Introduction to Robotics

DI Matthias Weyrer, BSc

What defines a Robot?

- **No uniform definition** of the term robot
- A few examples:

  - **Robotics Institute of America (RIA):**
    A robot is a reprogrammable, multifunctional manipulator designed to move material, parts, tools, or specialized devices, through variable programmed motions for the performance of a variety of tasks.

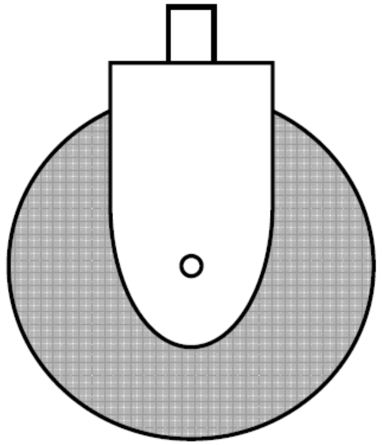
  - **Definition according to VDI (Richtlinie 2860):**
    Industrieroboter sind universell einsetzbare Bewegungsautomaten mit mehreren Achsen, deren Bewegungen hinsichtlich Bewegungsfolge und Wegen bzw. Winkeln frei (d.h. ohne mechanischen Eingriff) programmierbar und gegebenenfalls sensorgeführt sind. Sie sind mit Greifern, Werkzeugen oder anderen Fertigungsmitteln ausrüstbar und können Handhabungs- und/oder Fertigungsaufgaben ausführen.

  - …

Robot Categories

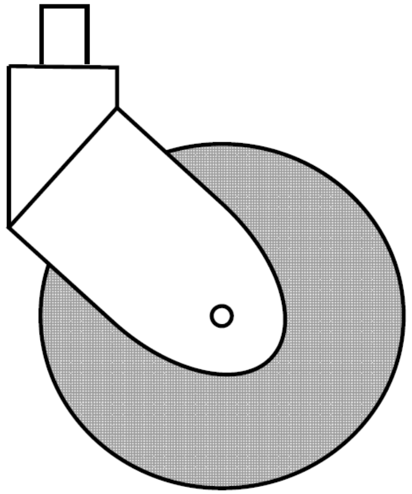
- Mobile Robots
  - Mobile Manipulators
- Service Robots
- Industrial Robots
  - Serial Manipulators
  - Parallel Manipulators
  - Collaborative Robots
- Humanoid Robots
  - Animal-like Robots

Basic Wheel Concepts

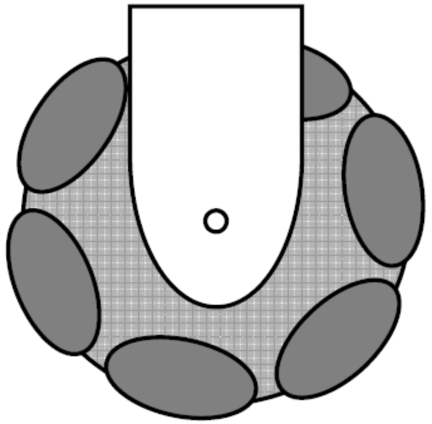


- **Standard wheel**
  - Steered or passive
  - Driven or not driven

- **Swivel wheel**
  - Usually no steering or powering task
  - Known from shopping carts

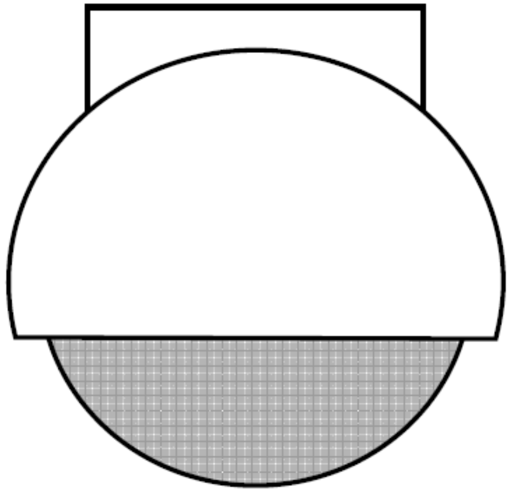


Basic Wheel Concepts

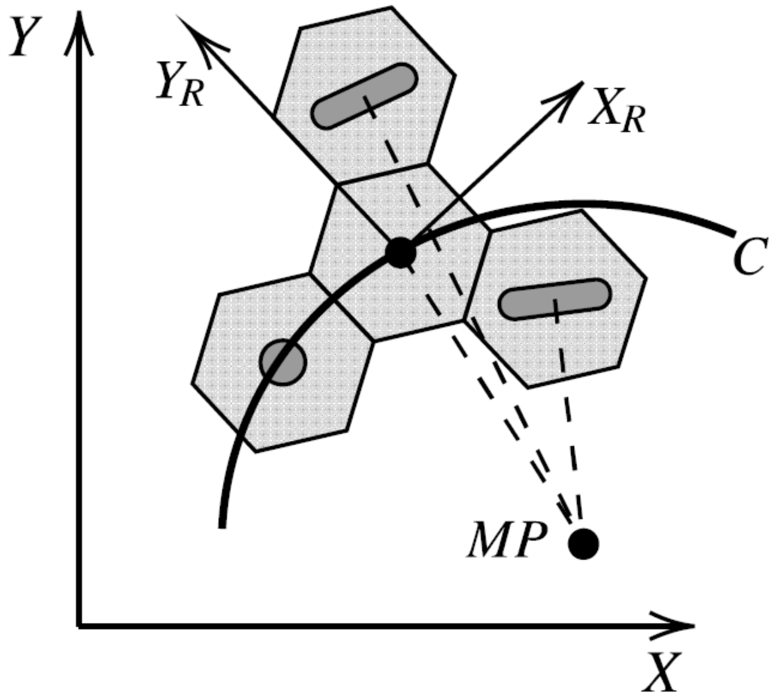


- **Omnidirectional wheel**
  - Free moving rolls around wheel circumference
  - Can move in all directions without slip

- **Ball wheel**
  - Can be actively powered/driven in all directions



Chassis Kinematics



- **Orientation and rotation speed of the wheels need to match**
  - To avoid slip of wheels
  - Depends on chassis design and geometry
  - Typical approach:

1. Define target position
2. Create planned path and velocity profile
3. Calculate parameters for each wheel in real-time
4. Deviations of planned path are being monitored and adjusted

AGV vs. Mobile Robots

- AGV = Automated Guided Vehicles
  - Use a fleet management system for navigation
  - E.g. wires in the floor, colored lines, etc.
  - Less flexible, requires adapting the infrastructure in case of a change
- Mobile Robots
  - No specialized infrastructure in its surrounding
  - Sensors and path planning are on-board
  - Easy remapping and learning of new targets

Service Robots

- The International Organization for Standardization defines a “service robot” as a robot “that performs useful tasks for humans or equipment excluding industrial automation applications”. (ISO 8373)

- Key definition is its direct purpose for humans

Service Robots

- Lawn trimming robot
- Vacuum robot
- Dairy robot

Service Robots

- Care robots

  - Care-o-Bot 3
  - Delivery tasks
  - Entertainment and communication
  - Support in case of emergencies
  - Robot assisted cleaning

Industrial robots

- **Serial manipulators**
- Defined by an open kinematic chain: Each arm segment is exactly connected to the following arm segment. The last segment is connected to an effector, where the kinematic chain ends. There is only a single connection to the base.

- **Parallel manipulators**
- Defined by a closed kinematic chain without an end. The effector can be reached by the base through various ways. There are more than one connections to the base.

Serial Manipulators

- Stäubli TX2-140
- Max. payload: 40 kg
- Operating range: 1510 mm
- Repeating precision: ± 0,05 mm

- KUKA KR 500 FORTEC
- Payload: 340-500 kg
- Operating range: 2485 - 3326 mm
- Repeating precision : ± 0,08 mm

Serial Manipulators

- KUKA KR10
- Scara Robot
- Number of axes: 4

Parallel Manipulators

- Fanuc M-2**i**A/3S
- Delta design
- Payload: 3 kg
- Operating reach: 800 mm
- Number of axes: 4

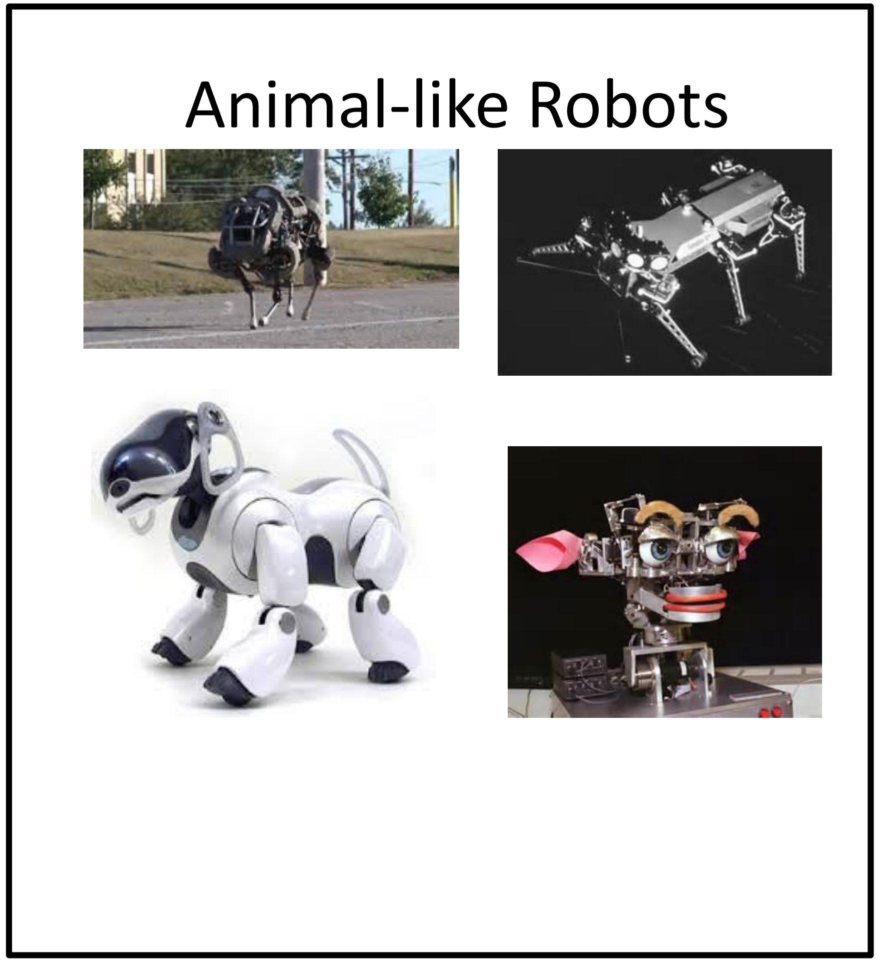
Collaborative Manipulators

- Commonly known by the term “cobot”
- Basic concepts:
  - Can collaborate with humans without the need of being physically separated by safety measures
  - Permanent measurement of torque and/or power supply in the robot‘s joints allow estimation of external force or influence
  - Maximum values can be defined in the robot controller so that it stops when surpassed
  - → Robot stops upon contact with an obstacle
  - → In reality the peak values for torque or pressure upon collision are not always met – safety evaluation is necessary!

Humanoid Robots

https://www.youtube.com/watch?v=uhND7Mvp3f4

Animal-like Robots



https://www.bostondynamics.com/spot

Overview of Gripper Technologies

- Typical application in robotics: Moving an object from position A to position B
- Robot needs to establish a connection to the object → gripper tool
- Type of object defines gripping strategy
  - Material
  - Shape
  - Weight
  - …

Overview of Gripper Technologies

- Most common gripper type: two finger gripper

Quelle: https://www.festo.com/cms/de-at_at/67728.htm

Quelle: https://onrobot.com/de/produkte/greifer-rg2

Gripper Categories

- Mechanical gripper
  - Passive mechanical gripper
- Pneumatic gripper
- Magnetic gripper
- Adhesive gripper

- Effect:
  - **Force pairing**
    Gripper applies pressure onto object
  - **Shape pairing**
    Gripper and object are perfectly adjusted to one another and needs less pressure to lift the object
  - **Material pairing**
    Harvesting adhesion forces between gripper and object

Mechanical Grippers

- Plier or hand-like design
- Object is lifted by applying pressure
- Driven pneumatically or electrically
- Large variety in available designs:
  - 4 different types of gripping approach, e.g.:
    - Tweezer grip
    - Scissor grip
    - Spread grip
    - Base grip

Mechanical Grippers

Quelle: https://www.plugandautomate.swiss/schunk-eoa-ur3510-jgp-100-1.html

Quelle: https://www.plugandautomate.swiss/zimmer-2-backen-parallelgreifer-hrc-05.html

Passive mechanical Grippers

Pneumatic Grippers

- Vacuum or suction cups
- Aspiration through compressed air
- Pinching through compressed air
- Large fields of application, such as:
  - Manipulating glass, CDs or any type of polished surface
  - Packaging industry: Non-rigid objects

Pneumatic Grippers

Quelle: https://www.plugandautomate.swiss/on-robot-vg10-vakuumgreifer.html

Quelle: https://www.plugandautomate.swiss/schmalz-fxcb-flaechengreifer.html

Magnetic Grippers

- Object is being attached and held magnetically
- Two main concepts:
  - Permanent magnet
    - Magnet can't be deactivated → Additional releasing mechanism necessary
    - No power supply necessary
  - Electromagnetic
    - Can be switched off / released

Magnetic Grippers

Quelle: https://www.plugandautomate.swiss/mg10-magnetgreifer.html

Adhesive Grippers

- Adhesion force:
  Molecular force at contact surface

- Harness the molecular Van-der-Waals force
- Residue-free gripping
- Sensitive gripping without external mechanical force
- No power supply needed

Adhesive Grippers

Quelle: https://www.plugandautomate.swiss/adheso-haftgreifer.html

External Sensors

- Laser scanners

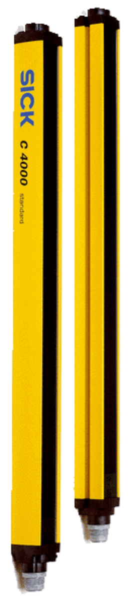
Quelle: https://www.sick.com/at/de/optoelektronische-schutzeinrichtungen/sicherheits-laserscanner/s3000-standard/c/g187231

- Light barrier

Quelle: https://www.sick.com/de/de/lichttaster-und-lichtschranken/lichttaster-und-lichtschranken/g6/c/g175965

Externe Sensorik

- Light curtain



Quelle: https://www.sick.com/de/de/optoelektronische-schutzeinrichtungen/sicherheits-lichtvorhaenge/c4000-standard/c/g202252

- Force-Torque Sensor

Quelle: https://onrobot.com/de/produkte/hex-6-achsiger-kraft-drehmoment-sensor

External Sensors

- Vision systems
  - Different Sensor Technologies
    - 2D-Cameras
    - RGB-Cameras
    - RGB-D Cameras
  - Various fields of application
    - Quality assurance
    - Calibration
    - Identification of object orientation on a conveyor belt
    - …

External Sensors

- Switches / Contact switches
- Proximity Sensors
  - Capacitive
  - Inductive
  - Laser
  - Radar
  - Ultrasound
  - Photoelectric sensors
  - Hall-Effect sensors
- Barcode Scanner
- Gyroscope/IMUs

Discussion

JOANNEUM RESEARCH ROBOTICS

Kinematics

DI Matthias Weyrer, BSc

What is Kinematics?

- Kinematics is a subfield of physics, developed in classical mechanics, that describes the motion of points, bodies (objects), and systems of bodies (groups of objects)
- without considering the forces that cause them to move.
  - This is done within the dynamics

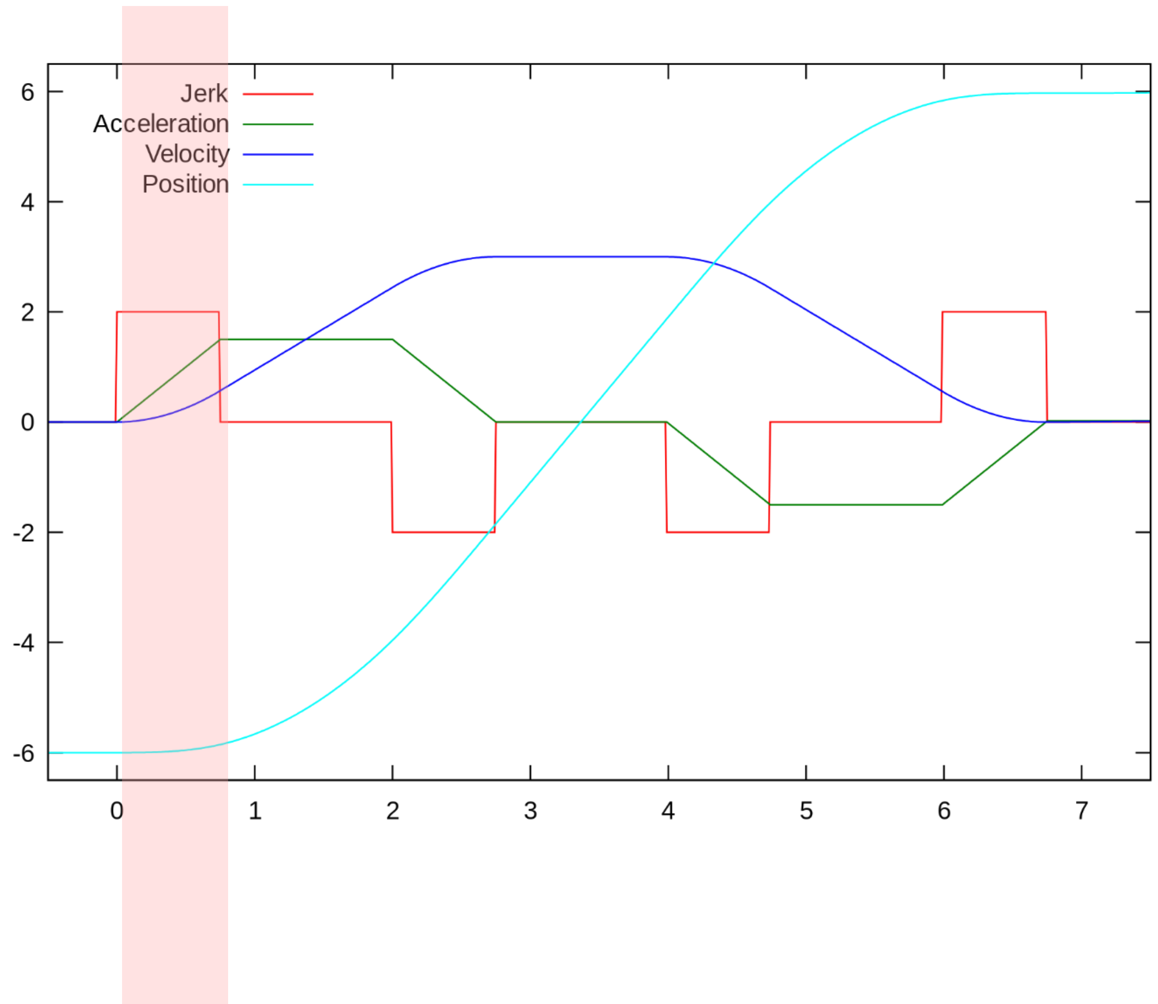
Important terms in the field of Kinematics

- Position
  - Distance: Meter (m)
  - Angle: Radiant (rad)
- Velocity
  - Rate of change of the position
  - SI-Unit: „Meter per Second“ (m / s) or „Radiant per Second“ (rad / s)
- Acceleration
  - Rate of change of the velocity
  - SI-Unit: „Meter per second squared“ (m / s^2) or „Radiant per second squared (rad / s^2)
- Jerk (german: Ruck)
  - Rate of change of the Acceleration
  - SI-Unit: „Meter per second to the power of 3“ (m / s^3) or „Radiant per second to the power of 3“ (rad / s^3)

Motion Planning

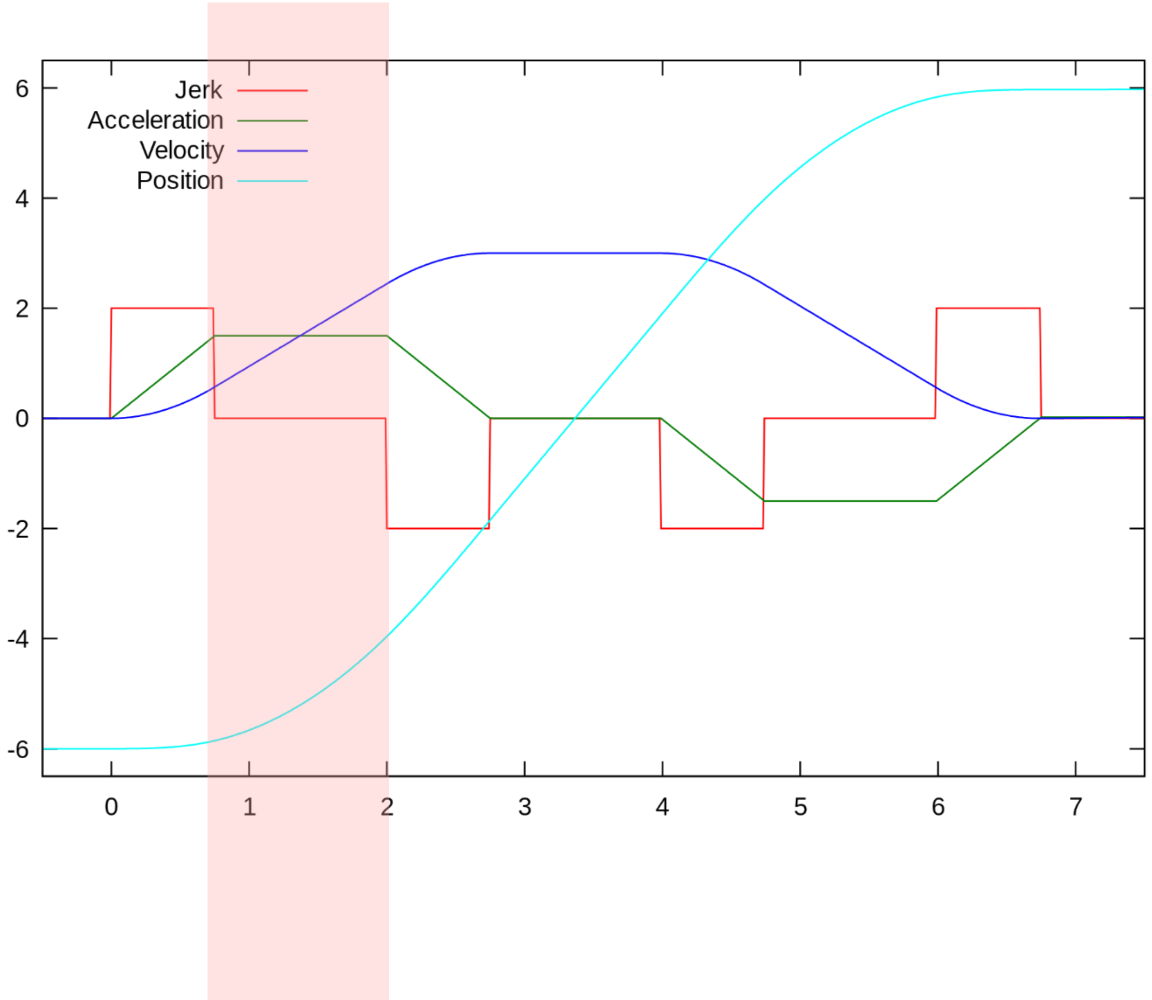
- Path vs. Trajectory
  - Path:
    - Describes the positions to the target
    - No time information
  - Trajektorie:
    - Includes position and time
- With pathplanning alone, we do not know when the robot is at a particular position
- Planning the path→ add time information → trajectory

Trajectory Planning



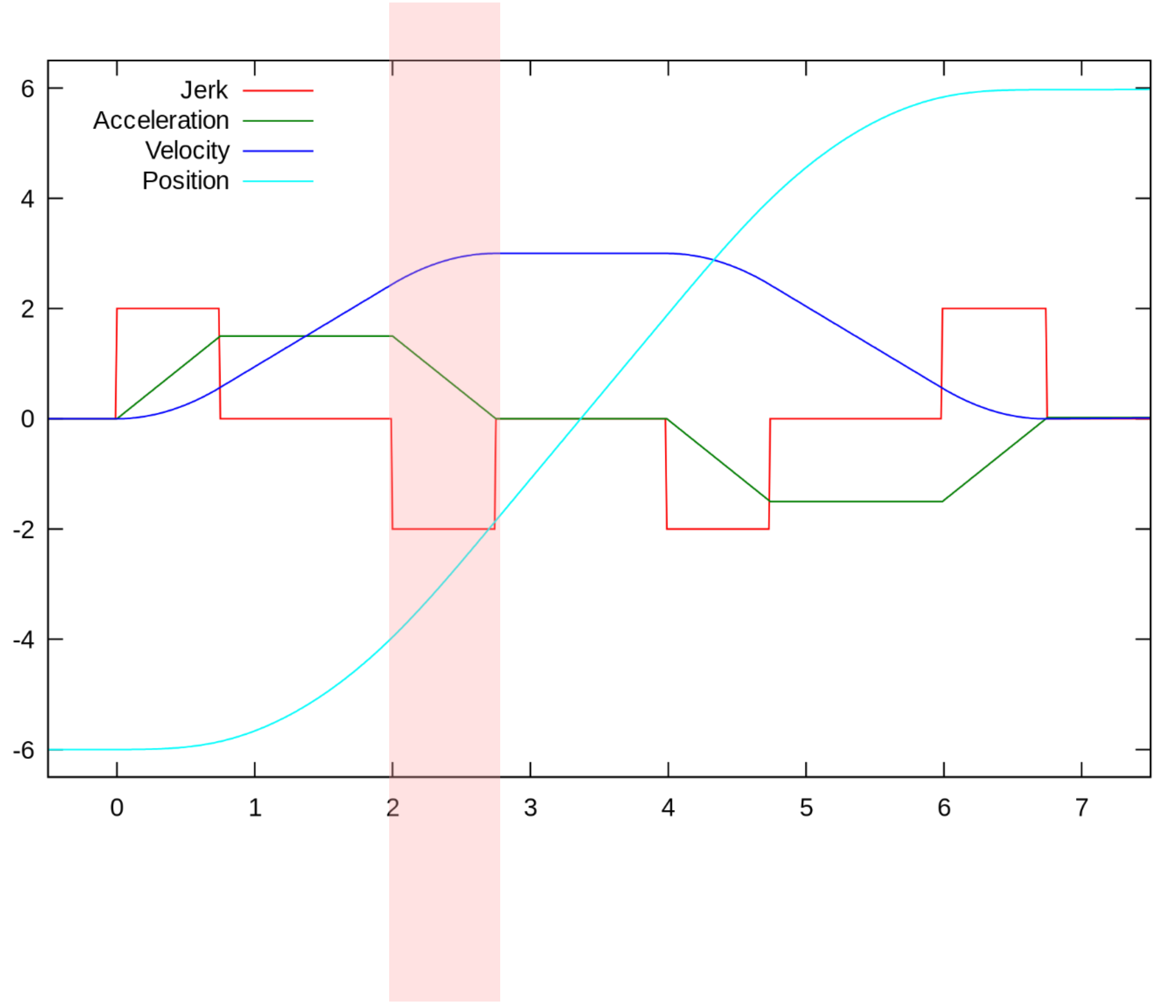
- Phase 1:
  - Jerk is at a constant value
  - Acceleration increases linearly
  - Velocity increases qudratically
  - Position increases cubicly

Trajectory Planning



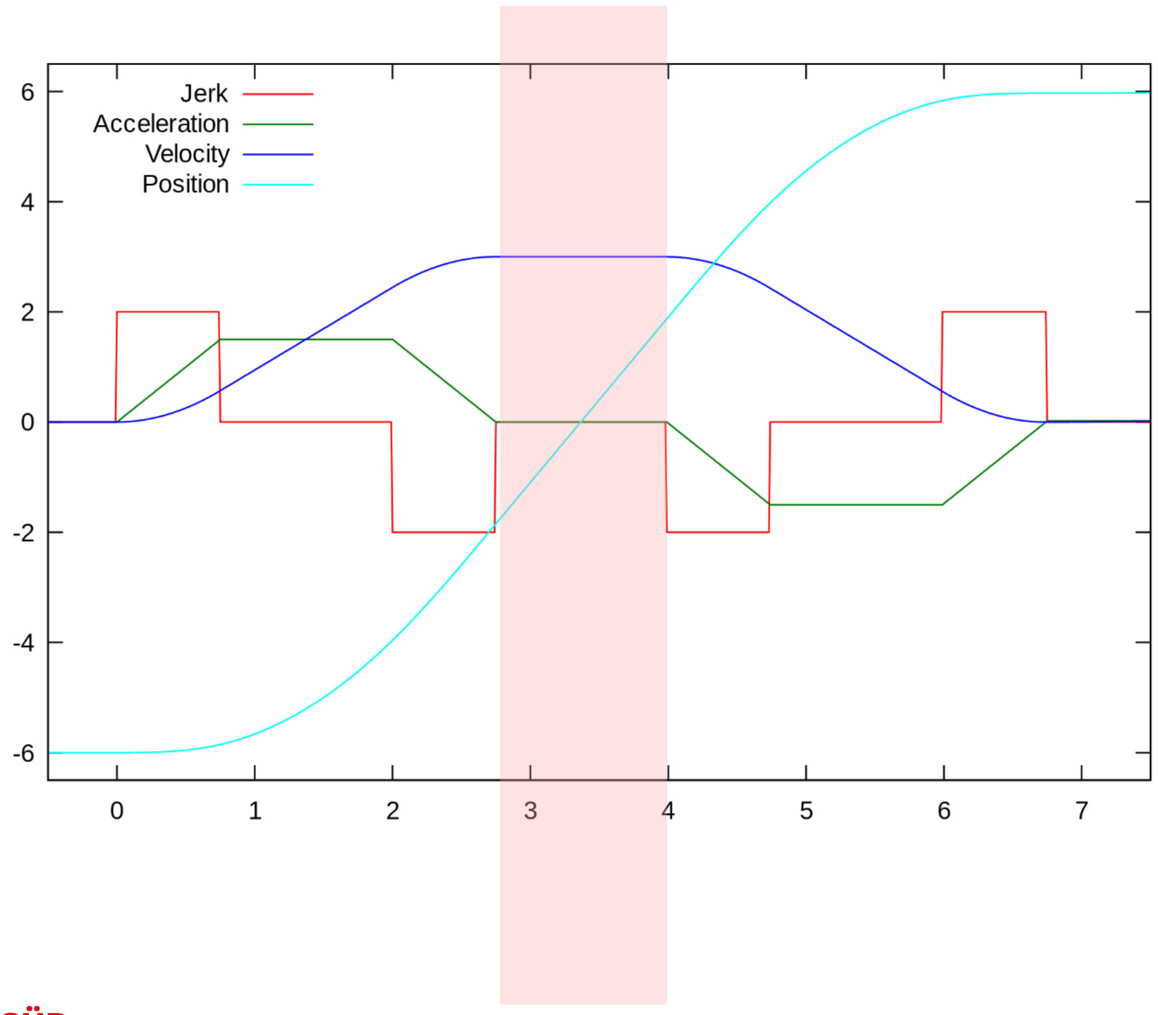
- Phase 2:
  - Jerk is zero
  - Acceleration is constant
  - Velocity increases linearly
  - Position changes quadratically

Trajectory Planning



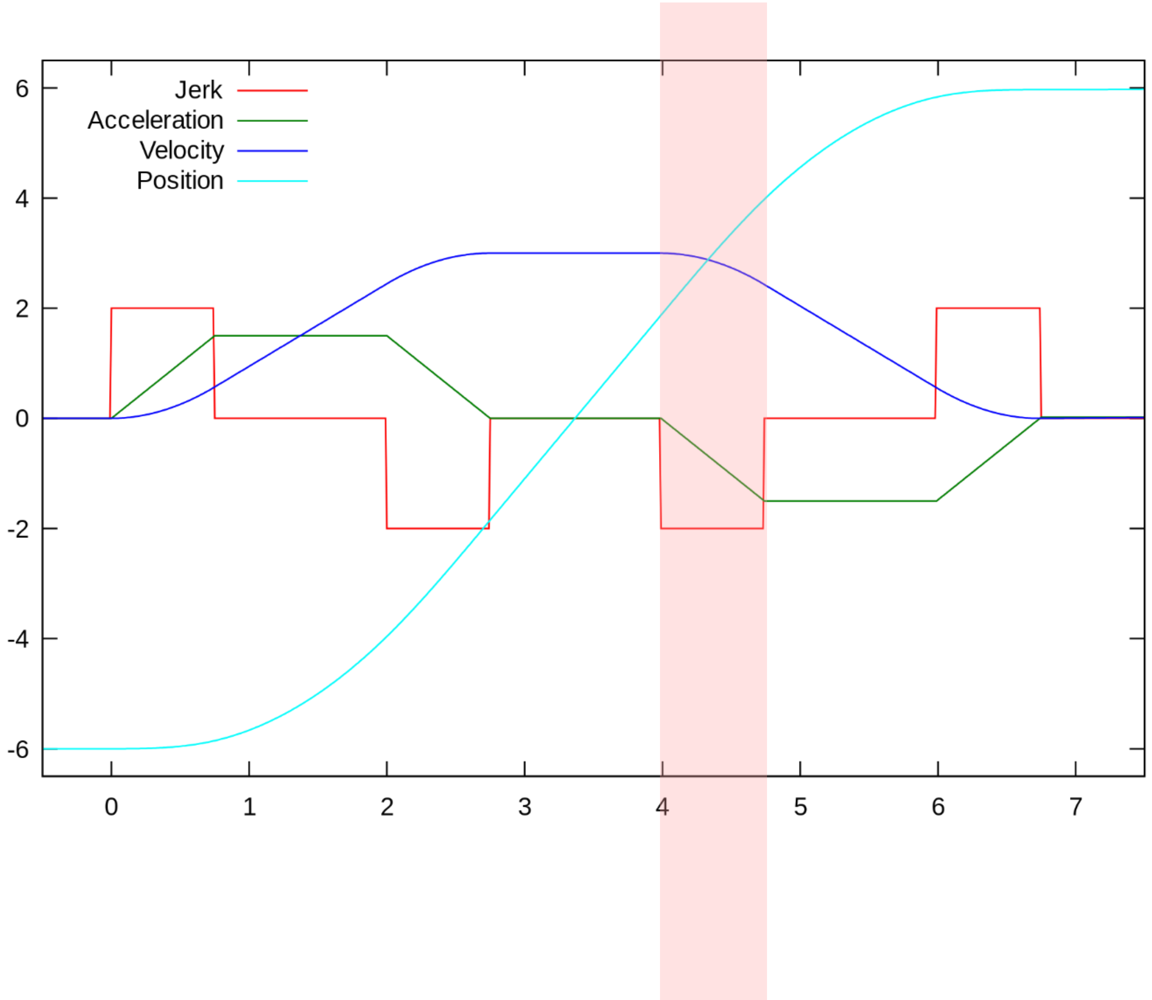
- Phase 3:
  - Jerk constant (oppsite sign)
  - Acceleration decreases linearly
  - Velocity increases slower and slower

Trajectory Planning



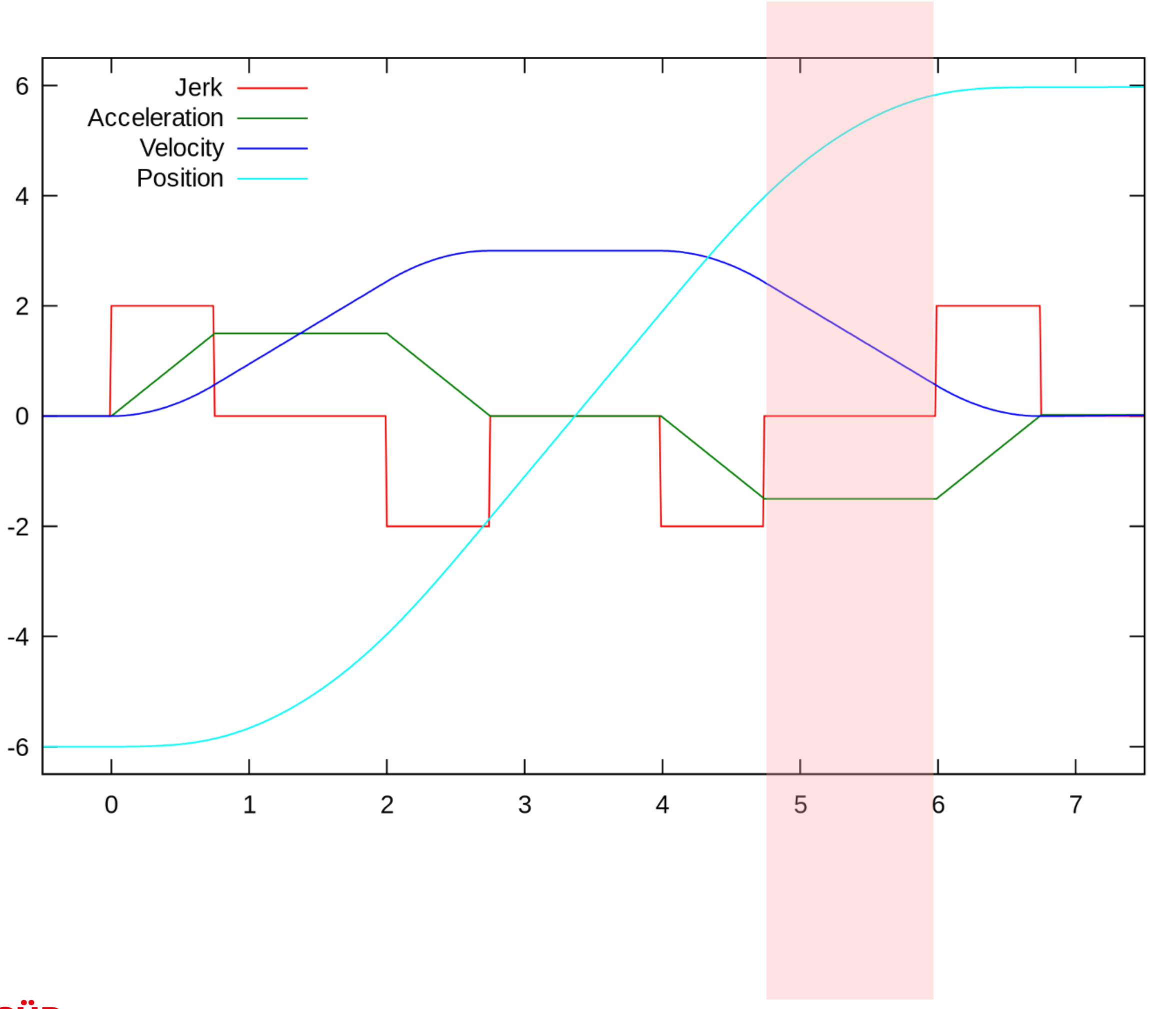
- Phase 4:
  - Jerk and acceleration at 0
  - Velocity is constant
  - Position increases linearly

Trajectory Planning



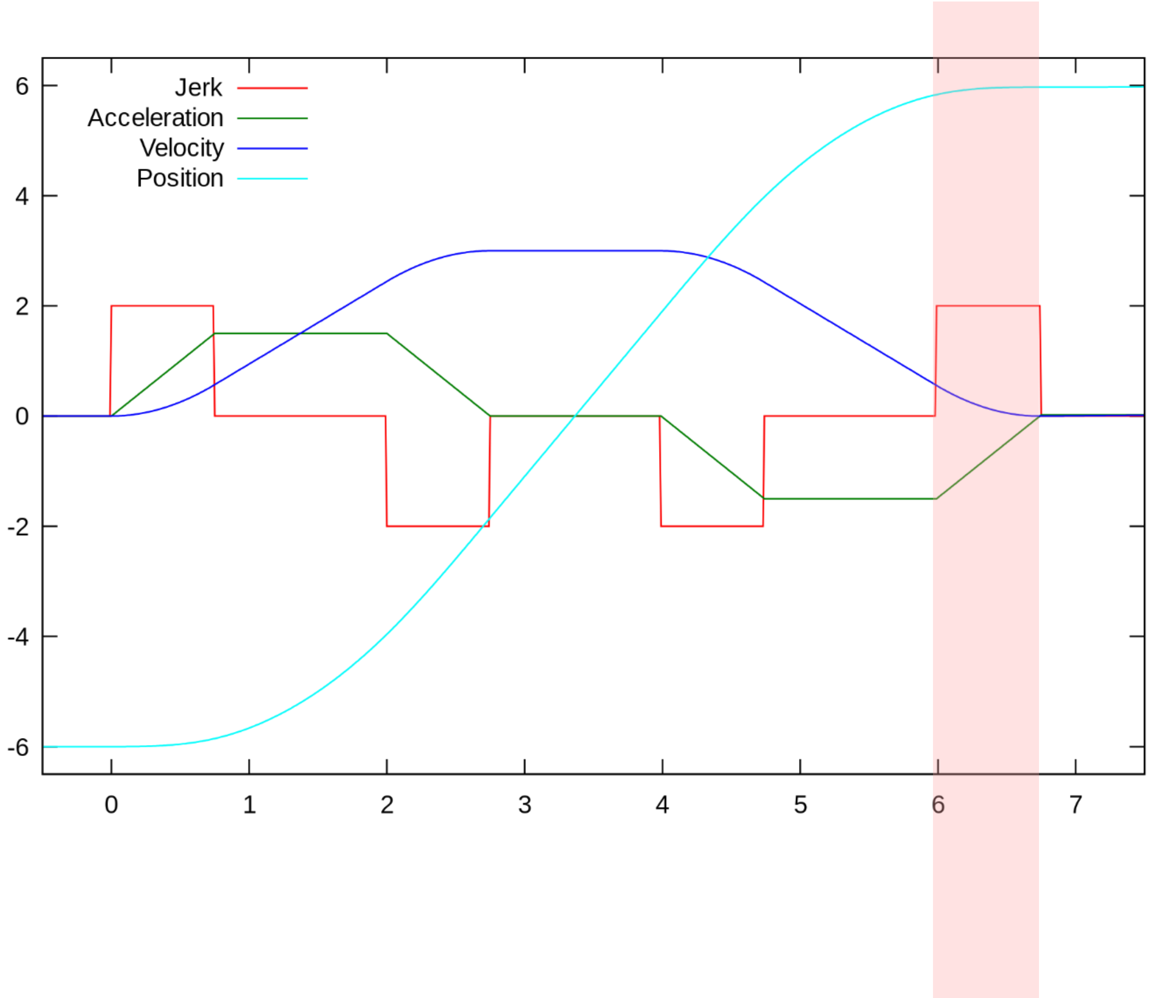
- Phase 5:
  - Jerk constant (<0)
  - Acceleration becomes negative
  - Velocity decreases quadratically

Trajectory Planning



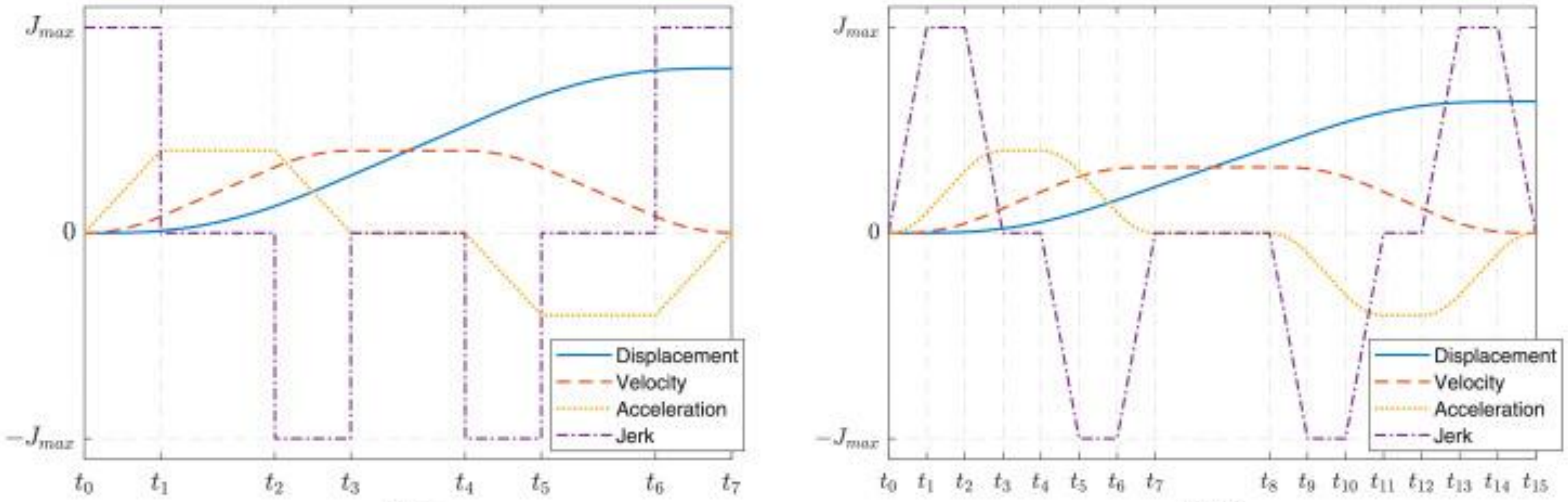
- Phase 6:
  - Jerk at zero
  - Acceleration constant < 0
  - Velocity decreases linearly

Trajectory Planning



- Phase 7:
  - Jerk constant > 0
  - Acceleration goes to 0
  - Velocity goes to 0
  - Position reaches target

Trapezoidal vs S-shaped acceleration



(a) Third order polynomial S-curve model.

(b) Fourth order polynomial S-curve model.

Source: Yi Fang et al., Smooth and time-optimal S-curve trajectory planning for automated robots and machines, Mechanism and Machine Theory, 2019

Trapezoidal vs S-shaped acceleration

Source: https://www.youtube.com/watch?v=qYJpl7SNoww

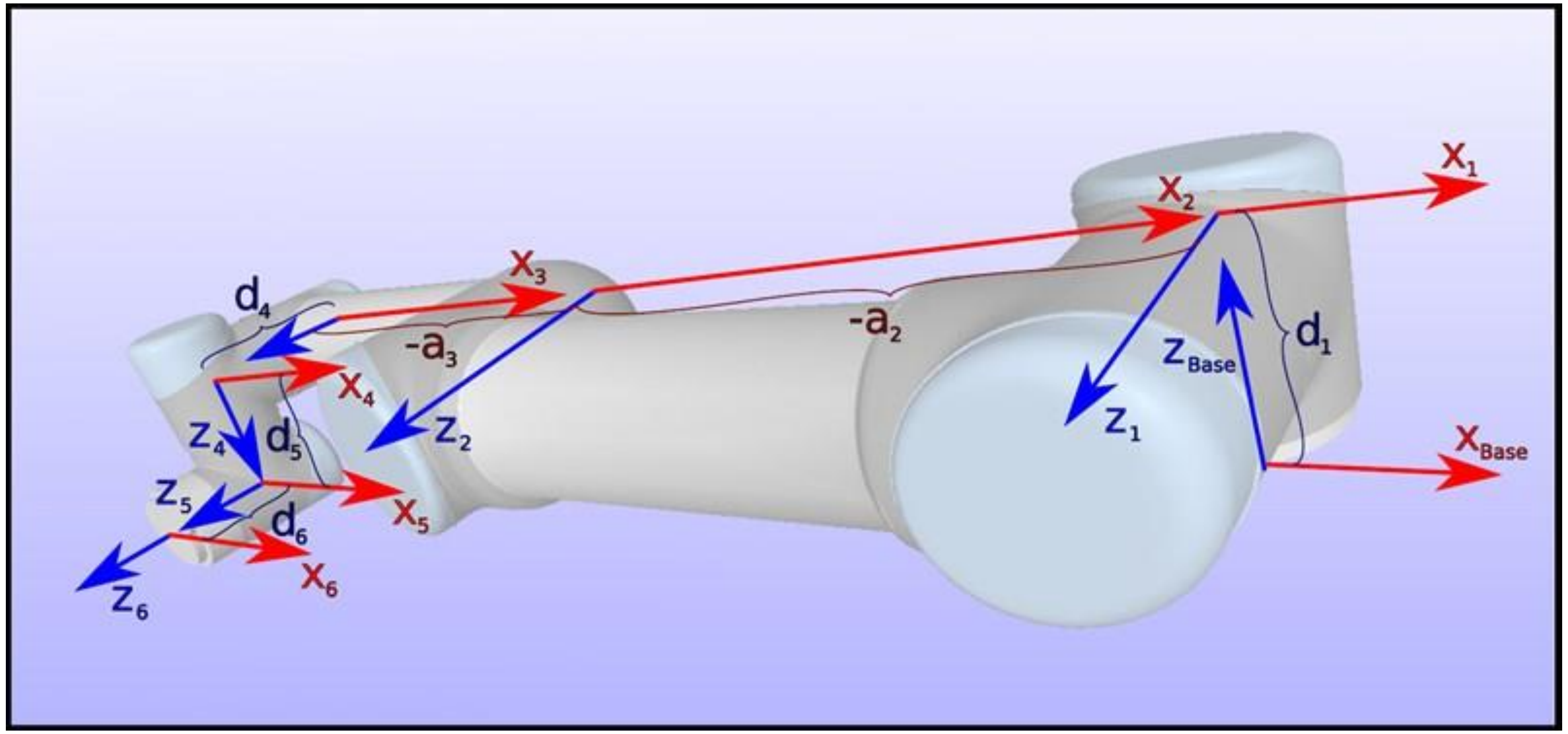
Reference Frames (Coordinate Systems)

- Usually local frames are assignes to different parts of the robot or workpieces
- Fast reprogramming of application: Only re-teach a single frame instead of all target positions
- Common application: Kinematic chains
  - Each part of the robots has its own local frame.
  - Position and orientation defined by transformations

Reference Frames (Coordinate Systems)

- Usage of reference frames is typical in robotics. The controllers and programming languages are designed to easily integrate different frames
- Usual frames:
  - **world frame:** Typicall fixed on the floor or static part
  - **base frame:** Robot base
  - **object frame:** fixed to the workpiece
  - **Tool0 frame:** flange of the robot, where the tool is mount
  - **tool frame:** Tool center point

Reference Frames (Coordinate Systems)



Source: https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/

DH-Parameter

- **Denavit-Hartenberg-Transformation**
- Transformations between frames in kinematic chains
- Standard in forward kinematics
- Transformationen defined with 4 parameter:
  1. Rot(x,α) – Rotation **α** around the x-axis
  2. Trans(x, a) – Translation **a** along the x-axis
  3. Trans(z, d) – Translation **d** along the z-axis
  4. Rot(z, ϴ) – Rotation **ϴ** (joint angle) around the z-axis
- DH-Parameter: (**a, α, d, ϴ**) for every joint

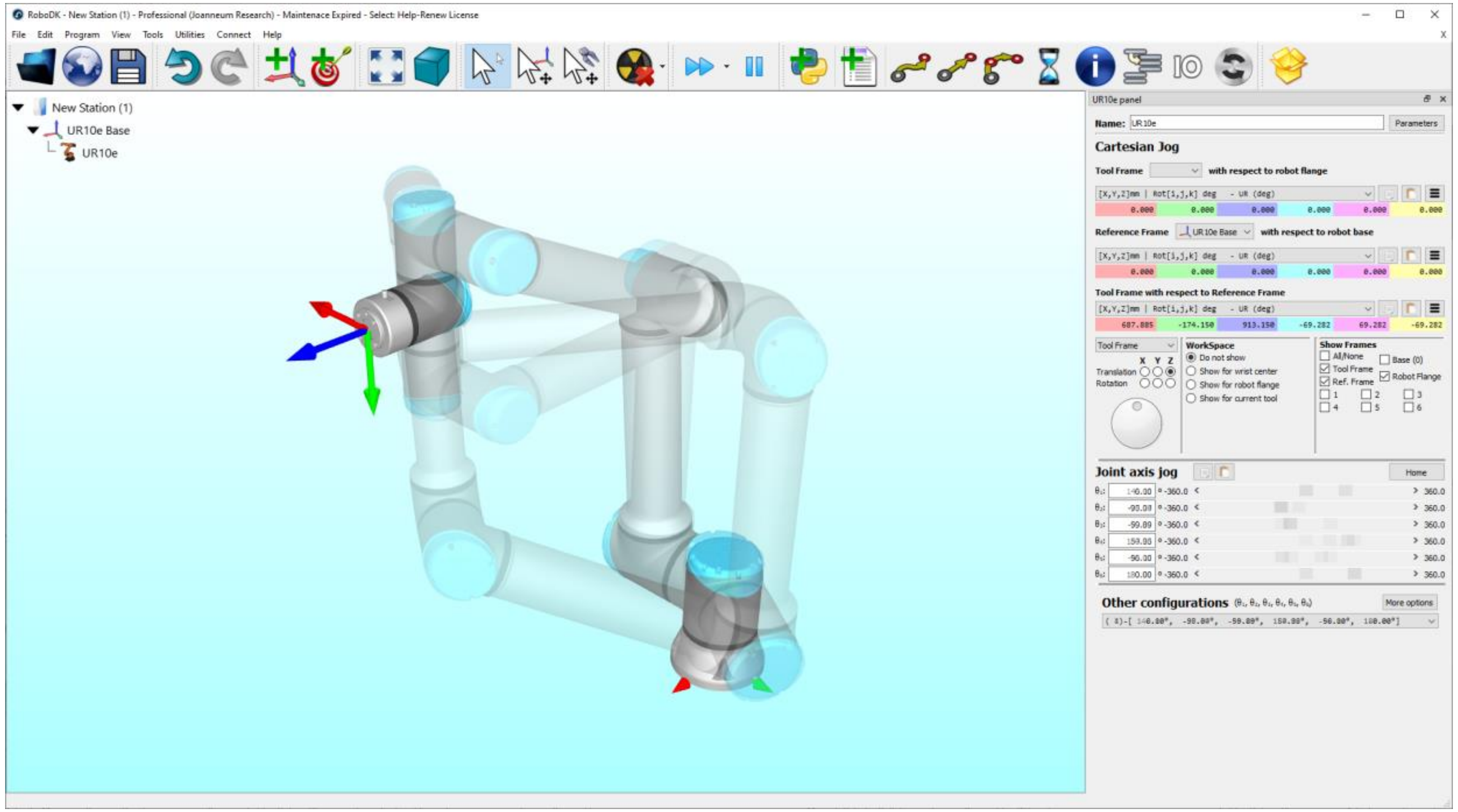
Forward Kinematics

- Known:
  - Joint-positions of the robot (e.g. angular positions)
  - Geometry of the Robot
- Goal:
  - Pose of end-effectos in space
  - Pose = Position & Orientation
  - In 3D-space 6 degrees of freedom
    - Position (**x, y, z**)
    - Orientation (**rx, ry, rz**)
    - Attention: Different forms to express the orientation!
      - Euler-Winkel
      - Axis-Angle
      - Rotationmatrix
      - Quaternionen

Inverse Kinematics

- Known:
  - Pose (Position & Orientation) of the end-effector
  - Geometry of robot
- Goal:
  - Joint positions (e.g. angular positions)
- Usually there exist multiple solutions
  - E.g.. „elbow up“ or „elbow down“
- For robots with more than 6 joints (=kinematically redundant robots) there are usually infinite solutions
  - E.g. elbow can be anywhere on a circle
  - → potential for optimization or obstacle avoidance

Inverse Kinematics



Degrees of Freedom (DoF)

- Number of joints = DoF of a manipulator
- Typically: 6 joints
  - 3 for positioning
  - 3 for orientation
- Less than 6 joints:
  - Not every point can be reached at any orientation
- More than 6 joints:
  - kinematically redundant robots
  - E.g. work behind an obstacle
  - Increased complexity vor calculations and control

Workspace

- Workspace: All points in space, that the end-effector can reach with all possible motions of the manipulator
- 2 sub-categories:
  - reachable workspace:
    All points that can be reached,
    without considering the orientations
  - dextrous workspace:
    All points that can be reached with
    arbitrary orientation

Velocity Kinematics (1)

- Forward:
  Joint velocities → EE-velocities
- Inverse:
  EE-velocities → Joint velocities
- EE-velocities = Twist:
  - Vector with 6 elements $\dot{\boldsymbol{v}} = [\dot{x} \quad \dot{y} \quad \dot{z} \quad \dot{r_x} \quad \dot{r_y} \quad \dot{r_z}]^T$
  - $\dot{x}$, $\dot{y}$, $\dot{z}$ … translationcal velocities
  - $\dot{r_x}$, $\dot{r_y}$, $\dot{r_z}$ … rotational velocities

Porges, Oliver, et al. "Reachability and dexterity: Analysis and applications for space robotics." *Workshop on Advanced Space Technologies for Robotics and Automation-ASTRA*. 2015.

Velocity Kinematics (2)

- $\dot{\boldsymbol{v}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}$
  - $\dot{\boldsymbol{v}}$ … Twist
  - $\boldsymbol{J}(\boldsymbol{q})$ … Jacobian matrix
    - Very important matrix in kinematics
    - 6 x n matrix, n = number of joints
    - If n=6 it is a quadratic matrix and can (usually) be inverted
    - In Singularities: Rank(J) < n → can not be inverted
  - $\dot{\boldsymbol{q}}$ … vector with joint velocities
- $\dot{\boldsymbol{q}} = \boldsymbol{J}^{-1}(\boldsymbol{q})\dot{\boldsymbol{x}}$
  - Only possible, when J has full rank and n=6
  - If $n \neq 6$: usage of the „Pseudo-Inverse“

Porges, Oliver, et al. "Reachability and dexterity: Analysis and applications for space robotics." *Workshop on Advanced Space Technologies for Robotics and Automation-ASTRA*. 2015.

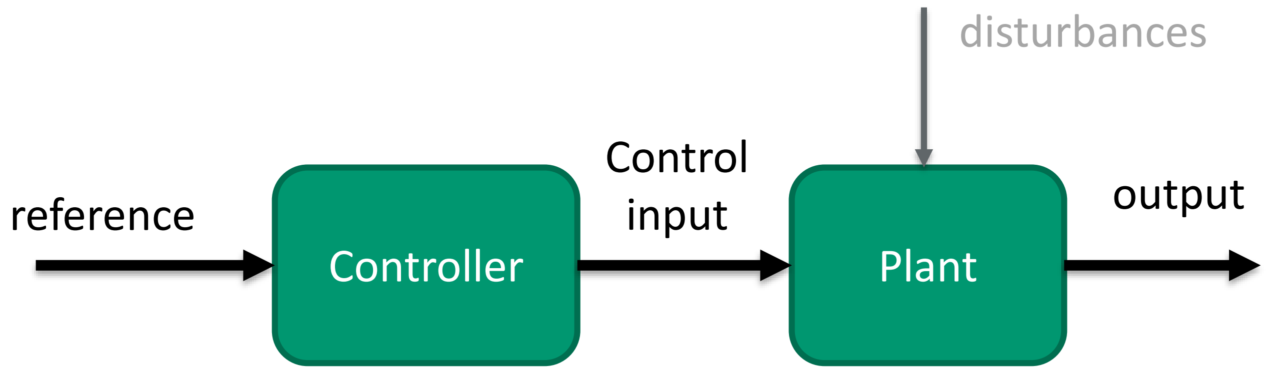
Robot Control

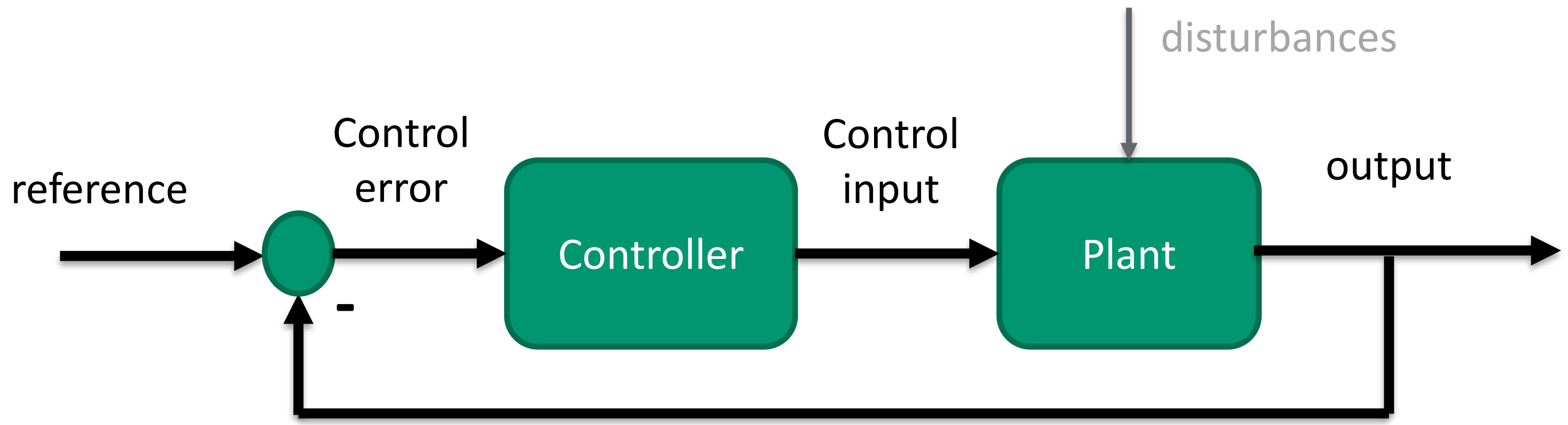
DI Matthias Weyrer, BSc

Three tasks to control a robot…

- Path planning:
  - The path in task space has to be defined
  - Usually from actual position to a target
  - Optionally: Collision avoidance, …
  - Output: Position without any time information
- Trajektory planning
  - Create the time-information
  - Consideration of velocities, accelerations (and jerk)
- Trajektory tracking:
  - Control the robot, to execute the trajectory
  - Usual control input: Motor current of the joints

Feedback vs. Feedforward Control





Important terms

- Direct dynamics
  - Input: Forces/torques, positions, velocities
  - Output: Joint accelerations
  - Computation of the time evolution of $\ddot{\boldsymbol{q}}(t)$ (and then of $\dot{\boldsymbol{q}}(t)$ and $\mathbf{q}(t)$), given the vector of generalized forces (=torques and/or forces) $\boldsymbol{\tau}(t)$ applies to the joints and, in case, the external forces applied to the end-effector, and the initial conditions $\mathbf{q}(t=0)$ and $\dot{\boldsymbol{q}}(t=0)$.

$$\boldsymbol{\tau}(t) \quad \Longrightarrow \quad \ddot{\mathbf{q}}(t) \quad (\ \dot{\mathbf{q}}(t), \mathbf{q}(t)\ )$$

- Inverse dynamics
  - Input: Joint accelerations, positions and velocities
  - Output: Forces/Torques
  - Coputation of the vector $\boldsymbol{\tau}(t)$ necessary to obtain a desired trajectory $\ddot{\boldsymbol{q}}(t)$, $\dot{\boldsymbol{q}}(t)$, $\mathbf{q}(t)$, once the forces applied of the end-effector are known.

$$\ddot{\mathbf{q}}(t), \quad \dot{\mathbf{q}}(t), \quad \mathbf{q}(t) \quad \Longrightarrow \quad \boldsymbol{\tau}(t)$$

Why dynamic modelling of a manipulator?

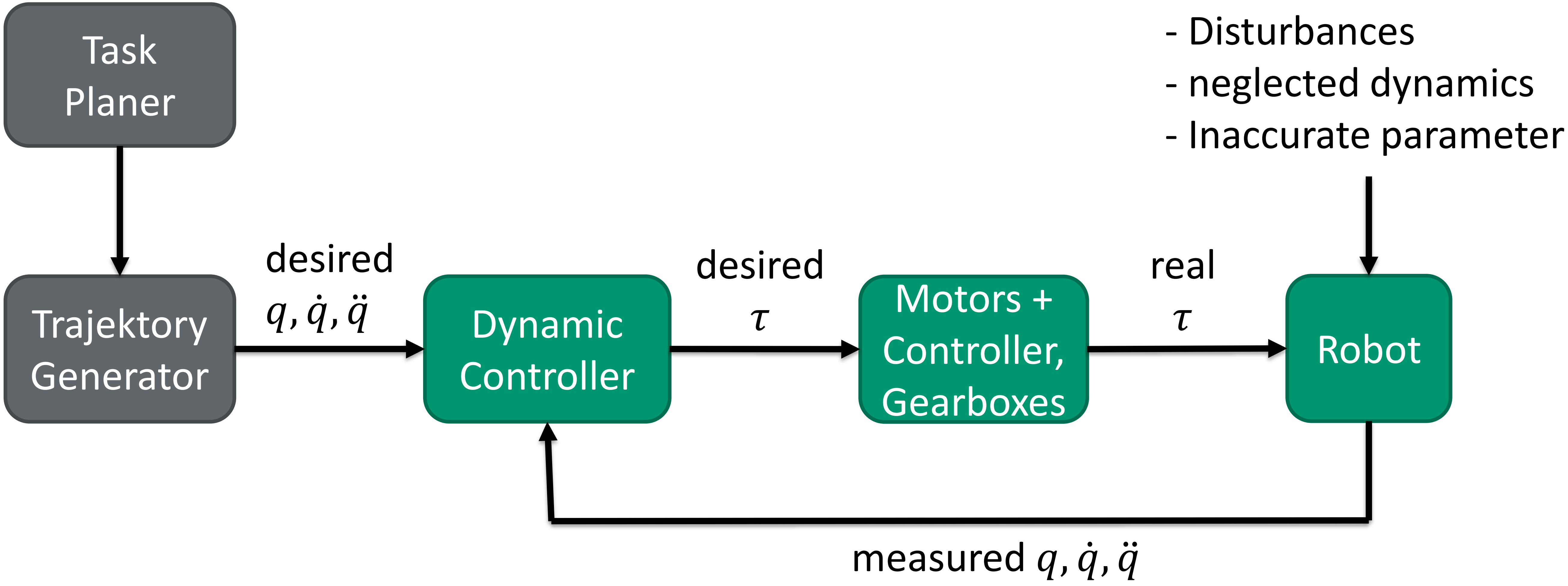
- Simulation:
  - Test desired motions without resorting to real experimentation
- Analysis and synthesis of suitable control algorithms
- Analysis of the structural properties of the manipulator since the design phase

Dynamic Model of a manipulator

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}^T(\mathbf{q})\mathbf{F}_c$$

- $M(q)$ … Joint-space initia matrix
- C$(q, \dot{q})$ … Coriolis and centrifugal forces
- D$(\dot{q})$ … friction
- g(q) … gravity
- τ … joint torques (and/or forces)
- $J^T(q)F_C$ … external (contact) forces

Control Architecture (1)



Control Architecture (2)

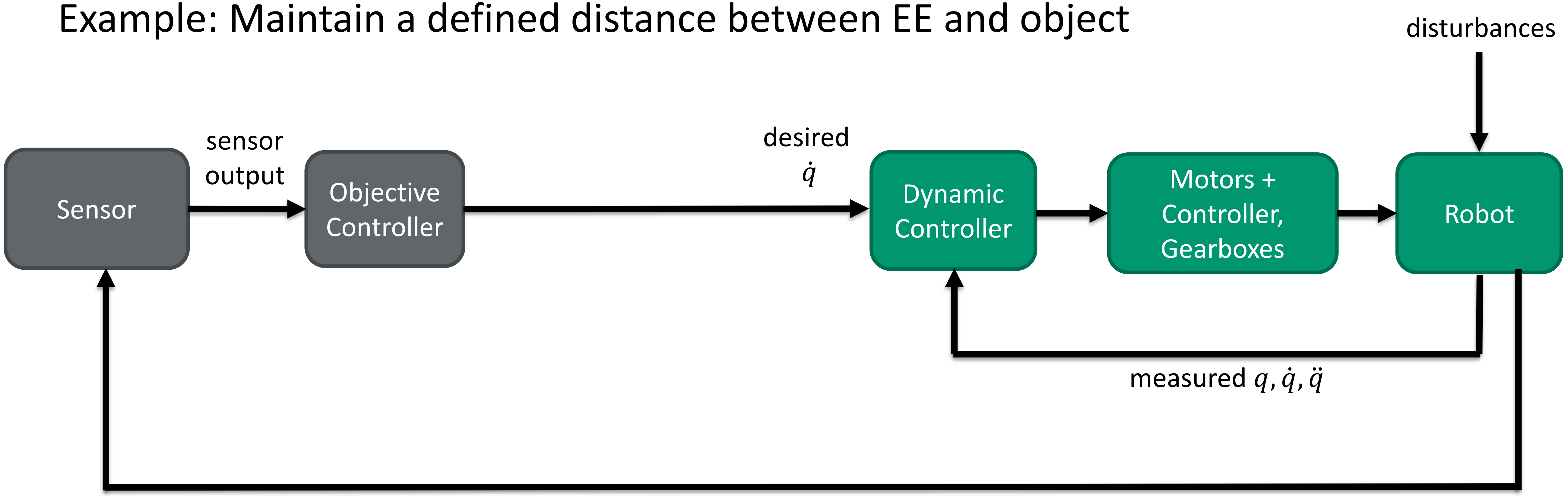
- Typical structure for move commands (moveJ or moveL)
- Inner control loop (dynamic joint control) runs at high frequency (usually 500Hz – 1kHz)
- Usually a PD-Controller is used to control the joint positions
- Jumps on the input (e.g. desired joint positions or velocities) would cause rapid motions of the manipulator
- → Trajectories have to be planned accordingly
  - Controlled accelerations and decelerations

Real-time control (Servoing)

- Motion of the robot is not know a-priori
- Calculate desired robot motion in real-time based on sensor input
  - Vision sensor
  - Force/Torque sensor
  - Distance sensor
- Example: Keep target object in the center of the camera-image
- Usual Control inputs:
  - Joint velocities
  - Joint positions
  - EE velocities
  - EE positions

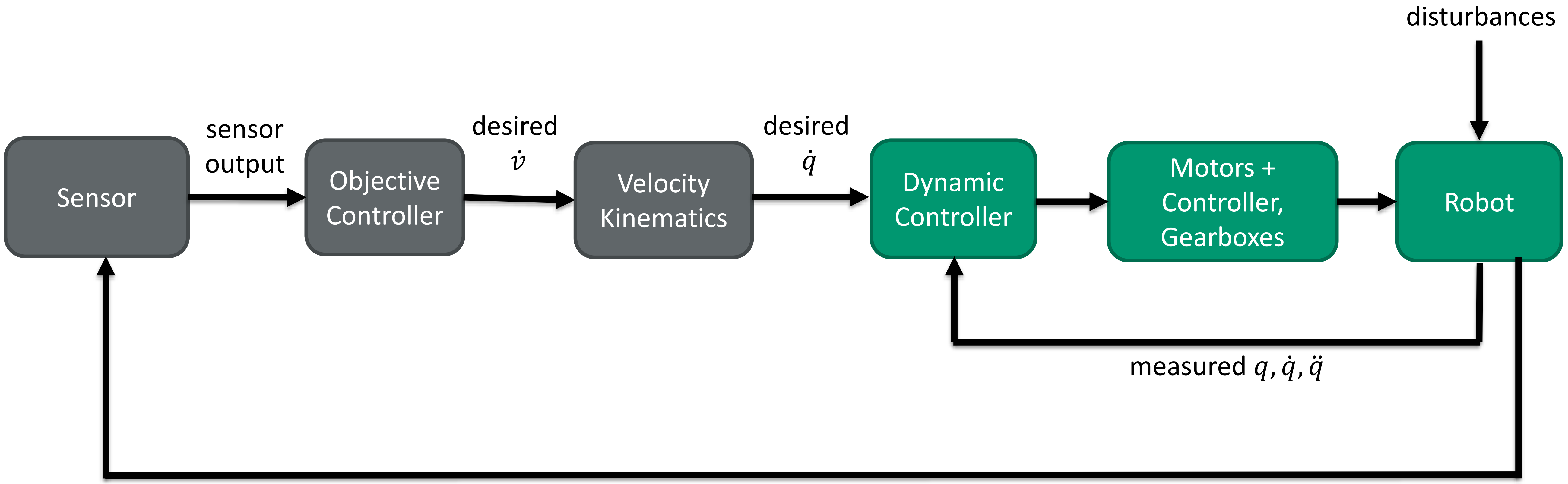
Real-time control - Joint velocities

Example: Maintain a defined distance between EE and object



Most cases: We want to specify the motion of the EE → Velocity kinematics

Real-time control - EE velocities



Real-time control (Servoing)

- Example applications:
  - Visual tracking of object
  - Compliance/admittance control based on force/torque measurements
- Attention:
  - Rapid changes of the control input might lead to rapid motions of the robot
  - Not all robots offer the possibility to use these control techniques
  - Usually real-time capabilities are needed (fixed cycle-time)
- Combination of Trajectory tracking and servoing possible:
  - Execute a precalculated trajectory, perform online corrections based on sensor data

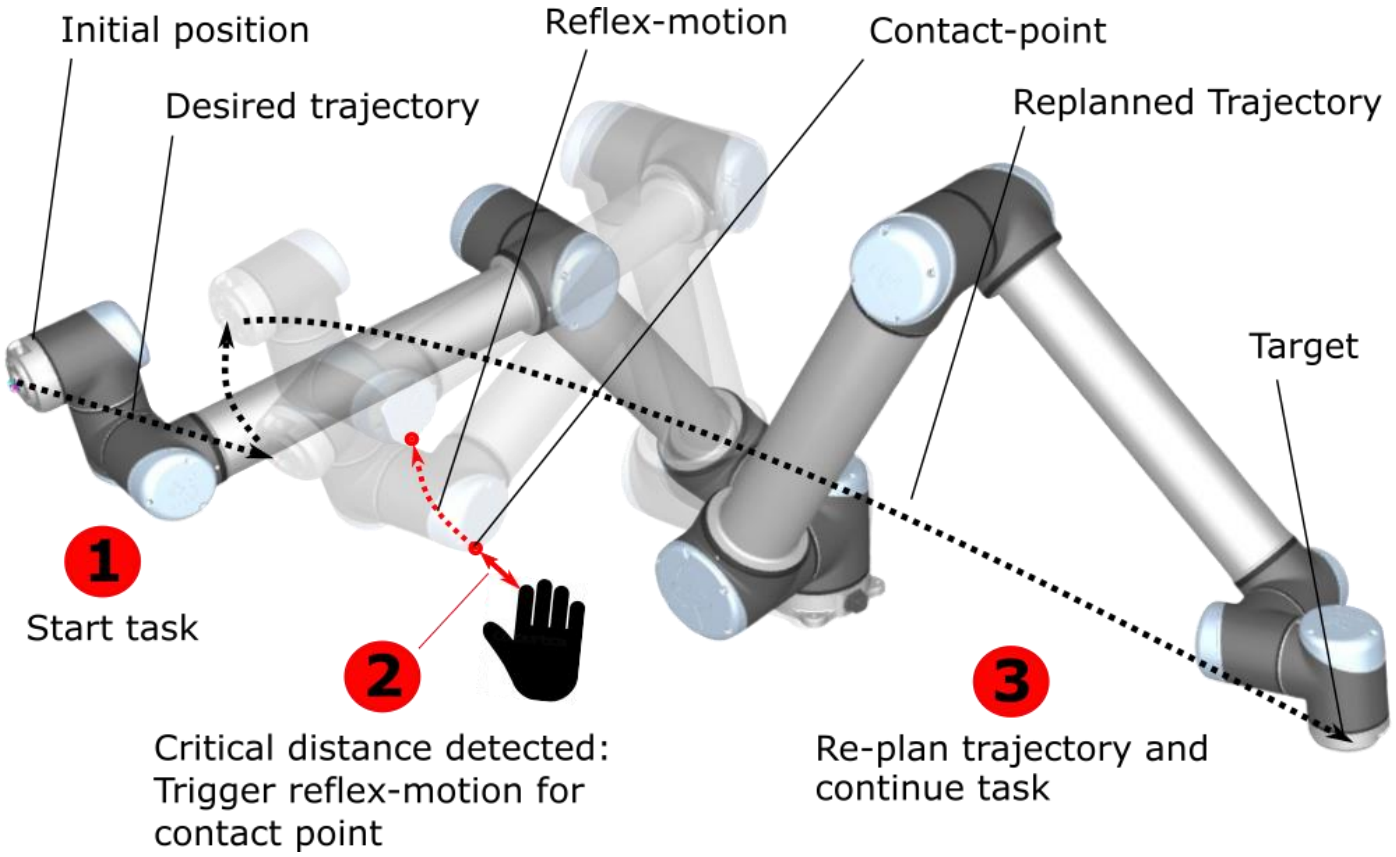
Collision Avoidance

- Within industrial environments, usually the workspace of the robot is known
  - No human enters the workcell while robot is moving
  - no unexpected objects present
  - → No Collision avoidance needed
- Dynamic environments
  - Presence of humans and other object must be expected
  - Examples:
    - Collaborative Applications
    - Mobile robots
  - Collisions must be considerd → Collision avoidance strategy

Collision Avoidance

- Perceive the state of workspace with sensors
  - 3D Cameras, Radar, Laser,…
  - Detect humans and external objects
- Two main approaches:
  - Reactive collision avoidance
    - Plan and start a trajectory without the knowledge of any obstacles
    - During the execution, determine distances to obstacles and trigger a reaction of the robot e.g. pause the motion, adapting the trajectory, start a reflex motion of the robot
  - Planning of collision free trajectories
    - Positions of obstacles are considered in the trajectory planning
    - Very complex, high computational effort (high dimensional problem)

Reactive Collision Avoidance

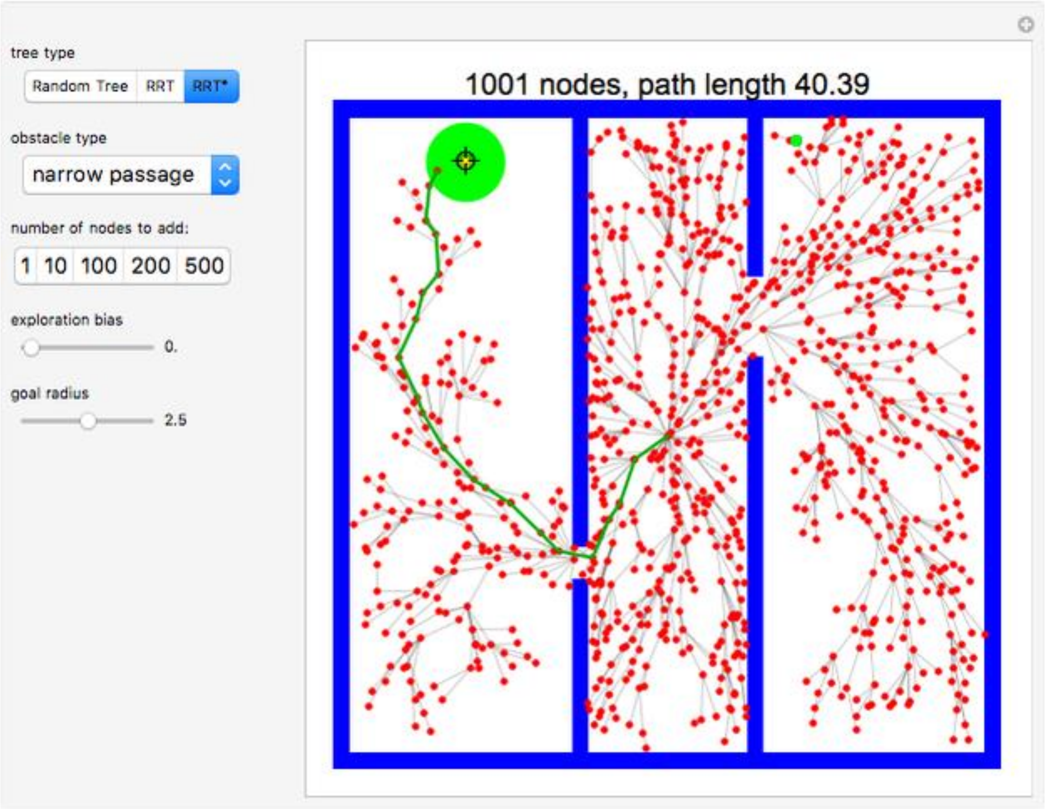


Planning of collision free Trajectories

- probabilistic, sample-based algorithms
  - Probabilistic Road Maps (PRMs) [1]
  - Rapidly-exploring Random Trees (RRTs) [2]
  - Fast Matching Tree (FMT) [3]
  - Stable Sparse RRT (SST) [4]
- Artificial potential field algorithms
  - Use virtual forces to avoid collisions
  - Problem: Local minima

[1] Kavraki, L. E., Svestka, P., Latombe, J. C., and Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces.
[2] LaValle, S., and Kuffner, J. (2001). "Rapidly-exploring Random Trees: Progress and prospects," in *Algorithmic and Computational Robotics: New Directions*
[3] Janson, L., Schmerling, E., Clark, A., and Pavone, M. (2013). Fast marching tree: a fast marching sampling-based method for optimal motion planning in many dimensions
[4] Bekris, K. E., Littlefield, Z., and Yanbo, L. Y. (2016). Asymptotically optimal sampling-based kinodynamic planning.

Planning of collision free Trajectories



https://demonstrations.wolfram.com/RapidlyExploringRandomTreeRRTAndRRT/

Probabilistic, Sample-based Algorithms

- Trajectory generation for a 6-DoF manipulator is a high dimensional problem
- High computational effort to check distances between detected objects and the geomety of the robot (not only the EE)
- Result may not be the optimal (or a desired) trajectory
- Computation time of the trajectory often not predictable

Robot Calibration

DI Matthias Weyrer, BSc

JOANNEUM RESEARCH ROBOTICS

DIHSÜD Digital Innovation Hub

FFG Forschung wirkt.

LAND KÄRNTEN

Das Land Steiermark

Robot Calibration

- Is a process to improve the accuracy of robots
- Industrial robots are highly **repeatable** but **not accurate**
- Calibration is needed to identify certain parameters in the kinematic structure of a robot
  - Relative positions of robot links

Why robot calibration?

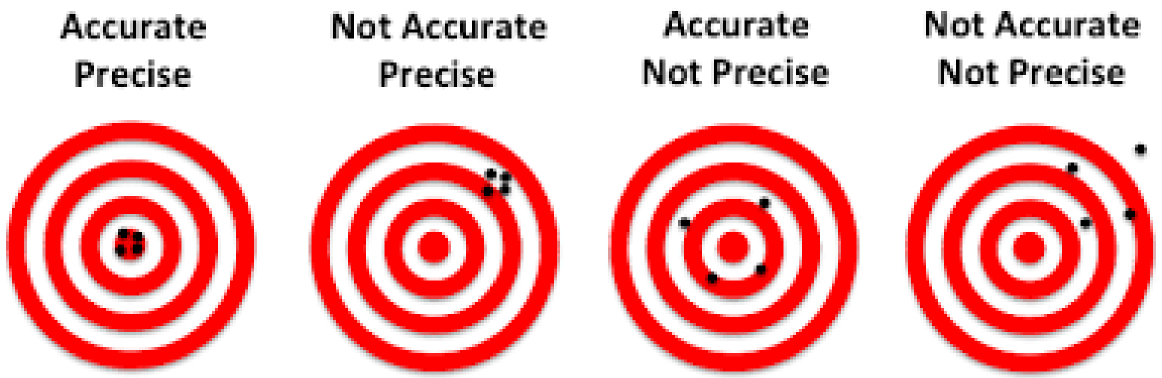
- Calibrated robots have a higher absolute and relative positioning accuracy
- Important for
  - offline-programming
  - migration of a program to another robot of the same type (robot exchangability)

Three levels of calibrations

- Level 1:
  - Difference between actual and reported joint positions
- Level 2: Kinematic calibration
  - Concerns the entire robot geometry
  - Includes angle offsets and link lengths
- Level 3: Non-kinematic calibration
  - Modelling of errors other dan geometric defaults
  - E.g. stiffness, joint compliance
- Usually Level-1 and Level-2 are sufficient for most practical needs

Accuracy criteria and error sources

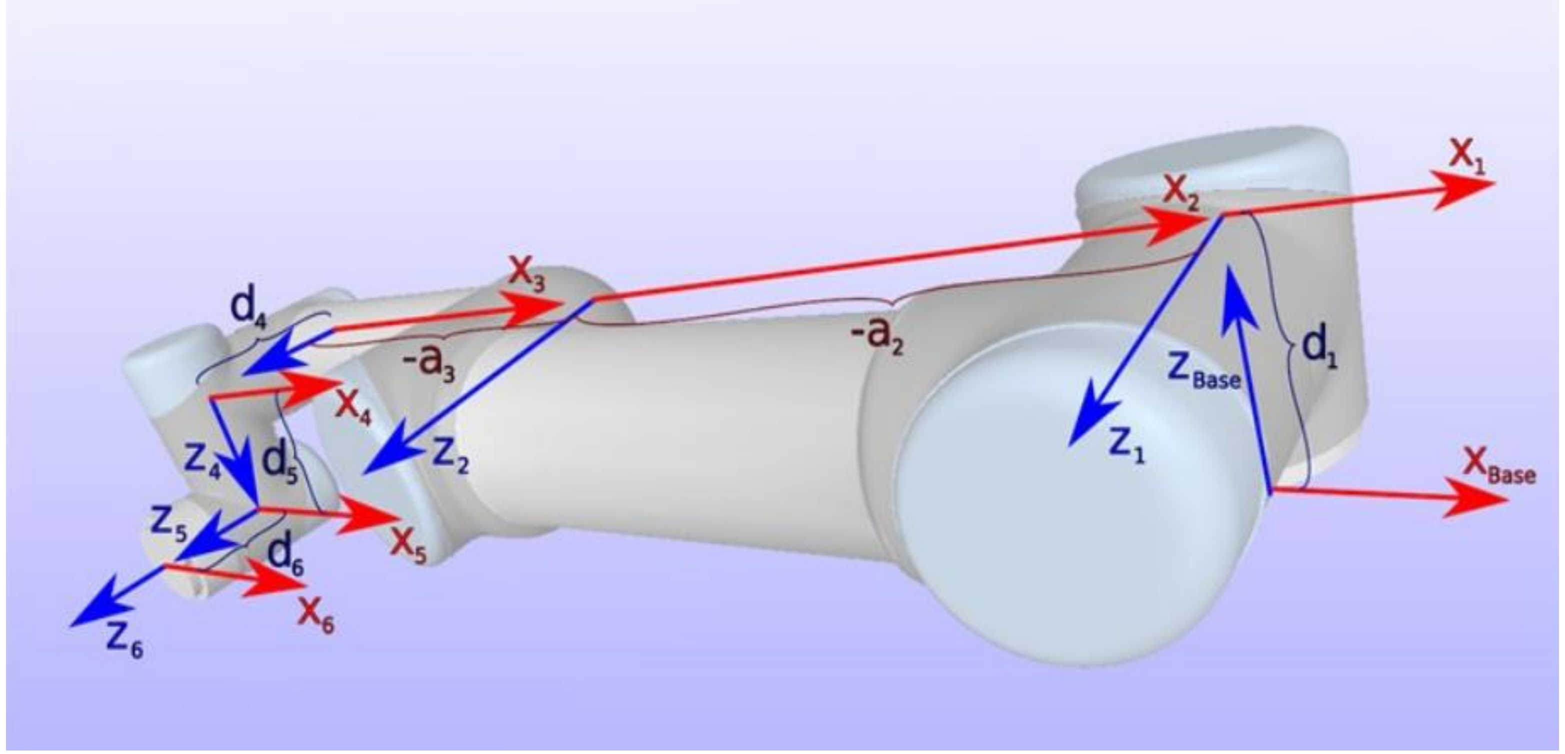
- International standard ISO 9283:
  - Suggests test procedures in order to obtain appropriate parameter values
  - Most important criteria
    - Pose accuracy
    - Pose repeatability



Source: https://danielmiessler.com/blog/difference-precision-accuracy/

DH-Parameters

UR10e				
Kinematics	theta [rad]	a [m]	d [m]	alpha [rad]
Joint 1	0	0	0.1807	π/2
Joint 2	0	-0.6127	0	0
Joint 3	0	-0.57155	0	0
Joint 4	0	0	0.17415	π/2
Joint 5	0	0	0.11985	-π/2
Joint 6	0	0	0.11655	0



https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/

DIHSÜD Digital Innovation Hub

Universal Robots

https://www.universal-robots.com/download/manuals-e-series/calibration/calibration-manual-e-series/

https://www.siemens-pro.ru/docs/ur/calibrationManual.pdf

Example of calibration service

Video:
https://www.creaform3d.com/en/metrology-solutions/services/robot-absolute-accuracy-calibration-services

Hand eye calibration

- Determining the transformation between a robot end-effector and a sensor (e.g. camera)
- Usually:
  - Capturing a set of images of a static object of known geometry (e.g. chessboard like calibration plate)
  - Images are taken from different poses (robot configurations)
  - Two unknown transformations:
    - Base-Frame to calibration object
    - EE to camera
  - Optimization used to determine the transformations