# Tuning Model Predictive Control for LTI Systems

# Contents map

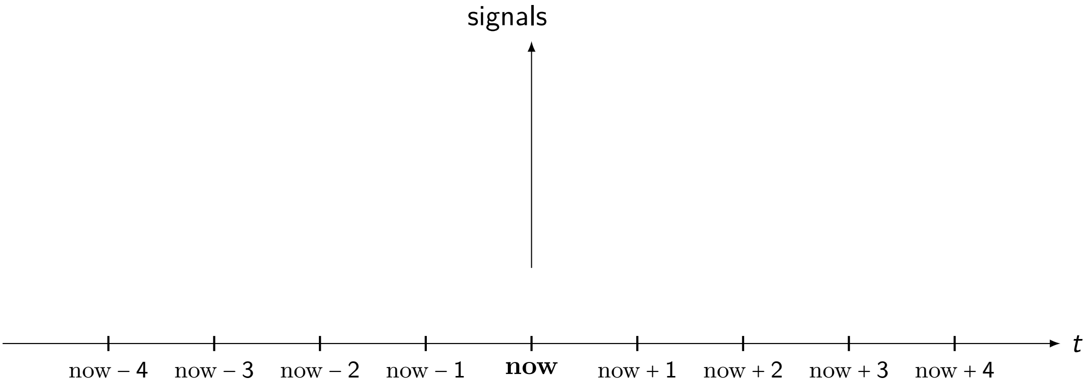
| developed content units | taxonomy levels |
|---|---|
| tuning MPC | u2, e2 |

| prerequisite content units | taxonomy levels |
|---|---|
| MPC fundamentals | u1, e1 |
| LTI systems | u1, e1 |

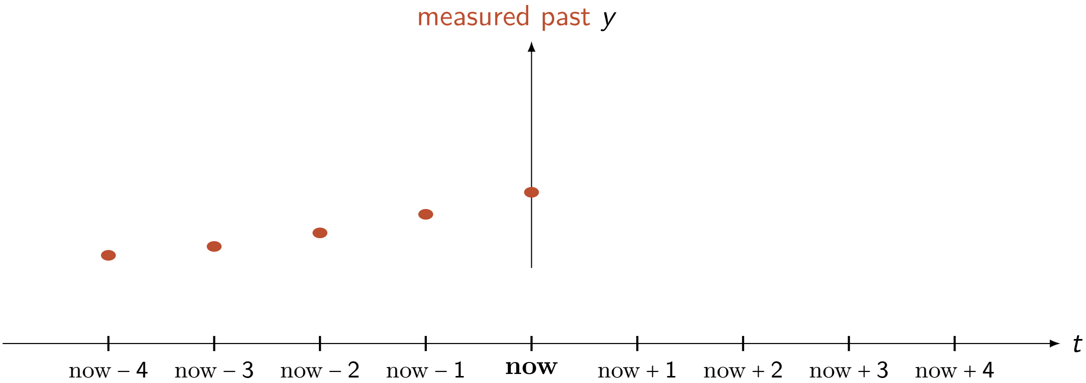
# Main ILO of sub-module "Tuning Model Predictive Control for LTI Systems"

> Design and tune an MPC controller for LTI systems to meet specified performance criteria

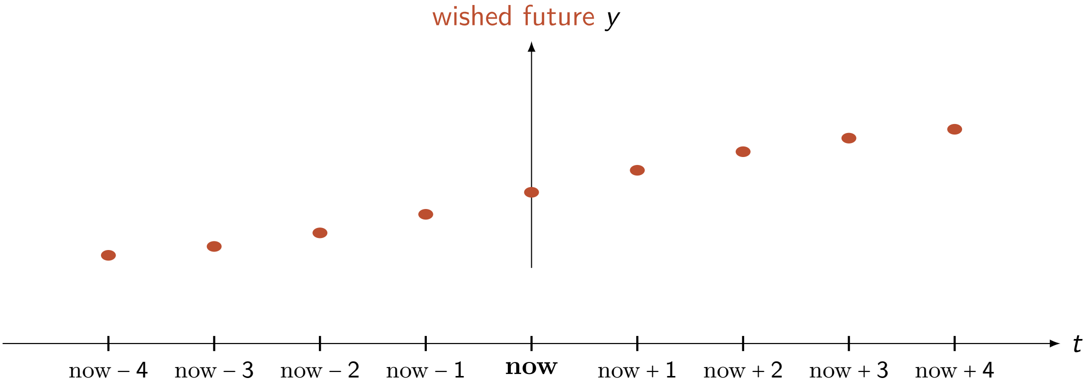
# The working principle, graphically



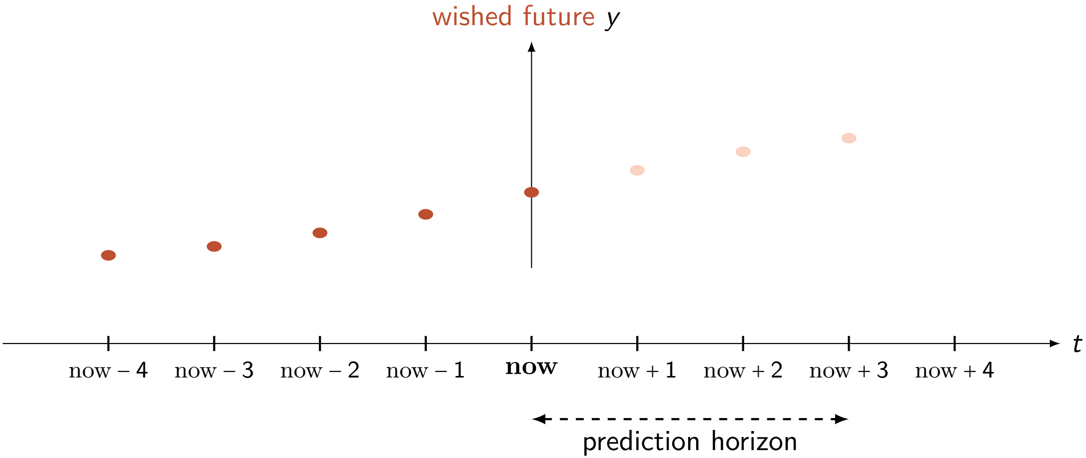
# The working principle, graphically



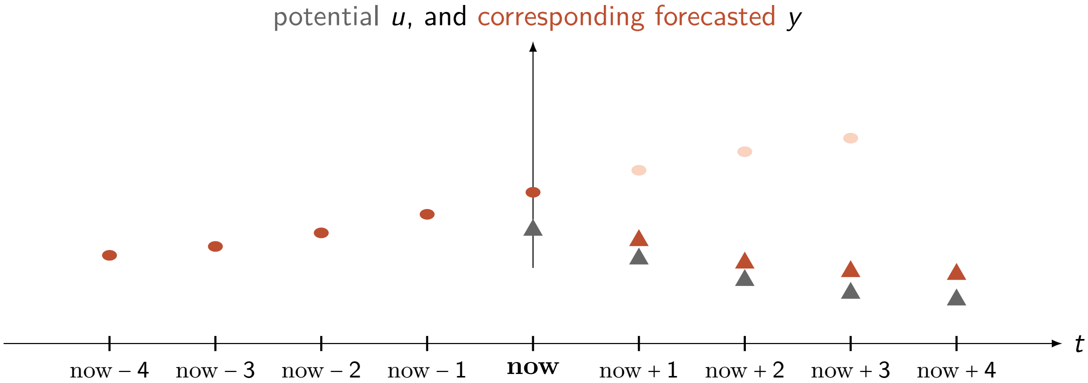
# The working principle, graphically



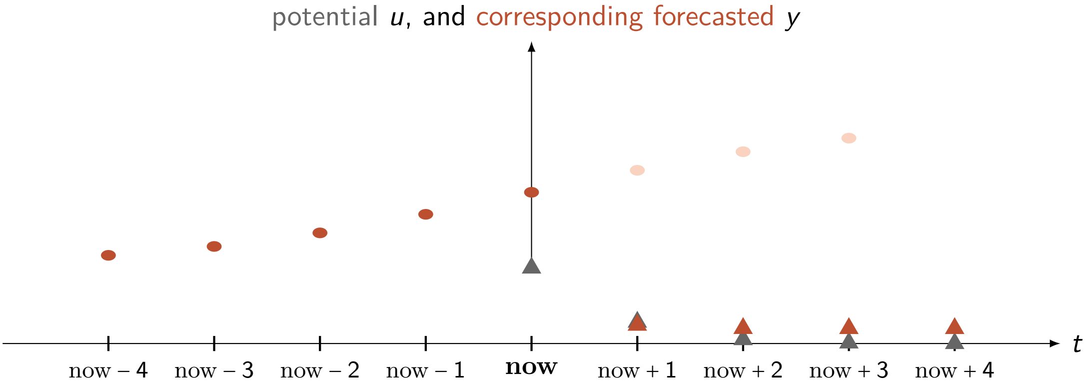
# The working principle, graphically



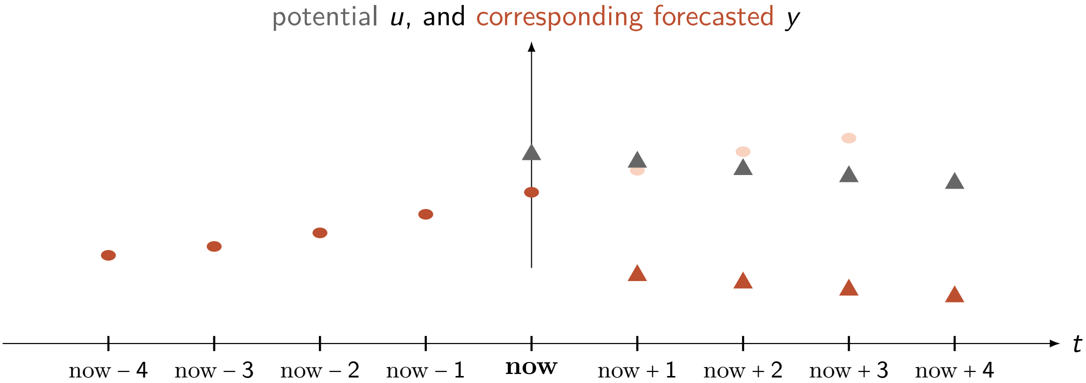
# The working principle, graphically



# The working principle, graphically

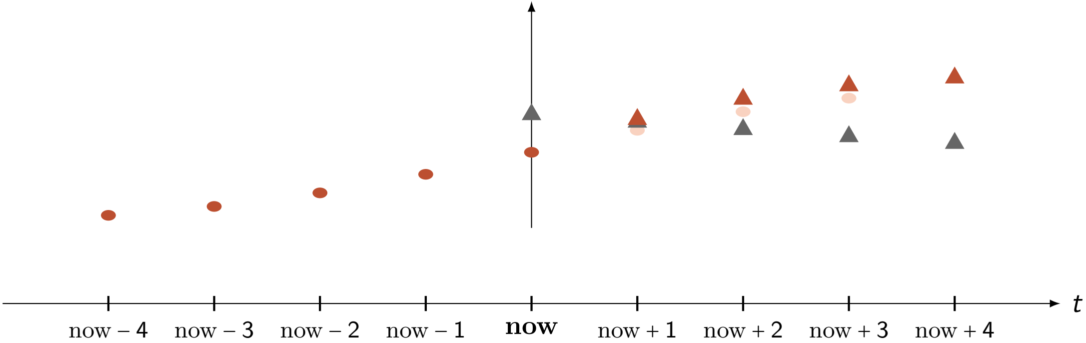


# The working principle, graphically

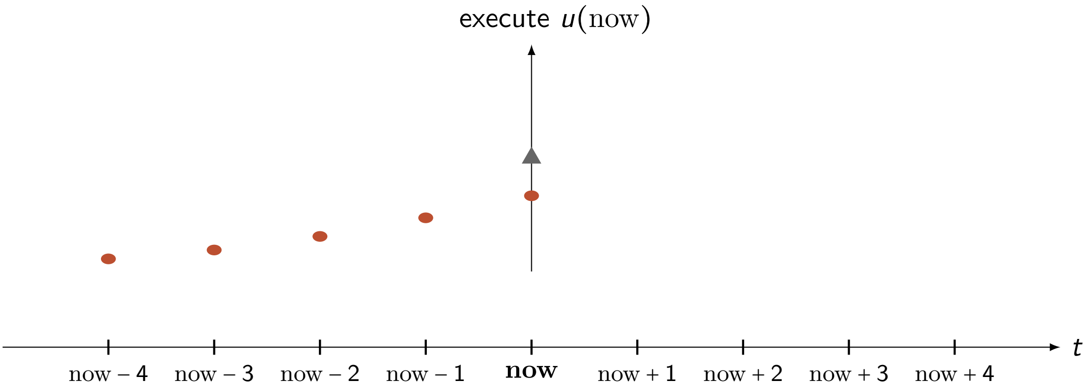


# The working principle, graphically

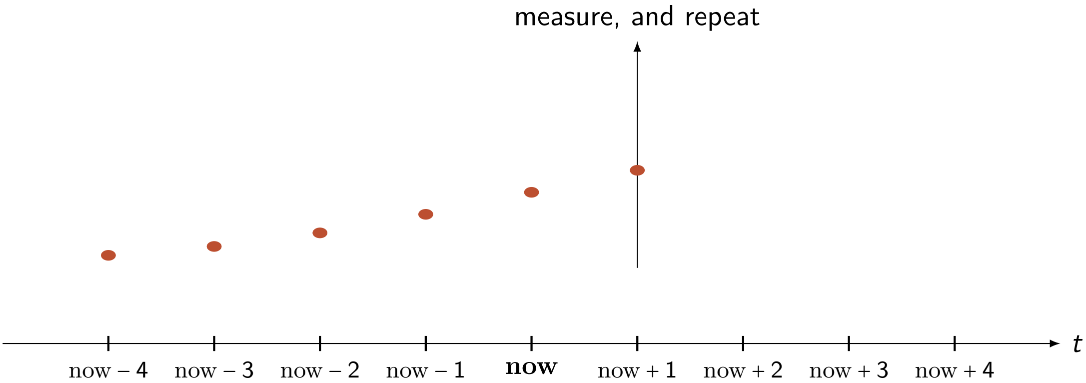
best $u$ = "best" forecasted $y$



# The working principle, graphically



# The working principle, graphically



# MPC in formulas

## (for LTI systems)

assumed dynamics: $x_{k+1} = Ax_k + Bu_k$

optimization problem:

$$\min_{u[0],\ldots,u[N]} \sum_{k=0}^{N-1} \underbrace{x[k]^T Q x[k]}_{\text{state cost}} + \underbrace{u[k]^T R u[k]}_{\text{control cost}} + \underbrace{x[N]^T P x[N]}_{\text{terminal cost}}$$

$$\text{s.t.} \quad x_{k+1} = Ax[k] + Bu[k] \quad \forall k \in \{0, \ldots, N-1\}$$

$$u_{\min} \leq u[k] \leq u_{\max}$$

$$x_{\min} \leq x[k] \leq x_{\max}$$

$$x[0] = x(t) \quad \text{(initial condition)}$$

# Key parameters

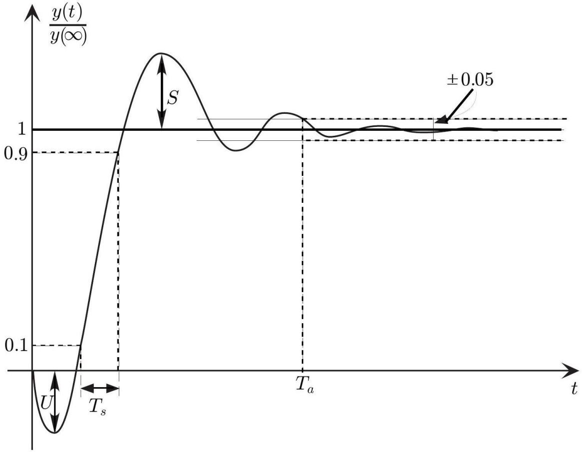
$$\min_{u[0],\ldots,u[N]} \sum_{k=0}^{N-1} \underbrace{x[k]^T Q x[k]}_{\text{state cost}} + \underbrace{u[k]^T R u[k]}_{\text{control cost}} + \underbrace{x[N]^T P x[N]}_{\text{terminal cost}}$$

- prediction horizon $N$
- weight matrices $Q$, $R$, $P$
- the constraints parameters $u_{\text{min}}, u_{\text{max}}, x_{\text{min}}, x_{\text{max}}$

# But which performance criteria shall we optimize?

Standard options:

- settling time
- overshoot
- control effort
- robustness
- computational efficiency



# General trade-offs

$$\min_{u[0],\ldots,u[N]} \sum_{k=0}^{N-1} \underbrace{x[k]^T Q x[k]}_{\text{state cost}} + \underbrace{u[k]^T R u[k]}_{\text{control cost}} + \underbrace{x[N]^T P x[N]}_{\text{terminal cost}}$$

- $\uparrow N \implies$ better performance but more computations
- $\uparrow Q \implies$ faster state convergence but more aggressive control
- $\uparrow R \implies$ smoother control but slower response

# Tuning methodology

$$\min_{u[0],\ldots,u[N]} \sum_{k=0}^{N-1} \underbrace{x[k]^T Q x[k]}_{\text{state cost}} + \underbrace{u[k]^T R u[k]}_{\text{control cost}} + \underbrace{x[N]^T P x[N]}_{\text{terminal cost}}$$

at every iteration, evaluate the performance and iteratively refine the parameters

- start with the infinite horizon equivalent (i.e., LQR)
- move to a shorter prediction horizon (5-20 samples)
- then adjust the weights ($Q$ first, then $R$)

# Summarizing

Design and tune an MPC controller for LTI systems to meet specified performance criteria

- determining performance requirements require simulating and evaluating to iteratively refine parameters

# Most important python code for this sub-module

# Model predictive control python toolbox

`https://www.do-mpc.com/en/latest/`

# Self-assessment material

# Question 1

What is the primary effect of increasing the Q matrix in MPC tuning?

**Potential answers:**

I: Reduced computational requirements
II: Smoother control actions
III: Faster state convergence
IV: Increased robustness to disturbances

# Question 2

What is the fundamental purpose of the terminal cost (P) in MPC?

**Potential answers:**

I: To reduce the computational complexity of the optimization
II: To ensure stability by approximating infinite horizon behavior
III: To enforce hard constraints on the system states
IV: To prioritize certain states over others in the transient response
V: I do not know

# Question 3

Why might increasing the prediction horizon N improve controller performance?

**Potential answers:**

I: It allows using larger Q matrices in the cost function
II: The controller can account for longer-term system behavior
III: It reduces the need for state constraints
IV: It makes the optimization problem convex
V: I do not know

# Question 4

What is the primary consequence of setting $R = 0$ in the MPC cost function?

**Potential answers:**

I: The controller will become unstable
II: The state constraints will be ignored
III: The controller may use arbitrarily large control inputs
IV: The prediction horizon becomes irrelevant
V: I do not know

# Question 5

Which of these represents a fundamental trade-off in MPC tuning?

**Potential answers:**

I: Between continuous-time and discrete-time formulations
II: Between state estimation and control computation
III: Between performance and computational complexity
IV: Between linear and nonlinear system models
V: I do not know

# Question 6

What is the main advantage of MPC compared to LQR control?

**Potential answers:**

I: MPC always requires less computational power
II: MPC guarantees global optimality for nonlinear systems
III: MPC can explicitly handle state and input constraints
IV: MPC doesn't require a system model
V: I do not know

# Recap of sub-module “Tuning MPC for LTI Systems”

- MPC performance depends on careful parameter selection
- Prediction horizon affects stability and computation
- Weight matrices balance state vs control objectives
- Systematic tuning follows an iterative procedure