Configuration Space I

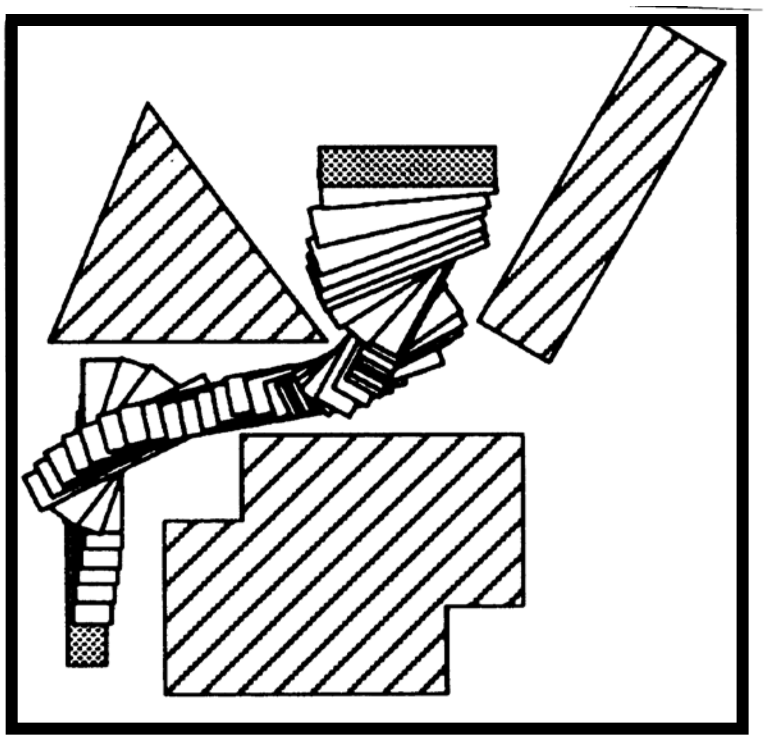
Sung-Eui Yoon
(윤성의)

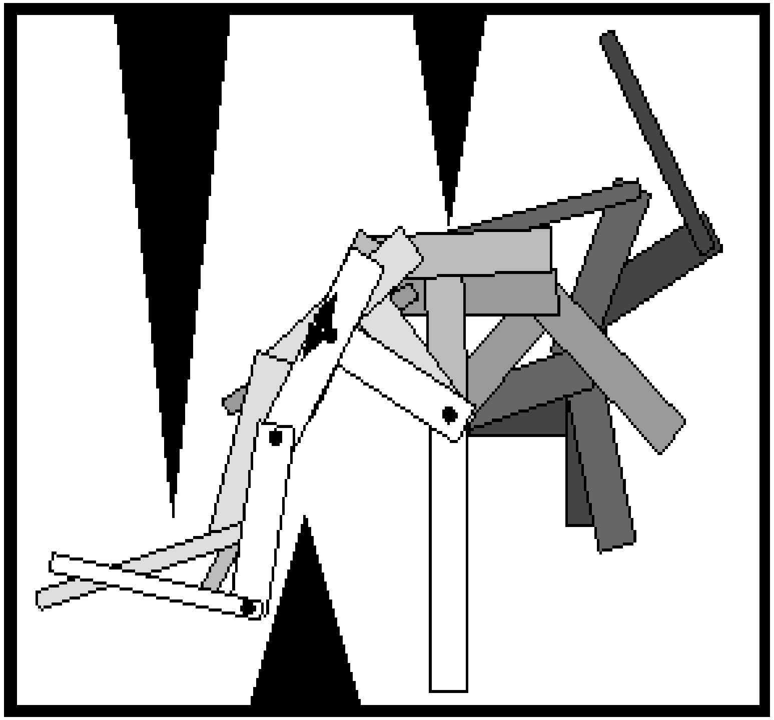
Course URL:
http://sglab.kaist.ac.kr/~sungeui/MPA

Class Objectives

- **Configuration space**
  - **Definitions and examples**
  - **Obstacles**
  - **Paths**
  - **Metrics**

What is a Path?

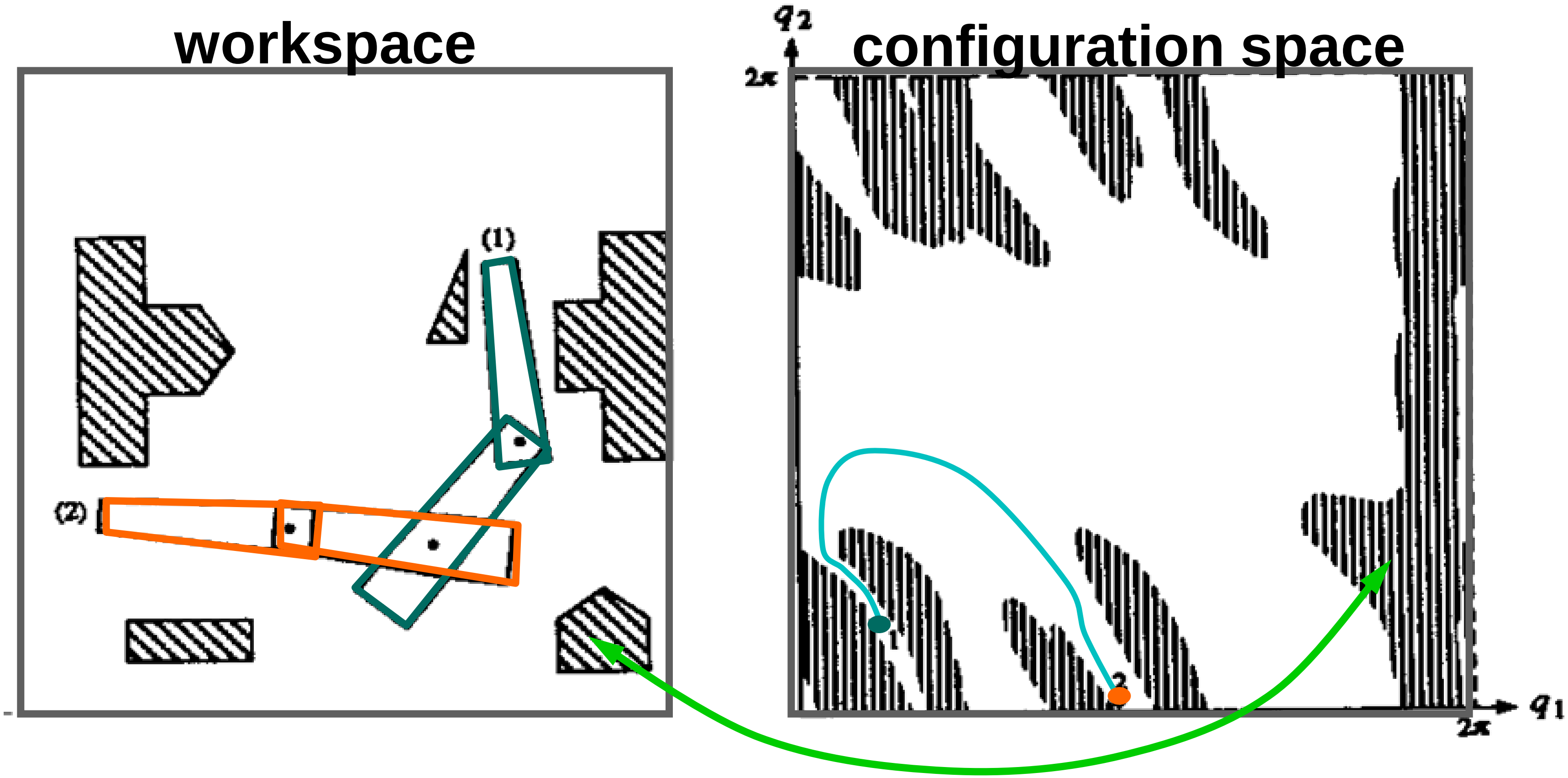




Rough Idea

- **Convert rigid robots, articulated robots, *etc.* into points**
- **Apply algorithms for moving points**

Mapping from the Workspace to the Configuration Space

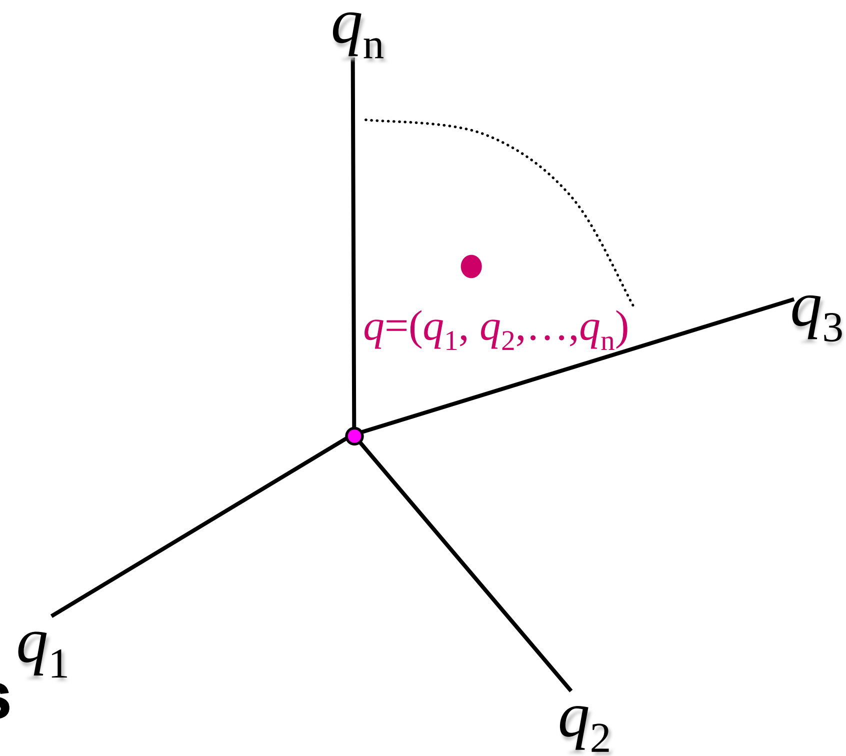


Configuration Space

- Definitions and examples
- Obstacles
- Paths
- Metrics

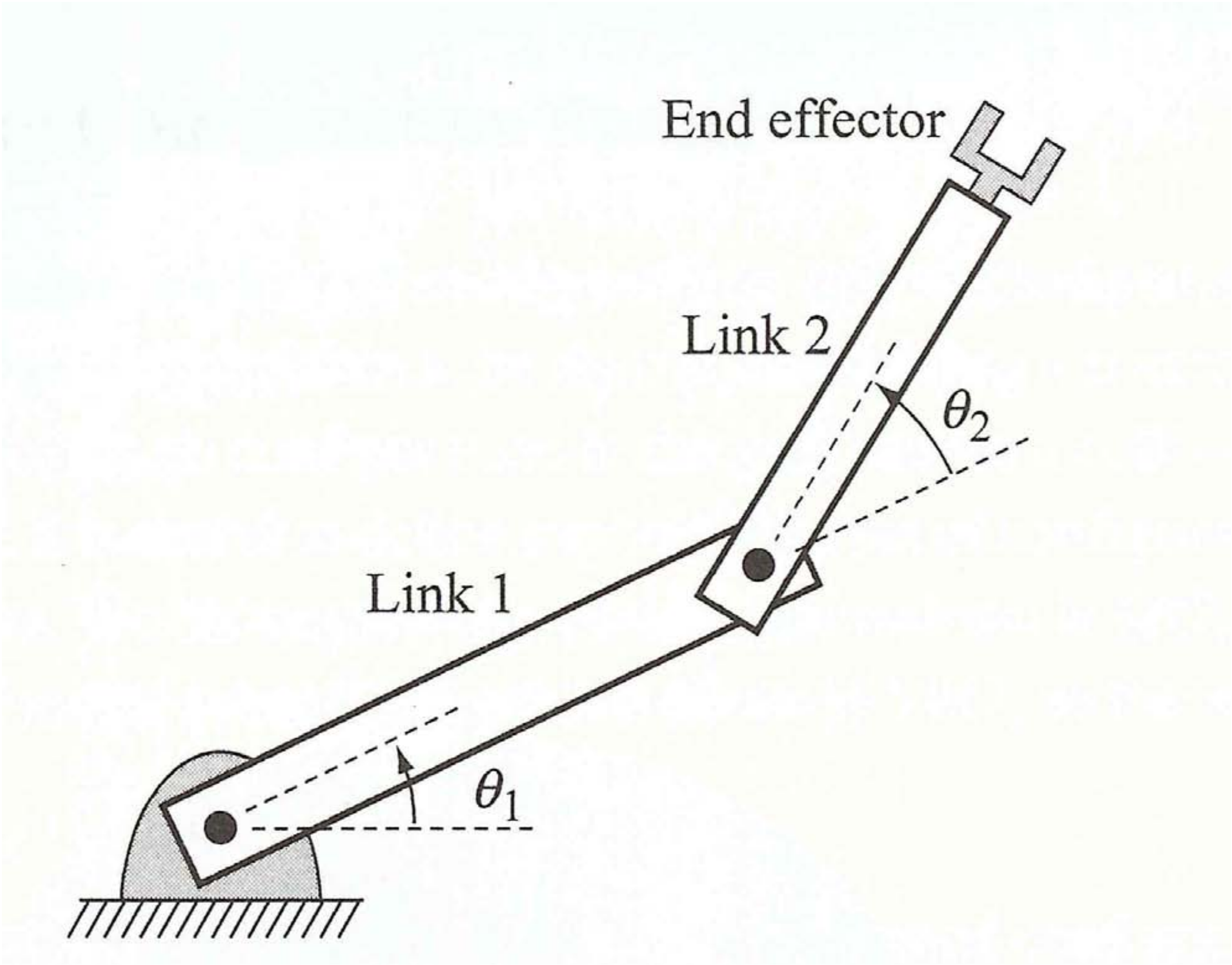
Configuration Space (C-space)

- The **configuration** of an object is a complete specification of the position of **every** point on the object
  - Usually a configuration is expressed as a vector of position & orientation parameters: $q = (q_1, q_2,\ldots,q_n)$
- The **configuration space** C is the set of all possible configurations
  - A configuration is a point in C

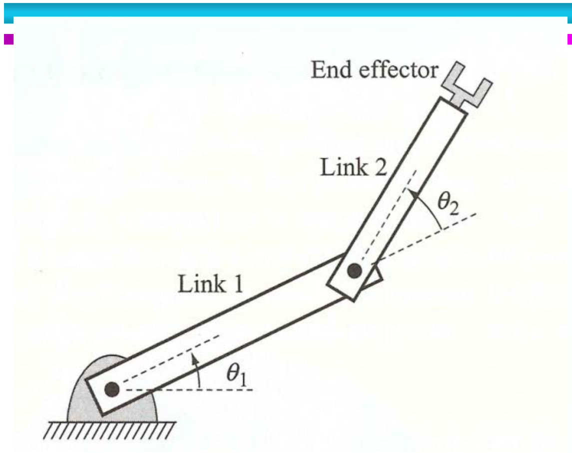


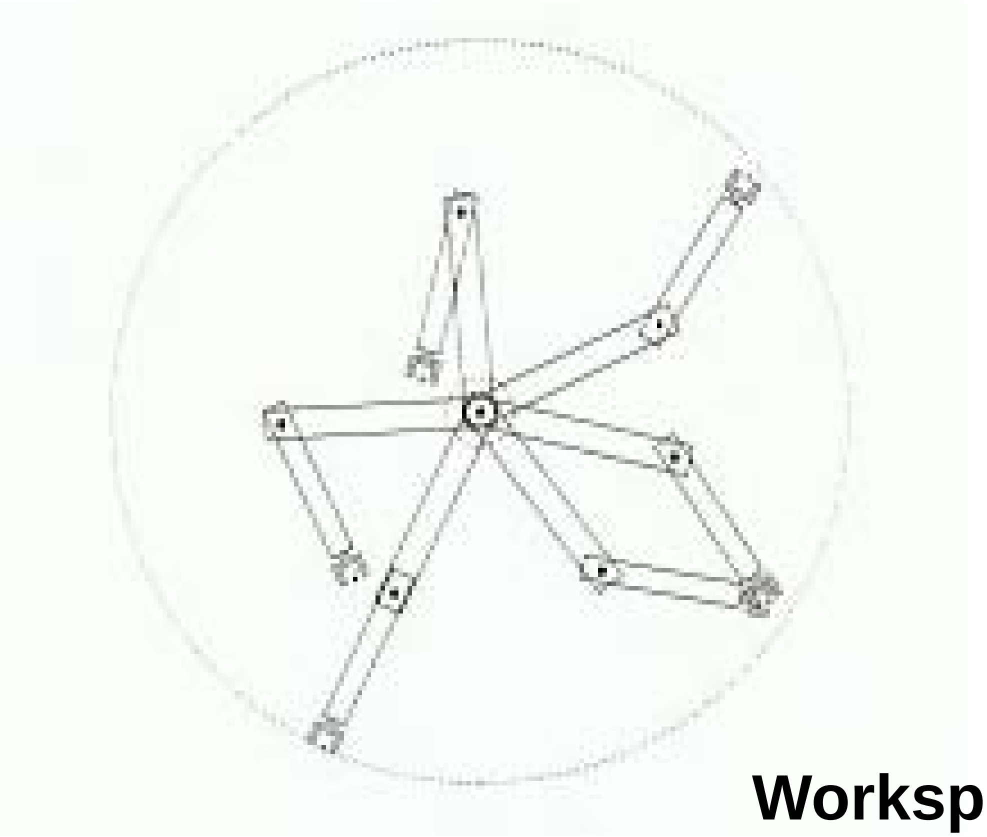
KAIST

Examples of Configuration Spaces



Examples of Configuration Spaces

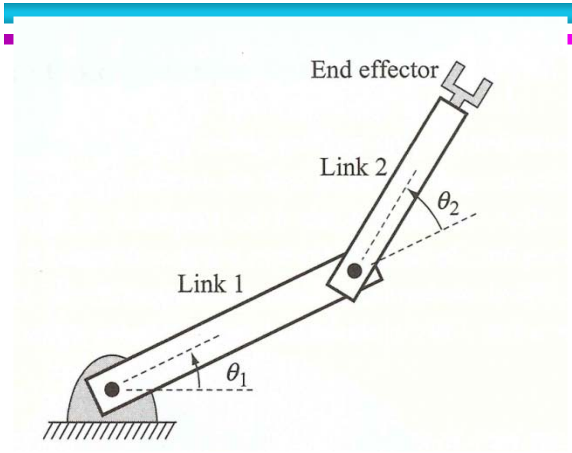




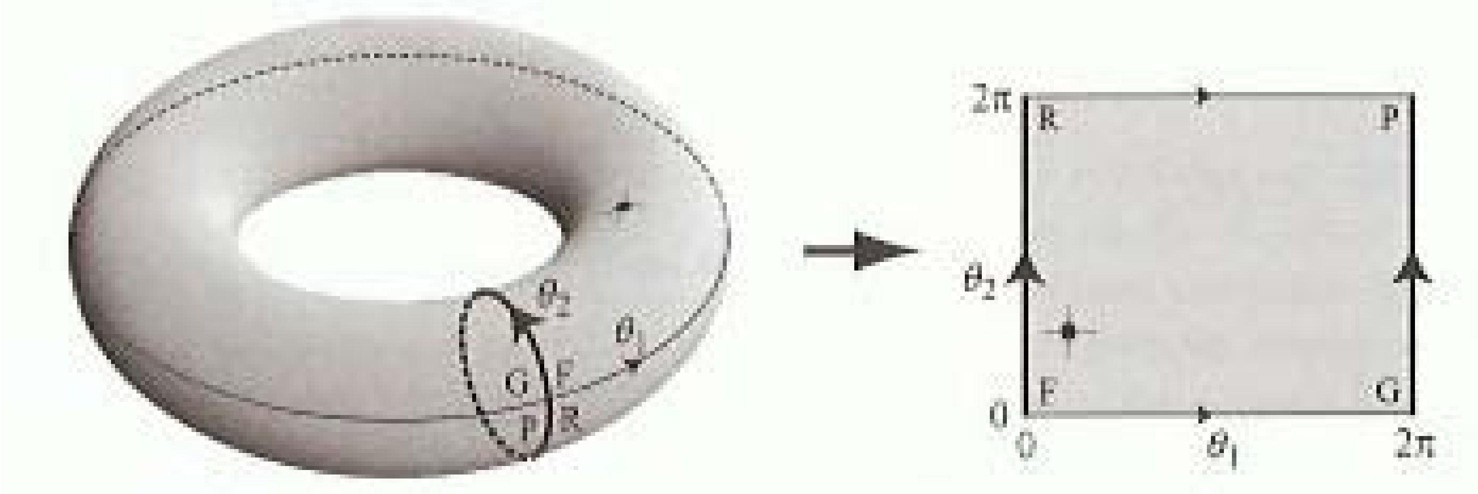
Workspace

This is not a valid C-space!

Examples of Configuration Spaces

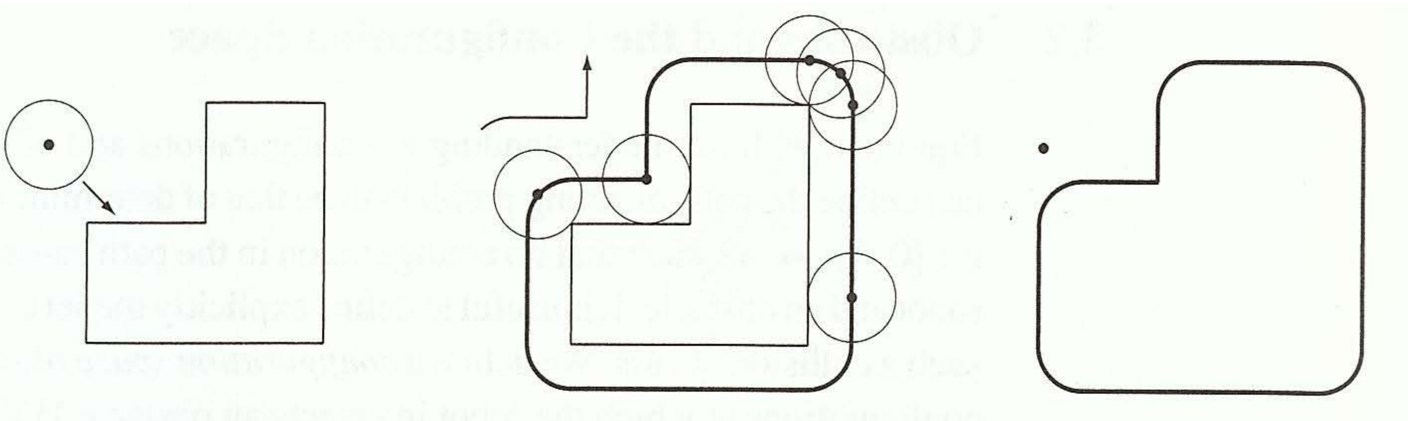


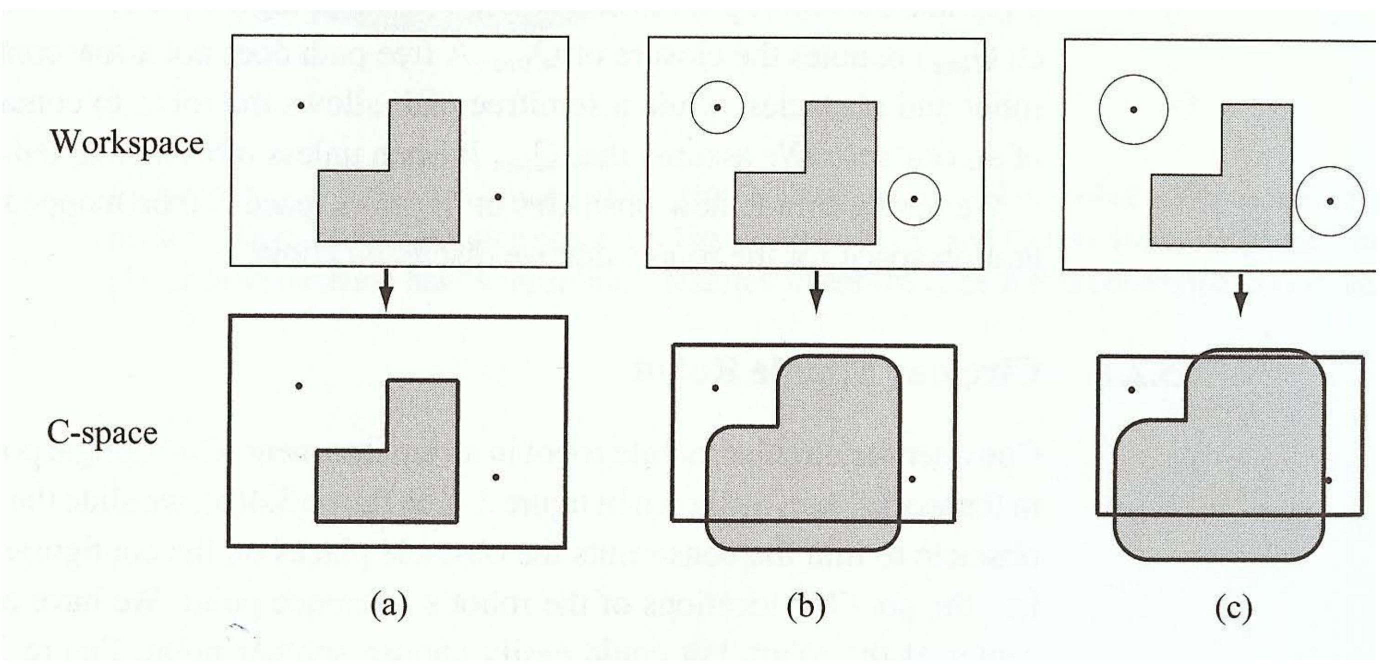
The topology of C is usually **not** that of a Cartesian space R^n.



$S^1 \times S^1 = T^2$

Examples of Circular Robot





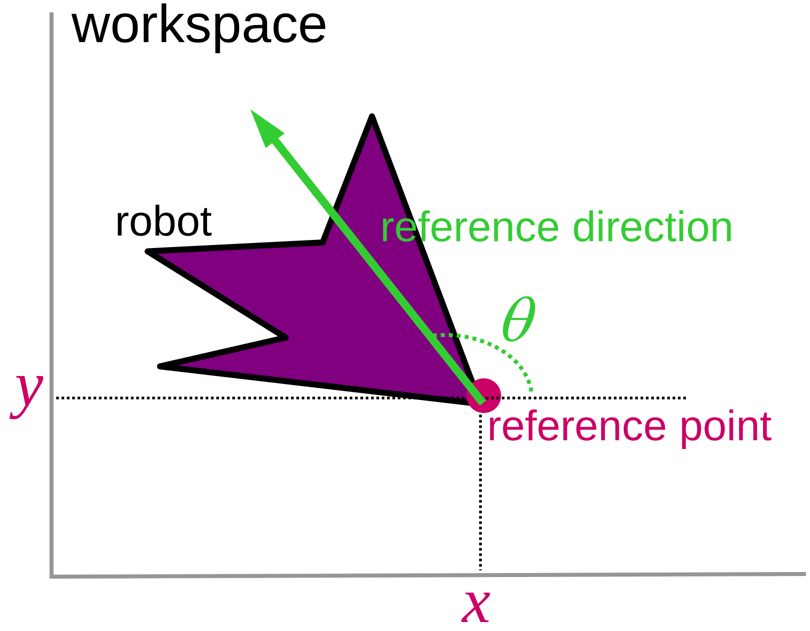
Dimension of Configuration Space

- The dimension of the configuration space is the minimum number of parameters needed to specify the configuration of the object completely

- It is also called the number of degrees of freedom (dofs) of a moving object

KAIST

Example: Rigid Robot in 2-D Workspace



- **3-parameter specification:** $q = (x, y, \theta)$ **with** $\theta \in [0, 2\pi)$.
  - **3-D configuration space**

Example: Rigid Robot in 2-D workspace

- **4-parameter specification:** $q = (x, y, u, v)$ **with** $u^2+v^2 = 1$. **Note** $u = \cos\theta$ **and** $v = \sin\theta$
- **dim of configuration space = 3**
  - **Does the dimension of the configuration space (number of dofs) depend on the parametrization?**

Holonomic and Non-Holonomic Contraints

- **Holonomic constraints**
  - $g(q, t) = 0$

- **Non-holonomic constraints**
  - $g(q, q', t) = 0$

Computation of Dimension of C-Space

- Suppose that we have a rigid body that can translate and rotate in 2D workspace
  - Start with three points: A, B, C (6D space)

- We have the following (holonomic) constraints
  - Given A, we know the dist to B: d(A,B) = |A-B|
  - Given A and B, we have similar equations: d(A,C) = |A-C|, d(B,C) = |B-C|

- Each holonomic constraint reduces one dim.
  - Not for non-holonomic constraint

KAIST

Example: Rigid Robot in 3-D Workspace

- **We can represent the positions and orientations of such robots with matrices (i.e., SO (3) and SE (3))**

SO (n) and SE (n)

- **Special orthogonal group, SO(n), of n x n matrices R,**

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

that satisfy:

$$r_{1i}^2 + r_{2i}^2 + r_{3i}^2 = 1 \text{ for all } i,$$
$$r_{1i} r_{1j} + r_{2i} r_{2j} + r_{3i} r_{3j} = 0 \text{ for all } i \neq j,$$
$$\det(R) = +1$$

Refer to the 3D Transformation at the undergraduate computer graphics.

- **Given the orientation matrix R of SO (n) and the position vector p, special Euclidean group, SE (n), is defined as:**

$$\begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}$$

Example: Rigid Robot in 3-D Workspace

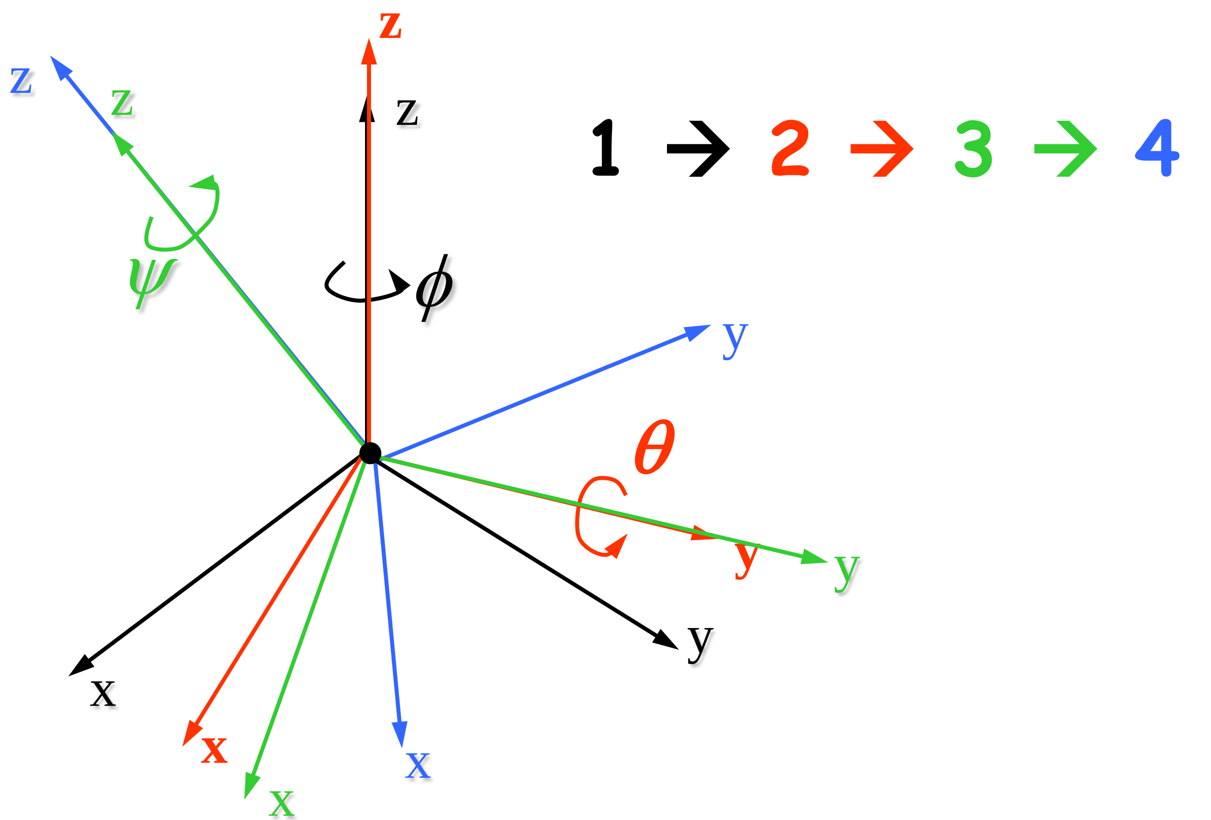
- q = (position, orientation) = $(x, y, z, ???)$

- **Parametrization of orientations by matrix:**
  $q = (r_{11}, r_{12}, \ldots, r_{33}, r_{33})$ **where** $r_{11}, r_{12}, \ldots, r_{33}$ **are the elements of rotation matrix**

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \in SO(3)$$

Example: Rigid Robot in 3-D Workspace

- **Parametrization of orientations by Euler angles:** (ϕ,θ,ψ)

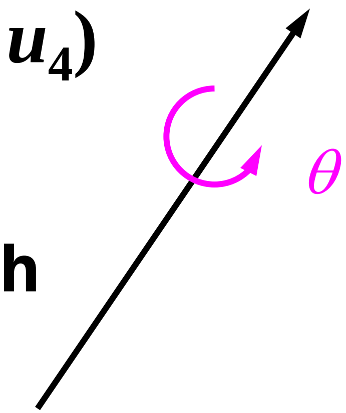


KAIST

Example: Rigid Robot in 3-D Workspace

- **Parametrization of orientations by unit quaternion:** $u = (u_1, u_2, u_3, u_4)$ **with** $u_1^2 + u_2^2 + u_3^2 + u_4^2 = 1.$
  - **Note** $(u_1, u_2, u_3, u_4) = (\cos\theta/2,\ n_x\sin\theta/2,\ n_y\sin\theta/2,\ n_z\sin\theta/2)$ **with** $n_x^2 + n_y^2 + n_z^2 = 1$

$\mathbf{n} = (n_x, n_y, n_z)$



  - **Compare with representation of orientation in 2-D:** $(u_1, u_2) = (\cos\theta, \sin\theta)$

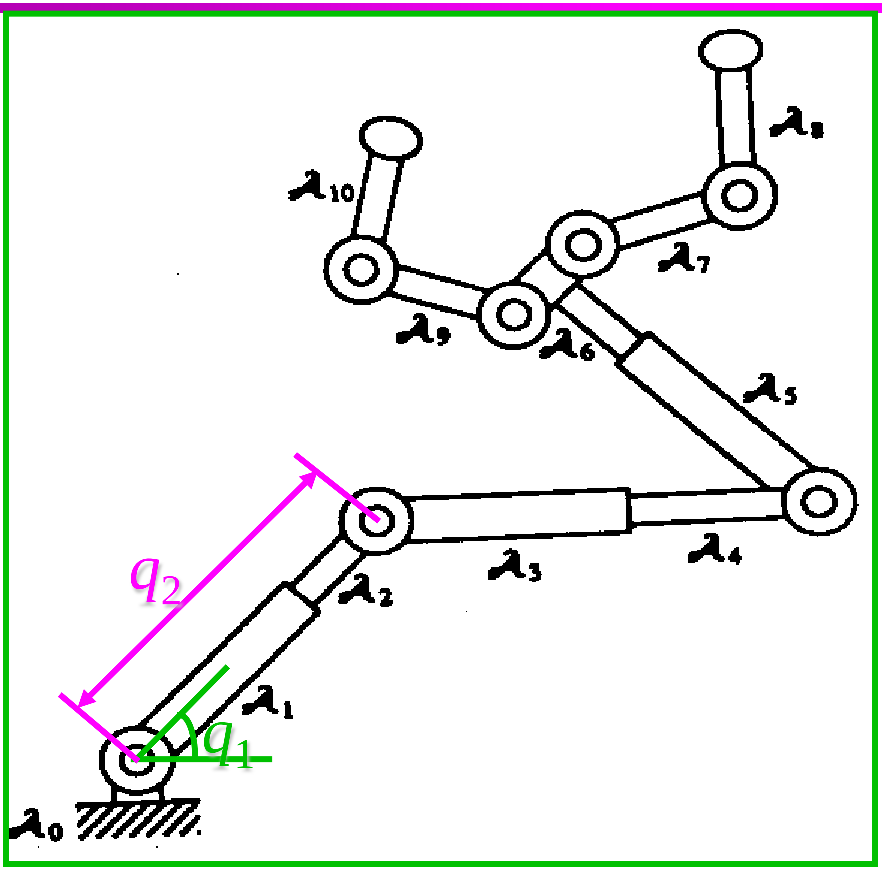
Example: Rigid Robot in 3-D Workspace

- **Advantage of unit quaternion representation**
  - **Compact**
  - **No singularity**
  - **Naturally reflect the topology of the space of orientations**

- **Number of dofs = 6**
- **Topology:** R^3 **x** SO(3)

KAIST

Example: Articulated Robot



An articulated object is a set of rigid bodies connected at the joints.

- $q = (q_1, q_2, \ldots, q_{2n})$
- **Number of dofs** = $2n$
- **What is the topology?**

Class Objectives were:

- Configuration space
  - Definitions and examples
  - Obstacles
  - Paths
  - Metrics

Additional Homework

- **For the first class in every week:**
  - **Find two papers at ICRA/IROS**
  - **Go over abstracts and browse papers**
  - **Submit a short summary (just a few paragraphs) of these two papers**

KAIST

Next Time….

- Configuration space
  - Definitions and examples
  - Obstacles
  - Paths
  - Metrics