Configuration Space

Jane Li

Assistant Professor
Mechanical Engineering Department, Robotic Engineering Program
Worcester Polytechnic Institute

Quiz (10 pts)

- (5 pts) Describe one challenge that novice user faces in the teleoperation of TRINA?
- (5 pts) Explain one method to help with this problem

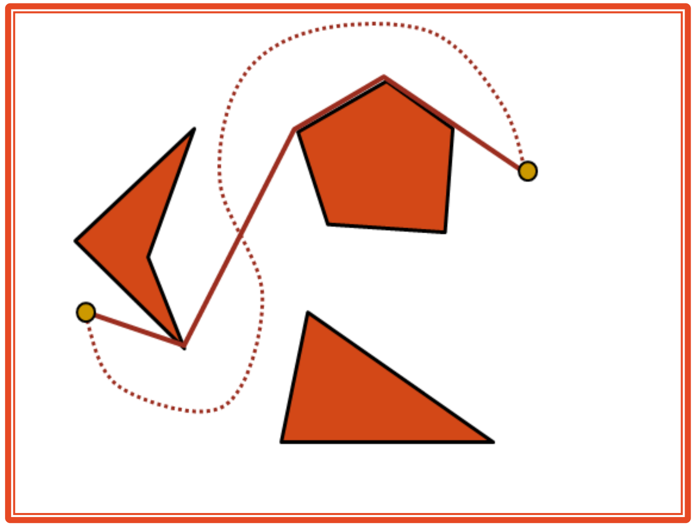
Challenges

- Motion
  - Many DOFs to control
  - Coordinated dexterous manipulation
  - User interface are not intuitive
- Perception
  - Hard to perceive spatial relationship through multiple 2D images
  - Lack of tactile sensing

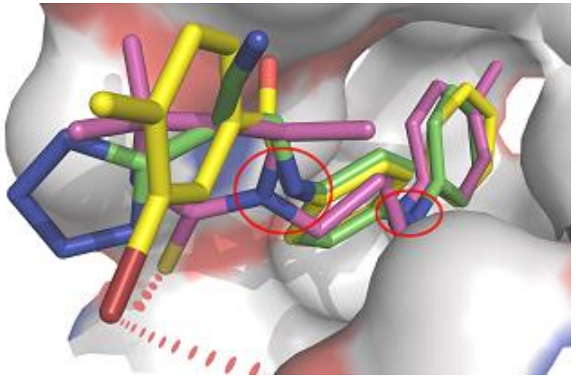
Configuration space

Recap

- Plan paths for a point in 2D → simple
- Real-world robots are **complex**, often **articulated** bodies



A space where the robots could be treated as points?



Configuration space

- **Configuration q**
  - A specification of the position of **every** point on the object.
  - Expressed as a vector of the **DOF** of the robot

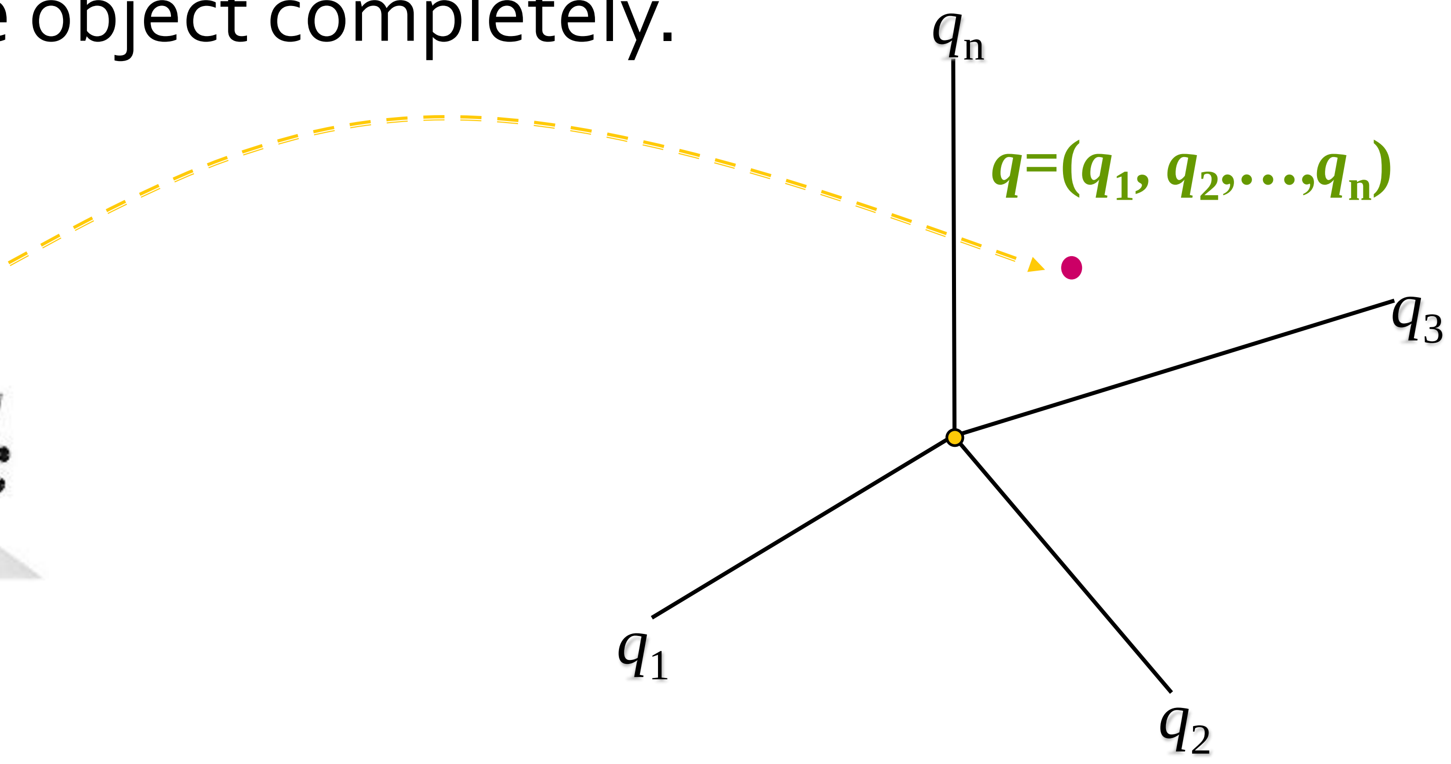
$$q = (q_1, q_2, \ldots, q_n)$$

- **Configuration space C**
  - The set of all possible configurations

A configuration q is a point in C

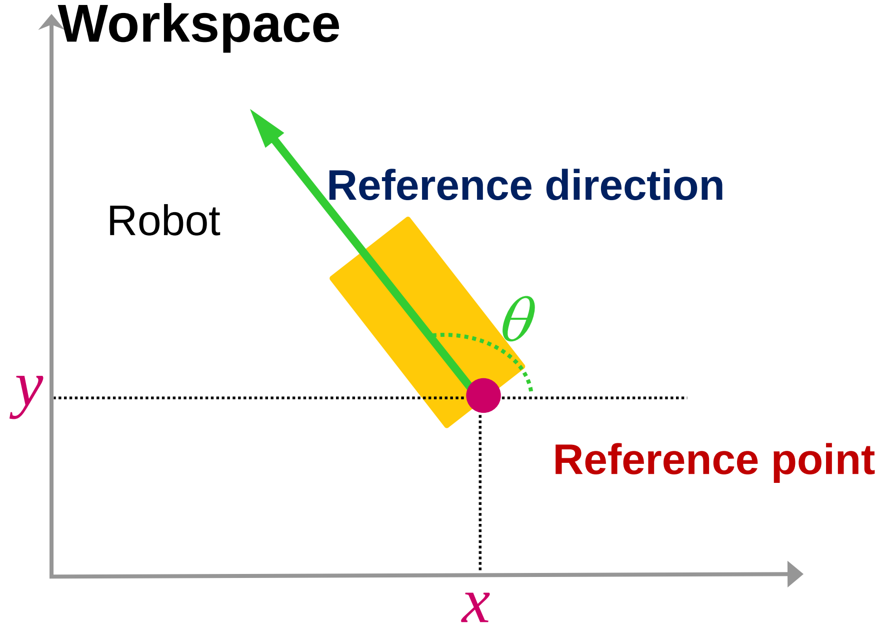
Dimension of Configuration Space

- The **minimum** number of DOF needed to specify the configuration of the object completely.



Example – A Rigid 2D Mobile Robot

- 3-parameters: $q = (x, y, \theta)$ with $\theta \in [0, 2\pi)$.
- C-space dimension = 3
- Topology?
  - $SE(2) = R^2 \times S^1$
- Shape of C-space?
  - Cylinder



Example – Rigid Robot in 3D workspace

- q = (position, rotation) = (x, y, z, ???)
- Representations for rotation?
  - Euler Angles – yaw, pitch roll
  - 3X3 Transform Matrices
  - Unit quaternion
- Regardless of the representation, rotation in 3D is 3 DOF

Example – Rigid Robot in 3D workspace

- C-space dimension = **6**

- Topology?
  - $SE(3) = R^3 \times SO(3)$

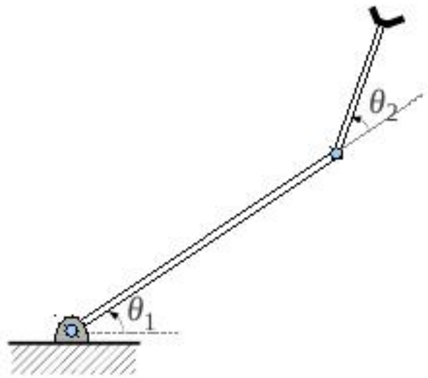
Configuration Space for Articulated Objects

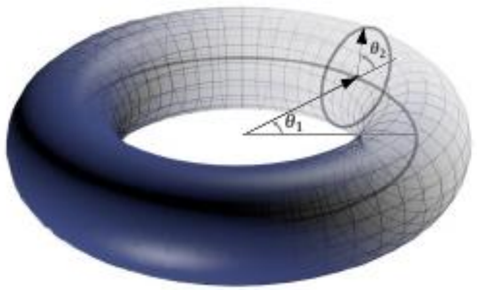
- Articulated object
  - A set of rigid bodies connected by joints

- For articulated robots (arms, humanoids, etc.), the DOF are **usually** the joints of the robot
  - Exceptions?

Configuration Space for Articulated Objects

- Topology of two-link manipulator?





With joint limits?

Path and Trajectory in C-Space

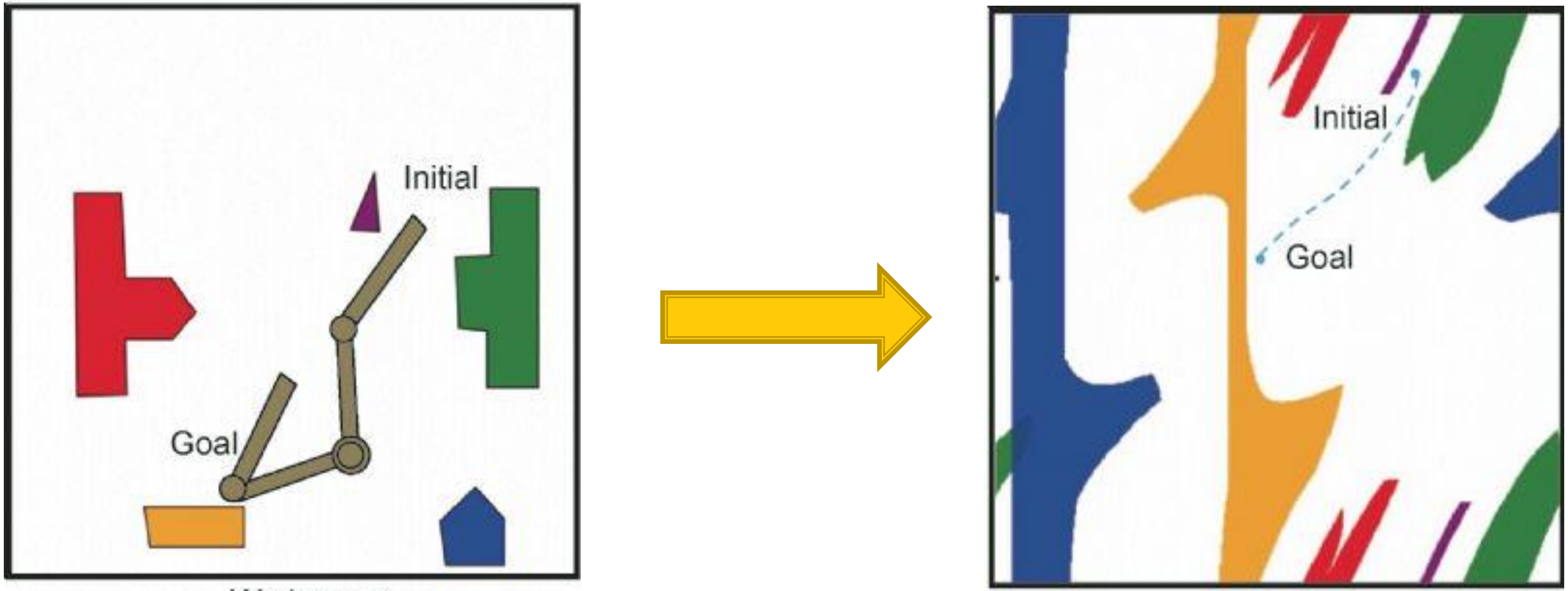
- Path
  - A continuous curve connecting two configurations q_{start} and q_{goal}

$$\tau : s \in [0,1] \rightarrow \tau(s) \in C$$

- Trajectory
  - A path parameterized by time

$$\tau : t \in [0,T] \rightarrow \tau(t) \in C$$

Obstacles in C-space



Workspace

Configuration space

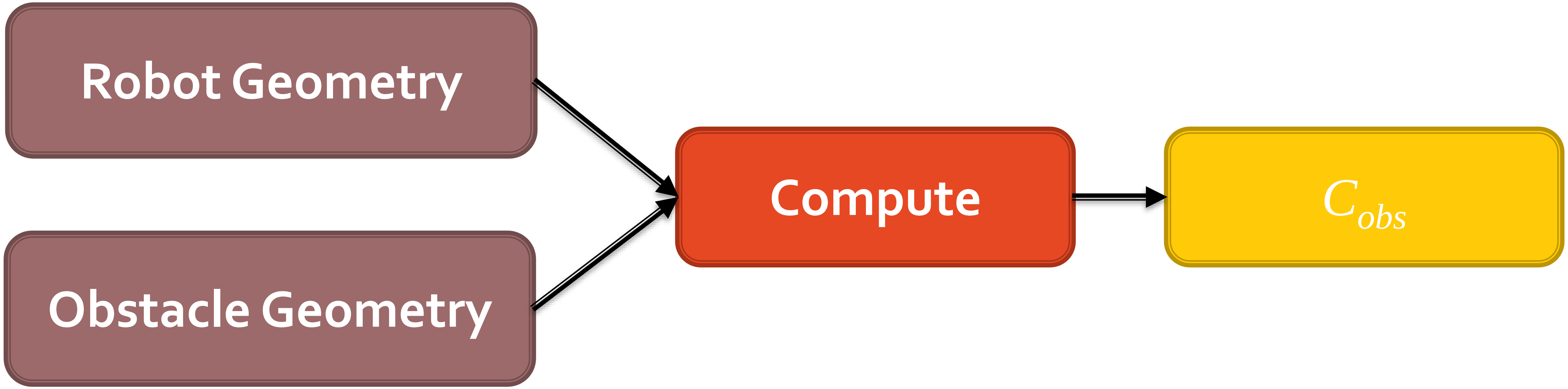
Configuration space obstacle

- (Collision)-free configuration – $\boldsymbol{q}$
  - Robot placed at $\boldsymbol{q}$ has no intersection with any obstacle in the workspace

- Free Space – C_{free}
  - A subset of C that contains all free configurations

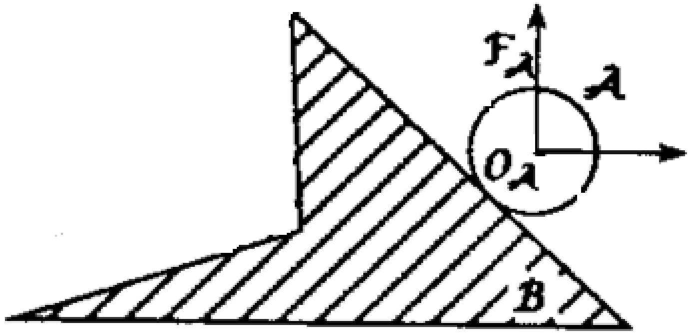
- Configuration space obstacle – C_{obs}
  - A subset of C that contains all configurations where the robot collides with **workspace obstacles** or with **itself**

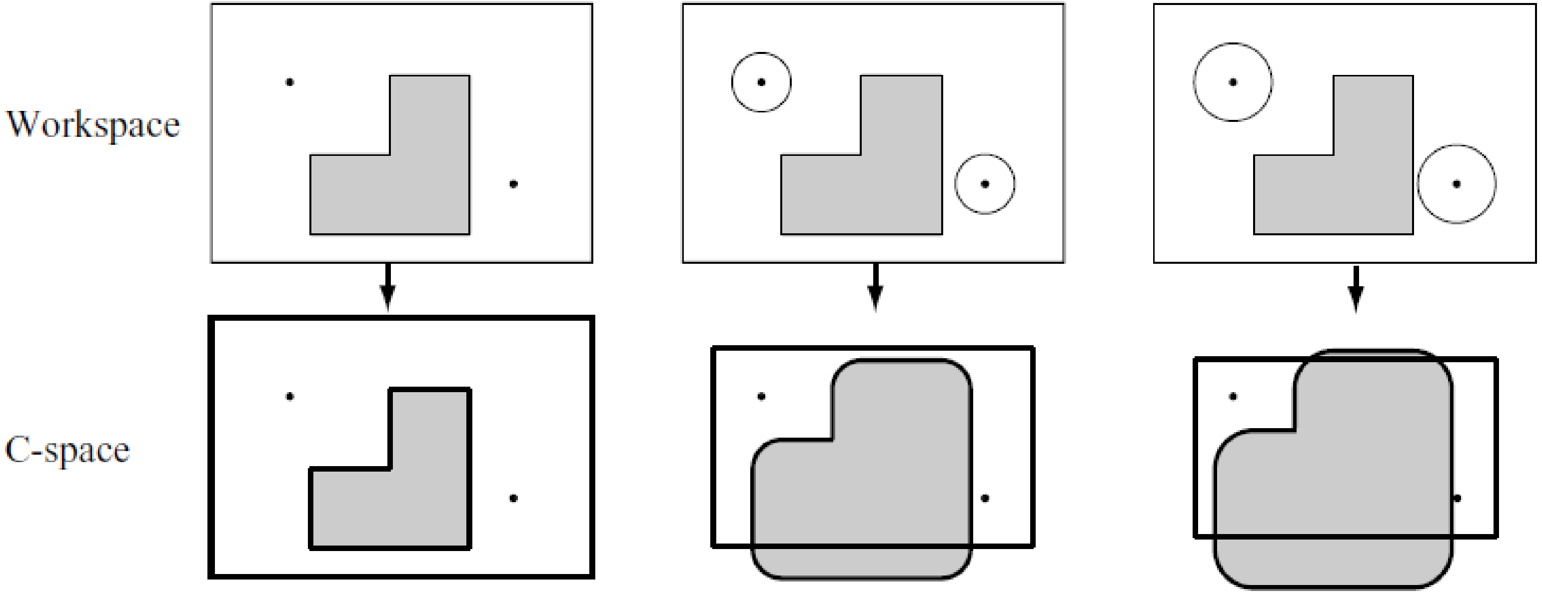
How to compute C_{obs} ?



Example – 2D Robot without Rotation

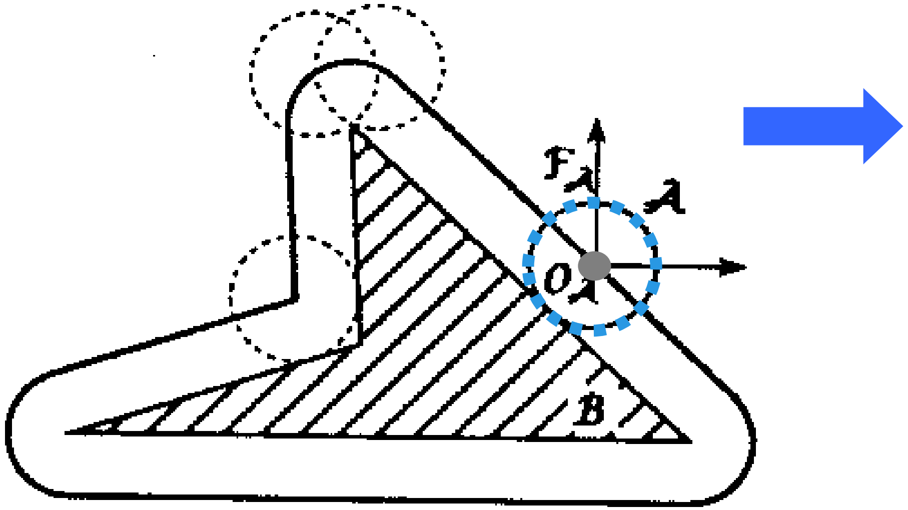
- A simple setup
  - Disc in 2D space → not a point anymore
  - Polygonal obstacle in task space





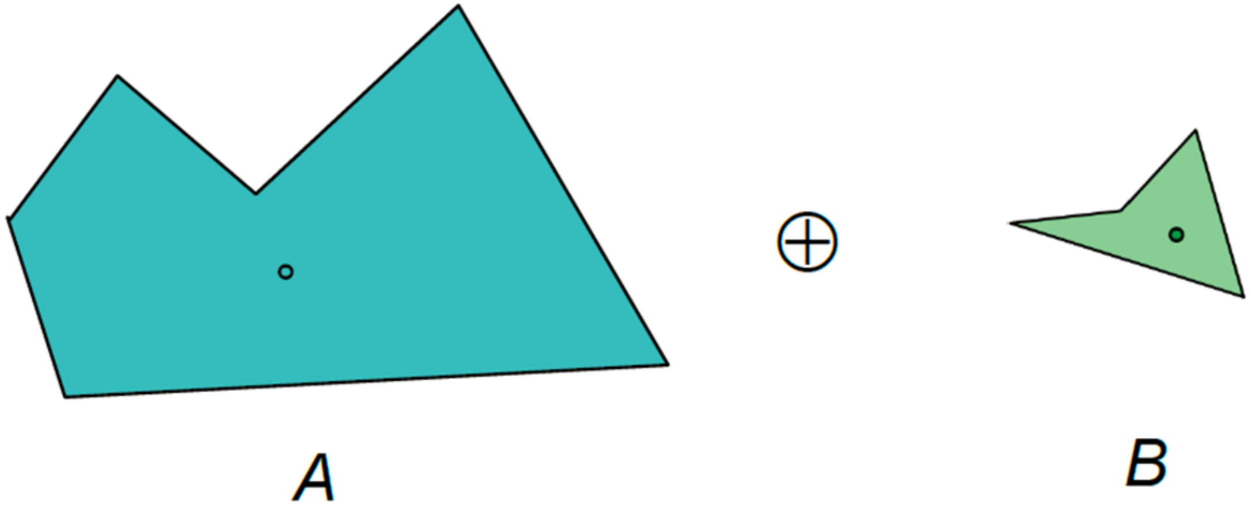
Example – 2D Robot without Rotation

Workspace
(2D)

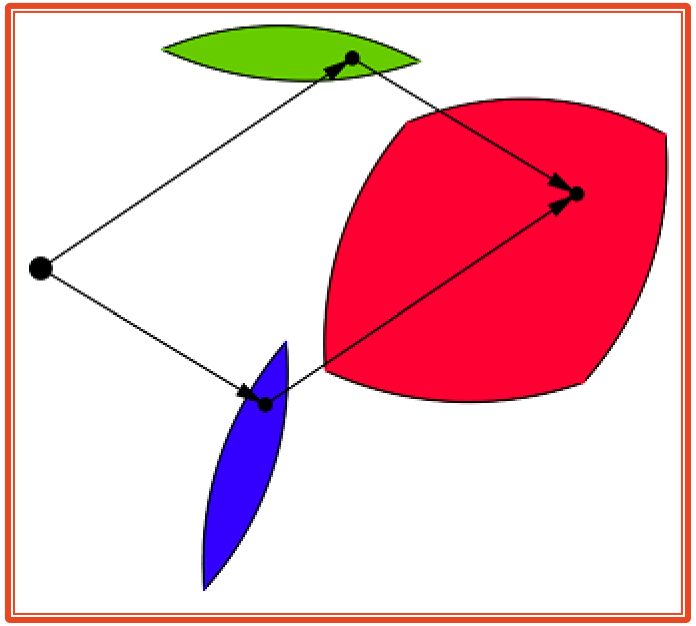


configuration space
(2D)

Minkowski Sum

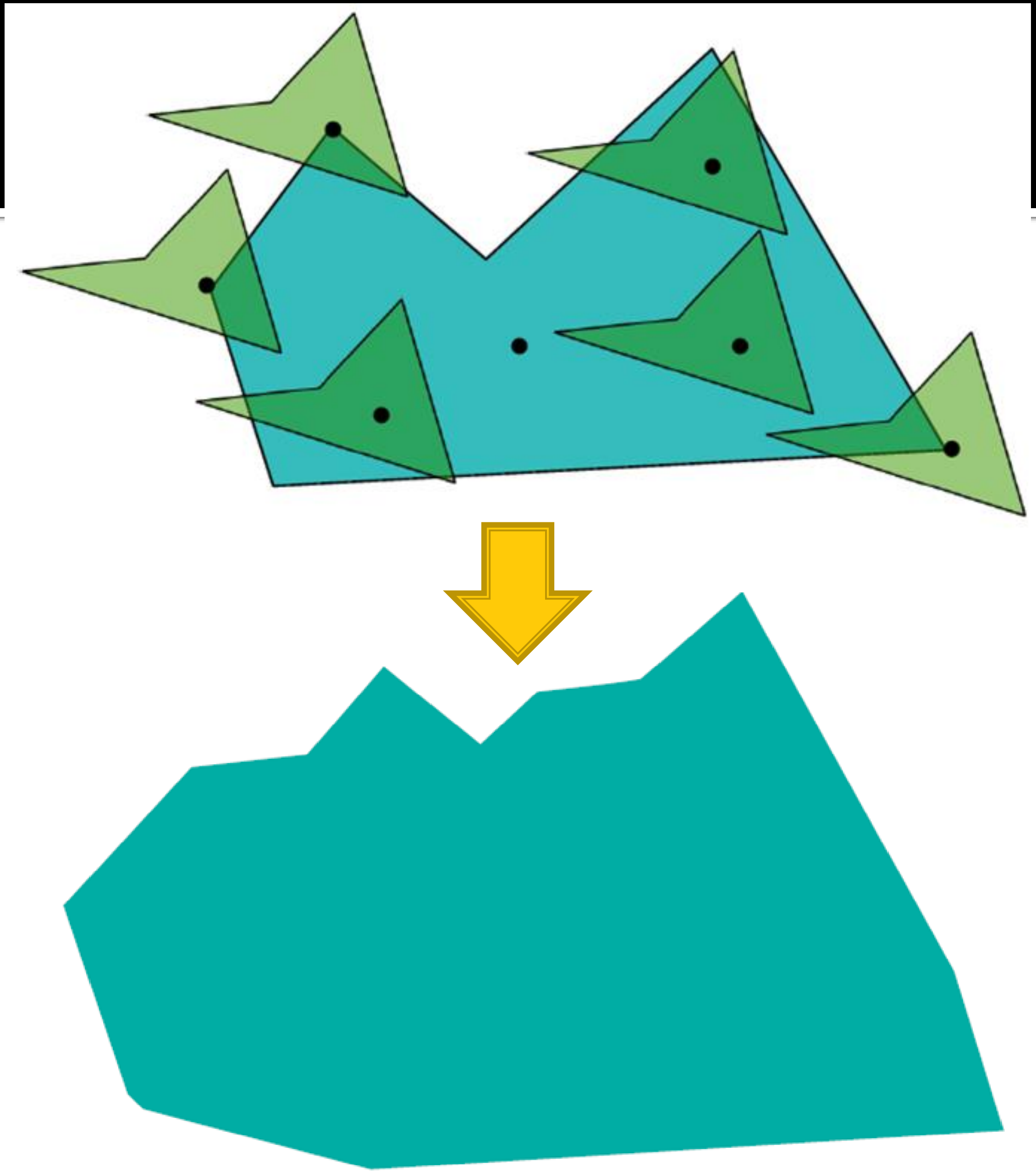


$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$



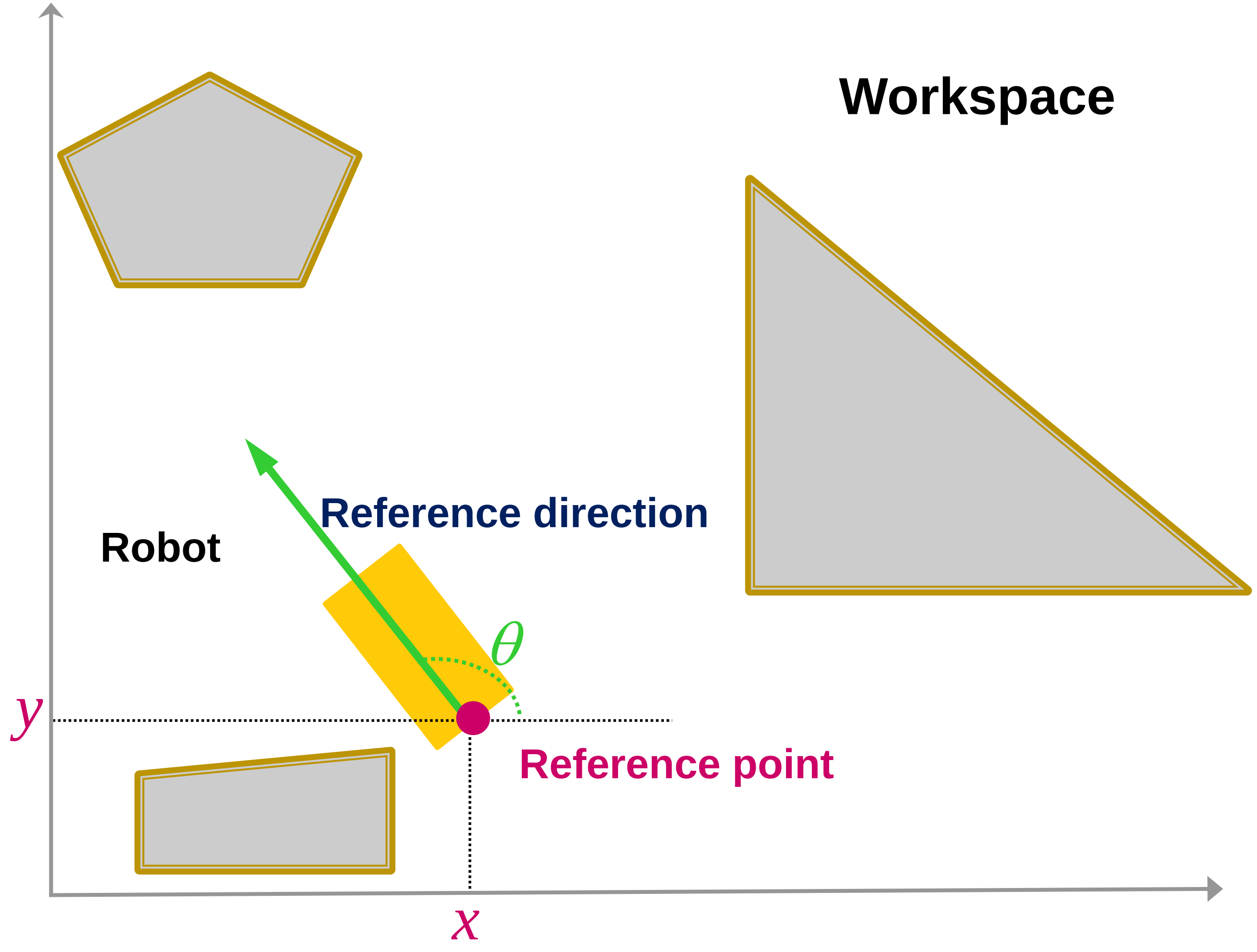
Minkowski Sum

- Dip B into paint
- Put B's origin on A's border
- Translate it along A's edge
- Sum = the painted area

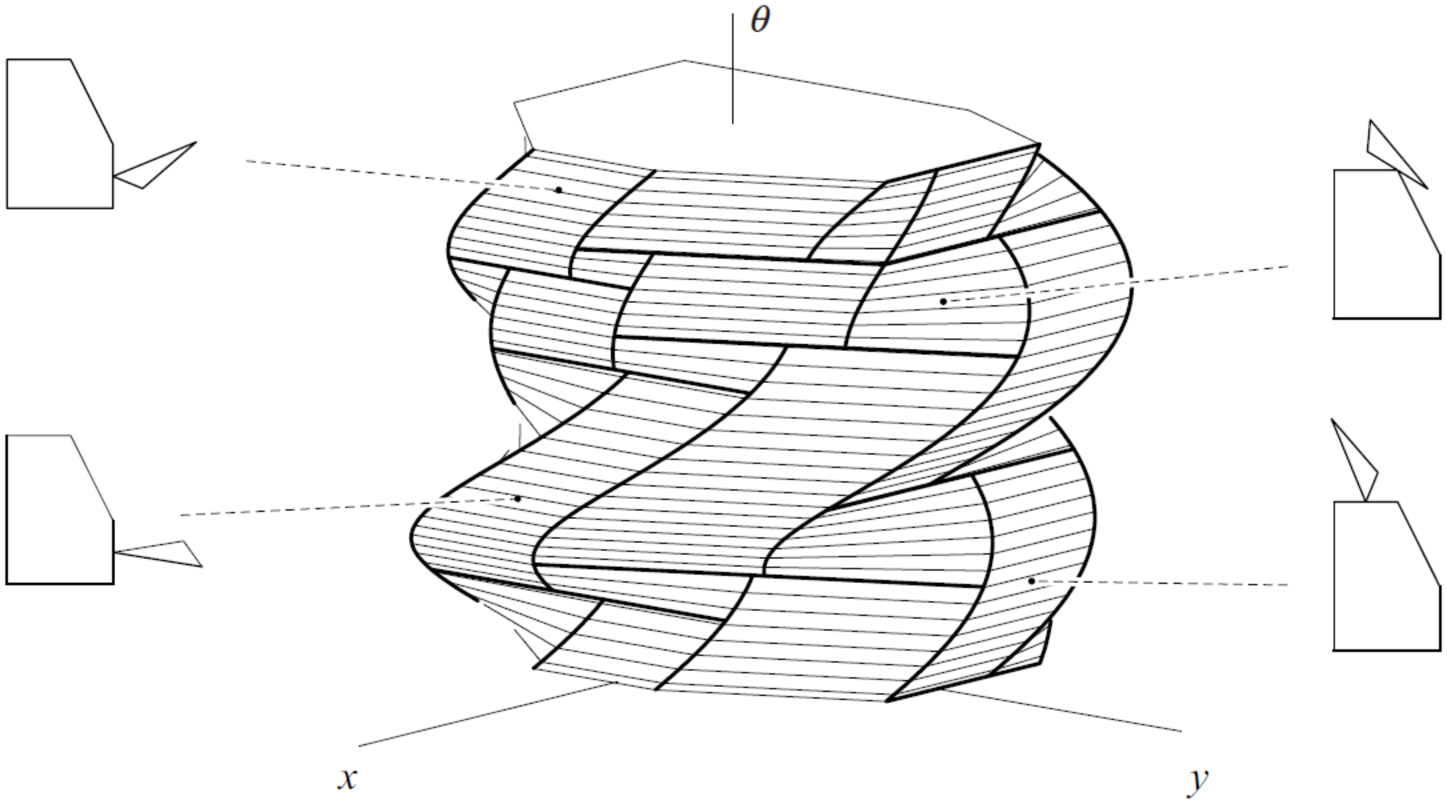


Example – 2D Robot with Rotation

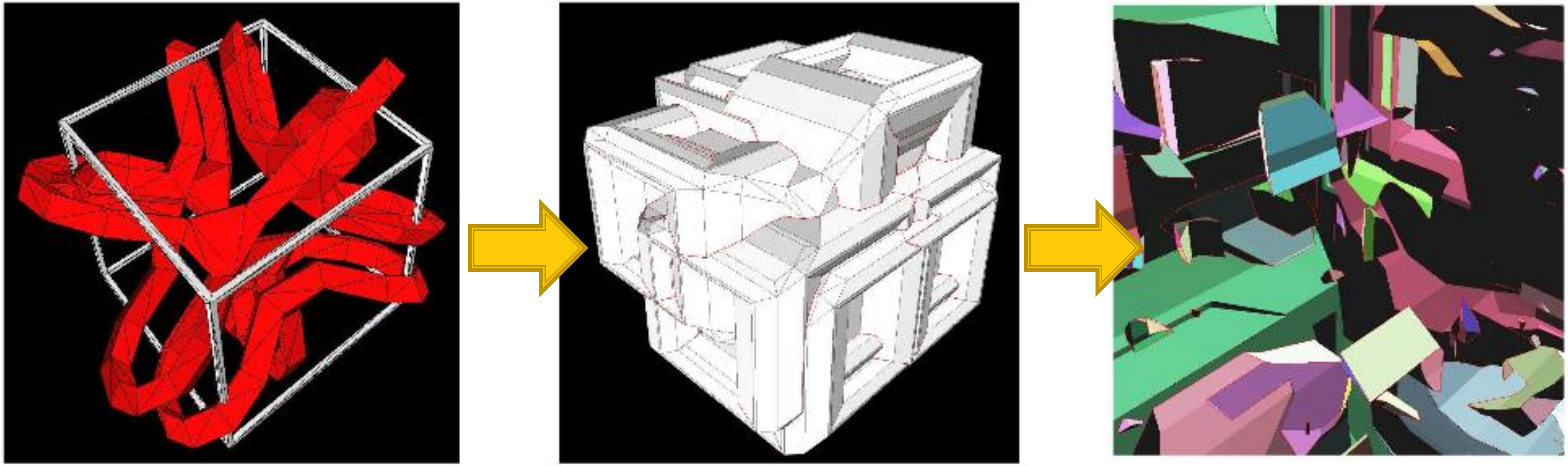
- C-space?
- Minkowski Sum?



Example – 2D Robot with Rotation



High-dimensional space



Discussion

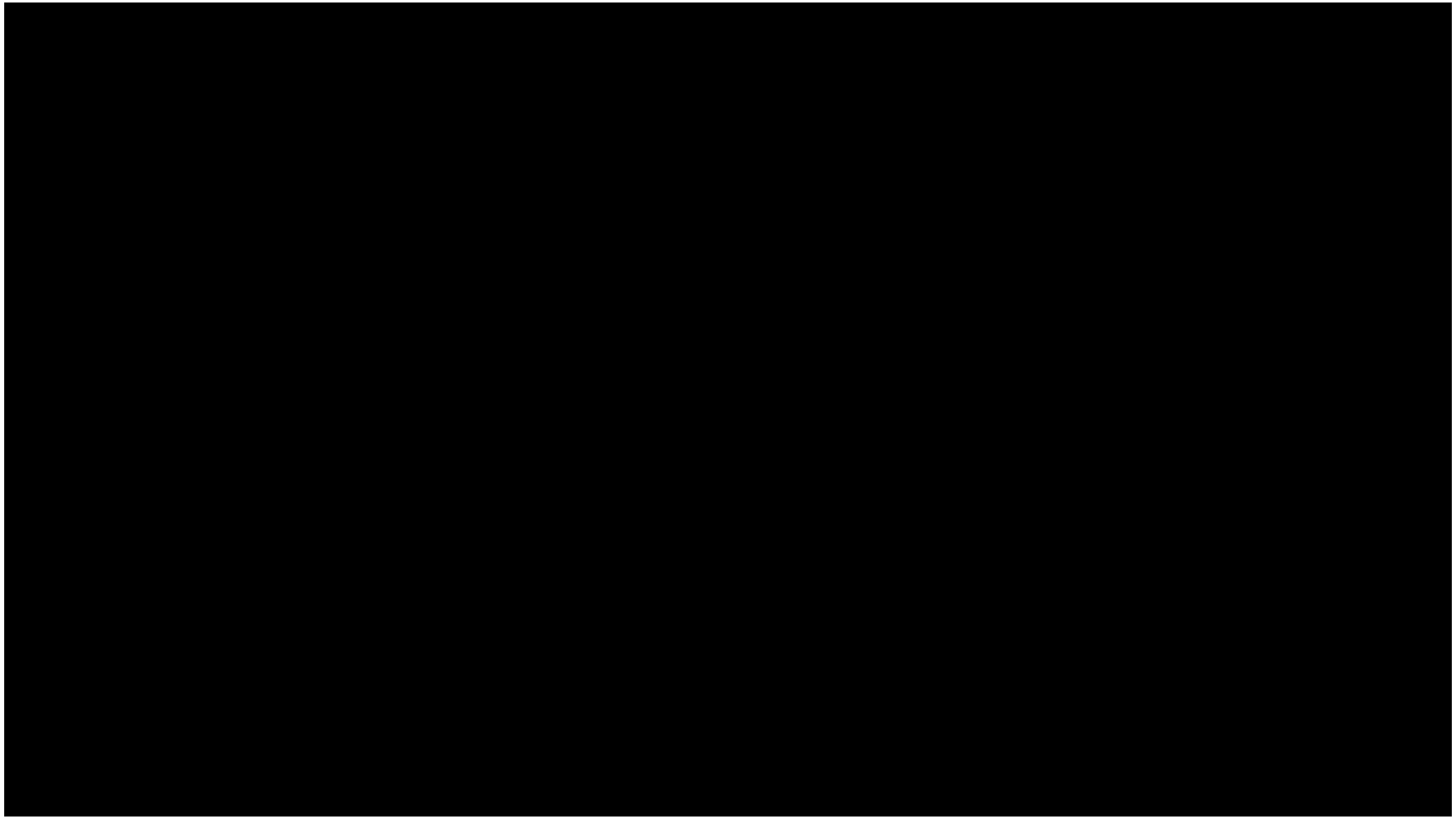
- Do we need to have an explicit representation of C-obstacles to do path planning?
  - Exact method?
  - Approximate method?
  - Sampling-based method?

Topology of C-Space

- Why topology matters?

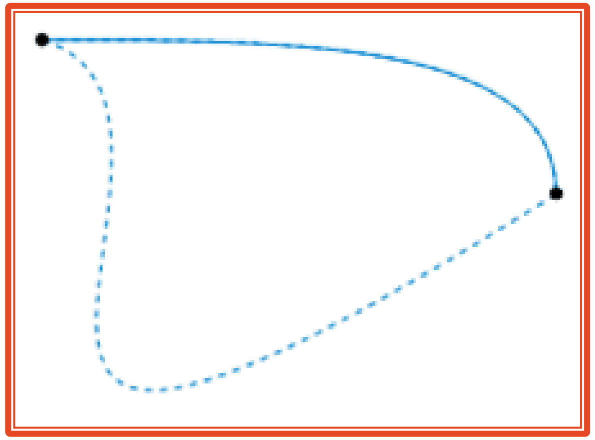
Because coffee mug is indeed a donut!

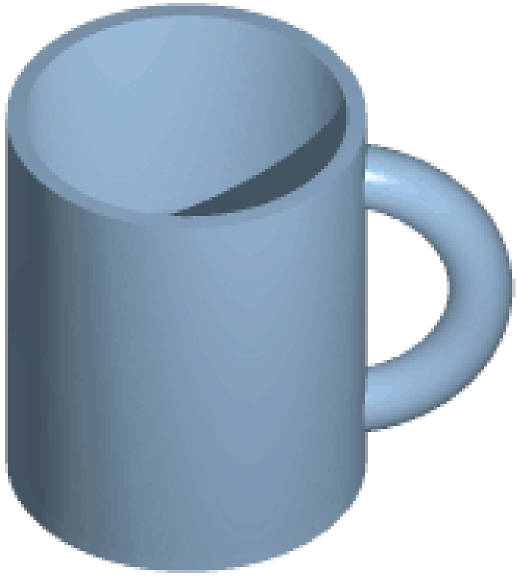
Topological properties are very useful



Homotopic paths

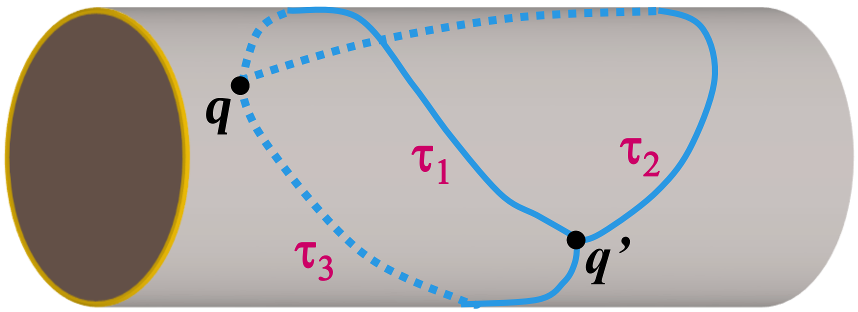
- Two paths with the same endpoints is **homotopic** if one path can be deformed into continuously deformed into the other



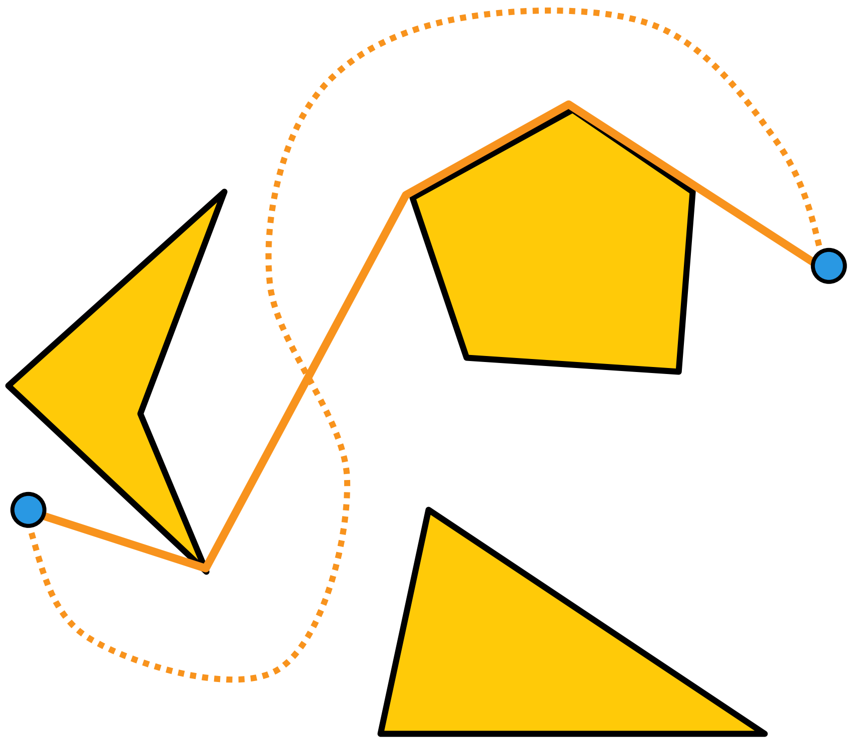


Homotopic class of paths

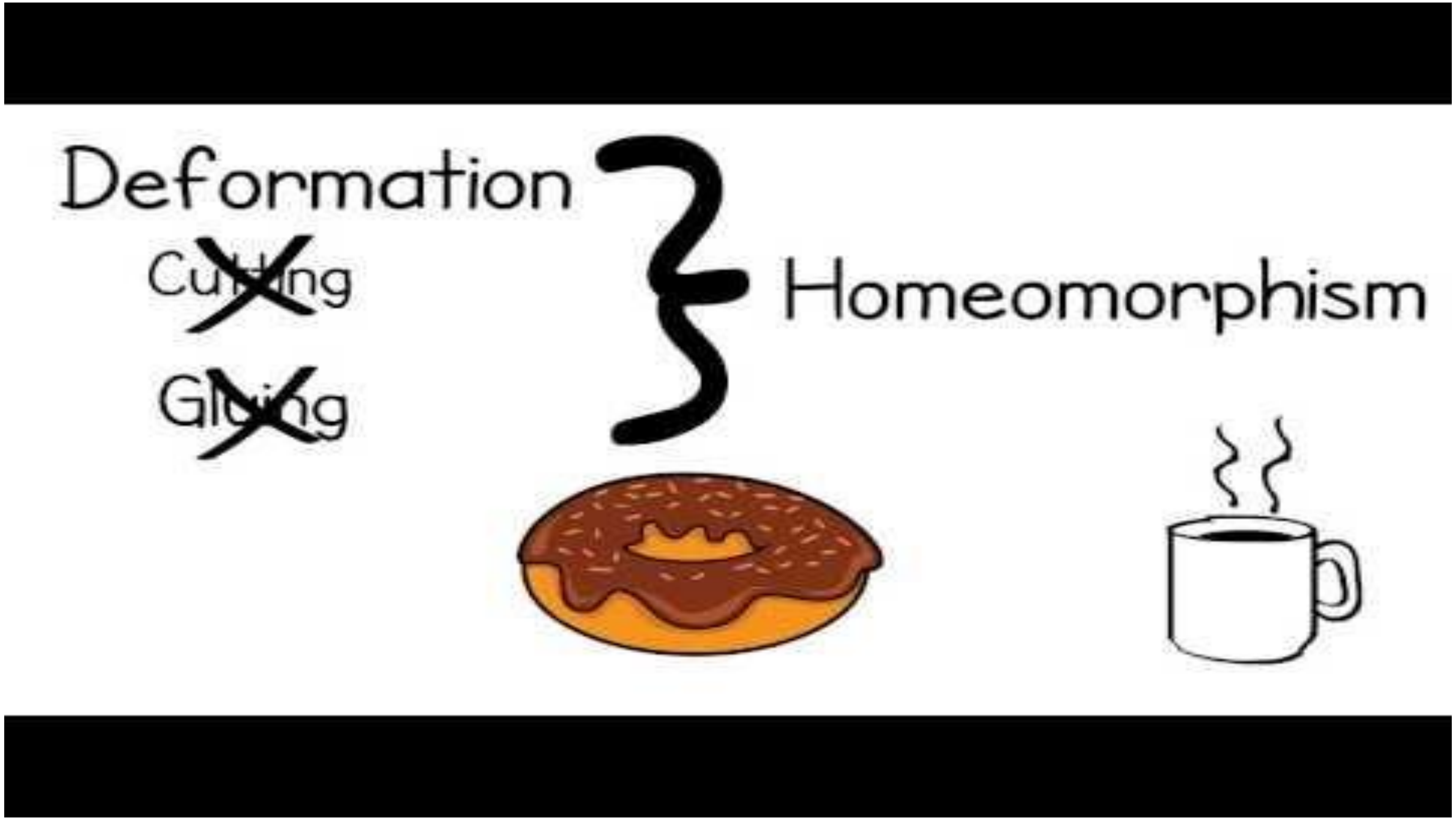
- On a cylinder surface without ends



Which paths are homotopic?



Topology and homeomorphism



Connectedness of C-Space

- C is **connected**
  - If every two configurations can be connected by a path.
- C is **simply-connected**
  - if any two paths connecting the **same** endpoints are **homotopic**.
  - Examples: R^2 or R^3
- Otherwise C is multiply-connected.
  - Example?

Distance in C-space

- A distance function ***d*** in configuration space **C** is a function

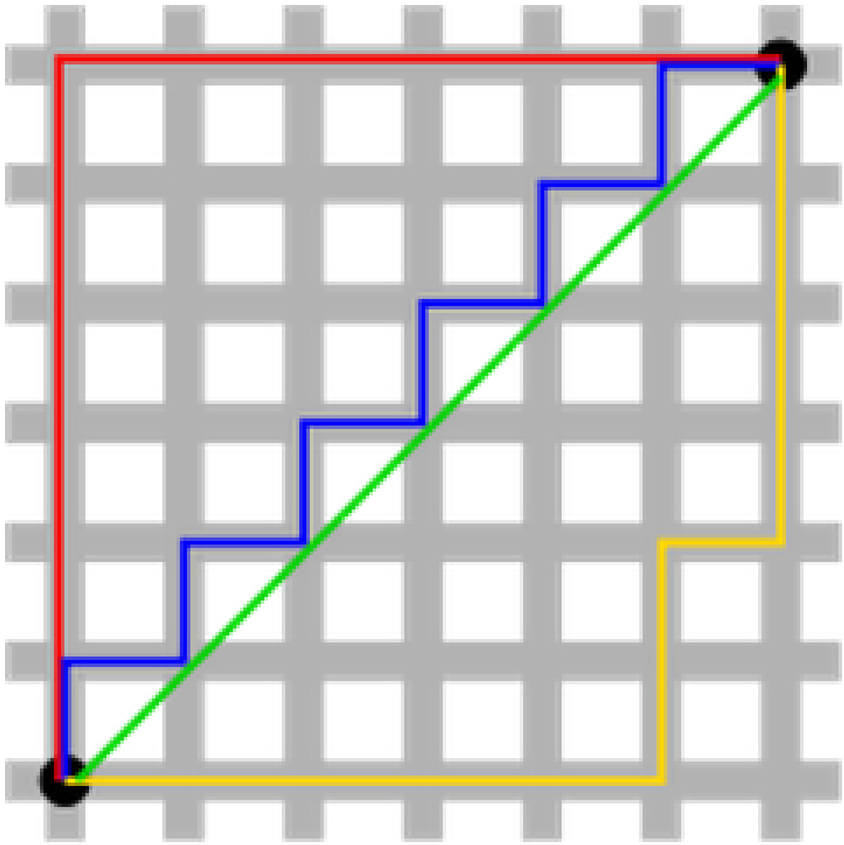
$$d : (q, q') \in C^2 \rightarrow d(q, q') \geq 0$$

- $d(q, q') = 0$ if and only if $q = q'$,
- $d(q, q') = d(q', q)$,
- $d(q, q') <= d(q, q'') + d(q'', q')$

Discussion

- Do we need a specialized distance metric in C-space to do path planning?
- Metrics for distance?
  - Euclidian distance
  - Other metrics?

Distance in C-space



	a	b	c	d	e	f	g	h	
8	5	4	3	2	2	2	2	2	8
7	5	4	3	2	1	1	1	2	7
6	5	4	3	2	1		1	2	6
5	5	4	3	2	1	1	1	2	5
4	5	4	3	2	2	2	2	2	4
3	5	4	3	3	3	3	3	3	3
2	5	4	4	4	4	4	4	4	2
1	5	5	5	5	5	5	5	5	1
	a	b	c	d	e	f	g	h	

Distance metrics

- L1-norm (Manhattan distance)

$$d_1(\mathbf{p},\mathbf{q}) = \|\mathbf{p}-\mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|,$$

- L2-norm (Euclidian distance)

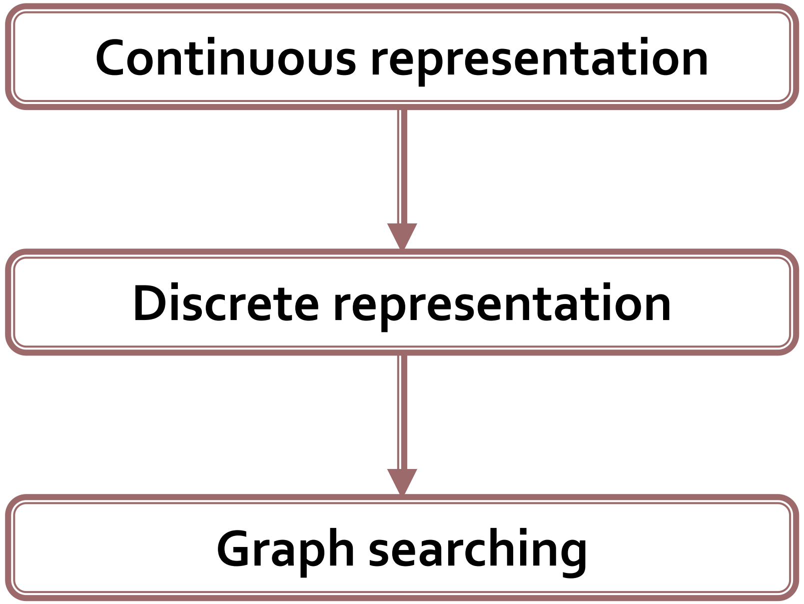
$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2}$$

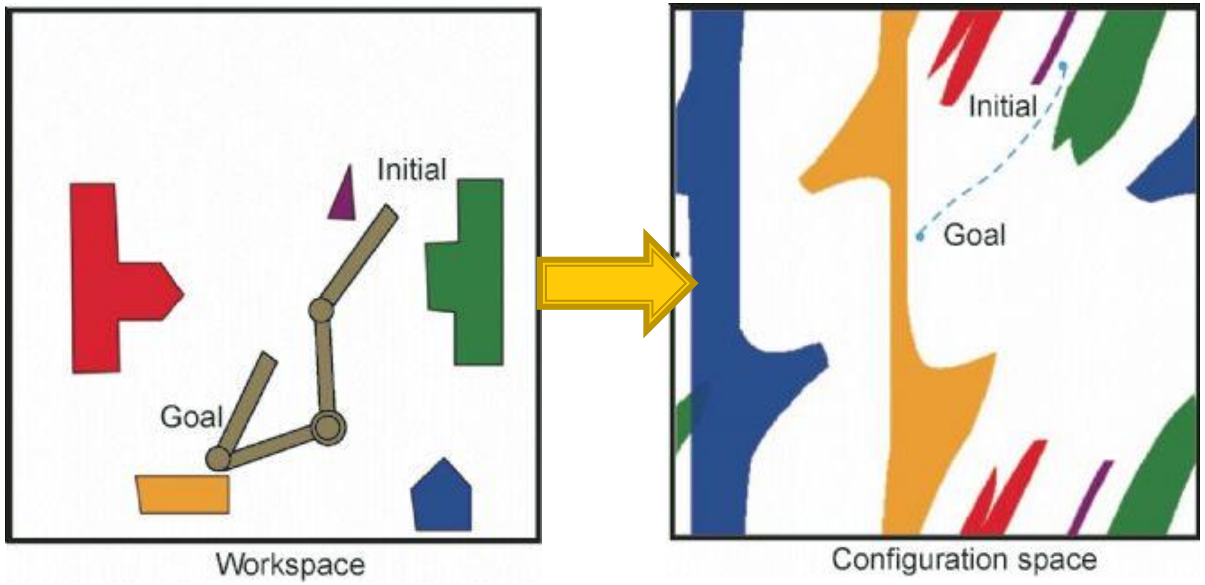
- L_∞-norm (chessboard distance)

$$D_{\mathrm{Chebyshev}}(p,q) := \max_i(|p_i - q_i|).$$

Reading Assignment

- Once specified a C-space and its obstacle, we should be able to discretize it and search for a path

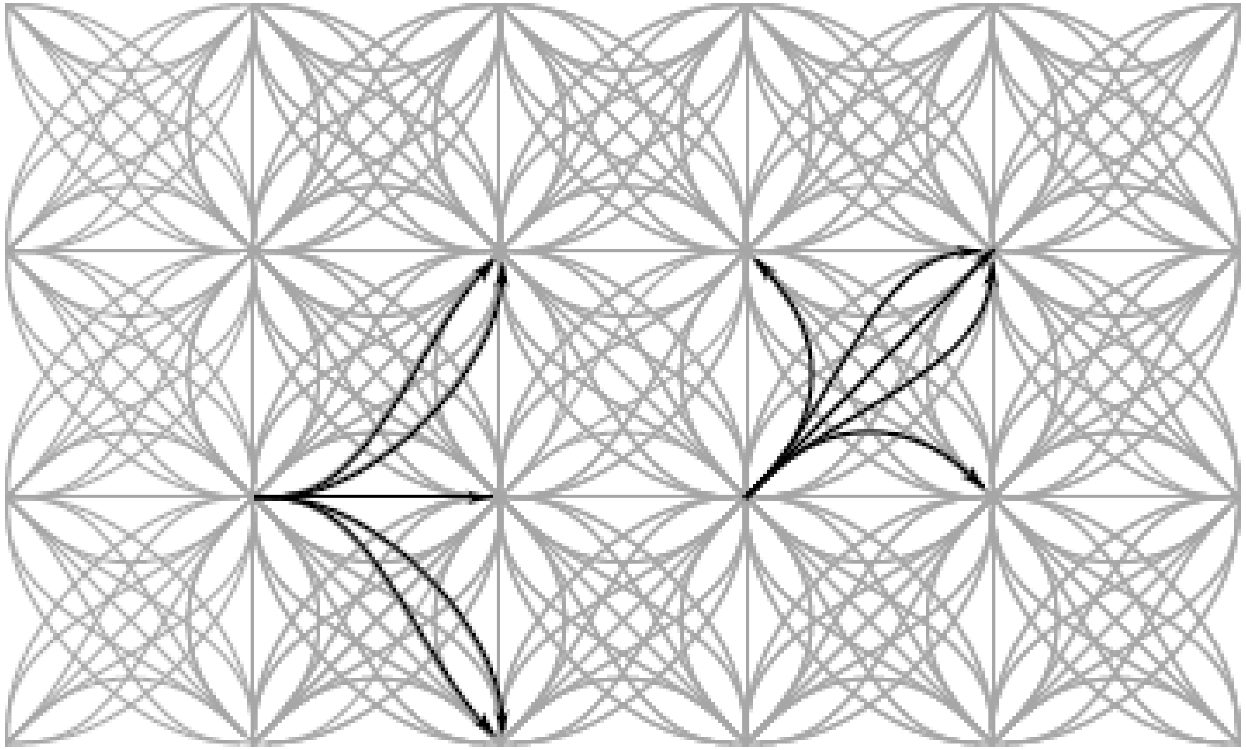


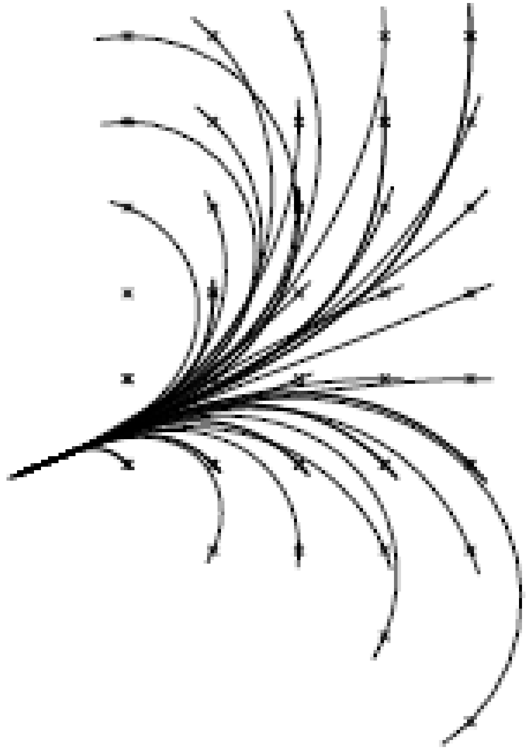


How about the C-space of a self-driving car?

- Discretization
  - Exact method, approximate method?
- Search
  - Can we get from one cell to another, directly? Why?
- How does the C-space look like?

State lattice





How to handle moving obstacles?

Assignment – individual paper review

- Paper
  - Ziegler, J., & Stiller, C. (2009, October). Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios. In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009. (pp. 1879-1884)
- Topic
  - Spatiotemporal state lattice → Due on Friday (Feb 2) at noon
  - Present student talk on Friday? → Submit by Thursday (Feb 1) by mid-night

Student talk – James Kuszmaul
Complexity of sweeping-line algorithm

End