Lecture 10: More Model Predictive Control (MPC)

- Recap: Model Predictive Control (MPC)
- Recap: Feasibility&stability
- Output feedback
- Target calculation
- Offset-free MPC (integral action in MPC)

Reference: B&H Ch. 4.2.3-4.2.4

Linear MPC; open loop dynamic optimization

$$\min_{z \in \mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2} x_{t+1}^\top Q_{t+1} x_{t+1} + d_{xt+1} x_{t+1} + \frac{1}{2} u_t^\top R_t u_t + d_{ut} u_t + \frac{1}{2} \Delta u_t^\top S \Delta u_t$$

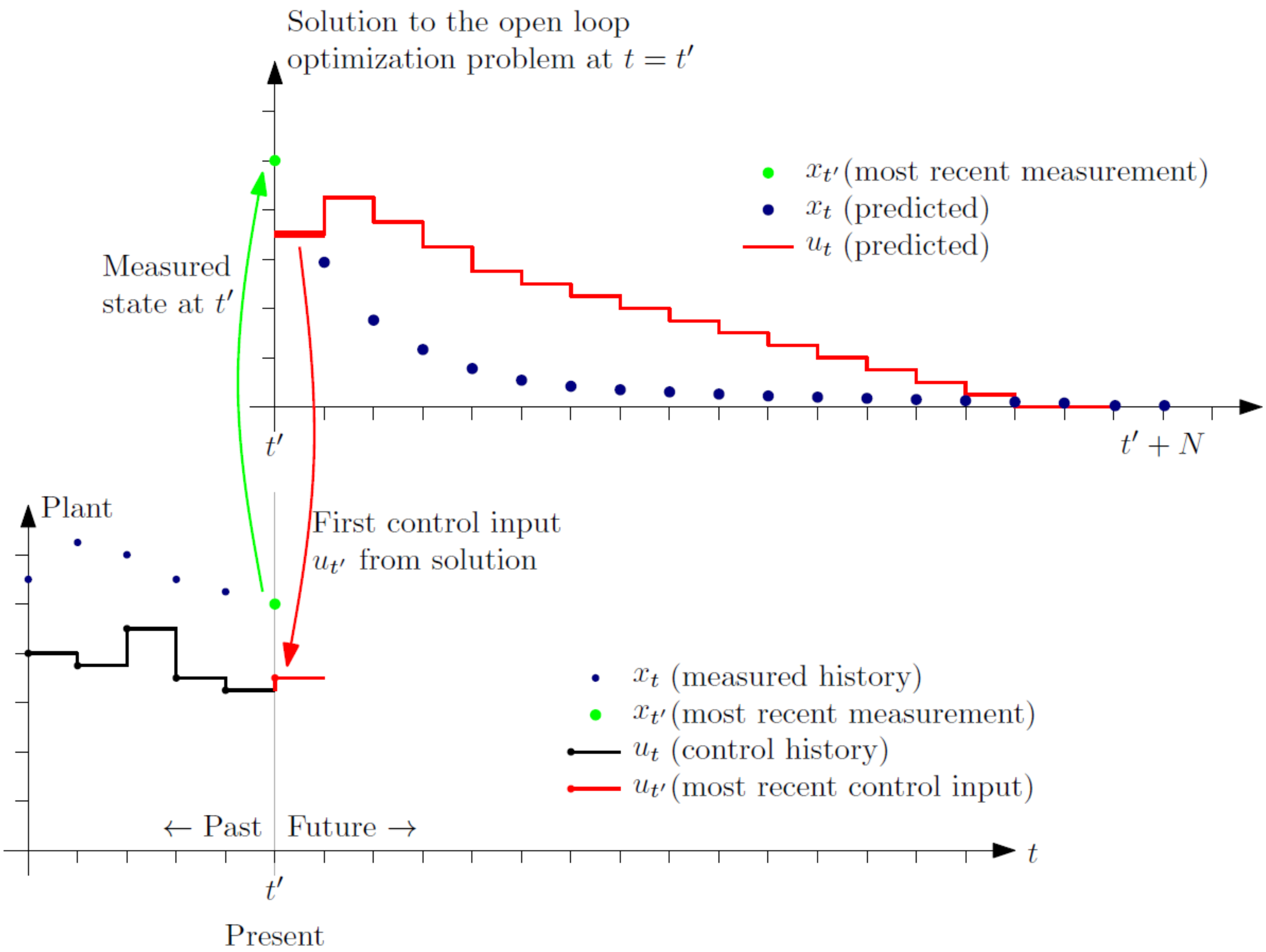
subject to

$$\begin{aligned}
x_{t+1} &= A_t x_t + B_t u_t, \quad t = \{0, \dots, N-1\} \\
x^{\text{low}} &\leq x_t \leq x^{\text{high}}, \quad t = \{1, \dots, N\} \\
u^{\text{low}} &\leq u_t \leq u^{\text{high}}, \quad t = \{0, \dots, N-1\} \\
-\Delta u^{\text{high}} &\leq \Delta u_t \leq \Delta u^{\text{high}}, \quad t = \{0, \dots, N-1\} \\
Q_t &\succeq 0 \quad t = \{1, \dots, N\} \\
R_t &\succeq 0 \quad t = \{0, \dots, N-1\}
\end{aligned}$$

where

$$\begin{aligned}
x_0 &\text{ and } u_{-1} \text{ is given} \\
\Delta u_t &:= u_t - u_{t-1} \\
z^\top &:= (u_0^\top, x_1^\top, \dots, u_{N-1}^\top, x_N^\top) \\
n &= N \cdot (n_x + n_u)
\end{aligned}$$

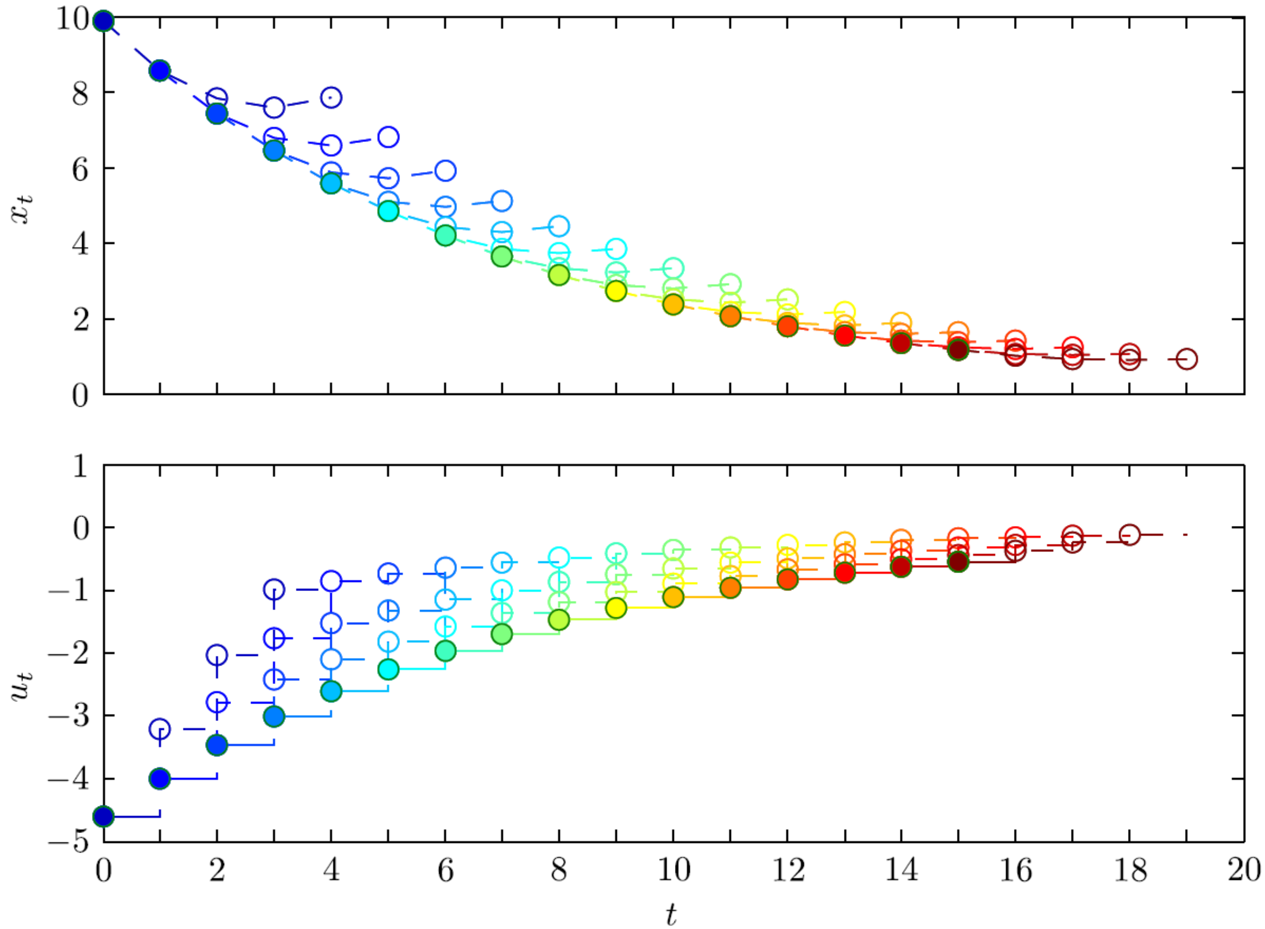
Model predictive control principle



Open-loop vs closed-loop

$$\min \sum_{t=0}^{4} x_{t+1}^2 + 4\, u_t^2$$

$$\text{s.t.} \qquad x_{t+1} = 1.2 x_t + 0.5 u_t, \quad t = 0, \dots, 3$$



MPC and feasibility

Is there always a solution to the MPC open-loop optimization problem?

- Not necessarily – state (or output) constraints may become infeasible, for example after a disturbance
- Practical solution: Soft constraints (or “exact penalty” formulations)
  - “Soften” state constraints by adding “slack variables”

$$
\begin{aligned}
\min_{z \in \mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2} x_{t+1}^\top Q_{t+1} x_{t+1} + \frac{1}{2} u_t^\top R_t u_t + \rho^\top \epsilon \\
\text{s.t.} \quad x_{t+1} = A_t x_t + B_t u_t, \quad t = \{0, \dots, N-1\} \\
x^{\text{low}} - \epsilon \leq x_t \leq x^{\text{high}} + \epsilon, \quad t = \{1, \dots, N\}, \qquad \epsilon > 0 \\
\vdots
\end{aligned}
$$

  - Soft constraints complicates stability theory
- Side-remark: In MPC stability theory, “recursive feasibility” is an important concept (when you don’t have soft constraints)
  - Recursive feasibility: feasibility now implies feasibility in the future
  - In general, proving recursive feasibility is similar to proving stability

MPC optimality implies stability?

$$\min \sum_{t=0}^{1} x_{t+1}^2 + r\, u_t^2$$

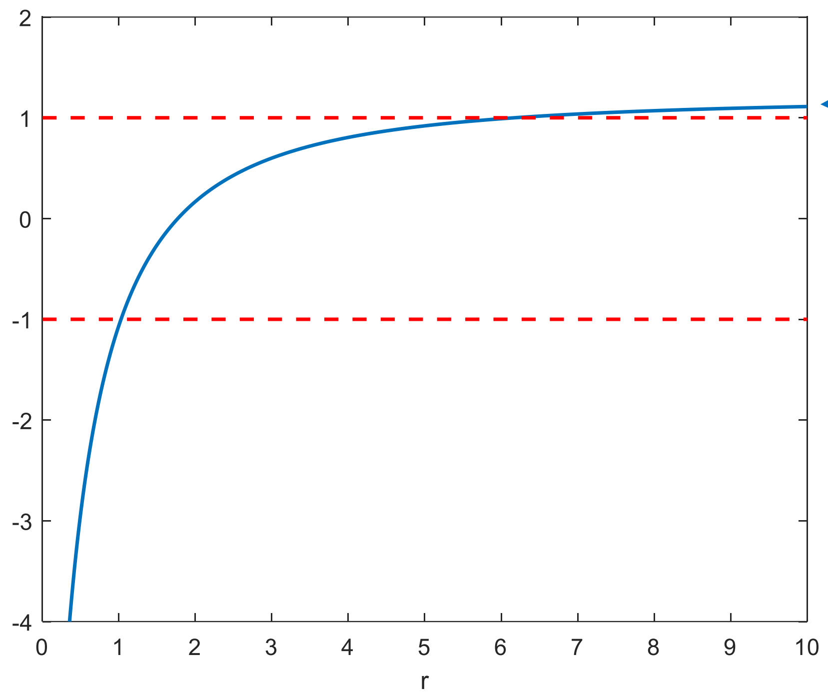
$$\text{s.t.} \quad x_{t+1} = 1.2 x_t + u_t, \quad t = 0, 1$$

MPC solution

$$u_t = -\frac{1.2 + 2.64r}{1 + 3.2r + r^2} x_t$$

MPC closed loop

$$x_{t+1} = \underbrace{\left(1.2 - \frac{1.2 + 2.64r}{1 + 3.2r + r^2}\right)} x_t$$



MPC and stability

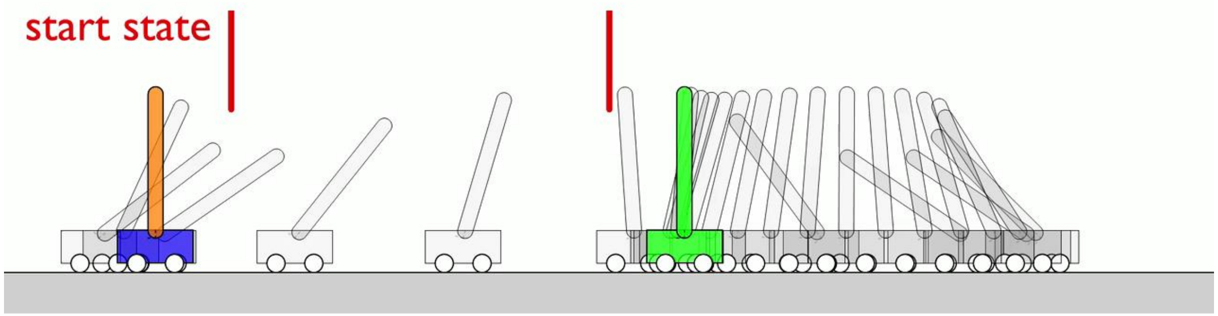
Requirements for stability:

- Stabilizability ((A,B) stabilizable)
- Detectability ((A,D) detectable)
  - D is a matrix such that $Q = D^\top D$ (that is, "D is matrix square root of Q")
  - Detectability: No modes can grow to infinity without being "visible" through Q

How to design MPC schemes with guaranteed *nominal* stability:

- Choose prediction horizon equal to infinity (usually not possible)
- Change the optimization problem (add terminal cost/terminal constraints) such that
  - The new problem is an "upper approximation" of infinite horizon problem
  - The constraints holds after the prediction horizon
- For given N, choose Q and R such that MPC is stable (cf. example)
  - Difficult, and not always possible!
- Typically, in practice: Choose N "large enough"
  - Usually works; since for MPC, plant is often open loop stable
  - What is "large enough"? Longer than dominating dynamics, but shorter can be OK

Offset-free MPC (or MPC with integral action)



From description:

- MPC with nonlinear model and a linear (input) disturbance model with one disturbance state: $x_t = f(x_t,u_t)+B_d\, d_t$. All states are measured ($y_t = x_t$).
- A linear observer is designed as a steady-state Kalman filter for the linearized augmented model at the final equilibrium.
- The forward-looking nature of the MPC controller allows to react to disturbances by considering obstacles in the environment and drastic replanning when necessary.

- From “Offset-free MPC explained: novelties, subtleties, and applications” - G. Pannocchia, M. Gabiccini, A. Artoni, NMPC 2015 - Seville, Spain September 17 - 20, 2015.