اصول علم ربات – جلسه چهارم

Fundamentals of Robotics – Lecture 04

Pose, Configuration Space, Constraints, DOF, Representations 2

دکتر مهدی جوانمردی

زمستان ۱۴۰۰

[slides adapted from Gianni Di Caro, @CMU with permission]

Recap of main concepts

- **Generalized coordinates**: n parameters $\boldsymbol{q} = (q_1, q_2, \ldots, q_n)$ that are sufficient to uniquely describe system configuration relative to some reference (frame, configuration)
- **State of the system**: (Generalized coordinates, Generalized velocities), represented in the phase space

Configuration space (C-space): the n-dimensional space identified by the generalized coordinates defining the set of all possible robot configurations (based on robot's structure and environmental constraints). Usually, it is a *non-Euclidean manifold.*

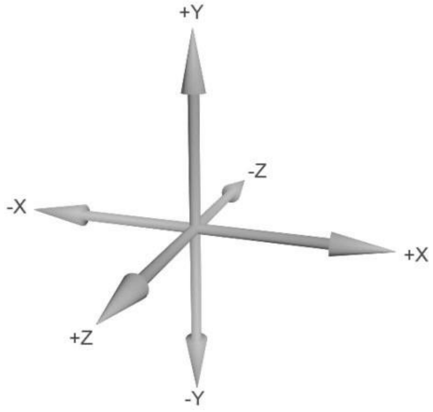
- A ***geometric / holonomic*** *constraint* is expressed through "positional" variables, e.g., (α, β, ϕ_1, ϕ_2, x, y, θ, …), it only involves generalized coordinates, not their derivatives. It limits *the motion of the system to a manifold of the configuration space*, depending on the initial conditions

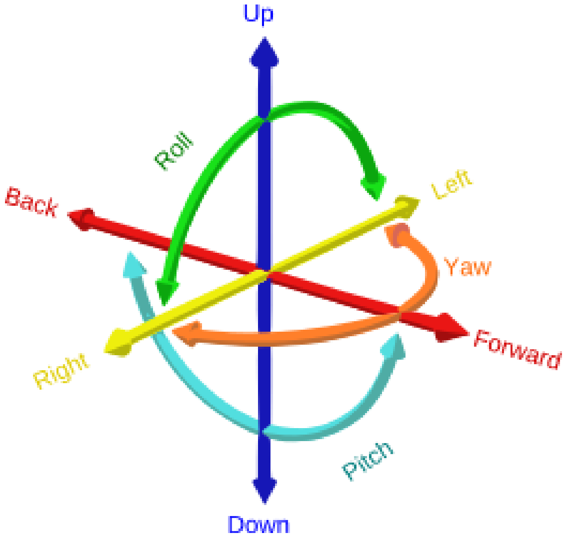
Degrees of freedom: A system whose configuration is described by n *independent* generalized coordinates has n degrees of freedom.

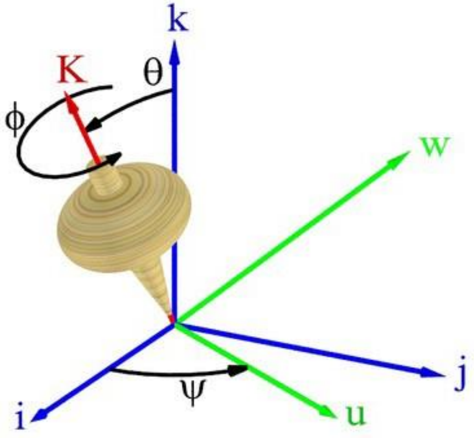
If there are *m independent functional relations* (holonomic constraints) among a chosen set of n generalized coordinates, the number of DOF is $n - m$*:* (number of variables - number of independent equations)

Degrees of freedom of a rigid body in 3D

How many DOF?

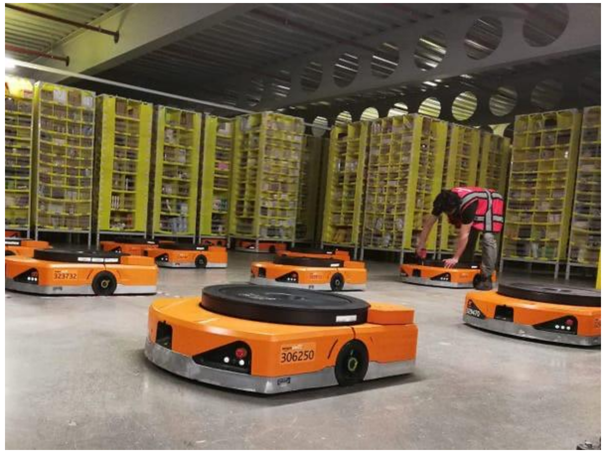




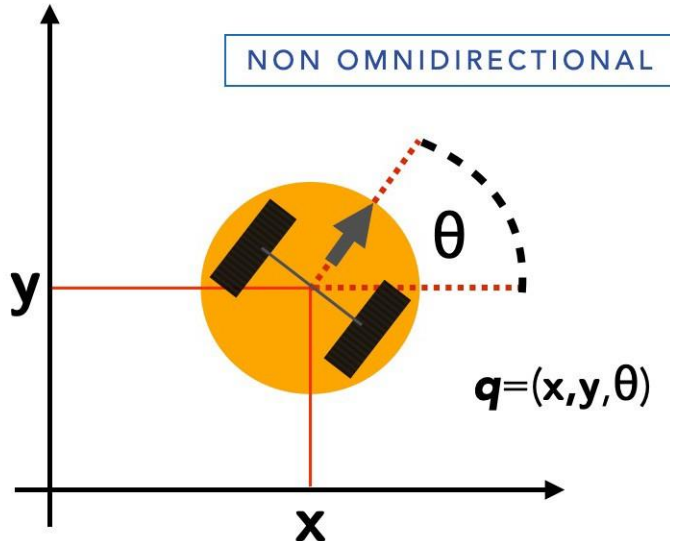


Single Rigid body in 3D → 6 DOF

Degrees of freedom of a rigid body in 2D (Planar)

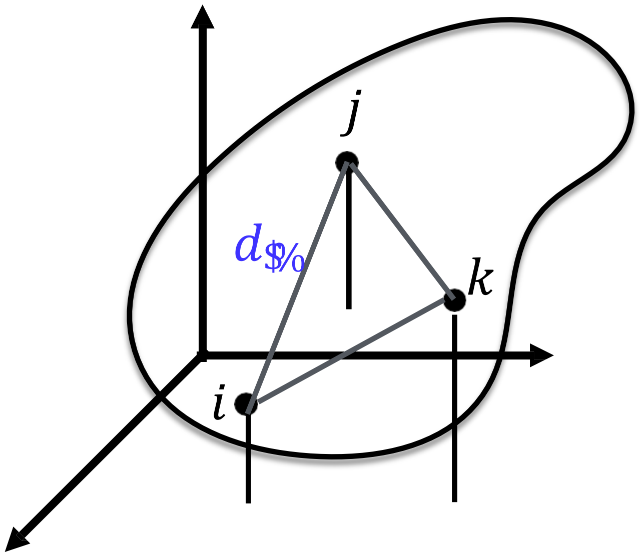


How many DOF?



Rigid body in 2D → 3 DOF

Degrees of freedom of a rigid body in 3D



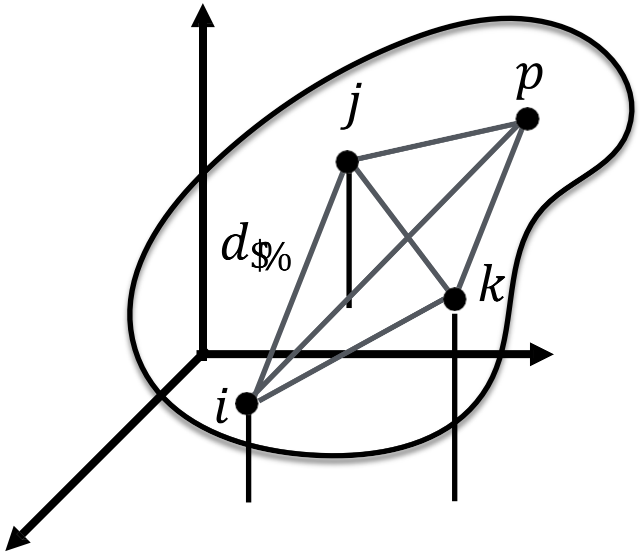
- A rigid body is modeled as a system of *at least* three **non-collinear particles** whose positions relative to one another remain fixed. i.e., distance $d_{\$\%}$ between any two particles i and j remains constant throughout the motion (due to internal forces).
- In general, a rigid body is made of $N \gg 3$ particles
- To specify the position (x, y, z) of each particle , we would need $n = 3N$ generalized coordinates

❖ Distance constraint between all pairs of particles (holonomic, scleronomic):

$$d_{\mathrm{i,j}} = constant_{\mathrm{\,i,j}}, \forall i \neq j = 1,2, \ldots N \Rightarrow C_{\mathrm{N}} = \frac{N(N-1)}{2} \text{ constraints}$$

- Are the # of DOF equal to $3N - C_{\mathrm{N}}$? No, not all $C_{\&}$ constraints are independent!
- We know that the rigid body has 6 DOF …

Degrees of freedom of a rigid body in 3D

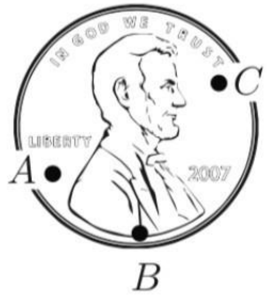


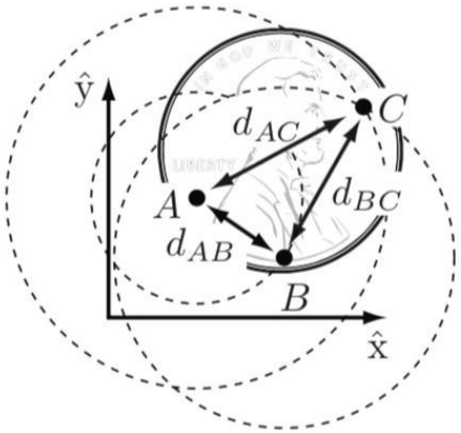
- A system of 3 particles in 3D needs 9 generalized coordinates. There are 3 independent distance constraints → 9 - 3 = 6 DOF

- What about a new particle p? → 3 more coordinates + 3 more constraints → 0 freedoms

- Any additional point would contribute with 3 more coordinates but will determine 3 more independent constraint equations (wrt the original three points, all other distances are fixed depending on these) → 0 freedoms

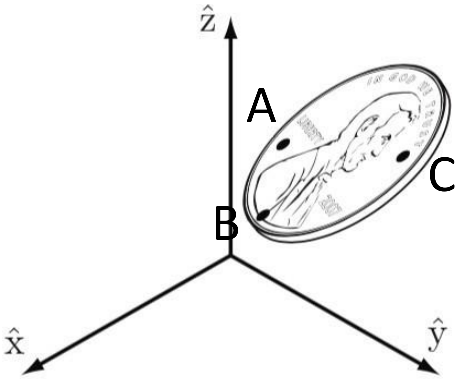
Degrees of freedom of a rigid body: Coin in a plane

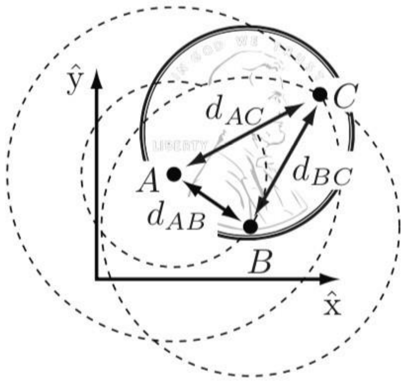




- DOF of a coin in a plane: freedoms choosing three arbitrary reference points (i.e., **collinear** particles) with given constant distances between them
- Once the location (x, y) of A is chosen (2 freedoms), B must lie on a circle of radius $d_{\mathrm{A\,B}}$ centered at A (1 freedom, angle θ)
- Once the location of B is chosen, C must lie at the intersection of circles centered at A and B → 0 freedom
- The coin in the plane has 3 DOF: (x, y, θ)

Degrees of freedom of a rigid body: Coin in 3D

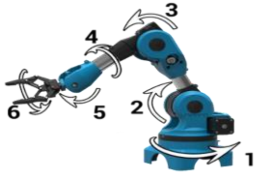




- Point A can be placed freely in the space → **3 freedoms** (x, y, z)
- Location of B is subject to the constraint d_{AB}: it must lie on the *sphere* of radius d_{AB} centered at A→ 3-1 = **2 freedoms** (φ, ψ) (e.g., latitude and longitude on the sphere)
- Location of point C must lie at the intersection of spheres centered at A and B of radius d_{AC}, d_{BC} ,respectively
- The intersection of two spheres is a *circle*, that can be parametrized by an angle → 1 **freedom** (θ)
- DOF = 3 + 2 + 1 = 6

DOF and robot control (we'll see it later)








	dim $\mathcal{C}$	Degrees of freedom	Number of actuators	Actuation	Rolling constraints	Holonomic
Train	1	1	1	full		✓
2-joint robot arm	2	2	2	full		✓
6-joint robot arm	6	6	6	full		✓
10-joint robot arm	10	10	10	over		✓
Hovercraft	3	3	2	under		
Car	3	3	2	under	✓	
Helicopter	6	6	4	under		
Fixed wing aircraft	6	6	4	under		
DEPTHX AUV	6	6	6	full		✓

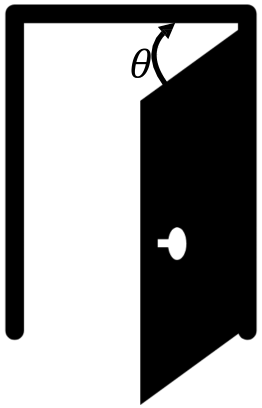
DOF and robot control

	dim $\mathcal{C}$	Degrees of freedom	Number of actuators	Actuation	Rolling constraints	Holonomic
Train	1	1	1	full		✓
2-joint robot arm	2	2	2	full		✓
6-joint robot arm	6	6	6	full		✓
10-joint robot arm	10	10	10	over		✓
Hovercraft	3	3	2	under		
Car	3	3	2	under	✓	
Helicopter	6	6	4	under		
Fixed wing aircraft	6	6	4	under		
DEPTHX AUV	6	6	6	full		✓

- DOF / dimension of the C-space defines the number of parameters the robot can independently act upon to change its configuration: If there is an actuator for each DOF then each DOF is **controllable**
- If not all DOF are directly controllable the control problems are (much) harder → Underactuation
- The number of controllable DOF determines how easy/hard the robot control problem will be

- Holonomic robots: # of controllable DOF is the same as the # DOF
- Non holonomic robots: # of controllable DOF is lesser than the # of DOF (we don't have full controls!)
- Redundant robot: # of controllable DOF is larger then # of total DOF (over actuated robot)
- E.g., Human Arm 6 DOF – Position and orientation of the Fingertip in 3D space: 7 actuators - 3 shoulder, 1 elbow, 3 wrist (it would only require 6DOFs)

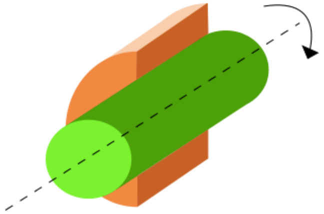
DOF of a multi-link robot: a door

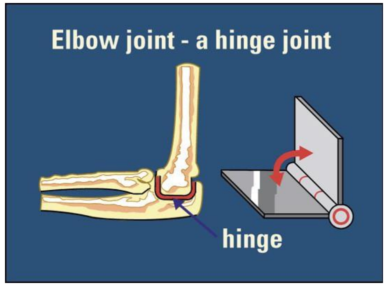


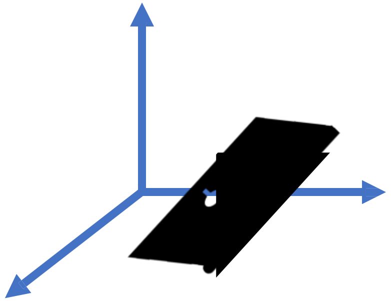
$q = (\theta)$ $\quad C = S^1$

Revolute (hinge) joint: motion is only permitted in one plane

Door system has 1 DOF

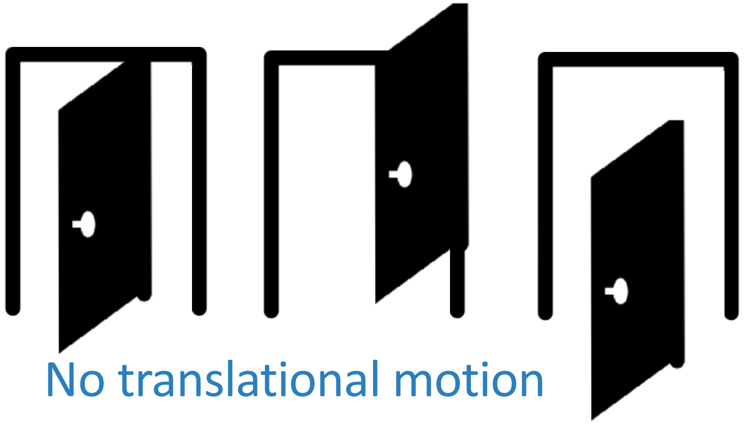




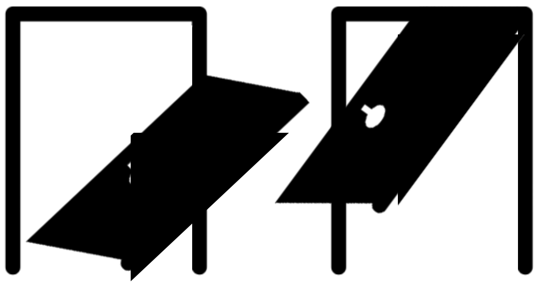


Without the joint, the door would be free to move in the 3D space → 6 DOF

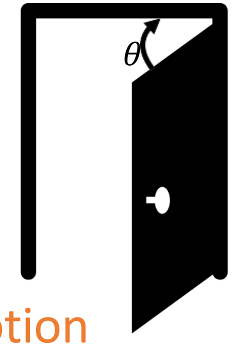
- A joint *connects* two rigid bodies (door, wall) and can be regarded in a *dual way*:
  - As allowing some freedom of motion between the two bodies, in this case, one freedom
  - As imposing constraints on the motion of one rigid body relative to the other, five constraints in this case



No translational motion

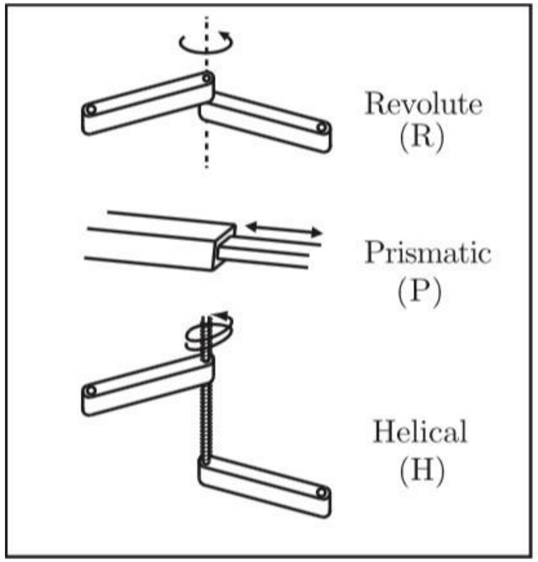


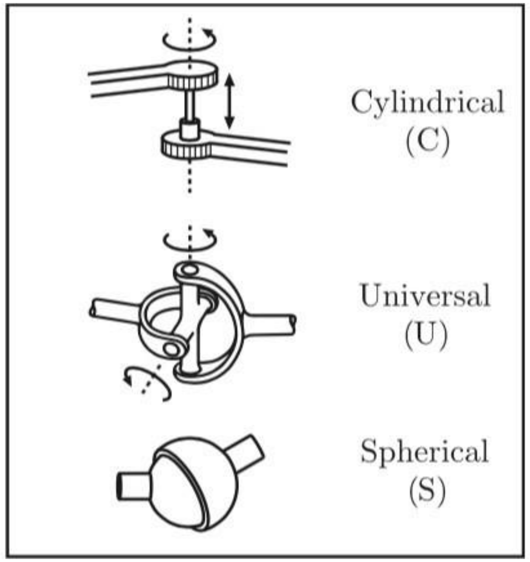
No roll and pitch



Yaw motion

DOF of a joint (multi-link robot)

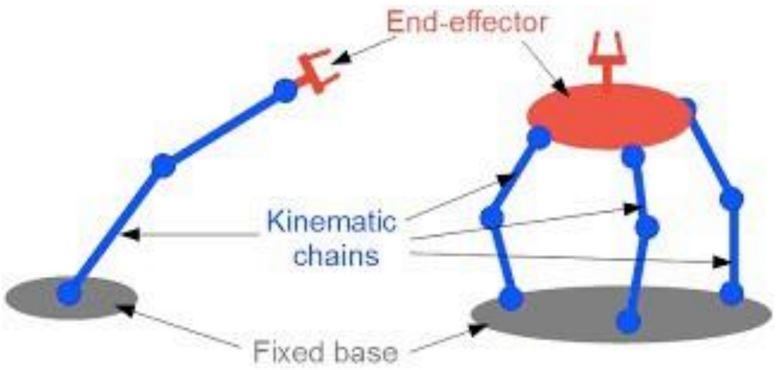




Joint type	dof f	Constraints c between two planar rigid bodies	Constraints c between two spatial rigid bodies
Revolute (R)	1	2	5
Prismatic (P)	1	2	5
Helical (H)	1	N/A	5
Cylindrical (C)	2	N/A	4
Universal (U)	2	N/A	4
Spherical (S)	3	N/A	3

DOF provided by the joint =
DOF(rigid body) - # constraints imposed by the joint

Open chain vs. closed (kinematic) chain mechanisms



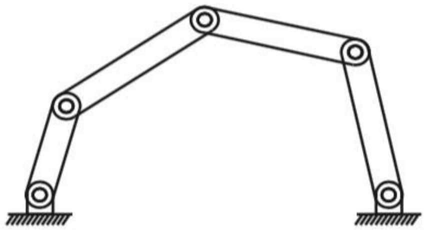
- We are eventually interested in acting upon the degrees of freedom of the robot to control the pose of the **end-effector**

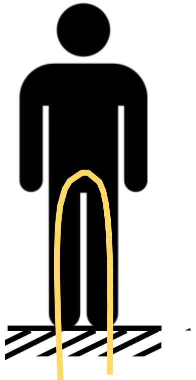
Open-chain (serial) mechanisms:
any mechanism that **doesn't have closed loops**

If the arm-hand can move freely

All the joints are actuated

Closed-chain mechanisms:
any mechanism that **has closed loops among the links**

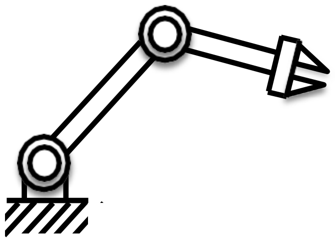




Only a **subset** of the joints may be actuated (i.e., some joints may be *passive*)

DOF of a multi-link robot: Grubler's formula

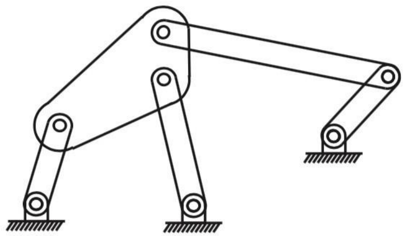
- Consider a mechanism consisting of N links, where ground is also regarded as a link
- J = number of joints
- m = number of degrees of freedom in the space in which the mechanism functions (m = 3 for planar mechanism, m = 6 for spatial mechanisms),
- f_i= number of freedoms provided by joint i
- c_i = number of constraints imposed by joint i, where $f_i + c_i = m, \quad \forall i$



Then, the **number of degrees of freedom of the robot** is:

Grubler's formula

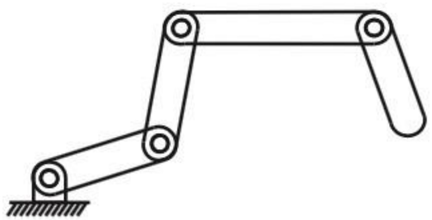
$$
\begin{aligned}
\text{dof} &= \underbrace{m(N-1)}_{\text{rigid body freedoms}} - \underbrace{\sum_{i=1}^{J} c_i}_{\text{joint constraints}} \\
&= m(N-1) - \sum_{i=1}^{J} (m - f_i) \\
&= m(N-1-J) + \sum_{i=1}^{J} f_i.
\end{aligned}
$$



- If all joint constraints are not independent (i.e., there are *redundant joints*) the formula only provides a ***lower bound***

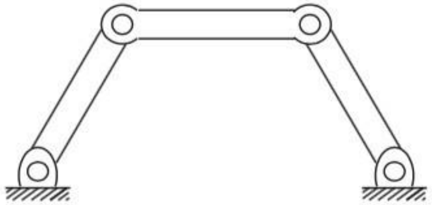
Application of Grubler’s formula

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



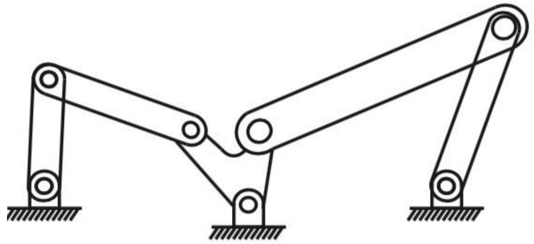
k-link planar serial chain
(kR robot, k revolute joints)

4+1 links
4 joints
f_i=1
m=3
DOF=4

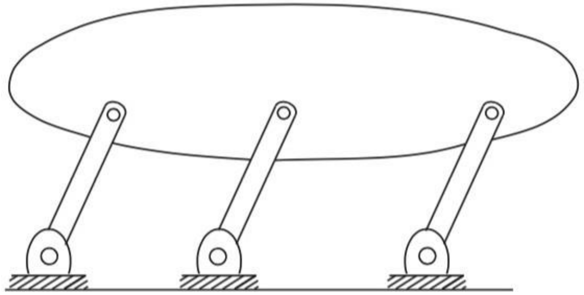


Planar 4-bar linkage (with ground): DOF = 1

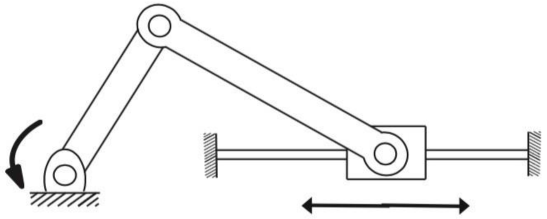
m=3
N=4
4 joints
f_i=1
DOF=1



Watt six-bar linkage: DOF = 1

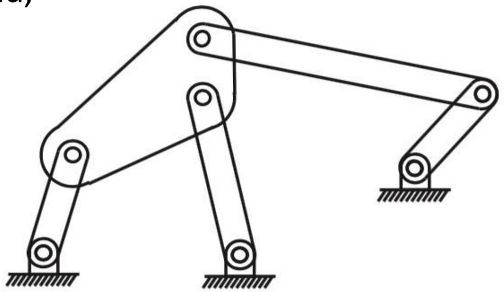


Planar parallelogram linkage: DOF = 1

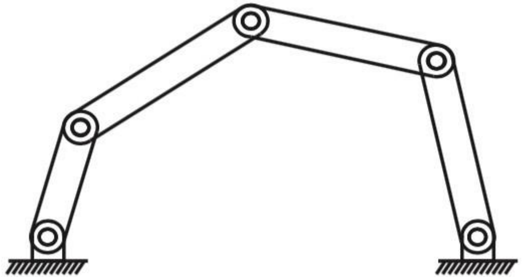


Planar slider-crank linkage: DOF = 1

N=6, 5 links plus ground,
7 revolute joints:
DOF= 3(6-1-7)+7 = 1



Stephenson six-bar linkage: DOF = 1



N=5, 4 links plus ground,
5 revolute joints:
DOF= 3(5-1-5)+5 = 2

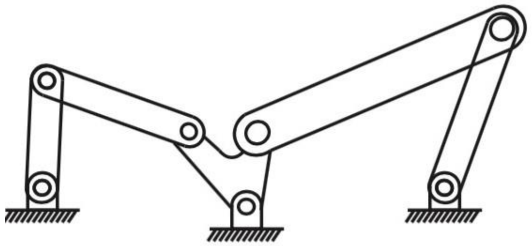
Planar four-bar linkage: DOF = 2

Application of Grubler’s formula

$$\text{dof} = \underbrace{m(N-1)}_{\text{rigid body freedoms}} - \underbrace{\sum_{i=1}^{J} c_i}_{\text{joint constraints}}$$

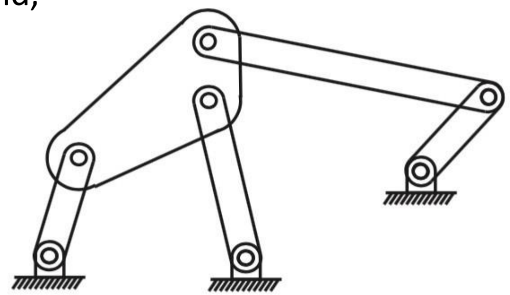
$$= m(N-1) - \sum_{i=1}^{J}(m - f_i)$$

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

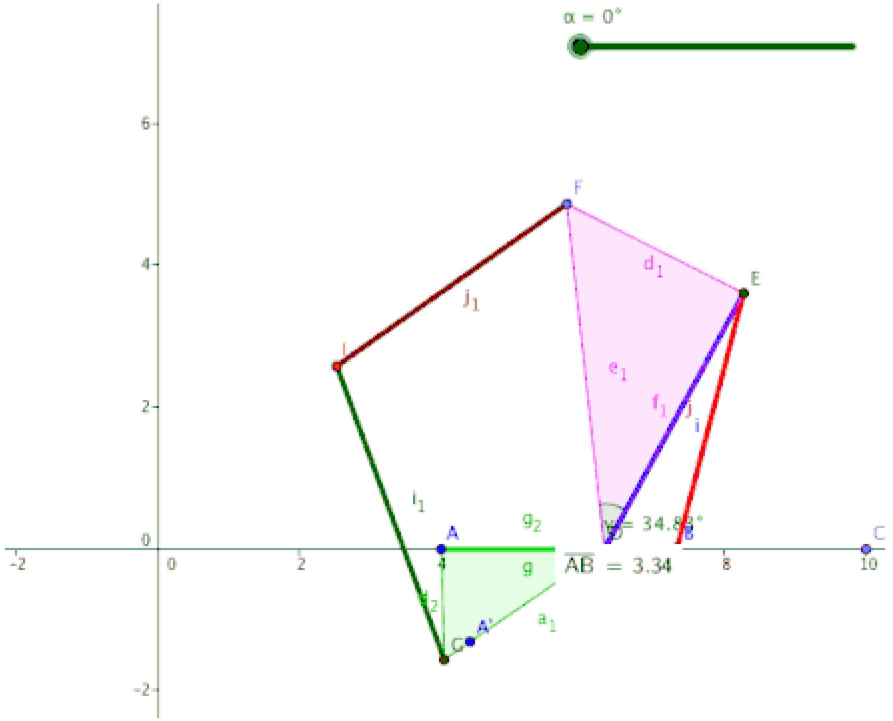


Watt six-bar linkage: DOF = 1

N=6, 5 links plus ground,
7 revolute joints:
DOF= 3(6-1-7)+7 = 1



Stephenson six-bar linkage: DOF = 1

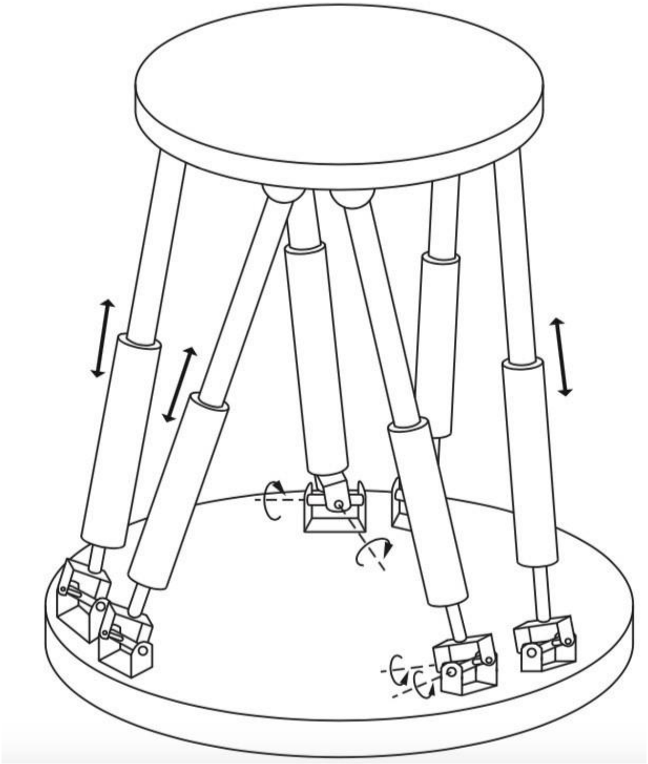


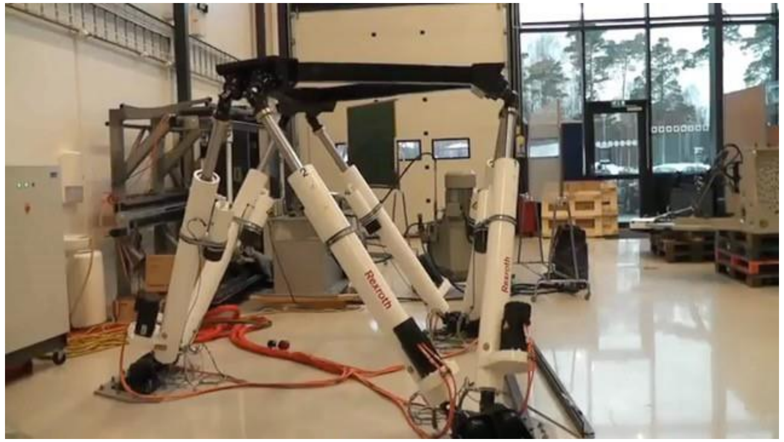
Application of Grubler's formula

$$\text{dof} = \underbrace{m(N-1)}_{\text{rigid body freedoms}} - \underbrace{\sum_{i=1}^{J} c_i}_{\text{joint constraints}}$$

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

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





Stewart–Gough platform

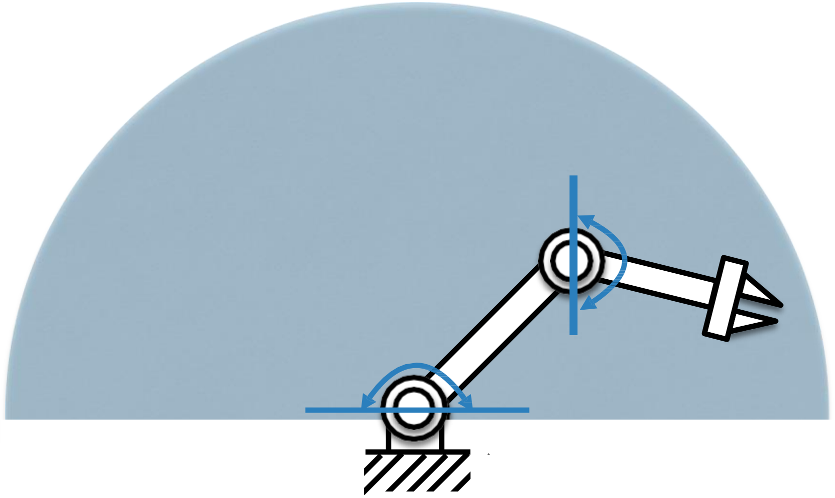
14 links (including ground platform)
6 Universal joints, 2 freedoms (legs-ground)
6 Prismatic joints, 1 freedom
6 Spherical joints, 3 freedoms (legs-upper platform)

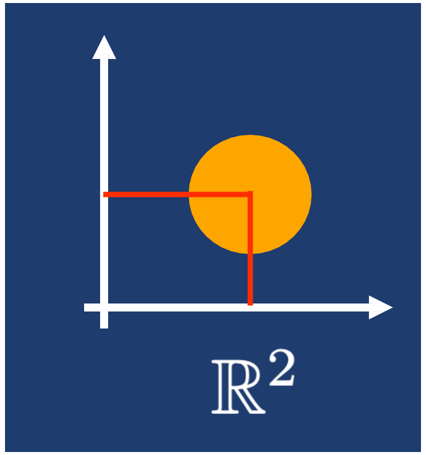
DOF = 6

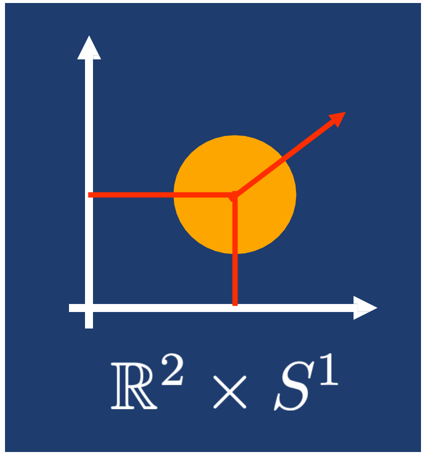
Workspace

Workspace $\mathcal{W}$: A robot's workspace (or workspace envelope) is the set $\mathcal{W}$ of all points that the robot, based on its structure, can feasibly reach in the physical embedding volume to perform its "work".

- It depends both on robot structure and what the user targets has important for the work to be done (e.g., the orientation of the end-effector might be irrelevant)
- For a planar kinematic chain, the workspace can be either a subset of $\mathbb{R}^2$ or a subset of $\mathbb{R}^2 \times S^1$ if orientation matters for the robot "work"







- Same for a mobile robot in the open plane, depending whether the orientation of the robot matters or not

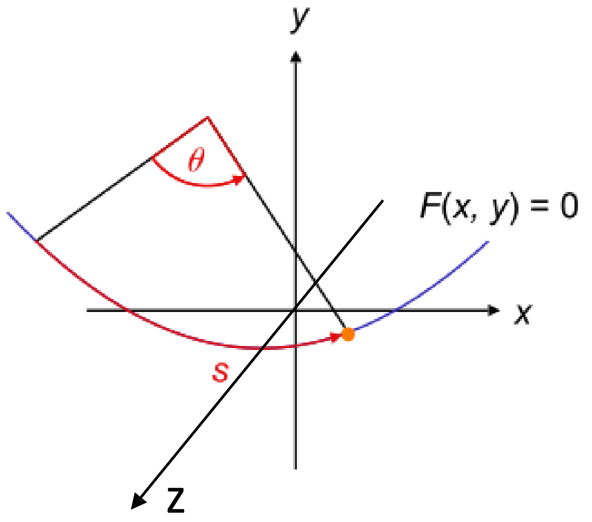
Task space

Task space $\mathcal{T}$: Set of all possible (not necessarily achievable) *poses* in a user-defined space related to the task the robot is used for.

- Example: A robot train, constrained to move on a rail from a given starting point

- Task is motion along the rail: $\boldsymbol{\mathcal{T}} \subset \mathbb{R}$, $\boldsymbol{\mathcal{T}} \equiv C$
- Task asks about the (x, y) position of the train in a plane: $\boldsymbol{\mathcal{T}} \subset \mathbb{R}^2$
- Task requires 3D robot train position and orientation: $\boldsymbol{\mathcal{T}} \subset \mathbb{R}^3 \times S^3$

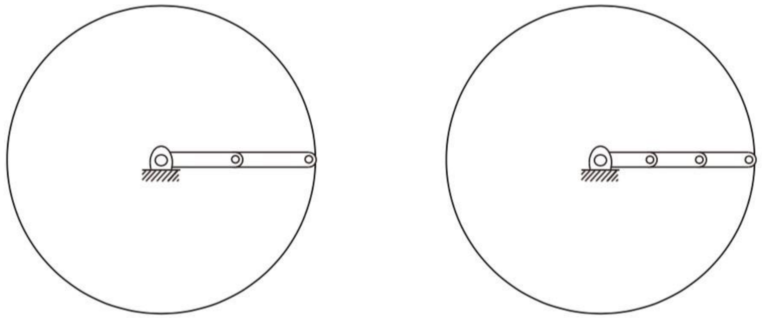
➢ In the last two cases the dimension of the task space exceeds the dimension of the configuration space: robot moves on a manifold in the (higher dimensional) task space, there is a mapping from q to $\boldsymbol{\xi} \in \boldsymbol{\mathcal{T}}$



$$q = (s)$$

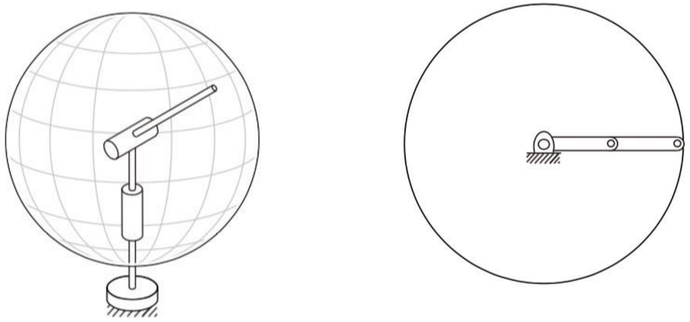
Properties of Task space and workspace

- A point in the task space or the workspace may be achievable by more than one robot configuration: the point is not a full specification of robot's configuration. → Different motion plans / poses can be used for 'work/task'
- Some points in the task space may not be reachable at all by the robot
- By definition, all points in the workspace are reachable by at least one configuration of the robot
- Two robot mechanisms with different C-spaces may have the same workspace



If orientation of the end-effector doesn't matter

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



Spaces for a SCARA robot

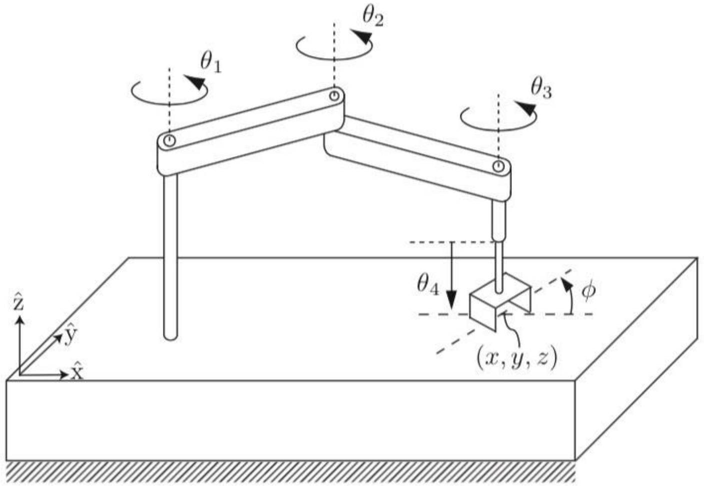
- Selective Compliance Articulated Robot Arm (SCARA, 1981)
  - Compliant in (x, y) but quite "limited" in the z axis → "Selective compliance"
  - Two-link structure similar to human arm → "Articulated"
  - Good for extend / retract, pick-up and move/place tasks

https://youtu.be/sWrLojkCgVw

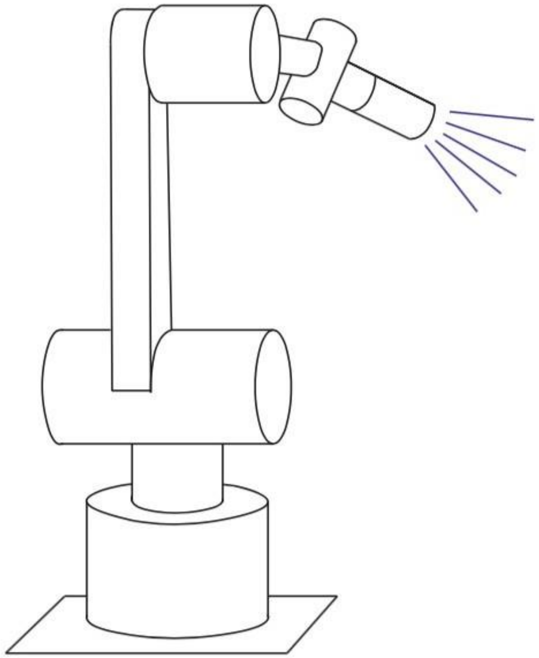
SCARA are faster, but more complex to control and expensive than Cartesian robots

https://global.yamaha-motor.com/business/robot/lineup/xyx/

- RRRP open chain
- The end-effector configuration is completely described by four parameters (x, y, z, ϕ)
- $\mathscr{T} = \mathbb{R}^3 \times S^1$
- $\mathscr{W} \subset \mathbb{R}^3$ corresponding to the reachable (x, y, z) points, since all ϕ orientations are achievable at any reached point



Spaces for a spray-painting robot



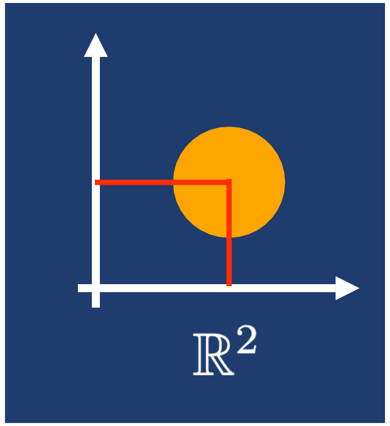
https://www.youtube.com/watch?v=nee3vYTJSr4

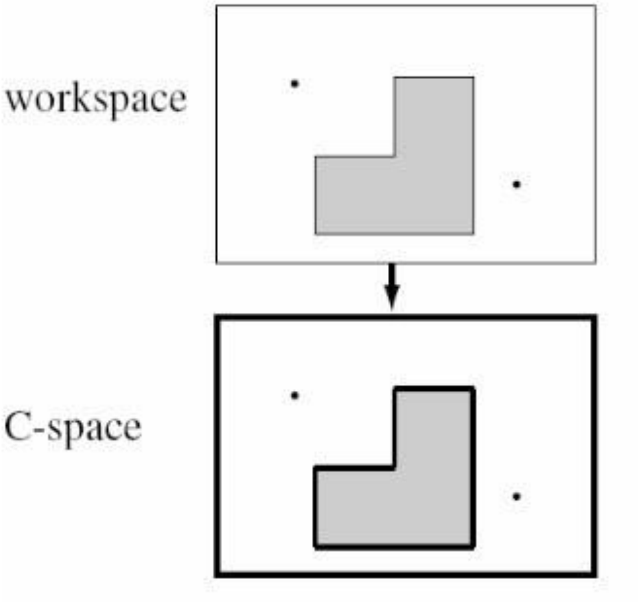
- For the task, important are:
  - Cartesian position of the spray nozzle
  - Direction in which the spray nozzle is pointing
  - Rotations about the nozzle axis axis (which points in the direction in which paint is being sprayed) usually don’t matter

$\mathcal{T}$ is $\mathbb{R}^3 \times S^2$

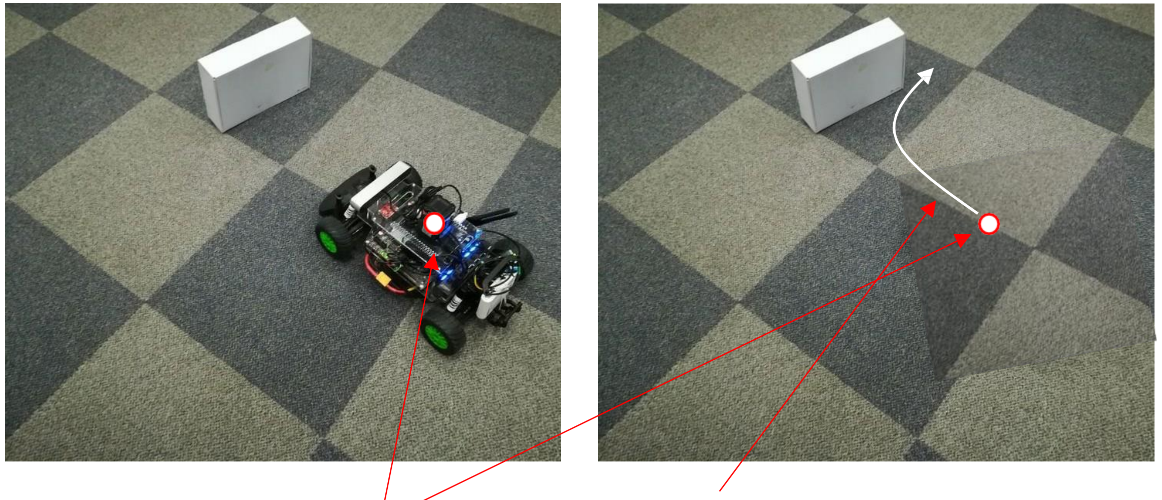
$\mathcal{W}$ can be the same $\mathbb{R}^3 \times S^2$ or just $\mathbb{R}^3$

C-space and Workspace of a mobile robot in presence of obstacles





(a)

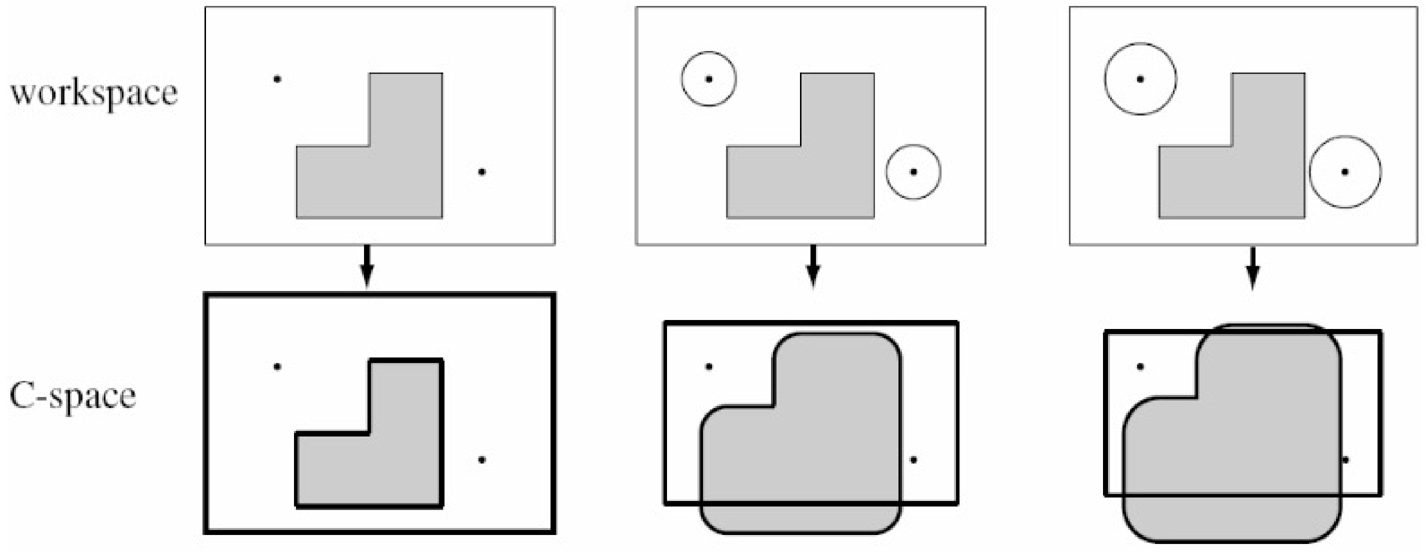


Point abstraction

Trajectory feasible for the point abstraction but not in practice, robot would crash into obstacle!

- For a planar omnidirectional mobile robot (single body) in an open space, both Workspace and C-space are $\mathbb{R}^2$, or a regular subset X of it (orientation doesn't matter because of omnidirectionality).
- However, in presence of objects obstructing potential robot navigation (obstacles), the subset X must take these into account, becoming a more complex manifold.
- In figure (a) the point abstraction is used to *reduce* the robot to a point. In this case, $X \subseteq \mathbb{R}^2$ is the set corresponding to the white background, which is obstacle free. However, using this abstraction we are including in $\mathcal{C}$ and $\mathcal{W}$ configuration points that wouldn't be accessible to the robot because of its actual finite size and shape!

C-space and Workspace of a mobile robot in presence of obstacles

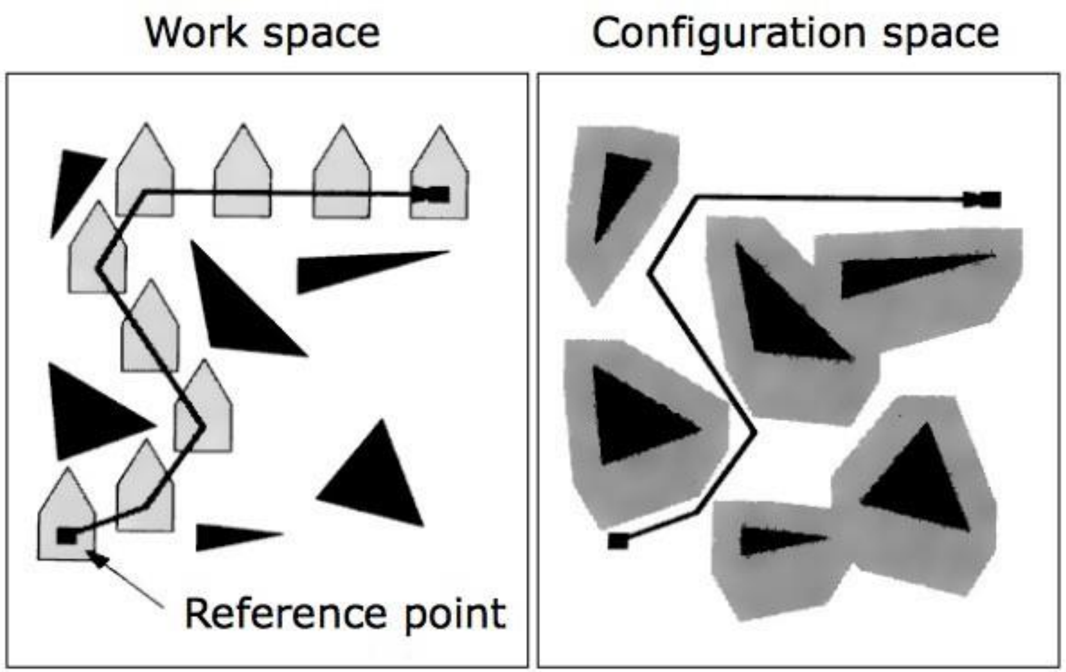


(a) (b) (c)

- To simplify the definition of $\mathcal{C}$ and $\mathcal{W}$ as well as to ensure that the corresponding sets allow a feasible planning and execution of navigation paths, a common approach consists in reducing the robot to a finite size shape, possibly according to a convex embedding shaped (e.g., a circle) that virtually surrounds the robot, fully accounting for its real physical spatial occupancy, this is shown in the figures (b) and (c)
- A larger the convex physical embedding (e.g., (b) can help to ensure safety during actual navigation: robot will be considered larger than it really is, such that planned paths will take it distant enough from obstacles
- In the C-space, this corresponds to use a point reduction but *blow* the obstacles by sliding the robots along their edges, creating a C-space which is geometrically a relatively complex manifold, as is its shown

C-space and point reduction

Special case: The robot is a *polygonal* one and can only *translate*



For *motion planning and navigation,* a mobile robot of any shape can be "reduced" to a reference point, as long as all reasonings are done in a C-space where the obstacles have been *inflated* to reflect real robot's spatial occupancy

Important concepts to take home so far

- Coordinate frames and Coordinate systems
- Robot pose, positions and orientations of robot's (multiple) bodies
- Relative poses, World and Local frames
- Rigid body assumption, Point abstraction
- Generalized coordinates and Configuration space (C-space), Phase space
- Holonomic constraints (geometric constraints in the configuration space)
- Mobile vs. Arm robots
- Single body vs. Multiple body robots
- Open chain vs. Closed chain robots
- Links and Joints
- Types of joints
- DOF - Dimension of the C-space
- DOFs of a joints
- Grubler's formula
- Workspace and Task space
- C-space for mobile robots and point reduction