Robot Motion Control and Planning

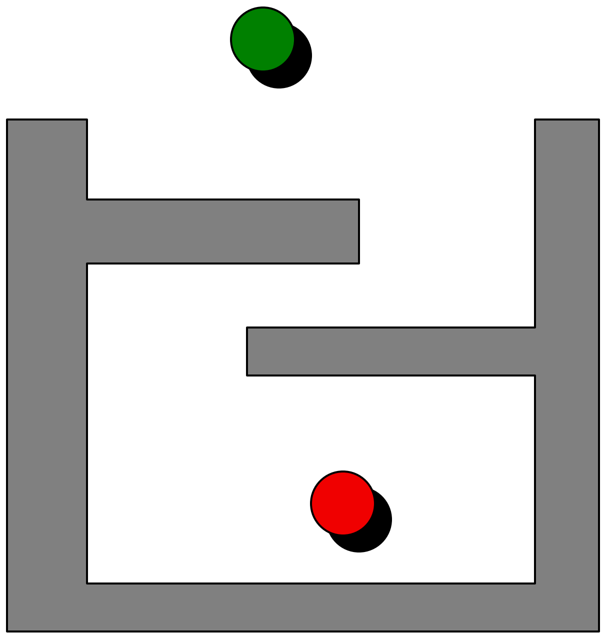
http://www.ceng.metu.edu.tr/~saranli/courses/ceng786

Lecture 3 – Configuration Space

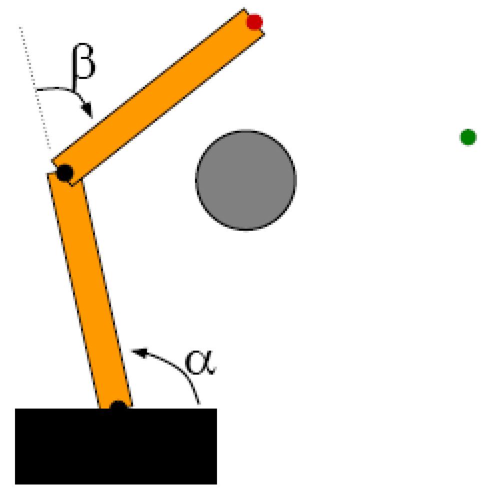
Uluç Saranlı

http://www.ceng.metu.edu.tr/~saranli

What if the Robot is not a Point?



A large wheeled robot should
probably not
be modeled as a point...



Nor should robots with extended
linkages that may contact obstacles

What is the *position of the robot* in such situations?

Configuration Space

- A key concept for motion planning is a **configuration**:
  - *a complete specification of the position of every point in the system*
- A simple example: a robot that translates but does not rotate in the plane:
  - what is a sufficient representation of its configuration?
- The space of all configurations is the **configuration space** or **Cspace.**
- **Workspace** is either the ambient space, or the set of reachable points by an end-effector

C-space formalism:
Lozano-Perez '79

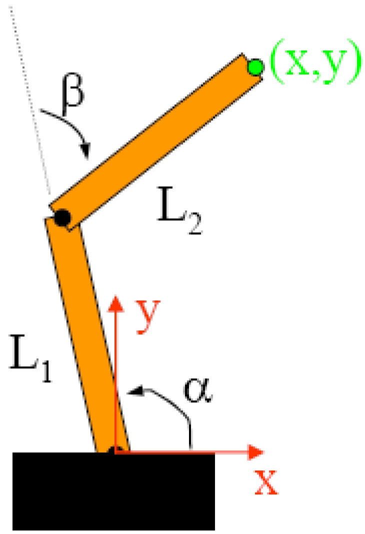
Some Other Examples of C-Space

- A rotating bar fixed at a point
  - what is its C-space?
  - what is its workspace?

- A rotating bar that translates along the rotation axis
  - what is its C-space?
  - what is its workspace?

- A two-link manipulator
  - what is its C-space?

Robot Manipulators



Keeping it "simple"

$c_\alpha = \cos(\alpha)$, $s_\alpha = \sin(\alpha)$

$c_\beta = \cos(\beta)$, $s_\beta = \sin(\beta)$

$c_+ = \cos(\alpha+\beta)$, $s_+ = \sin(\alpha+\beta)$

- What are this arm's forward kinematics?
  - i.e. How does its position depend on its joint angles?

  Find (x,y) in terms of α and β

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} L_1 c_\alpha \\ L_1 s_\alpha \end{pmatrix} + \begin{pmatrix} L_2 c_+ \\ L_2 s_+ \end{pmatrix} \quad \text{Position}$$

- Inverse kinematics?
  - Finding joint angles from Cartesian coordinates
  - Algebraic or geometric approaches

  Given (x,y), what are the values of α and β

$$\gamma = \cos^{-1}\left(\frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}\right)$$

$$\beta = 180 - \gamma$$

$$\alpha = \sin^{-1}\left(\frac{L_2 \sin(\gamma)}{x^2 + y^2}\right) + \tan^{-1}(y/x)$$

atan2(y,x)

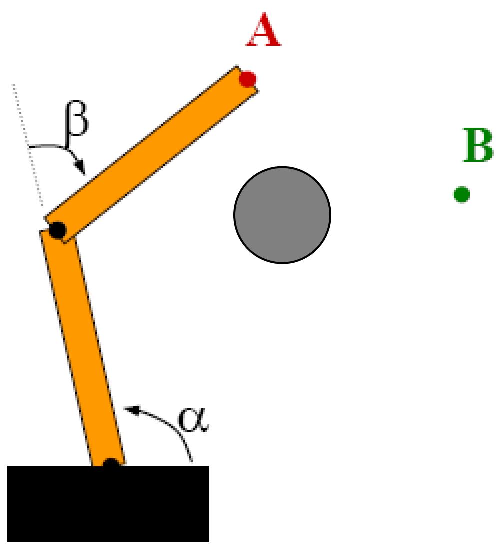
Some Other Examples of C-Space

- A rotating bar fixed at a point
  - what is its C-space?
  - what is its workspace?

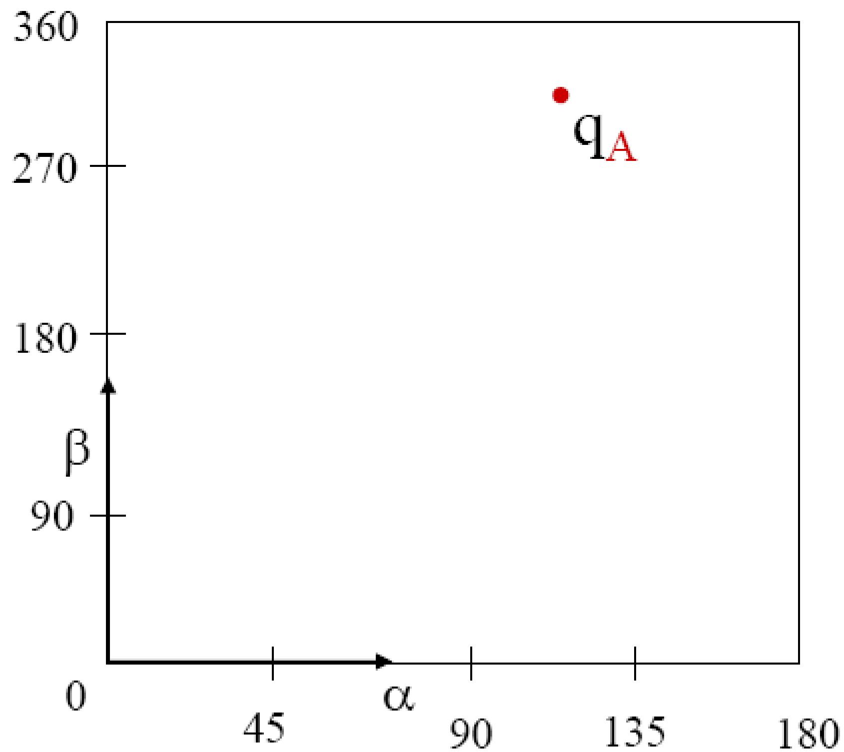
- A rotating bar that translates along the rotation axis
  - what is its C-space?
  - what is its workspace?

- A two-link manipulator
  - what is its C-space?
  - what is its workspace?
  - Suppose there are joint limits, does this change the C-space?
  - The workspace?

Configuration Space



Put an obstacle in the robot's workspace



Torus

(wraps horizontally and vertically)

Where can we put q_B ?

Obstacles in C-Space

- Let q denote a point in a configuration space Q
- The path planning problem is to find a mapping c:[0,1]→ Q s.t. no configuration along the path intersects an obstacle
- Recall a workspace obstacle is WO_i
- A *configuration space obstacle* QO_i is the set of configurations q at which the robot intersects WO_i, *that is*

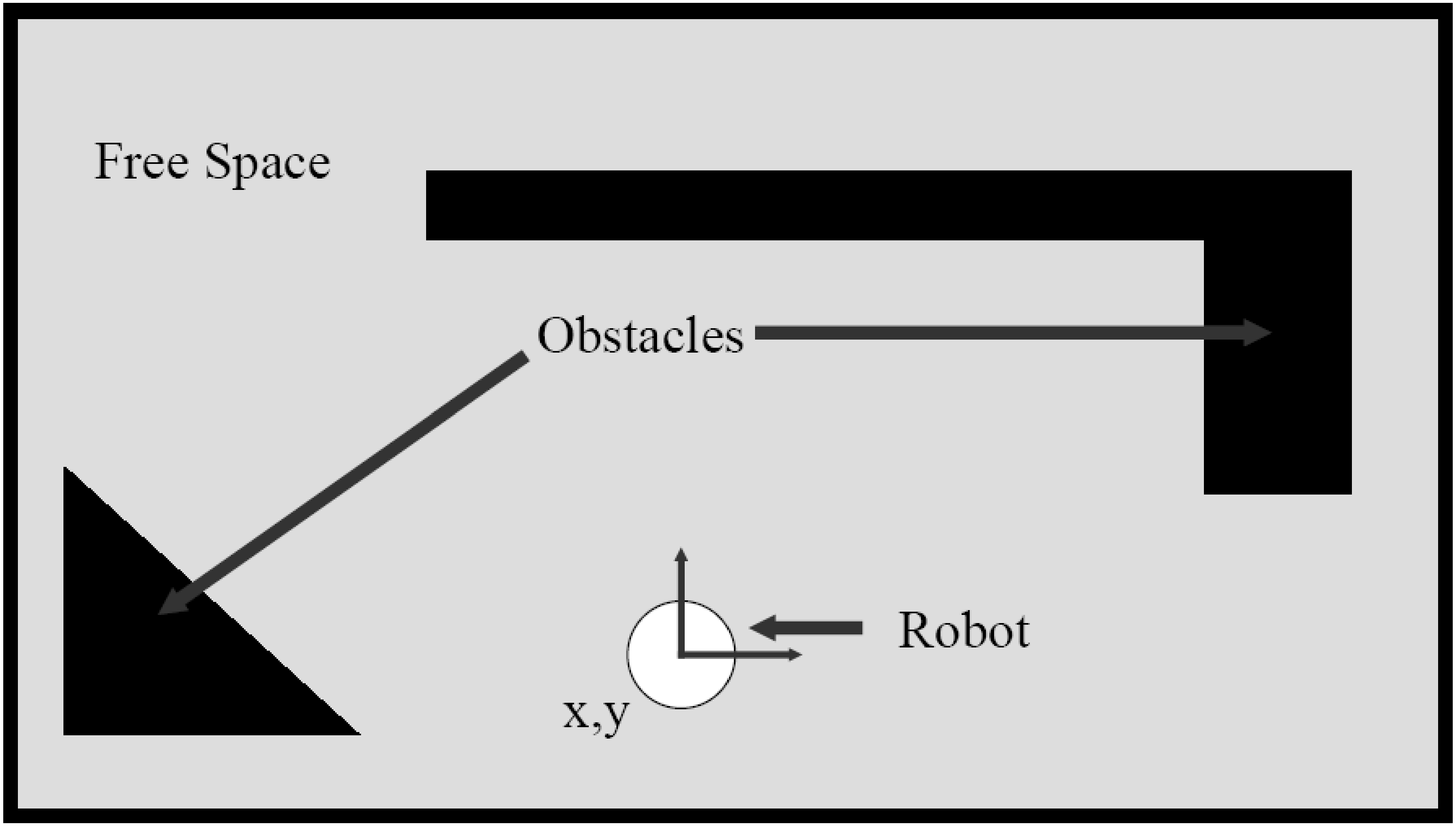
$$QO_i = \{q \in Q | R(q) \bigcap WO_i \neq 0\}$$

- The *free configuration space* (or just *free space*) Q_{free} *is*

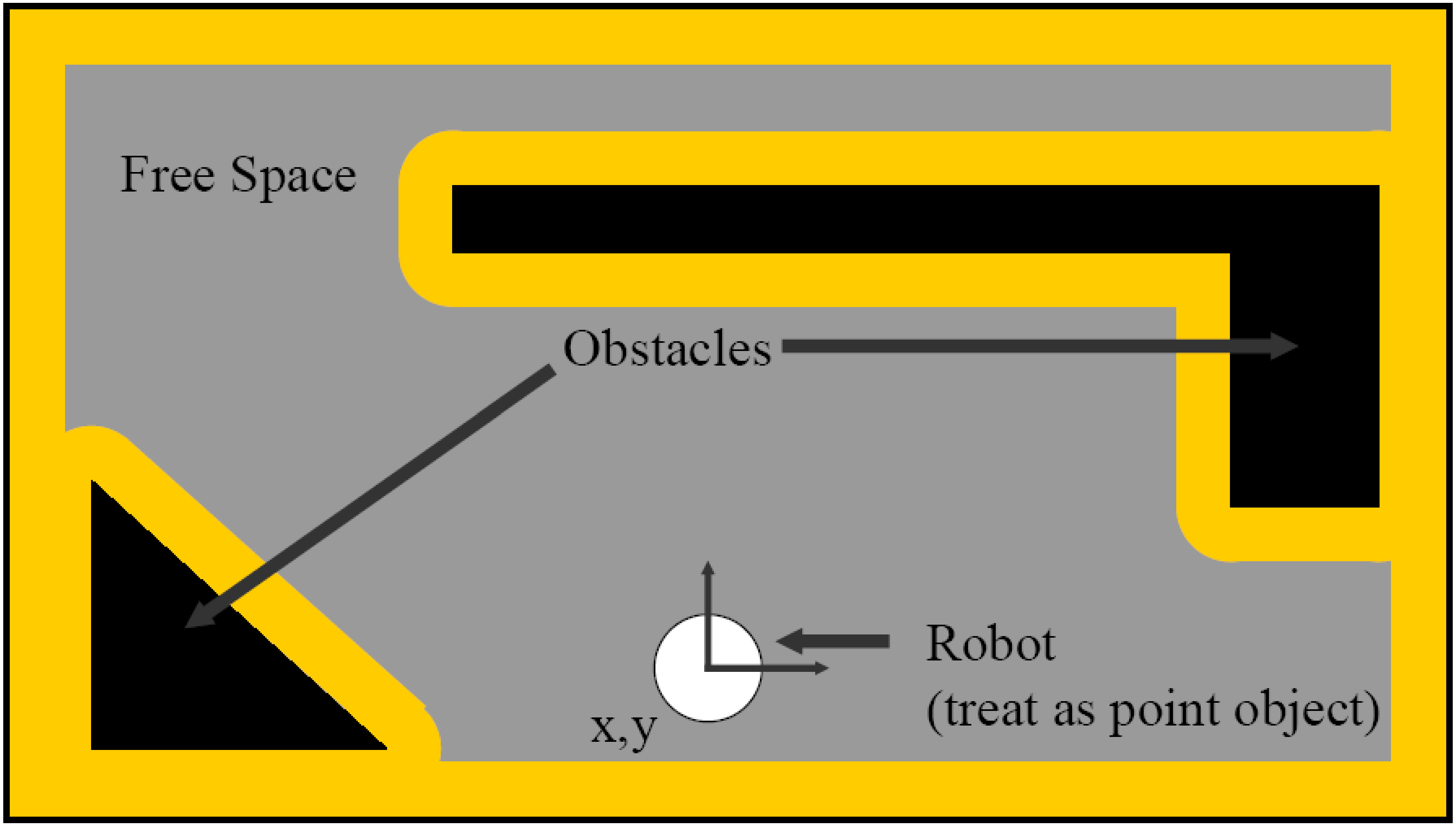
$$Q_{\text{free}} = Q - (\bigcup QO_i)$$

- The free space is generally an open set
- A *free path* is a mapping c:[0,1]→ Q_{free}
- A *semifree path* is a mapping c:[0,1]→ cl(Q_{free}) (cl stands for closure)

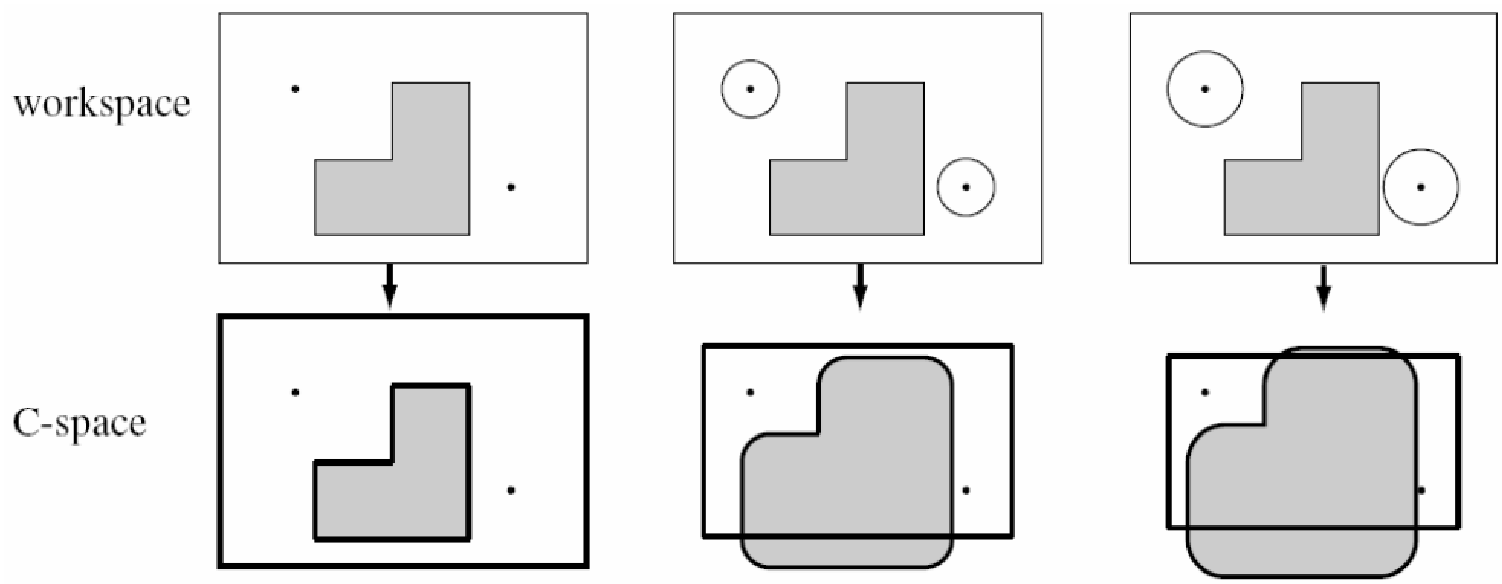
Example World (Circular Robot)



Configuration Space (accomodates robot size)



Trace The Boundary of the Workspace



$$QO_i = \{q \in Q | R(q) \bigcap WO_i \neq 0\}$$

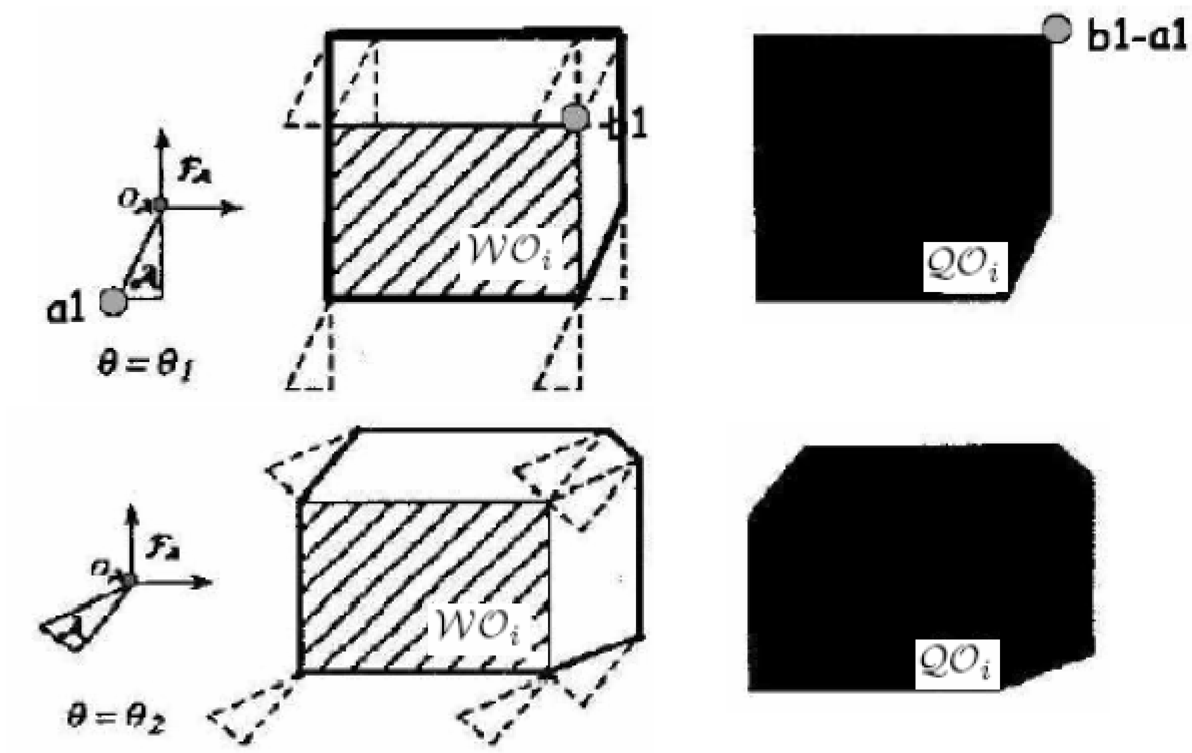
A consistent reference point must be picked on the robot!

What about non-circular robots?

Robots with both position and body angle?

When Only Translation is Allowed

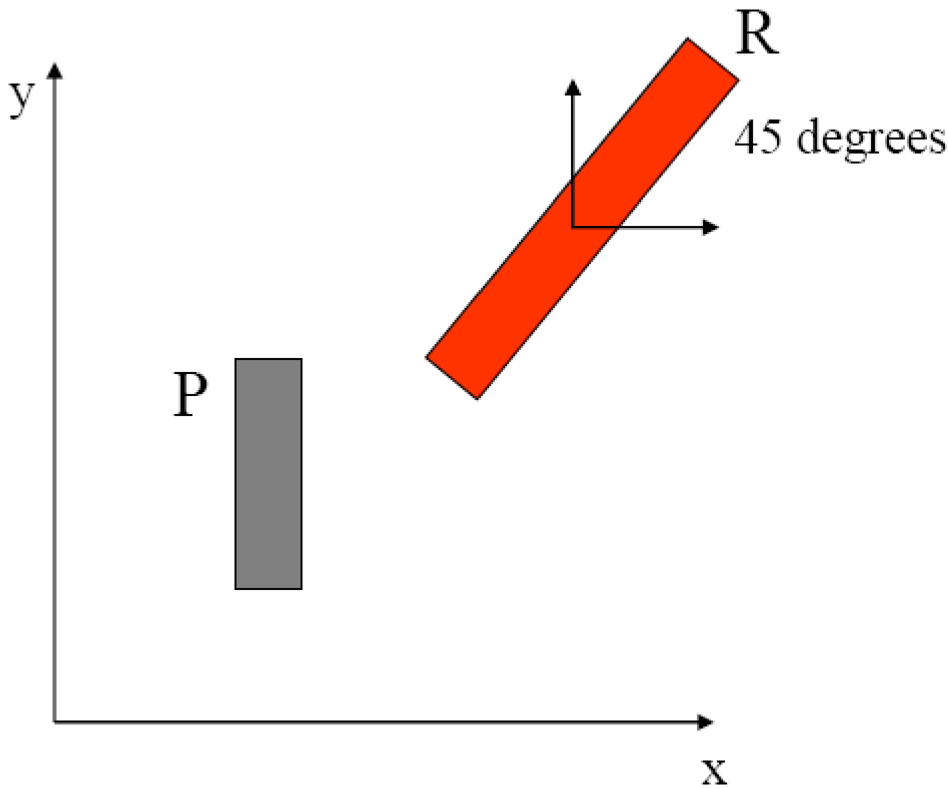
For a fixed robot angle, we can build QO_i
Choice of reference point makes a difference

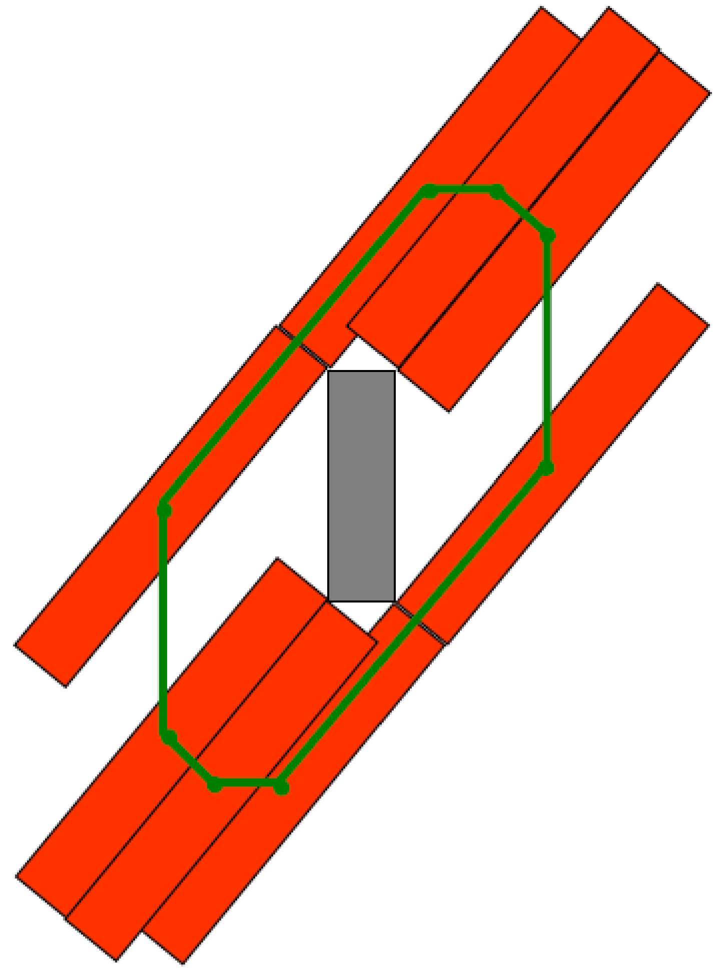


$$QO_i = \{q \in Q | R(q) \bigcap WO_i \neq 0\}$$

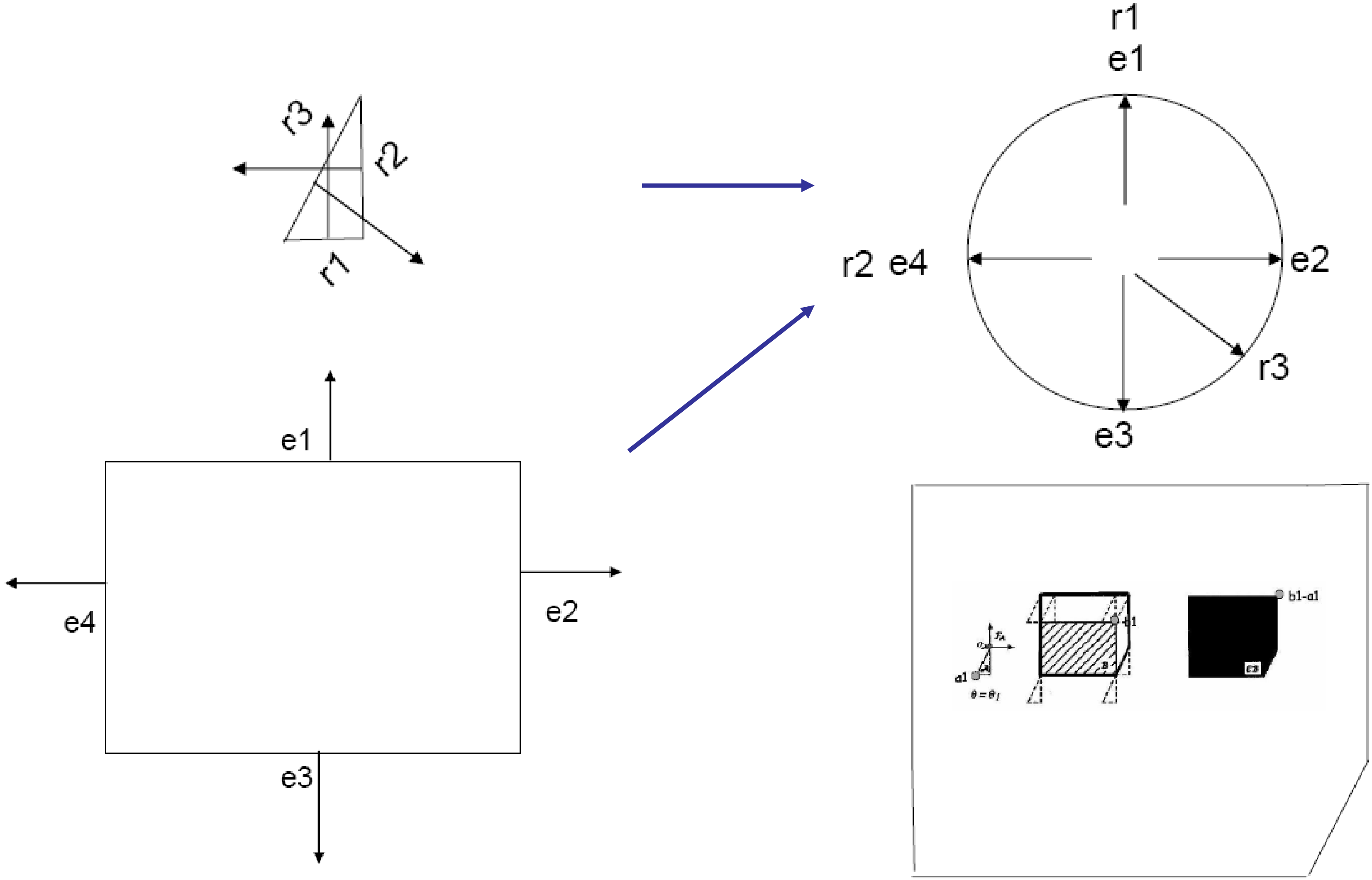
Cross Section of the C-Space

Assuming a fixed angle of 45 degrees, we are taking a cross section of the C-space

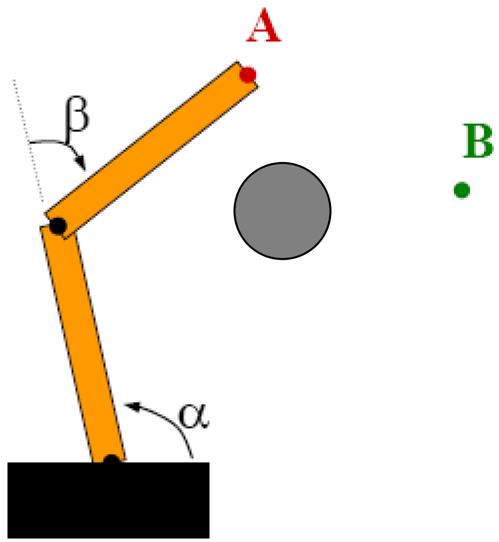




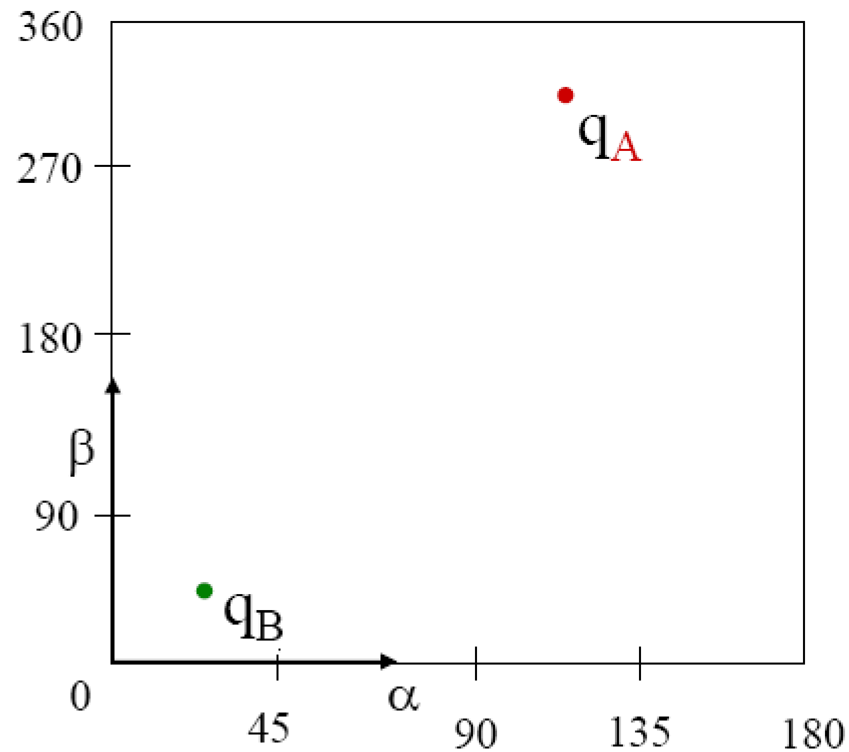
Star Algorithm: Polygonal Obstacles



Obstacles for a Manipulator Arm

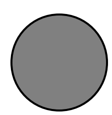


Obstacle in the robot's workspace

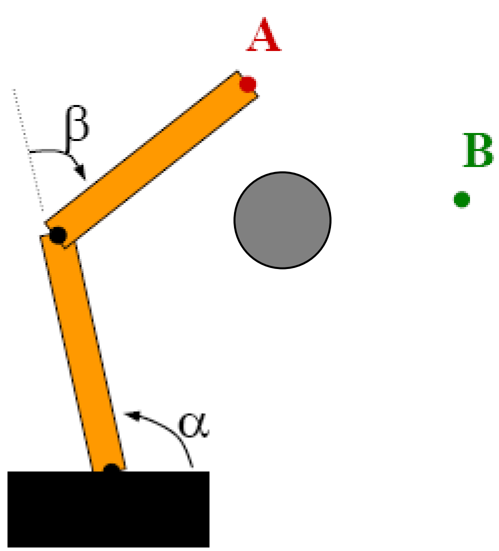


Torus

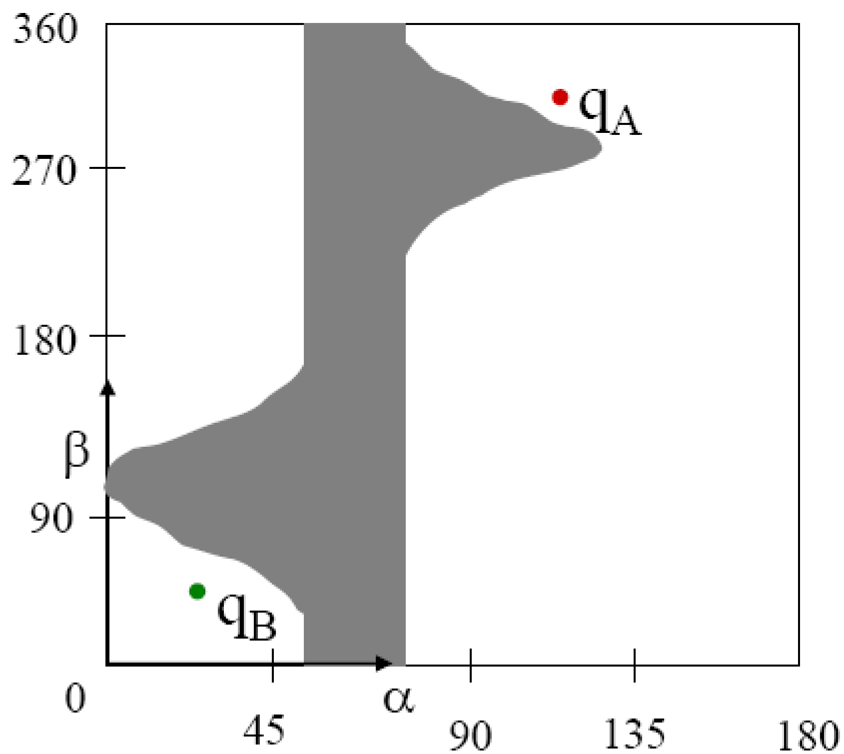
(wraps horizontally and vertically)

Where can we put the  ?

Configuration Space Obstacle



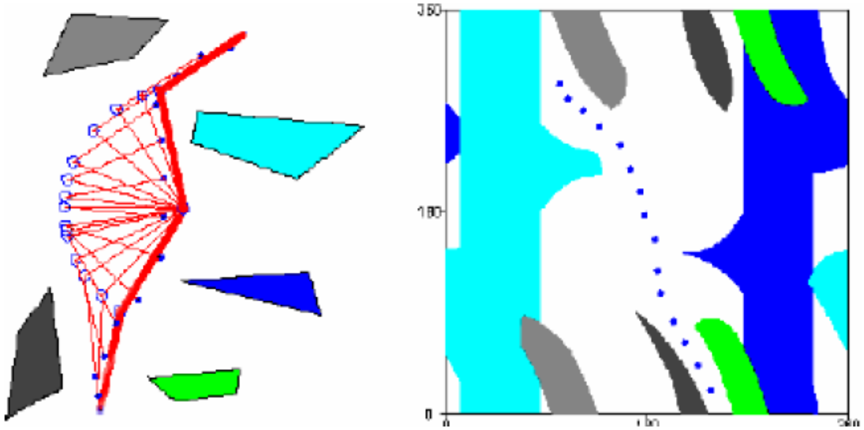
Obstacle in the robot's workspace

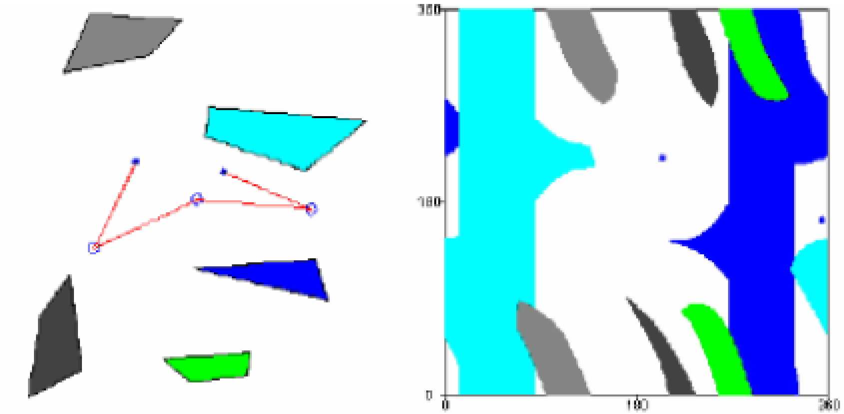


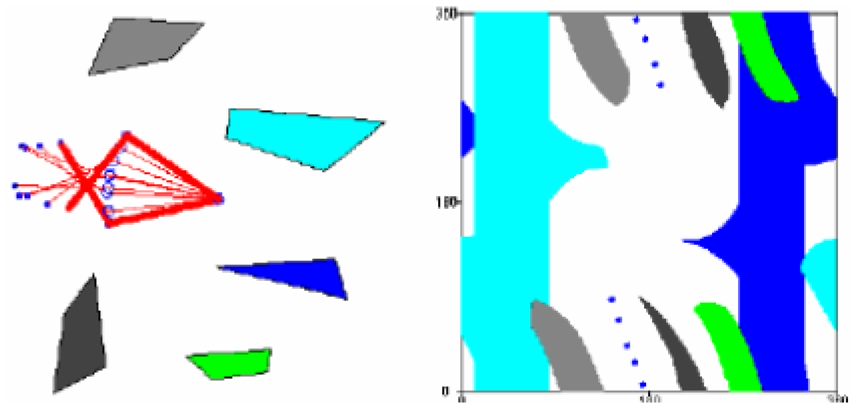
How do we get from A to B?

The C-space representation of this obstacle…

Two-Link Path





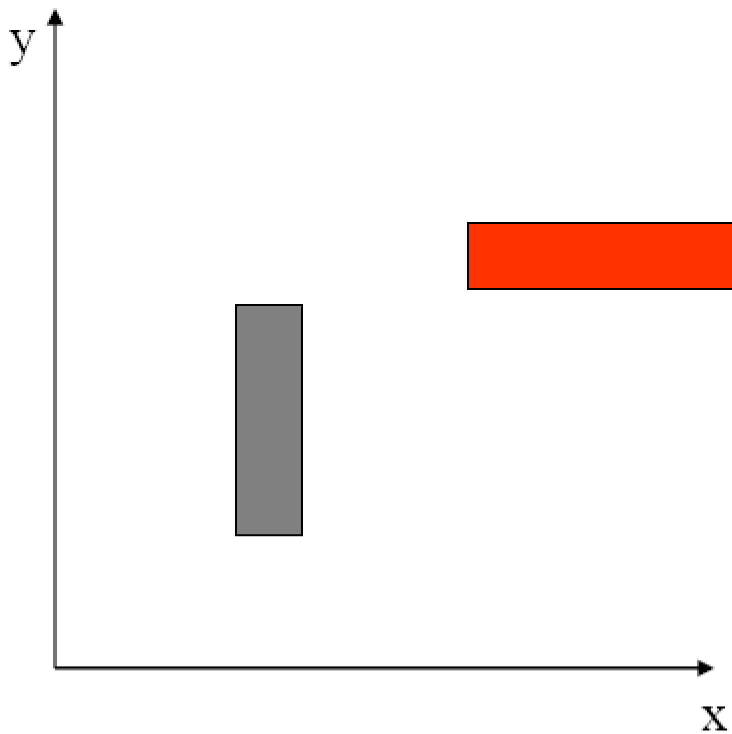


Properties of C-space Obstacles

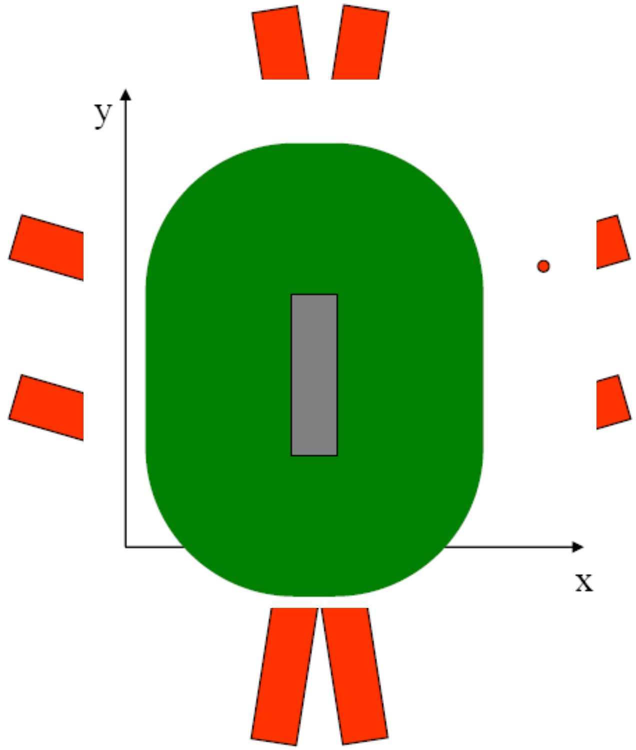
- If the robot and the WO_i are _________ then ______
  - Convex, then QO_i are convex
  - Closed, then QO_i are closed
  - Compact, then QO_i are compact
  - Algebraic, then QO_i are algebraic
  - Connected, then QO_i are connected

Additional Dimensions

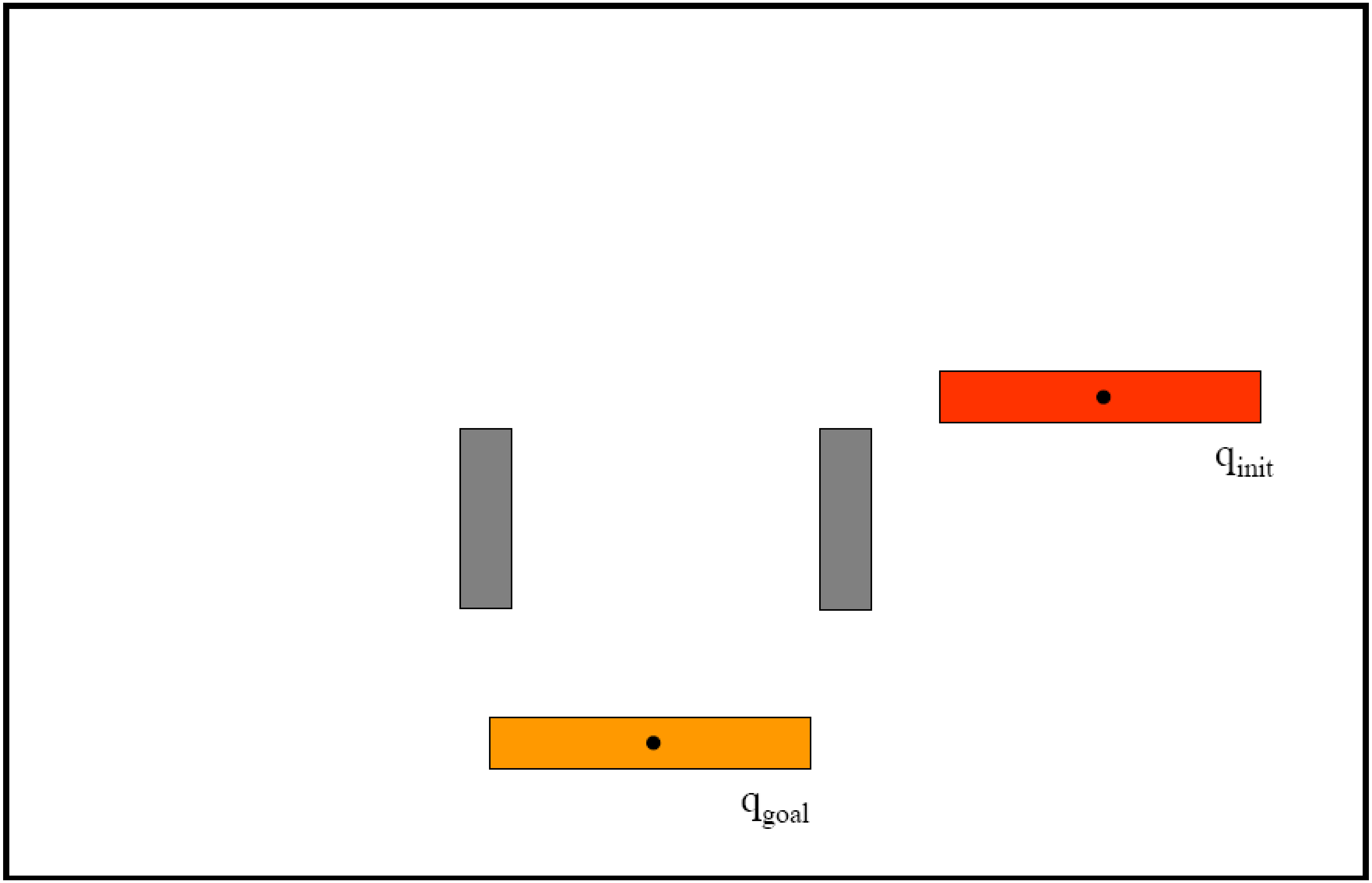
If the robot can both translate and rotate,
What would the configuration of the rectangular robot look like?



A 2D possibility

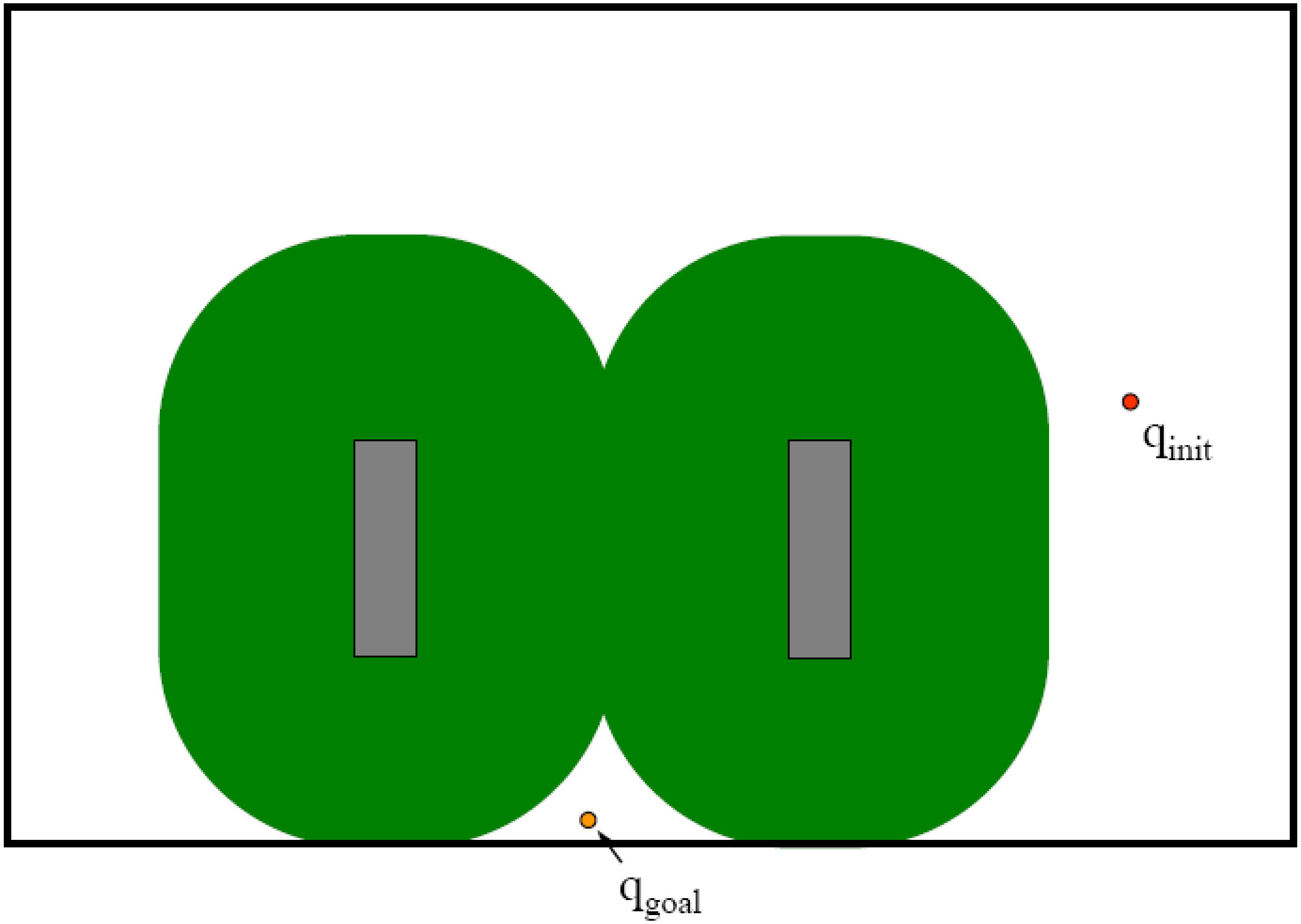


A Serious Problem?

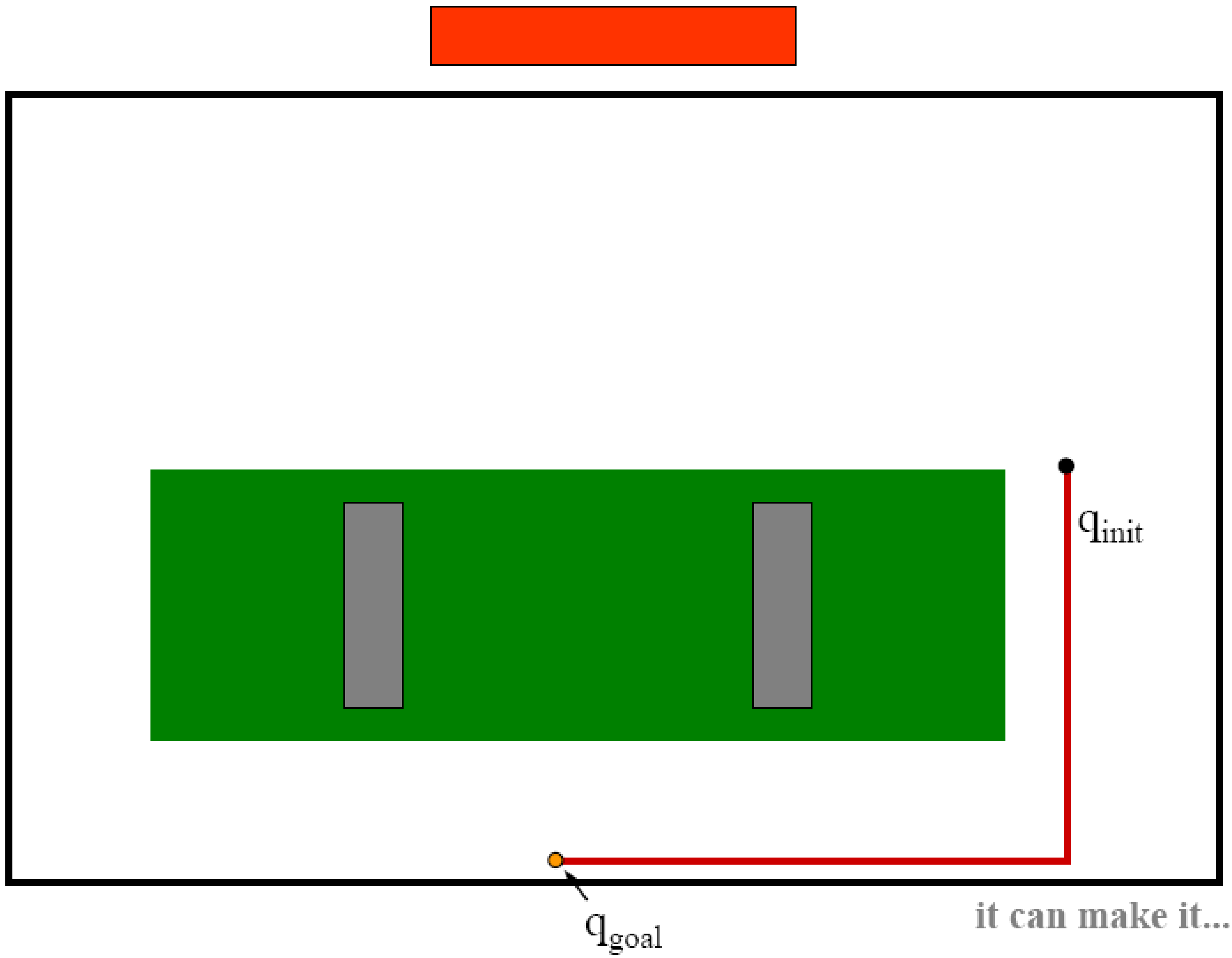


A Serious Problem?

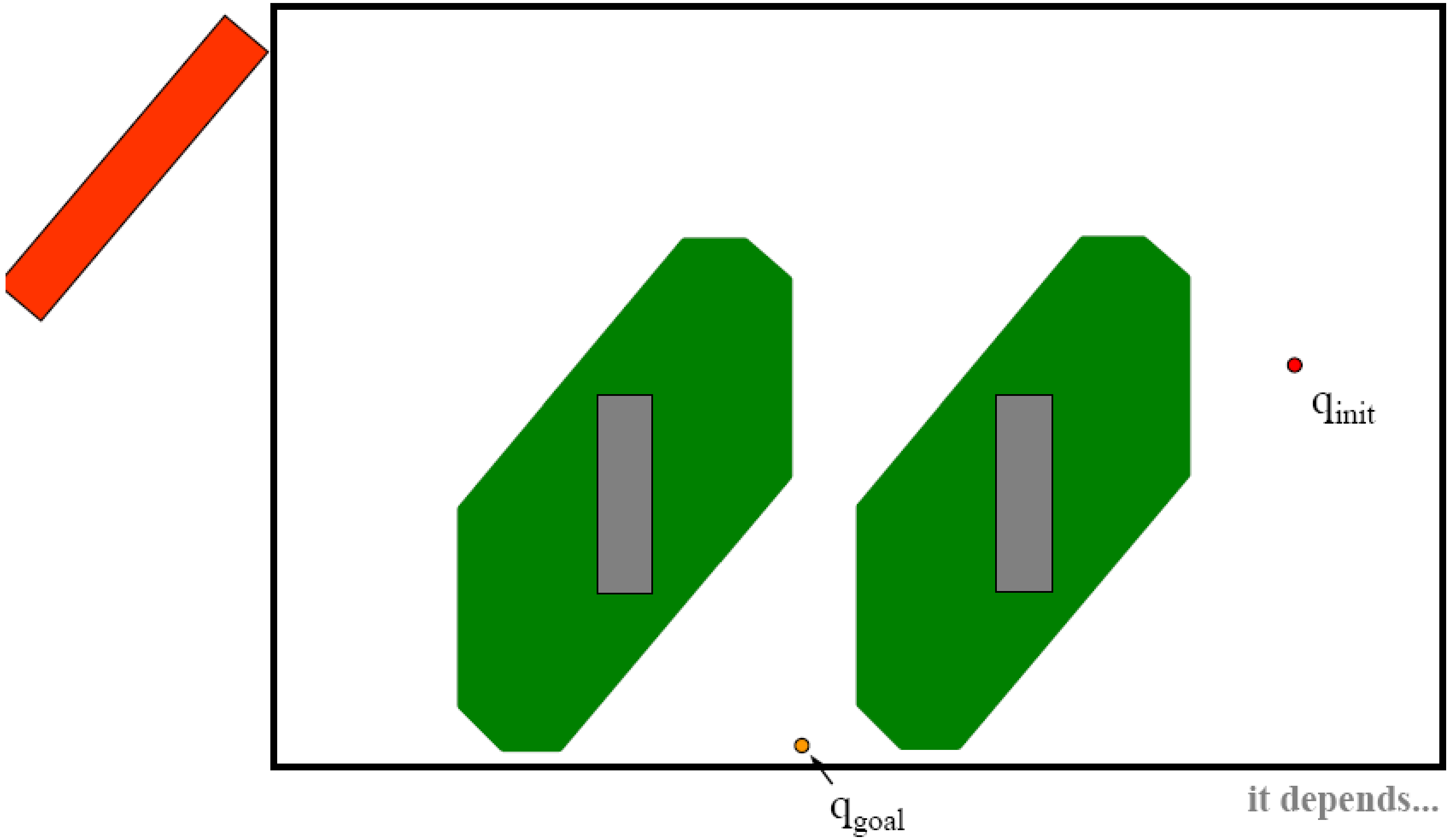
Looks like we need one more dimension (was it obvious?)



When the robot is at one orientation...

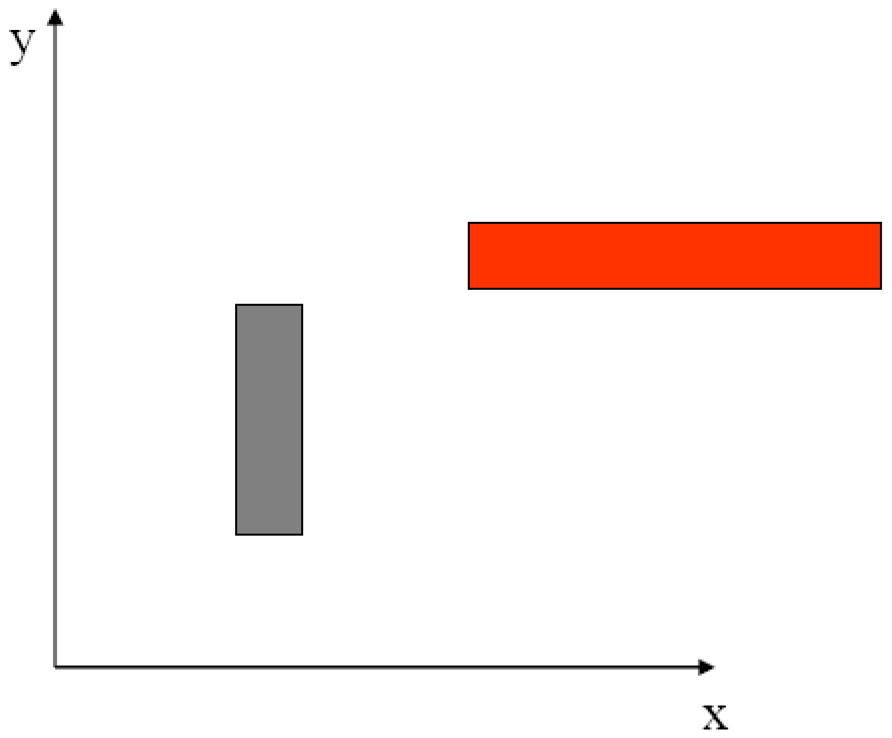


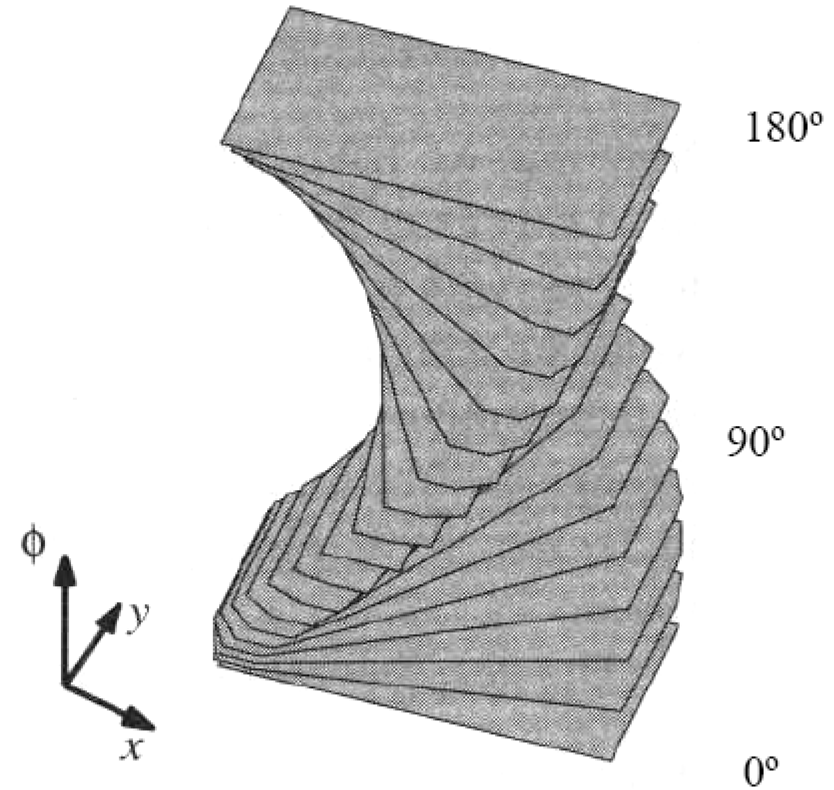
When the robot is at another orientation...



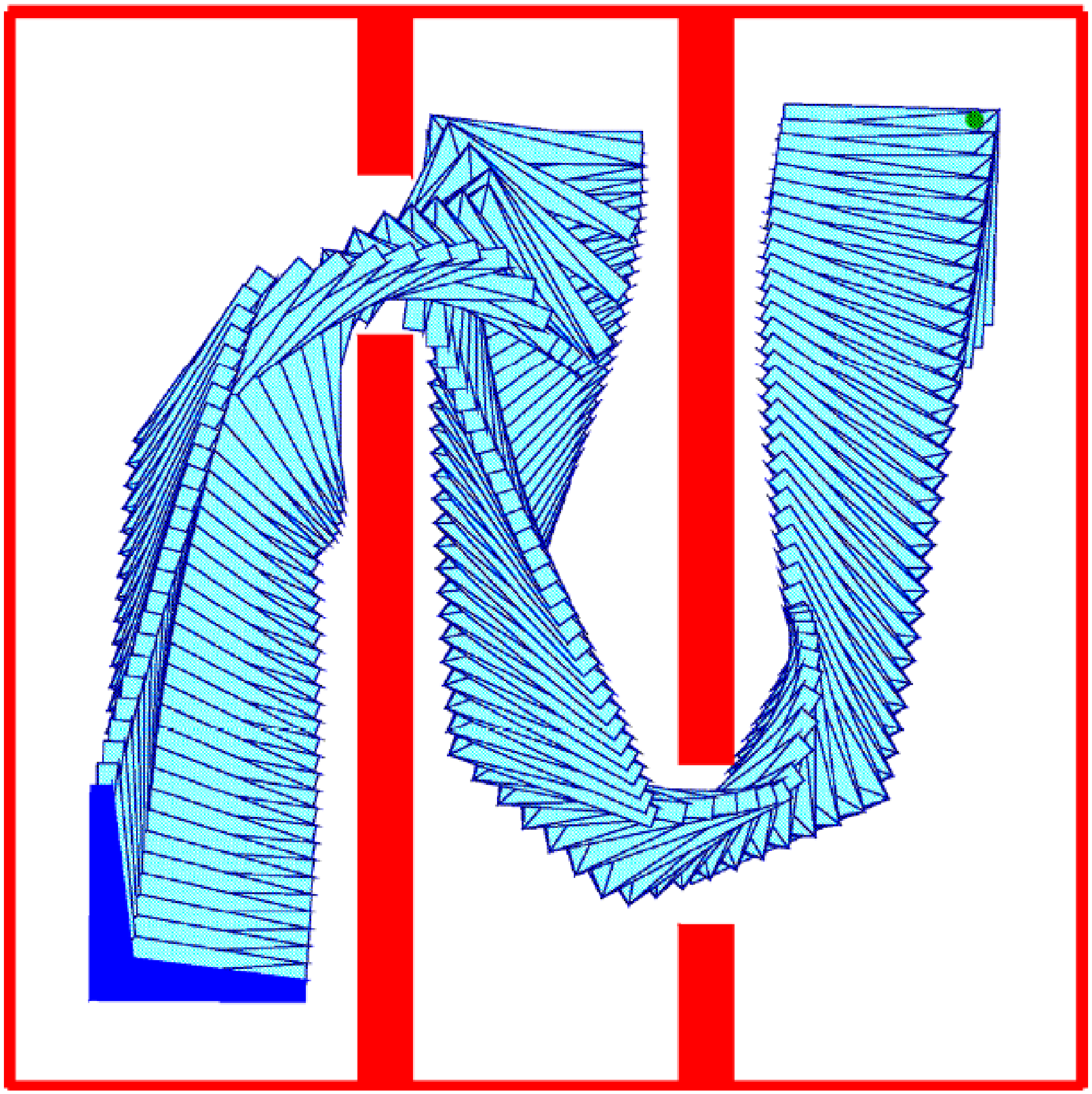
Additional Dimensions

If the robot can both translate and rotate,
What would the configuration of the rectangular robot look like?

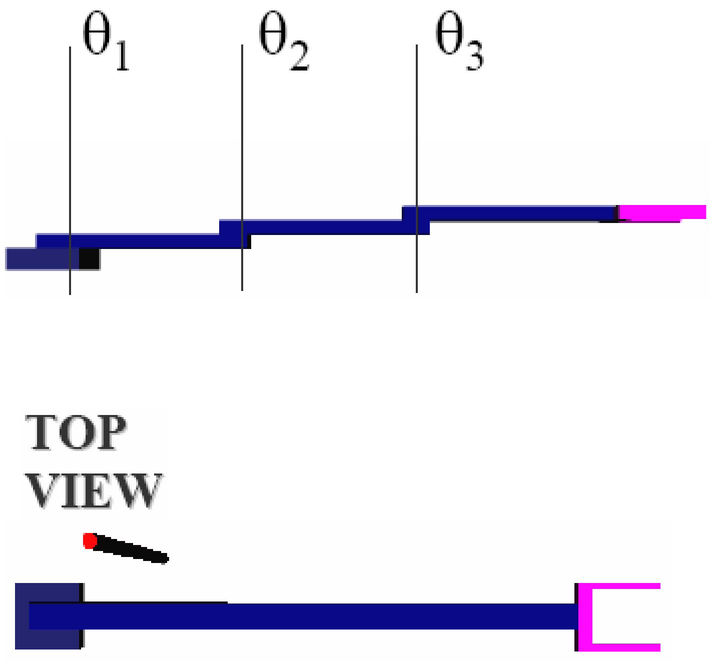




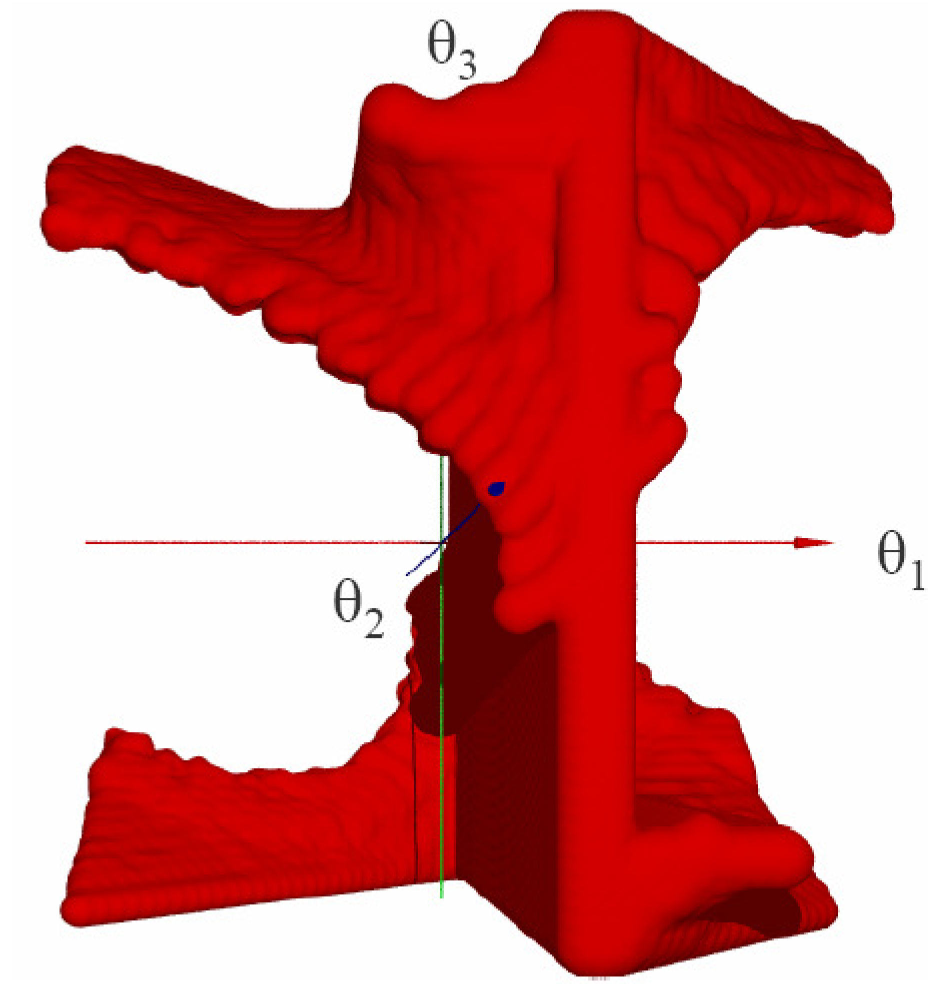
2D Rigid Object



A Planar Robot Arm

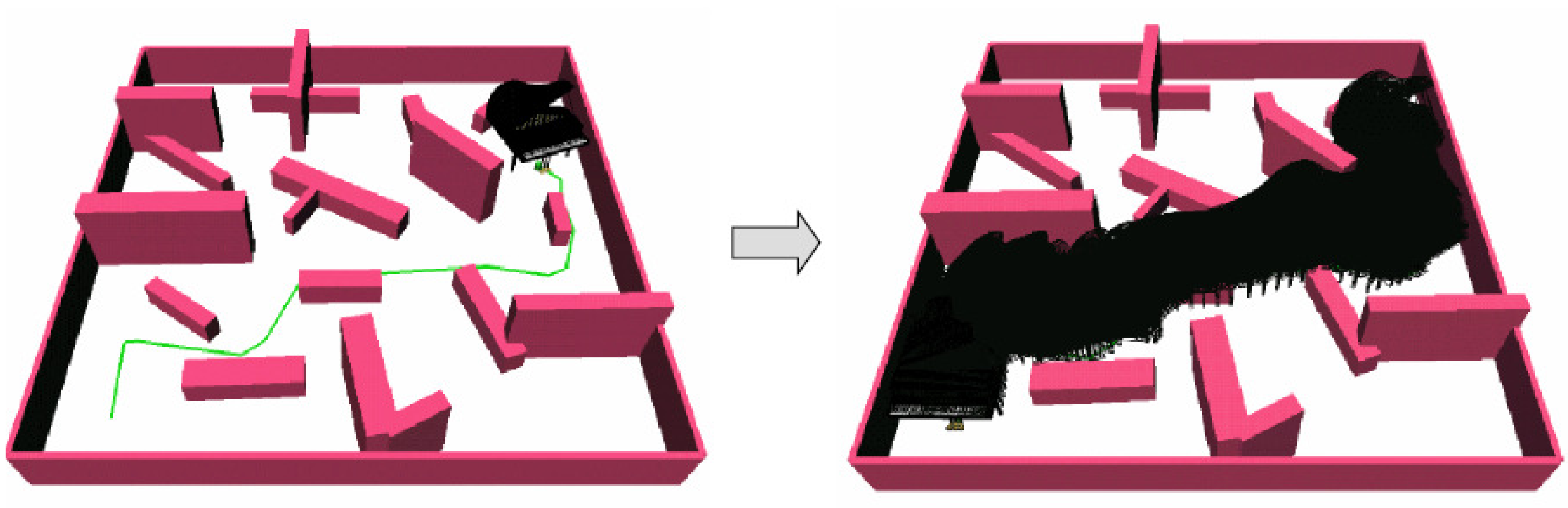


workspace

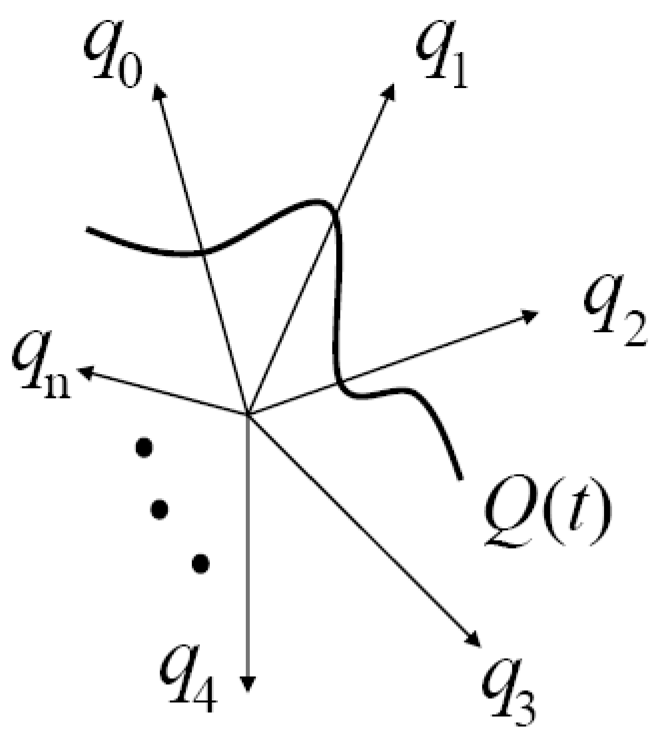


C-space

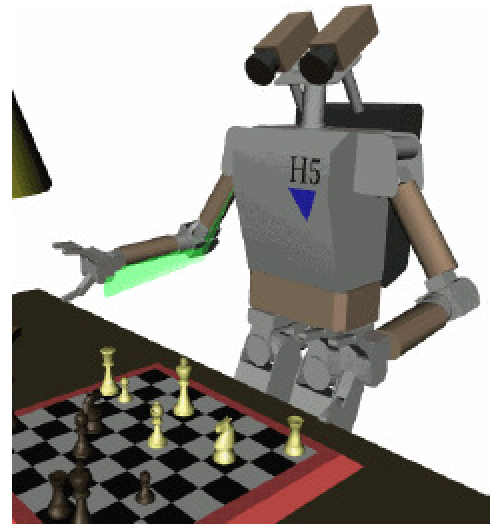
Moving a Piano



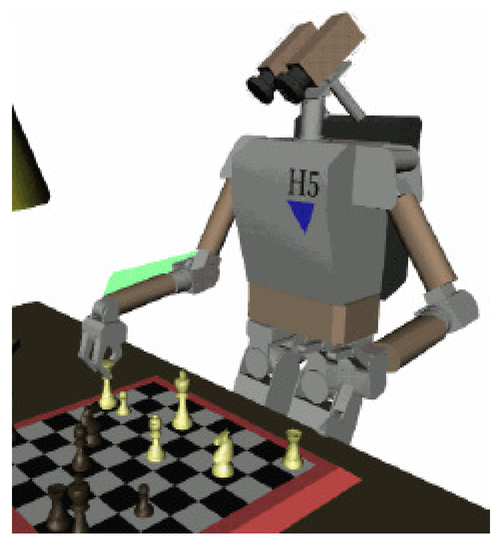
Motion of a Humanoid Robot



$$Q(t) = \begin{bmatrix} q_0(t) \\ \vdots \\ q_n(t) \end{bmatrix} \quad t \in [0, T]$$



INIT:
$Q(0)$



GOAL:
$Q(T)$

The Topology of the Configuration Space

- Topology is the “intrinsic character” of a space

- Two spaces have different topologies if cutting and pasting is required to make them the same (e.g. a sheet of paper vs. a mobius strip)
  - think of rubber figures --- if we can stretch and reshape “continuously” without tearing, one into the other, they have the same topology

- A basic mathematical mechanism for talking about topology is thehomeomorphism.

Why Study Topology?

- Extend results from one space to another: spheres to stars
- Understand and compare different representations
- Know where you are
- Others?

Homeo- and Diffeomorphisms

- Recall mappings:
  - $\varphi: S \rightarrow T$
  - If each element of φ goes to a unique T, φ is *injective* (or 1-1)
  - If each element of T has a corresponding preimage in S, then φ is *surjective* (or onto).
  - If φ is surjective and injective, then it is bijective (in which case an inverse, φ^{-1} exists).
  - φ is *smooth* if derivatives of all orders exist (we say φ is C^{∞})

- If $\varphi: S \rightarrow T$ is a bijection, and both φ and φ^{-1} are continuous, φ is a *homeomorphism;* if such a φ exists, S and T are *homeomorphic.*

- If homeomorphism where both φ and φ^{-1} are smooth is a *diffeomorphism.*

Some Examples

- How would you show a square and a rectangle are diffeomorphic?
- How would you show that a circle and an ellipse are diffeomorphic (implies both are topologically S^1)
- Interestingly, a “racetrack” is not diffeomorphic to a circle
  - composed of two straight segments and two circular segments
  - at the junctions, there is a discontinuity; it is therefore not possible to construct a smooth map!
  - How would you show this (hint, do this for a function on $\Re^1$ and think about the chain rule)
  - Is it homeomorphic?

Local Properties

Ball: $B_\epsilon(p) = \{p' \in \mathcal{M} | d(p, p') < \epsilon\}$

Neighborhood:

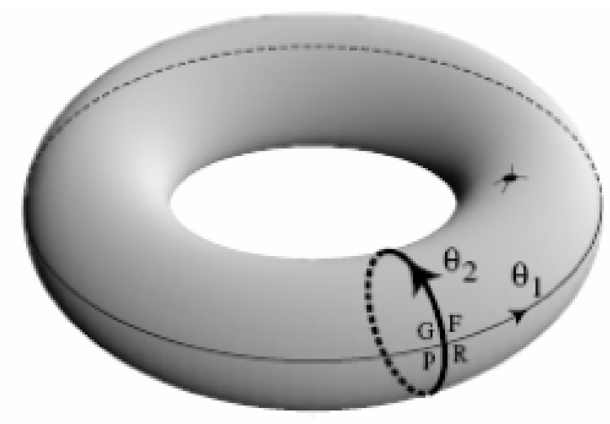
$$p \in \mathcal{M} \quad \mathcal{U} \subseteq \mathcal{M} \;\; \text{with} \;\; p \in \mathcal{U} \;\; \text{such that for every} \;\; p' \in \mathcal{U}, \quad B_\epsilon(p') \subset \mathcal{U}$$

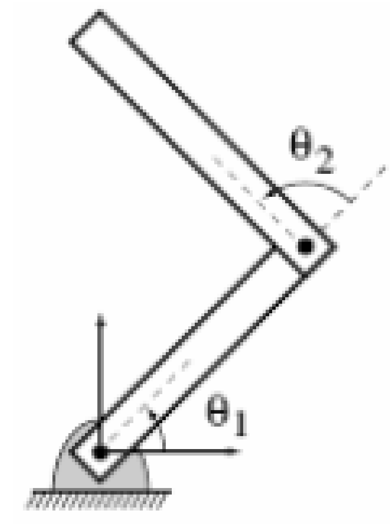
- Manifolds
  - A space S *locally diffeomorphic* (homeomorphic) to a space T if each p in S, there is a neighborhood containing it for which a diffeomorphism (homeomorphism) to some neighborhood of T exists.
  - S^1 is locally diffeomorphic to $\Re^1$
  - The sphere is locally diffeomorphic to the plane (as is the torus)
  - A set S is a *k-dimensional manifold* if it is locally **homeomorphic** to $\Re^k$

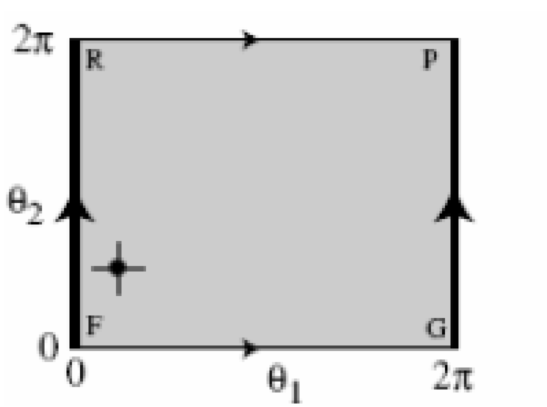
Charts and Differentiable Manifolds

- A Chart is a pair (U,φ) such that U is an open set in a k-dimensional manifold and φ is a diffeomorphism from U to some open set in $\Re^k$
  - think of this as a "coordinate system" for U (e.g. lines of latitude and longitude away form the poles).
  - The inverse map is a parameterization of the manifold
- Many manifolds require more than one chart to cover (e.g. the circle requires at least 2)
- An *atlas* is a set of charts that
  - cover a manifold
  - are smooth where they overlap (the book defines the notion of C^∞ related for this; we will take this for granted).
- A set S is a *differentiable manifold of dimension n* if there exists an atlas from S to $\Re^n$
  - For example, this is what allows us (locally) to view the (spherical) earth as flat and talk about translational velocities upon it.

Parameterization of the Torus







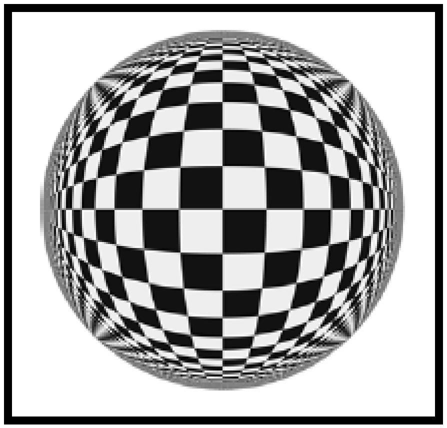
$(\theta_1, \theta_2) \in \mathbb{R}^2$,

problems at $\theta_i = \{0, 2\pi\}$.

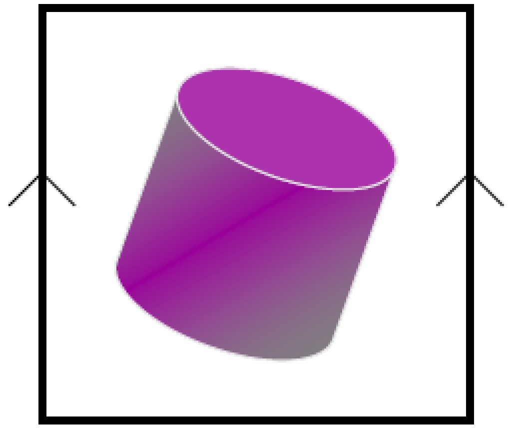
A Few Final Definitions

- A manifold is *path-connected* if there is a path between any two points.
- A space is *compact* if it is closed and bounded
  - configuration space might be either depending on how we model things
  - compact and non-compact spaces cannot be diffeomorphic!
- With this, we see that for manifolds, we can
  - live with “global” parameterizations that introduce odd singularities (e.g. angle/elevation on a sphere)
  - use atlases
  - embed in a higher-dimensional space using constraints
- Some prefer the latter as it often avoids the complexities associated with singularities and/or multiple overlapping maps

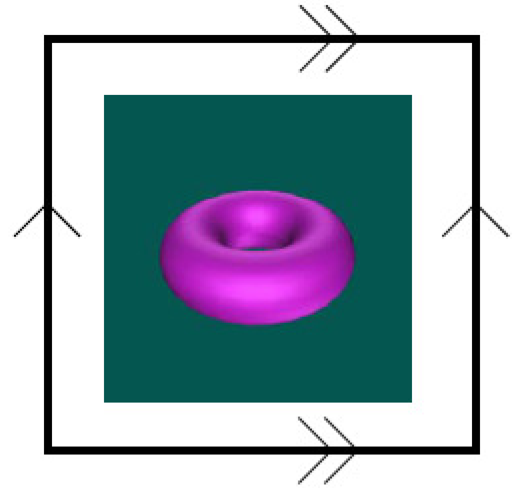
2D Manifolds



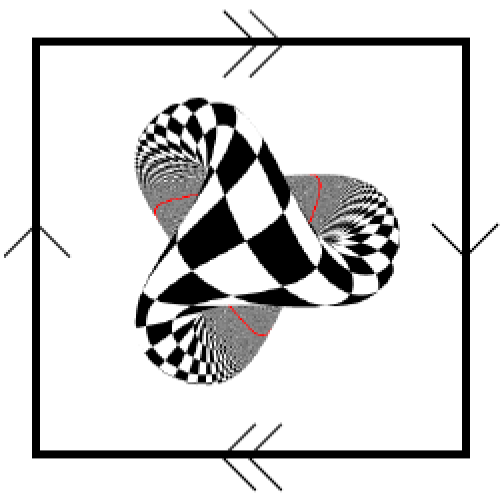
real plane



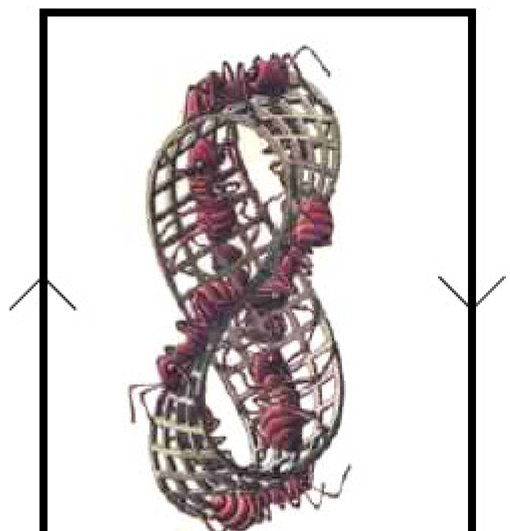
cylinder



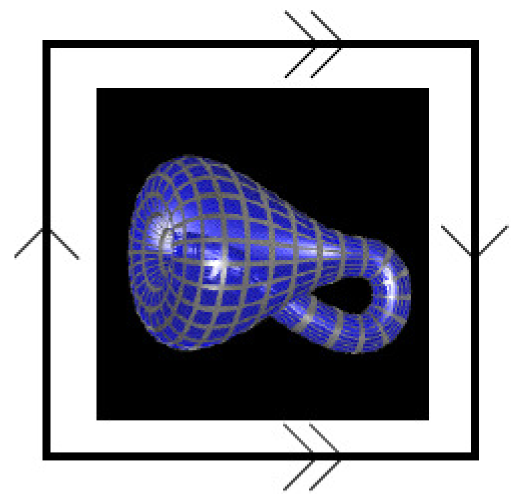
torus



projective plane



mobius strip



klein bottle

Minor Notational Points

- $\mathfrak{R}^1 \times \mathfrak{R}^1 \times \ldots \times \mathfrak{R}^1 = \mathfrak{R}^n$
- $S^1 \times S^1 \times \ldots \times S^1 \neq S^n$ (= T^n, the n-dimensional torus)
- S^n is the n-dimensional sphere
- Although S^n is an n-dimensional manifold, it is not a manifold of a single chart --- there is no single, smooth, invertible mapping from S^n to R^n
  - they are not ??morphic?

Representing Rotations

- Consider S^1 --- rotation in the plane
- The action of a rotation is to, well, rotate --> $R_\theta: \Re^2 \rightarrow \Re^2$
- We can represent this action by a matrix R that is applied (through matrix multiplication) to points in $\Re^2$

$$\begin{matrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{matrix}$$

- Note, we can either think of rotating a point through an angle, or rotate the **coordinate system (or frame)** of the point.

Geometric Transforms

- Now, using the idea of homogeneous transforms,we can write:

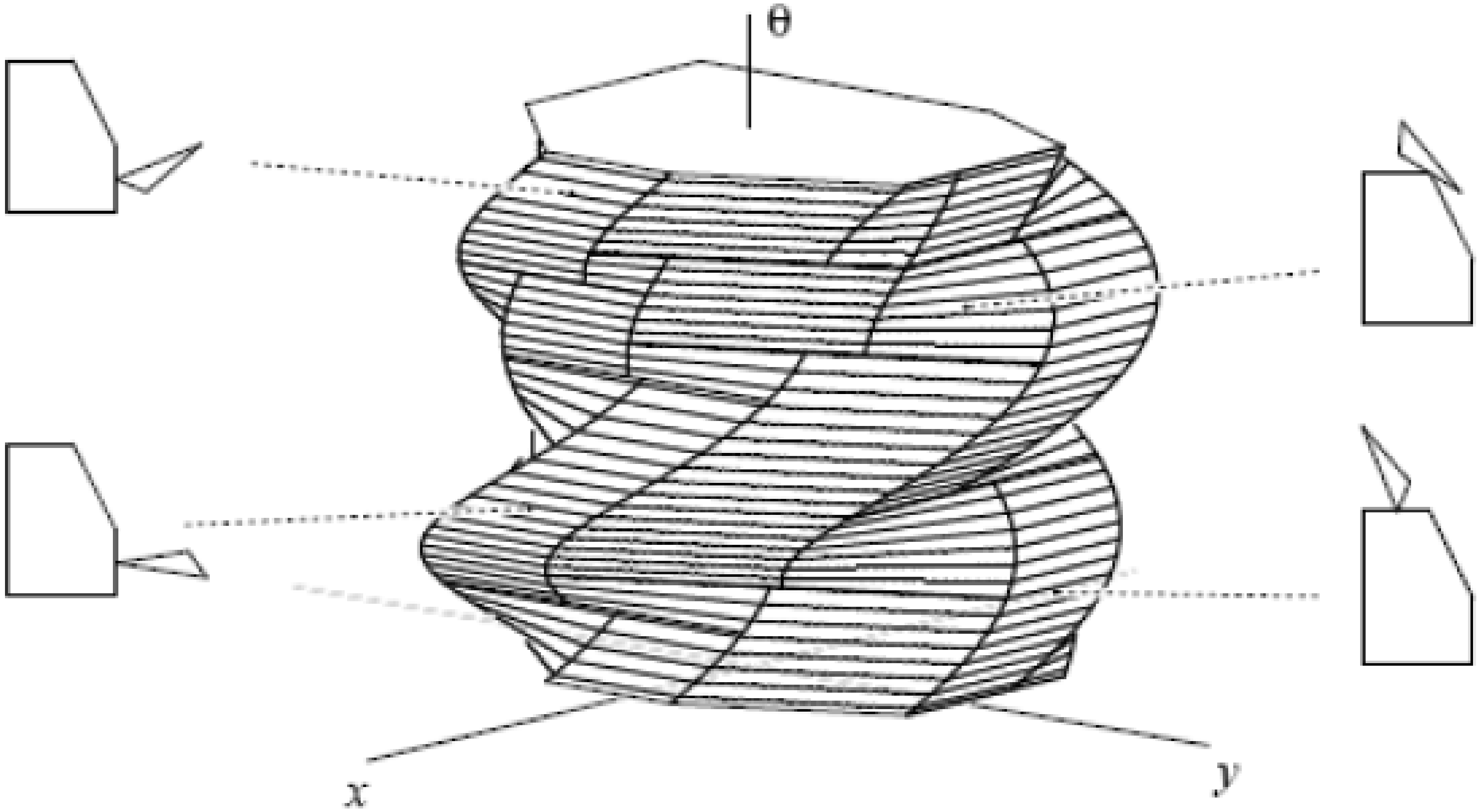
$$p' = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} p$$

- The group of rigid body rotations SO(2) × $\Re$(2) is denoted SE(2) (for special Euclidean group)

$$R = \begin{bmatrix} \tilde{x_1} & \tilde{y_1} \\ \tilde{x_2} & \tilde{y_2} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \in SO(2)$$

- This space is a type of torus

SE(2)

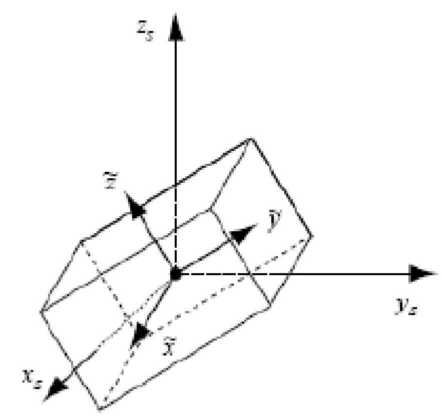


From 2D to 3D Rotation

- One can think of a 3D rotation as a rotation about different axes:
  - rot(x,θ) rot(y,θ) rot(z,θ)
  - there are many conventions for these (see Appendix E)
    - Euler angles (ZYZ) --- where is the singularity (see eqn 3.8)
    - Roll Pitch Yaw (ZYX)
    - Angle axis coordinates
    - Quaternions

- The space of rotation matrices has its own special name: SO(n) (for special orthogonal group of dimension n). It is a manifold of dimension n.

$$R = \begin{bmatrix} \tilde{x}_1 & \tilde{y}_1 & \tilde{z}_1 \\ \tilde{x}_2 & \tilde{y}_2 & \tilde{z}_2 \\ \tilde{x}_3 & \tilde{y}_3 & \tilde{z}_3 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \in SO(3)$$



- What is the derivative of a rotation matrix?
  - A tricky question --- what is the topology of that space ;-)

Geometric Transforms

- Now, using the idea of homogeneous transforms,we can write:

$$p' = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} p$$

- The group of rigid body rotations SO(3) × $\Re$(3) is denoted SE(3) (for special Euclidean group)

$$SE(n) \equiv \begin{bmatrix} SO(n) & \mathbb{R}^n \\ 0 & 1 \end{bmatrix}$$

- What does the inverse transformation look like?

Examples

Type of robot	Representation of Q
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 the three-space	$\mathbb{R}^3$
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	$SE(2) \times T^n$

$S^1 \times S^1 \times \ldots \times S^1$ (n times) $= T^n$, the n-dimensional torus

$S^1 \times S^1 \times \ldots \times S^1$ (n times) $\neq S^n$, the n-dimensional sphere in $\mathbb{R}^{n+1}$

$S^1 \times S^1 \times S^1 \neq SO(3)$

$SE(2) \neq \mathbb{R}^3$

$SE(3) \neq \mathbb{R}^6$

More Example Configuration Spaces
(contrasted with workspace)

- Holonomic robot in plane:
  - workspace $\Re^2$
  - configuration space $\Re^2$
- 3-joint revolute arm in the plane
  - Workspace, a torus of outer radius L1 + L2 + L3
  - configuration space T^3
- 2-joint revolute arm with a prismatic joint in the plane
  - workspace disc of radius L1 + L2 + L3
  - configuration space $T^2 \times \Re$
- 3-joint revolute arm mounted on a mobile robot (holonomic)
  - workspace is a “sandwich” of radius L1 + L2 + L3
  - $\Re^2 \times T^3$
- 3-joint revolute arm floating in space
  - workspace is $\Re^3$
  - configuration space is T^3

Dimension of the Configuration Space

- The dimension is the number of parameter necessary to uniquely specify configuration
- One way to do this is to explicitly generate a parameterization (e.g with our 2-bar linkage)

- Another is to start with too many parameters and add (independent) constraints
  – suppose I start with 4 points in the plane (= 8 parameters), A, B, C, D
  – Rigidity requires d(A,B) = c_1 (1 constraints)
  – Rigidity requires d(A,C) = c_2 and d(B,C) = c_3 (2 constraints)
  – Rigidity requires d(A,D) = c_4 and d(B,D) = c_5 and ??? (?? constraints)
  – HOW MANY D.O.F?

- The question is:
  – How many DOF do you need to move freely in 3-space?

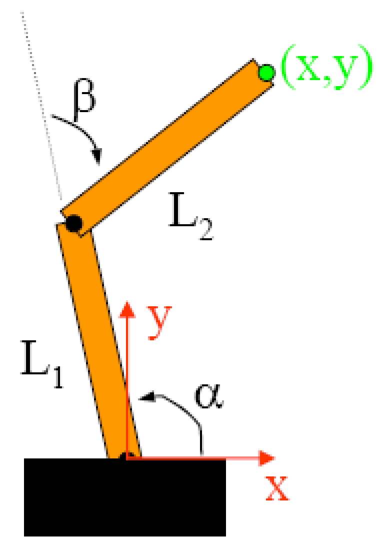
More on Dimension

- $\Re^1$ and SO(2) are
  - one dimensional manifolds
- $\Re^2$, S^2 and T^2 are
  - two dimensional manifolds
- $\Re^3$, SE(2) and SO(3) are
  - three dimensional manifolds
- $\Re^6$, T^6 and SE(3) are
  - six dimensional manifolds

Transforming Velocity

- Recall forward kinematics $K: Q \rightarrow \mathcal{W}$
- The *Jacobian* of K is the n × m matrix with entries
  - $J_{i,j} = d\,K_i / d\,q_j$
- The Jacobian transforms velocities:
  - $dw/dt = J\,dq/dt$
- If square and invertible, then
  - $dq/dt = J^{-1}\,dw/dt$
- Example: our favorite two-link arm...

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} L_1 c_\alpha \\ L_1 s_\alpha \end{pmatrix} + \begin{pmatrix} L_2 c_+ \\ L_2 s_+ \end{pmatrix}$$



Useful Observations

- The Jacobian maps configuration velocities to workspace velocities

- Suppose we wish to move from a point A to a point B in the workspace along a path p(t) (a mapping from some time index to a location in the workspace)
  - dp/dt gives us a velocity profile --- how do we get the configuration profile?
  - Are the paths the same if choose the shortest paths in workspace and configuration space?

Holonomic vs. Non-Holonomic Systems

- The previous constraints were *holonomic* -- they constrained theconfiguration of the system.
  – g(q, t) = 0 (note they can be time-varying!)

- *Non-holonomic* constraints are of the form
  – g(q,dq/dt,t) = 0 (position *and* velocity)

- Example: A mobile robot with location x and orientation θ has motion
  – dx/dt = (v(t) cos(θ), v(t) sin(θ))
  – Note that the kinematics of this system involves integration!

Nonholonomicity

- Scout robot (and many other mobile robots) share a common (if frustrating) property: they have **nonholonomic** constraints.
  - makes it more difficult to navigate between two arbitrary points
  - need to resort to techniques like parallel parking

Another (informal) definition, a robot is **nonholonomic** if it *can not* move to change its pose instantaneously in all available directions within its workspace (although the complete set of motions spans the workspace

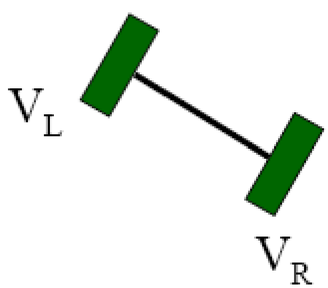
E.g. A car moves in x,y, theta, but can only go forward and backward along a curve

Nonholonomicity

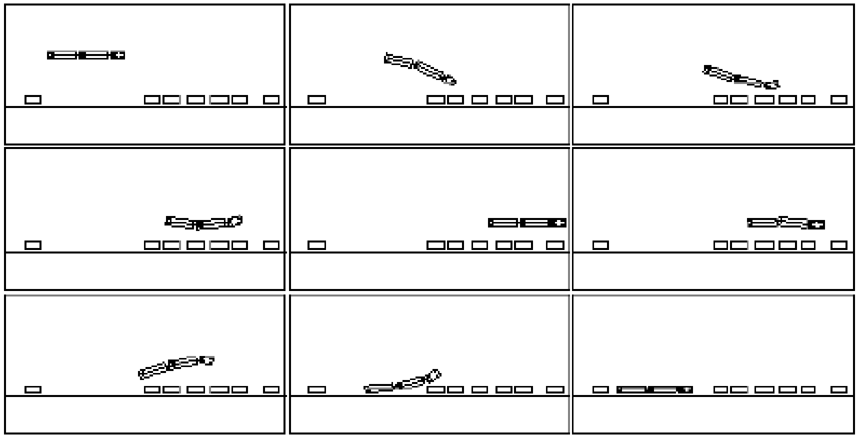
- Scout robot (and many other mobile robots) share a common (if frustrating) property: they have **nonholonomic** constraints.
  - makes it more difficult to navigate between two arbitrary points
  - need to resort to techniques like parallel parking

By definition, a robot is **nonholonomic** if it *can not* move to change its pose instantaneously in all available directions

differential-drive robots are nonholonomic



multiple-trailer rigs are "very" nonholonomic



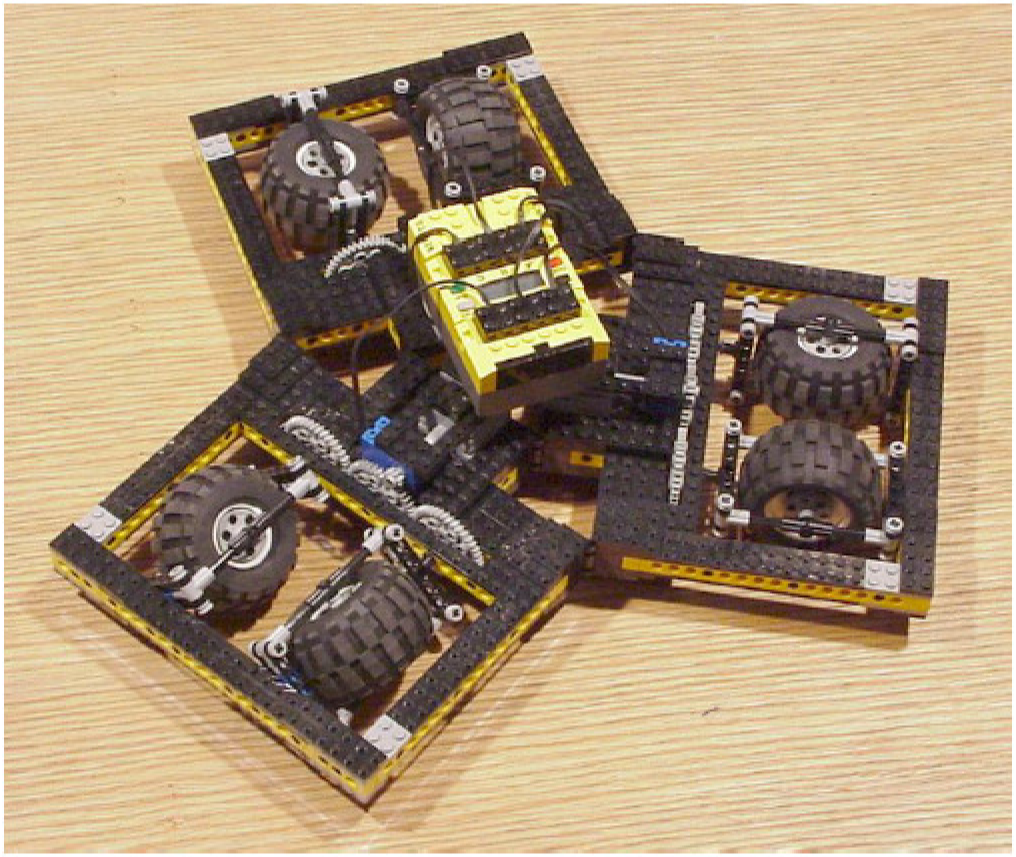
Holonomic Robots

Navigation is simplified considerably if a robot *can* move instantaneously in any direction, i.e., is **holonomic**.

Omniwheels

Mecanum wheels

Holonomic Designs



Killough Platform

Summary

- Configuration spaces, workspaces, and some basic ideas about topology
- Types of robots: holonomic/nonholonomic, serial, parallel
- Kinematics and inverse kinematics
- Coordinate frames and coordinate transformations
- Jacobians and velocity relationships

T. Lozano-Pérez.
Spatial planning: A configuration space approach.
IEEE Transactions on Computing, C-32(2):108-120, 1983.