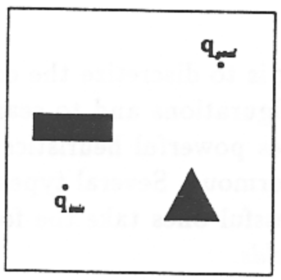
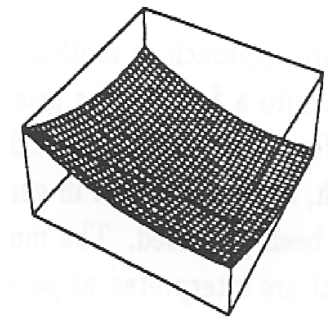
# Configuration Space

Jane Li
Assistant Professor
Mechanical Engineering & Robotics Engineering
http://users.wpi.edu/~zli11

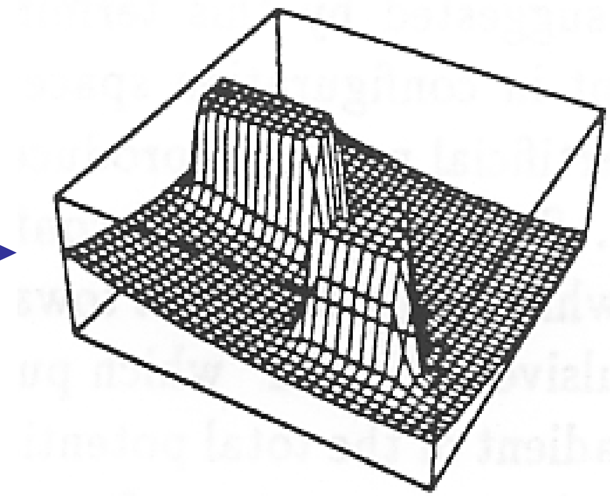
# Potential field

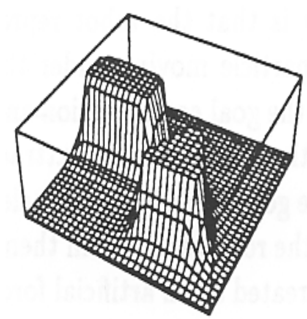
$$\phi_{att} = \frac{1}{2} k_{att} (x - x_{goal})^2$$





+





$$\phi_{rep} = \begin{cases} \frac{1}{2} k_{rep} \left( \frac{1}{\rho} - \frac{1}{\rho_0} \right)^2 & \text{if } \rho \le \rho_0, \\ 0 & \text{if } \rho > \rho_0 \end{cases}$$

$\boldsymbol{k_{att}}, \boldsymbol{k_{rep}}$ : positive scaling factors

$\boldsymbol{x}$ : position of the robot

$\boldsymbol{\rho}$ : distance to the obstacle

$\boldsymbol{\rho_0}$ : distance of influence

# Attractive & repulsive fields

$$F_{\text{att}} = -\nabla \phi_{\text{att}} = -k_{\text{att}}(x - x_{\text{goal}})$$

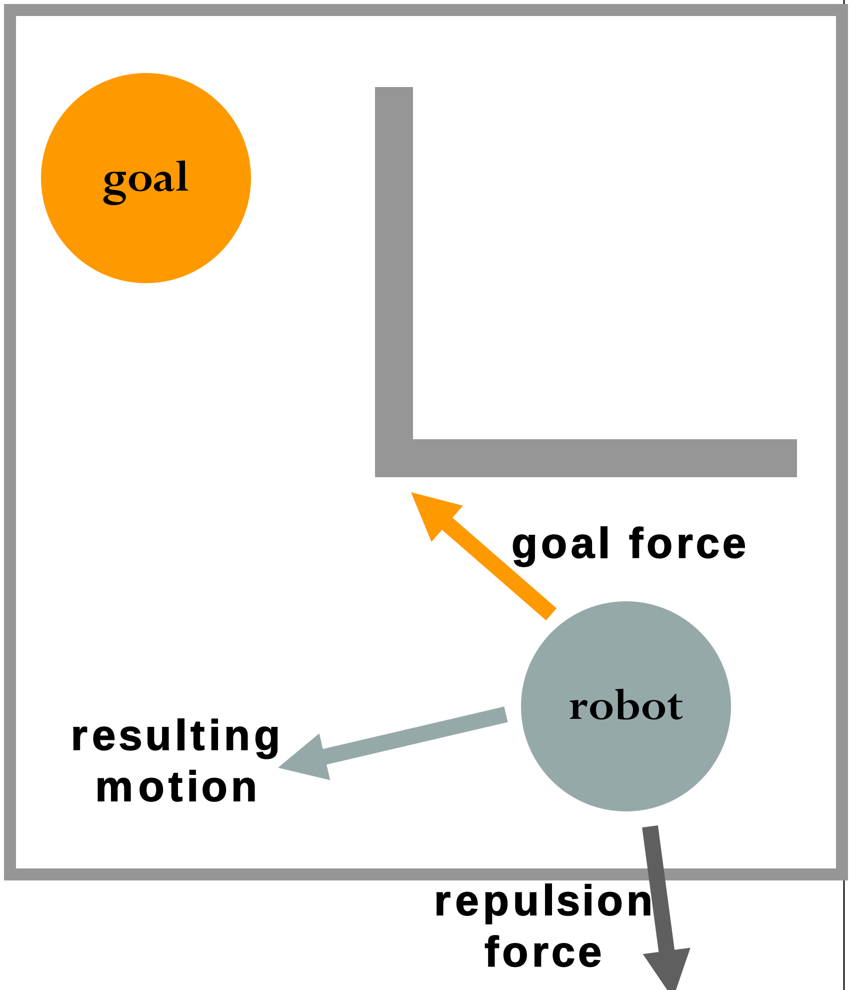
$$F_{\text{rep}} = -\nabla \phi_{\text{rep}} = \begin{cases} k_{\text{rep}} \left( \frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} \frac{\partial \rho}{\partial x} & \text{if } \rho \le \rho_0, \\ 0 & \text{if } \rho > \rho_0 \end{cases}$$

$k_{\text{att}}$, $k_{\text{rep}}$ : positive scaling factors

$x$ : position of the robot

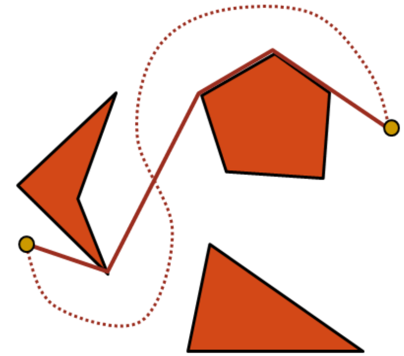
$\rho$ : distance to the obstacle

$\rho_0$ : distance of influence

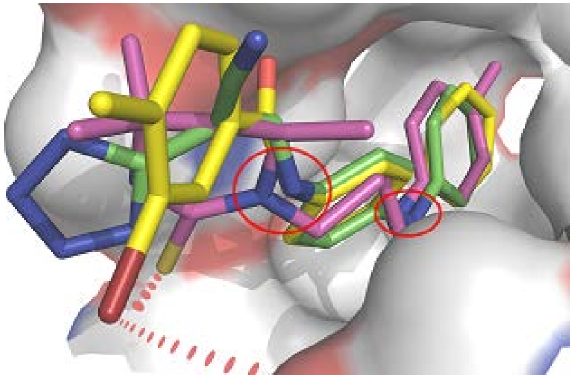



[Khatib, 1986]

# Recap



- We learned about how to plan paths for a point



- Real-world robots are **complex**, often **articulated** bodies
  - What if we invented a **space** where the robots could be treated as **points**?

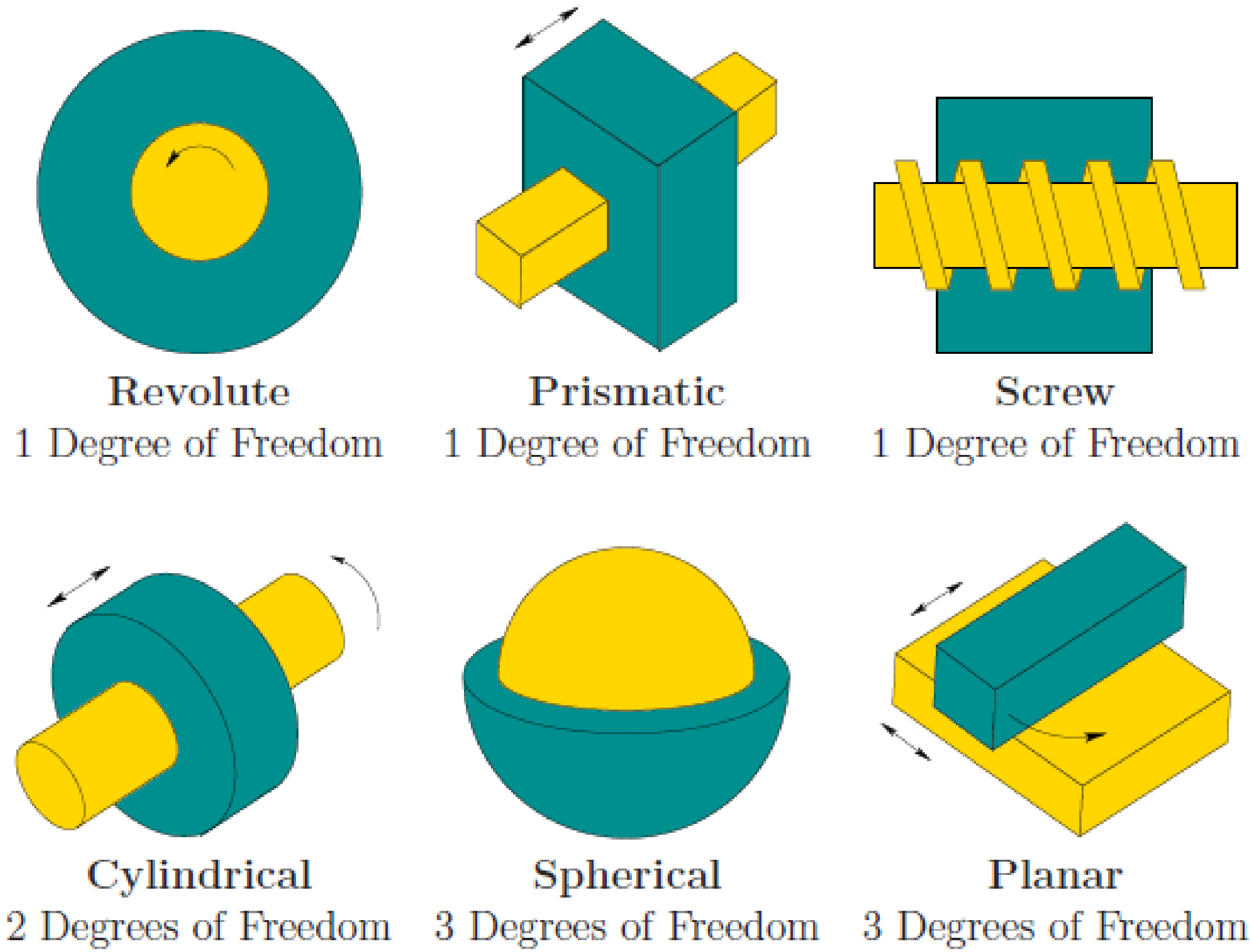
# Definition

- **Configuration** – a specification of the position of **every** point on the object.
  - A configuration q is usually expressed as a vector of the **Degrees of Freedom (DOF)** of the robot

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

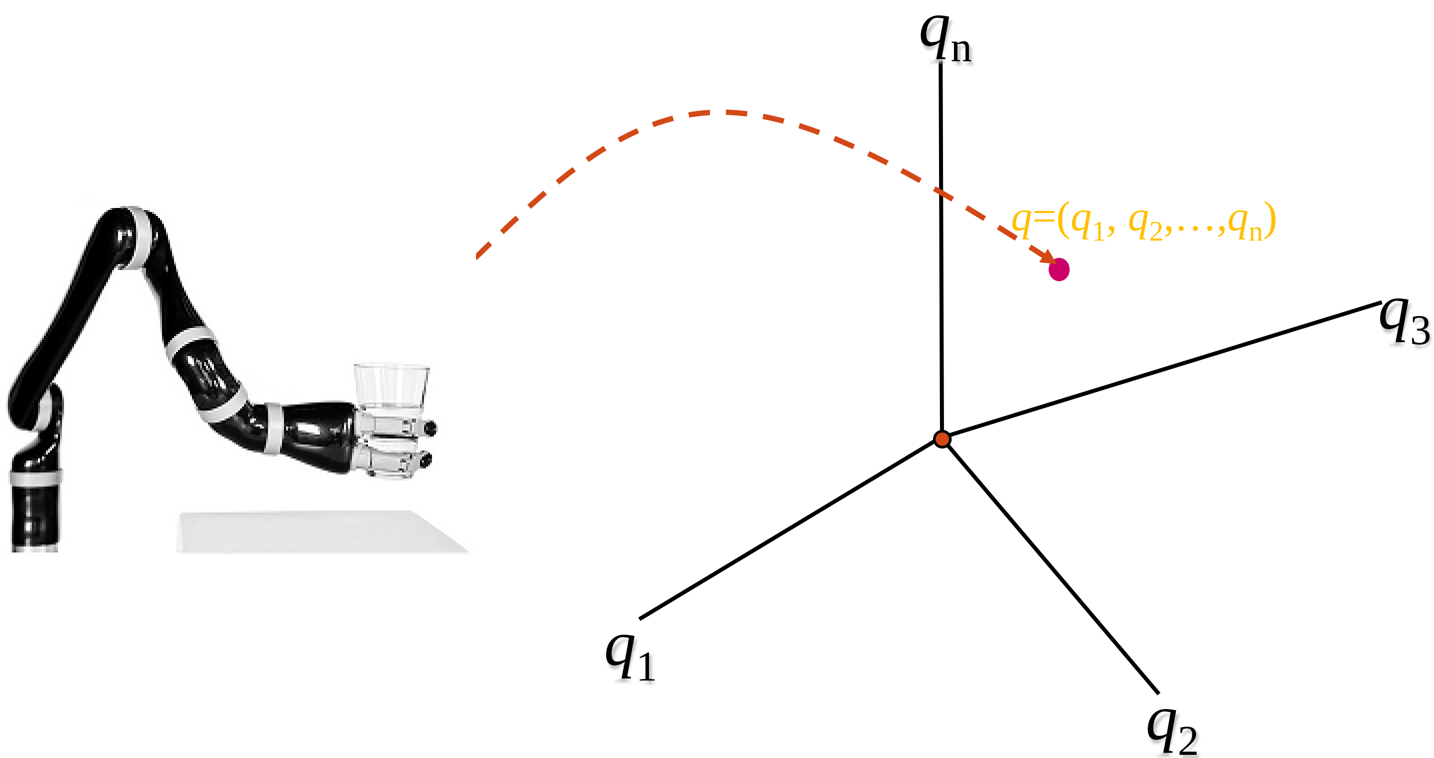
- **Configuration space C** – the set of all possible configurations.
  - A configuration $q$ is a point in $C$

# Degree of Freedom – Examples



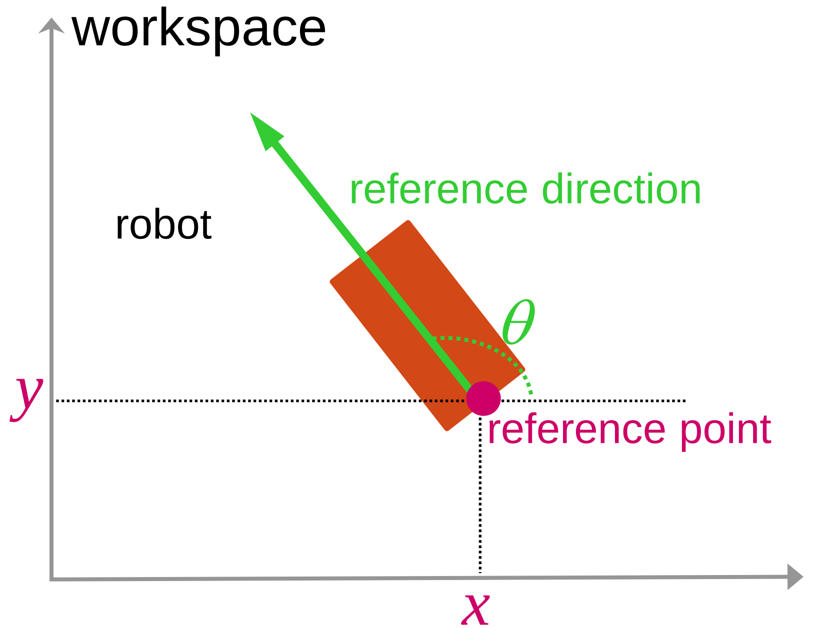
# Dimension of Configuration Space

- Dimension of a Configuration Space
  - The **minimum** number of DOF needed to specify the configuration of the object completely.

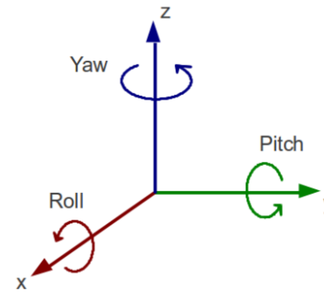


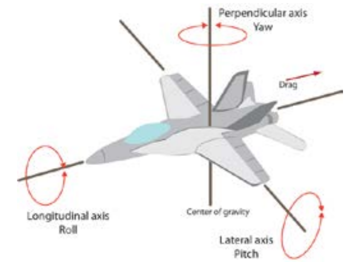
# Example – A Rigid 2D Mobile Robot

- 3-parameters: $q = (x, y, \theta)$ with $\theta \in [0, 2\pi)$.
  - 3D configuration space
  - Topology: $SE(2) = R^2 \times S^1$ (a 3D 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

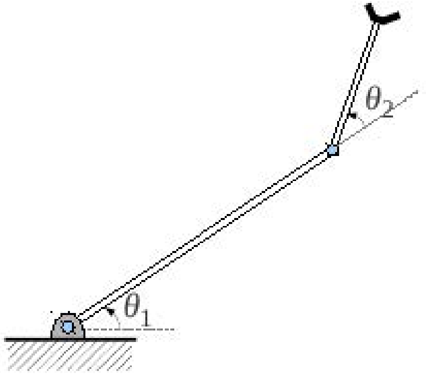
$$R_z(\varphi) = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

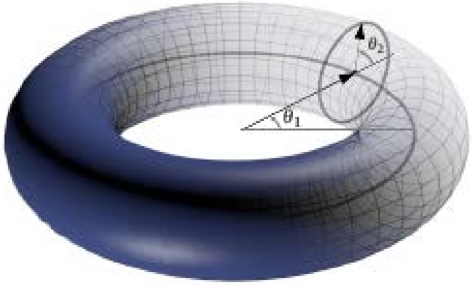
$$\mathbf{q} = \cos\frac{\theta}{2} + (u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k})\sin\frac{\theta}{2}$$

- C-space dimension = **6**
- Topology: SE(3) = $R^3$ x 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?
- Topology of two-link manipulator?
  - With joint limits?



# Paths and Trajectories in C-Space

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

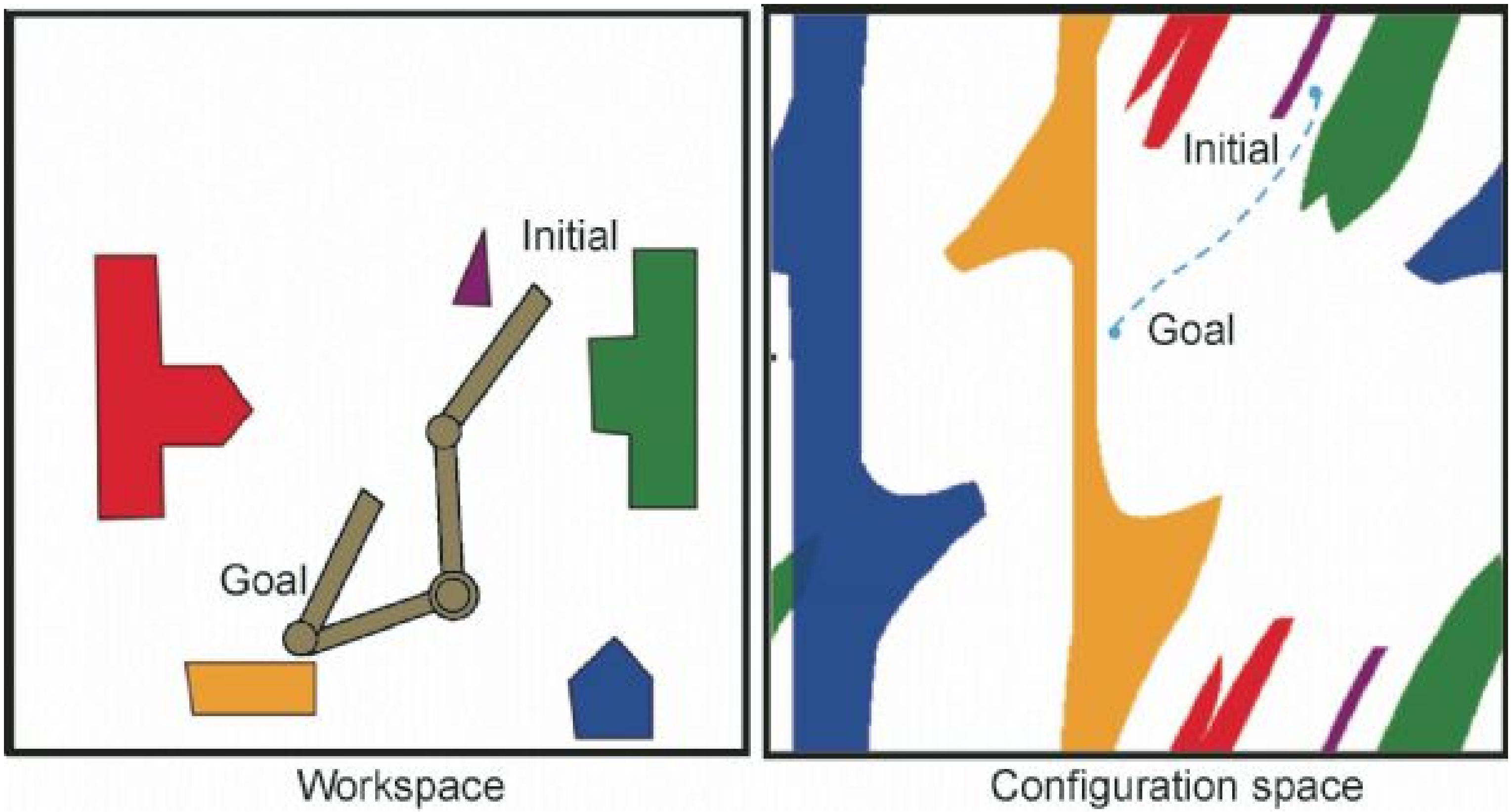
$$\tau : s \in [0,1] \rightarrow \tau(s) \in C$$

Such that $\tau(0) = q_{start}$ and $\tau(1) = q_{goal}$.

- Trajectory
  - A path parameterized by time

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

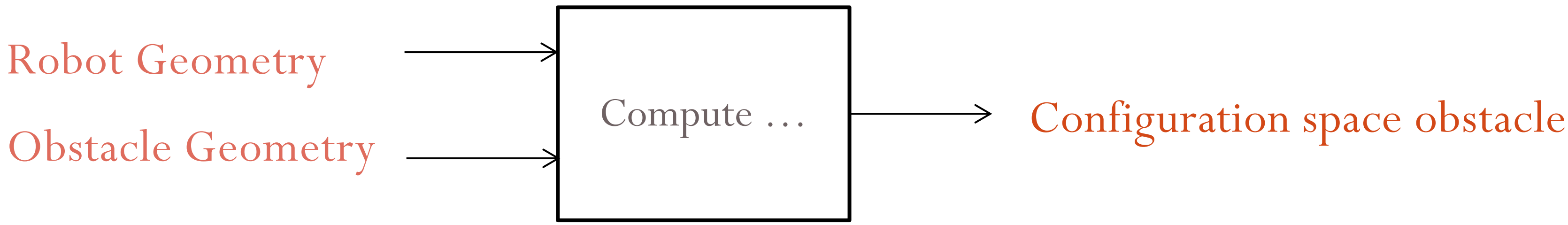
# Obstacles in C-space



# Obstacles in C-space

- (Collision)-free configuration – $q$
  - Robot placed at $q$ 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** (self-collision)

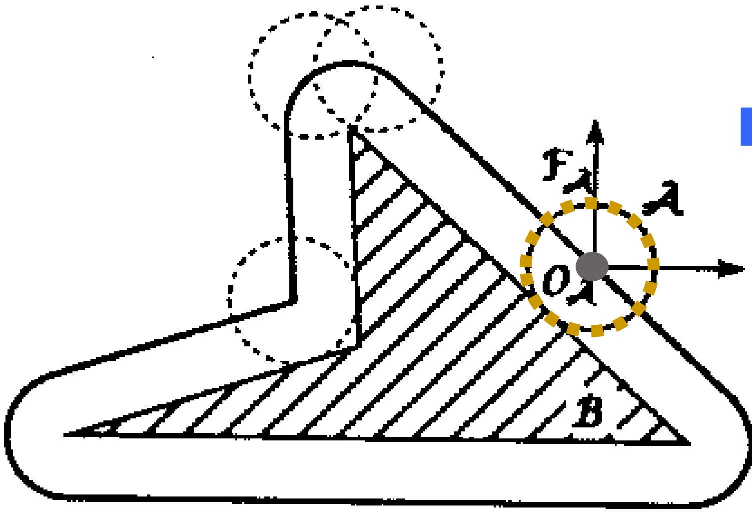
# How to compute $C_{obs}$ ?




- A simple example
  - 2D translating robot
  - Polygonal obstacle in task space

# Example – Disc in 2D workspace

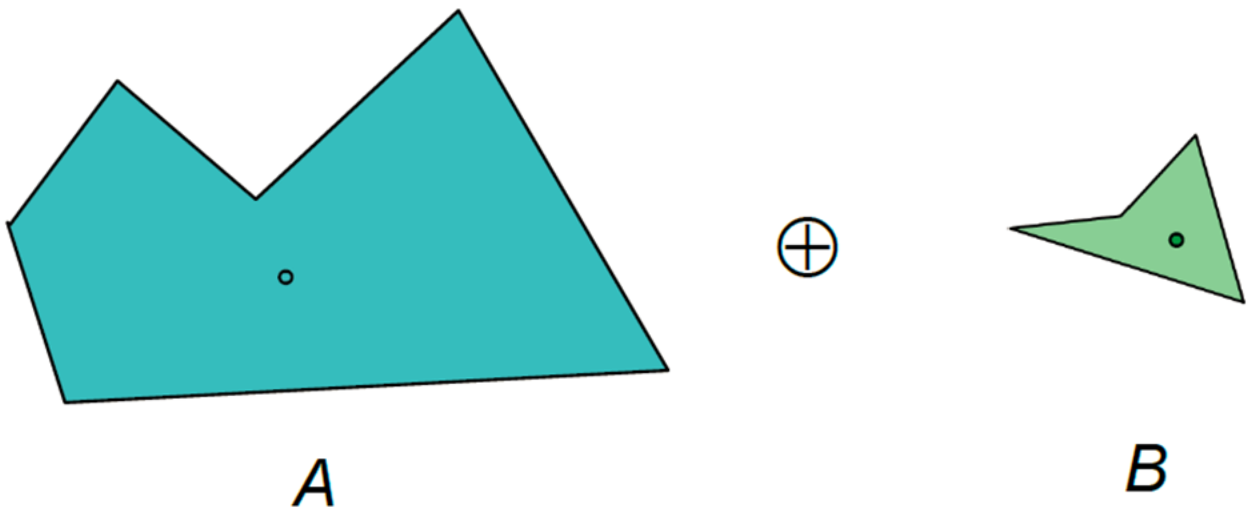
**Workspace (2D)**




**configuration space (2D)**

# Minkowski Sum

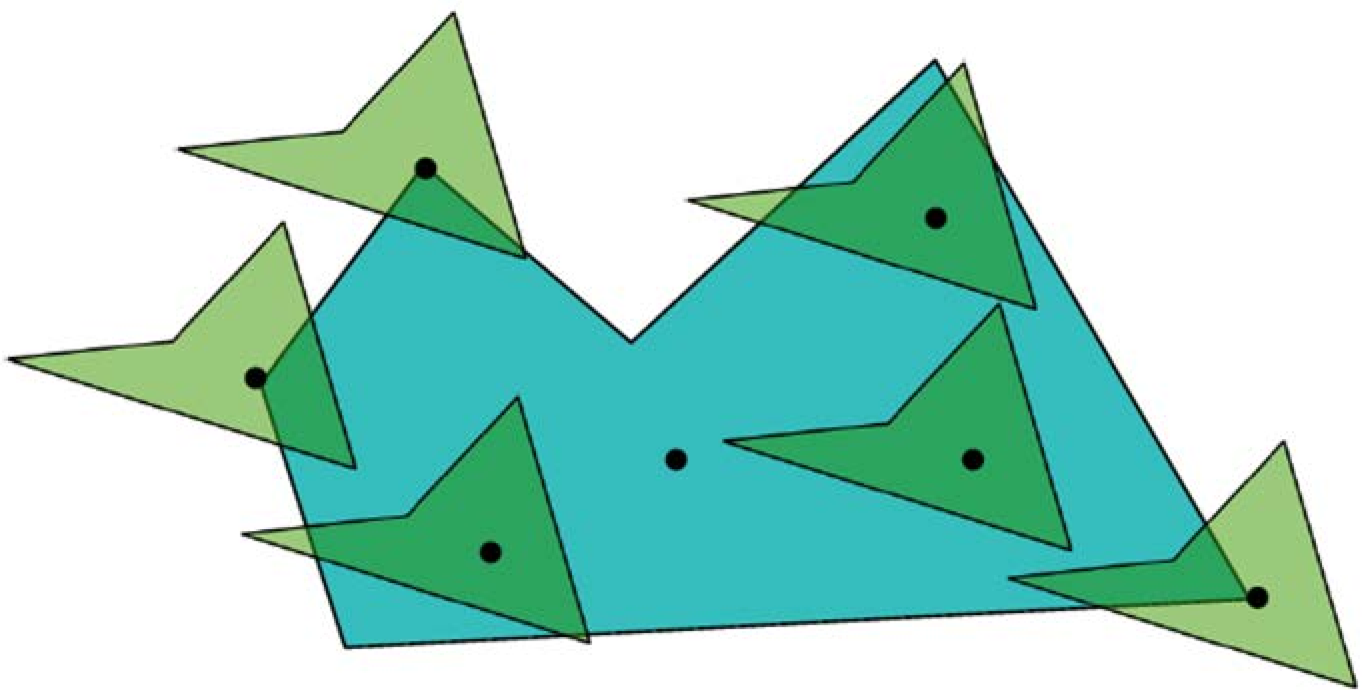
$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$



Modified based on Slides
by Prof. David Hsu, University of Singapore

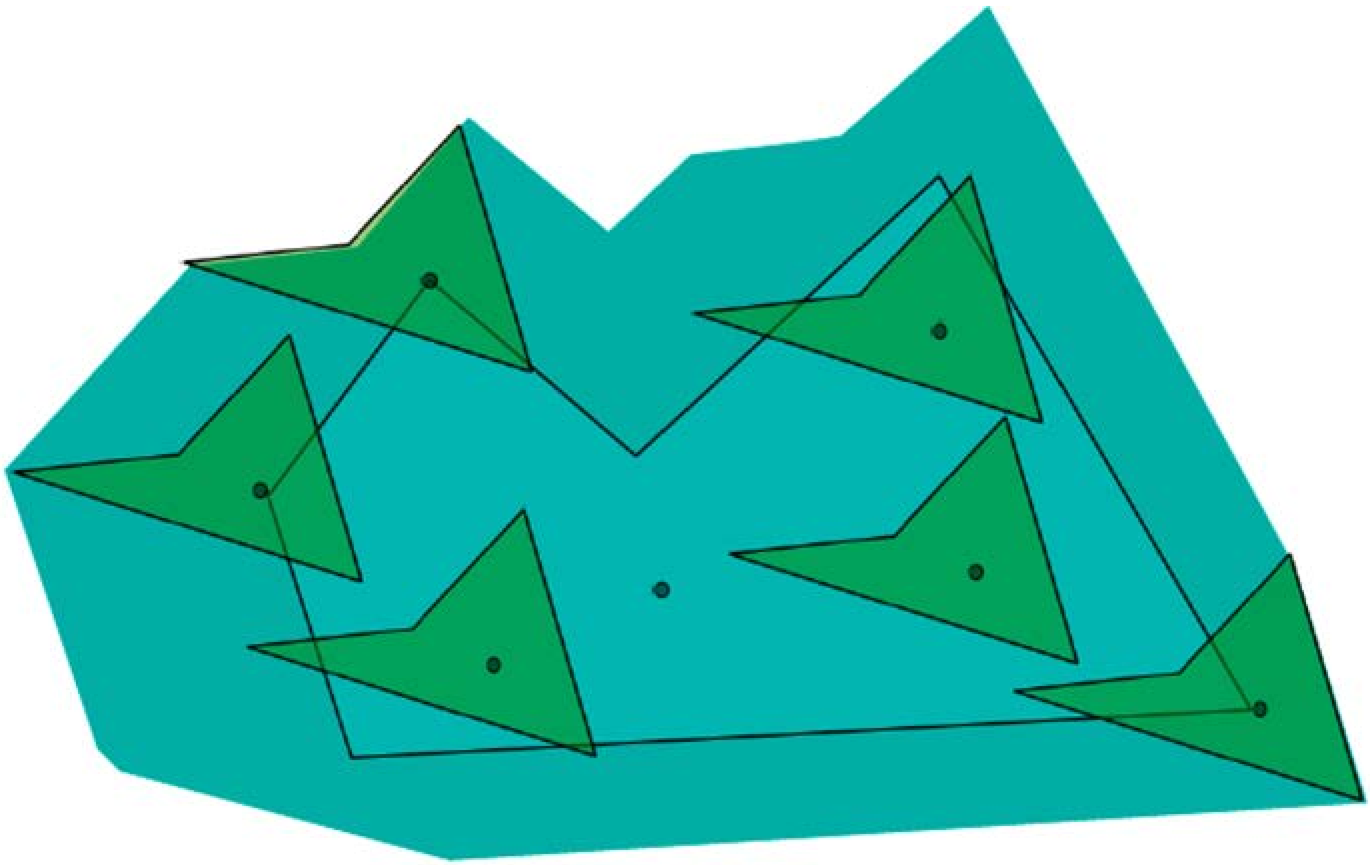
# Minkowski Sum

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



Modified based on Slides
by Prof. David Hsu, University of Singapore

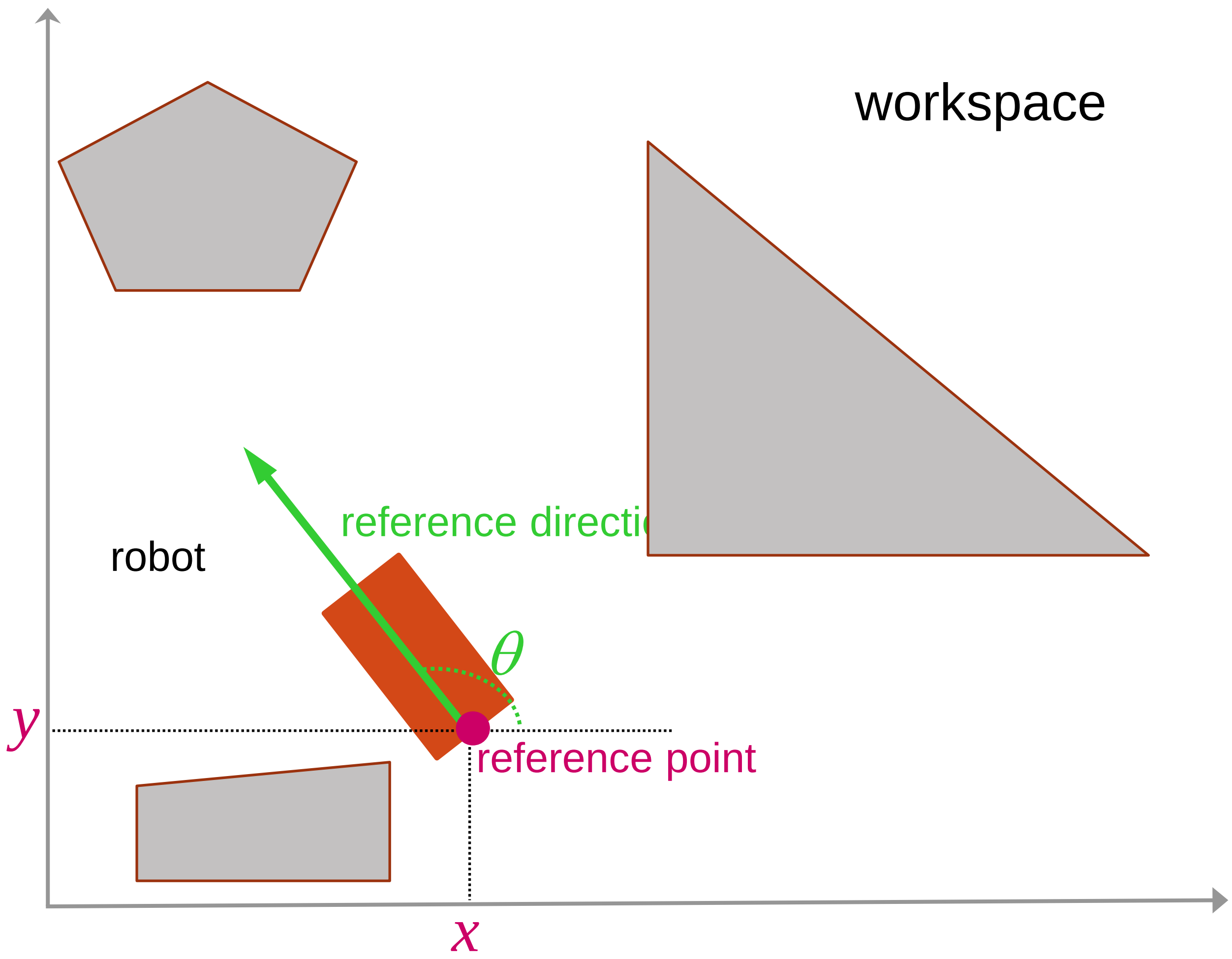
# Minkowski Sum



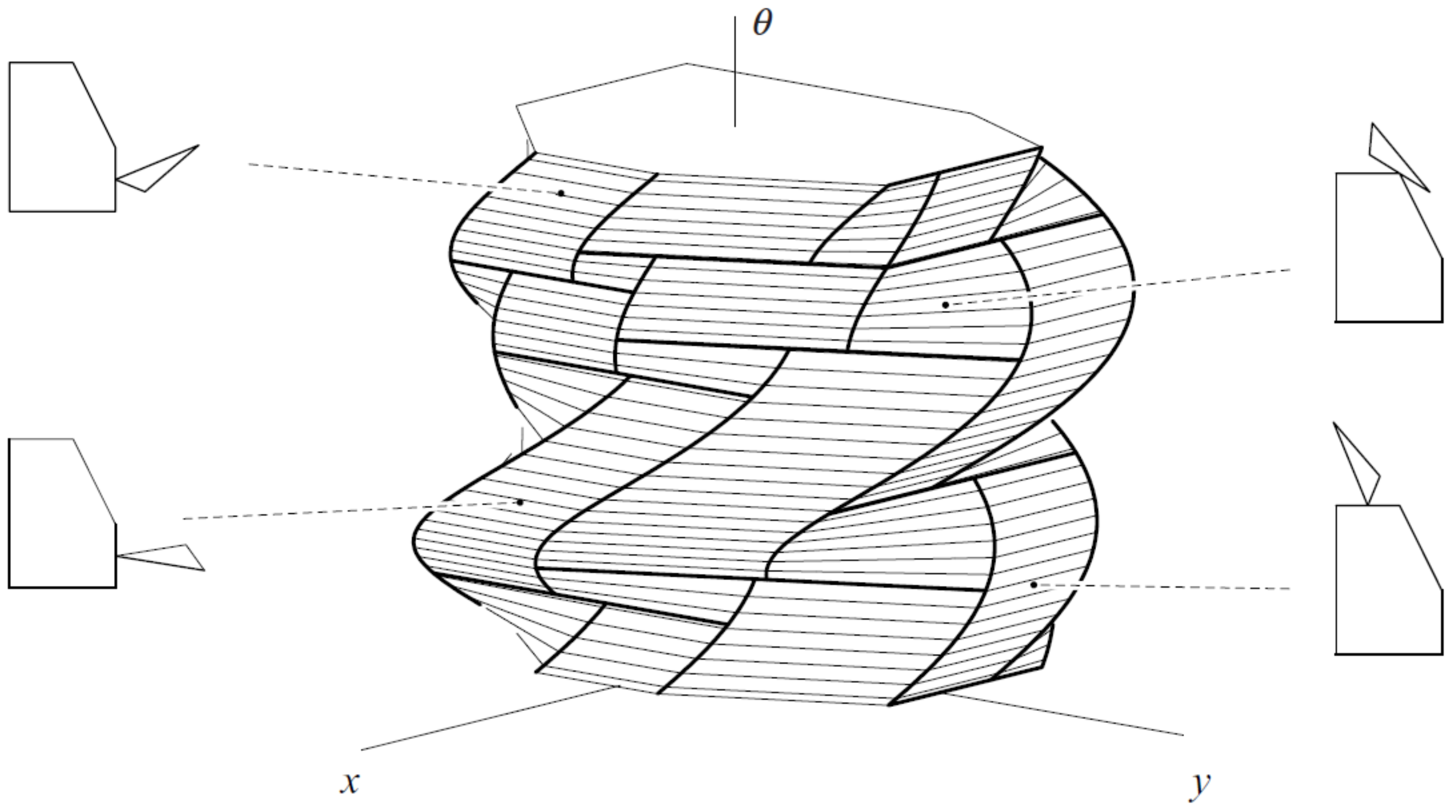
# Minkowski Sum

Modified based on Slides
by Prof. David Hsu, University of Singapore

# Example – 2D Robot with Rotation



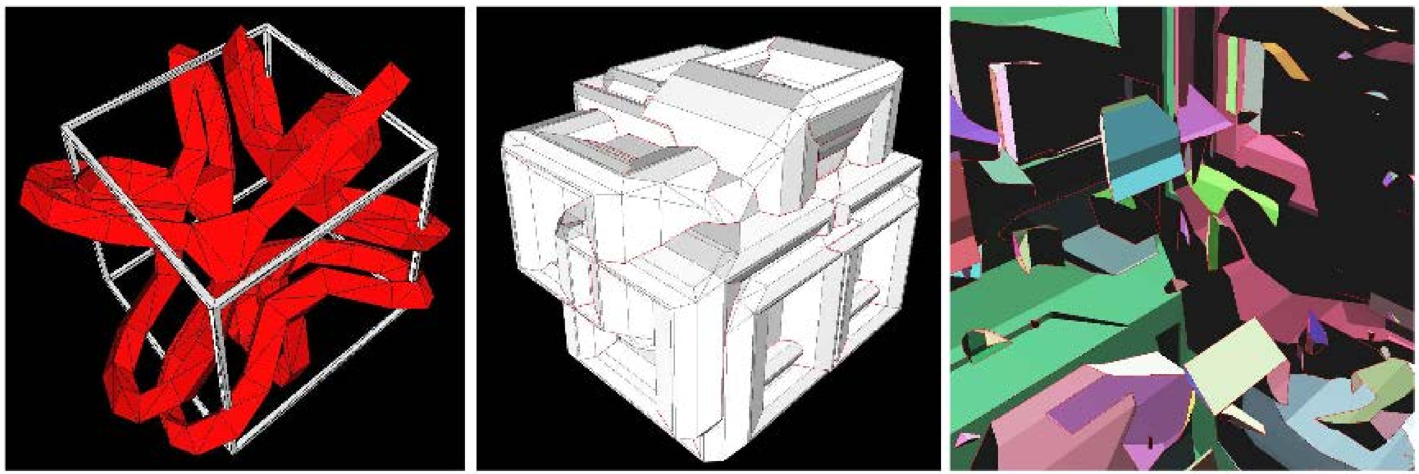
# Example – 2D Robot with Rotation



Modified based on Slides
by Prof. David Hsu, University of Singapore

# Minkowski Sum

- Can Minkowski Sums be computed in higher dimensions efficiently?



**Find a configuration that keeps the knot interlocked but without colliding with the cubic frame?**

Computing the Minkowski sum of **non-convex** polyhedra – **Time Complexity**: $O(n^3 m^3)$

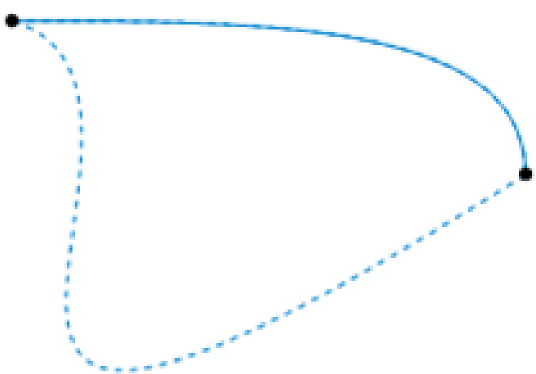
# Why need to study the topology of C-space?

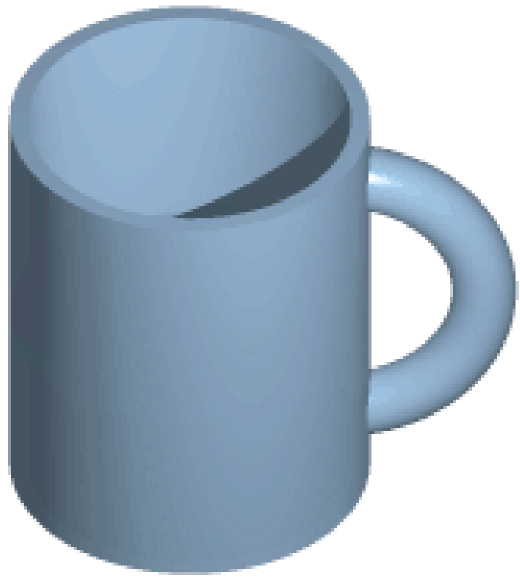
- Because in topology, a coffee mug can be equivalent to a donut

Two paths τ and τ' with the **same endpoints** is

**Homotopic**

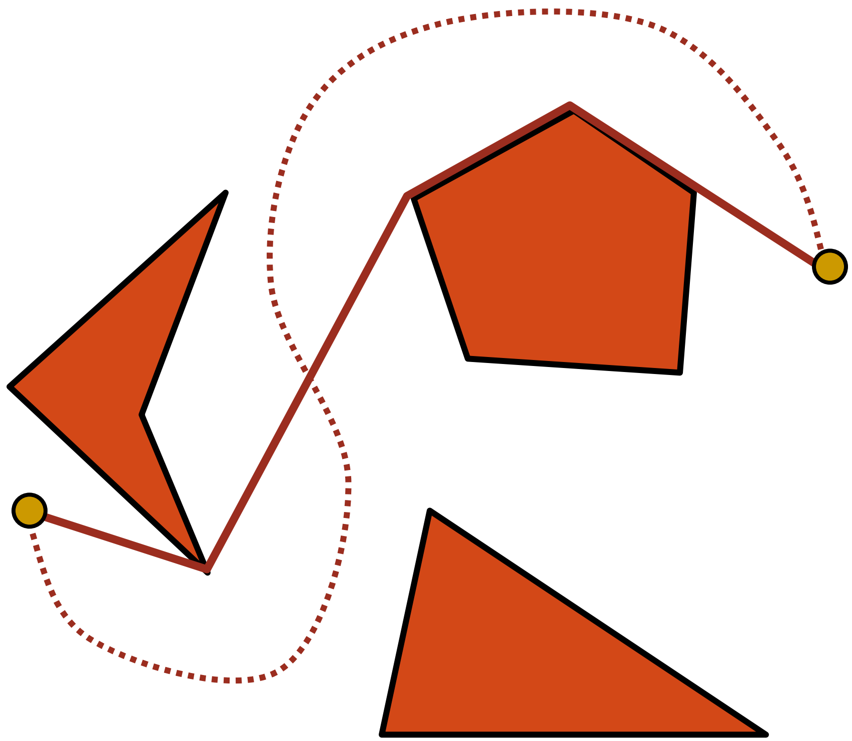
If one path can be deformed into **continuously** deformed into the other





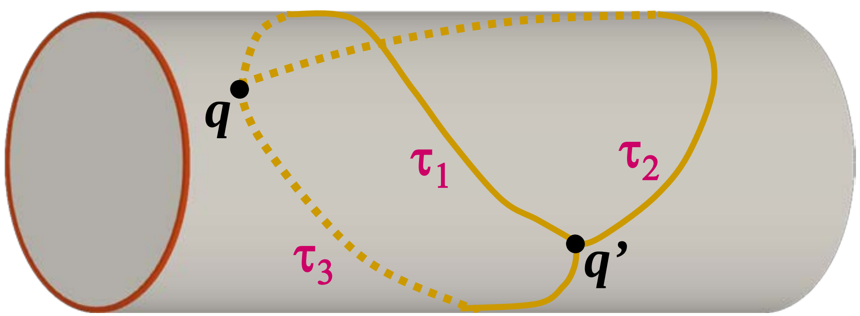
# Homotopic paths

- A homotopic class of paths
  - All paths that are homotopic to one another.



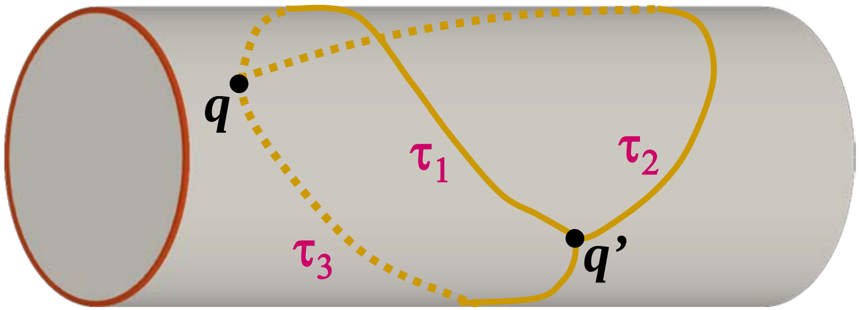
# Homotopic paths

- A cylinder without top and bottom
- $\tau_1$ and $\tau_2$ are homotopic
- $\tau_1$ and $\tau_3$ are not homotopic



# 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.
  - Can you think of an example?



# Distance in C-space

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

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

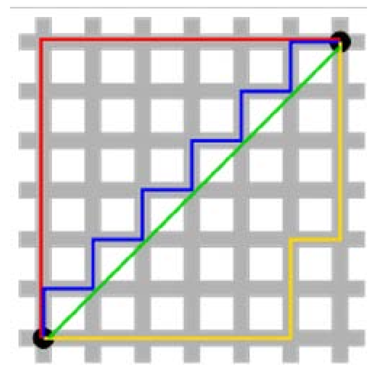
Such that

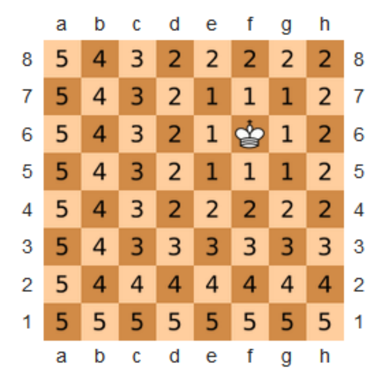
- $d(q, q') = 0$ if and only if $q = q'$,
- $d(q, q') = d(q', q)$,
- $d(q, q') <= d(q, q'') + d(q'', q')$ .

# Discussion

- Do we need to have an explicit representation of C-obstacles to do path planning?

- Do we need a specialized distance metric in C-space to do path planning?
  - Can we use Euclidian distance between configurations?
  - Can we use Euclidian distance for all the problems?





# Distance metric

- L1-norm (Manhattan distance) – follow the grid, like a taxi driver

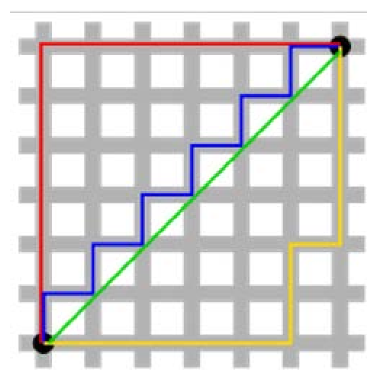
$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|,$$

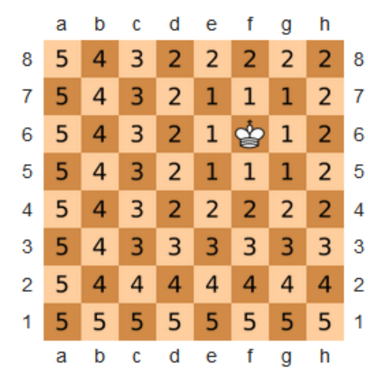
- L2-norm (Euclidian distance)

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

- $L_\infty$-norm (chessboard distance)

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

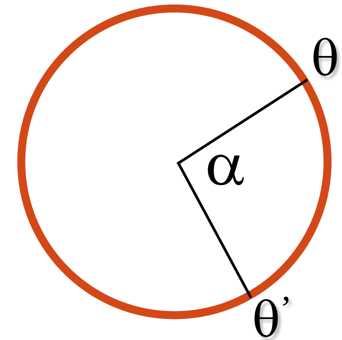




- Read
  - Principles: Appendix H – Graph representation and basic search
- HW1 is posted
  - Due 2/1 at 12 noon

# Examples in $R^2 \times S^1$

- Consider $R^2 \times S^1$
  - $q = (x, y, \theta)$, $q' = (x', y', \theta')$ with $\theta, \theta' \in [0, 2\pi)$
  - $\alpha = \min \{ |\theta - \theta'|, 2\pi - |\theta - \theta'| \}$




$$d(q,q') = \max_{a \in A} \| a(q) - a(q') \|$$

- $$= \max_{a \in A} \sqrt{(x - x')^2 + (y - y')^2} + \alpha r_a$$

$$= \sqrt{(x - x')^2 + (y - y')^2} + \alpha \max_{a \in A} r_a$$

$$= \sqrt{(x - x')^2 + (y - y')^2} + \alpha r_{\max}$$