Intelligent Robot Control

Lecture 2: Introduction to Control

Tadej Petrič

tadej.petric@ijs.si

Institut
"Jožef Stefan"
Ljubljana, Slovenija

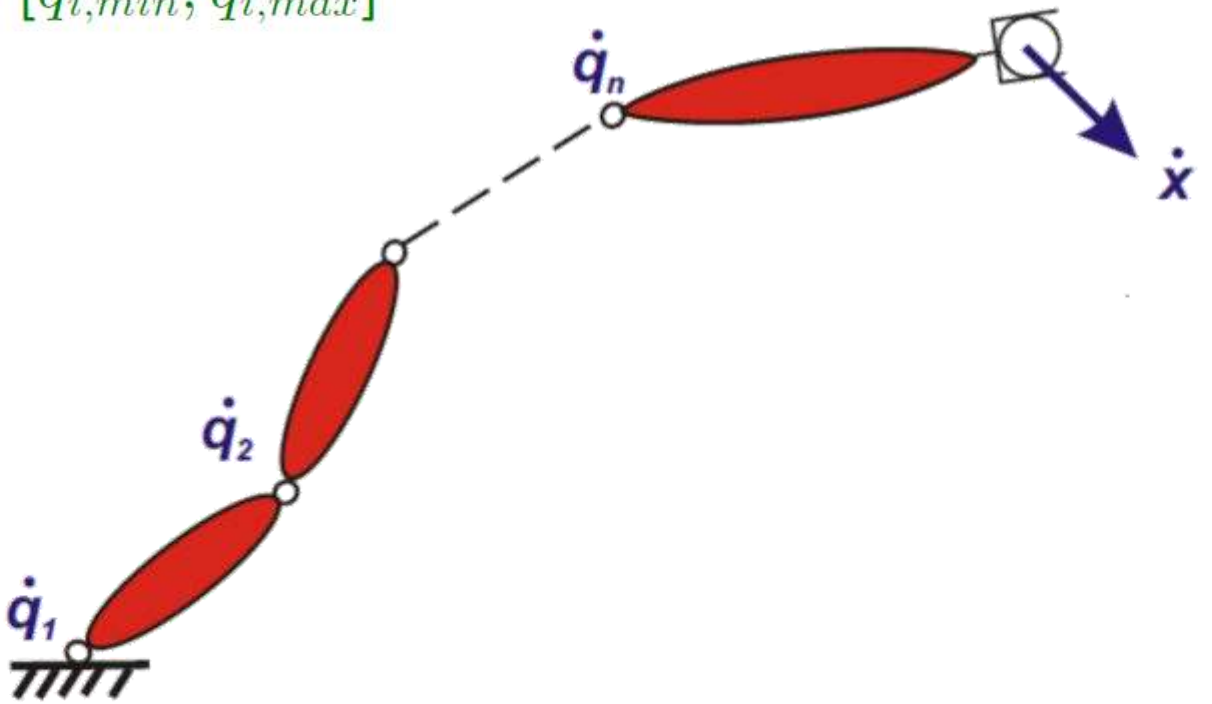
Serial link manipulators

- Configuration on n-DOF robot is described by joint coordinates

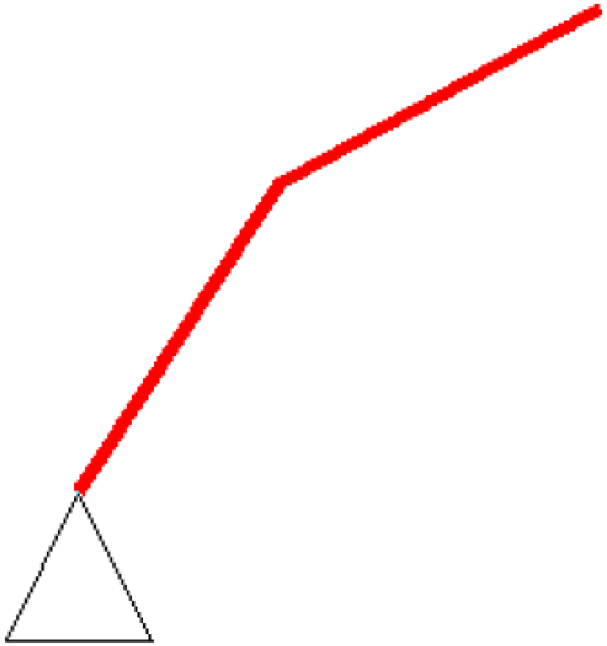
$$\boldsymbol{q} = [q_1, q_2, \ldots, q_n]^T \qquad q_i \in Q_i = [q_{i,min}, q_{i,max}]$$

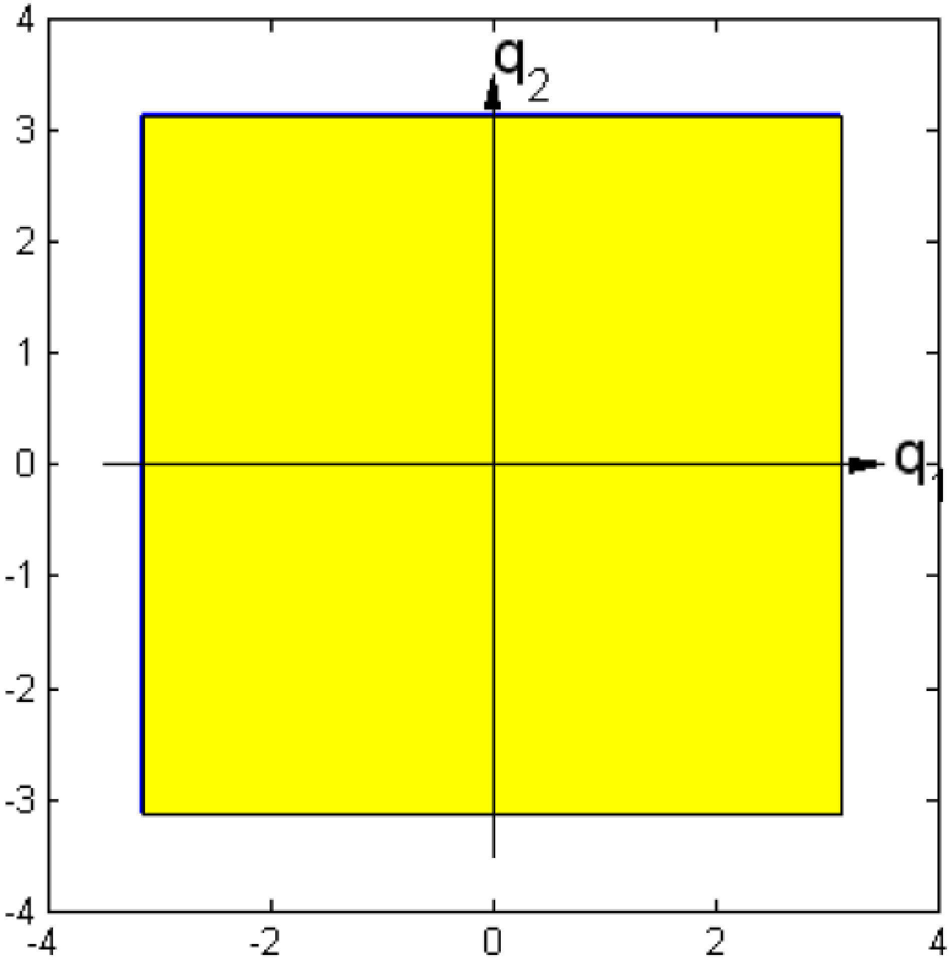
- Configuration space:

$$CS : Q_1 \times Q_2 \times \ldots \times q_n$$



2R Manipulator





$CS : [-\pi, \pi] \times [-\pi, \pi]$

Forward kinematics

- We relate the configuration to the position and orientation of the end-effector
- Workspace (WS) is space the robot can reach (taskspace, operational space). The workspace coordinates are usually

$$\boldsymbol{x} = [x, y, z, \alpha, \beta, \gamma]^T \qquad (m \times 1) \text{ vector}$$

- Usually is 6-dimensional ($m = 6$), but can be less dimensional ($m < 6$)
- Forward kinematics relates CS and WS, i.e. given the conguration q find the function

$$x = f(q)$$

Inverse kinematics

- The inverse kinematics problem is: given the position of the end-effector x in WS find the joint coordinates q

$$q = f^{-1}(x)$$

- Existence: For a solution x must lie in WS.
- Reachable workspace: is the subspace of WS where the robot can reach all positions with all orientations
- Dexterous workspace: is the space where the robot can reach all positions with at least one orientation

Velocities

- To find the relation between the velocities. we have to differentiate

$$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{q})$$

- with respect to t

$$\frac{\partial \boldsymbol{x}}{\partial t} = \frac{\partial}{\partial t}\boldsymbol{f}(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}(\boldsymbol{q})}{\partial \boldsymbol{q}} \frac{\partial \boldsymbol{q}}{\partial t}$$

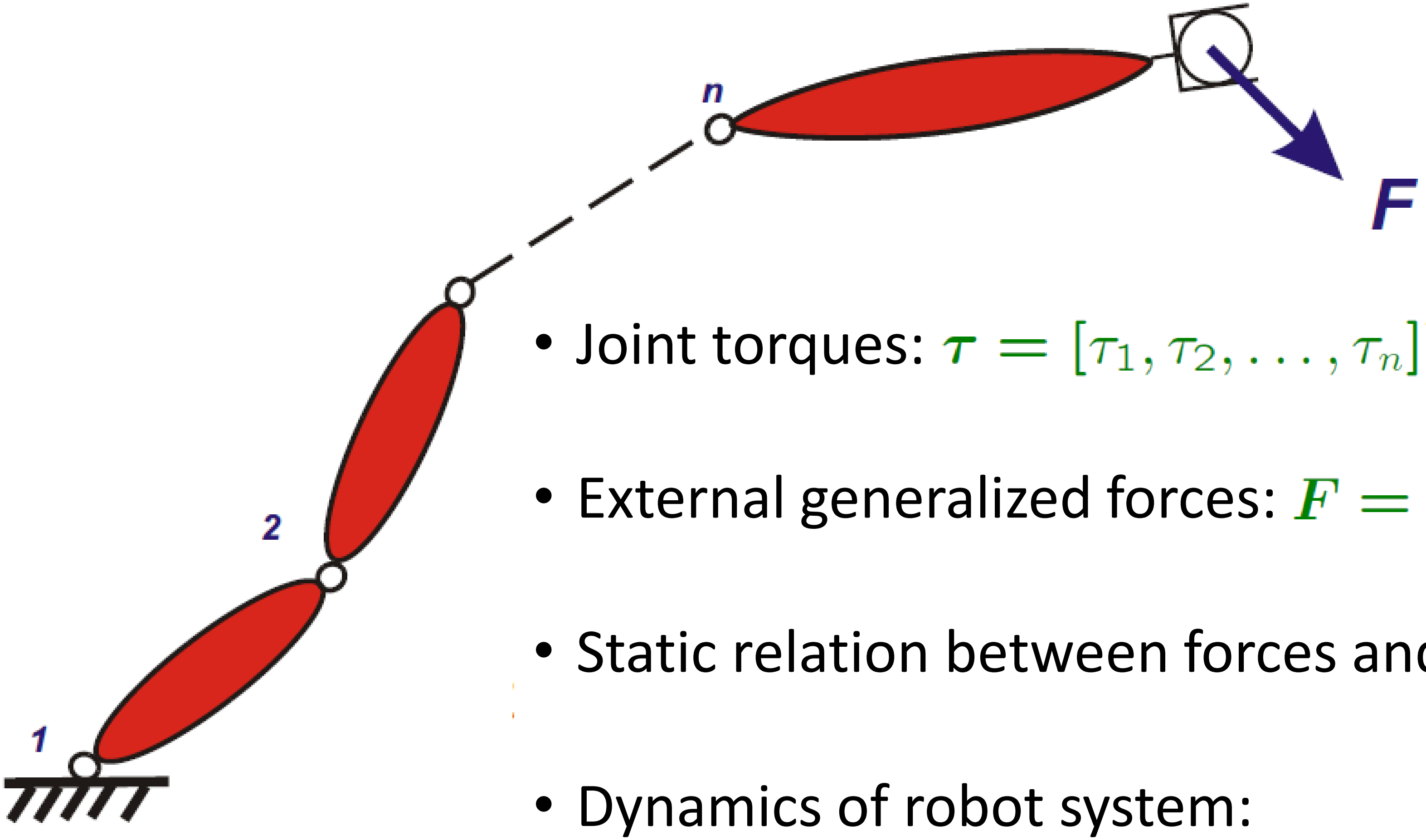
$$\dot{\boldsymbol{x}} = \mathbf{J}\dot{\boldsymbol{q}}$$

$$\mathbf{J} = \frac{\partial \boldsymbol{f}(\boldsymbol{q})}{\partial \boldsymbol{q}} \qquad (m \times n) \text{ Jacobian matrix}$$

- Assuming that $\mathbf{J}^{-1}$ exists, we can get the relation between the $\dot{\boldsymbol{x}}$ and $\dot{\boldsymbol{q}}$

$$\dot{q} = \mathbf{J}^{-1}\dot{x}$$

Robot dynamic models

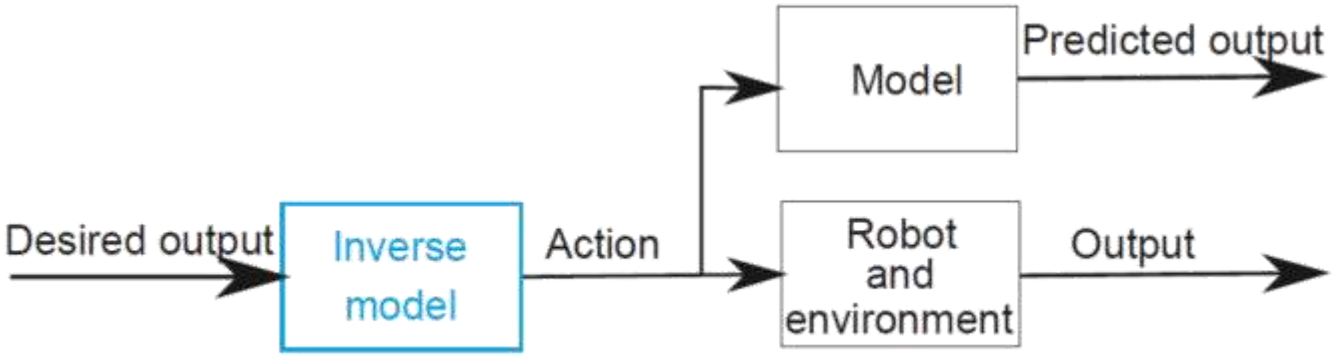


- Joint torques: $\boldsymbol{\tau} = [\tau_1, \tau_2, \ldots, \tau_n]^T$
- External generalized forces: $\boldsymbol{F} = [F_x, F_y, F_z, M_x, M_y, M_z]^T$
- Static relation between forces and torques: $\boldsymbol{\tau} = \mathbf{J}^T \boldsymbol{F}$
- Dynamics of robot system:

$$\tau = \mathrm{H}(q)\ddot{q} + \mathrm{C}(q, \dot{q})\dot{q} + g(q) - \mathbf{J}^T \boldsymbol{F}$$

What is robot control?

- Robot control is the problem of defining joint torques generated by actuators in joints necessary to execute the desired task.



- To predict the motion caused by action we need the model of the robot and the environment.
- To define the action necessary for the desired motion we need the inverse model.
- Using observations of the systems states to change actions → Close-loop control

Robot control levels

- Robot control is not just a close-loop control.
- Control levels:

Strategic	Learning
	Task planning
Tactic	Motion coordination
	Trajectories generation
Executive	Movement execution
	Close-loop control

- Which levels are implemented depends on the task complexity.
- **Higher task complexity → Higher levels included**

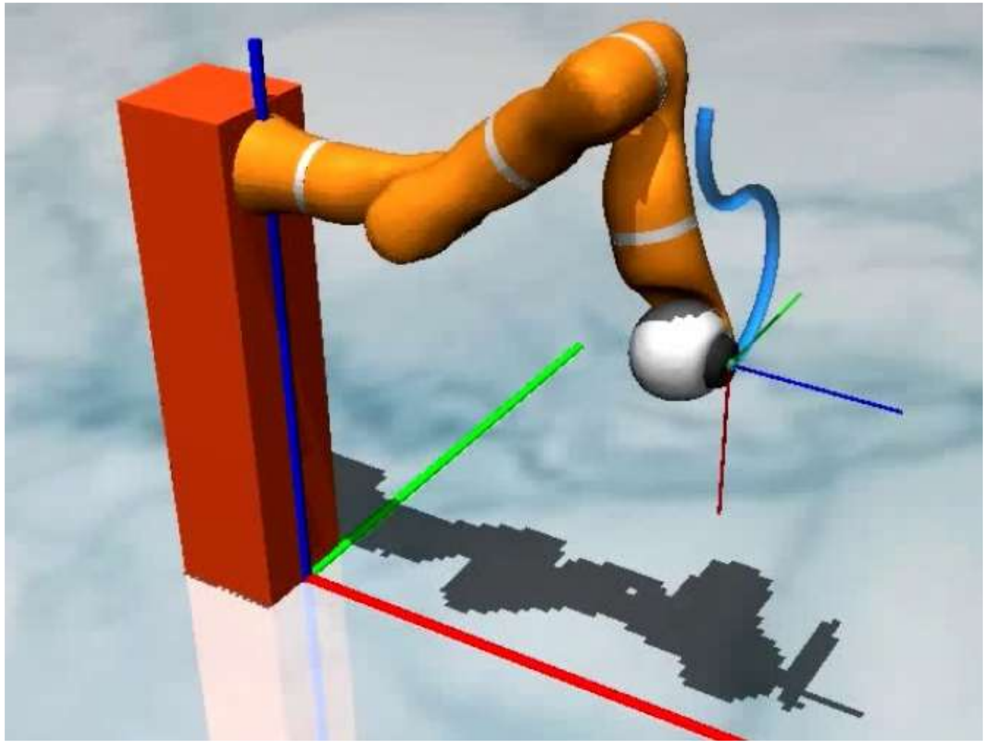
What do we mean by robot control?

- Different level of definitions may be given to robot control
  - successful task completion
  - accurate execution of a trajectory
  - zero position error
- Control system is hierarchically organized

- Different cooperating models, objectives, methods are used at the various control layers

Control strategies

- The selected control strategy influences the characteristics of the robot system.
- The strategy selection depends on:
- robot task:
  - task workspace
  - free motion or motion in contact
  - constraints (e.g. velocity, acceleration, torques)
  - . . .
- mechanical structure of the robot manipulator
- actuators (electric, hydraulic, pneumatic)
- gears (ration, compliance,. . .)
- sensors (joint torques, external forces, vision, . . .)
- . . .



Evaluation of control performance

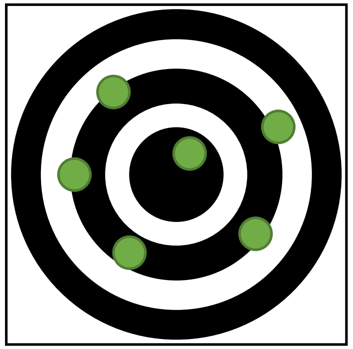
- Execution quality in nominal conditions
  - Velocity and/or speed of task performance
  - Accuracy and /or repeatability (in static and dynamic terms)
  - Energy efficiency and/or requirements

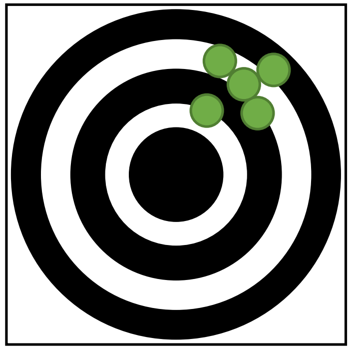
- Robustness in perturbed conditions
  - Adaptation to changing environments
  - High repeatability despite disturbances, changes of parameters, uncertainties, modeling errors

- Performance can be improved thanks to models and by a generalized use of feedback, using more sensor information

Examples of static positioning

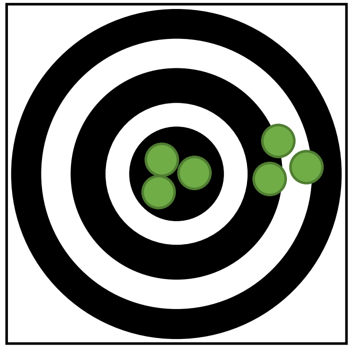
Poor accuracy
Poor repeatability

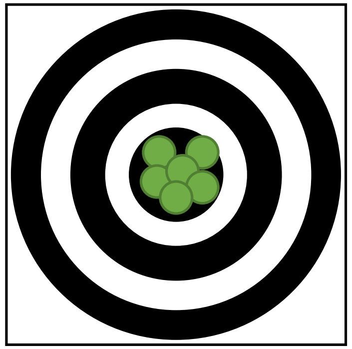




Poor accuracy
Good repeatability

Good accuracy
Poor repeatability





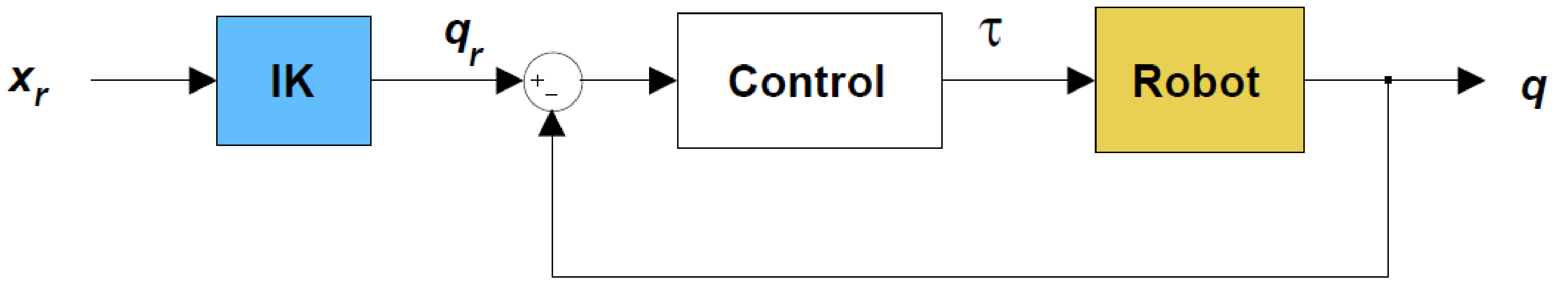
Good accuracy
Good repeatability

Control schemes and uncertainty

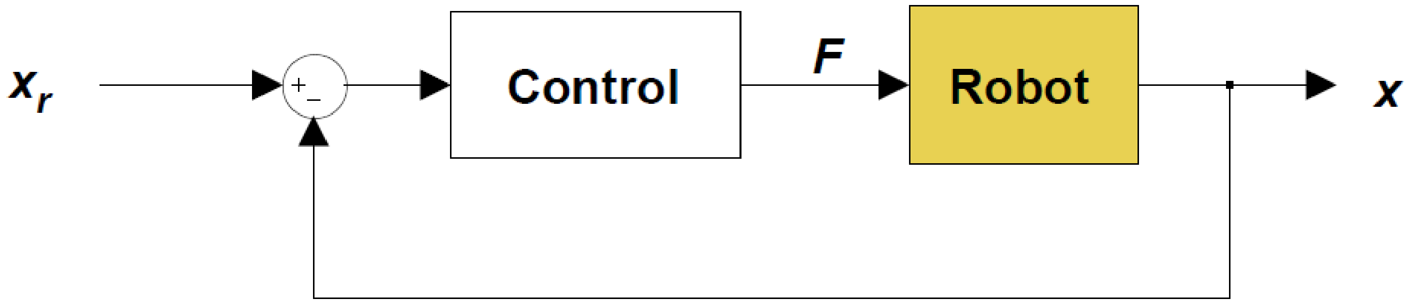
- Feedback control
  - Insensitivity to mild disturbances and small variations of parameters
- Robust control
  - Tolerates relatively large uncertainties of known range
- Adaptive control
  - Improves performance on line, adapting the control law to a priori unknown range of uncertainties and/or large (but not too fast) parameter variations
- Intelligent control
  - Performance improved based on experience: LEARNING

Basic robot control schemes

- Typically for robots
  - tasks are defined in operational workspace as end-effector motion or end-effector forces
  - robot actions are generated in the joint space.
- Joint space control:

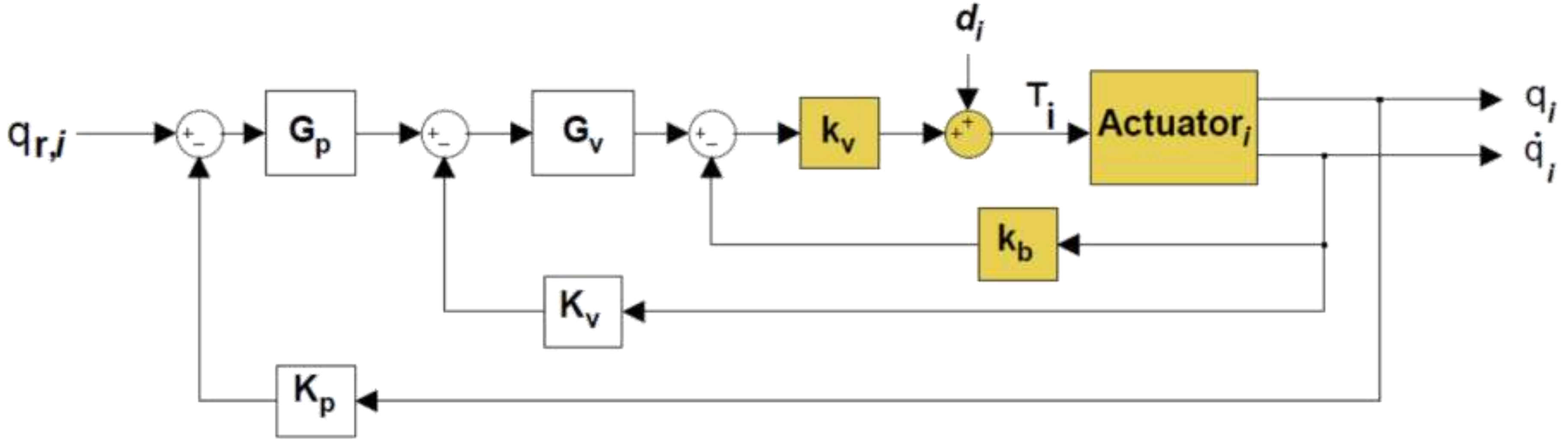


- Task space control:



Independent control

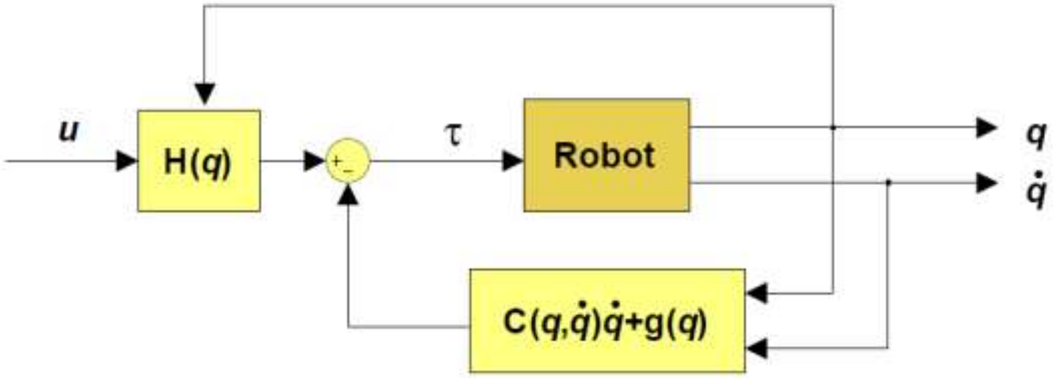
- Basic robot control is the decentralized control: each robot joint is an independent servosystem and interactions between joint are considered as disturbances.



- Positioning and manipulation
- Standard techniques for control design (RL, LQR, . . .)
- Small interactions for high-ratio gears
- Widely used in industrial robots

Inverse dynamics control

- For high accuracy and fast motion the robot dynamics has to be considered in the control.
- Computed torques technique:



$$\tau = \mathrm{H}(q)u + \mathrm{C}(q,\dot{q})\dot{q} + g(q)$$
$$u = \ddot{q}$$

- Due to compensation of nonlinearities and interactions in the close-loop the robot system is decomposed into independent subsystems and the control problem is reduced to a simpler problem of controlling independent linear systems.

Inverse dynamics control

- The dynamic close-loop properties are defined by the selection of the outer loop control algorithm. For example, using PD controller

$$u = \ddot{q}_r + \mathbf{K}_v \dot{\mathbf{e}}_q + \mathbf{K}_p \mathbf{e}_q$$

- the close-loop dynamics of a system becomes

$$\ddot{\mathbf{e}}_q + \mathbf{K}_v \dot{\mathbf{e}}_q + \mathbf{K}_p \mathbf{e}_q = 0$$

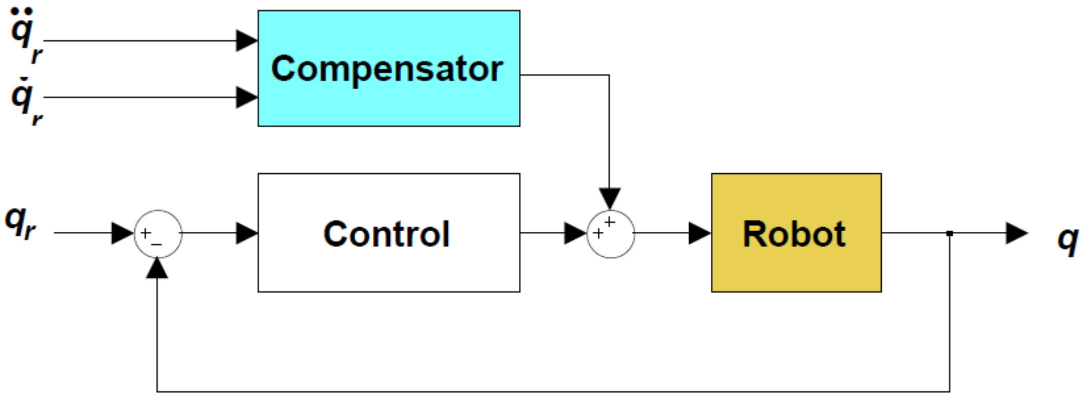
- Drawbacks:
  - Exact dynamic robot model must be known
  - High computational complexity
- Solution: An approximate or partial model can be used. The errors are compensated by the outer-loop control algorithms. Suitable approaches:
  - Variable structure control
  - Adaptive control
  - Independent joint compensation

Compensators

- Some tasks require exact motion execution, i.e. good trajectory tracking for fast motion.
- Close-loop control schemes do not assure the required tracking accuracy.

$$\ddot{e}_q + \mathbf{K}_v \dot{e}_q + \mathbf{K}_p e_q = 0$$

- Tracking error can be reduced by using addition feedforward compensators.



- Better accuracy without affecting the stability of the system.
- It is necessary to know the system (models).
- Real system constraints can influence the accuracy of the system.

Task space control

- Most tasks are defined as the end-effector motion, i.e. a motion the task space.
- Using the kinematic transformation we could transform the task form task space to joint space and design the control in joint space. When accurate motion is required or different behavior in different task space directions required, it is necessary to design the control in the task space.
- Relation between the joint space velocities and the task space velocities

$$\dot{x} = \mathbf{J}\dot{q}$$

- If the inverse relation exists

$$\dot{q} = \mathbf{J}^{\dagger}\dot{x}$$

- then the dynamics of the robot system in the task space is given in the form

$$F = \Lambda(x)\ddot{x} + \Gamma(x, \dot{x})\dot{x} + \xi(x) - F_{ext}$$

Optimal control

- The optimal control problem is to define the state trajectories of the dynamic system, which in some time optimize a selected performance index considering boundary conditions and different constraints.

$$\mathbf{J} = \int_0^{t_f} L(\boldsymbol{x}(t), \boldsymbol{u}(t), t)\mathrm{d}t$$

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f$$

Robot constraints:

- joint torques
- joint velocities
- joint limits
- task constraints

What to optimize:

- minimal torques
- minimal task execution time
- optimal pose

Time optimal control

- In many cases (palletizing, manipulation, spot welding) it is desired to have short cycle times.
- Problem: To define robot motion in such a way that the robot will move along the desired path considering all constraints in minimal time.

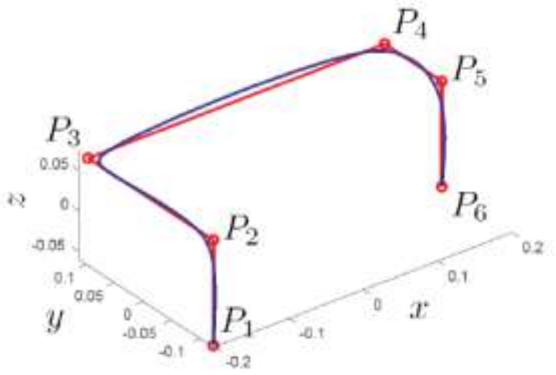
$$t_f = \int_{s_0}^{s_f} \frac{1}{\dot{s}} \mathrm{d}s$$

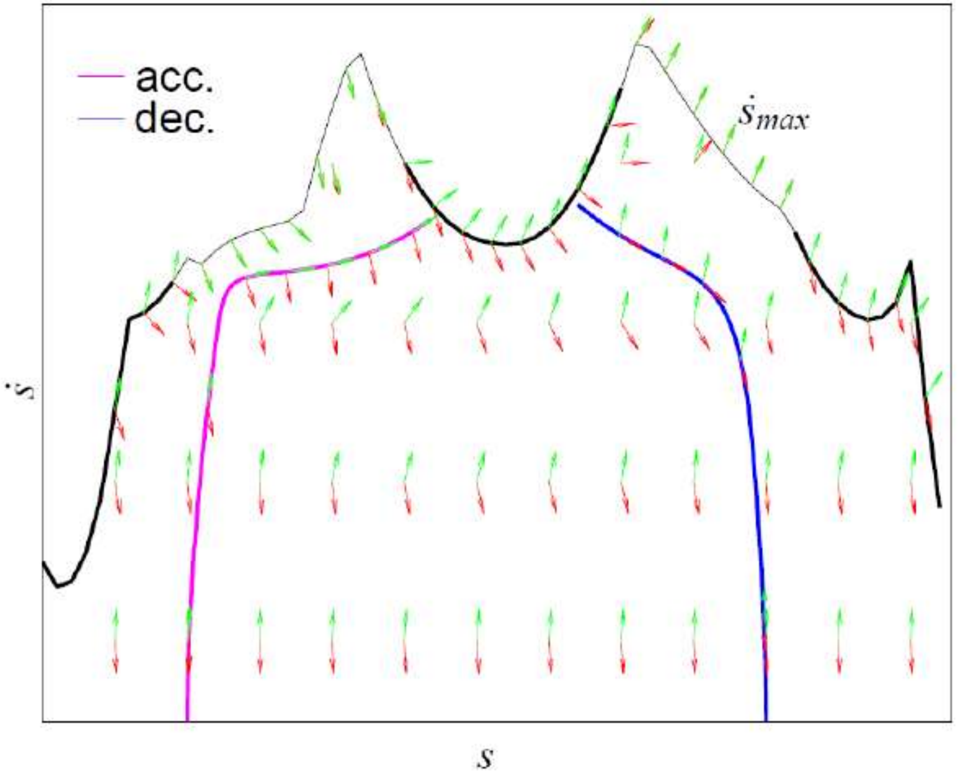
- Constraints:
  - robot manipulator (torques, velocities)
  - task (path, velocities, accelerations).
- Strategy:
  - find the optimal path
  - find the optimal velocity prole

Time optimal control

- Robot dynamics constraint by the path $\boldsymbol{x} = \boldsymbol{f}(s)$ is given in the form

$$\boldsymbol{\tau} = \boldsymbol{a}_1(s)\ddot{s} + \boldsymbol{a}_2(s)\dot{s}^2 + \boldsymbol{a}_3(s)\dot{s} + \boldsymbol{a}_4(s)$$





$\boldsymbol{f}') + \boldsymbol{h}(\boldsymbol{q}, \mathbf{J}^{-1}\boldsymbol{f}')$