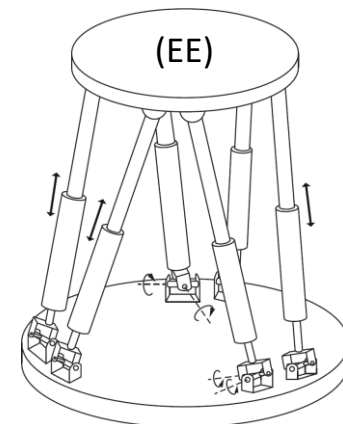


Ch3: Configuration Space

Introduction

Robot Mechanical Structure

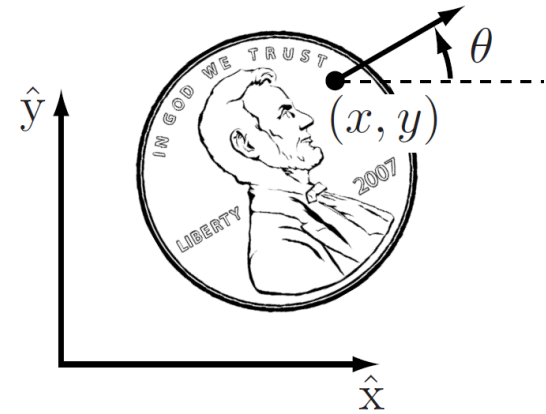
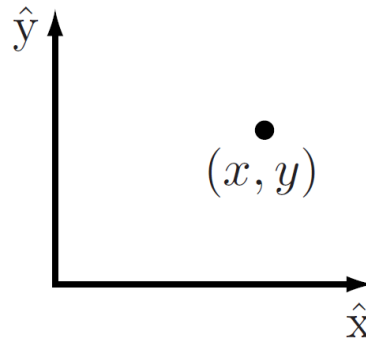
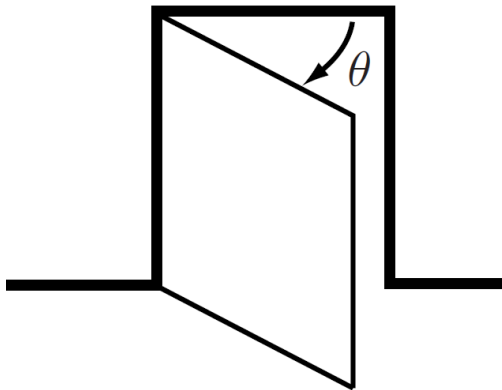
- A robot is mechanically constructed by connecting a set of bodies, called **Links**, to each other using various types of **Joints**.
 - * All the robots considered in this course have links that can be modeled as **rigid bodies**.
- **Actuators**, such as electric motors, deliver forces and torques to the joints, thereby causing motion of the robot.
- An **End-Effector** (EE), such as a gripper or hand for grasping and manipulating objects, is attached to a specific link.



Configuration, DOF, and C-Space of a Robot

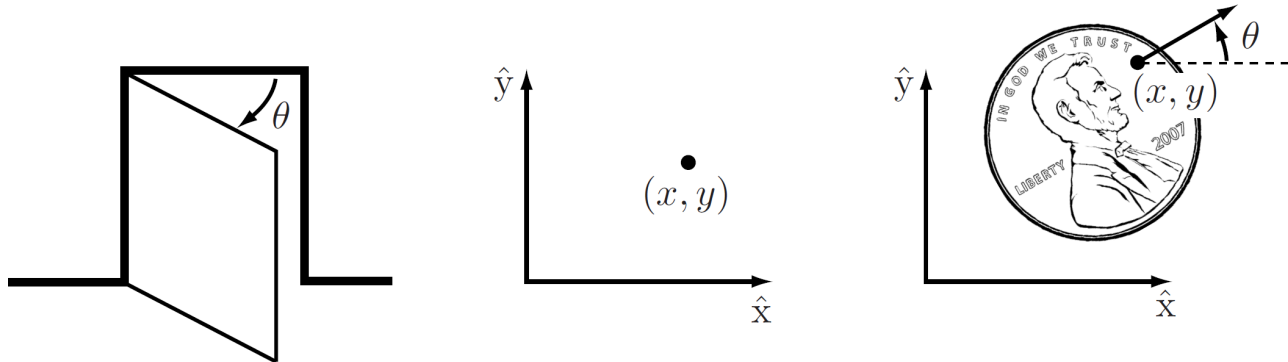
Configuration: A complete specification of the positions of all points of a robot/mechanism.

Since the robot's links are rigid and of a known shape/geometry, only a few numbers are needed to represent its configuration.

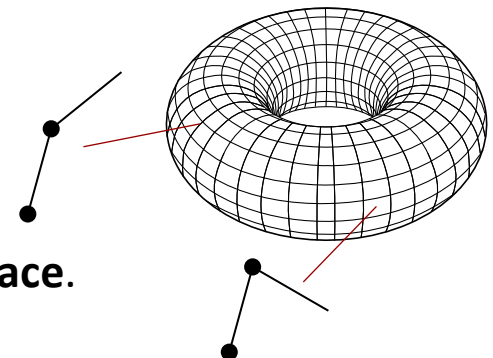


Configuration, DOF, and C-Space of a Robot

Degrees of Freedom (DOF): The minimum number n of real-valued coordinates needed to represent the **configuration** of a robot/mechanism.



Configuration Space (C-Space): The n -dimensional space containing all possible configurations of the robot/mechanism.



* The **configuration** of a robot is represented by a point in its **C-space**.

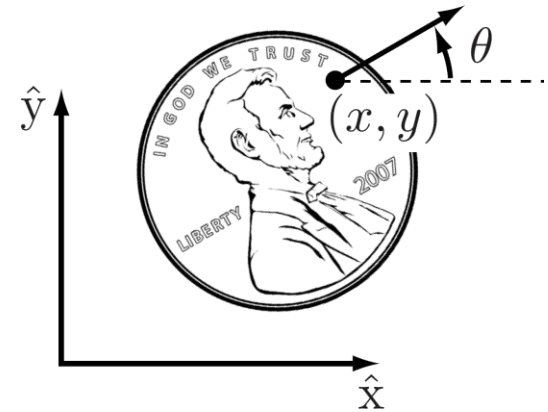
Degrees of Freedom (DOF)

DOFs of a Rigid Body in 2D Space

Example: Number of DOFs of a coin on a plane



3 DOFs

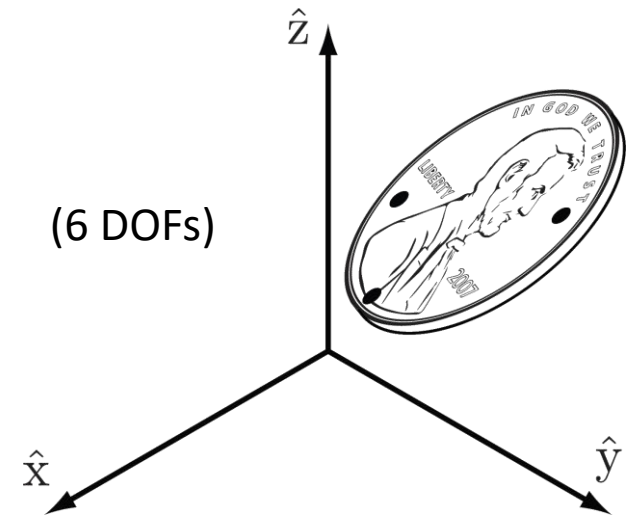


A general rule for determining the number of DOFs of rigid bodies :

$$\text{DOF} = (\text{number of variables}) - (\text{number of independent equations/constraints})$$

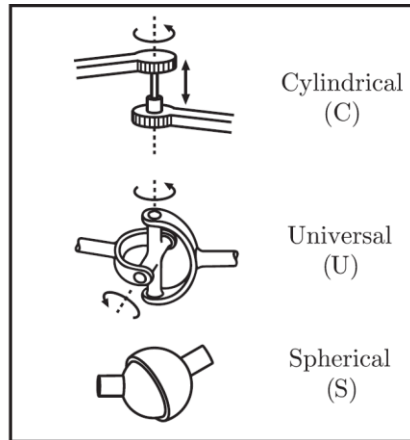
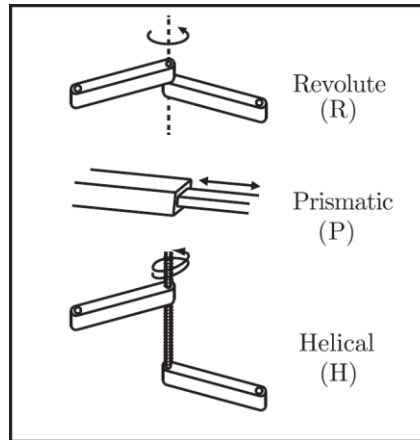
DOFs of a Rigid Body in 3D Space

Example: Number of DOFs of a coin in space



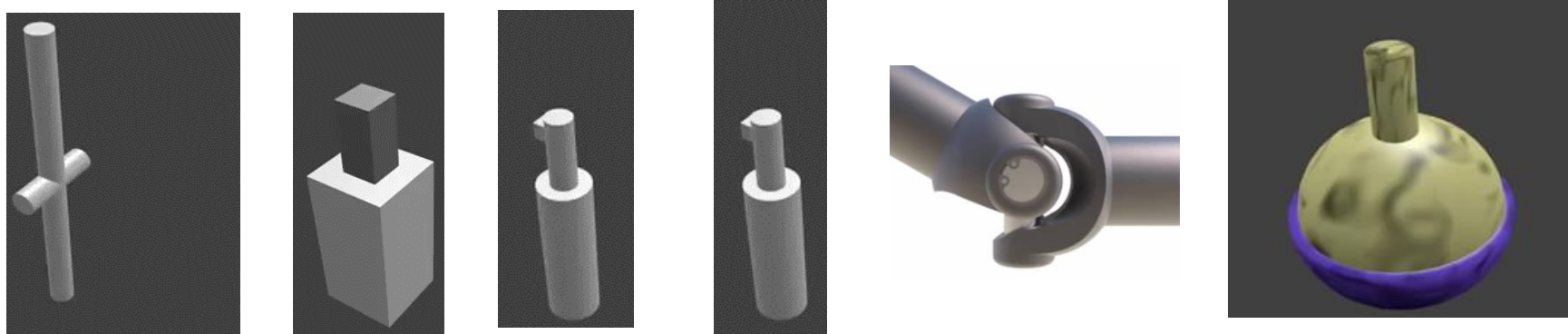
In summary, a **spatial rigid body**, has **six** degrees of freedom and a **planar rigid body** has **three** degrees of freedom.

DOFs of Robots: Typical Robot Joints



Joint type	dof f	Constraints c between two planar rigid bodies	Constraints c between two spatial rigid bodies
Revolute (R)	1	2	5
Prismatic (P)	1	2	5
Helical (H)	1	N/A	5
Cylindrical (C)	2	N/A	4
Universal (U)	2	N/A	4
Spherical (S)	3	N/A	3

Note: Every joint connects exactly two links.



$$\# \text{ of DOFs of a joint} = (\# \text{ of DOFs of a rigid body}) - (\# \text{ of constraints provided by a joint})$$

DOFs of Robots: Grübler's Formula

A general rule for determining the number of DOFs of mechanisms consist of rigid bodies:

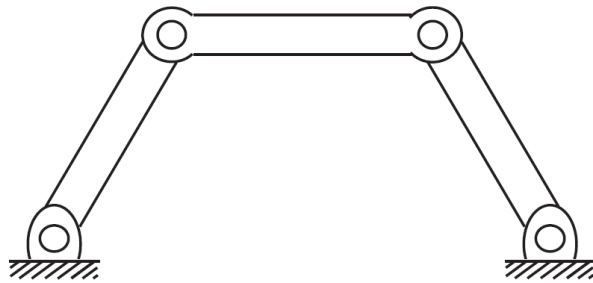
$$\text{DOF} = (\text{sum of freedoms of the bodies}) - (\text{number of } \underline{\text{independent constraints}})$$

Grübler's Formula for the number of degrees of freedom of mechanisms/robots:

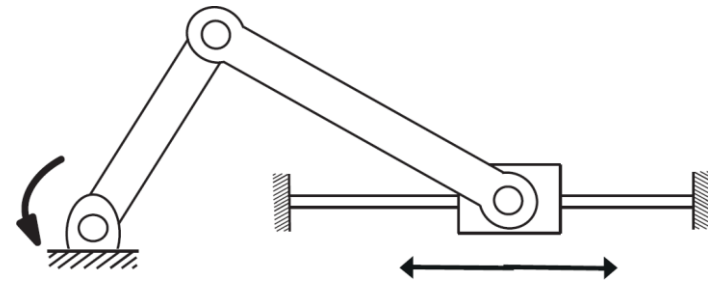
$$\text{dof} = m(N - 1 - J) + \sum_{i=1}^J f_i$$

Note: This formula fails when the joint constraints are not independent!

Examples: Number of DOFs

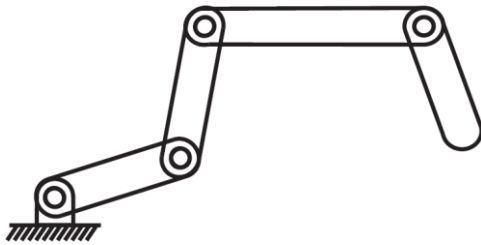


(Four-bar linkage)

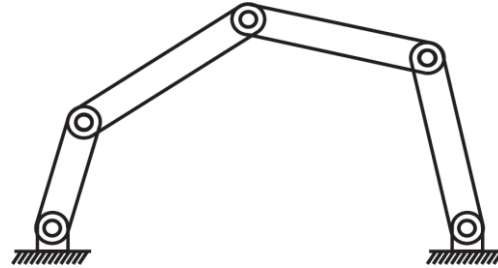


(Slider-crank mechanism)

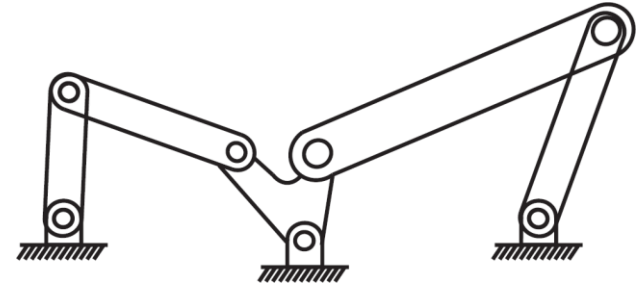
Examples: Number of DOFs



*k*R robot

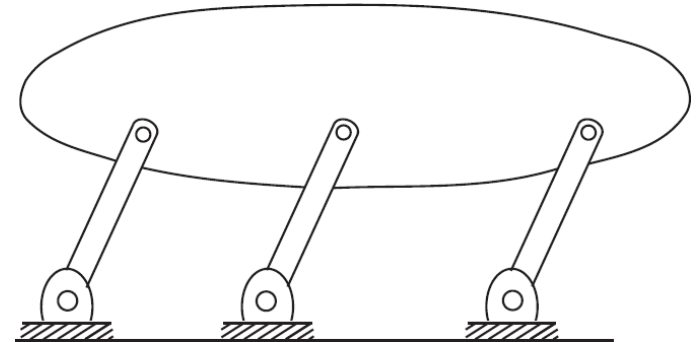
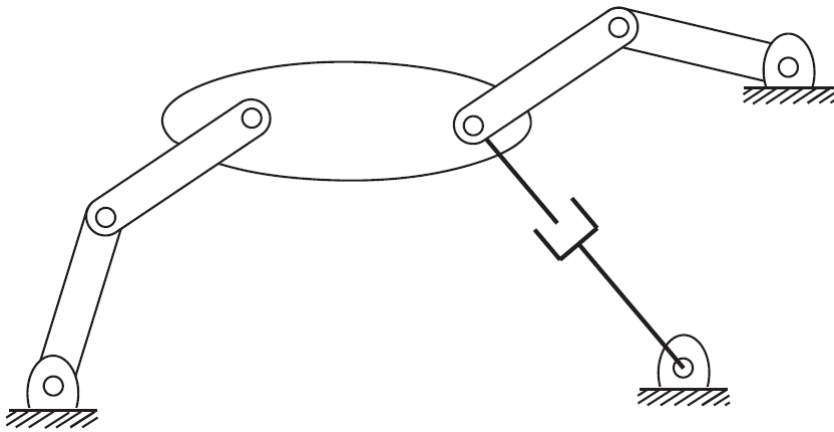


Five-bar linkage

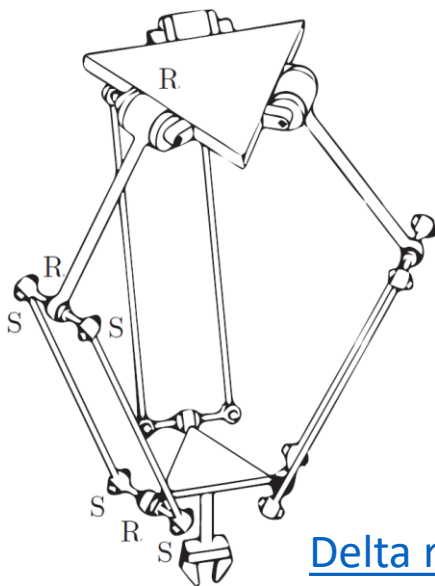


Watt six-bar linkage

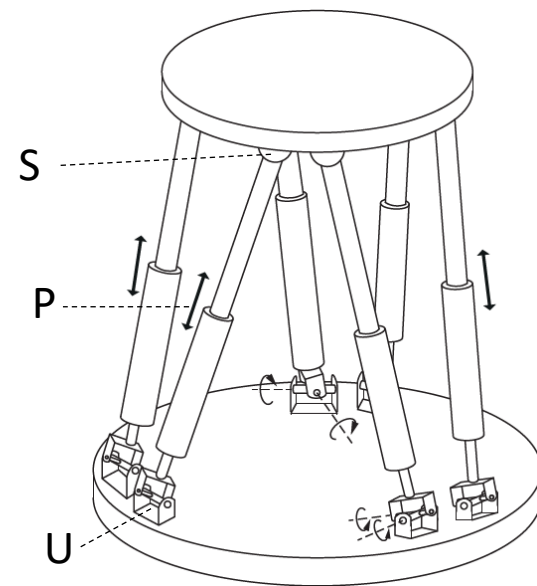
Examples: Number of DOFs



Examples: Number of DOFs



Delta robot



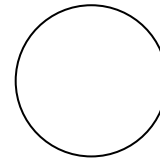
Stewart–Gough platform

Configuration Space Topology and Representation

Topologies of 1D C-Space

System

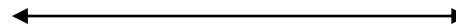
(a) A point moving on a Circle
(or any closed loop):



C-Space Topology

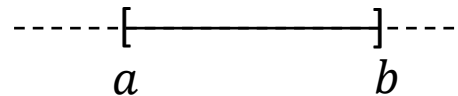
S^1

(b) A point moving on a Line:



\mathbb{E}^1 or \mathbb{R}^1

(c) A point moving on a Closed
Interval of Line:



$[a, b] \subset \mathbb{R}^1$

Two spaces are **topologically equivalent** if one can be continuously deformed into the other without cutting or gluing.



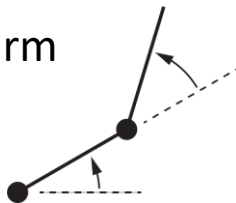
Topologies of 2D C-Space

System

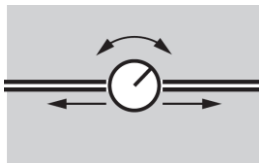
A point moving on a plane

Spherical pendulum

2R robot arm



Rotating sliding robot



C-Space Topology



$$\mathbb{R}^1 \times \mathbb{R}^1 = \mathbb{R}^2$$

(or $\mathbb{E}^1 \times \mathbb{E}^1 = \mathbb{E}^2$)



$$S^2$$



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



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

(or $\mathbb{E}^1 \times S^1$)

C-Space: More Examples

- A rigid body in the plane
- A PR robot arm
- A mobile robot with a 2R robot arm
- A rigid body in three dimensions

C-Space: More Examples

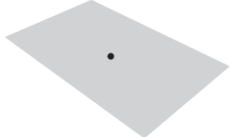

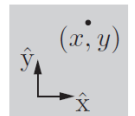
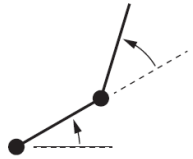

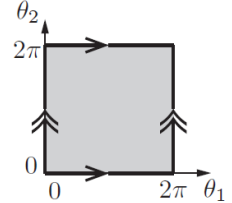
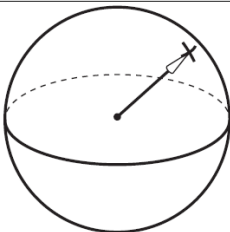

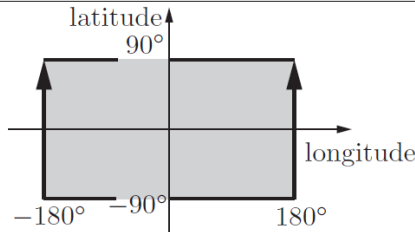
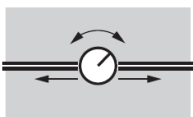

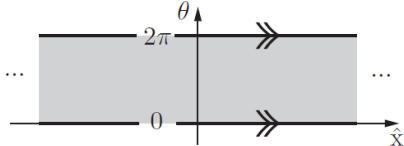
- Hexrotor UAV with two 5-DOF arms (without and with arm joint limits)



C-Space Representation

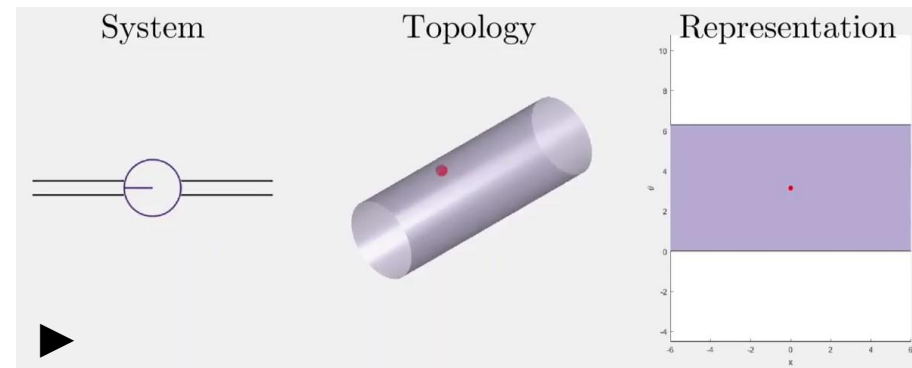
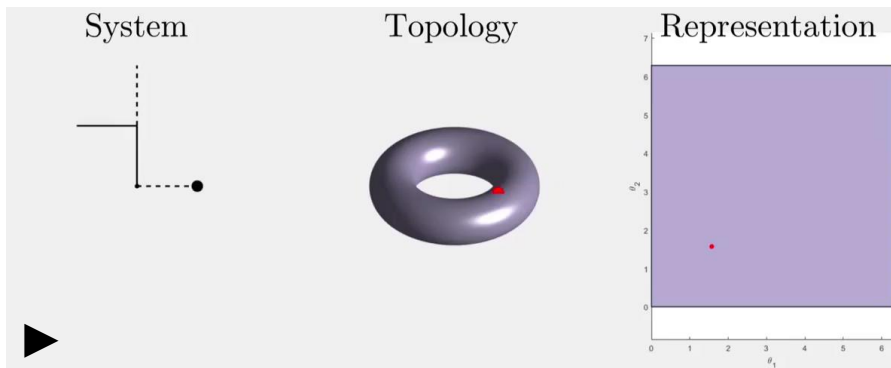
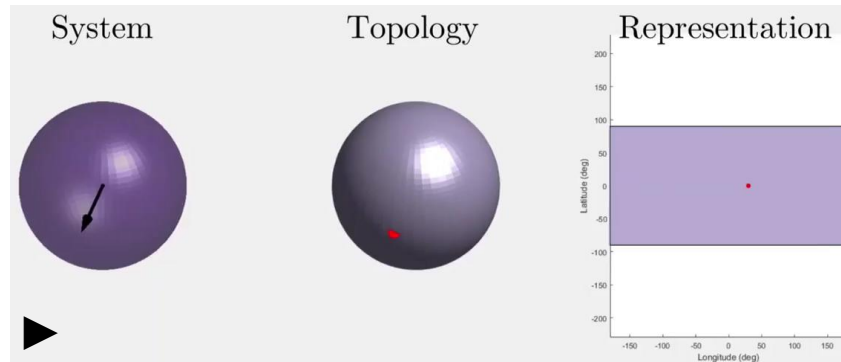
To perform computations, we must have a numerical representation of the space, consisting of a set of real numbers.

Note that the topology of a space is a fundamental property of the space itself and is **independent of how we choose coordinates to represent points in the space.**

system	topology	sample representation	system	topology	sample representation
 point on a plane	 \mathbb{R}^2	 \mathbb{R}^2	 2R robot arm	 $T^2 = S^1 \times S^1$	 $[0, 2\pi) \times [0, 2\pi)$
 spherical pendulum	 S^2	 $[-180^\circ, 180^\circ) \times [-90^\circ, 90^\circ)$	 rotating sliding knob	 $\mathbb{E}^1 \times S^1$	 $\mathbb{R}^1 \times [0, 2\pi)$

Explicit & Implicit Representations

A choice of n coordinates or parameters (in Euclidian space) to represent an n -dimensional C-space is called an **explicit representation** of the C-space.



Disadvantage: **Singularities of Representation**

Explicit & Implicit Representations

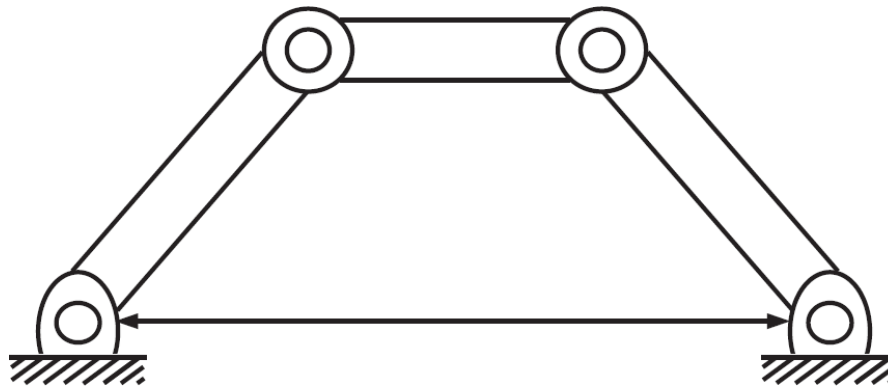
To overcome the Singularities of Representation:

Use an implicit representation which views the n -dimensional space as embedded in a Euclidean space of more than n dimensions.

Configuration and Velocity Constraints

Configuration and Velocity Constraints

For robots containing one or more closed loops, usually an **implicit representation** is more easily obtained than an explicit parametrization.



C-Space: one-dimensional space

Joint Space (J-Space): three-dimensional space

Holonomic Constraints

For general robots containing one or more closed loops:

- Implicit representation of C-space: $\boldsymbol{\theta} = [\theta_1, \dots, \theta_n]^T \in \mathbb{R}^n$
(or J-Space)

- Constraint (loop-closure equations): $\mathbf{g}(\boldsymbol{\theta}) = \begin{bmatrix} g_1(\theta_1, \dots, \theta_n) \\ \vdots \\ g_k(\theta_1, \dots, \theta_n) \end{bmatrix} = \mathbf{0}$

(a set of k independent equations, with $k \leq n$)

- Such constraints are known as **holonomic constraints**.
- These constraints reduce the dimension of implicit representation of C-space.

$$\Rightarrow \text{DOF} = n - k$$

Pfaffian Constraints

Let's suppose that a closed-chain robot is in motion. $\frac{d\mathbf{g}(\boldsymbol{\theta})}{dt} = \mathbf{0}$

$$\begin{bmatrix} \frac{\partial g_1(\boldsymbol{\theta})}{\partial \theta_1} & \dots & \frac{\partial g_1(\boldsymbol{\theta})}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_k(\boldsymbol{\theta})}{\partial \theta_1} & \dots & \frac{\partial g_k(\boldsymbol{\theta})}{\partial \theta_n} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \vdots \\ \dot{\theta}_n \end{bmatrix} = \mathbf{0} \quad \Rightarrow \quad \frac{\partial \mathbf{g}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{A}(\boldsymbol{\theta}) \dot{\boldsymbol{\theta}} = \mathbf{0}$$

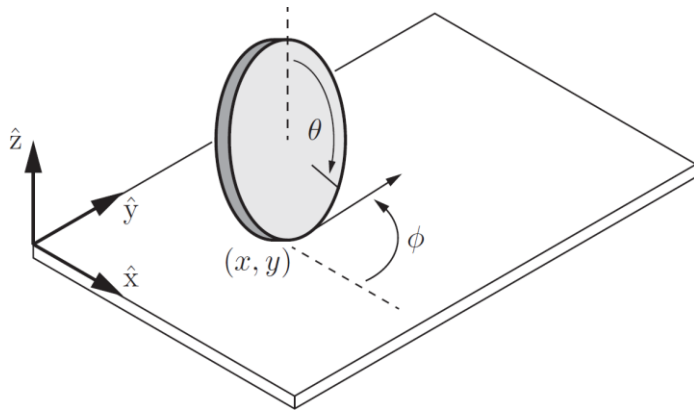
Velocity constraints of this form
are called **Pfaffian Constraints**.

* If a Pfaffian constraint is integrable, the equivalent configuration constraints are
Holonomic Constraints.

Nonholonomic Constraints

* If a Pfaffian constraint of the form $\mathbf{A}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \mathbf{0}$ is nonintegrable to equivalent configuration constraints, it is called a **Nonholonomic Constraint**.

Example: An upright coin of radius r rolling (without slipping) on a plane.



$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = r\dot{\theta} \begin{bmatrix} \cos \phi \\ \sin \phi \end{bmatrix}$$

Nonholonomic constraints reduce the dimension of the feasible velocities of the system but do not reduce the dimension of the C-space.

Workspace and Task Space

Workspace and Task Space

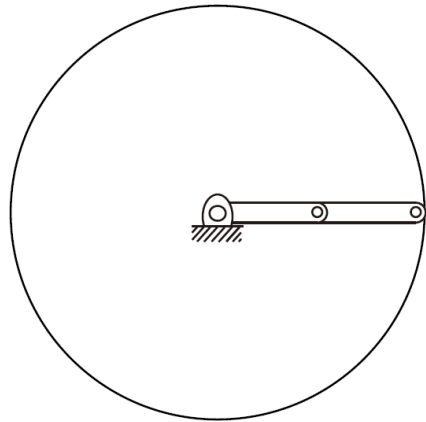
Workspace: The configuration space of the end-effector that the robot can reach (by at least one configuration of the robot), which is primarily determined by the robot's structure and independent of the task.

Task Space (T-Space): The space of configurations as specified by the robot's task itself and independent of the robot.

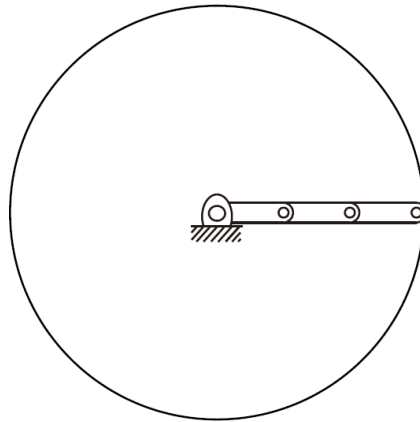
Examples:

- Task space of a robot plotting with a pen on a piece of paper:
- Task space of a robot manipulating a rigid body:
- Task space for operating a laser pointer:
- Task space for carrying a tray of glasses to keep them vertical:

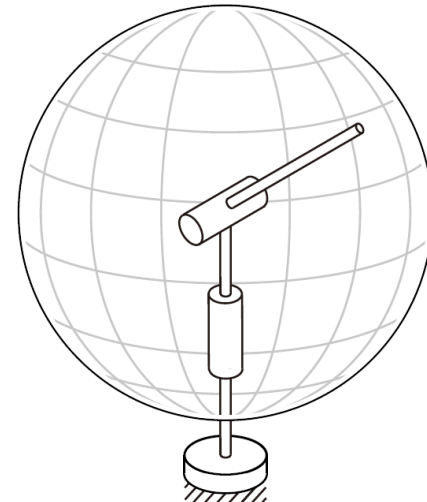
Workspace and Task Space



(2R)



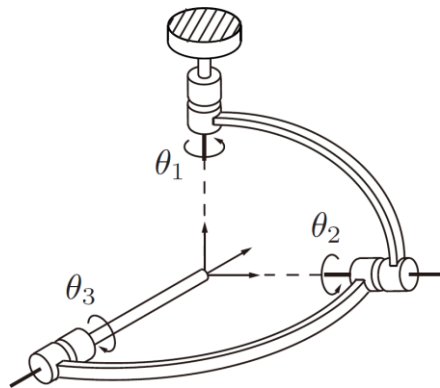
(3R)



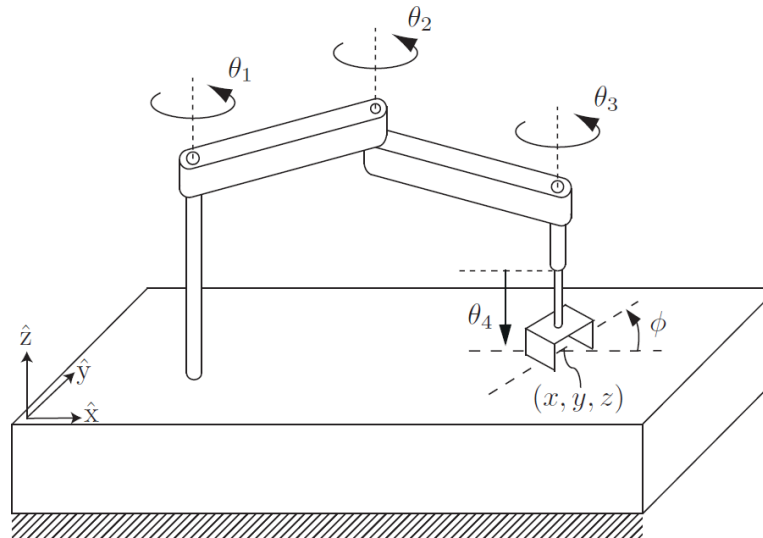
(2R)

- Two mechanisms with different C-spaces may have the same workspace.
- Two mechanisms with the same C-space may also have different workspaces.

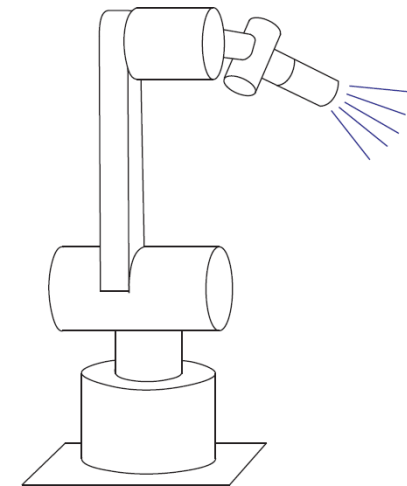
Workspace and Task Space: Some Examples



(3R wrist mechanism)
(for pointing a laser)



(SCARA Robot)
(RRRP)



(A spray-painting robot)
(6R)