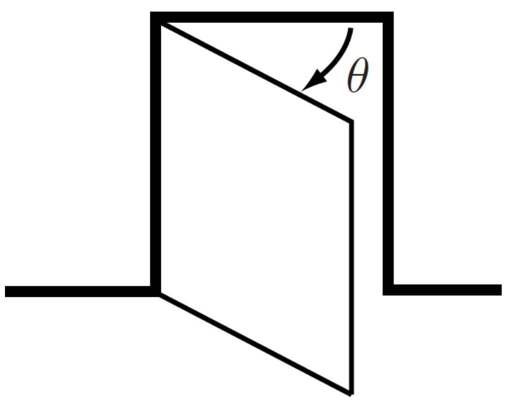
Ch3: Configuration Space and Workspace

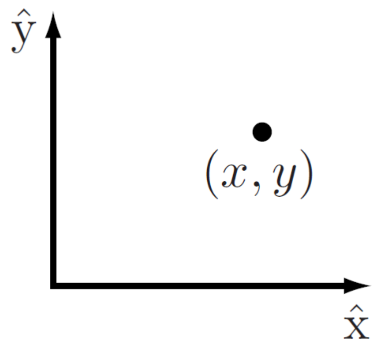
Configuration Space

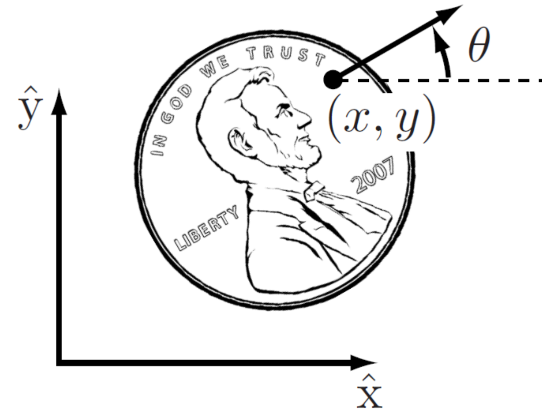
Configuration, DOF, and C-Space of a Robot

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.

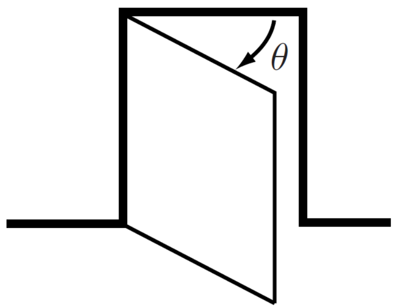


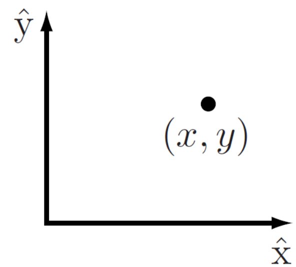


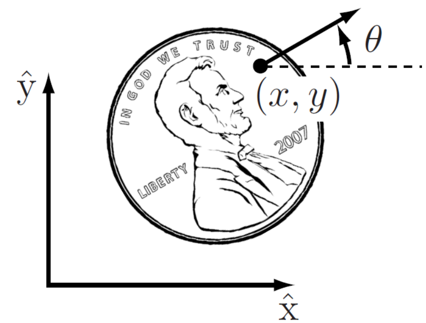


Configuration, DOF, and C-Space of a Robot

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

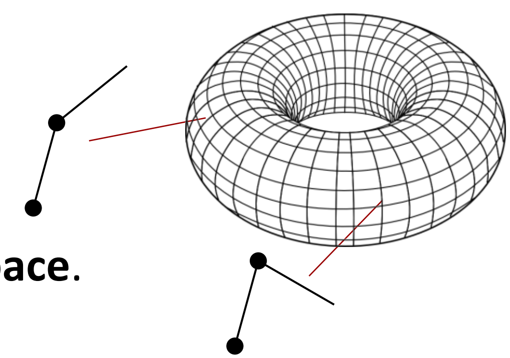






Configuration Space (C-Space): The n-dimensional space containing all possible configurations of the robot/mechanism.

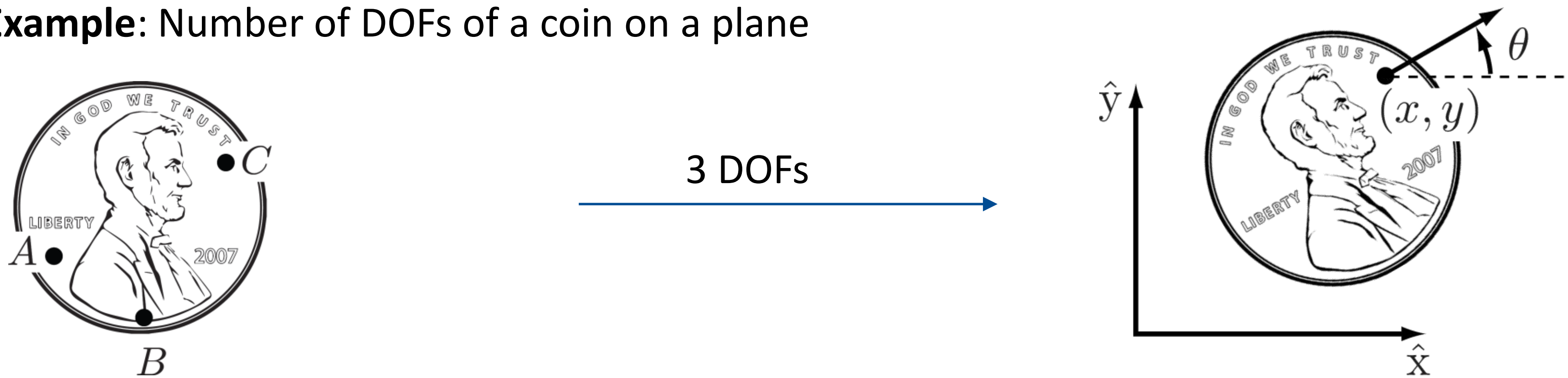
* The **configuration** of a robot is represented by a <u>point</u> in its **C-space**.



Degrees of Freedom (DOF)

DOFs of a Rigid Body in 2D Space

Example: Number of DOFs of a coin on a plane



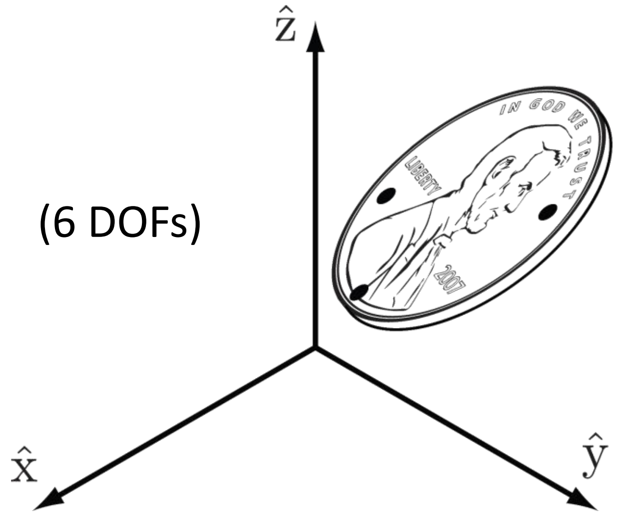
A general rule for determining the number of DOFs of rigid bodies :

DOF = (number of variables) – (number of independent equations/constraints)

DOFs of a Rigid Body in 3D Space

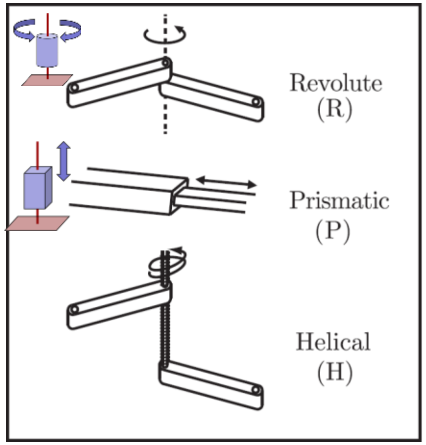
Example: Number of DOFs of a coin in space

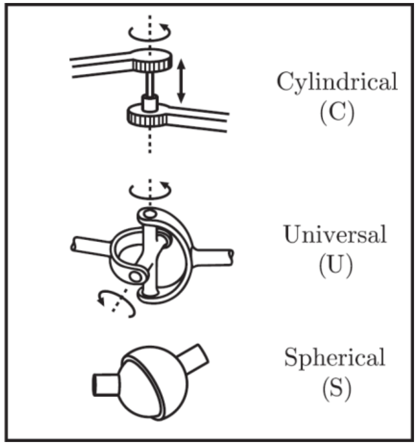
(6 DOFs)



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

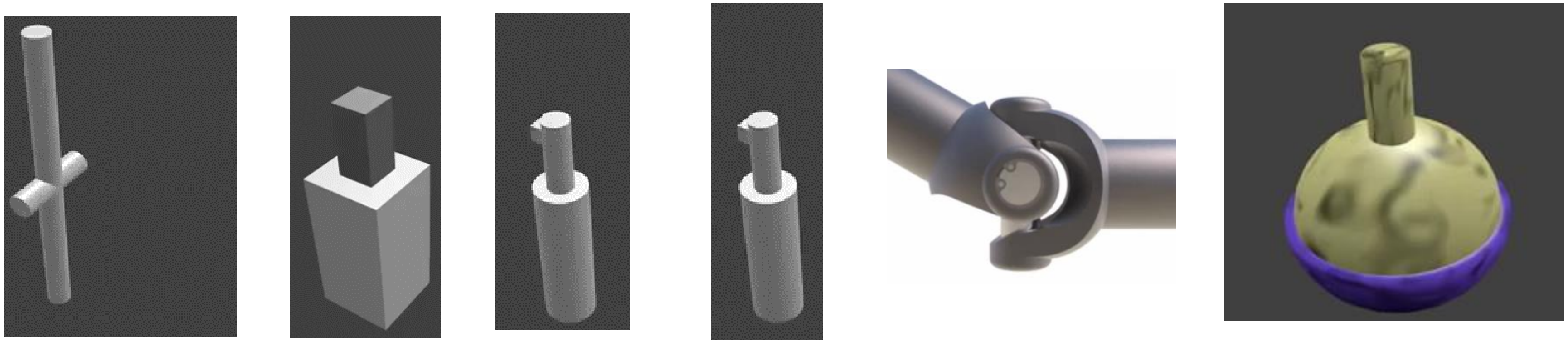
DOFs of Robots: Typical Robot Joints





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.



# of DOFs of a joint = (# of DOFs of a rigid body) – (# 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:

DOF = (sum of freedoms of the bodies) − (number of <u>independent</u> constraints)

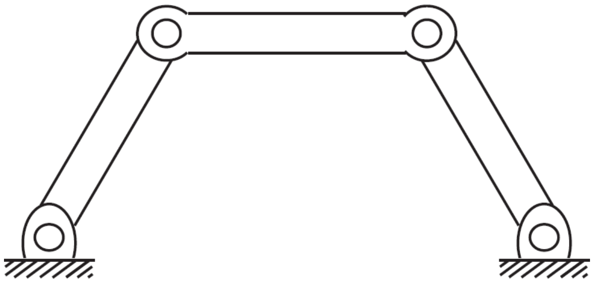
Grübler's Formula for the number of degrees of freedom of mechanisms/robots:

$$\begin{aligned}\text{dof} &= \underbrace{m(N-1)}_{\text{rigid body freedoms}} - \underbrace{\sum_{i=1}^{J} c_i}_{\text{joint constraints}} \\ &= m(N-1) - \sum_{i=1}^{J} (m - f_i) \\ &= m(N-1-J) + \sum_{i=1}^{J} f_i.\end{aligned}$$

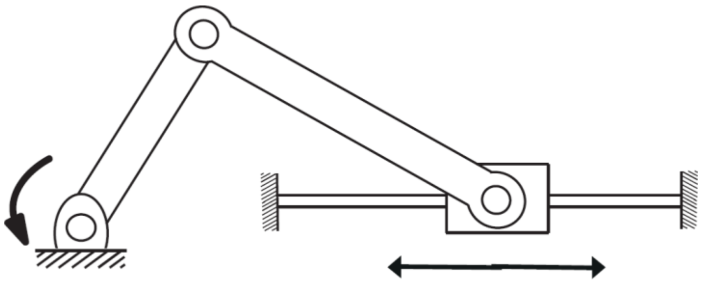
N: number of links (including ground),
J: number of joints,
m: number of DOFs of a rigid body ($m = 3$ for planar mechanisms and $m = 6$ for spatial mechanisms),
f_i: number of freedoms provided by joint i,
c_i: number of constraints provided by joint i ($f_i + c_i = m$ for all i).

Note: This formula holds in "generic" cases, but it fails under certain configurations of the links and joints, such as when the joint constraints are not independent.

Examples: Number of DOFs

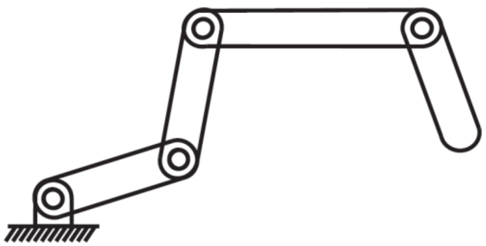


(Four-bar linkage)

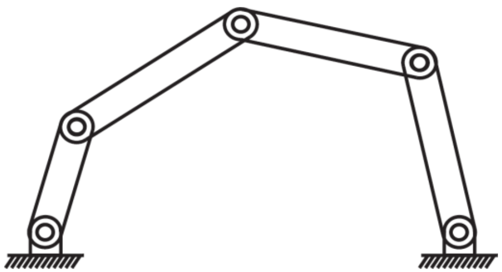


(Slider-crank mechanism)

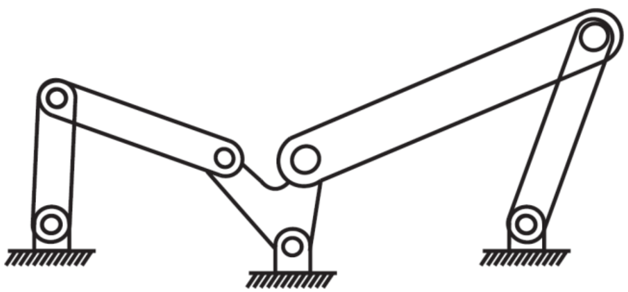
Examples: Number of DOFs



*k*R robot

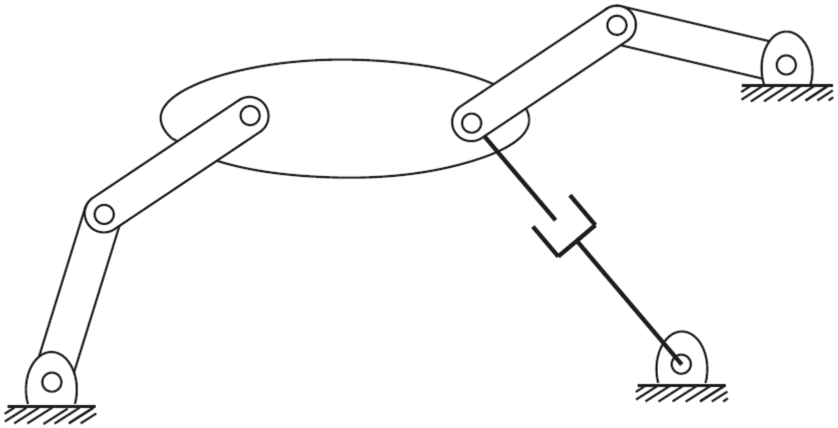


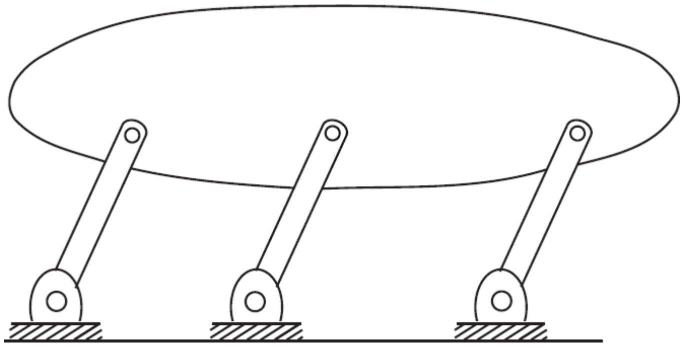
Five-bar linkage



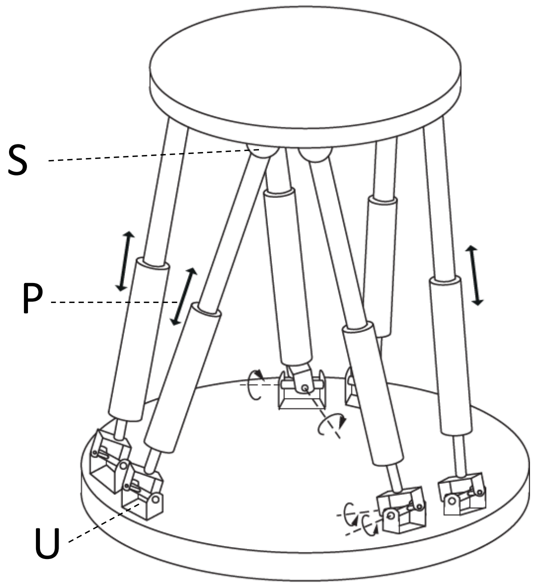
Watt six-bar linkage

Examples: Number of DOFs

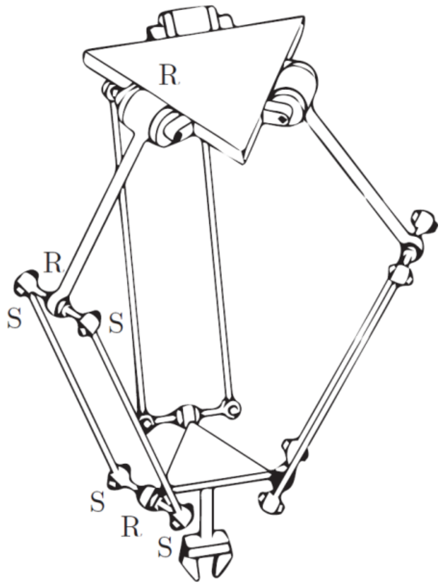




Examples: Number of DOFs



Stewart–Gough platform ▶



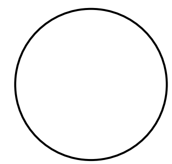
Delta Robot ▶

Configuration Space Topology

Topologies of 1D C-Space

System — **C-Space Topology**

(a) A point moving on a Circle
(or any closed loop):
(Example: A revolute joint without joint limit)



S^1

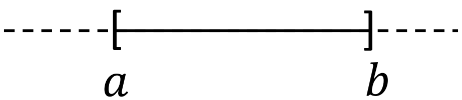
(b) A point moving on a Line:

(Example: A prismatic joint without joint limit)

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

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

(Example: A revolute or prismatic joint with joint limit)



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

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

Topologies of 2D C-Space

System | **C-Space Topology**

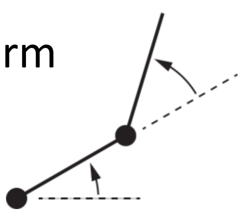
A point moving on a plane

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

Spherical pendulum

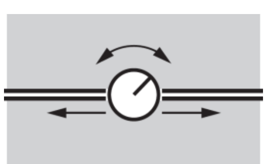
S^2

2R robot arm



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

Rotating sliding robot



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

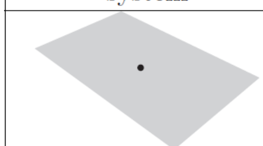
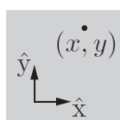
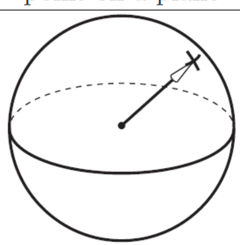
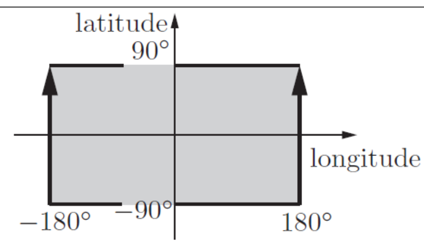
- Hexrotor UAV with two 5-DOF arms (without and with arm joint limits)

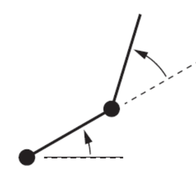
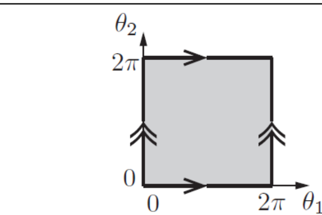
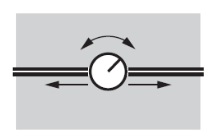
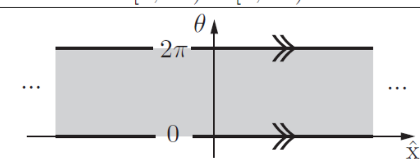
Configuration Space Representation

C-Space Representation

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

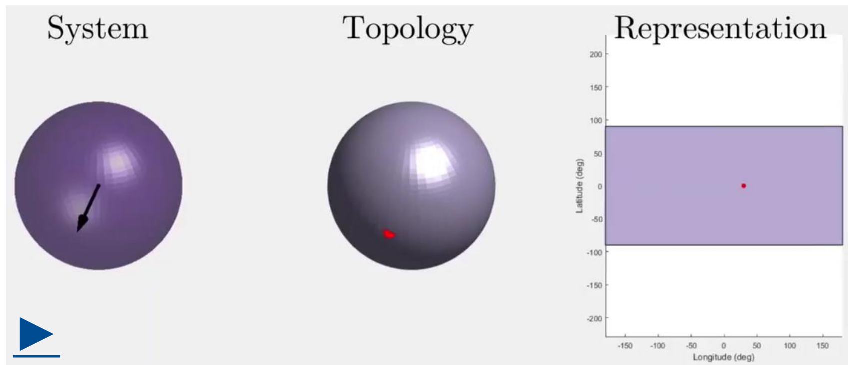
Note: The topology of a space is a fundamental property of the space itself and is **independent of how we choose coordinates to represent points in the space**.

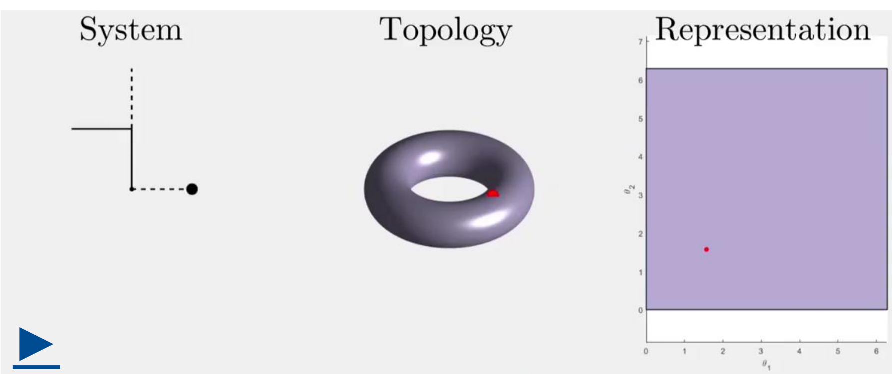
system	topology	sample representation
 point on a plane	 $\mathbb{E}^2$	 $\mathbb{R}^2$
 spherical pendulum	 S^2	 $[-180°, 180°) \times [-90°, 90°]$

system	topology	sample representation
 2R robot arm	 $T^2 = S^1 \times S^1$	 $[0, 2\pi) \times [0, 2\pi)$
 rotating sliding knob	 $\mathbb{E}^1 \times S^1$	 $\mathbb{R}^1 \times [0, 2\pi)$

Explicit & Implicit Representations

A choice of <u>n coordinates</u> or parameters in Euclidian space to represent an <u>n-dimensional</u> C-space is called an **explicit representation** of the C-space.





Disadvantage: Possibility of existence of **Representation Singularities**.

Explicit & Implicit Representations

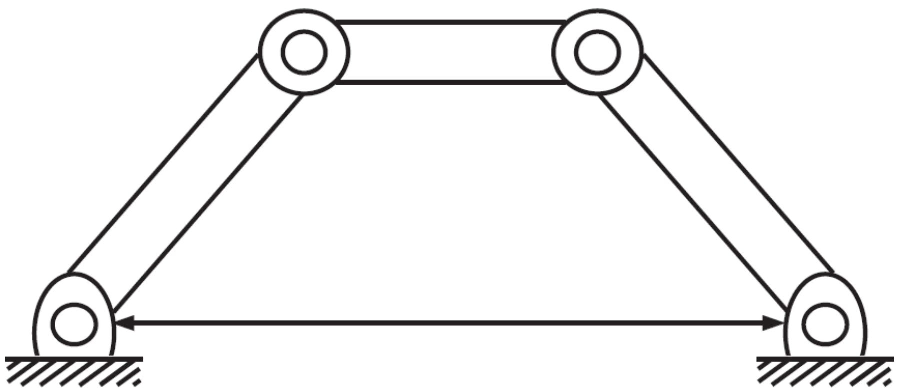
To overcome the Singularities of Representation:

Use an implicit representation which views the n-dimensional C-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

Note: For open-loop mechanisms dimension of C-Space is the same as dimension of J-Space.

Holonomic Constraints

For general robots containing one or more closed loops:

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

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

(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.

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

The C-space can be viewed as a manifold of dimension $n - k$ embedded in $\mathbb{R}^n$.

Pfaffian Constraints

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

$$\begin{bmatrix} \frac{\partial g_1(\boldsymbol{\theta})}{\partial \theta_1} & \cdots & \frac{\partial g_1(\boldsymbol{\theta})}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_k(\boldsymbol{\theta})}{\partial \theta_1} & \cdots & \frac{\partial g_k(\boldsymbol{\theta})}{\partial \theta_n} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \vdots \\ \dot{\theta}_n \end{bmatrix} = \boldsymbol{0} \quad \Rightarrow \quad \frac{\partial \boldsymbol{g}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \boldsymbol{0} \quad \Rightarrow \quad \begin{array}{c} \boldsymbol{A}(\boldsymbol{\theta}) \in \mathbb{R}^{k\times n} \\ \boldsymbol{A}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \boldsymbol{0} \end{array}$$

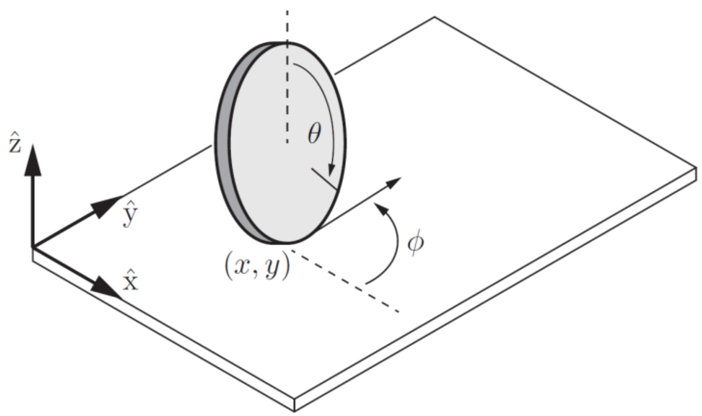
Velocity constraints of this form are called **Pfaffian Constraints**.

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

Nonholonomic Constraints

- If a Pfaffian constraint of the form $\boldsymbol{A}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \mathbf{0}$ 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.



$$\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix} = r\dot{\theta}\begin{bmatrix}\cos\phi\\ \sin\phi\end{bmatrix}$$ (no-slip rolling constraint)

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

* **Holonomic constraints** are constraints on ***configuration***, **nonholonomic constraints** are constraints on ***feasible velocity***.

Workspace and Task Space

Workspace and Task Space

The **Workspace** of a manipulator is the total volume (typically in Euclidian space $\mathbb{R}^3$) swept out by the **end-effector** as the manipulator executes all possible motions. The workspace is constrained by the geometry of the manipulator as well as mechanical constraints on the joints. Workspace is independent of the task.

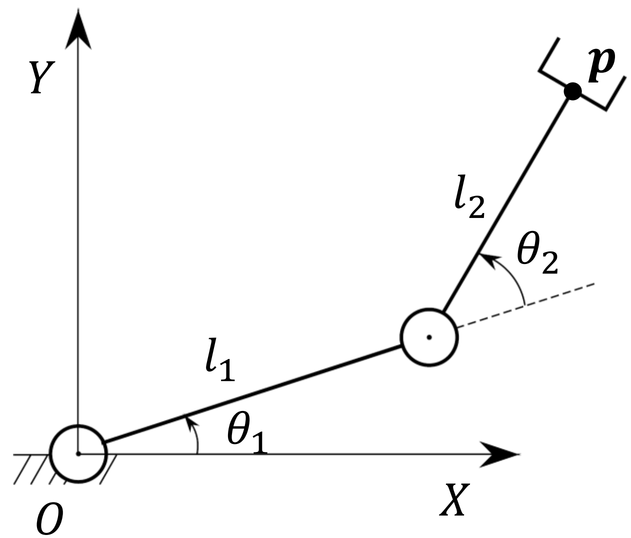
Two types of workspace:

- **Reachable Workspace**: the entire set of points in $\mathbb{R}^3$ reachable by the manipulator with at least one orientation.
- **Dexterous Workspace**: the entire set of points in $\mathbb{R}^3$ that the manipulator can reach with any arbitrary orientation of the end-effector.
  - The dexterous workspace is a subset of the reachable workspace.

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

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 of a Planar 2R Robot



If $l_1 \neq l_2$:

Reachable Workspace: $WS_1 = \{\boldsymbol{p} \in \mathbb{R}^2 : |l_1 - l_2| \leq \|\boldsymbol{p}\| \leq l_1 + l_2\} \subset \mathbb{R}^2$

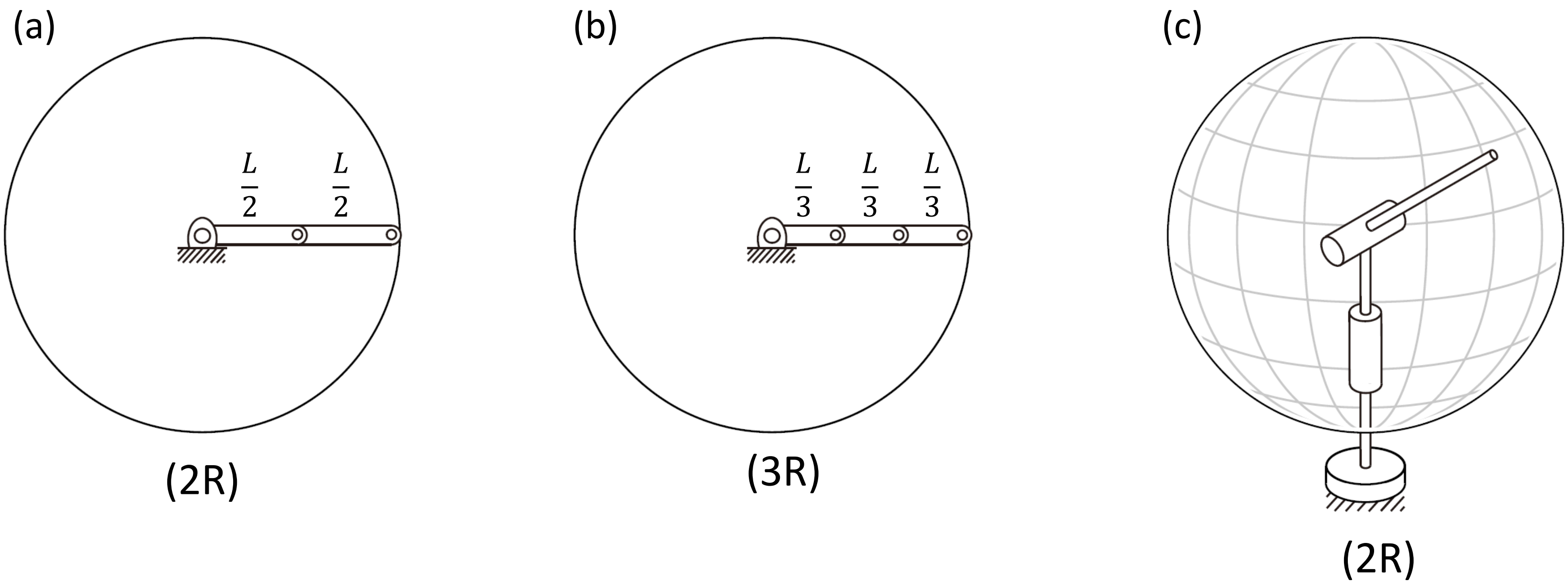
Dexterous Workspac: $WS_2 = \emptyset$

If $l_1 = l_2 = l$:

Reachable Workspace: $WS_1 = \{\boldsymbol{p} \in \mathbb{R}^2 : \|\boldsymbol{p}\| \leq 2l\} \subset \mathbb{R}^2$

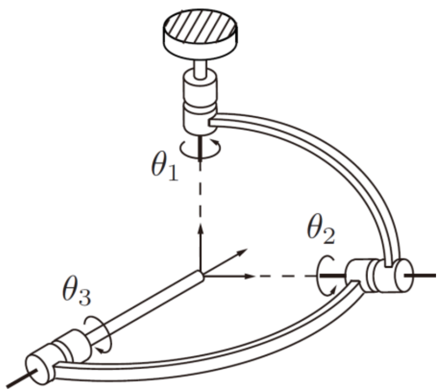
Dexterous Workspac: $WS_2 = \{\boldsymbol{p} = \boldsymbol{0}\}$ (all feasible orientations at the origin!)

C-Space and Workspace

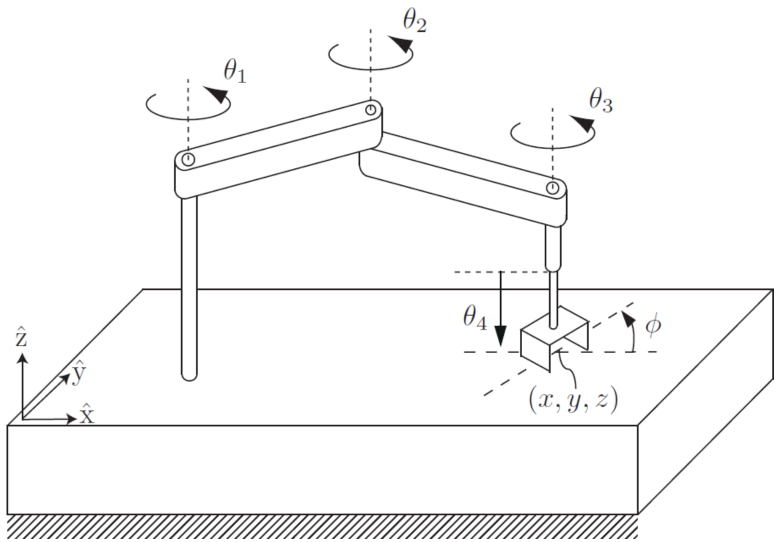


- Two mechanisms with different C-spaces may have the same workspace: (a), (b)
- Two mechanisms with the same C-space may also have different workspaces: (a), (c)

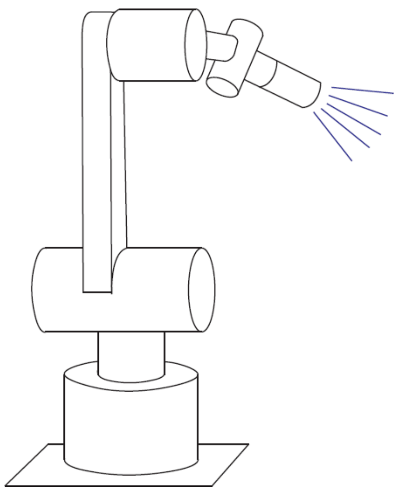
C-Space, Workspace, and Task Space: Some Examples



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



(SCARA Robot)
(RRRP)



(A spray-painting robot)
(6R)