Introduction & Recap

FIELD AND SERVICE ROBOTICS

UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II
DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE DELL'INFORMAZIONE

www.prisma.unina.it

Locomotion system

- Mobile robot
  - One or more rigid bodies equipped with a locomotion systems
  - The locomotion system classifies the mobile robots
- Wheeled mobile robots
  - Rigid body (base or chassis)
  - Wheels
  - (Trailers)
- Legged mobile robots
  - Multiple interconnected rigid bodies
  - Extremities (feet) periodically in contact with the ground
- Flying robots
  - Rigid body (frame)
  - Propellers
  - Fixed wings or multi-rotor (vertical take-off and landing)
- Underwater robots
  - As flying robots, but in the water

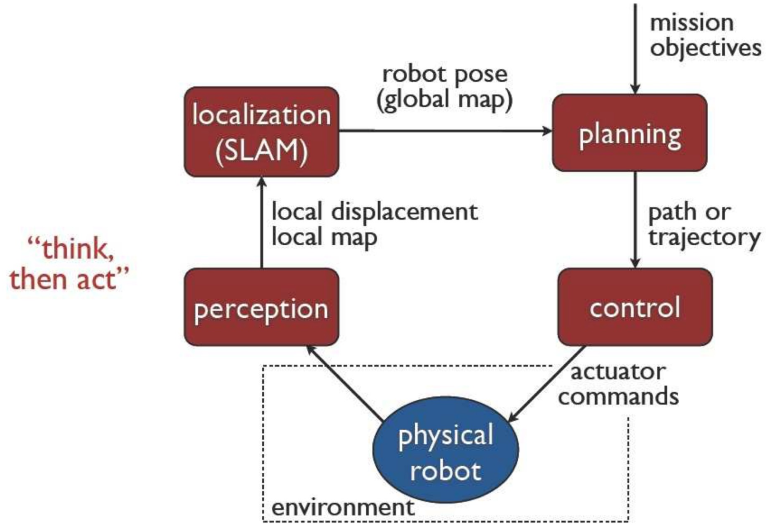
Main challenges - Localization

- Localization
  - Robots are not bolt
  - Encoders are still present (as in the wheels) but many unmodelled effects get in the way
  - Direct kinematics is a set of differential equations
  - Global positioning systems (GPS) can help
    - It cannot be used anywhere
    - Inaccurate for some tasks
- Odometry
  - Technique to estimate the robot position from on-board and off-board sensors
  - Simultaneous localization and mapping (SLAM)

Main challenges – Motion Planning & Control

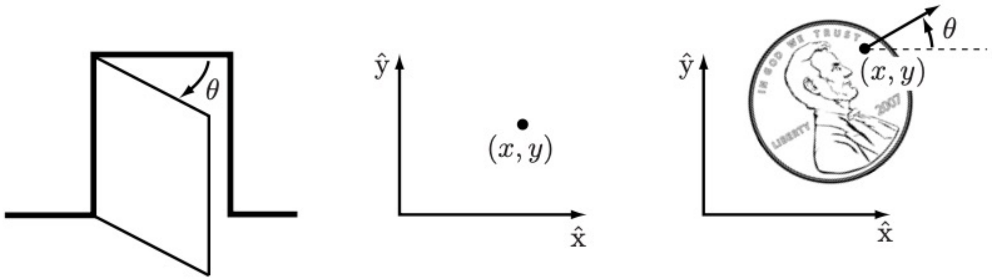
- Motion planning
  - Plan the path and the time law (trajectory) to reach the goal, taking into account:
    - Robot limitation
    - Obstacles
    - Prevented motion directions
- Motion Control
  - Control the robot motion along the planning path to track the desired time law, taking into account:
    - Robot dynamics

- Deliberative architecture - Think, then act
  - Perception
    - Propioceptive
      - Position, orientation, velocity
    - Exteroceptive
      - Obstacles position, other robots position, people, …
  - Sensors
    - Propioceptive
      - Encoders, inertia measurement units (IMU)
    - Exteroceptive
      - GPS, cameras, tactile sensors, proximity sensors, range finder, …
- Reactive architecture – Don't think, act
- Hybrid architecture – Think and act simultaneously
- Behavioural architecture – Think the action modality



Configuration space

- The robot's configuration is the specification of the positions of all points of the robots
  - Since we consider rigid elements only, only a few numbers are needed to represent its configuration



Courtesy from [1]

- The number of *degrees of freedom* (DoF) of a robot is the smallest number of real-valued coordinates needed to represent its configuration *[1]*
  - In the above example, the coin side (heads or tails) covers a discrete set and not a continuous range

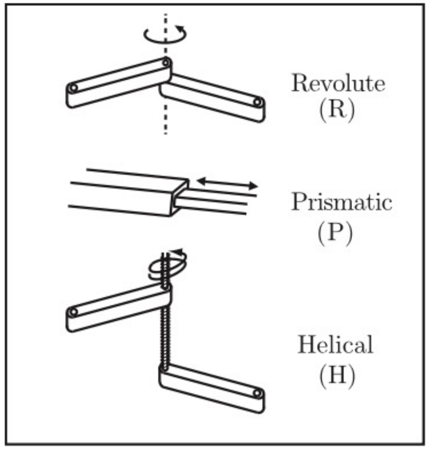
[1] K.M. Lynch, F.C. Park, "Modern Robotics. Mechanics, planning, and control," Cambridge University Press, 2017.

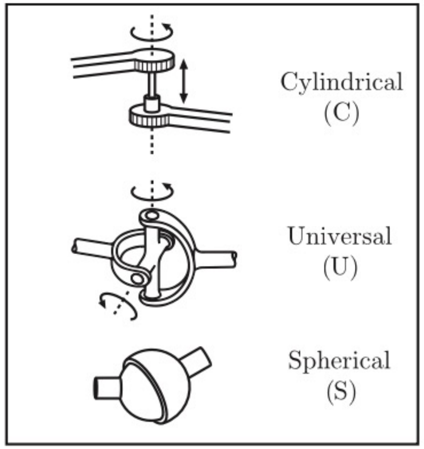
- Definition *[1]*
  - *The configuration of a robot is a complete specification of the position of every point of the robot. The minimum number n of real-valued coordinates needed to represent the configuration is the number of degrees of freedom of the robot. The n −dimensional space containing all possible configurations of the robot is called the* configuration space *($\mathcal{C}$ −space). The configuration of a robot is represented by a point in its $\mathcal{C}$ −space*

Degrees of freedom of a rigid body

- The number of DoFs depends on the constraints
  - Degrees of freedom = sum of freedoms of the bodies – number of independent constraints
  - The same rigid body can have different DoFs if the constraints change during the motion or the task or the time
- A rigid body moving in the 3D space, called spatial rigid body, has six DoFs
- A rigid body moving in the 2D space, called planar rigid body, has three DoFs

- A robot manipulator has links and joints
  - Joints





Joint type	dof f
Revolute (R)	1
Prismatic (P)	1
Helical (H)	1
Cylindrical (C)	2
Universal (U)	2
Spherical (S)	3

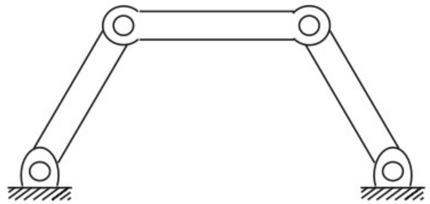
- Grübler's formula *[1]*
  - *Consider a mechanism consisting of N links. Let J be the number of joints, m the number of DoFs of a rigid body ($m = 3$ for planar mechanisms and $m = 6$ for spatial mechanisms), f_i be the number of freedoms provided by the i −th joint. Then Grübler's formula for the number of DoFs of the robot is*

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

  - *This formula holds only if all joint constraints are independent. If they are not independent then the formula provides a lower bound on the number of DoFs*

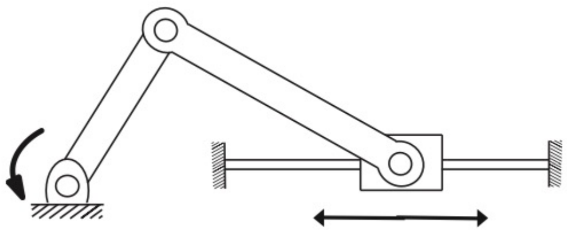
$\text{DoFs} = m(N-1-J) + \sum_{i=1}^{J} f_i$

- Four-bar linkage



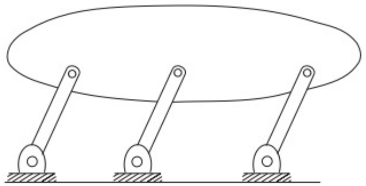
  - DoFs = $3(4-1-4)+4=1$

- Slider-crank



  - DoFs = $3(4-1-4)+4=1$

- Parallelogram linkage



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

  - DoFs = $3(5-1-6)+6=0$
  - A mechanism with 0 DoFs is by definition a rigid structure. However, it is clear that the mechanism has 1 DoF. Indeed, any one of the three parallel links, with its two joints, has not effect on the motion of the mechanism. In other words, the constraints provided by the joints are not independent as instead required by Grübler's formula.

- Aerial manipulator

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

- Six propellers plus two 5-DoFs arms
  - Body fixed on the ground
    - 6 rotating propellers (1 link plus 1 rotational joint each)
    - 5 rotational joints for each arm
    - 1 ground
    - DoFs = $6(6+5+5+1-1-6-5-5)+6+5+5=16$
  - Flying body
    - 6 DoFs given by the spatial rigid body
    - DoFs $=16+6=22$

Examples: Grübler's formula

- Mobile manipulator

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

- KUKA youBot with 4 mecanum-wheels (omnidirecitonal base) plus 5-DoF robotic arm
  - 4 wheels (1 link plus 1 rotational joint each)
  - A planar rigid body (3 DoFs)
  - 5 rotational joints for the arm
  - 12 DoFs

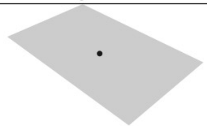
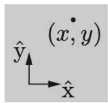
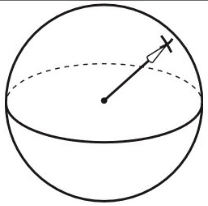
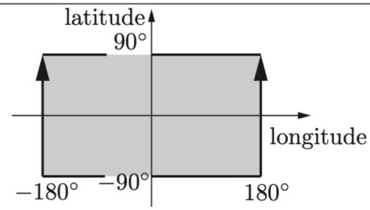
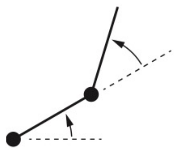
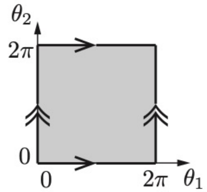
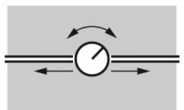
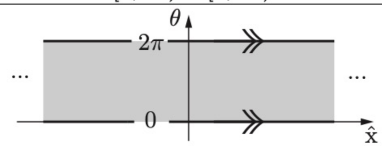
Configuration space topology

- Until now we saw the dimension of the robot's configuration space, that is the number of DoFs
- The shape of the $\mathcal{C}$ −space is also important
  - Two $\mathcal{C}$ −spaces may have the same number of DoFs but they can differ in shape
- Two $\mathcal{C}$ −spaces are topologically equivalent if one can be continuously deformed to the other without cutting or gluing
  - A sphere can be deformed into a football simply by stretching. These spaces are topologically equivalent.
  - A sphere is not topologically equivalent to a plane because you have to cut the sphere
- Topologically distinct 1D spaces are
  - Circle, S^1
  - Line, $\mathbb{R}^1$
    - It is an1D Euclidian space
  - Closed interval of the line, $I = [a, b] \subset \mathbb{R}^1$
- Angular coordinates
  - Circle, S^1
  - Sphere, S^2
    - S^2 is not topologically equivalent to $S^1 \times S^1$

Configuration space topology

- The topology of a space is a fundamental property of the space itself and it is independent of how we choose coordinates to represent points in the space
  - A unit circle can be represented by just the angle, θ, on the circumferences, or by the Cartesian coordinates, (x, y), subject to the constraint $x^2 + y^2 = 1$. No matter what the choice of coordinates is, the space itself does not change
- Some $\mathcal{C}-$spaces are expressed as the Cartesian product of two or more spaces of lower dimension
  - Points in these $\mathcal{C}-$spaces can be represented as the union of the representations of points in the lower-dimensional spaces

Configuration space topology

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

Courtesy from [1]

Configuration space representation

- To perform computations, we must have a numerical representation of the $\mathcal{C}$ −space, consisting of a set of real numbers
  - This implies a choice and it it is not as fundamental as the topology of the space, which is independent of the representation
- A choice of n coordinates, or parameters, to represent a n −dimensional space is called an explicit parameterization of the space
  - This holds unless the representation has one or more singularities
    - The latitude-longitude representation of a sphere is unsatisfactory close to the poles. The poles are singularities of the representation since a small steps can result in a large change in coordinates (there is a jump from 180 to -180 degrees, for instance). The location of these singularities has nothing to do with the sphere itself, which looks the same everywhere.
  - If you can assume that the configuration never approaches a singularity of the representation, you can ignore this issue, otherwise there are the following ways to overcome the problem

Configuration space representation

- ## Coordinate chart
  - You can use one coordinate chart on the space, where each coordinate chart is an explicit parameterization covering only a portion of the space such that, within each chart, here is no singularity
  - If we define a set of singularity-free coordinate charts that overlap each other and cover the entire space, the charts are said to form an atlas of the space
    - An atlas of the planet Earth consists of several maps that together cover the Earth
  - A disadvantage of this approach is the need in switching between charts
- ## Implicit representation
  - An implicit representation views the n −dimensional space as embedded in a Euclidean space of more than n dimensions
  - An implicit representation uses the coordinates of the higher-dimensional space, but subjects these coordinates to constraints that reduce the number of DoFs (see the circle example some slides before)
  - A disadvantage of this approach is that the representation has more numbers than the number of DoFs
  - An advantage is that there are no singularities in the representation
- The non-Euclidean shape of many C −spaces motivates our use of implicit representations

Configuration space topology and representation: Examples

- Aerial manipulator

- Six propellers plus two 5-DoFs arms
  - Body fixed on the ground
    - 6 rotating propellers: $S^1 \times \cdots \times S^1 = T^6$ (6 times)
    - 5 rotational joints for each arm: $S^1 \times \cdots \times S^1 = T^5$ (5 times for each arm without joint limits)
      - 5 rotational joints for each arm: $I^1 \times \cdots \times I^1 = I^5$ (5 times for each arm with joint limits)
    - Topology $T^6 \times T^5 \times T^5 = T^{16}$ (or $T^6 \times I^5 \times I^5 = T^6 \times I^{10}$)
  - Flying body
    - Spatial rigid body $\mathbb{R}^3 \times S^2 \times S^1$
    - Topology $T^{16} \times \mathbb{R}^3 \times S^2 \times S^1$ (or $T^6 \times I^{10} \times \mathbb{R}^3 \times S^2 \times S^1$)

Configuration space topology and representation: Examples

- Mobile manipulator

- KUKA youBot with 4 mecanum-wheels (omnidirecitonal base) plus 5-DoF robotic arm
  - 4 wheels: $S^1 \times \cdots \times S^1 = T^4$ (4 times)
  - Planar rigid body: $\mathbb{R}^2 \times S^1$
  - 5 rotational joints for the arm: $S^1 \times \cdots \times S^1 = T^5$ (5 times without joint limits)
    - 5 rotational joints for the arm: $I^1 \times \cdots \times I^1 = I^5$ (5 times with joint limits)
  - Topology: $\mathbb{R}^2 \times S^1 \times T^9$ (or $\mathbb{R}^2 \times S^1 \times T^4 \times I^5$)

Task space and workspace

- The task space is a space in which the robot's task can be naturally expressed
  - The task space to write on a piece of paper would be $\mathbb{R}^2$
  - If the task is to manipulate a rigid body, a natural representation of the task space is the one represented by the position and the orientation of the frame attached to the robot's end-effector (operational space)
  - The decision of how to define the task space is driven by the task and it is independent from the robot
- The workspace is a specification of the configurations that the end-effector of the robot (or the robot itself if it is moving) can reach
  - The definition of the workspace is primarily driven by the robot's structure, independently of the task
- The task space and the workspace differs from the C $-$space
  - A point in the task space or the workspace may be achievable by more than one robot configuration, meaning the the point is not a full specification of the robot's configuration
    - In a 7-DoFs robots, the six-DoFs pose (position + orientation) of its end-effector does not fully specify the robot's configuration
  - By definition, all points in the workspace are reachable by at least one configuration of the robot

Underactuation

- Hamilton's principle
  - *The motion of an arbitrary mechanical system occurs in such a way that the definite integral*

$$A = \int_{t_i}^{t_f} L(q, \dot{q})\, dt$$

becomes stationary for arbitrary possible variations of the configuration of the system, provided the initial and final configurations of the system are prescribed (Lanczos, 1949).

    - $L(q, \dot{q})$ is a scalar function known as Lagrangian
    - $q \in \mathbb{R}^n$ is the vector of generalised coordinates
    - $\dot{q} \in \mathbb{R}^n$ is the vector of generalised velocities
  - The "action" (the previous integral) is stationary if the following so-called Euler-Lagrange equation is satisfied

$$\frac{d}{dt}\left(\frac{\partial L(q, \dot{q})}{\partial \dot{q}}\right) - \frac{\partial L(q, \dot{q})}{\partial q} = f$$

    - $f \in \mathbb{R}^n$ is the vector of generalised forces
  - The Euler-Lagrange equation describes the dynamics of mechanical systems with n degrees of freedom (DoFs)

Control-affine equations of motion

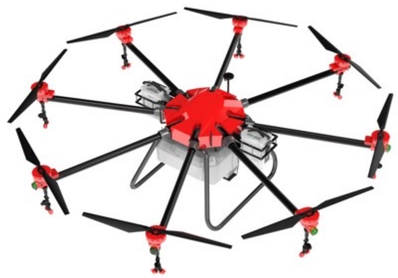
- The Euler-Lagrange equation $\frac{d}{dt}\left(\frac{\partial L(q,\dot{q})}{\partial \dot{q}}\right) - \frac{\partial L(q,\dot{q})}{\partial q} = f$ can be expressed as
$$M(q)\ddot{q} + h(q,\dot{q}) = f$$
  - $M(q) \in \mathbb{R}^{n\times n}$ is the symmetric and positive definite inertia (mass) matrix
  - $h(q,\dot{q}) \in \mathbb{R}^{n}$ is the lumped vector with gravitational, centripetal, and Coriolis terms
- The mathematical model can also be written as
$$\ddot{q} = -M(q)^{-1}h(q,\dot{q}) + M(q)^{-1}f$$
- The vector of generalised forces can be different from the real control inputs to the system
  - Ball and plate example
    - The generalised forces act on the rolling ball, while the real control inputs tilt the plate
  - The mapping is $f = B(q)\tau$
    - $B(q) \in \mathbb{R}^{n\times m}$ the input matrix, $\tau \in \mathbb{R}^{m}$ the real control input vector
- Therefore
$$\ddot{q} = b(q,\dot{q}) + G(q,\dot{q})\tau$$
    - $G(q,\dot{q}) = M(q)^{-1}B(q)$with, with
  - The general form can be written as
$$\ddot{q} = f(q,\dot{q},\tau,t)$$

- Underactuation definition *[3]*
  - *A second-order differential equation* $\ddot{q} = f(q, \dot{q}, \tau, t)$ *is fully actuated in the state* $x = (q, \dot{q})$ *and the time* t *if the resulting map* f *is surjective: for every* $\ddot{q}$ *there exists a* u *which produces the desired response. Otherwise it is underactuated (in* x *at time* t*)*
  - As the dynamics for many robots is in the control-affine form $\ddot{q} = b(q, \dot{q}) + G(q, \dot{q})\tau$, this implies that a control-affine system is underactuated if

$$rank\big(G(q, \dot{q})\big) < \dim[q]$$

[3] R. Tedrake, "Underactuated robotics: Algorithms for walking, running, swimming, flying, and manipulation," Course Notes for MIT 6.832, downloaded in January 2022

- Notice that one of the most common cases for underactuation is $\dim[\tau] < \dim[q]$, which trivially implies that $G(q, \dot{q})$ is not full rank
  - However, this does not implies that if you have more actuators than DoFs then the system is fully actuated (!)



  - Consider the octocopter above in a flat configuration
    - While flying, the system as 6 DoFs (see slides above)
    - The system has 8 actuators
    - We will see that this configuration can apply only three rotations and one translation along the vertical direction of the body frame (i.e., $\mathrm{rank}[G(q, \dot{q})] = 4$), therefore this system with more actuators than DoFs is underactuated (!)

- Notice that whether or not a system is underactuated may depend on the current state, constraints, and time
  - Usually, a system is underactuated if it is underactuated in all states and times
- However, notice that the given underactuation definition for a system as it is gives as a result that most of the system are underactuated
  - A quadrotor is underactuated because it has four control inputs but it has ten DoFs
  - A wheeled robot with four independent actuators for each of the wheels is underactuated because the robot has 7 DoFs (three given by the planar rigid body chassis and four given by the wheels)
  - The second-order linear system $\ddot{x} = \tau, |\tau| \leq 1$ is underactuated (!) because there is no τ which can produce the acceleration, for instance, $\ddot{x} = 2$
  - …

- Therefore, often, we will refer to underactuation as a property of the mathematical model used to model our robotic system
  - Imagine a two-link robot with two actuators. A typical model with rigid links could be fully actuated, but if we add extra DoFs to model small amounts of flexibility in the links then that system model could be underactuated. These two models describe the same robot, but at different levels of fidelity. Two actuators might be enough to completely control the joint angles, but not the joint angles and the flexing modes of the link.
- We will then use a subset of the DoFs to describe our model, often expressed in the task space
  - These variables will be often indicated with $q \in \mathbb{R}^n$ as well, with n the dimension of this space and not the DoFs
    - For instance, in the wheeled robots we will not consider the wheel angles or in the drones we will not consider the propeller angles
- Even with this position, we see that most of robots still remains underactuated
  - A quadrotor is underactuated because it has four control inputs but it has six variables to control in the Cartesian space
  - A car is underactuated because it has two control inputs (steering and forward/backward speed) but at least three DoFs given by the planar rigid body chassis
    - Instead, the model of a wheeled robot with four independent actuators for each of the wheels is fully-actuated now, because the robot has at least three DoFs given by the planar rigid body chassis
  - The human body has an incredible number of actuators (muscles), and in many cases has multiple muscles per joint; despite having more actuators than position variables, when we jump through the air, there is no combination of muscle inputs that can change the ballistic trajectory of the center of mass (barring aerodynamic effects)