

Lecture 8 - MPC under uncertainty

Dinesh Krishnamoorthy

Department of Engineering Cybernetics, NTNU

Fall 2026

Introduction

- ▶ MPC requires a prediction model. But models are far from perfect.
- ▶ Disturbance and model mismatch can cause constraint violations, suboptimal performance, or instabilities.

Uncertain linear system evolution

Consider the linear system with bounded additive uncertainty

$$x^+ = Ax + Bu + w, \quad w \in \mathcal{W}$$

Given the current state x_0 , where can the state be k steps in the future?

$$x_1 = Ax_0 + Bu_0 + w_0$$

$$x_2 = A^2x_0 + ABu_0 + Bu_1 + Aw_0 + w_1$$

$$x_3 = A^3x_0 + A^2Bu_0 + ABu_1 + Bu_2 + A^2w_0 + Aw_1 + w_2$$

$$\vdots$$

$$x_k = A^k x_0 + \sum_{i=0}^{k-1} A^i B u_{k-1-i} + \sum_{i=0}^{k-1} A^i w_{k-1-i}$$

Uncertain evolution is the nominal system + offset caused by the disturbance.

Uncertain linear system evolution

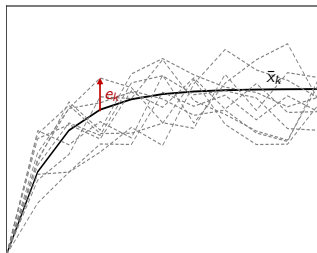
Consider the nominal system $\bar{x}^+ = A\bar{x} + Bu$ alongside the real system $x^+ = Ax + Bu + w$, and define the error $e = x - \bar{x}$

$$e_1 = A(x_0 - \bar{x}_0) + Bu_0 - Bu_0 + w_0 = Ae_0 + w_0$$

$$e_2 = Ae_1 + w_1 = A^2e_0 + Aw_0 + w_1$$

$$\vdots$$

$$e_k = A^k e_0 + \sum_{i=0}^{k-1} A^i w_{k-1-i}$$



Disturbance reachable set

- ▶ Define the linear transform $A\mathcal{W} := \{Aw, \quad \forall w \in \mathcal{W}\}$
- ▶ Assume x_0 is known, i.e., $e_0 = 0$
- ▶ The error at time step k lies in the disturbance reachable set

$$\mathcal{E}_k := \mathcal{W} \oplus A\mathcal{W} \oplus A^2\mathcal{W} \oplus \dots \oplus A^{k-1}\mathcal{W}$$

Idea! Precompute $\mathcal{E}$ and use it to define a tightened constraint set.

Tightened Constraint set

If nominal MPC is robust w.r.t. to the tightened constraint set, then the uncertain system will be feasible for all possible disturbance realizations.

To tighten constraints, we need the *Pontryagin difference* (a.k.a. Minkowski difference). For two sets $\mathcal{A}$ and $\mathcal{B}$, it is defined by

$$\mathcal{A} \ominus \mathcal{B} := \{a | a + b \in \mathcal{A}, \forall b \in \mathcal{B}\}$$

At each times step k , we impose the tightened constraint

$$x_k \in \mathcal{X} \ominus \mathcal{E}_k$$

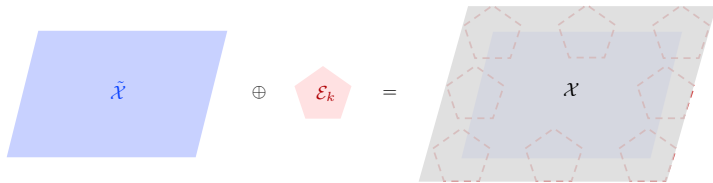
or in other words, the Minkowski sum of DRS $\mathcal{E}_k$ and the tightened constraint set $\tilde{\mathcal{X}}$ should result in the original constraint set $\mathcal{X}$

$$\tilde{\mathcal{X}} \oplus \mathcal{E} = \mathcal{X}$$

Tightened constraint set adds back-off effect

Tightened constraint set + Disturbance reachable set = Constraint set

$$\tilde{\mathcal{X}} \oplus \mathcal{E}_k = \mathcal{X}$$

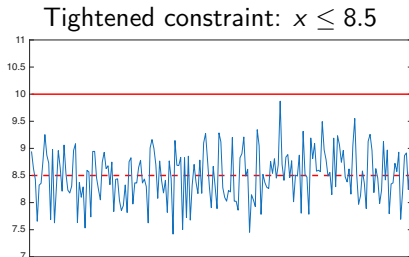
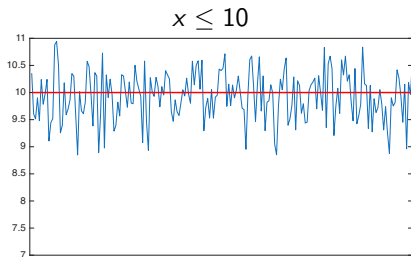


Use $\tilde{\mathcal{X}}$ in the MPC formulation!

Tightened constraint set adds back-off effect

Tightened constraint set + Disturbance reachable set = Constraint set

$$\tilde{\mathcal{X}} \oplus \mathcal{E}_k = \mathcal{X}$$



Use $\tilde{\mathcal{X}}$ in the MPC formulation!

Robust open-loop MPC

Solve the MPC problem, but with the tightened constraint set!

- Precompute DRS $\mathcal{E}_k$
- Tighten the constraint set:
 $\tilde{\mathcal{X}}_k = \mathcal{X} \ominus \mathcal{E}_k$
- Use this in the MPC formulation

$$\begin{aligned} \min \quad & \sum_{k=0}^{N-1} \ell(\bar{x}_k, u_k) + V_f(\bar{x}_N) \\ \text{s.t.} \quad & \bar{x}_{k+1} = A\bar{x}_k + Bu_k \\ & \bar{x}_k \in \tilde{\mathcal{X}}_k, \quad u_k \in \mathcal{U} \\ & \bar{x}_0 = x(t) \\ & \bar{x}_N \in \tilde{\mathcal{X}}_f \end{aligned}$$

Uses the nominal dynamics $\bar{x}_{k+1} = A\bar{x}_k + Bu_k$ as prediction model, but relies on the tightened constraints to ensure the actual state satisfies $x(t) \in \mathcal{X}$.

What about terminal set ?

- ▶ For recursive feasibility, we need terminal set to be invariant under some stabilizing control law K .
- ▶ In the presence of uncertainty, terminal set must be **Robust positive invariant (RPI)**, i.e., we need to ensure that the state will remain in $\mathcal{X}_f$ regardless of the disturbance realization.

A set $\mathcal{X}_f$ is said to be RPI for $x^+ = Ax + Bu + w$ with $w \in \mathcal{W}$ if

$$Ax + Bu + w \in \mathcal{X}_f, \forall (x, w) \in \mathcal{X}_f \times \mathcal{W}$$

Given a set $\mathcal{X}_f$ and closed-loop dynamics $x^+ = (A + BK)x + w$ for some stabilizing K , the robust pre-set of $\mathcal{X}_f$ is the set of states that evolve into $\mathcal{X}_f$ after one step regardless of the disturbance realization, i.e.,

$$\text{pre}(\mathcal{X}_f; \mathcal{W}) := \{x \mid (A + BK)x + w \in \mathcal{X}_f, \forall w \in \mathcal{W}\}$$

What about performance under uncertainty?

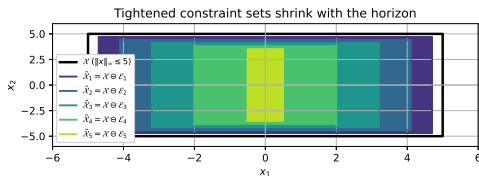
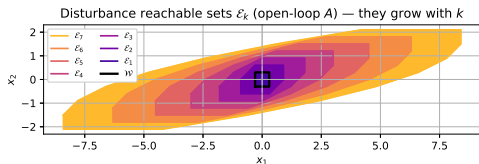
Tightened constraint set ensures robust constraint satisfaction, but what about performance?

- ▶ Nominal cost: $\min_u \ell(\bar{x}_k, u_k)$
- ▶ Expected cost: $\min_u \ell(\mathbb{E}[x_k], u_k)$
- ▶ Worst case cost: $\min_u \max_w \ell(x_k, u_k)$

Example

$$x_{k+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_k + w, \quad w \in \mathcal{W}$$

with $\mathcal{W} = \{w : \|w\|_\infty \leq 0.3\}$, $\mathcal{X} = \{x : \|x\|_\infty \leq 5\}$, $\mathcal{U} = [-1, 1]$,



Largest robustly feasible horizon: $N = 5$

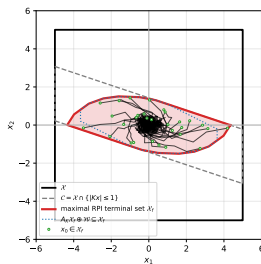
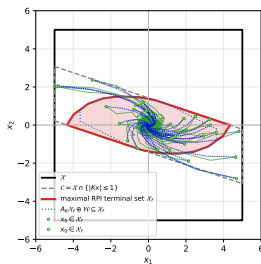
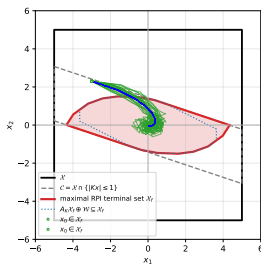
Example

- Nominal Terminal set = based on LQR control law K

$$\mathcal{X}_f := \{x \in \mathcal{X} : Kx \in [-1, 1]\}$$

- Terminal set under uncertainty = Maximal RPI set

$$\tilde{\mathcal{X}}_f = \text{pre}(\mathcal{X}_f; \mathcal{W}) := \{x \mid (A + BK)x + w \in \mathcal{X}_f, \forall w \in \mathcal{W}\}$$



Conclusion

- ▶ Robust *open-loop* MPC with tightened constraint set and an RPI terminal set is recursively feasible.
- ▶ Robust *open-loop* MPC falls apart after a few steps because the open-loop $\mathcal{E}_k$ explodes.
- ▶ Tightened constraint sets $\tilde{\mathcal{X}} = \emptyset$ for large $N \Rightarrow$ MPC becomes infeasible.
- ▶ Robust *open-loop* MPC potentially has a very small region of attraction, particularly for systems where A is unstable.

This is because we are using **open-loop** optimization.

Open-loop optimization: We choose the optimal trajectory u_k based on the nominal system dynamics $\bar{x}_k$ rather than the actual state $x_k = \bar{x}_k + e_k$

Fix: **Closed-loop optimization:** Put the feedback inside the prediction.

Open-loop is conservative: we threw away feedback

Robust *open-loop* MPC optimizes a single input sequence $u = (u_0, \dots, u_{N-1})$ that must work for *every* disturbance realization.

- ▶ The controller cannot **react**: it commits to u_k before seeing $w_0, \dots, w_{k-1}$.
- ▶ This is why $\mathcal{E}_k$ grew and $\mathcal{X} \ominus \mathcal{E}_k$ emptied out after a few steps.

The real controller *will* measure the state and act on it. So we should optimize over **feedback policies** $\pi_k(\cdot)$, not over a fixed sequence:

$$\pi_k : \mathcal{X} \rightarrow \mathcal{U}, \quad u_k = \pi_k(x_k).$$

Closed-loop (feedback) optimization: optimize over *control laws*, so the input can adapt to whatever disturbance actually occurs.

Closed-loop Optimization: correct, but intractable

The min-max problem over feedback policies:

$$\begin{aligned} \min_{\pi_0, \dots, \pi_{N-1}} \max_{\mathbf{w}_0, \dots, \mathbf{w}_{N-1} \in \mathcal{W}} & \sum_{k=0}^{N-1} \ell(x_k, \pi_k(x_k)) + V_f(x_N) \\ \text{s.t. } & x_{k+1} = Ax_k + B\pi_k(x_k) + \mathbf{w}_k \\ & x_k \in \mathcal{X}, \pi_k(x_k) \in \mathcal{U} \quad \forall \mathbf{w}_0, \dots, \mathbf{w}_{k-1} \in \mathcal{W} \end{aligned}$$

- ▶ The decision variables π_k are **functions** $\Rightarrow$ infinite-dimensional optimization.
- ▶ Exact solution = robust dynamic programming: curse of dimensionality, intractable in general.
- ▶ Constraints must hold for *all* disturbance sequences (a set that grows with N).

Optimizing over *arbitrary* policies is intractable.

Idea: restrict the policy to a simple, fixed structure we *can* optimize.

Recall from Lecture 1: Exact DP - Illustrative Example

- Consider a system

$$x_{k+1} = x_k + u_k$$

with $x_k \in \mathbb{R}$ and $u_k \in \mathbb{R}$

- OCP:

$$\min \sum_{k=0}^{N-1} (x_k^2 + u_k^2) + x_N^2$$

with $N = 3$

Exact DP - Illustrative Example

- ▶ Consider a system $x_{k+1} = x_k + u_k$ with $x_k \in \mathbb{R}$, $u_k \in \mathbb{R}$
- ▶ OCP: $\min \sum_{k=0}^{N-1} (x_k^2 + u_k^2) + x_N^2$, with $N = 3$

$$J_k^*(x_k) = \min_{u_k} \ell(x_k, u_k) + J_{k+1}^*(x_{k+1})$$

at stage $k = 2$, begin with $J(3) = x_3^2$

$$\begin{aligned} J_2(x_2) &= x_2^2 + u_2^2 + x_3^2 \\ &= x_2^2 + u_2^2 + (x_2 + u_2)^2 \\ \nabla_{u_2} J_2(x_2) &= 2u_2 + 2(x_2 + u_2) = 0 \\ \Rightarrow u_2^* &= -0.5x_2, \quad J_2^*(x_2) = 1.5x_2^2 \end{aligned}$$

Exact DP - Illustrative Example

- ▶ Consider a system $x_{k+1} = x_k + u_k$ with $x_k \in \mathbb{R}$, $u_k \in \mathbb{R}$
- ▶ OCP: $\min \sum_{k=0}^{N-1} (x_k^2 + u_k^2) + x_N^2$, with $N = 3$

$$J_k^*(x_k) = \min_{u_k} \ell(x_k, u_k) + J_{k+1}^*(x_{k+1})$$

$$\text{at stage } k = 2 \Rightarrow u_2^* = -0.5x_2, \quad J_2^*(x_2) = 1.5x_2^2$$

at stage $k = 1$

$$\begin{aligned} J_1(x_1) &= x_1^2 + u_1^2 + J_2^*(x_2) \Rightarrow x_1^2 + u_1^2 + 1.5x_2^2 \\ &= x_1^2 + u_1^2 + 1.5(x_1 + u_1)^2 \\ \nabla_{u_1} J_1(x_1) &= 2u_1 + 3(x_1 + u_1) = 0 \\ &\Rightarrow u_1^* = -3/5x_1, \quad J_1^*(x_1) = 8/5x_1^2 \end{aligned}$$

Exact DP - Illustrative Example

- ▶ Consider a system $x_{k+1} = x_k + u_k$ with $x_k \in \mathbb{R}$, $u_k \in \mathbb{R}$
- ▶ OCP: $\min \sum_{k=0}^{N-1} (x_k^2 + u_k^2) + x_N^2$, with $N = 3$

$$J_k^*(x_k) = \min_{u_k} \ell(x_k, u_k) + J_{k+1}^*(x_{k+1})$$

$$\text{at stage } k = 2 \Rightarrow u_2^* = -0.5x_2, \quad J_2^*(x_2) = 1.5x_2^2$$

$$\text{at stage } k = 1 \Rightarrow u_1^* = -3/5x_1, \quad J_1^*(x_1) = 8/5x_1^2$$

at stage $k = 0$

$$\begin{aligned} J_0(x_0) &= x_0^2 + u_0^2 + J_1^*(x_1) \Rightarrow x_0^2 + u_0^2 + 8/5x_1^2 \\ &= x_0^2 + u_0^2 + 8/5(x_0 + u_0)^2 \\ \nabla_{u_0} J_0(x_0) &= 2u_0 + 16/5(x_0 + u_0) = 0 \\ &\Rightarrow u_0^* = -8/13x_0, \quad J_0^*(x_0) = 21/26x_0^2 \end{aligned}$$

Exact DP - Illustrative Example

- ▶ Consider a system $x_{k+1} = x_k + u_k$ with $x_k \in \mathbb{R}$, $u_k \in \mathbb{R}$
- ▶ OCP: $\min \sum_{k=0}^{N-1} (x_k^2 + u_k^2) + x_N^2$, with $N = 3$

$$J_k^*(x_k) = \min_{u_k} \ell(x_k, u_k) + J_{k+1}^*(x_{k+1})$$

$$\text{at stage } k = 2 \Rightarrow u_2^* = -0.5x_2, \quad J_2^*(x_2) = 1.5x_2^2$$

$$\text{at stage } k = 1 \Rightarrow u_1^* = -3/5x_1, \quad J_1^*(x_1) = 8/5x_1^2$$

$$\text{at stage } k = 0 \Rightarrow u_0^* = -8/13x_0, \quad J_0^*(x_0) = 21/26x_0^2$$

Go **backward** to compute the optimal cost, go **forward** to construct the optimal sequence

Open-loop optimization with $w_k = 0$

at stage $k = 0$

$$\Rightarrow u_0^* = -8/13x_0, \quad \Rightarrow x_1 = x_0 - 8/13x_0 = 5/13x_0$$

at stage $k = 1$

$$\Rightarrow u_1^* = -3/5x_1 = -3/13x_0, \quad \Rightarrow x_2 = 5/13x_0 - 3/13x_0 = 2/13x_0$$

at stage $k = 2$

$$\Rightarrow u_2^* = -0.5x_2 = -1/13x_0, \quad \Rightarrow x_3 = 2/13x_0 - 1/13x_0 = 1/13x_0$$

Feedback versus open-loop optimization

$$x_{k+1} = x_k + u_k + w_k$$

Open-loop optimization with $w_k = 0$

$$u^0(x_0) = [-(8/13)x_0, -(3/13)x_0, -(1/13)x_0]$$

$$x^0(x_0) = [x_0, (5/13)x_0, (2/13)x_0, (1/13)x_0]$$

Dynamic Programming

$$\mu := (\mu_0(\cdot), \mu_1(\cdot), \mu_2(\cdot))$$

$$\mu_0 = -(8/13)x_0 \Rightarrow x^0(0) = x_0$$

$$u^0(0) = -(8/13)x_0$$

$$\mu_1 = -(3/5)x_1 \Rightarrow x^0(1) = (5/13)x_0$$

$$u^0(1) = -(3/13)x_0$$

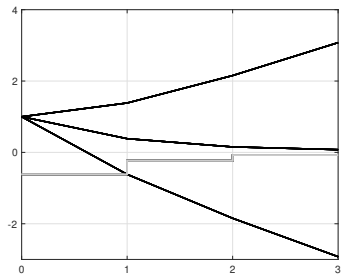
$$\mu_2 = -(1/2)x_2 \Rightarrow x^0(2) = (2/13)x_0$$

$$u^0(2) = -(1/13)x_0$$

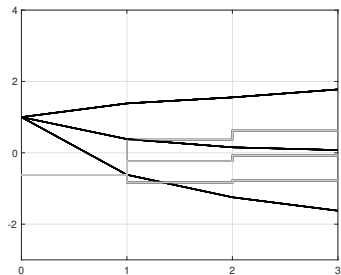
$$\Rightarrow x^0(3) = (1/13)x_0$$

Feedback versus open-loop optimization

Open-loop



Feedback - DP

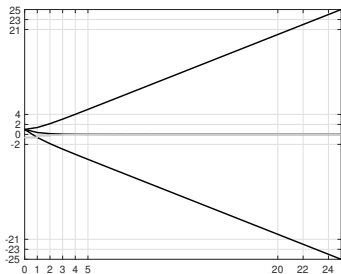


Feedback versus open-loop optimization

Open-loop

$$|x_{w=-1}(N) - x_{w=1}(N)| = 2N$$

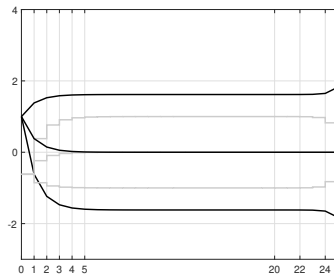
$N \rightarrow \infty$, error explodes



Feedback - DP

$$|x_{w=-1}(N) - x_{w=1}(N)| \rightarrow 3.24$$

$N \rightarrow \infty$, error stays bounded



Tube MPC: optimize over a restricted policy class

Restrict the feedback law to a pre-chosen **affine** form: pick a stabilizing gain K and optimize only the **nominal** part.

$$\pi_k(x) = \bar{u}_k + K(x - \bar{x}_k) \quad \implies \quad \underbrace{e = x - \bar{x}}_{\text{error}} : \quad e^+ = (A + BK)e + w$$

- ▶ K (fixed) provides the *feedback* $\Rightarrow$ the error stays in a bounded tube $\mathcal{E}_\infty$, independent of the optimization.
- ▶ We only optimize the **nominal** sequence $(\bar{x}_k, v_k) \Rightarrow$ back to a *finite-dimensional, convex* problem.
- ▶ The reaction to disturbances is baked into K , so we recover feedback without solving the intractable policy problem.

Tube MPC = closed-loop optimization over the affine policy class $\pi_k(x) = \bar{u}_k + K(x - \bar{x}_k)$: a tractable surrogate for optimizing over policies.

Tube-MPC for linear systems with additive disturbances

Key idea: Separate control authority into two parts:

1. $\bar{u}(t)$ that steers the nominal system $\bar{x}_{k+1} = A\bar{x}_k + B\bar{u}_k$ to the origin
2. compensate for deviations from this system, such that

$$u(t) = \bar{u}(t) + K(x(t) - \bar{x}(t)) = \bar{u}(t) + Ke(t)$$

with nominal control law K that stabilizes the nominal system

Error dynamics is given by:

$$\begin{aligned} e_{k+1} &= x_{k+1} - \bar{x}_{k+1} \\ &= Ax_k + Bu_k + w_k - A\bar{x}_k - B\bar{u}_k \\ &= Ax_k + B\bar{u}_k + BKe_k + w_k - A\bar{x}_k - B\bar{u}_k \\ &= Ae_k + BKe_k + w_k = (A + BK)e_k + w_k \\ \Rightarrow e^+ &= (A + BK)e + w \end{aligned}$$

Tube MPC for linear systems with additive disturbances

Given the error dynamics

$$e^+ = (A + BK)e + w$$

Uncertain evolution of the error is given by

$$e_k = (A + BK)^k e_0 + \sum_{i=0}^{k-1} (A + BK)^i w_{k-1-i}$$

Since $w \in \mathcal{W}$ and $(A + BK)$ is stable, there must be some **minimum robust positive invariant** set $\mathcal{E}_\infty$, such that $e_k \in \mathcal{E}_\infty$, for all $k \geq 0$.

$$\mathcal{E}_\infty = \bigoplus_{i=0}^{\infty} (A + BK)^i \mathcal{W}$$

Instead of tightening constraint with $\mathcal{E}_k = \bigoplus_{i=0}^k A^i \mathcal{W}$, tighten constraints with $\mathcal{E}_\infty = \bigoplus_{i=0}^{\infty} (A + BK)^i \mathcal{W}$

Minimum RPI set

How to compute $\mathcal{E}_\infty = \bigoplus_{i=0}^{\infty} (A + BK)^i \mathcal{W}$?

- Compute upto a finite s

$$\bigoplus_{i=0}^{s-1} (A + BK)^i \mathcal{W}$$

- Ensure $A + BK$ is stable
- Calculate the Remainder Bound

$$\frac{1}{1 - \alpha} (A + BK)^s \mathcal{W}, \alpha \in (0, 1]$$

Minimum RPI set is given by:

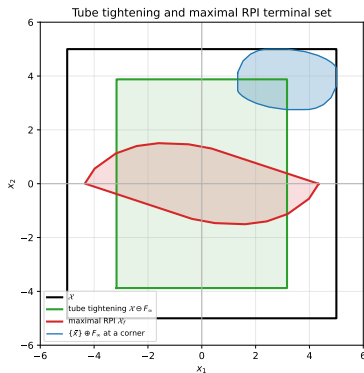
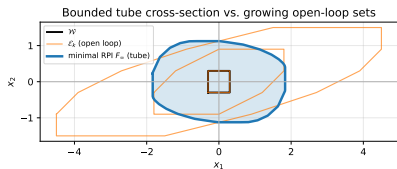
$$\mathcal{E}_\infty = \bigoplus_{i=0}^{s-1} (A + BK)^i \mathcal{W} + \frac{1}{1 - \alpha} (A + BK)^s \mathcal{W}$$

Tube MPC

New state constraints:

$$\bar{x}_k \oplus \mathcal{E}_\infty \in \mathcal{X}$$

i.e., tightened constraint set $\tilde{\mathcal{X}}_k = \mathcal{X} \ominus \mathcal{E}_\infty \forall k$

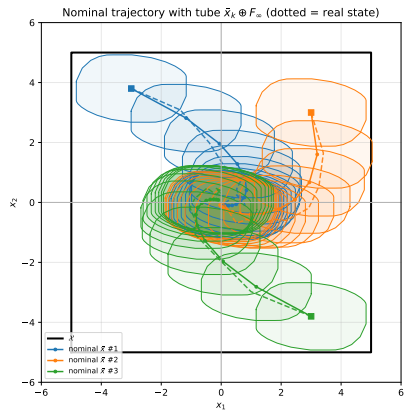
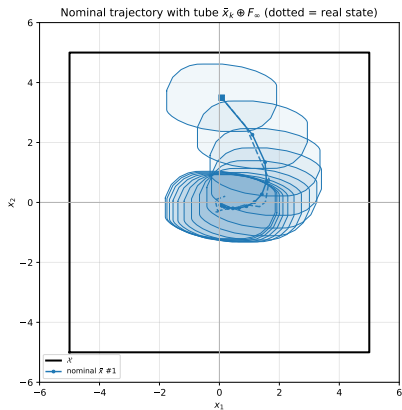


Tube MPC

$$\begin{aligned} \min_{\bar{x}_0, \bar{u}_0, \dots} \quad & \sum_{k=0}^{N-1} \ell(\bar{x}_k, \bar{u}_k) + V_f(\bar{x}_N) \\ \text{s.t.} \quad & \bar{x}_{k+1} = A\bar{x}_k + B\bar{u}_k && \text{(nominal dynamics)} \\ & \bar{x}_k \in \mathcal{X} \ominus \mathcal{E}_\infty && \text{(state constraint)} \\ & \bar{u}_k \in \mathcal{U} \ominus K\mathcal{E}_\infty && \text{(Input constraint)} \\ & \bar{x}_N \in \tilde{\mathcal{X}}_f && \text{(RPI terminal set)} \\ & x(t) \in \bar{x}_0 \oplus \mathcal{E}_\infty && \text{(Tightened feasible set)} \end{aligned}$$

- ▶ Starting the nominal states in $\mathcal{X} \ominus \mathcal{E}_\infty$ guarantees that the whole tube stays inside $\mathcal{X}$.
- ▶ K - the fixed ancillary (error-feedback) gain, $K(x - \bar{x})$, whose only job is to keep the error in the tube $\mathcal{E}_\infty$.
- ▶ nominal input $\bar{u}_k$ produced by the QP.

Example



Tube MPC: what it quietly assumes

Tube MPC is elegant, but it leans on three assumptions that are easy to miss.

- ▶ **Needs a disturbance set $\mathcal{W}$.** You must supply an *explicit, bounded* set $\mathcal{W}$ (usually a box/polytope). Choosing it is a modelling burden: too large $\Rightarrow$ conservative; too small $\Rightarrow$ not actually robust.
- ▶ **Additive uncertainty only.** The whole construction rests on $x^+ = Ax + Bu + w$, so the error obeys $e^+ = (A + BK)e + w$. **Parametric / multiplicative** uncertainty (uncertain A, B , model parameters) is *state- and input-dependent* and does *not* fit the additive $w \in \mathcal{W}$ template.
- ▶ **Worst-case, not probabilistic.** $\mathcal{W}$ is a hard bound; a rare large disturbance is treated like a typical one $\Rightarrow$ built-in conservatism.

Tube MPC: nonlinear systems break the machinery

For $x^+ = f(x, u) + w$ the linear tube tricks no longer hold:

- ▶ **No superposition.** The error $e = x - \bar{x}$ obeys $e^+ = f(x, u) - f(\bar{x}, \bar{u}) + w$, which depends on *where* you are $\Rightarrow$ no single, constant RPI cross-section $\mathcal{E}_\infty$.
- ▶ **Minkowski-sum reachability fails.** $\mathcal{W} \oplus A\mathcal{W} \oplus \dots$ relies on linear maps; there is no such closed form for a nonlinear f .
- ▶ **Ancillary K is not enough.** A fixed linear $e^+ = (A + BK)e + w$ cannot keep the error bounded globally; you need nonlinear / local controllers, contraction metrics, or incremental Lyapunov functions $\Rightarrow$ hard to design and often conservative.
- ▶ **Sets are no longer polytopes.** Robust invariant tubes for nonlinear systems are generally non-convex and expensive to compute and represent.

When does the tube construction fail?

The minimal RPI set $\mathcal{E}_\infty = \bigoplus_{i \geq 0} (A + BK)^i \mathcal{W}$ is only useful under conditions that are easy to violate:

- ▶ **$A + BK$ must be strictly stable.** If not, the sum does *not converge* $\Rightarrow \mathcal{E}_\infty$ is **unbounded**, not usable.
- ▶ **The tube must fit inside the constraints.** Even a bounded $\mathcal{E}_\infty$ is useless if it is too big, e.g.,

$$\mathcal{X} \ominus \mathcal{E}_\infty = \emptyset \quad \text{or} \quad \mathcal{U} \ominus K\mathcal{E}_\infty = \emptyset.$$

Then *no* nominal problem exists $\Rightarrow$ tube MPC is infeasible from the start.

- ▶ Large $\mathcal{W}$, slow/poorly-damped $A + BK$, or tight $\mathcal{X}, \mathcal{U}$ all push the tube past the constraints.

Tube MPC is a *pre-computed, worst-case, additive-linear* recipe. Once any of these breaks, we need a more flexible way to carry uncertainty through the optimization.

Motivation: from tubes to multi-stage MPC

Idea: don't bound uncertainty in a set and pre-tighten — **optimize over a discrete set of scenarios** directly (a scenario tree).

- ▶ **Nonlinear-friendly.** Each scenario is propagated through the *true* model $x^+ = f(x, u)$; no RPI sets, no Minkowski sums.
- ▶ **Beyond additive.** Scenarios can differ in *parameters* (A, B, θ) , not just an added $w \Rightarrow$ handles parametric / multiplicative uncertainty.
- ▶ **Less conservative (recourse).** Branching lets future inputs *adapt* after the uncertainty is revealed — closed-loop, not a single worst-case sequence.
- ▶ **No invariant-set computation.** Robustness comes from the tree, not from pre-computed tubes or terminal RPI sets.

Price to pay: the problem grows with the number of scenarios (and the robust horizon) — this trade-off is what multi-stage MPC is all about.

Multistage MPC approximation of DP

Ideal: Optimize over control policies

$$\boldsymbol{\mu} = (\mu_0, \mu_1, \mu_2 \dots, \mu_{N-1})$$

Sampled control policy

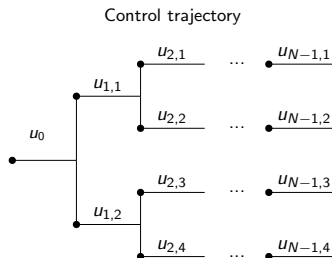
$$\mu_0 = \{u_0\}$$

$$\mu_1 = \{u_{1,1}, u_{1,2}\}$$

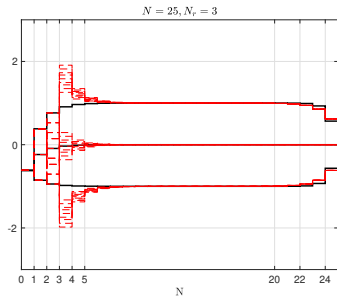
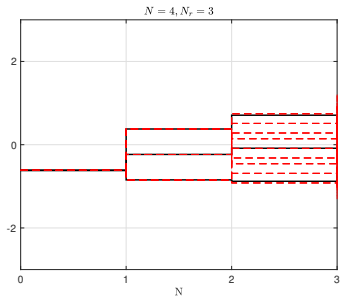
$$\mu_2 = \{u_{2,1}, u_{2,2}, u_{2,3}, u_{2,4}\}$$

$$\vdots$$

$$\mu_{N-1} = \{u_{N-1,1}, u_{N-1,2}, u_{N-1,3}, u_{N-1,4}\}$$

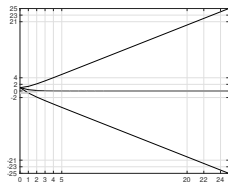


Multistage MPC approximation of DP



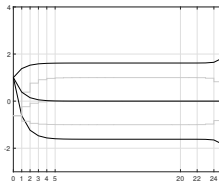
Feedback versus open-loop optimization

Open-loop



$$|x_{w=-1}(N) - x_{w=1}(N)| = 2N$$

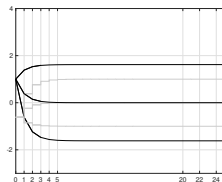
Feedback - DP



$$|x_{w=-1}(N) - x_{w=1}(N)| \rightarrow 3.24$$

$$N \rightarrow \infty$$

Feedback - multistage

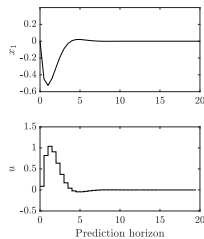


$$|x_{w=-1}(N) - x_{w=1}(N)| \rightarrow 3.24$$

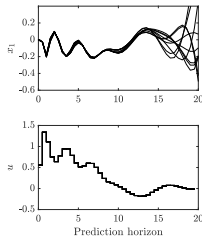
$$N \rightarrow \infty$$

Solution strategies

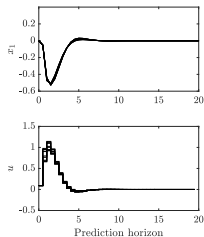
Perfect information



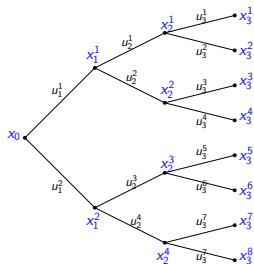
Open-loop optimization



Closed-loop optimization



Multistage scenario MPC



- ▶ Discrete scenario tree
- ▶ Introduces notion of feedback
- ▶ Not dependent on uncertainty description
- ▶ additive/multiplicative/parametric uncertainty
- ▶ tracking/economic costs
- ▶ Very practical formulation for MPC under uncertainty

Optimization over different control trajectories $\approx$ Optimization over control policies

Why Multistage MPC?

Certainty equivalence

$$\min_{\mathbf{x}_k, \mathbf{u}_k} \sum_{k=0}^{N-1} \ell(\mathbf{x}_k, \mathbf{u}_k)$$

s.t.

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{p}_0) \quad \forall k \in \mathcal{K}$$

$$\mathbf{x}_0 = \hat{\mathbf{x}}_t$$

$$\mathbf{x}_k \in \mathcal{X}, \quad \mathbf{u}_k \in \mathcal{U} \quad \forall k \in \mathcal{K}$$

Multistage MPC

$$\min_{\mathbf{x}_{k,j}, \mathbf{u}_{k,j}} \sum_{j=1}^S \omega_j \sum_{k=0}^{N-1} \ell(\mathbf{x}_{k,j}, \mathbf{u}_{k,j})$$

s.t

$$\mathbf{x}_{k+1,j} = \mathbf{f}(\mathbf{x}_{k,j}, \mathbf{u}_{k,j}, \mathbf{p}_{k,j}) \quad \forall j \in \mathcal{S}, \forall k \in \mathcal{K}$$

$$\mathbf{x}_{0,j} = \hat{\mathbf{x}}_t \quad \forall j \in \mathcal{S}$$

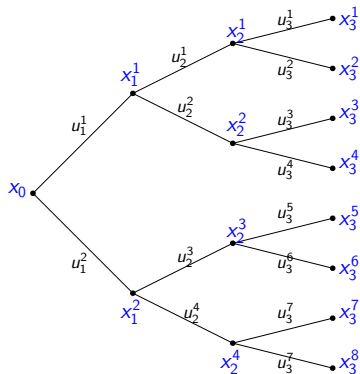
$$\mathbf{x}_{k,j} \in \mathcal{X}, \quad \mathbf{u}_{k,j} \in \mathcal{U} \quad \forall j \in \mathcal{S}, \forall k \in \mathcal{K}$$

$$\sum_{j=1}^S \bar{\mathbf{E}}_j \mathbf{u}_j = 0$$

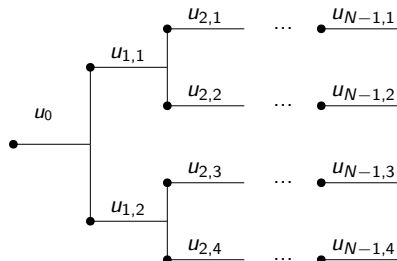
Easier to implement and maintain

Non-anticipativity constraints/Causality constraints

$$u_{k,i} = u_{k,j}, \text{ if } x_{k,i} = x_{k,j}$$

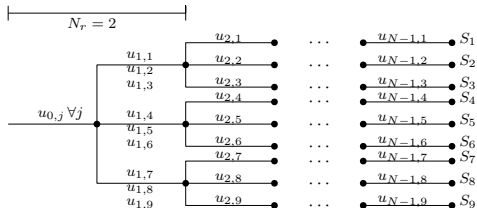
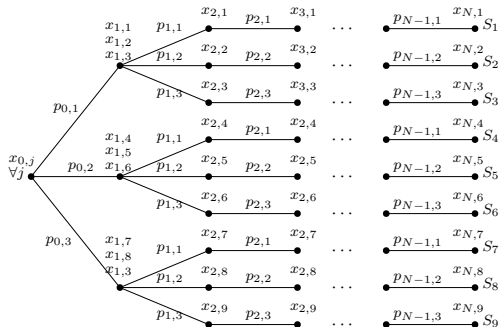


Control trajectory



Non-anticipativity constraints/Causality constraints

$$u_{k,i} = u_{k,j}, \text{ if } x_{k,i} = x_{k,j}$$



Example: Nonlinear Economic MPC

