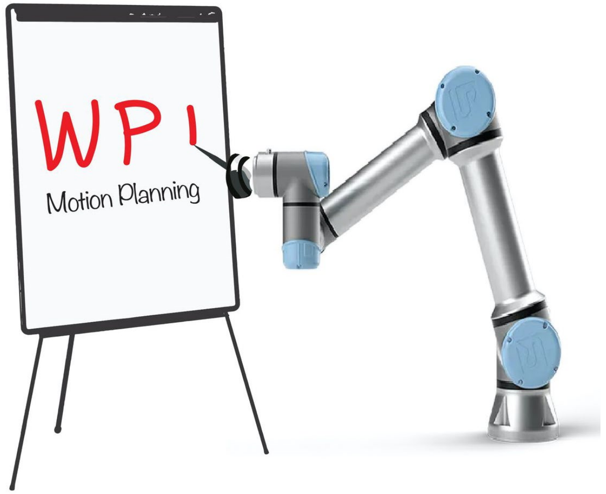
RBE550
Motion Planning
Configuration Space Obstacles

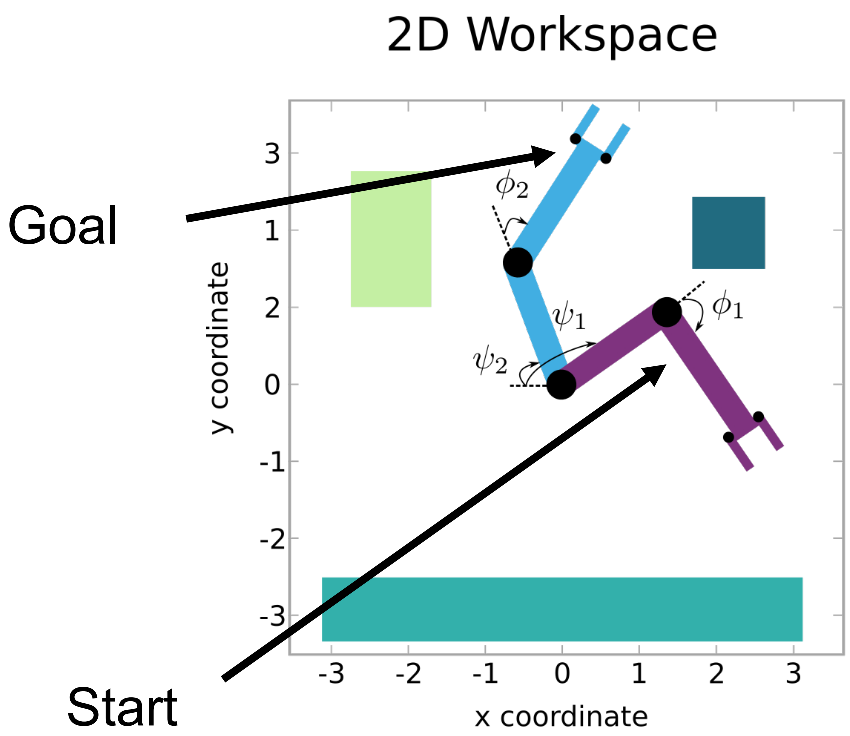


Constantinos Chamzas
www.cchamzas.com
www.elpislab.org

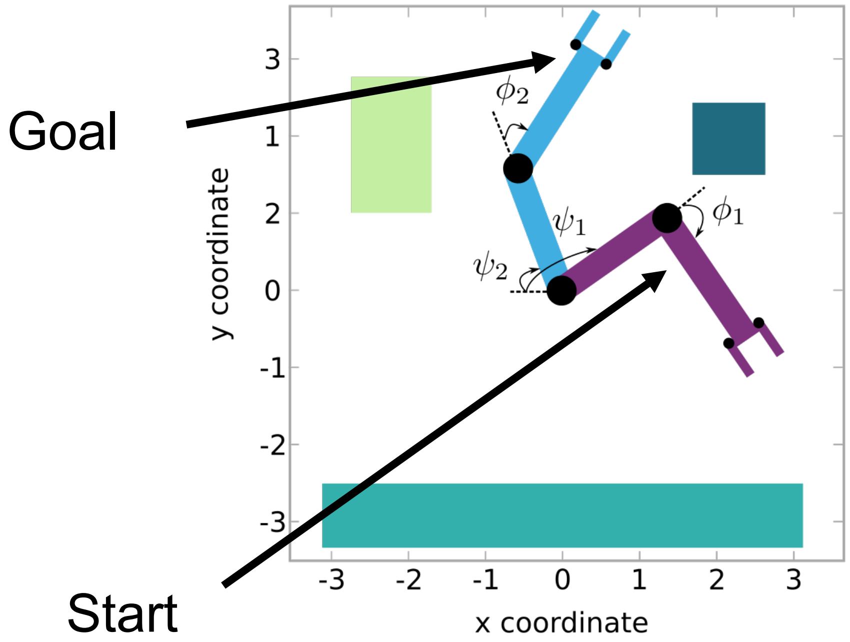
Disclaimer and Acknowledgments

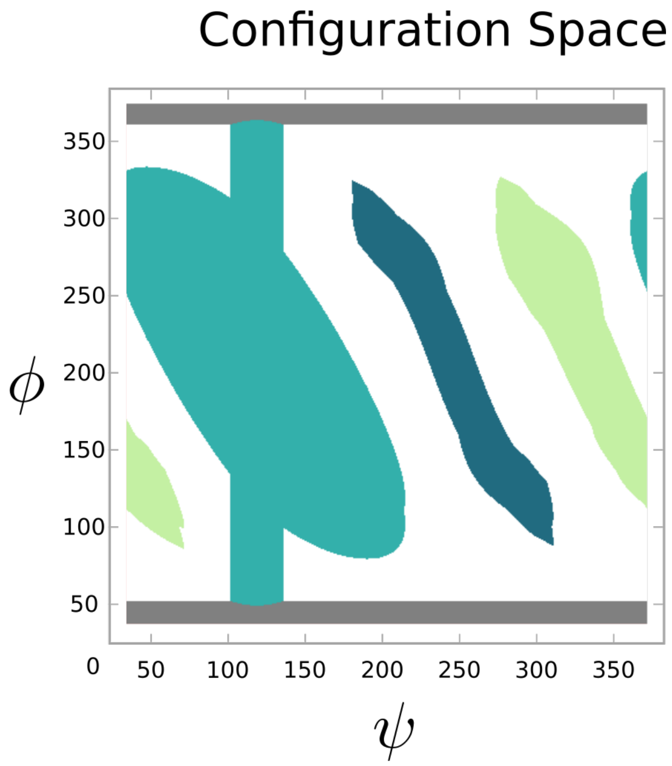
The slides are a compilation of work based on notes and slides from Constantinos Chamzas, Lydia Kavraki, Zak Kingston, Howie Choset, David Hsu, Greg Hager, Mark Moll, G. Ayorkor Mills-Tetty, Hyungpil Moon, Zack Dodds, Nancy Amato, Steven Lavalle, Seth Hutchinson, George Kantor, Dieter Fox, Vincent Lee-Shue Jr., Prasad Narendra Atkar, Kevin Tantiseviand, Bernice Ma, David Conner, Morteza Lahijanian, Erion Plaku, and students taking comp450/comp550 at Rice University.

Last Time: Configuration Space

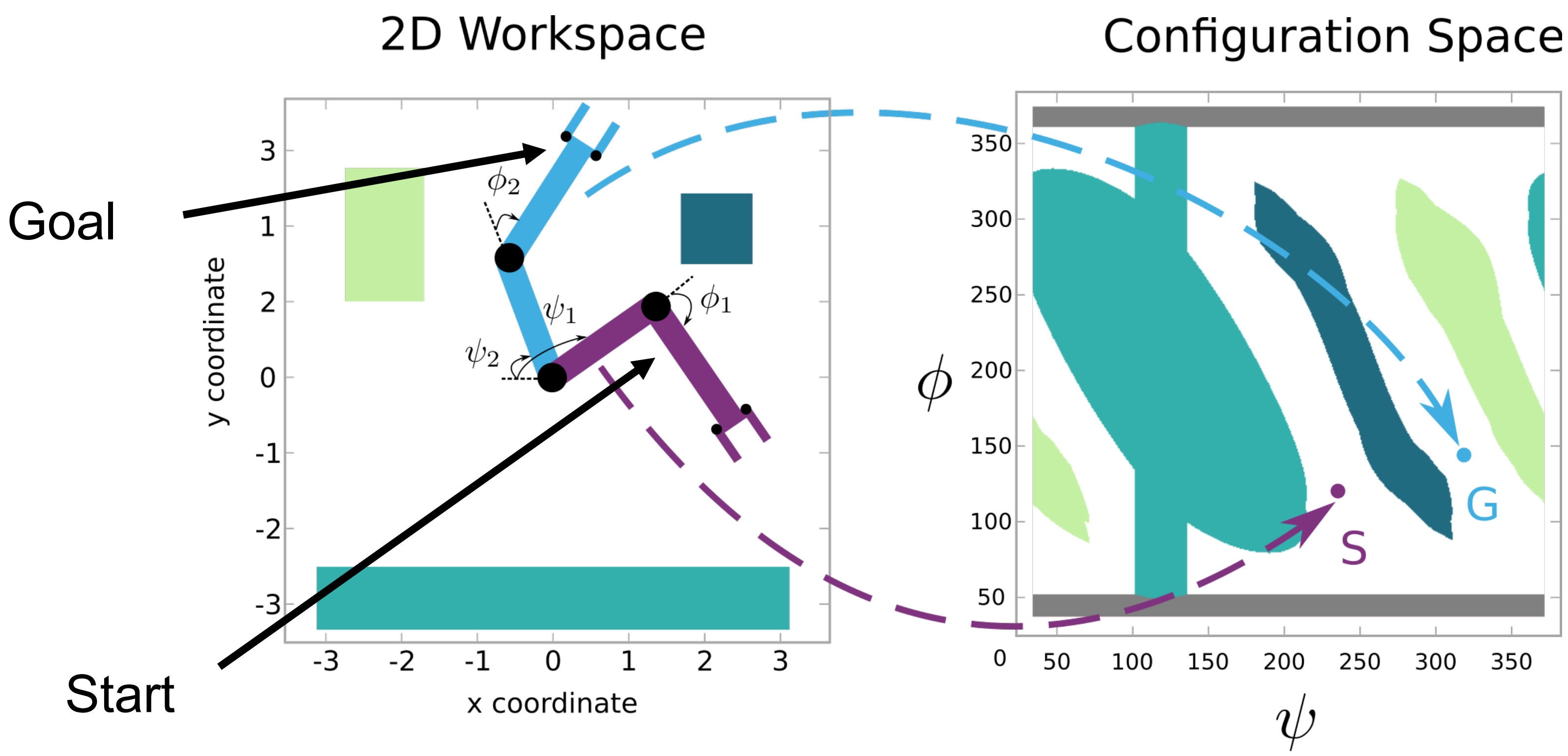


Big Idea: Configuration Space

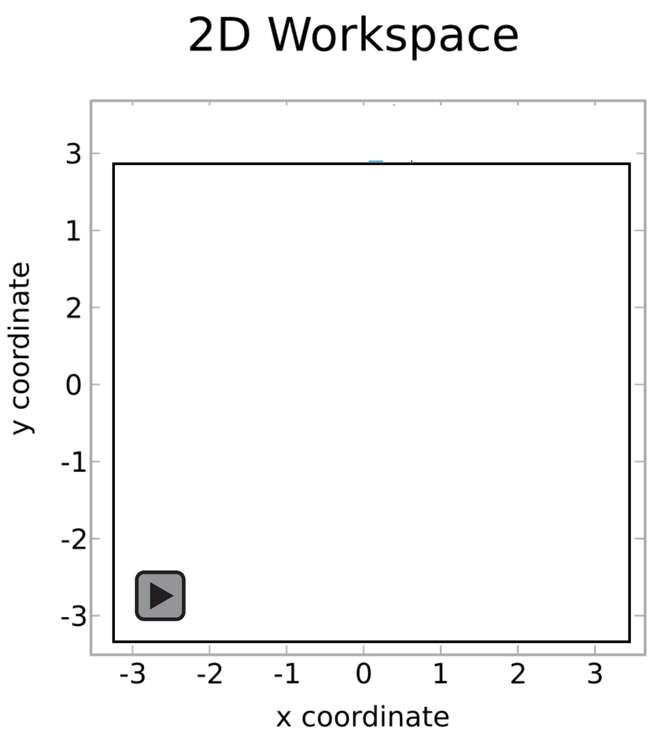


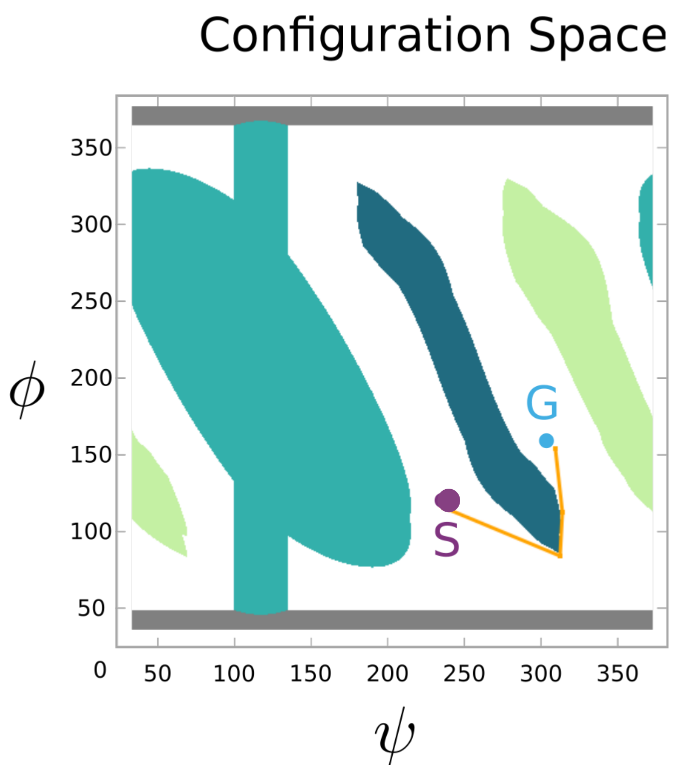


Big Idea: Configuration Space



Valid Path in Configuration space is valid in the Workspace





Now we can use point Planning Algorithms to plan in the C-Space!

Configuration Space Definitions

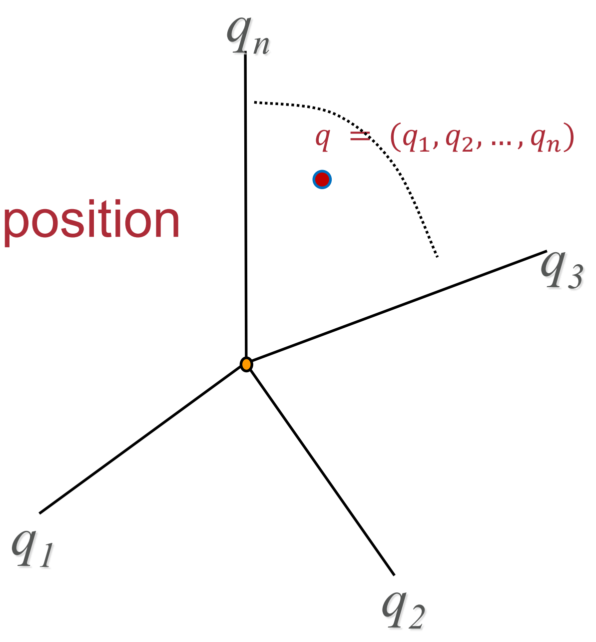
- The configuration of a moving object is a specification of the position of every point on the object.

- Usually a configuration is expressed as a vector of position & orientation parameters:

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

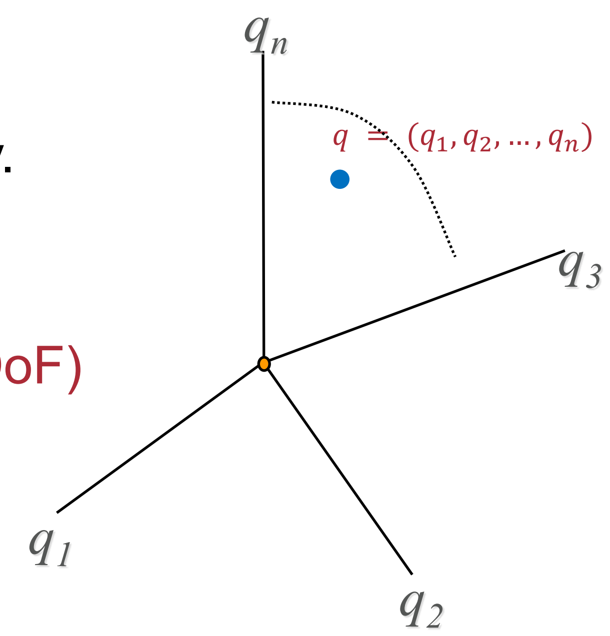
- The configuration space C is the set of all possible configurations.

- A configuration is a point in C



Configuration Space

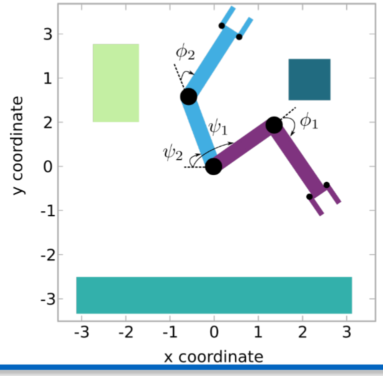
- The dimension of a configuration space:
  - The minimum number of parameters needed to specify the configuration of the object completely.
  - also called the number of degrees of freedom (DoF) of a moving object (Robot).



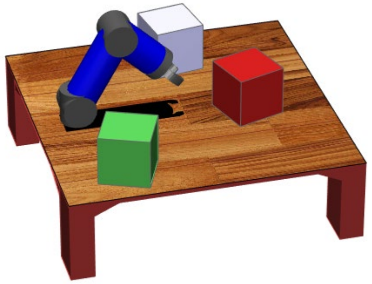
Configuration Spaces for Robots

Workspace

2-Dof Robot

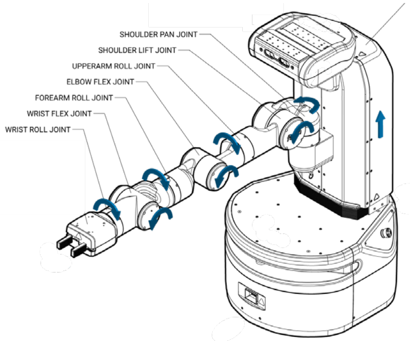


3-Dof Robot

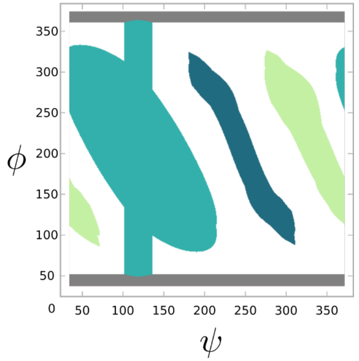


Das et al, *TRO* 2020

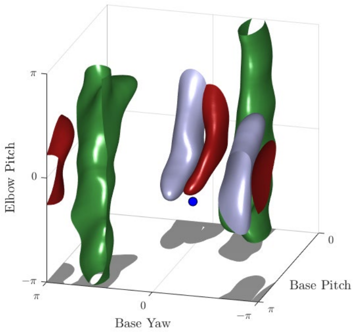
8-Dof Robot



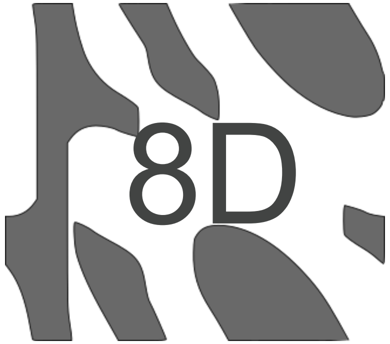
Configuration Space



2 Dimensions



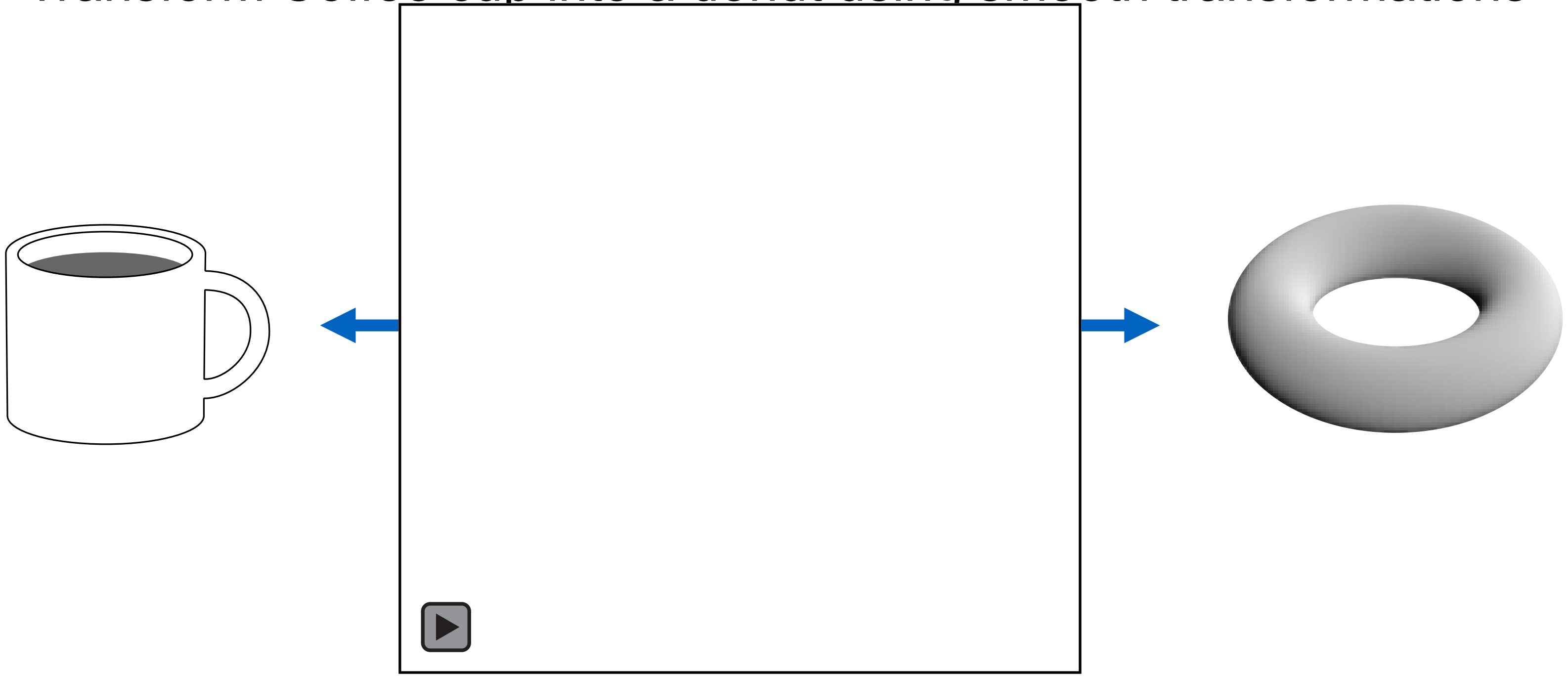
3 Dimensions



8 Dimensions

Topology: Characterizing High-Dim Spaces

- Transform Coffee cup into a donut using smooth transformations



No cutting, tearing or pasting !!

Topological Equivalence (Homeomorphism)

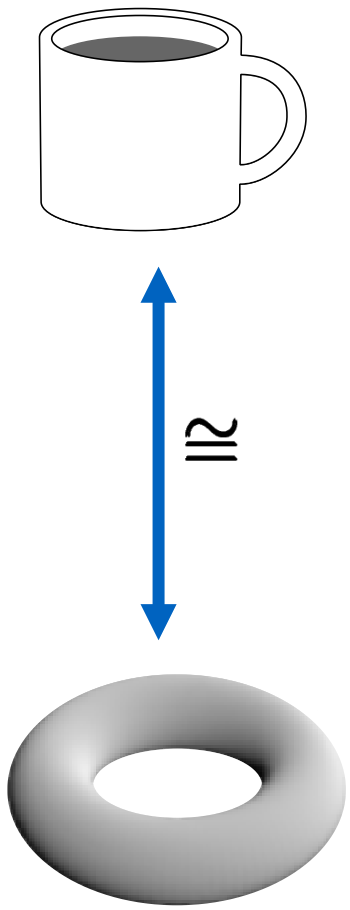
- **Interval Homeomorphism:** Any open interval of real line is homeomorphic to any other open interval

(0,1) $\cong$ (0,5), via $x \mapsto 5x$

(0,1) $\ncong$ (0,1]

(0,1) $\cong$ (1,∞), via $x \mapsto 1/x$

(-1,1) $\cong$ (-∞,∞), via $x \mapsto \tan^{-1}(\pi x/2)$



Special Orthogonal Group SO(2) and SO(3)

Special Orthogonal Group SO(2), SO(3), also known as the group of rotation matrices, is the set of all 2x2, 3x3 (respectively) real matrices that satisfy (a) $R^T R = I$ and det R = 1

(A group consists of a set of elements and an operation –matrix multiplication here- such that for all A, B in the group, the following properties are satisfied (a) closure, (b) associativity, and (c)identity element existence.)

Special Euclidean Group SE(2)

Special Euclidean Group SE(2) characterizes rotations and translations for 2D rigid bodies through a single matrix.

$$\left\{ \begin{pmatrix} R(\theta) & t \\ 0 & 1 \end{pmatrix} | R(\theta) \in SO(2), t \in \mathbb{R}^2 \right\}$$

$R^2 x\ SO(2)$: topological representation

SE(2): matrix group representation

We will use interchangeably: SE(2) $\sim R^2\ x\ SO(2)$

Special Euclidean Group SE(3)

Special Euclidean Group SE(3) characterizes rotations and translations for 3D rigid bodies through a single matrix.

$$\left\{ \begin{pmatrix} \boldsymbol{R}(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) & \boldsymbol{t} \\ \boldsymbol{0} & \boldsymbol{1} \end{pmatrix} \mid \boldsymbol{R}(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) \in \boldsymbol{SO}(\boldsymbol{3}), \boldsymbol{t} \in \boldsymbol{R}^{\boldsymbol{3}} \right\}$$

$R^3 x\ SO(3)$: topological representation

SE(3): matrix group representation

We will use interchangeably: SE(3) $\sim R^3\ x\ SO(3)$

Recap:Topological Spaces

- Some important topological spaces:
  - $\mathbb{R}$: real number line
  - $\mathbb{R}^n$: n-dimensional Cartesian space
  - S^1: boundary of circle in 2D
  - S^2: surface of sphere in 3D
  - $SO(2)$, $SO(3)$: set of 2D, 3D orientations (special orthogonal group)
  - $SE(2)$, $SE(3)$: set of rigid 2D, 3D translations and rotations (special Euclidean group)
  - $A \times B$: Cartesian product, power notation $A^n = A \times A \times \cdots \times A$
  - $T = S^1 \times S^1$: torus
- Homeomorphism $\sim$ denotes topological equivalence
  - Continuous mapping with continuous inverse (bijective)
    - Cube $\sim S^2$
    - $SO(2) \sim S^1$
    - $SE(3) \sim \mathbb{R}^3 \times SO(3)$

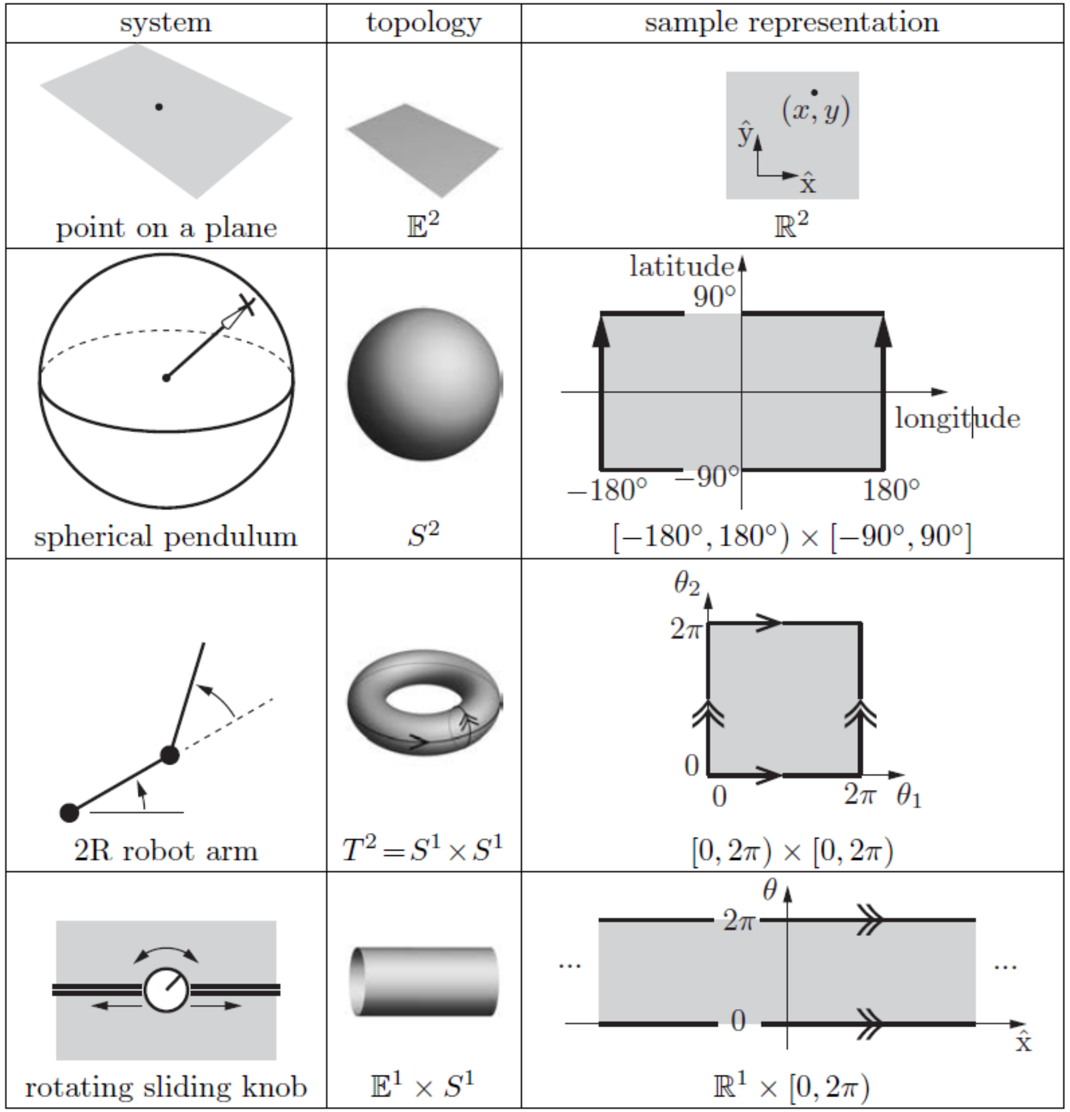
Topology of some common robots C-spaces

- Examples of some common robots

Type of robot	Representation of C-Space
Mobile robot translating in the plane	$\mathbb{R}^2$
Mobile robot translating and rotating in the plane	$SE(2)$ or $\mathbb{R}^2 \times S^1$
Rigid body translating in 3-D	$\mathbb{R}^3$
A spacecraft	$SE(3)$ or $\mathbb{R}^3 \times SO(3)$
An n-joint revolute arm	T^n
A planar mobile robot with an attached n-joint arm	$SE(2) \times T^n$

- Note that:
- $S^1 \times S^1 \times \cdots \times S^1 = T^n$, n-dimensional torus
- $S^1 \times S^1 \times \cdots \times S^1 \neq S^n$, n-dimensional sphere
- $S^1 \times S^1 \times S^1 \neq SO(3)$
- $SE(2) \neq \mathbb{R}^3$
- $SE(3) \neq \mathbb{R}^6$

Some more Topology Examples

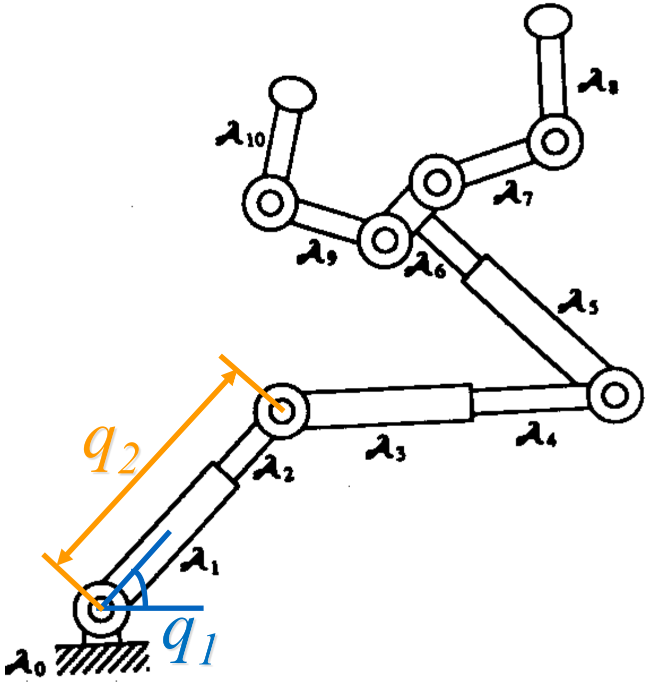


Topology of C-Spaces

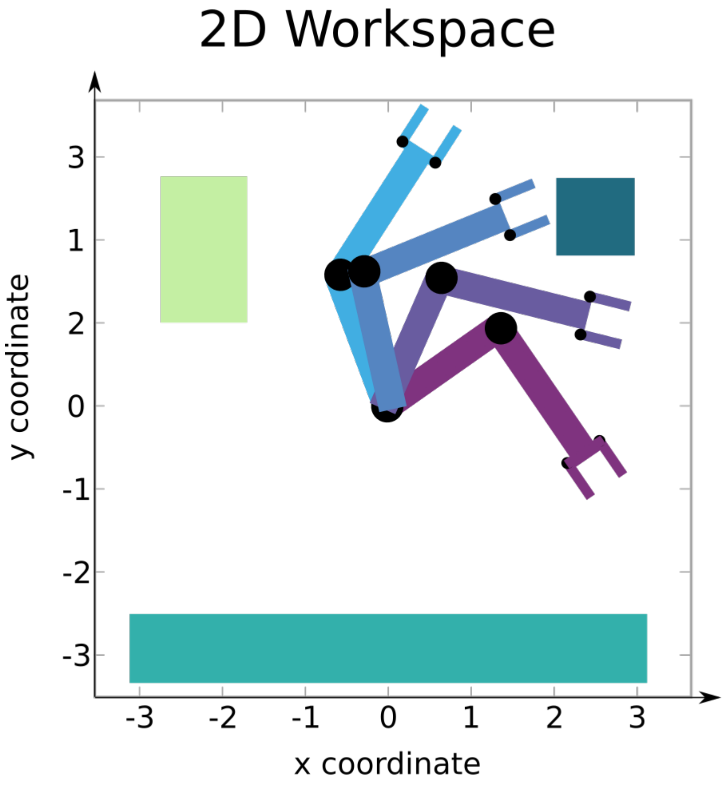
- Example: articulated robot
  - An articulated object is a set of rigid bodies connected at the joints.

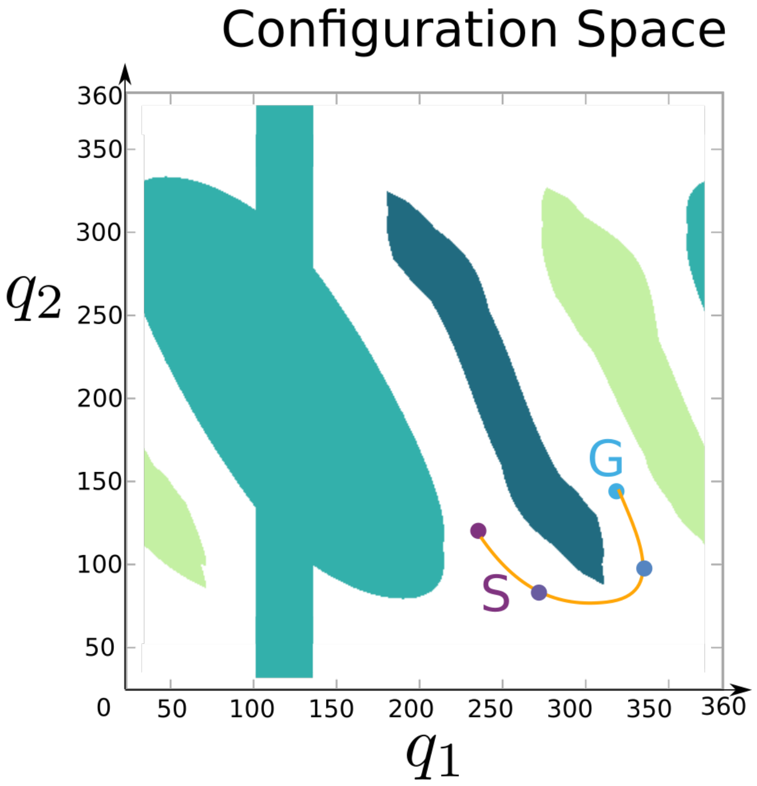
C-space: $(S^1)^7 \times I^3$

I: Real intervals



Path in Configuration Space must avoid C-space obstacles





Obstacles in the Configuration Space

Workspace

- A is the robot in the workspace $W \subseteq \mathbb{R}^2$ or $W \subseteq \mathbb{R}^3$
- $WO \subseteq W$ is the set of all obstacles in W

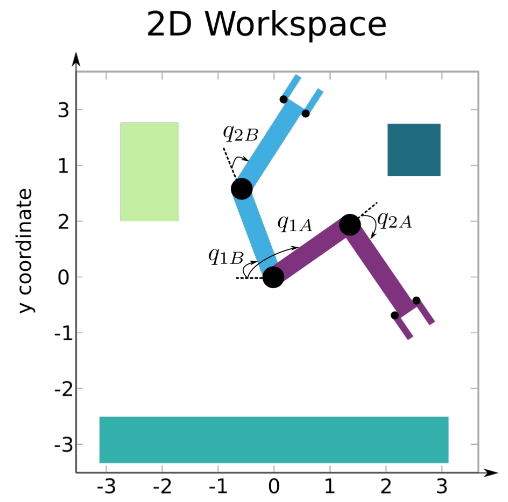
C-space

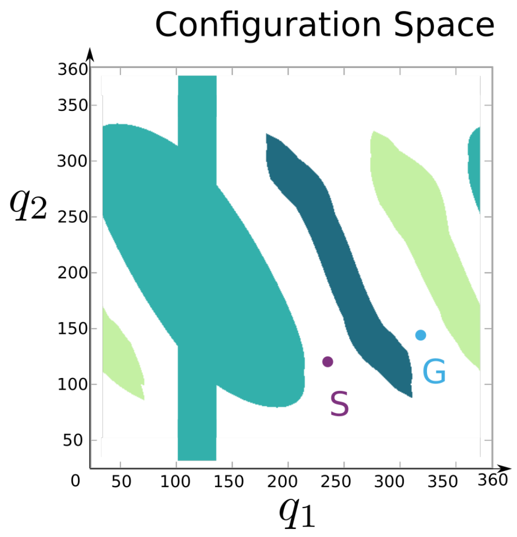
- C-space: the set of all configurations q
- *Obstacle region* $C_{obs} \subseteq C$ is defined as

$$C_{obs} = \{q \in C \mid A(q) \cap WO \neq \emptyset\},$$

where $A(q)$ is the robot in W placed at configuration q

i.e., the set of all configurations q at which $A(q)$, the transformed robot, intersects the obstacle set





Obstacles in the Configuration Space

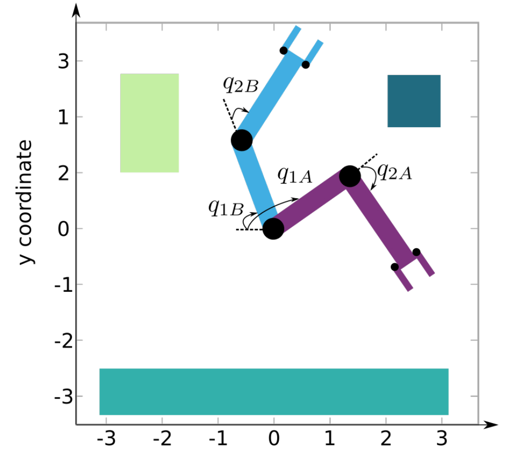
- A configuration q is collision-free, or free, if the robot placed at q does not intersect any obstacles in the workspace

- The *free space* $\mathcal{C}_{\text{free}}$ is the set of free configurations

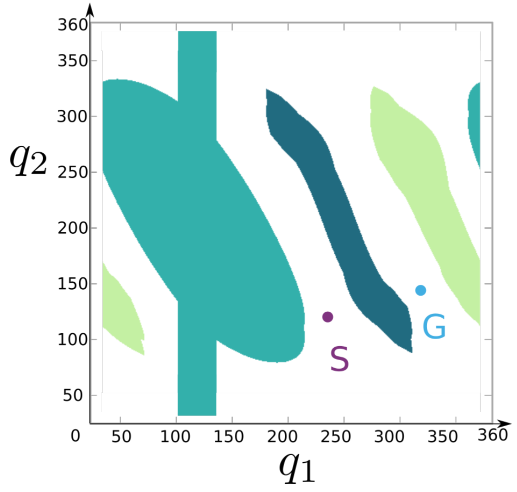
$$\mathcal{C}_{\text{free}} = \mathcal{C} \setminus \mathcal{C}_{obs}$$

- If $\mathcal{A}(q)$ "touches" $\mathcal{WO}$, then $q \in \mathcal{C}_{obs}$

2D Workspace



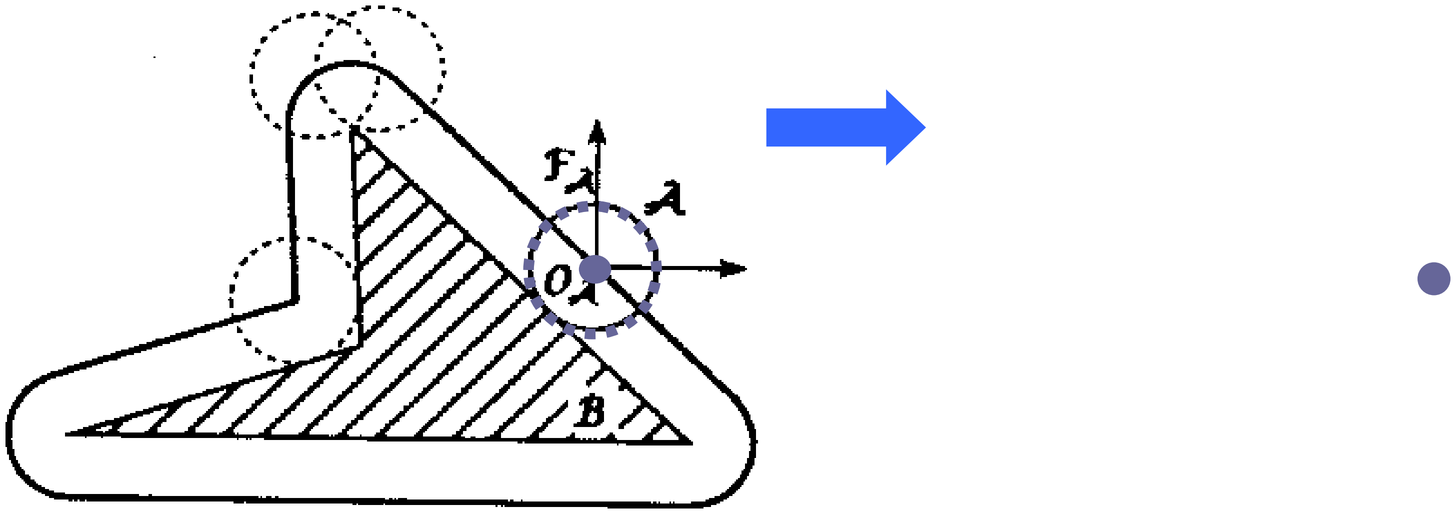
Configuration Space



Obstacles in the Configuration Space

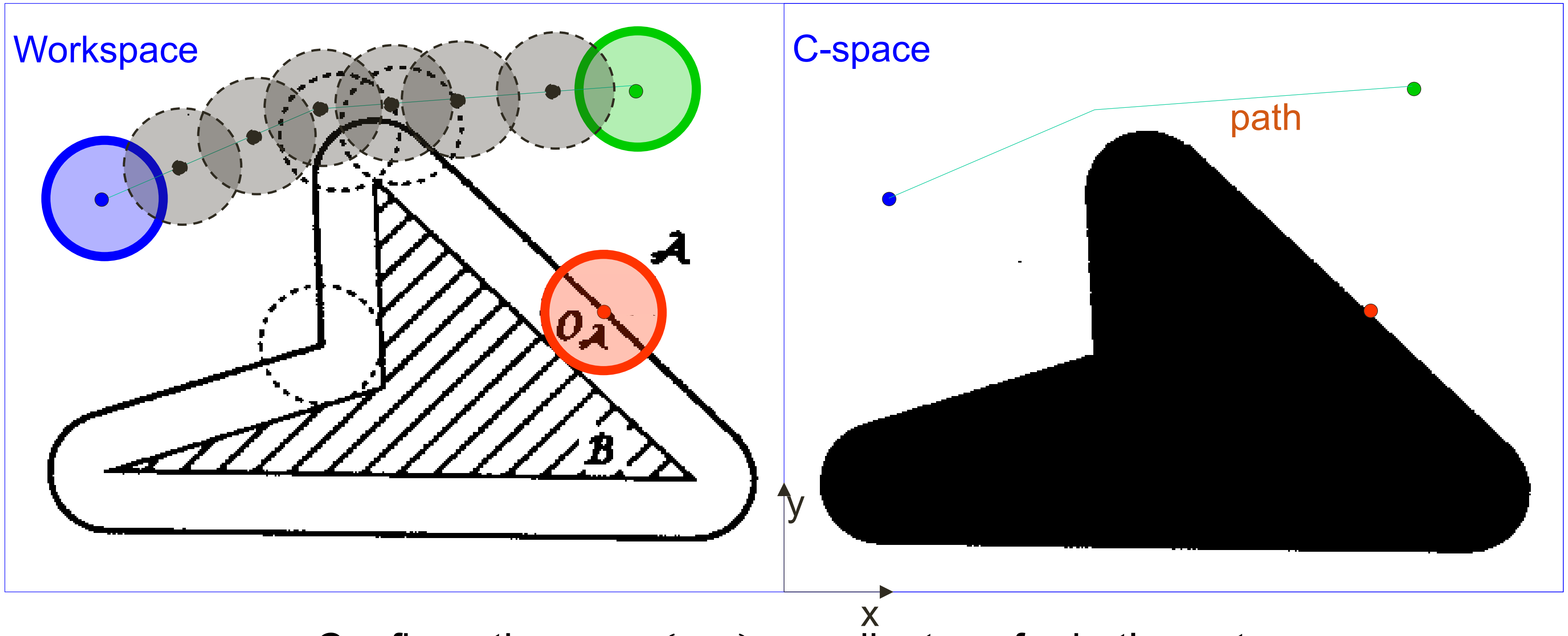
- 2-D disc robot (translation only)

workspace



Obstacles in the Configuration Space

- 2-D disc robot (translation only)



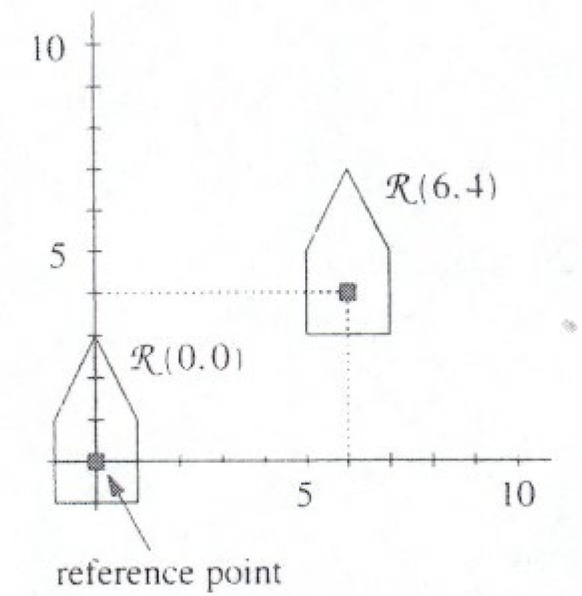
- Configuration: $q = (x, y)$ coordinates of robot's center
- configuration space $C = \mathbb{R}^2$
- free space C_{free} = the set of collision-free configurations

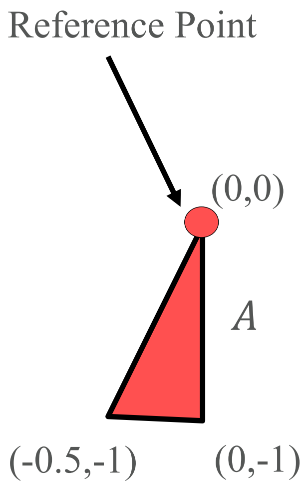
Obstacles in the Configuration Space

- **Problem:**
  - **Given** a convex polygonal robot (moving object) translating in 2-D workspace with polygonal obstacles,
  - **Compute** the c-space obstacles (polygons)

How fast is this computation?

Robot is Defined With a Reference Point





So when we say the robot in configuration q=(0,0) we mean with respect to the reference point

C-space Obstacles

- If O is an obstacle in the workspace and A is a moving object, then the C-space obstacle corresponding to O is

$$O \ominus A$$

Minkowski difference

Minkowski Sum and Differences

- The Minkowski sum and difference allow the fast computation of configurations space obstacles

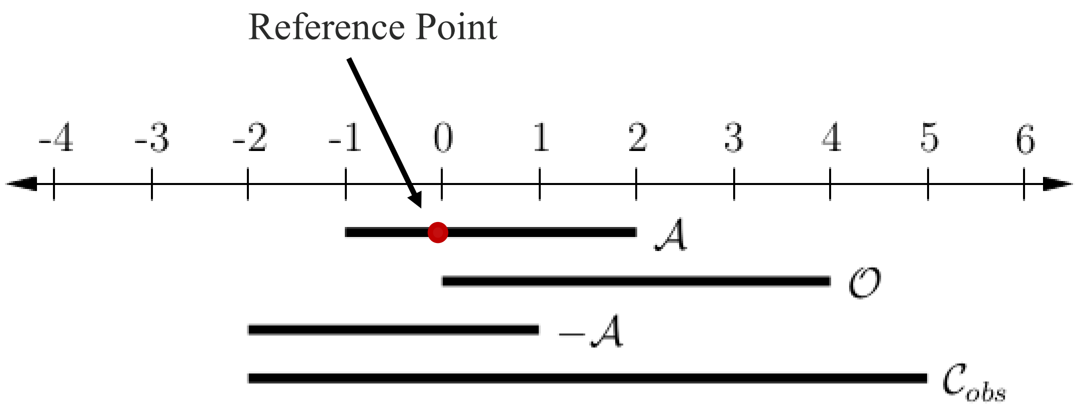
- The Minkowski sum of two sets P and Q, denoted by $P \oplus Q$, is defined as

$$P \oplus Q = \{ p + q \mid p \in P \ , \ q \in Q \}$$

- Similarly, the Minkowski difference is defined as

$$P \ominus Q = \{ p - q \mid p \in P \ , \ q \in Q \}$$

C-Space Obstacle in a 1D Case



$A = [-1, 2]$ $O = [0\,, 4]$

$-A = [-2, 1]$

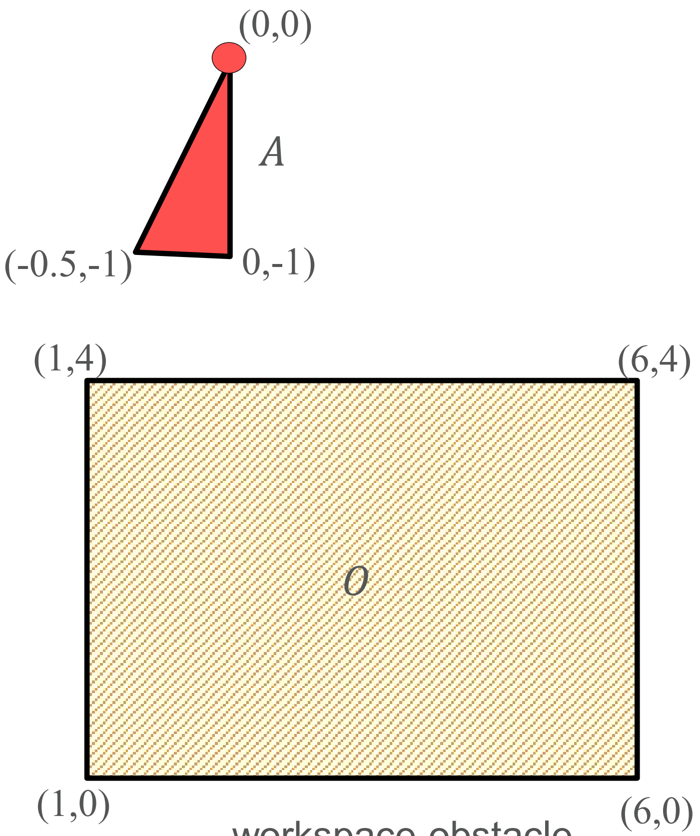
$C_{obs} = O \ominus A = O + (-A) = [-2, 5]$

C-Space obstacles

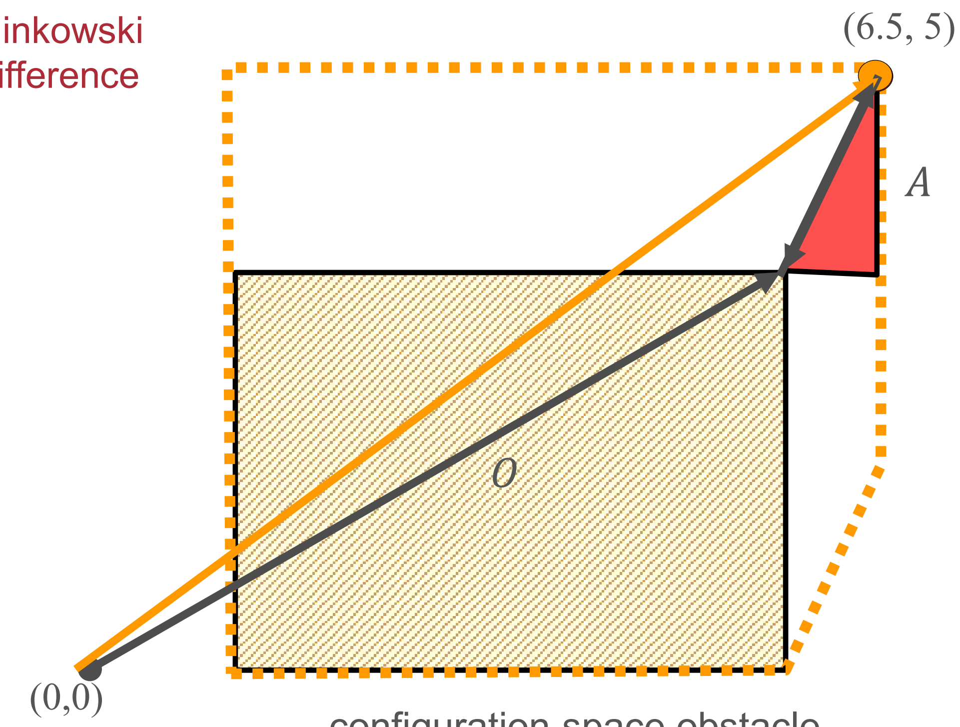
- If O is an obstacle in the workspace and A is a moving object, then the C-space obstacle corresponding to O is

$$O \ominus A$$

Minkowski difference



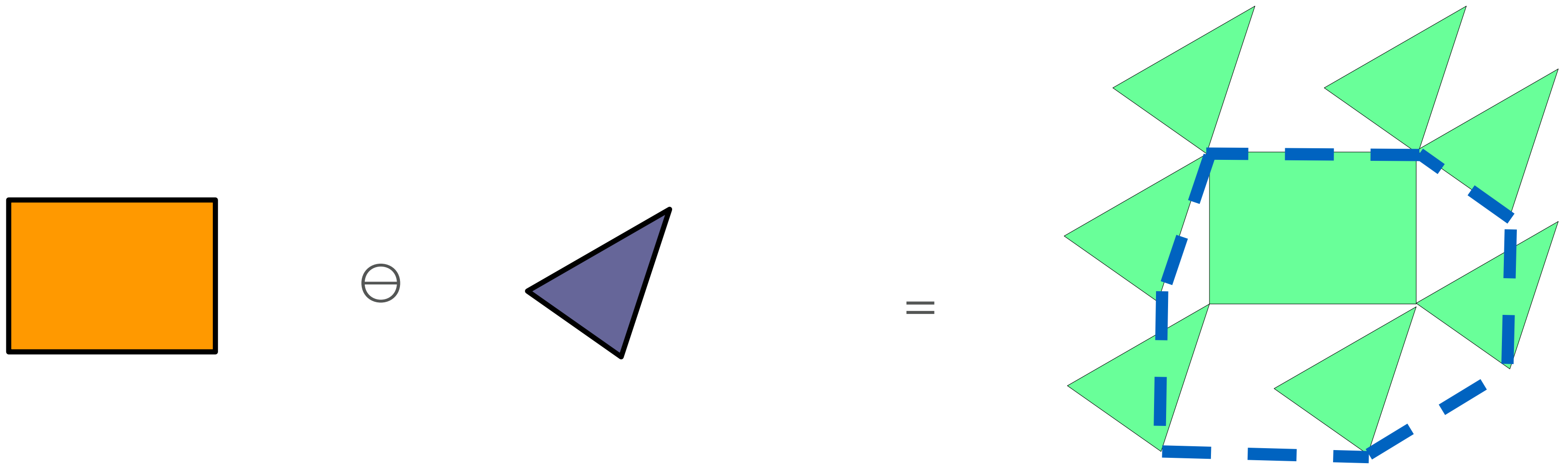
workspace obstacle



configuration space obstacle

Minkowski sum and Convex Polygons

- The Minkowski sum of two convex polygons P and Q of m and n vertices respectively is a convex polygon $P \oplus Q$ of $m + n$ vertices.
- The vertices of $P \oplus Q$ are the "sums" of vertices of P and Q.
- The vertices of $P \ominus Q$ are the "sums" of vertices of P and $-Q$.



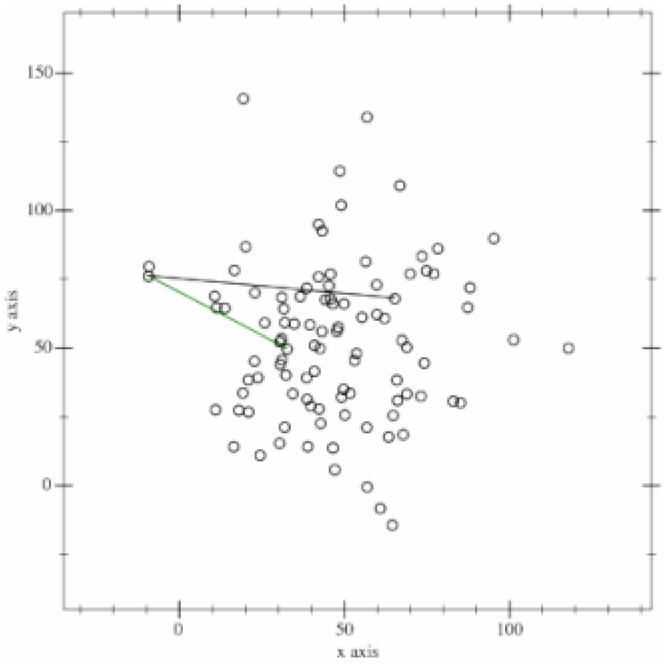
Efficient Computation

Theorem: If O and A are convex then

$$O \ominus A = O \oplus (-A) = conv\ (\,vert(O) \oplus vert(-A))$$

where vert(A) are the vertices of (A) and conv(X) is the convex hull of set X.

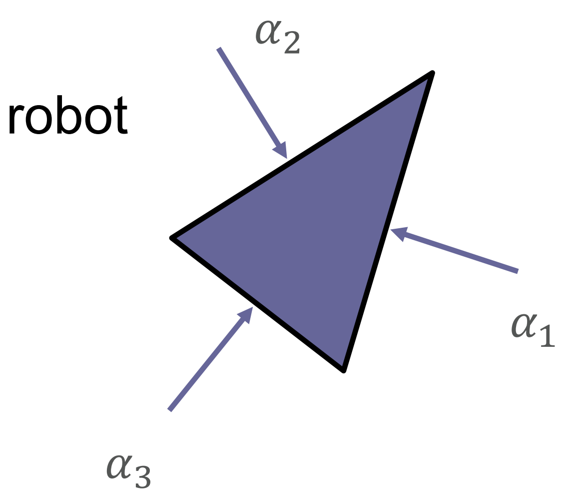
Creating Convex Obstacles: Gift Wrapping Algorithm

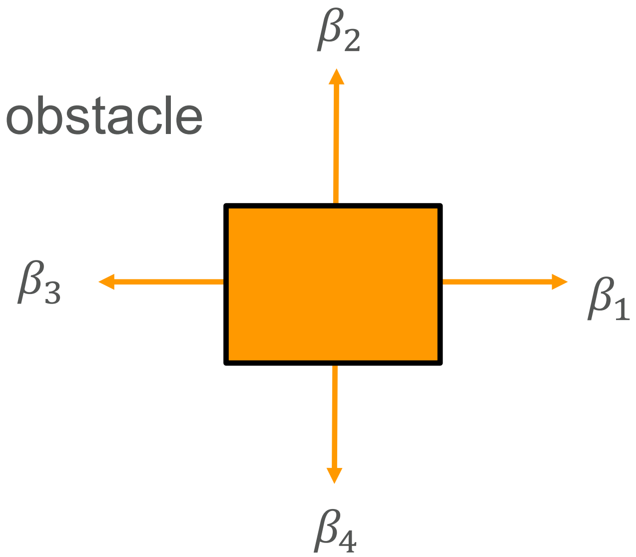


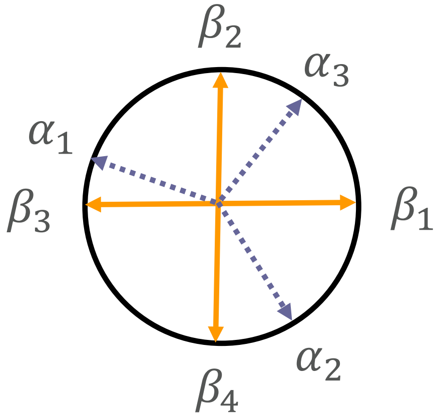
- For 2D case it is called the Jarvis March
- It has O(nh) complexity
  - n number of points
  - h number of points in the convex hull

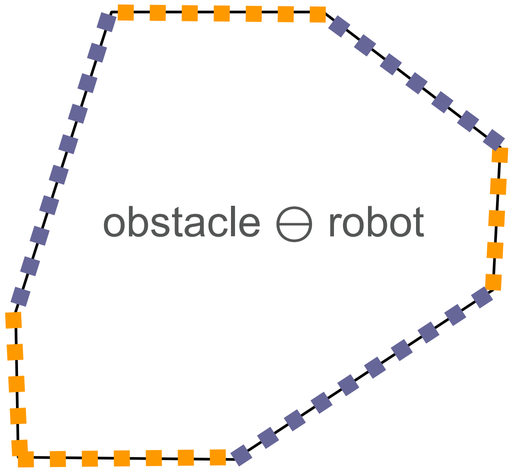
Animation of the gift wrapping algorithm. The red lines are already placed lines, the black line is the current best guess for the new line, and the green line is the next guess

Computing C-Space Obstacles in Linear Time (Alg1)







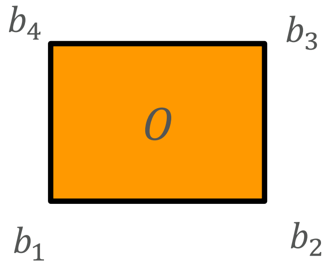


Choose the edges according to the order of angles

[Lozano-Perez / Guibas et al]

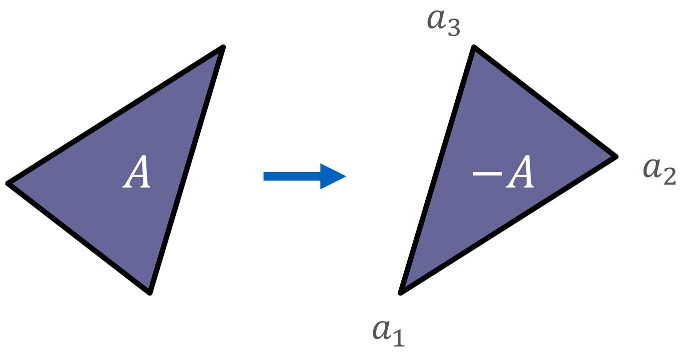
Computing C-Space Obstacles in Linear Time (Alg2)

Linear algorithm for: $O \ominus A = O \oplus -A$



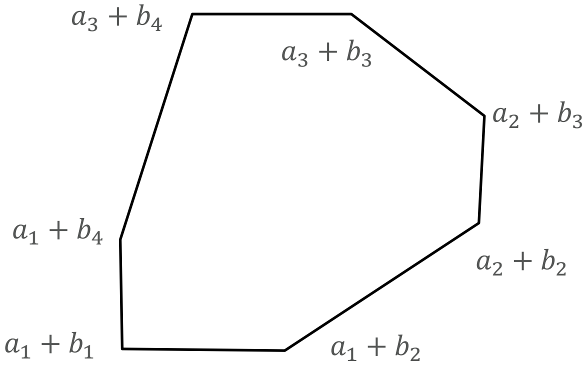
Vertices: b_1 b_2 b_3 b_4 b_1

Angles: 0^o 90^o 180^o 270^o



Vertices: a_1 a_2 a_3 a_1

Angles: 25^o 135^o 245^o



Vertices: start with $a_1 + b_1$

Increment the one w/ smaller angle

$a_1 + b_2$

$a_2 + b_2$

$a_2 + b_3$

$a_3 + b_3$

$a_3 + b_4$

$a_1 + b_4$

Computing C-Space Obstacles in Linear Time

- Linear algorithm for: $V \oplus W$

Input: convex polygons $V = (v_1, \dots, v_n)$, $W = (w_1, \dots, w_m)$, where the vertices are in counterclockwise order with v_1 and w_1 are the vertices with the smallest y coordinates

Output: ordered list of vertices of $V \oplus W$

$i \leftarrow 1,\ j \leftarrow 1$

repeat

    Add $v_i + w_j$ as a vertex to $V \oplus W$

    if $angle(v_i, v_{i+1}) < angle(w_j, w_{j+1})$ **then**

        $i \leftarrow i + 1$

    else if $angle(v_i, v_{i+1}) > angle(w_j, w_{j+1})$ **then**

        $j \leftarrow j + 1$

    else

        $i \leftarrow i + 1$

        $j \leftarrow j + 1$

    end if

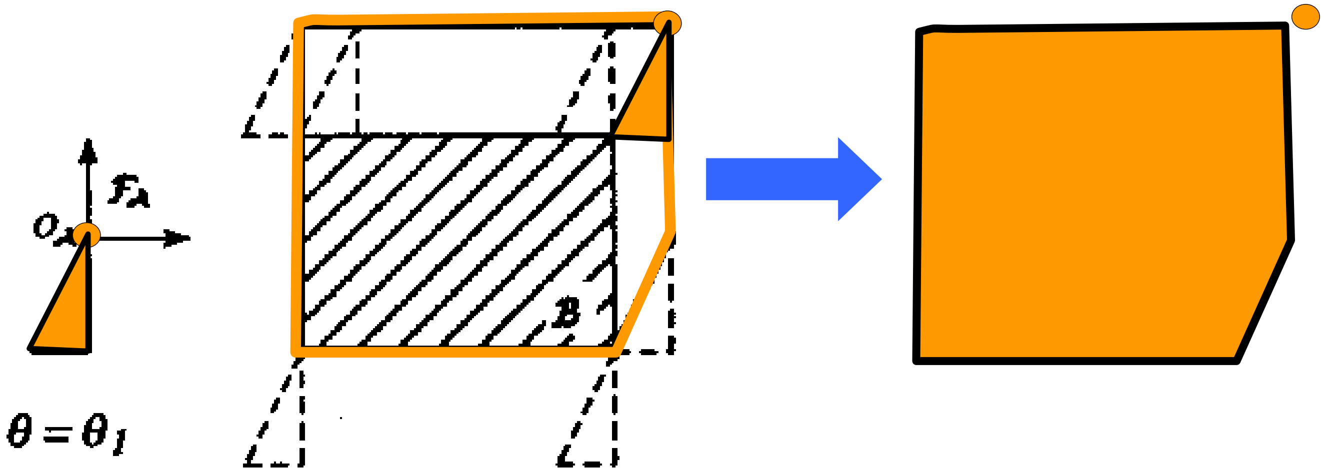
Until $i = n + 1$ and $j = m + 1$

Polygonal robot translating in 2-D workspace

- Only translation

workspace

Configuration space



Computational efficiency

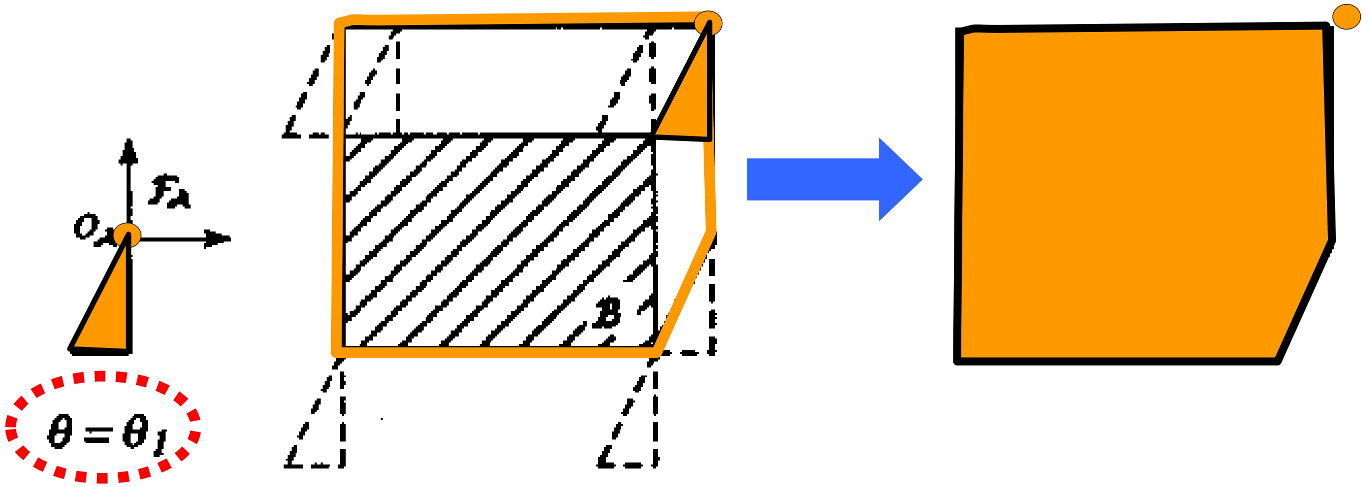
- Running time: $O(n+m)$
- Space: $O(n+m)$
- Non-convex obstacles and robots
  - Decompose into convex polygons (e.g., triangles or trapezoids), compute the Minkowski sums, and take the union
  - Complexity of Minkowksi sum can be as high as $O(n^2m^2)$
- Extendable to 3-D workspace (remember no rotations yet)

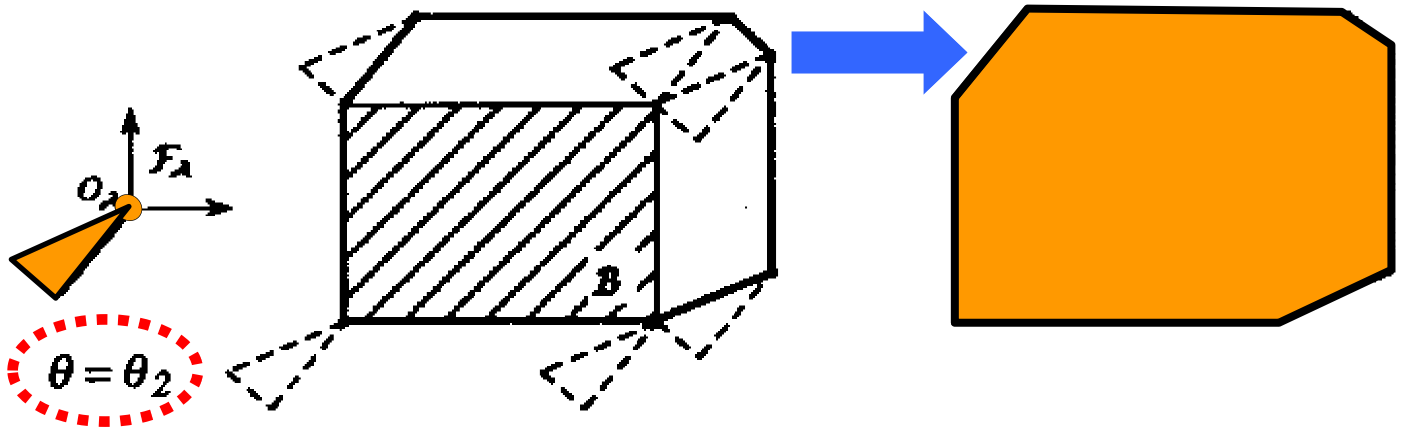
Polygonal robot in 2-D workspace

- Translation & rotation

Workspace

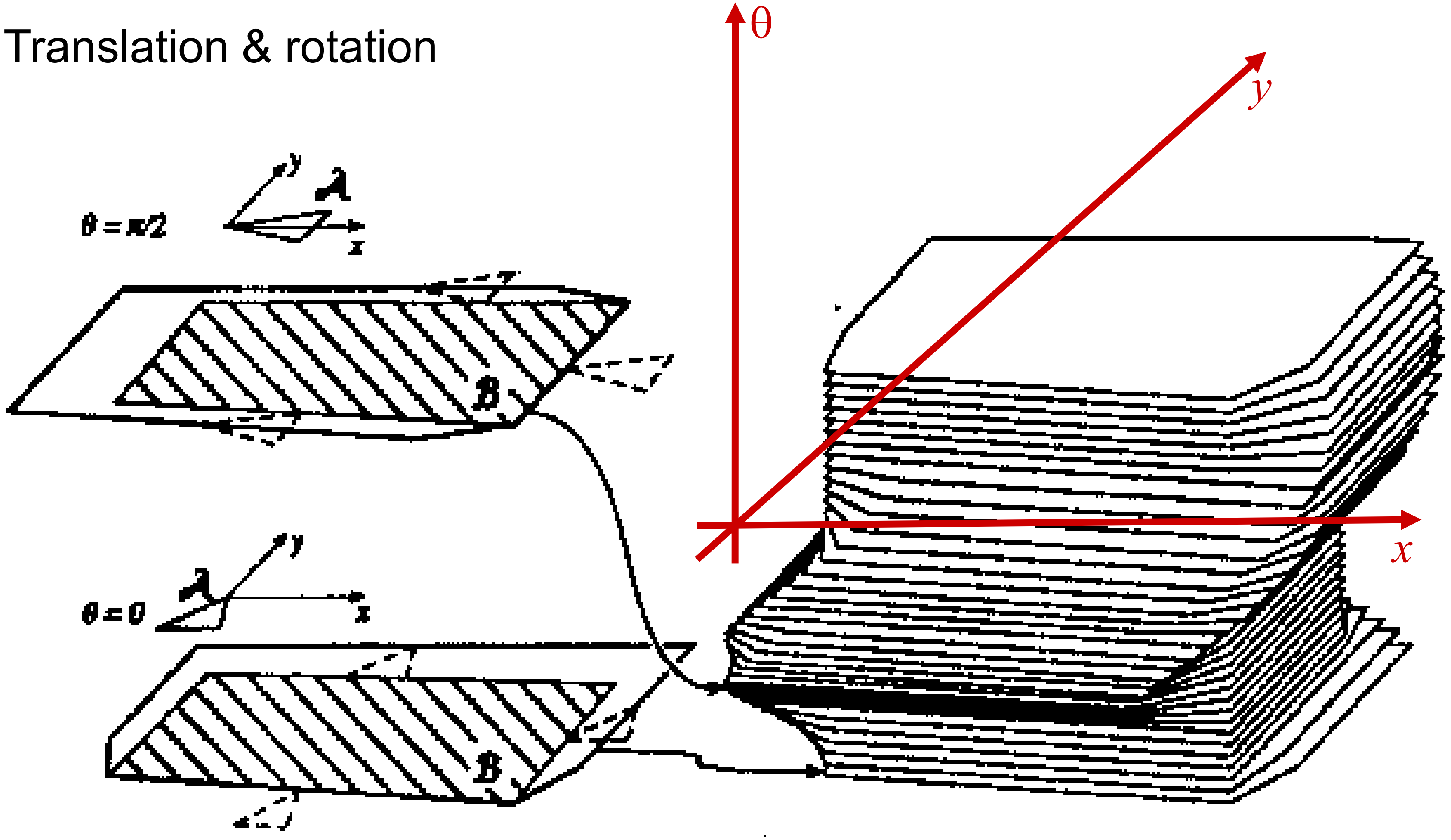
Configuration space



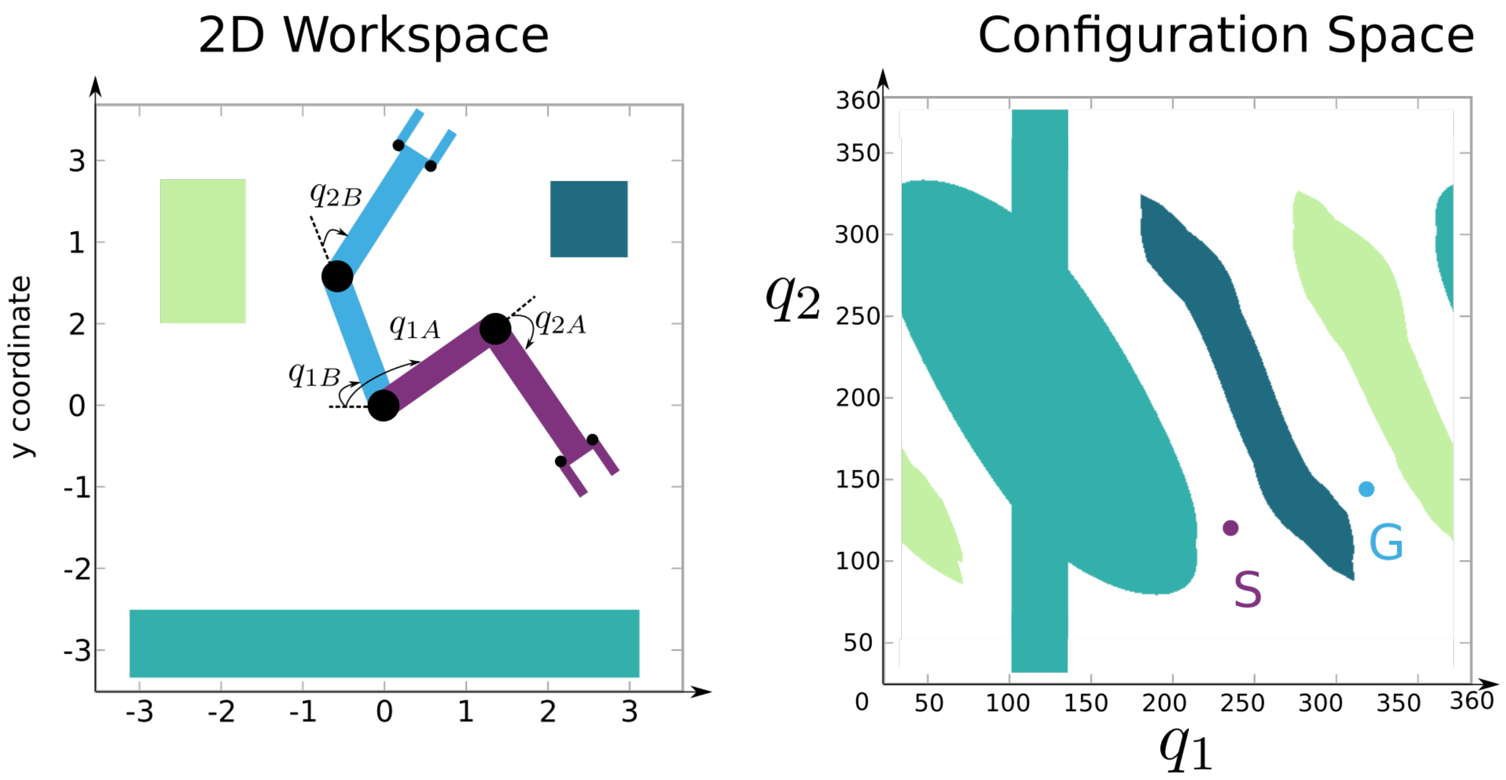


Polygonal robot in 2-D workspace

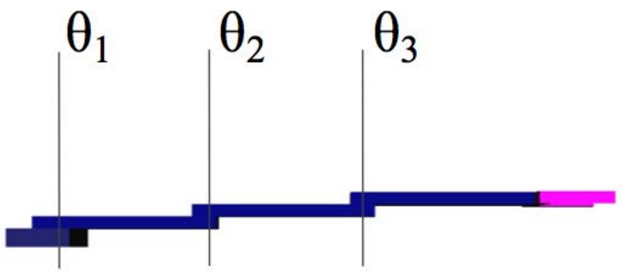
- Translation & rotation



2-link Articulated Robot C-Space

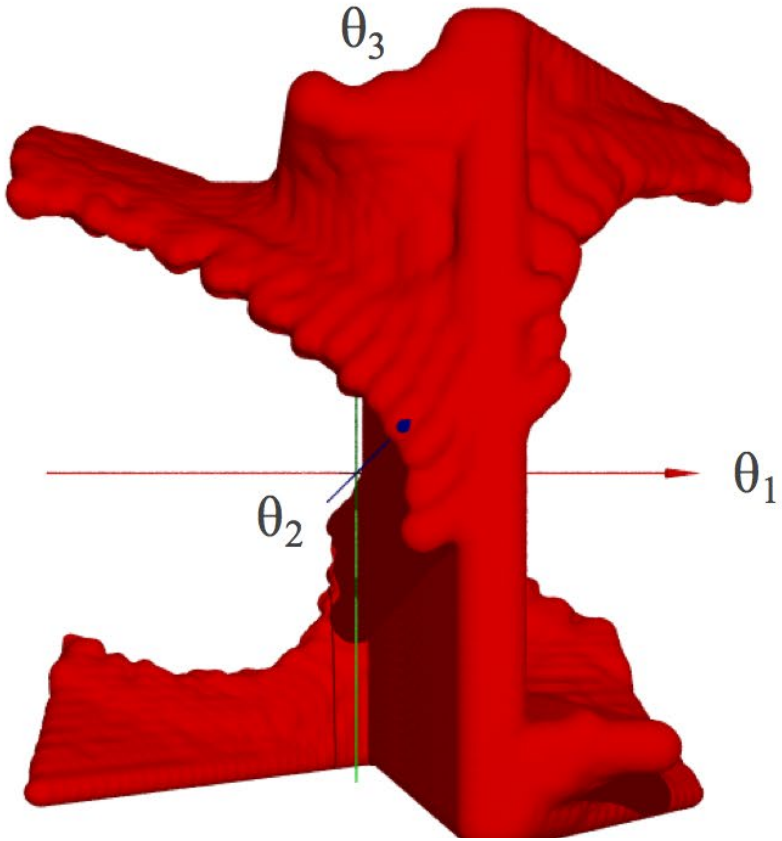


3-link Articulated Robot C-Space



side view

top view



Path in the Configuration Space

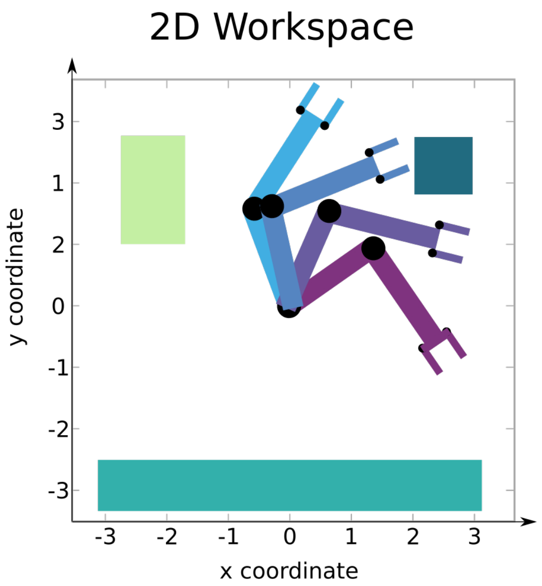
- A *path* in C is a continuous curve connecting two configurations q and q' :

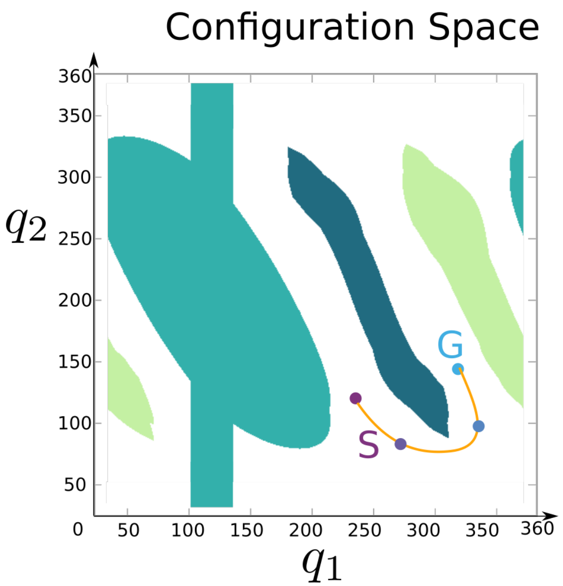
$$\tau : [0,1] \rightarrow C$$

curve is parameterized, and the parameter is normalized

such that $\tau(0) = q$ and $\tau(1) = q'$.

$\tau(s)$ for $s \in [0,1]$ is a point





Constraints on Paths

- A *trajectory* is a path parameterized by time:

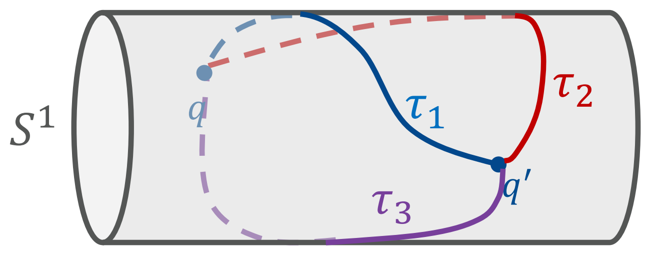
$$\tau: [0, T] \to C$$

such that $\tau(0) = q$ and $\tau(T) = q'$

Homotopic Paths

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

- Example: $\mathbb{R}^1 \times S^1$

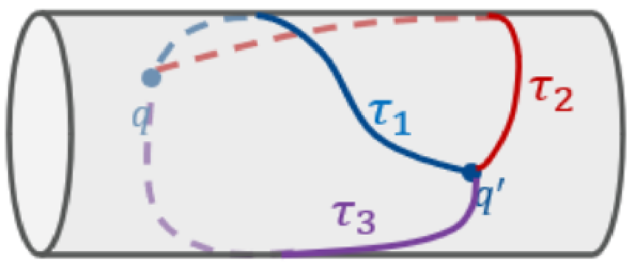


$\mathbb{R}^1 \times S^1$

  - τ_1 and τ_2 are homotopic

  - τ_1 and τ_3 are not homotopic

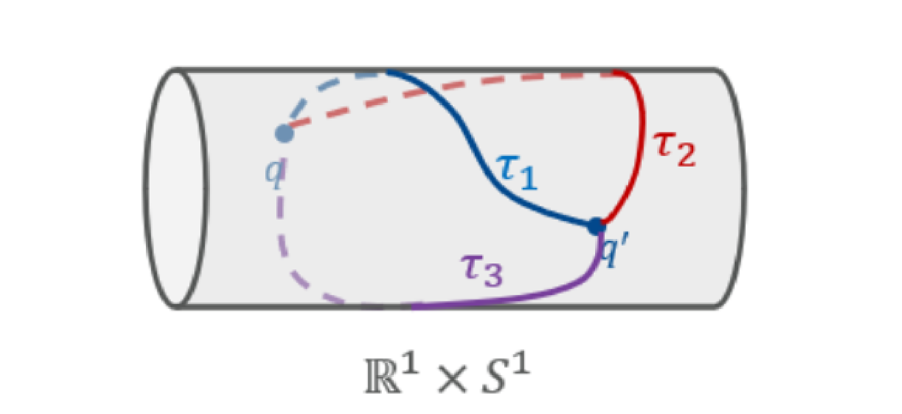
How many homotopy classes of paths exist on a cylinder surface



$\mathbb{R}^1 \times S^1$

- 2
- 3
- 1
- None of the above

How many homotopy classes of paths exist on a cylinder surfance



2

0%

3

0%

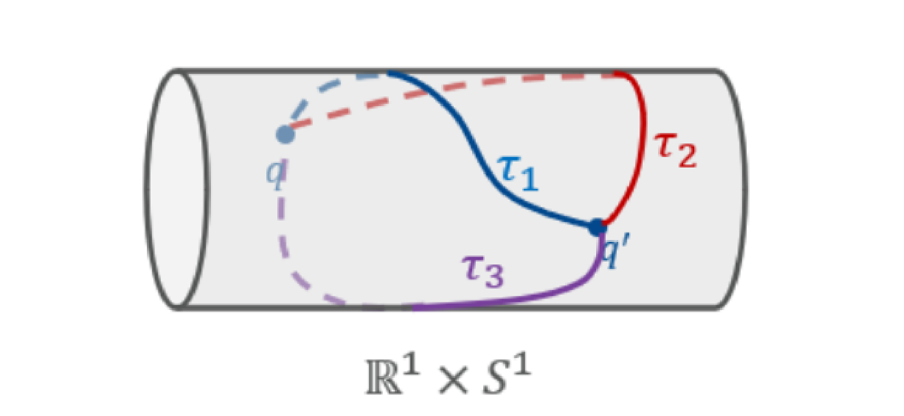
1

0%

None of the above

0%

How many homotopy classes of paths exist on a cylinder surfance



2

0%

3

0%

1

0%

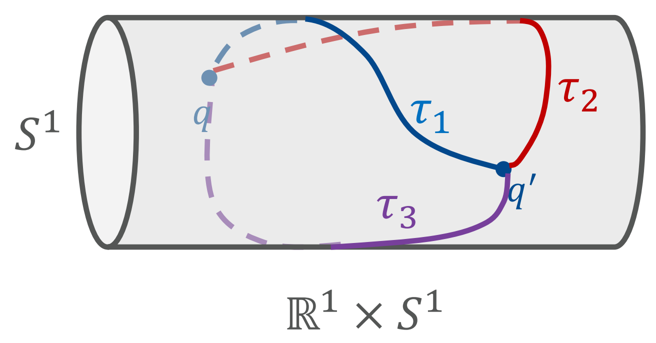
None of the above

0%

Homotopic Paths

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

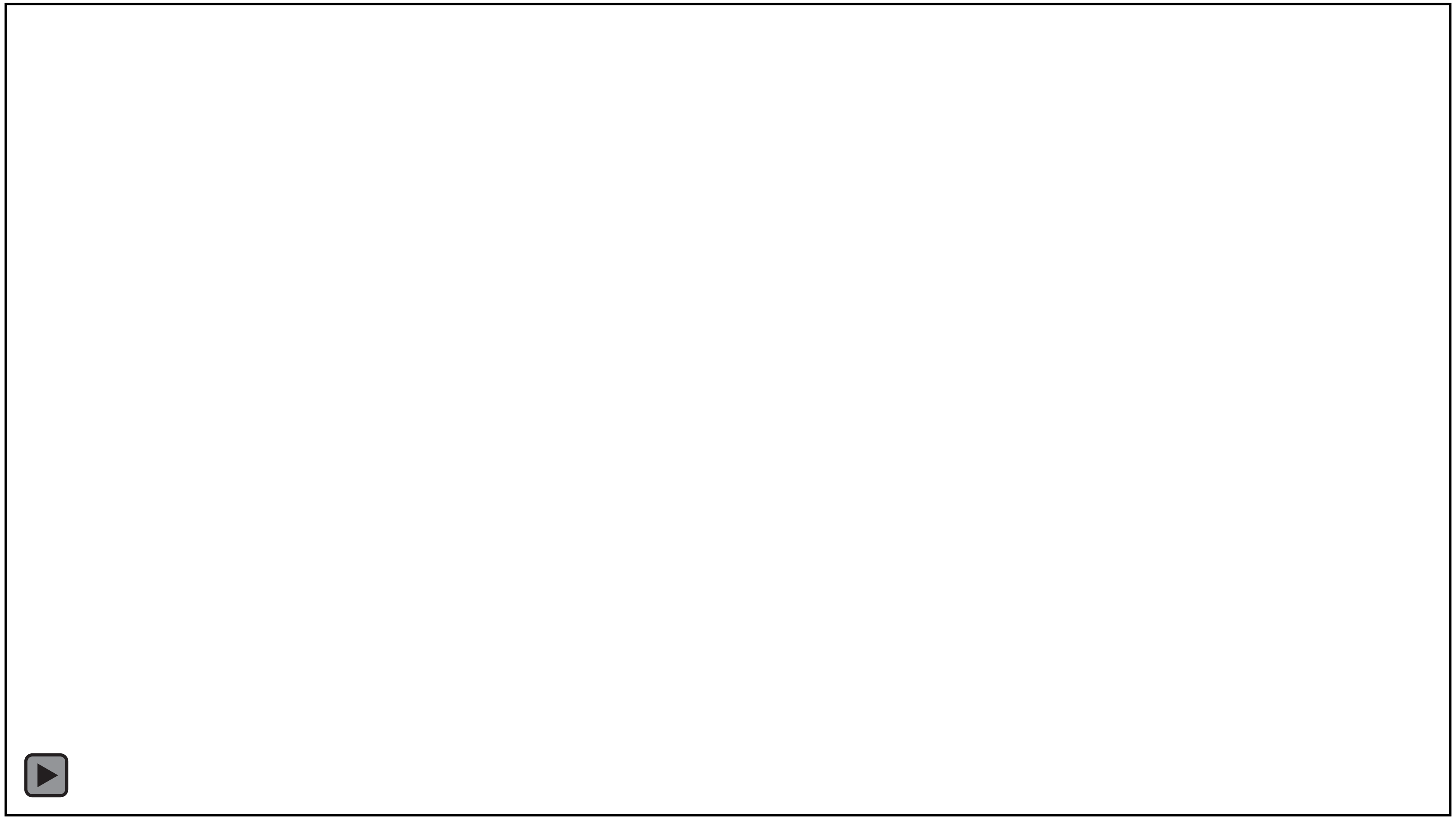
- Example: $\mathbb{R}^1 \times S^1$



  - τ_1 and τ_2 are homotopic
  - τ_1 and τ_3 are not homotopic
  - In this example, there is an infinite number of homotopy classes

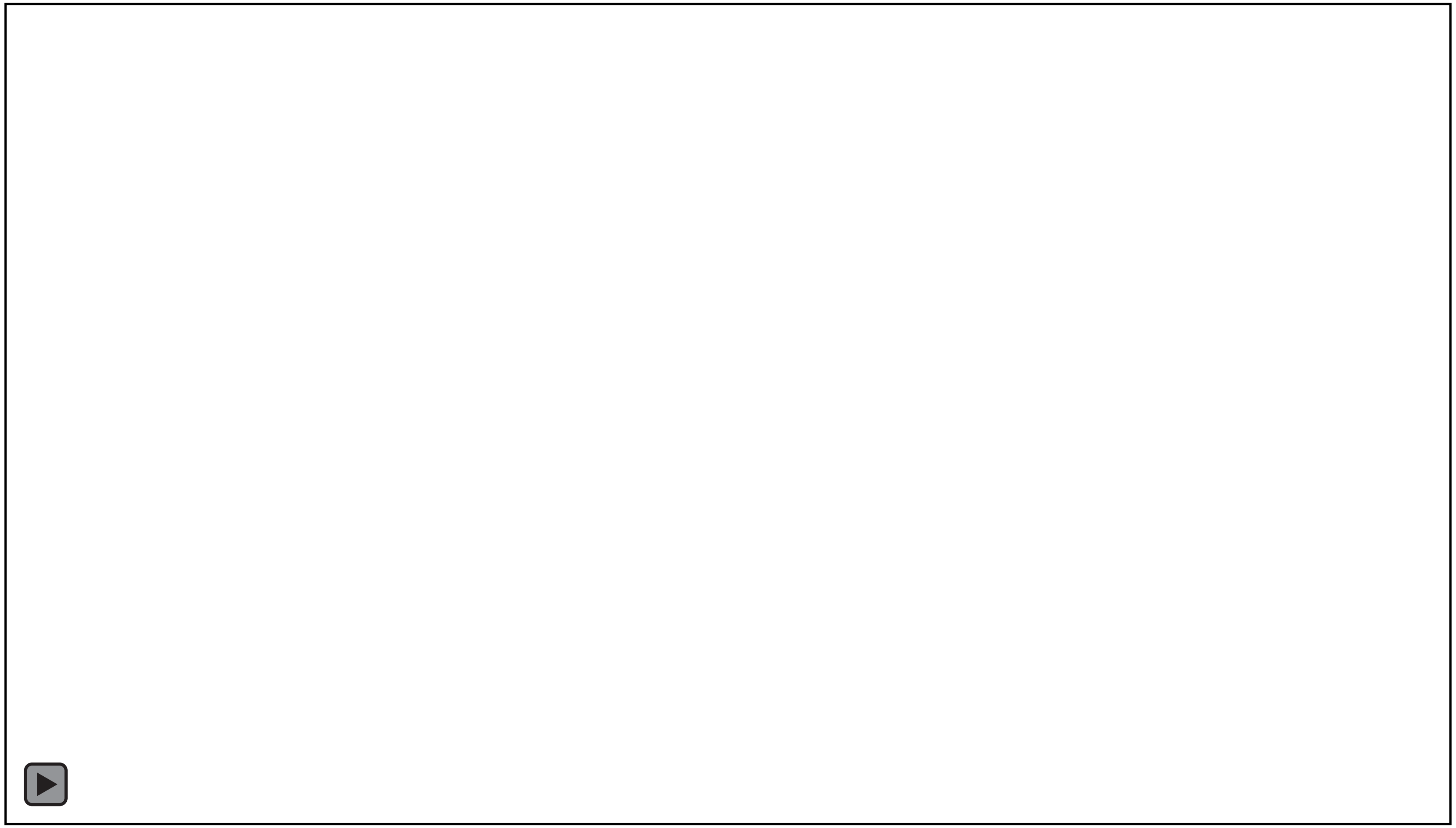
Homotopic Paths

- Homotopic paths in $\mathbb{R}^1 \times S^1$



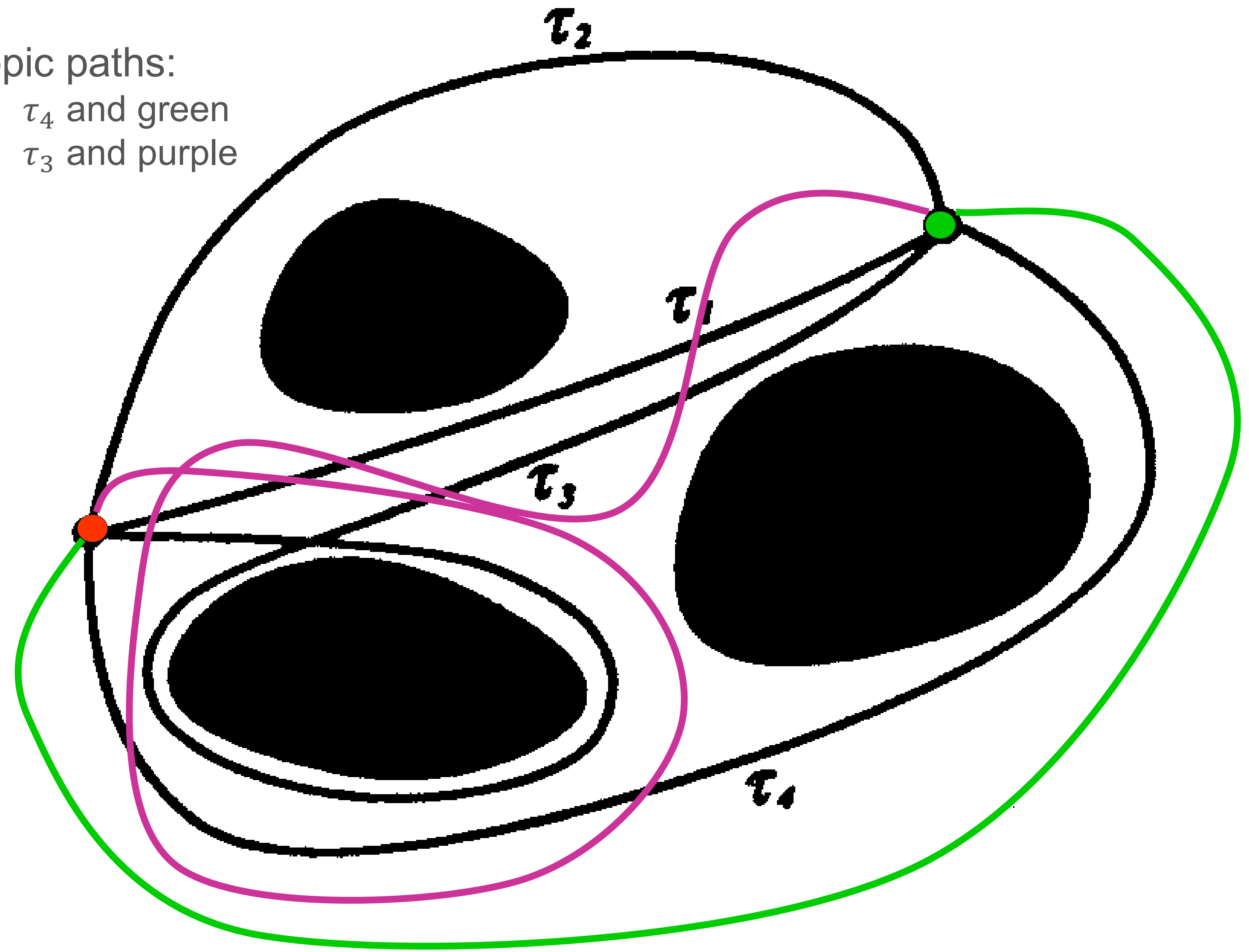
Homotopic Paths

- Non-homotopic paths in $\mathbb{R}^1 \times S^1$



Homotopic Paths

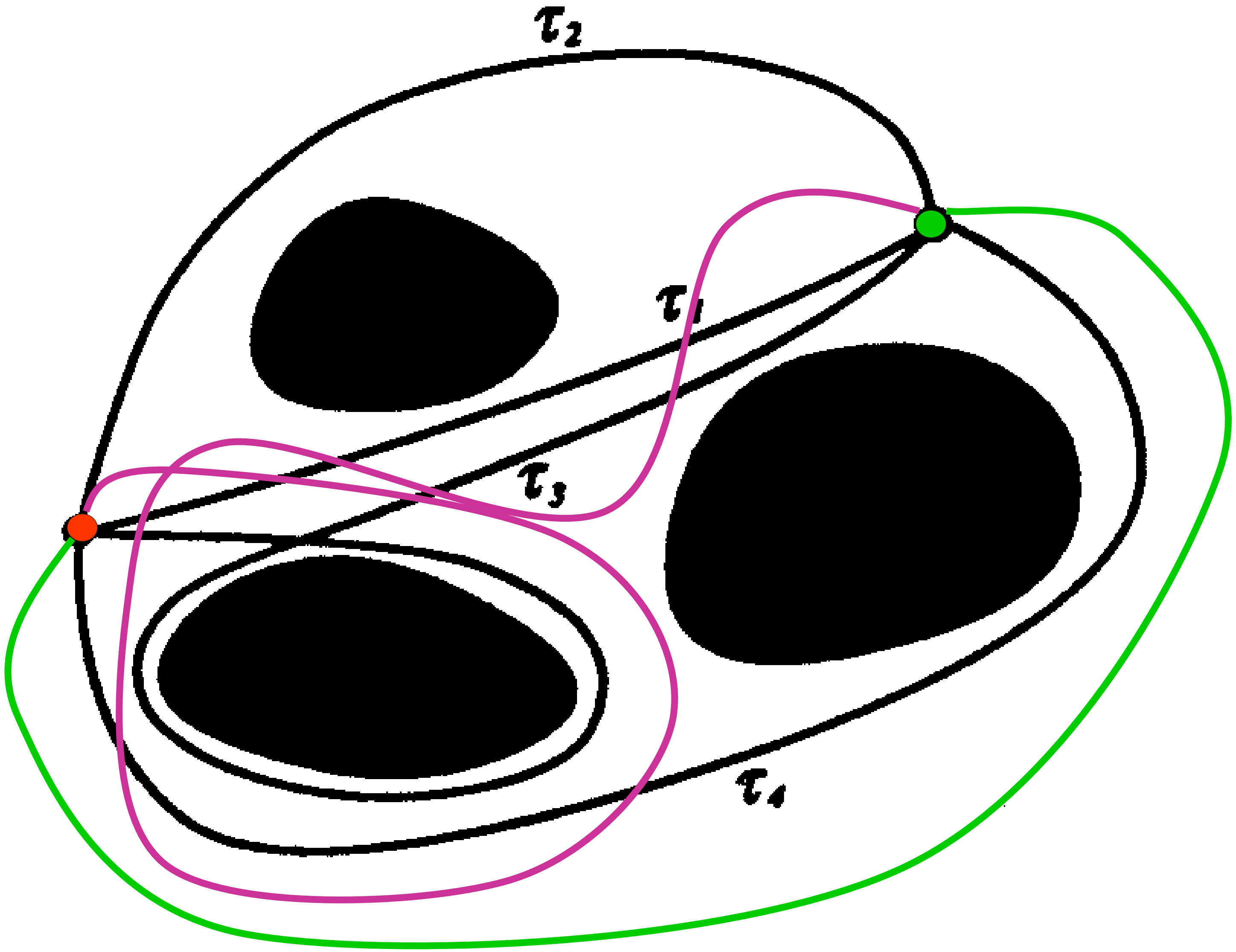
- Homotopic paths:
  - τ_4 and green
  - τ_3 and purple



Connectedness of C-Space

- C is connected if every two configurations can be connected by a path.
- C is simply-connected if every two paths connecting the same endpoints are homotopic.
  - Examples: $\mathbb{R}^2$ or $\mathbb{R}^3$
- Otherwise C is multiply-connected.

Paths and Connectedness of C-Space



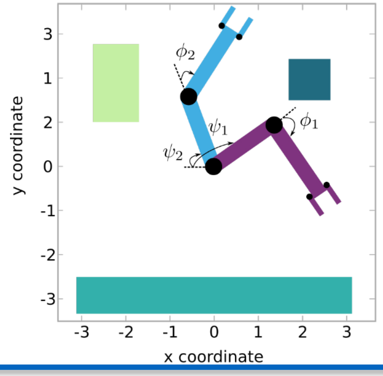
Today’s Overview

1. C-Space Idea
2. Topology of C-space
3. C-space Obstacles
4. C-Space Paths

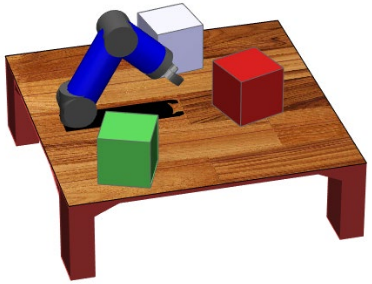
Configuration spaces for high-Dof Robots

Workspace

2-Dof Robot

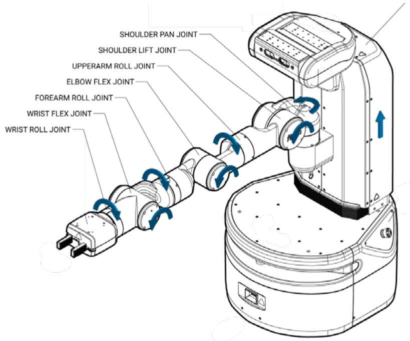


3-Dof Robot

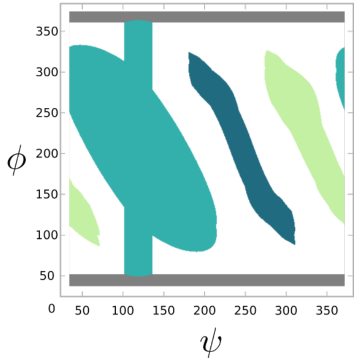


Das et al, *TRO* 2020

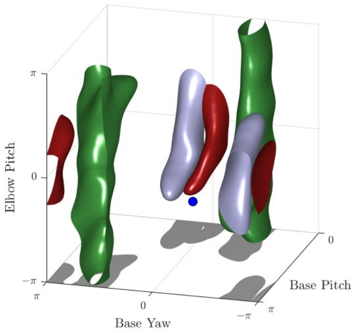
8-Dof Robot



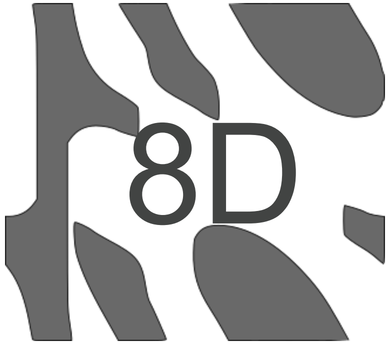
Configuration Space



2 Dimensions



3 Dimensions



8 Dimensions

Can we compute these high-Dof obstacles?