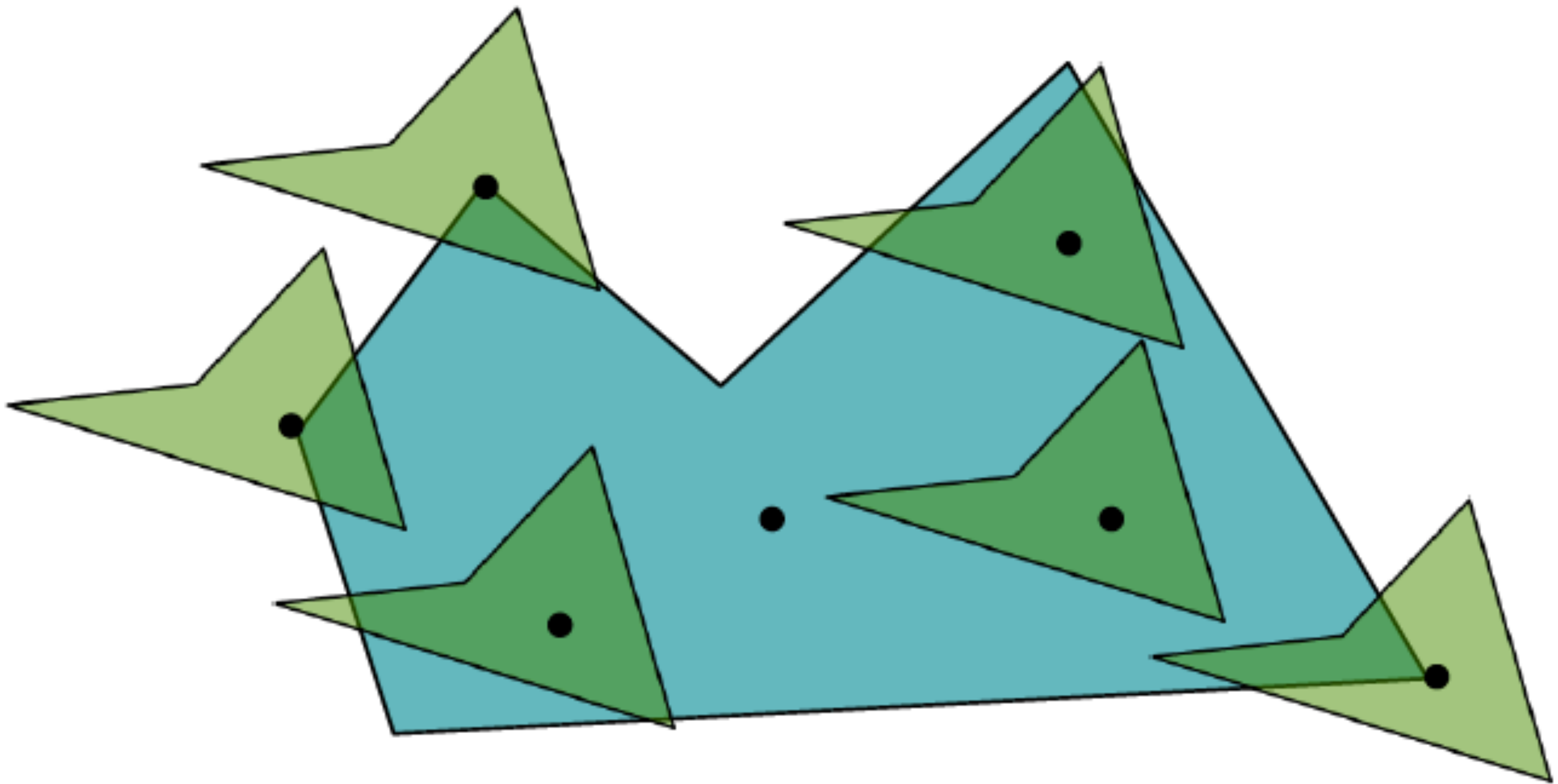


Figure by Dinesh Manocha, UNC

Penn Engineering

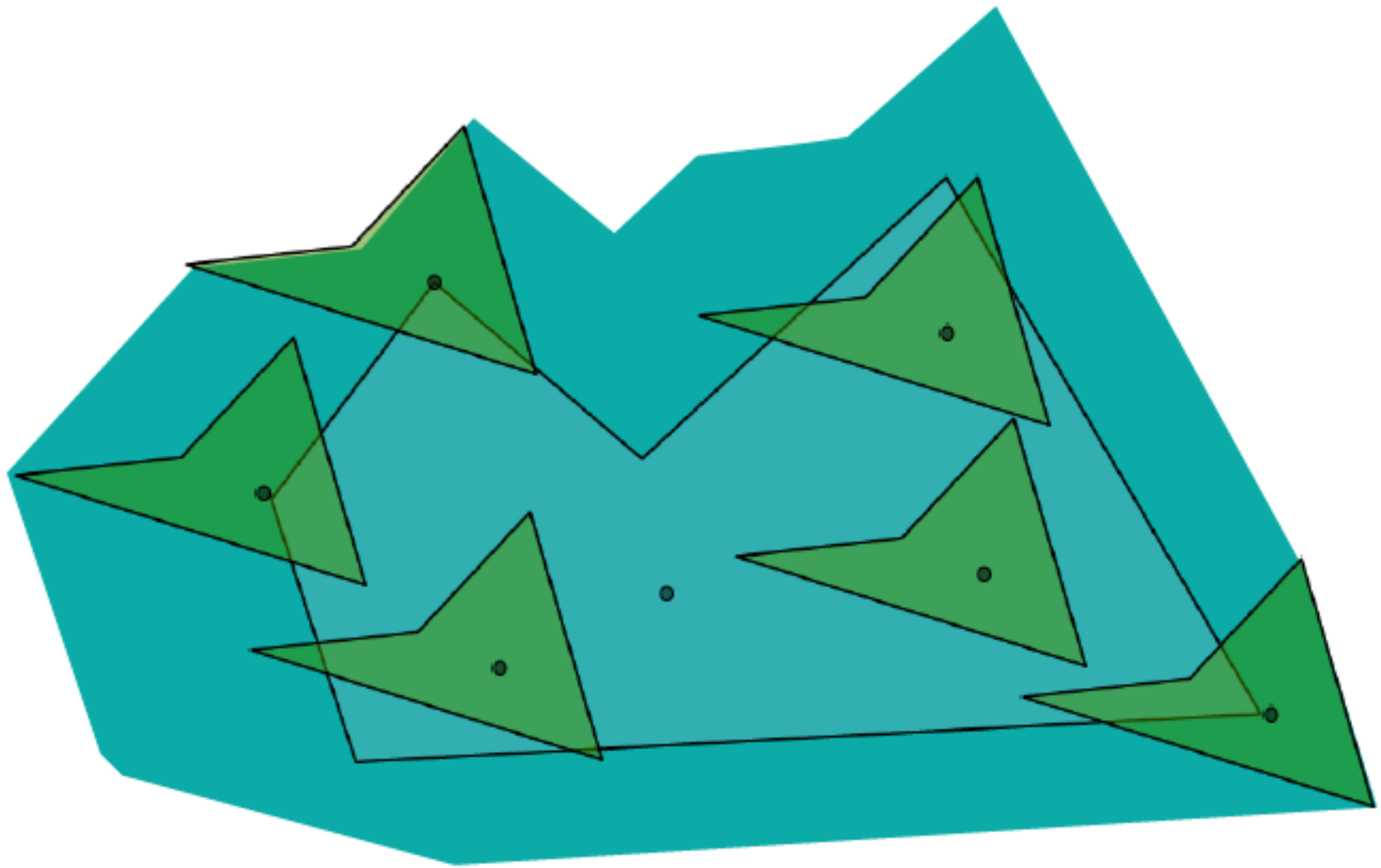


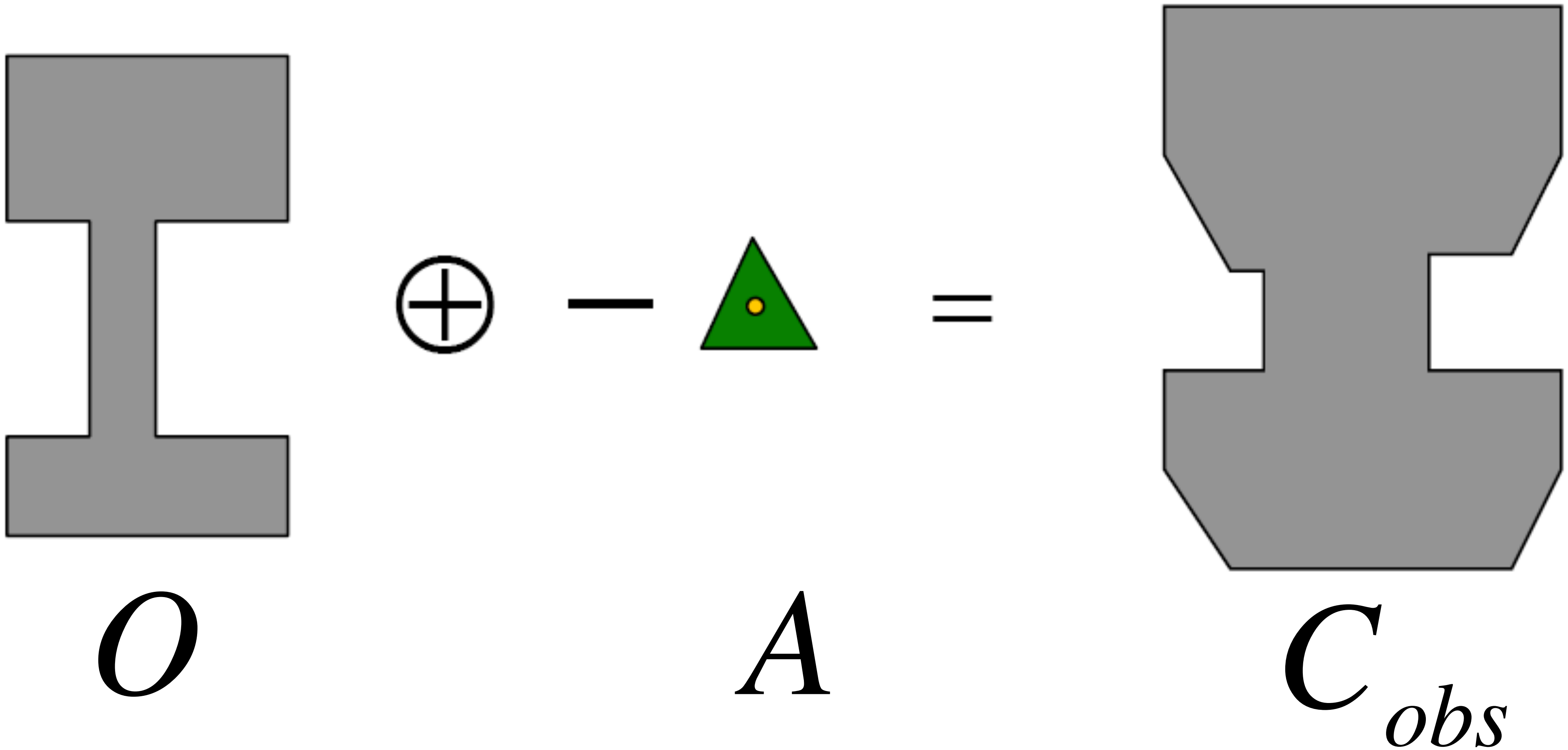
Figure by Dinesh Manocha, UNC

$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$

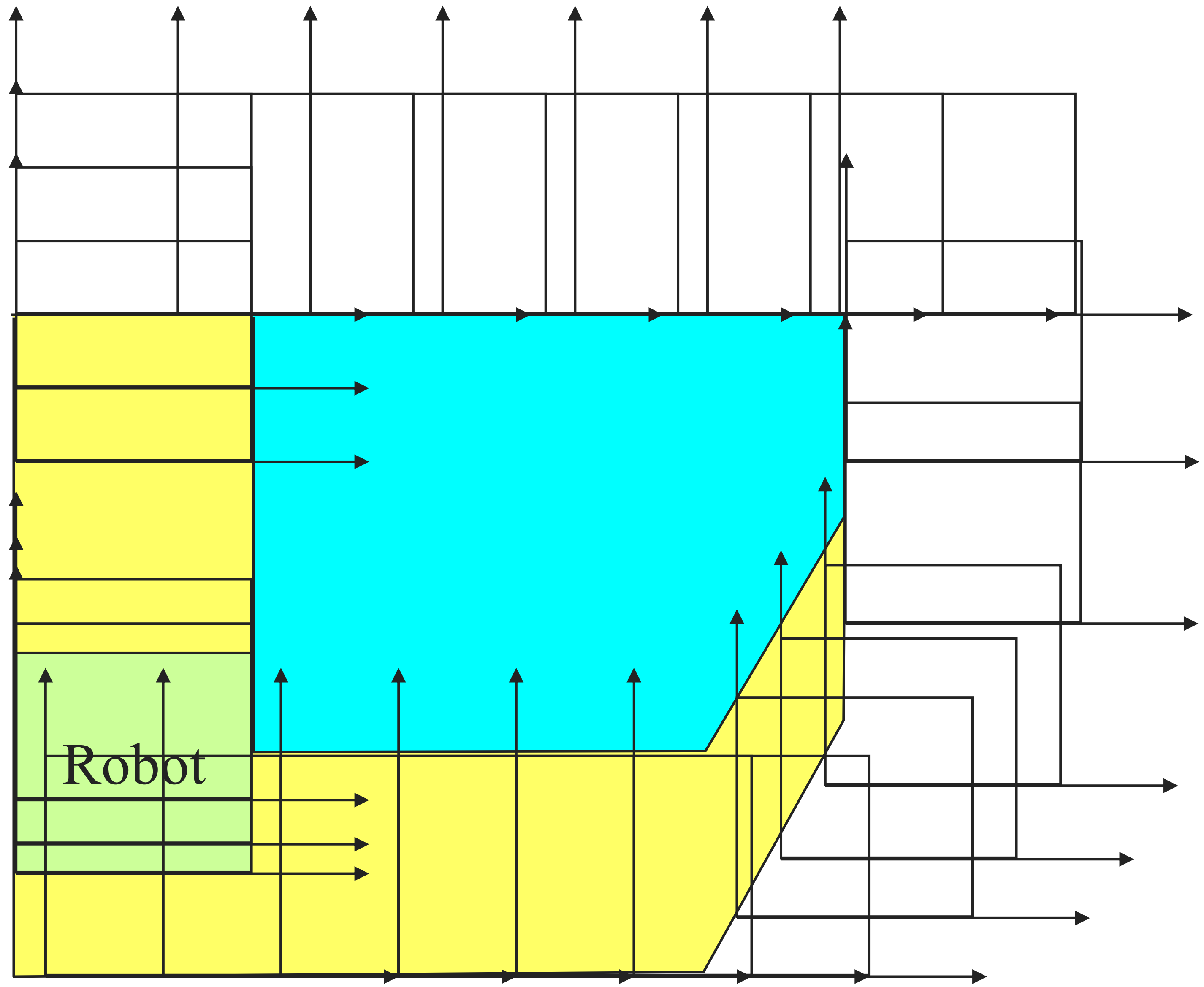
Figure by Dinesh Manocha, UNC

Lozano-Perez and Wesley, 1979

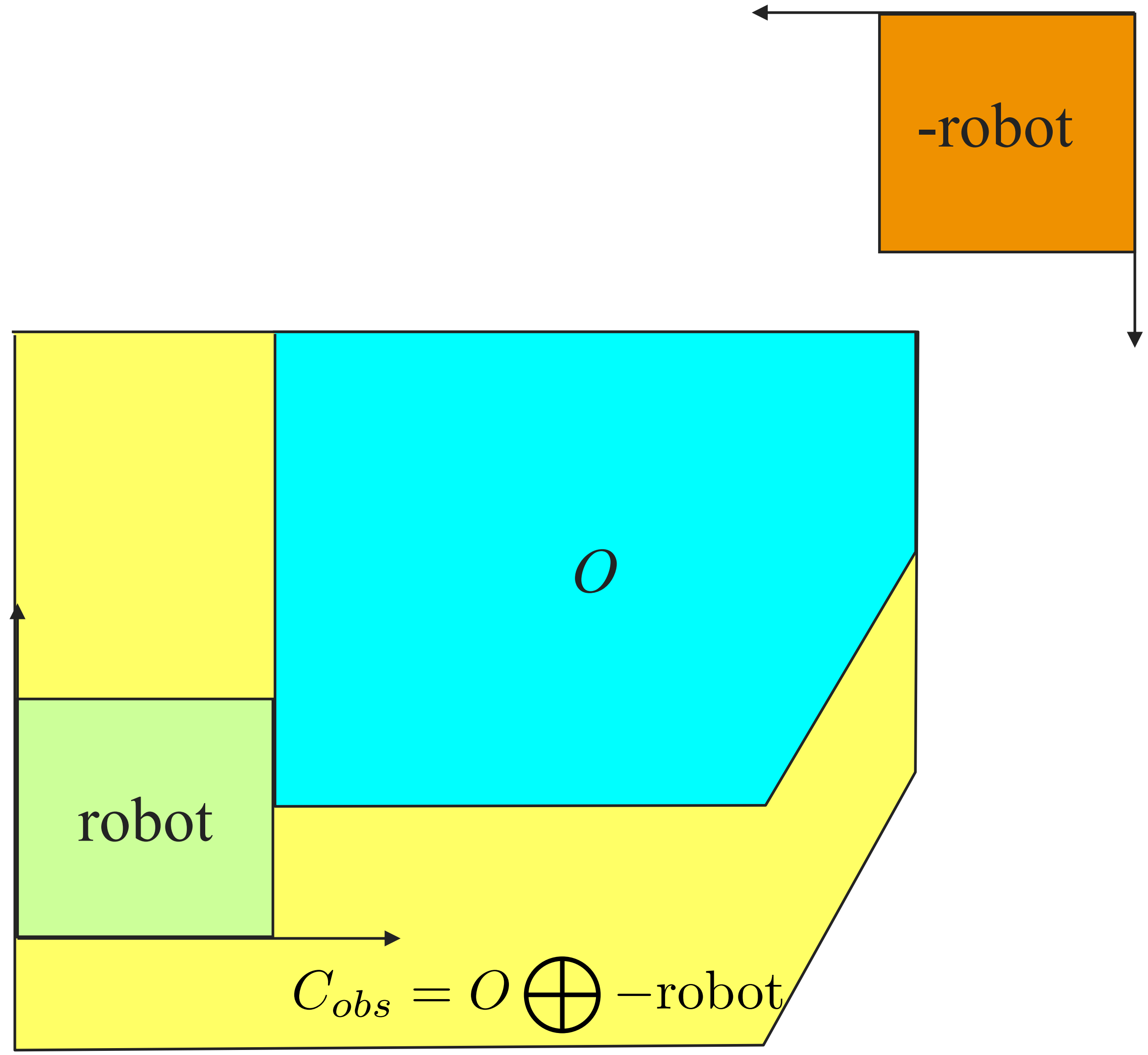
$$C_{obs} = O \oplus -A$$



Example



Example



Complexity

O – 2D convex polygon, m vertices

A – 2D convex polygon, n vertices

Minkowski sum is a convex polygon of $m+n$ vertices

Run time ~ $O(n+m)$

Nonconvex case is much harder

- decompose into convex polygons
- compute Minkowski sums
- take unions

Run time ~ $O(n^2m^2)$

Run time ~ $O(n^3m^3)$ in 3D

Penn Engineering

END OF SEGMENT 2

Obstacle Regions for Translation

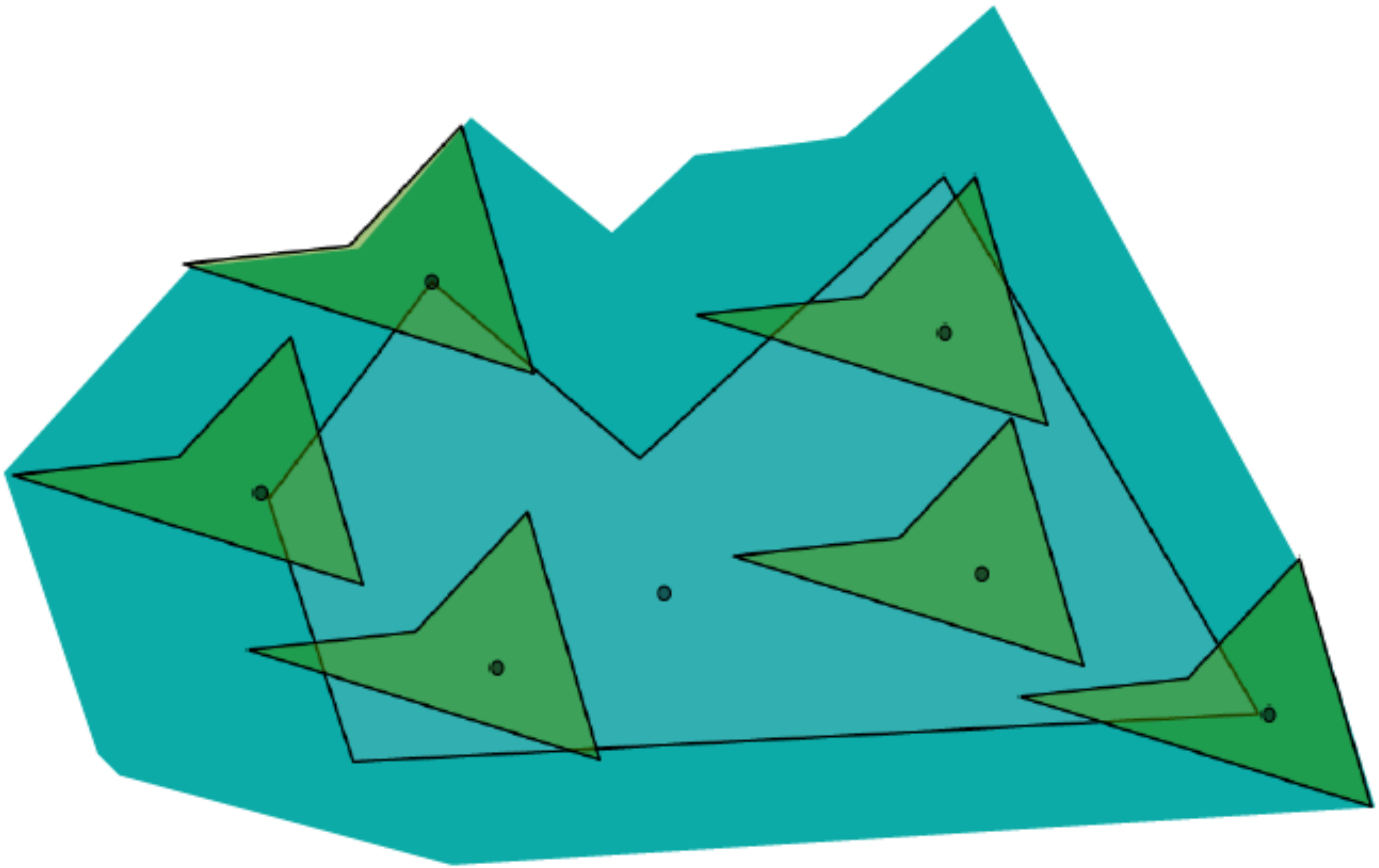
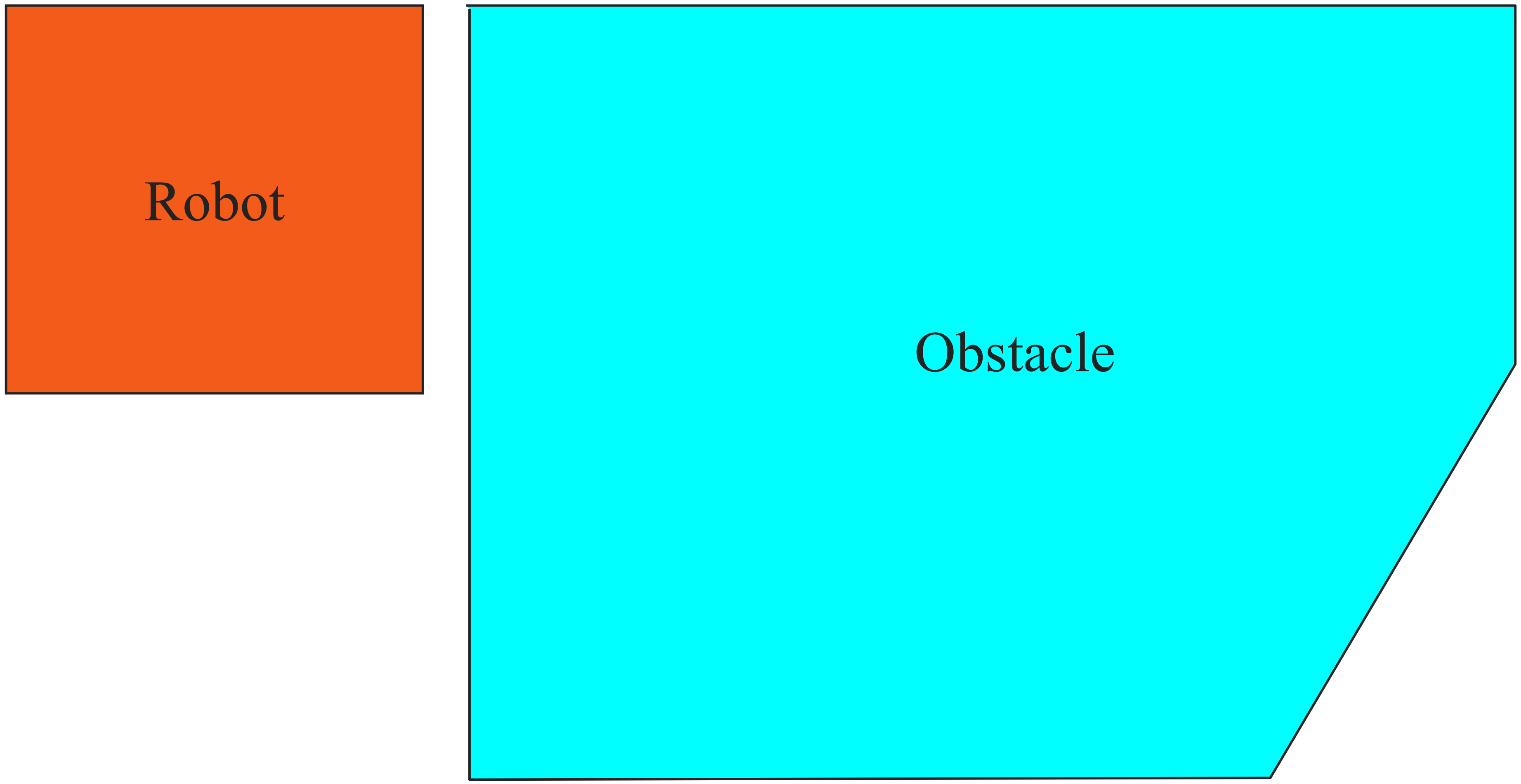


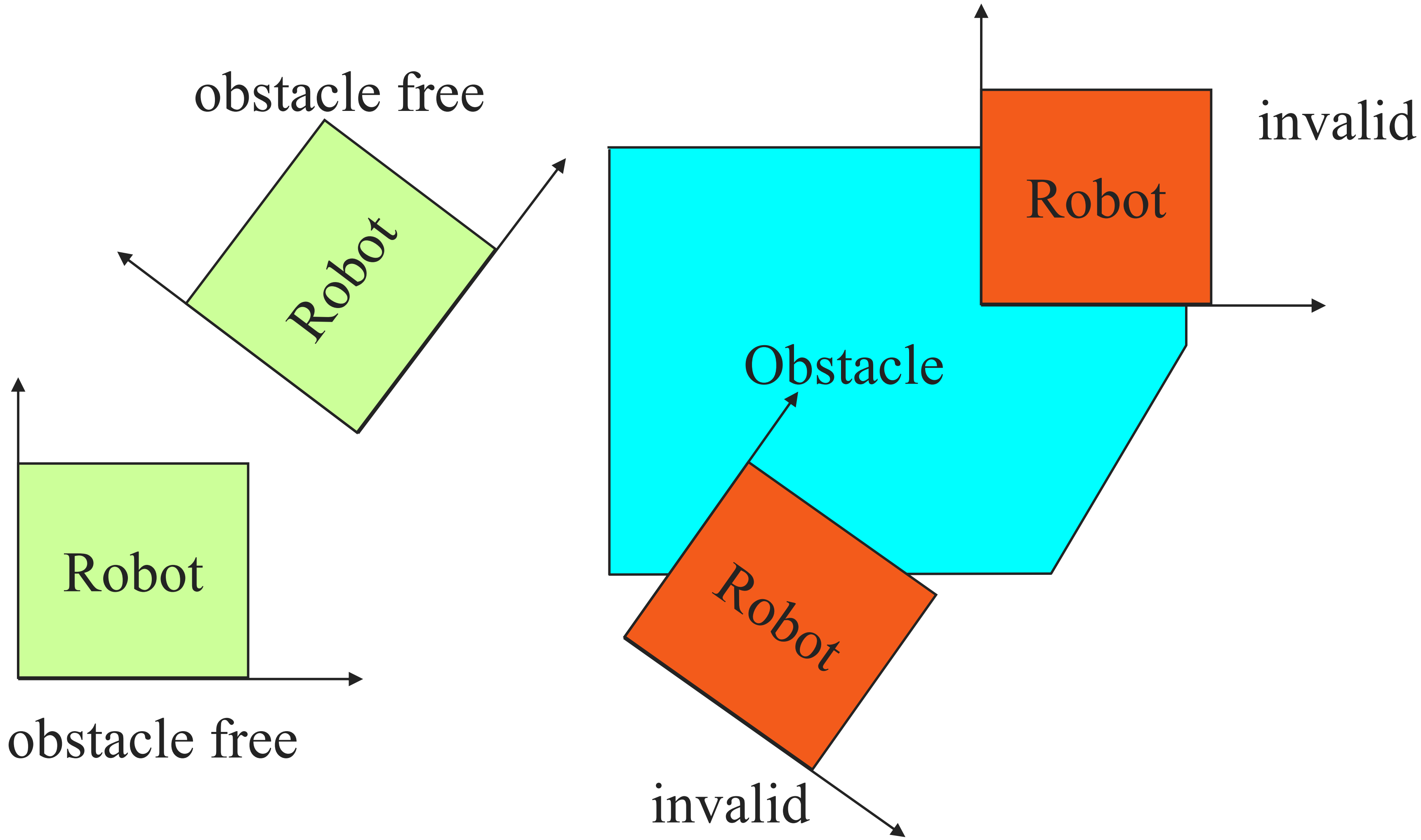
Figure by Dinesh Manocha, UNC

Penn Engineering

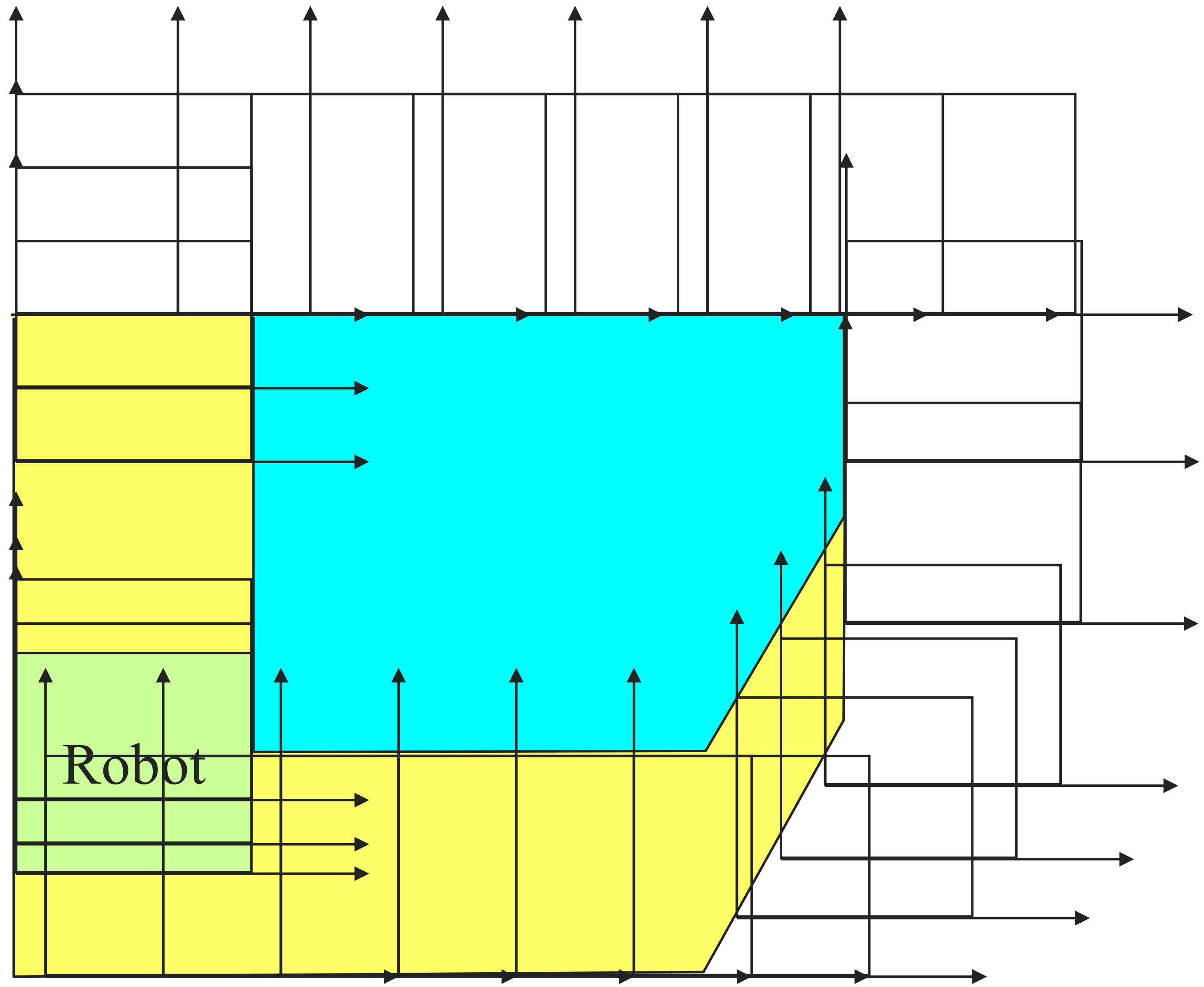
What if the robot is rigid body that can translate <u>and rotate</u> in the plane?



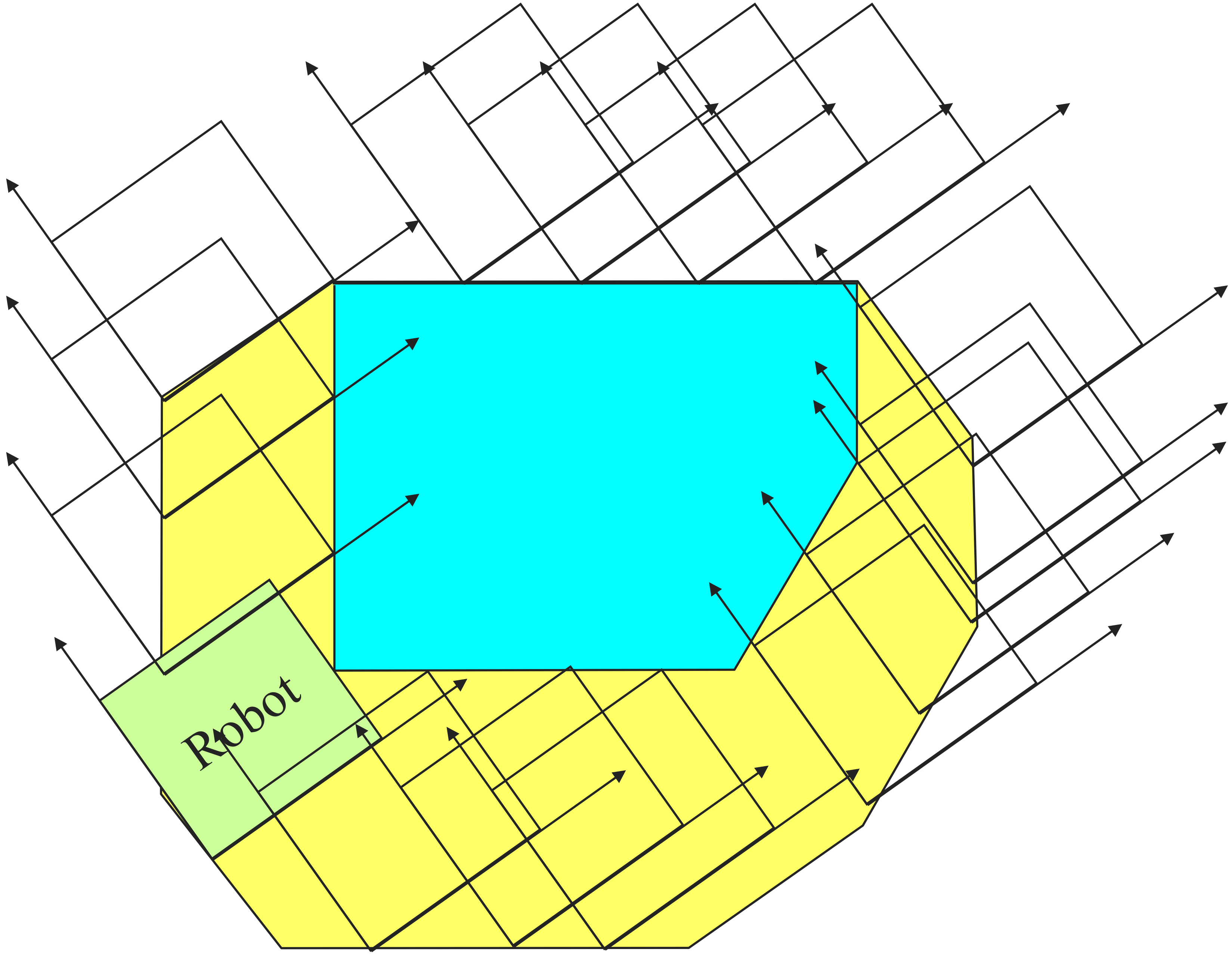
C-space


obstacle free
Robot
Robot
obstacle free
invalid
Robot
Obstacle
Robot
invalid

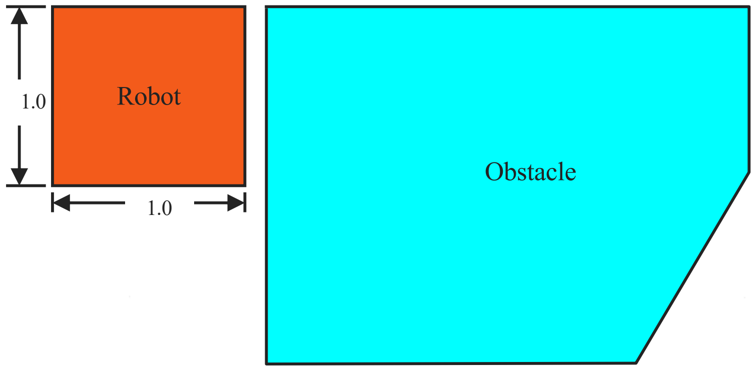
$C_{obs}\,(\theta=0)$

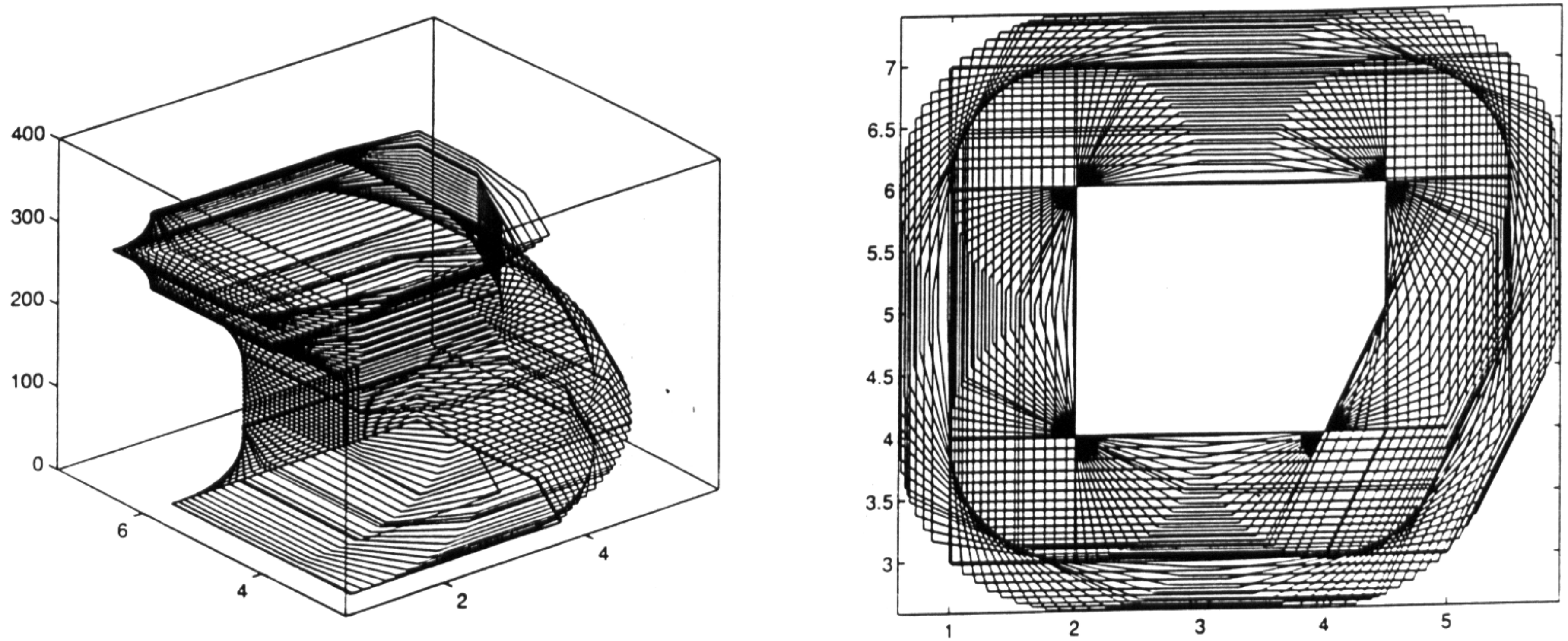


$C_{obs}\ (\theta=\theta_0\neq 0)$



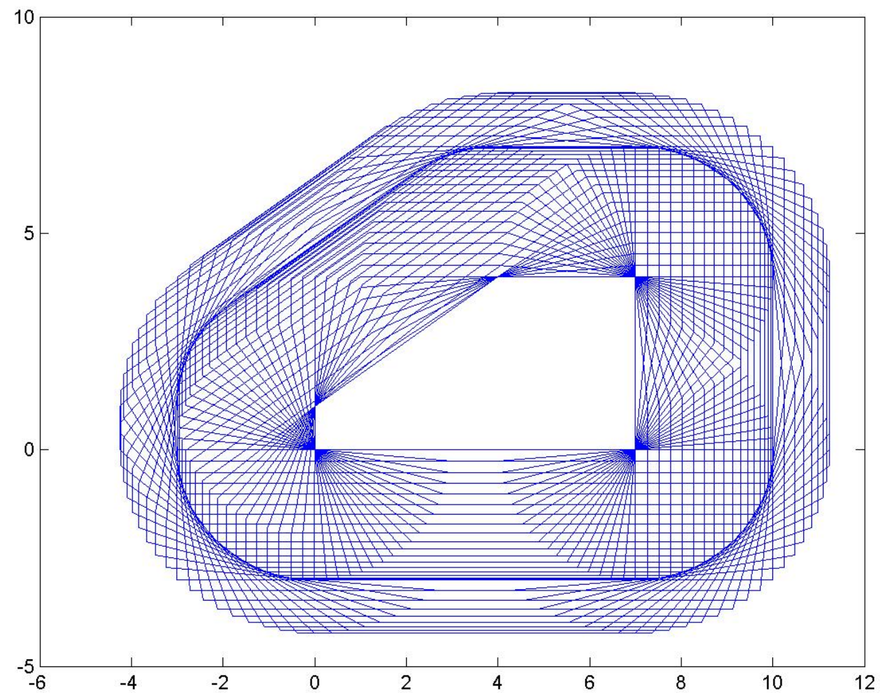
Slices of C_{obs}

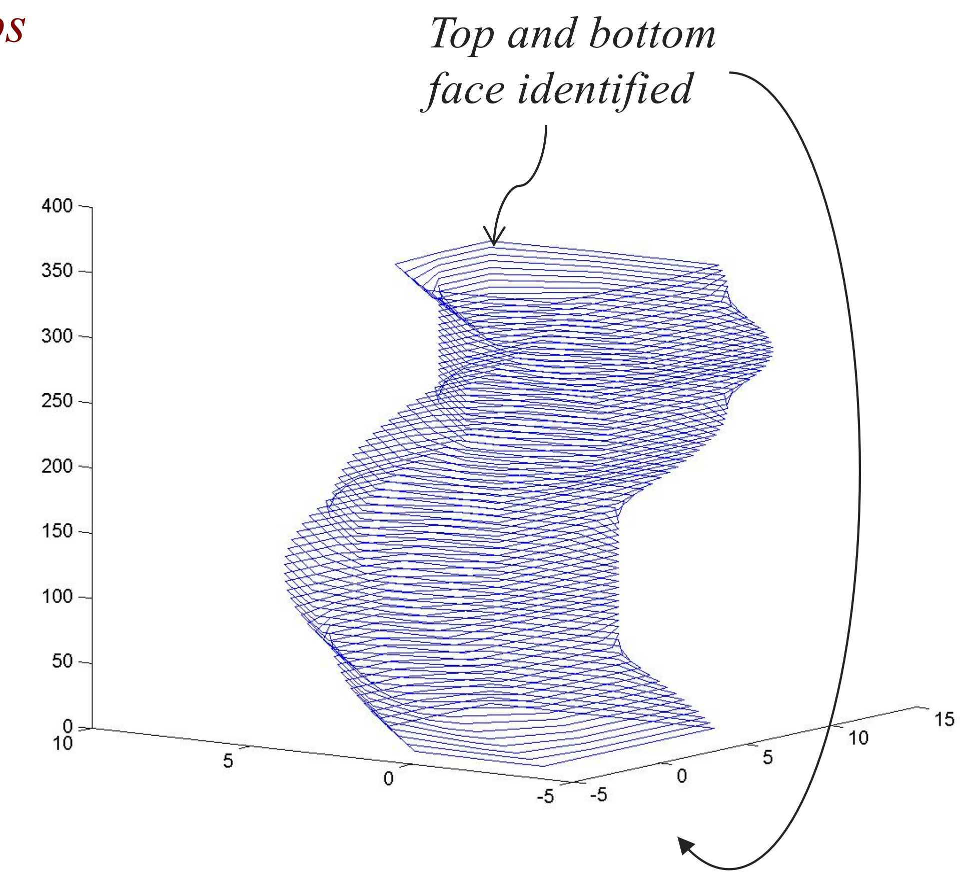




Each slice is a convex polygon of $\leq$ 4+5 vertices

C_{obs}





Details on computation with polygonal objects/robots available in Lavalle (3.1.1, 4.3)

C_{obs} is modeled as a union of semi-algebraic sets (Lavalle 3.1.2)