University of Pennsylvania, ESE 6190

Model Predictive Control

Chapter 13: Robust MPC

Prof. Manfred Morari

Spring 2023

Coauthors: Prof. Colin Jones, EPFL
Prof. Melanie Zeilinger, ETH Zurich

F. Borrelli, A. Bemporad, and M. Morari, Predictive Control for Linear and Hybrid Systems, Cambridge University Press, 2017. [Ch. 15].

Outline

1. Uncertainty Models
2. Impact of Bounded Additive Noise
3. Robust Open-Loop MPC
4. Closed-Loop Predictions
5. Tube-MPC
6. Nominal MPC with noise

Lecture Take Homes

1. MPC relies on a model, but models are far from perfect
2. Noise and model inaccuracies can cause:
    - Constraint violation
    - Sub-optimal behaviour can result
3. Persistent noise prevents the system from converging to a single point
4. Can incorporate some noise models into the MPC formulation
    - Solving the resulting optimal control problem is extremely difficult
    - Many approximations exist, but most are very conservative

Outline

1. Uncertainty Models
2. Impact of Bounded Additive Noise
3. Robust Open-Loop MPC
4. Closed-Loop Predictions
5. Tube-MPC
6. Nominal MPC with noise

MPC vs The Real World

Predictive control assumption:

$$x^+ = f(x, u)$$

- System evolves in a predictable fashion

The real world:

$$x^+ = g(x, u, w; \theta)$$

- Random noise w changes the evolution of the system
- Model structure is unknown
- Unknown parameters θ impact the dynamics

(w changes with time, θ is unknown, but constant)

This lecture: What can we hope to do in this (real) situation?

Recall: Goals of Constrained Control

Constrained system
$x^+ = f(x,u) \qquad (x,u) \in \mathcal{X}, \mathcal{U}$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$
2. Is stable: $\lim_{i\to\infty} x_i = 0$
3. Optimizes "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

What if f is only known approximately?

Goals of Robust Constrained Control

Uncertain Constrained System
$$x^+ = f(x, u, w; \theta) \qquad (x, u) \in \mathcal{X}, \mathcal{U} \quad w \in \mathbb{W} \quad \theta \in \Theta$$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Meeting these goals requires some knowledge/assumptions about the random values w and θ.

Examples of Common Uncertainty Models

Measurement / Input Bias
$$g(x, u, w; \theta) = f(x, u) + \theta$$ θ unknown, but constant

- Unexpected offset can cause constraint violation
- Offset doesn't change, or changes slowly with time
  - $\rightarrow$ Generally handled by estimating offset and compensating (Chapter 10)
- Constraint violation still possible before offset is estimated

Examples of Common Uncertainty Models

Linear Parameter Varying System

$$g(x, u, w; \theta) = \sum_{k=0}^{t} \theta_k A_k x + \sum_{k=0}^{t} \theta_k B_k u \ , \quad \mathbf{1}^T \theta = 1, \ \theta \geq 0$$

A_k, B_k known, θ_k unknown, but constant

- Actual system is linear - but exact dynamics unknown
- Preventing constraint violation requires considering **all** possible trajectories (very conservative)
- Often handled by estimating θ (adaptive control), since it is constant, or changes slowly
  - Very difficult if system is unstable

Examples of Common Uncertainty Models

Polytopic Uncertainty

$$g(x, u, w; \theta) = \sum_{k=0}^{t} w_k A_k x + \sum_{k=0}^{t} w_k B_k u \ , \quad \mathbf{1}^T w = 1, \ w \geq 0$$

A_k, B_k known, w_k unknown and changing at each sample time

- Dynamics change randomly at each point in time $\rightarrow$ nonlinear system
- Preventing constraint violation requires considering **all** possible trajectories (not conservative, since they can all happen)
- Commonly dealt with via **robust MPC**

We will not cover this case in this course, but analysis is similar to additive noise.

Examples of Common Uncertainty Models

Additive Stochastic Noise

$$g(x, u, w; \theta) = Ax + Bu + w$$

Distribution of w known

- Distribution of the disturbance is known
- Problem significantly more challenging (even to formulate the goals)
- Topic of active research

Examples of Common Uncertainty Models

Additive Bounded Noise

$$g(x, u, w; \theta) = Ax + Bu + w \ , \quad w \in \mathbb{W}$$

A, B known, w unknown and changing with each sample

- Dynamics are linear, but impacted by random, bounded noise at each time step
- Can model many nonlinearities in this fashion, but often a conservative model
- The noise is *persistent*, i.e., it does not converge to zero in the limit

The next lectures will focus on uncertainty models of this form.

Outline

Goals of Robust Constrained Control

Uncertain constrained linear system
$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$

Design control law $u = \kappa(x)$ such that the system:

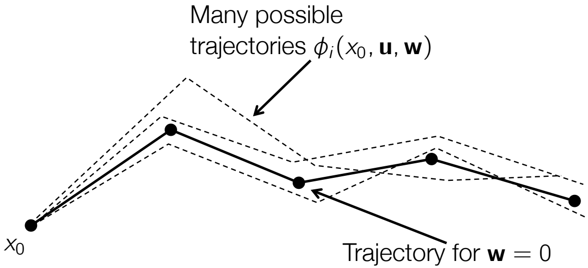
1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Challenge: Cannot predict where the state of the system will evolve
We can only compute a set of trajectories that the system *may* follow

Idea: Design a control law that will satisfy constraints and stabilize the system *for all possible disturbances*

Uncertain State Evolution

Given the current state x_0, the model $x^+ = Ax + Bu + w$ and the set $\mathbb{W}$, where can the state be i steps in the future?



Define $\phi_i(x_0, \vec{u}, \vec{w})$ as the state that the system will be in at time i if the state at time zero is x_0, we apply the input $\vec{u} := \{u_0, \dots, u_{N-1}\}$ and we observe the disturbance $\vec{w} := \{w_0, \dots, w_{N-1}\}$.

Uncertain State Evolution

Nominal system

$$x^+ = Ax + Bu$$

$$x_1 = Ax_0 + Bu_0$$
$$x_2 = A^2 x_0 + ABu_0 + Bu_1$$
$$\vdots$$
$$x_i = A^i x_0 + \sum_{k=0}^{i-1} A^k B u_{i-k}$$

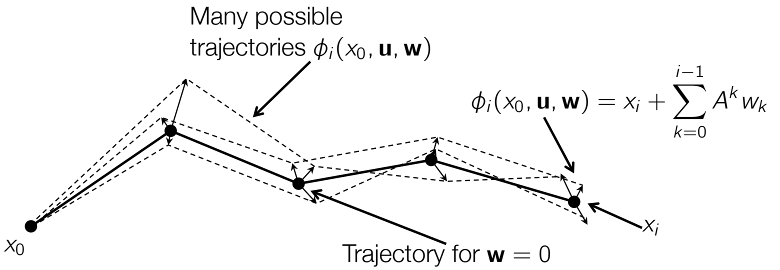
Uncertain system

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

$$\phi_1 = Ax_0 + Bu_0 + w_0$$
$$\phi_2 = A^2 x_0 + ABu_0 + Bu_1 + Aw_0 + w_1$$
$$\vdots$$
$$\phi_i = A^i x_0 + \sum_{k=0}^{i-1} A^k B u_{i-k} + \sum_{k=0}^{i-1} A^k w_{i-k}$$
$$\phi_i = x_i + \sum_{k=0}^{i-1} A^k w_{i-k}$$

Uncertain evolution is the nominal system + offset caused by the disturbance
(Follows from linearity)

Uncertain State Evolution



Outline

2. Impact of Bounded Additive Noise

   Choosing a cost to minimize

   Robust Constraint Satisfaction

Goals of Robust Constrained Control

Uncertain constrained linear system

$$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. **Optimizes (expected/worst-case) "performance"**
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Defining a Cost to Minimize

Previously, we defined some function that describes a 'good' trajectory:

$$J(x_0, \vec{u}) := \sum_{i=0}^{N-1} l(x_i, u_i) + V_f(x_N)$$

However, there are now many trajectories that *may* occur, depending on the disturbance $\vec{w}$.

The cost is now a function of the disturbance seen, and therefore *each possible trajectory has a different cost*:

$$J(x_0, \vec{u}, \vec{w}) := \sum_{i=0}^{N-1} l(\phi_i(x_0, \vec{u}, \vec{w}), u_i) + V_f(\phi_N(x_0, \vec{u}, \vec{w}))$$

Need to 'eliminate' the dependence on $\vec{w}$.

Defining a Cost to Minimize

Several common options:

- Minimize the expected value (requires some assumption on the distribution)

$$V_N(x_0, \vec{u}) := \mathbf{E}\left[J(x_0, \vec{u}, \vec{w})\right]$$

- Minimize the variance (requires some assumption on the distribution)

$$V_N(x_0, \vec{u}) := \mathrm{Var}\left(J(x_0, \vec{u}, \vec{w})\right)$$

- Take the worst-case

$$V_N(x_0, \vec{u}) := \max_{\vec{w} \in \mathbb{W}^{N-1}} J(x_0, \vec{u}, \vec{w})$$

- Take the nominal case

$$V_N(x_0, \vec{u}) := J(x_0, \vec{u}, 0)$$

Defining a Cost to Minimize

In this lecture we will assume the nominal case for simplicity.

$$V_N(x_0, \vec{u}) := J(x_0, \vec{u}, 0)$$

- We will ‘fluff’ over the stability proof, because we cannot demonstrate robust stability in this case (i.e., asymptotic convergence for all possible disturbances).
- The next lecture will introduce a new notion of stability that will allow us to analyse this case

Outline

2. Impact of Bounded Additive Noise

    Choosing a cost to minimize

    Robust Constraint Satisfaction

Goals of Robust Constrained Control

Uncertain constrained linear system
$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$

Design control law $u = \kappa(x)$ such that the system:

1. **Satifies constraints :** $\{x_i\} \subset \mathcal{X}$**,** $\{u_i\} \subset \mathcal{U}$ **for all disturbance realizations**
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) “performance”
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Robust Constraint Satisfaction

Recall: We break the MPC prediction into two parts.

$$\left.\begin{aligned} \phi_{i+1} &= A\phi_i + Bu_i + w_i \\ u_i &\in \mathcal{U} \\ \phi_i &\in \mathcal{X} \ \forall \vec{w} \in \mathbb{W}^N \end{aligned}\right\}$$

- $i = 0, \ldots, N-1$
- Optimize over control actions $\{u_0, \ldots, u_{N-1}\}$
- Enforce constraints explicitly by imposing $\phi_i \in \mathcal{X}$ and $u_i \in \mathcal{U}$ for all sequences $\vec{w}$

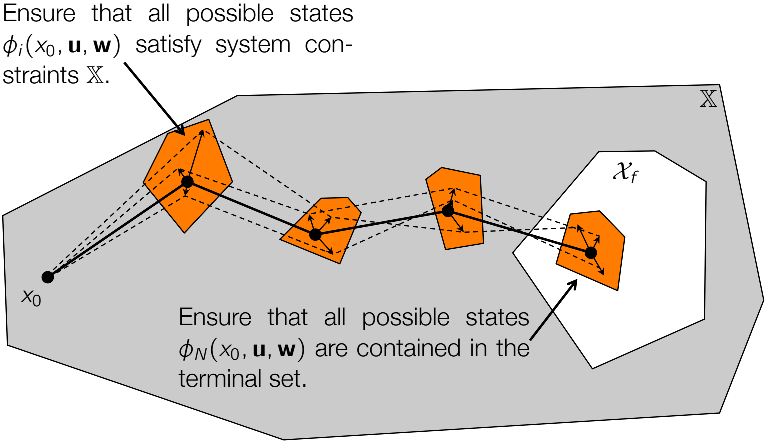
$$\left.\begin{aligned} \phi_{i+1} &= (A + BK)\phi_i + w_i \\ \phi_N &\in \mathcal{X}_f \end{aligned}\right\}$$

- $i = N, \ldots$
- Assume control law to be linear $u_i = K\phi_i$
- Enforce constraints implicitly by constraining ϕ_N to be in an *robust invariant set* $\mathcal{X}_f \subseteq \mathcal{X}$ and $K\mathcal{X}_f \subseteq \mathcal{U}$ for the system $\phi^+ = (A + BK)\phi + w$

In the following:

- Robustly enforcing constraints of a linear system
- Robustly ensuring constraints of the sequence $\phi_1, \ldots, \phi_{N-1}$

Robust Constraint Satisfaction



The idea: Compute a set of tighter constraints such that if **the nominal system** meets these constraints, then the uncertain system will too.
We then do MPC **on the nominal system.**

Robust Constraint Satisfaction

Recall: We break the MPC prediction into two parts.

$$\left.\begin{aligned} \phi_{i+1} &= A\phi_i + Bu_i + w_i \\ u_i &\in \mathcal{U} \\ \phi_i &\in \mathcal{X}\ \forall \vec{w} \in \mathbb{W}^N \end{aligned}\right\}$$

- $i = 0, \dots, N-1$
- Optimize over control actions $\{u_0, \dots, u_{N-1}\}$
- Enforce constraints explicitly by imposing $\phi_i \in \mathcal{X}$ and $u_i \in \mathcal{U}$ for all sequences $\vec{w}$

$$\left.\begin{aligned} \phi_{i+1} &= (A + BK)\phi_i + w_i \\ \phi_N &\in \mathcal{X}_f \end{aligned}\right\}$$

- $i = N, \dots$
- Assume control law to be linear $u_i = K\phi_i$
- Enforce constraints implicitly by constraining ϕ_N to be in an *robust invariant set* $\mathcal{X}_f \subseteq \mathcal{X}$ and $K\mathcal{X}_f \subseteq \mathcal{U}$ for the system $\phi^+ = (A + BK)\phi + w$

In the following:

- **Robustly enforcing constraints of a linear system**
- Robustly ensuring constraints of the sequence $\phi_1, \dots, \phi_{N-1}$

Reminder: Invariance

Constraint satisfaction, for an **autonomous** system $x^+ = f(x)$, or **closed-loop** system $x^+ = f(x, \kappa(x))$ for a **given** controller κ.

Positive Invariant set
A set $\mathcal{O}$ is said to be a positive invariant set for the autonomous system $x_{i+1} = f(x_i)$ if $$x_i \in \mathcal{O} \;\; \Rightarrow \;\; x_{i+1} \in \mathcal{O} \;\; , \;\; \forall i \in \{0, 1, \dots\}$$

If we have an invariant set $\mathcal{X}_f \subseteq \mathcal{X}$ and $\kappa(\mathcal{X}_f) \subseteq \mathcal{U}$, then it provides a set of initial states from which the trajectory will never violate the system constraints if we apply the controller κ.

Robust Invariant Set

Robust constraint satisfaction, for an **autonomous** system $x^+ = f(x, w)$, or **closed-loop** system $x^+ = f(x, \kappa(x), w)$ for a **given** controller κ.

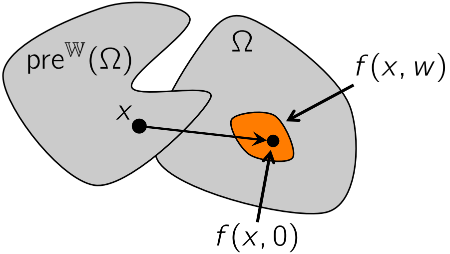
Robust Positive Invariant set
A set $\mathcal{O}^{\mathbb{W}}$ is said to be a robust positive invariant set for the autonomous system $x_{i+1} = f(x_i, w)$ if $$x \in \mathcal{O}^{\mathbb{W}} \;\Rightarrow\; f(x, w) \in \mathcal{O}^{\mathbb{W}} \;,\; \text{for all } w \in \mathbb{W}$$

Robust Pre-Sets

Robust Pre Set

Given a set Ω and the dynamic system $x^+ = f(x, w)$, the **pre-set** of Ω is the set of states that evolve into the target set Ω in one time step *for all values of the disturbance* $w \in \mathbb{W}$:

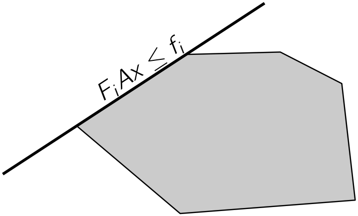
$$\text{pre}^{\mathbb{W}}(\Omega) := \{x \mid f(x, w) \in \Omega \text{ for all } w \in \mathbb{W}\}$$



Computing Robust Pre-Sets for Linear Systems

Goal: Given the system $f(x, w) = Ax + w$, and the set $\Omega := \{x \,|\, Fx \leq f\}$, compute $\mathrm{pre}^{\mathbb{W}}(\Omega)$.

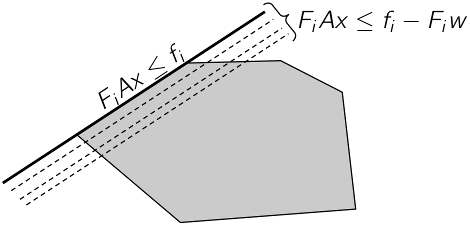
$$\mathrm{pre}^{\mathbb{W}}(\Omega) = \{x \,|\, Ax + w \in \Omega,\ \forall w \in \mathbb{W}\} = \{x \,|\, FAx + Fw \leq f,\ \forall w \in \mathbb{W}\}$$



Computing Robust Pre-Sets for Linear Systems

Goal: Given the system $f(x, w) = Ax + w$, and the set $\Omega := \{x \mid Fx \leq f\}$, compute $\text{pre}^{\mathbb{W}}(\Omega)$.

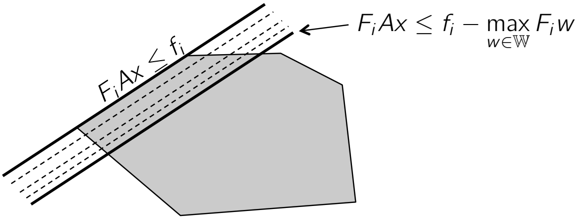
$$\text{pre}^{\mathbb{W}}(\Omega) = \{x \mid Ax + w \in \Omega,\ \forall w \in \mathbb{W}\} = \{x \mid FAx + Fw \leq f,\ \forall w \in \mathbb{W}\}$$



Computing Robust Pre-Sets for Linear Systems

Goal: Given the system $f(x, w) = Ax + w$, and the set $\Omega := \{x \,|\, Fx \le f\}$, compute $\mathrm{pre}^{\mathbb{W}}(\Omega)$.

$$\mathrm{pre}^{\mathbb{W}}(\Omega) = \{x \,|\, Ax + w \in \Omega,\ \forall w \in \mathbb{W}\} = \{x \,|\, FAx + Fw \le f,\ \forall w \in \mathbb{W}\}$$

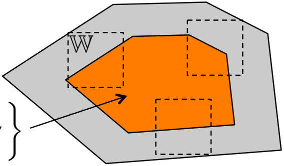


Computing Robust Pre-Sets for Linear Systems

Goal: Given the system $f(x, w) = Ax + w$, and the set $\Omega := \{x \,|\, Fx \leq f\}$, compute $\text{pre}^{\mathbb{W}}(\Omega)$.

$$\text{pre}^{\mathbb{W}}(\Omega) = \{x \,|\, Ax + w \in \Omega, \ \forall w \in \mathbb{W}\} = \{x \,|\, FAx + Fw \leq f, \ \forall w \in \mathbb{W}\}$$

$$\left\{ x \;\middle|\; FAx \leq f - \max_{w \in \mathbb{W}} Fw \right\}$$



$$\text{pre}^{\mathbb{W}}(\Omega) = \left\{ x \;\middle|\; FAx \leq f - \max_{w \in \mathbb{W}} Fw \right\} = \{x \,|\, FAx \leq f - h_{\mathbb{W}}(F)\} = A(\Omega \ominus \mathbb{W})$$

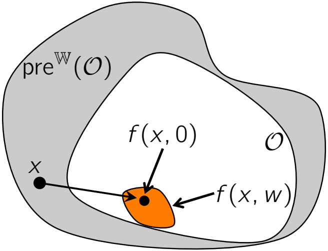
where $h_{\mathbb{W}}$ is the *support function* and $\ominus$ is called the Pontryagin difference.

Robust Invariant Set Conditions

Theorem: Geometric condition for robust invariance

A set $\mathcal{O}$ is a robust positive invariant set if and only if

$$\mathcal{O} \subseteq \mathrm{pre}^{\mathbb{W}}(\mathcal{O})$$



Computing Robust Invariant Sets

Conceptual Algorithm to Compute Robust Invariant Set

Input: f, $\mathcal{X}$, $\mathbb{W}$
Output: $\mathcal{O}_\infty^{\mathbb{W}}$

$\Omega_0 \leftarrow \mathcal{X}$
loop
  $\Omega_{i+1} \leftarrow \mathrm{pre}^{\mathbb{W}}(\Omega_i) \cap \Omega_i$
  if $\Omega_{i+1} = \Omega_i$ **then**
    return $\mathcal{O}_\infty = \Omega_i$
  end if
end loop

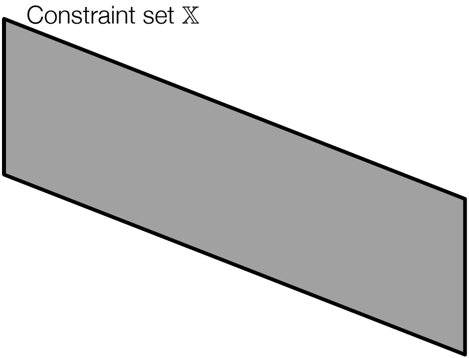
This is the same as for the nominal case, with $\mathrm{pre}(\Omega)$ replaced by $\mathrm{pre}^{\mathbb{W}}(\Omega)$.

Computing Robust Invariant Sets

$$x = (A + BK)x + w \qquad A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$$\mathcal{X} = \{x \,|\, \|x\|_\infty \leq 5,\ \|Kx\|_\infty \leq 1\} \qquad \mathbb{W} = \{w \,|\, \|w\|_\infty \leq 0.3\}$$

K is the LQR controller for $Q = 0.1I$, $R = 1$

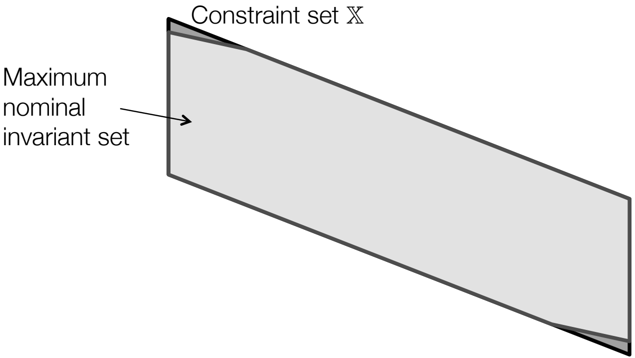


Computing Robust Invariant Sets

$$x = (A + BK)x + w \qquad A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$$\mathcal{X} = \{x \mid \|x\|_\infty \leq 5,\ \|Kx\|_\infty \leq 1\} \qquad \mathbb{W} = \{w \mid \|w\|_\infty \leq 0.3\}$$

K is the LQR controller for $Q = 0.1I$, $R = 1$

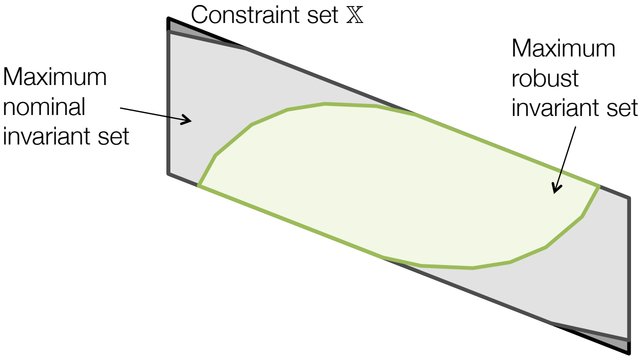


Computing Robust Invariant Sets

$$x = (A + BK)x + w \qquad A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$$\mathcal{X} = \{x \mid \|x\|_\infty \leq 5,\ \|Kx\|_\infty \leq 1\} \qquad \mathbb{W} = \{w \mid \|w\|_\infty \leq 0.3\}$$

K is the LQR controller for $Q = 0.1I$, $R = 1$

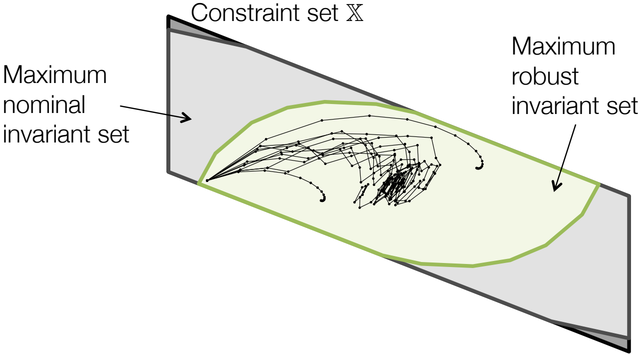


Computing Robust Invariant Sets

$$x = (A + BK)x + w \qquad A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$$\mathcal{X} = \{x \mid \|x\|_\infty \leq 5,\ \|Kx\|_\infty \leq 1\} \qquad \mathbb{W} = \{w \mid \|w\|_\infty \leq 0.3\}$$

K is the LQR controller for $Q = 0.1I$, $R = 1$

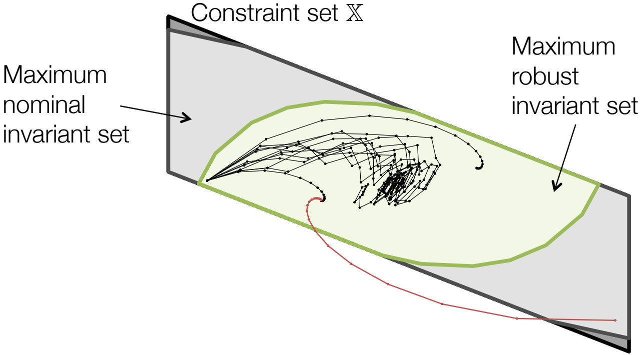


Computing Robust Invariant Sets

$$x = (A + BK)x + w \qquad A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$$\mathcal{X} = \{x \mid \|x\|_\infty \le 5,\ \|Kx\|_\infty \le 1\} \qquad \mathbb{W} = \{w \mid \|w\|_\infty \le 0.3\}$$

K is the LQR controller for $Q = 0.1I$, $R = 1$



Robust Constraint Satisfaction

Recall: We break the MPC prediction into two parts.

$$\left.\begin{aligned} \phi_{i+1} &= A\phi_i + Bu_i + w_i \\ u_i &\in \mathcal{U} \\ \phi_i &\in \mathcal{X}\ \forall \vec{w} \in \mathbb{W}^N \end{aligned}\right\}$$

- $i = 0, \dots, N-1$
- Optimize over control actions $\{u_0, \dots, u_{N-1}\}$
- Enforce constraints explicitly by imposing $\phi_i \in \mathcal{X}$ and $u_i \in \mathcal{U}$ for all sequences $\vec{w}$

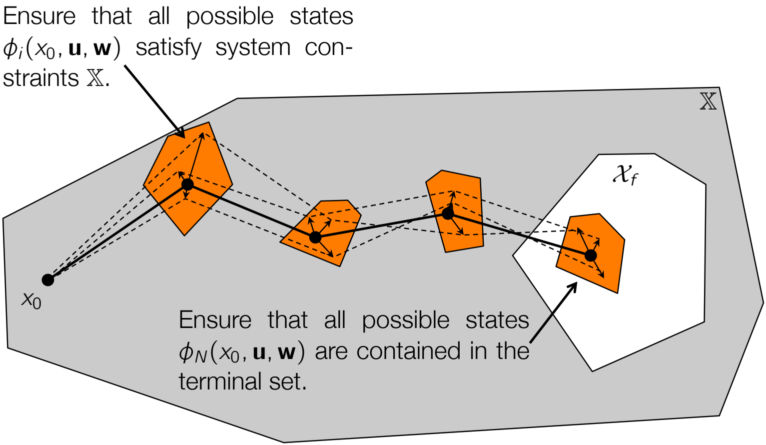
$$\left.\begin{aligned} \phi_{i+1} &= (A + BK)\phi_i + w_i \\ \phi_N &\in \mathcal{X}_f \end{aligned}\right\}$$

- $i = N, \dots$
- Assume control law to be linear $u_i = K\phi_i$
- Enforce constraints implicitly by constraining ϕ_N to be in an *robust invariant set* $\mathcal{X}_f \subseteq \mathcal{X}$ and $K\mathcal{X}_f \subseteq \mathcal{U}$ for the system $\phi^+ = (A + BK)\phi + w$

In the following:

- Robustly enforcing constraints of a linear system
- **Robustly ensuring constraints of the sequence** $\phi_1, \dots, \phi_{N-1}$

Robust Constraint Satisfaction



The idea: Compute a set of tighter constraints such that if **the nominal system** meets these constraints, then the uncertain system will too.
We then do MPC **on the nominal system.**

Uncertain State Evolution

Nominal system

$$x^+ = Ax + Bu$$

$$x_1 = Ax_0 + Bu_0$$
$$x_2 = A^2x_0 + ABu_0 + Bu_1$$
$$\vdots$$
$$x_i = A^i x_0 + \sum_{k=0}^{i-1} A^k B u_{i-k}$$

Uncertain system

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

$$\phi_1 = Ax_0 + Bu_0 + w_0$$
$$\phi_2 = A^2x_0 + ABu_0 + Bu_1 + Aw_0 + w_1$$
$$\vdots$$
$$\phi_i = A^i x_0 + \sum_{k=0}^{i-1} A^k B u_{i-k} + \sum_{k=0}^{i-1} A^k w_{i-k}$$
$$\phi_i = x_i + \sum_{k=0}^{i-1} A^k w_{i-k}$$

Goal: Ensure $\phi_i \in \mathcal{X}$ for all $\vec{w} \in \mathbb{W}^N$

Robust Constraint Satisfaction

Goal: Ensure that constraints are satisfied for the MPC sequence.

$$\phi_i(x_0, \vec{u}, \vec{w}) = \left\{ x_i + \sum_{k=0}^{i-1} A^k w_k \;\middle|\; \vec{w} \in \mathbb{W}^i \right\} \subseteq \mathcal{X}$$

Assume that $\mathcal{X} = \{x \,|\, Fx \leq f\}$, then this is equivalent to

$$Fx_i + F \sum_{k=0}^{i-1} A^k w_k \leq f \;\; \forall \vec{w} \in \mathbb{W}^i$$

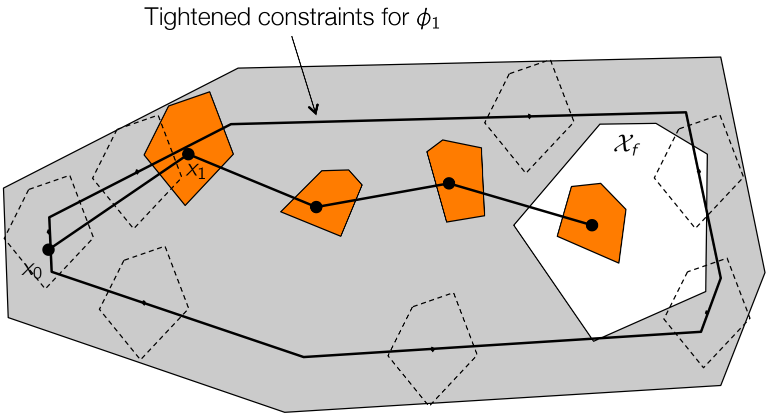
We've seen this before while computing the robust pre-set:

$$Fx_i \leq f - \max_{\vec{w} \in \mathbb{W}^i} F \sum_{k=0}^{i-1} A^k w_k = f - h_{\mathbb{W}^i}\left(F \sum_{k=0}^{i-1} A^k \right)$$

The support function can be pre-computed offline.
All we're doing is tightening the constraints on the nominal system

Robust Constraint Satisfaction

Goal: Ensure that constraints are satisfied for the MPC sequence.



Require: $x_i \in \mathcal{X} \ominus \begin{bmatrix} A^0 & \dots & A^{i-1} \end{bmatrix} \mathbb{W}^i$ and

Nominal x_i satisfies tighter constraints → Uncertain state does too

Terminal State Constraint

We also need to ensure that the N^{th} state $\phi_N(x_0, \vec{u}, \vec{w})$ is contained in the robust control invariant set $\mathcal{X}_f$:

$$\phi_N(x_0, \vec{u}, \vec{w}) \subseteq \mathcal{X}_f$$

This is handled in exactly the same fashion.

Outline

Goals of Robust Constrained Control

Uncertain constrained linear system
$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Putting it Together

Robust Open-Loop MPC

$$
\begin{aligned}
\min_{\vec{u}} \sum_{i=0}^{N-1} l(x_i, u_i) + V_f(x_N) \\
\text{subj. to } x_{i+1} &= Ax_i + Bu_i \\
x_i &\in \mathcal{X} \ominus \mathcal{A}_i \mathbb{W}^i \\
u_i &\in \mathcal{U} \\
x_N &\in \mathcal{X}_f \ominus \mathcal{A}_N \mathbb{W}^N
\end{aligned}
$$

where $\mathcal{A}_i := \begin{bmatrix} A^0 & A^1 & \dots & A^{i-1} \end{bmatrix}$ and $\mathcal{X}_f$ is a robust invariant set for the system $x^+ = (A + BK)x$ for some stabilizing K.

We do **nominal MPC**, but with tighter constraints on the states and inputs.

We can be sure that if the nominal system satisfies the tighter constraints, then the uncertain system will satisfy the real constraints.

Properties of Robust Open-Loop MPC

Robust Control Invariance

If $\vec{u}^\star(x)$ is the optimizer of the robust open-loop MPC problem, then the system $Ax + Bu_0^\star(x) + w \in \mathcal{X}$ for all $w \in \mathbb{W}$.

This follows because the trajectory we computed at the current time is feasible for *any* disturbance, and therefore it's feasible for the one that we actually observe.

We have shown the key property of robust MPC: robust invariance.

However, we have not shown convergence...

Goals of Robust Constrained Control

Uncertain constrained linear system
$x^+ = Ax + Bu + w \qquad (x,u) \in \mathcal{X},\mathcal{U} \qquad w \in \mathbb{W}$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. **Is stable: Converges to a neighbourhood of the origin**
3. Optimizes (expected/worst-case) "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

Analysis is not direct - we will consider this in a more general setting next week.

Goals of Robust Constrained Control

Uncertain constrained linear system

$$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) "performance"
4. **Maximizes the set** $\{x_0 \,|\,$ **Conditions 1-3 are met**$\}$

Robust open-loop MPC has a *very* small region of attraction!
This is why **you should not use it!**
It is a theoretical development to give you the concepts. Next week we will go through some more practical methods.

Lecture Take Homes

1. MPC relies on a model, but models are far from perfect
2. Noise and model inaccuracies can cause:
   - Constraint violation
   - Sub-optimal behaviour can result
3. Persistent noise prevents the system from converging to a single point
4. Can incorporate some noise models into the MPC formulation
   - Solving the resulting optimal control problem is extremely difficult
   - Many approximations exist, but most are very conservative

Goals of Robust Constrained Control

Uncertain constrained linear system
$x^+ = Ax + Bu + w \qquad (x, u) \in \mathcal{X}, \mathcal{U} \qquad w \in \mathbb{W}$

Design control law $u = \kappa(x)$ such that the system:

1. Satifies constraints : $\{x_i\} \subset \mathcal{X}$, $\{u_i\} \subset \mathcal{U}$ for all disturbance realizations
2. Is stable: Converges to a neighbourhood of the origin
3. Optimizes (expected/worst-case) "performance"
4. Maximizes the set $\{x_0 \,|\, \text{Conditions 1-3 are met}\}$

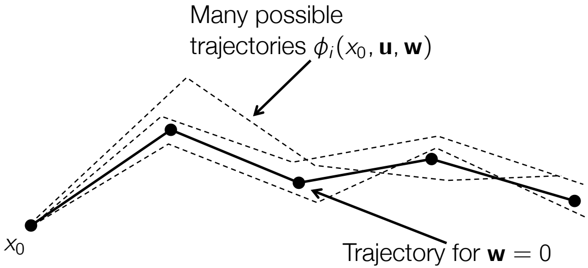
Challenge: Cannot predict where the state of the system will evolve

We can only compute a set of trajectories that the system *may* follow

Idea: Design a control law that will satisfy constraints and stabilize the system *for all possible disturbances*

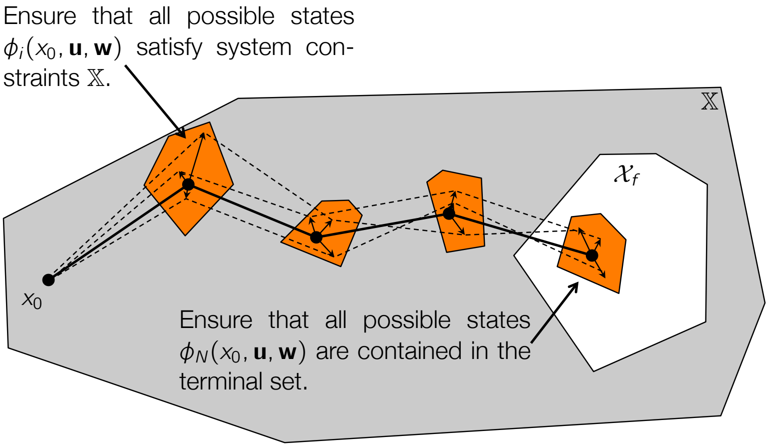
Uncertain State Evolution

Given the current state x_0, the model $x^+ = Ax + Bu + w$ and the set $\mathbb{W}$, where can the state be i steps in the future?



Define $\phi_i(x_0, \vec{u}, \vec{w})$ as the state that the system will be in at time i if the state at time zero is x_0, we apply the input $\vec{u} := \{u_0, \ldots, u_{N-1}\}$ and we observe the disturbance $\vec{w} := \{w_0, \ldots, w_{N-1}\}$.

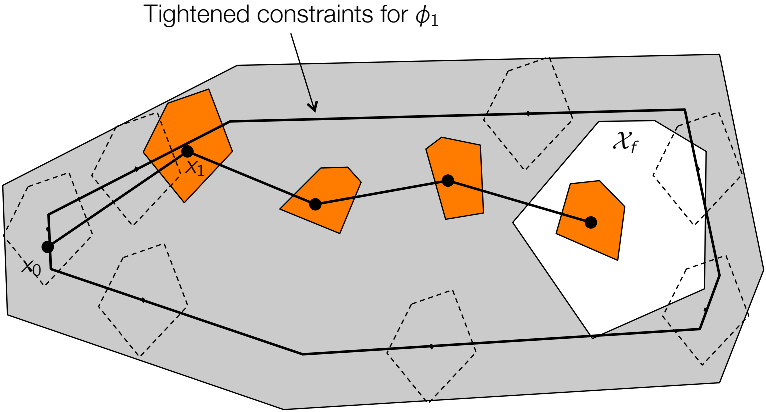
Robust Constraint Satisfaction



The idea: Compute a set of tighter constraints such that if **the nominal system** meets these constraints, then the uncertain system will too.
We then do MPC **on the nominal system.**

Robust Constraint Satisfaction

Goal: Ensure that constraints are satisfied for the MPC sequence.



Require: $x_i \in \mathcal{X} \ominus \begin{bmatrix} A^0 & \dots & A^{i-1} \end{bmatrix} \mathbb{W}^i$ and

Nominal x_i satisfies tighter constraints → Uncertain state does too

Putting it Together

Robust Open-Loop MPC

$$\begin{aligned}\min_{\vec{u}} \sum_{i=0}^{N-1} & l(x_i, u_i) + V_f(x_N) \\ \text{subj. to } & x_{i+1} = Ax_i + Bu_i \\ & x_i \in \mathcal{X} \ominus \mathcal{A}_i \mathbb{W}^i \\ & u_i \in \mathcal{U} \\ & x_N \in \mathcal{X}_f \ominus \mathcal{A}_N \mathbb{W}^N\end{aligned}$$

where $\mathcal{A}_i := \begin{bmatrix} A^0 & A^1 & \dots & A^{i-1} \end{bmatrix}$ and $\mathcal{X}_f$ is a robust invariant set for the system $x^+ = (A + BK)x$ for some stabilizing K.

We do **nominal MPC**, but with tighter constraints on the states and inputs.

We can be sure that if the nominal system satisfies the tighter constraints, then the uncertain system will satisfy the real constraints.

$\Rightarrow$ Downside is that $\mathcal{A}_i \mathbb{W}^i$ can be very large

Outline

1. Uncertainty Models
2. Impact of Bounded Additive Noise
3. Robust Open-Loop MPC
4. Closed-Loop Predictions
5. Tube-MPC
6. Nominal MPC with noise

MPC as a Game

Two players: Controller vs Disturbance

$$x^+ = f(x, u) + w$$

1. Controller chooses his move u
2. Disturbance decides on his move w **after seeing the controller's move**

MPC as a Game

Two players: Controller vs Disturbance

$$x^+ = f(x, u) + w$$

1. Controller chooses his move u
2. Disturbance decides on his move w **after seeing the controller's move**

What are we assuming when making robust predictions?

1. Controller chooses a **sequence** of N moves in the future $\{u_0, \ldots, u_{N-1}\}$
2. Disturbance chooses N moves **knowing all N moves of the controller**

We are assuming that the controller will do the same thing in the future no matter what the disturbance does!

Can we do better?

Closed-Loop Predictions

What should the future prediction look like?

1. Controller decides his first move u_0
2. Disturbance chooses his first move w_0
3. Controller decides his second move $u_1(x_1)$ **as a function of the first disturbance** w_0 **(recall** $x_1 = Ax_0 + Bu_0 + w_0$**)**
4. Disturbance chooses his second move w_1 as a function of u_1
5. Controller decides his third move $u_2(x_2)$ **as a function of the first two disturbances** w_0**,** w_1
6. ...

Closed-Loop Predictions

We want to optimize over a **sequence of functions** $\{u_0, \mu_1(\cdot), \ldots, \mu_{N-1}(\cdot)\}$, where $\mu_i(x_i) : \mathbb{R}^n \to \mathbb{R}^m$ is called a **control policy**, and maps the state at time i to an input at time i.

Notes:

- This is the same as making μ a function of the disturbances to time i, since the state is a function of the disturbances up to that point
- The first input u_0 is a function of the current state, which is known. Therefore it is not a function, but a single value.

The problem: We can't optimize over arbitrary functions!

Closed-Loop MPC

A solution: Assume some structure on the functions μ_i

Pre-stabilization $\mu_i(x) = Kx + v_i$

- Fixed K, such that $A + BK$ is stable
- Simple, often conservative

Linear feedback $\mu_i(x) = K_i x + v_i$

- Optimize over K_i and v_i
- Non-convex. Extremely difficult to solve...

Disturbance feedback $\mu_i(x) = \sum_{j=0}^{i-1} M_{ij} w_j + v_i$

- Optimize over M_{ij} and v_i
- Equivalent to linear feedback, but convex!
- Can be very effective, but computationally intense.

Tube-MPC $\mu_i(x) = v_i + K(x - \bar{x}_i)$

- Fixed K, such that $A + BK$ is stable
- Optimize over $\bar{x}_i$ and v_i
- Simple, and can be effective

We will cover tube-MPC in this lecture.

Outline

Tube-MPC

$$x^+ = Ax + Bu + w \qquad (x,u) \in \mathcal{X} \times \mathcal{U} \qquad w \in \mathbb{W}$$

The idea: Separate the available control authority into two parts

1. A portion that steers the noise-free system to the origin $z^+ = Az + Bv$
2. A portion that compensates for deviations from this system $e^+ = (A + BK)e + w$

We fix the linear feedback controller K offline, and optimize over the nominal trajectory $\{v_0, \dots, v_{N-1}\}$, which results in a convex problem.

[0] Further reading: D.Q. Mayne, M.M. Seron and S.V. Rakovic, Robust model predictive control of constrained linear systems with bounded disturbances, Automatica, Volume 41, Issue 2, February 2005

System Decomposition

Define a 'nominal', noise-free system:

$$z_{i+1} = Az_i + Bv_i$$

Define a 'tracking' controller, to keep the real trajectory close to the nominal

$$u_i = K(x_i - z_i) + v_i$$

for some linear controller K, which stabilizes the nominal system.

Define the error $e_i = x_i - z_i$, which gives the error dynamics:

$$\begin{aligned}
e_{i+1} &= x_{i+1} - z_{i+1} \\
&= Ax_i + Bu_i + w_i - Az_i - Bv_i \\
&= Ax_i + BK(x_i - z_i) + Bv_i + w_i - Az_i - Bv_i \\
&= (A + BK)(x_i - z_i) + w_i \\
&= (A + BK)e_i + w_i
\end{aligned}$$

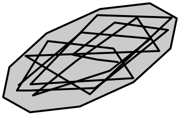
Error Dynamics

Bound maximum error, or how far the 'real' trajectory is from the nominal

$$e_{i+1} = (A + BK)e_i + w_i \qquad\qquad w_i \in \mathbb{W}$$

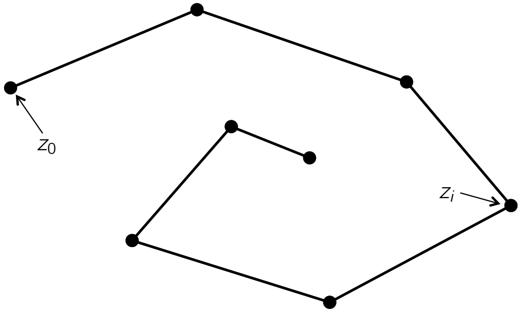
Dynamics $A + BK$ are stable, and the set $\mathbb{W}$ is bounded, so there is some set $\mathcal{E}$ that e will stay inside for all time.

We want the smallest such set (the 'minimal invariant set')



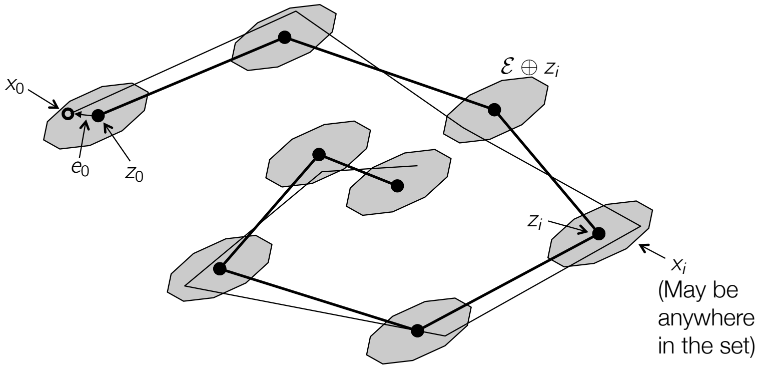
We will cover how to compute this set later

Tube-MPC : The Idea



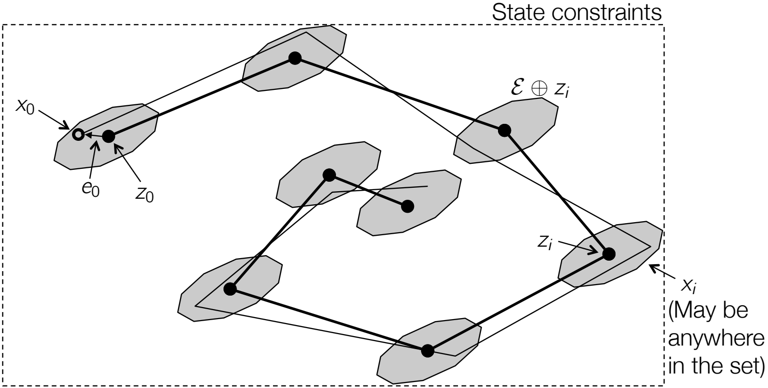
We want to ignore the noise and plan the **nominal trajectory**

Tube-MPC : The Idea



We know that the real trajectory stays 'nearby' the nominal one: $x_i \in z_i \oplus \mathcal{E}$ **because we plan to apply the controller** $u_i = K(x_i - z_i) + v_i$ **in the future** (we won't actually do this, but it's a valid sub-optimal plan)

Tube-MPC : The Idea



We must ensure that all possible state trajectories satisfy the constraints
This is now equivalent to ensuring that $z_i \oplus \mathcal{E} \subset \mathcal{X}$
(Satisfying input constraints is now more complex - more later)

Tube-MPC

What do we need to make this work?

- Compute the set $\mathcal{E}$ that the error will remain inside
- Modify constraints on nominal trajectory $\{z_i\}$ so that $z_i \oplus \mathcal{E} \subset \mathcal{X}$ and $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- Formulate as convex optimization problem

... and then prove that

- Constraints are robustly satisfied
- The closed-loop system is robustly stable

Tube-MPC

What do we need to make this work?

- **Compute the set $\mathcal{E}$ that the error will remain inside**
- Modify constraints on nominal trajectory $\{z_i\}$ so that $z_i \oplus \mathcal{E} \subset \mathcal{X}$ and $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- Formulate as convex optimization problem

...and then prove that

- Constraints are robustly satisfied
- The closed-loop system is robustly stable

Recall: Robust Invariant Set

Robust constraint satisfaction, for an **autonomous** system $x^+ = f(x, w)$, or **closed-loop** system $x^+ = f(x, \kappa(x), w)$ for a **given** controller κ.

Robust Positive Invariant set
A set $\mathcal{O}^{\mathbb{W}}$ is said to be a robust positive invariant set for the autonomous system $x_{i+1} = f(x_i, w)$ if $$x \in \mathcal{O}^{\mathbb{W}} \;\; \Rightarrow \;\; f(x, w) \in \mathcal{O}^{\mathbb{W}} \;\;, \;\; \text{for all } w \in \mathbb{W}$$

Previously we wanted the **maximum robust invariant set**, or the largest set in which our terminal control law works.

We now want the **minimum robust invariant set**, or the smallest set that the state will remain inside despite the noise.

Uncertain State Evolution

Consider the system $x^+ = Ax + w$ and assume that $x_0 = 0$.

Where can the state evolve to? (i.e., how close can we stay to the origin?)

$$
\begin{aligned}
x_1 &= w_0 \\
x_2 &= Ax_1 + w_1 = Aw_0 + w_1 \\
&\vdots \\
x_i &= \sum_{k=0}^{i-1} A^k w_k
\end{aligned}
$$

Assume that $w_i \in \mathbb{W}$ for all i. What is the set F_i that contains all possible states x_i?

$$
F_i = \bigoplus_{k=0}^{i-1} A^k \mathbb{W} \ , \quad F_0 := \{0\}
$$

where $P \oplus Q := \{x + y \,|\, x \in P,\ y \in Q\}$

Minimum Robust Invariant Set

As sum goes to infinity, we arrive at the **minimum robust invariant set** mRPI

$$F_\infty = \bigoplus_{k=0}^{\infty} A^k \mathbb{W} \ , \quad F_0 := \{0\}$$

If there exists an n such that $F_n = F_{n+1}$, then $F_n = F_\infty$

Minimal Invariant Set

Input: A
Output: F_∞
  $\Omega_0 \leftarrow \{0\}$
  loop
    $\Omega_{i+1} \leftarrow \Omega_i \oplus A^i \mathbb{W}$
    if $\Omega_{i+1} = \Omega_i$ **then**
      return $F_\infty = \Omega_i$
    end if
  end loop

- A finite n does not always exist, but a 'large' n is a good approximation
- If n is not finite, there are other methods of computing small invariant sets, which will be slightly larger than F_∞

Computing Minkowski Sums for Polyhedral Data

Given $P := \{x \,|\, Tx \leq t\}$ and $Q := \{x \,|\, Rx \leq r\}$, the Minkowski sum is:

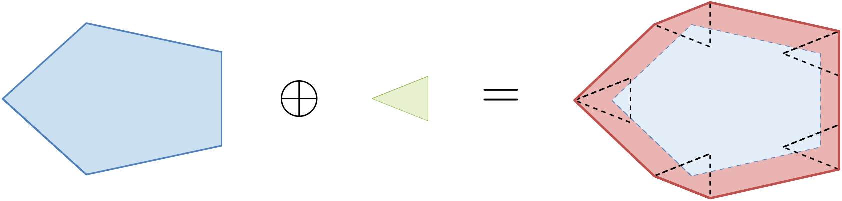
$$\begin{aligned} P \oplus Q &:= \{x + y \,|\, x \in P,\ y \in Q\} \\ &= \{z \,|\, \exists x, y\ z = x + y,\ Tx \leq t,\ Ry \leq r\} \\ &= \{z \,|\, \exists y\ Tz - Ty \leq t,\ Ry \leq r\} \\ &= \left\{ z \,\middle|\, \exists y\ \begin{bmatrix} T & -T \\ 0 & R \end{bmatrix} \begin{pmatrix} z \\ y \end{pmatrix} \leq \begin{pmatrix} t \\ r \end{pmatrix} \right\} \end{aligned}$$

This is a **projection** of a polyhedron from (z, y) onto z.

Minkowski Sums in MPT

Recall: We covered computation of projection in Lecture 4.

```
P = polytope(T,t);
Q = polytope(R,r);
Z = zeros(size(R,1),size(T,2));
P_plus_Q = projection(polytope([T -T; Z R], [t;r]), 1:size(T,2));
plot([P Q P_plus_Q]);
```

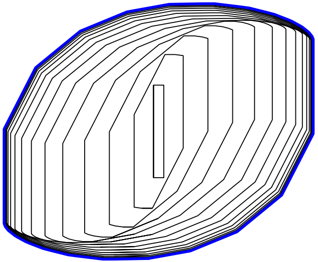


Example

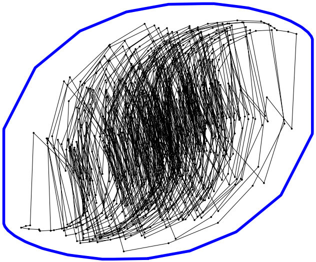
System dynamics

$$x^+ = \left(\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} K \right) x + w \quad \mathbb{W} := \{ w \mid |w_1| \leq 0.01,\ |w_2| \leq 0.1 \}$$

where K is the LQR controller for $Q = I$, $R = 10$.



Sets $A^i\mathbb{W}$ converging to minimal robust invariant set F_∞ in the limit



The state trajectory will stay in the set F_∞ for all time

Tube-MPC

What do we need to make this work?

- Compute the set $\mathcal{E}$ that the error will remain inside
- **Modify constraints on nominal trajectory** $\{z_i\}$ **so that** $z_i \oplus \mathcal{E} \subset \mathcal{X}$ **and** $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- Formulate as convex optimization problem

... and then prove that

- Constraints are robustly satisfied
- The closed-loop system is robustly stable

Noisy System Trajectory

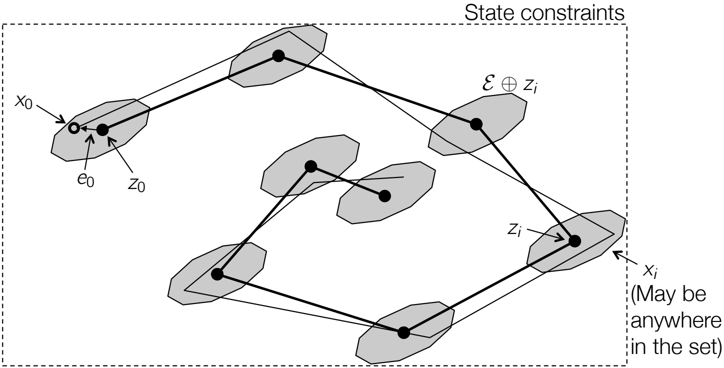
Given the nominal trajectory z_i, what can the noisy system trajectory do?

$$x_i = z_i + e_i$$

Don't know what error will be at time i, but it will be in the set $\mathcal{E}$

Therefore, x_i can only be up to $\mathcal{E}$ far from z_i

$$x_i \in z_i \oplus \mathcal{E} = \{z_i + e \,|\, e \in \mathcal{E}\}$$

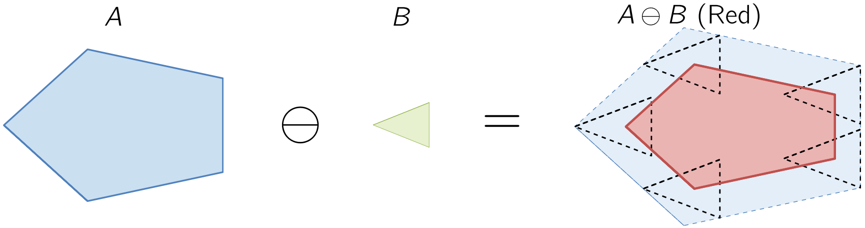


Pontryagin Difference

Pontryagin Difference

Let A and B be subsets of $\mathbb{R}^n$. The Pontryagin Difference is

$$A \ominus B := \{x \,|\, x + e \in A \;\forall e \in B\}$$



Lemma

$$x \in A \ominus B \Rightarrow x + e \in A \;\forall e \in B$$

We covered how to compute the Pontryagin Difference last week

Constraint Tightening

Goal: $(x_i, u_i) \in \mathcal{X} \times \mathcal{U}$ for all $\{w_0, \dots, w_{i-1}\} \in \mathbb{W}^i$

We want to work with the nominal system $z^+ = Az + Bv$ but ensure that the noisy system $x^+ = Ax + Bu + w$ satisfies the constraints.

Sufficient condition:

$$z_i \oplus \mathcal{E} \subseteq \mathcal{X} \qquad \Leftarrow \qquad z_i \in \mathcal{X} \ominus \mathcal{E}$$

The set $\mathcal{E}$ is known offline - we can compute the constraints $\mathcal{X} \ominus \mathcal{E}$ offline!

A similar condition holds for the inputs:

$$u_i \in K\mathcal{E} \oplus v_i \subset \mathcal{U} \qquad \Leftarrow \qquad v_i \in \mathcal{U} \ominus K\mathcal{E}$$

Tube-MPC

What do we need to make this work?

- Compute the set $\mathcal{E}$ that the error will remain inside
- Modify constraints on nominal trajectory $\{z_i\}$ so that $z_i \oplus \mathcal{E} \subset \mathcal{X}$ and $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- **Formulate as convex optimization problem**

. . . and then prove that

- Constraints are robustly satisfied
- The closed-loop system is robustly stable

Tube-MPC Problem Formulation

Tube-MPC

Feasible set: $$\mathcal{Z}(x_0) := \left\{ \vec{z}, \vec{v} \;\middle|\; \begin{array}{ll} z_{i+1} = Az_i + Bv_i & i \in [0,\ N-1] \\ z_i \in \mathcal{X} \ominus \mathcal{E} & i \in [0,\ N-1] \\ v_i \in \mathcal{U} \ominus K\mathcal{E} & i \in [0,\ N-1] \\ z_N \in \mathcal{X}_f & \\ x_0 \in z_0 \oplus \mathcal{E} & \end{array} \right\}$$

Cost function: $$V(\vec{z}, \vec{v}) := \sum_{i=0}^{N-1} l(z_i, v_i) + V_f(z_N)$$

Optimization problem: $$(\vec{v}^\star(x_0), \vec{z}^\star(x_0)) = \underset{\vec{v},\vec{z}}{\operatorname{argmin}} \{V(\vec{z}, \vec{v}) \,|\, (\vec{z}, \vec{v}) \in \mathcal{Z}(x_0)\}$$

Control law: $$\mu_{\text{tube}}(x) := K(x - z_0^\star(x)) + v_0^\star(x)$$

- Optimizing the nominal system, with tightened state an input constraints
- First tube center is optimization variable $\rightarrow$ has to be within $\mathcal{E}$ of x_0
- The cost is with respect to the tube centers
- The terminal set is with respect to the tightened constraints

Tube-MPC

What do we need to make this work?

- Compute the set $\mathcal{E}$ that the error will remain inside
- Modify constraints on nominal trajectory $\{z_i\}$ so that $z_i \oplus \mathcal{E} \subset \mathcal{X}$ and $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- Formulate as convex optimization problem

. . . and then prove that

- **Constraints are robustly satisfied**
- The closed-loop system is robustly stable

Tube-MPC Assumptions

Much the same as for nominal MPC:

1. The stage cost is a positive definite function, i.e. it is strictly positive and only zero at the origin

2. The terminal set is invariant **for the nominal system** under the local control law $\kappa_f(z)$:

   $$z^+ = Az + B\kappa_f(z) \in \mathcal{X}_f \quad \text{for all } z \in \mathcal{X}_f$$

   All **tightened state and input constraints** are satisfied in $\mathcal{X}_f$:

   $$\mathcal{X}_f \subseteq \mathbb{X} \ominus \mathcal{E},\ \kappa_f(z) \in \mathbb{U} \ominus K\mathcal{E} \text{ for all } z \in \mathcal{X}_f$$

3. Terminal cost is a continuous Lyapunov function in the terminal set $\mathcal{X}_f$:

   $$V_f(Az + B\kappa_f(z)) - V_f(z) \leq -l(z, \kappa_f(z)) \text{ for all } z \in \mathcal{X}_f$$

Robust Invariance

Thm: Robust Invariance of Tube-MPC

The set $\mathcal{Z} := \{x \mid \mathcal{Z}(x) \neq \emptyset\}$ is a robust invariant set of the system $x^+ = Ax + B\mu_{\text{tube}}(x) + w$ subject to the constraints $(x, u) \in \mathcal{X} \times \mathcal{U}$.

Let $(\left\{v_0^\star, \ldots, v_{N-1}^\star\right\}, \left\{z_0^\star, \ldots, z_N^\star\right\})$ be the optimal solution for time x_0.

At the next point in time, the state is:

$$x_1 = Ax_0 + BK(x_0 - z_0^\star) + Bv_0^\star + w \quad \text{for some } w \in \mathbb{W}$$

i.e., the state x_1 may have many possible values. We need to show that there exists a feasible solution for **all of them**.

By construction, the state x_1 is in the set $z_1 \oplus \mathcal{E}$ for all $\mathbb{W}$. Therefore (as in standard MPC), the sequence

$$(\left\{v_1^\star, \ldots, v_{N-1}^\star, \kappa_f(z_N^\star)\right\}, \{z_1^\star, \ldots, z_N^\star, Az_N^\star + B\kappa_f(z_N^\star)\})$$

is feasible for all x_1.

Tube-MPC

What do we need to make this work?

- Compute the set $\mathcal{E}$ that the error will remain inside
- Modify constraints on nominal trajectory $\{z_i\}$ so that $z_i \oplus \mathcal{E} \subset \mathcal{X}$ and $v_i \in \mathcal{U} \ominus K\mathcal{E}$
- Formulate as convex optimization problem

... and then prove that

- Constraints are robustly satisfied
- **The closed-loop system is robustly stable**

Robust Stability

Thm: Robust Stability of Tube-MPC

The state x of the system $x^+ = Ax + B\mu_{\text{tube}}(x) + w$ converges in the limit to the set $\mathcal{E}$.

As in standard MPC, we have the relationship:

$$
\begin{aligned}
J^\star(x_0) &= \sum_{i=0}^{N-1} l(z_i^\star, v_i^\star) + V_f(z_N^\star) \\
J^\star(x_1) &\leq \sum_{i=1}^{N} l(z_i^\star, v_i^\star) + V_f(z_{N+1}^\star) \\
&= J^\star(x_0) - \underbrace{l(z_0^\star, v_0^\star)}_{\geq 0} + \underbrace{V_f(z_{N-1}^\star) - V_f(z_N^\star) + l(z_N^\star, \kappa_f(z_N^\star))}_{\leq 0 \ (V_f \text{ is a Lyapunov function in } \mathcal{X}_f)}
\end{aligned}
$$

This shows that $\lim_{i\to\infty} J(z_0^\star(x_i)) = 0$, and therefore $\lim_{i\to\infty} z_0^\star(x_i) = 0$.

However, x_i does not tend to zero! It only stays within a robust invariant set centered at $z_0^\star(x_i)$: $\lim_{i\to\infty} \mathbf{dist}(x_i, \mathcal{E}) = 0$, where $\mathbf{dist}$ is any distance function.

Putting it all together: Tube MPC

To implement tube MPC:

— Offline —

1. Choose a stabilizing controller K so that $\|A + BK\| < 1$
2. Compute the minimal robust invariant set $\mathcal{E} = F_\infty$ for the system $x^+ = (A + BK)x + w$, $w \in \mathbb{W}$[1]
3. Compute the tightened constraints $\tilde{\mathcal{X}} := \mathcal{X} \ominus \mathcal{E}$, $\tilde{\mathcal{U}} := \mathcal{U} \ominus K\mathcal{E}$
4. Choose terminal weight function V_f and constraint $\mathcal{X}_f$ satisfying assumptions on slide 88

— Online —

1. Measure / estimate state x
2. Solve the problem $(\vec{v}^\star(x), \vec{z}^\star(x)) = \operatorname{argmin}_{\vec{v},\vec{z}} \{V(\vec{z},\vec{v}) \mid (\vec{z},\vec{v}) \in \mathcal{Z}(x)\}$ (Slide 86)
3. Set the input to $u = K(x - z_0^\star(x)) + v_0^\star(x)$

[1] Note that it is often not possible to compute the minimal robust invariant set, as it may have an infinite number of facets. Therefore, we often take an invariant outer approximation.

Example

System dynamics

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u + w \qquad \mathbb{W} := \{w \mid |w_1| \leq 0.01,\ |w_2| \leq 0.1\}$$

Constraints:

$$\mathcal{X} := \{x \mid \|x\|_\infty \leq 1\} \qquad \mathcal{U} := \{u \mid \|u\| \leq 1\}$$

Stage cost is:

$$l(z, v) := {z_i}^\top Q z_i + {v_i}^\top R v_i$$

where

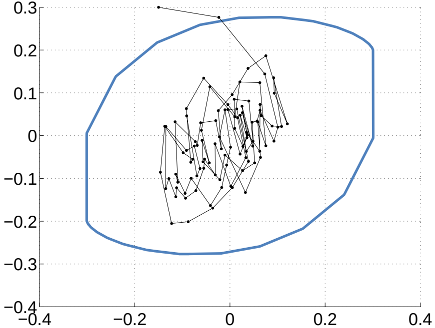
$$Q := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad R := 10$$

Offline Design - Compute Minimal Invariant Set

1. Choose a stabilizing controller K so that $\|A + BK\| < 1$
2. Compute the minimal robust invariant set $\mathcal{E} = F_\infty$ for the system $x^+ = (A + BK)x + w$, $w \in \mathbb{W}$

We take the LQR controller for $Q = I$, $R = 1$:

$$K := \begin{bmatrix} -0.5198 & -0.9400 \end{bmatrix}$$



Evolution of the system $x^+ = (A + BK)x + w$ for $x_0 = \begin{bmatrix} -0.1 & 0.2 \end{bmatrix}^T$

Offline Design - Tighten State Constraints

3. Compute the tightened constraints $\tilde{\mathcal{X}} := \mathcal{X} \ominus \mathcal{E}$, $\tilde{\mathcal{U}} := \mathcal{U} \ominus K\mathcal{E}$

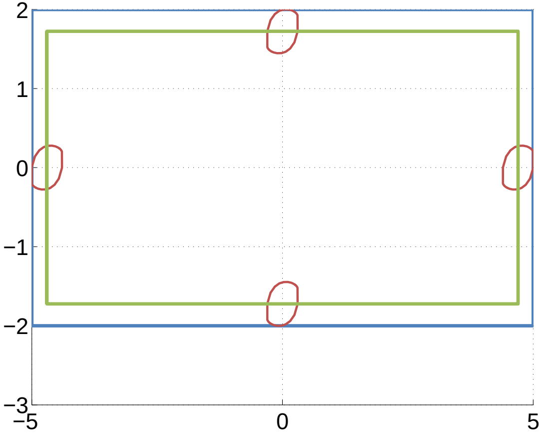
$$\mathcal{X} = \{x \mid \|x\|_\infty \leq 1\} = \left\{ x \,\middle|\, \begin{bmatrix} I \\ -I \end{bmatrix} x \leq \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \end{bmatrix} \right\}$$

If $\mathcal{E} = \{x \mid Fx \leq f\}$, then the tightened constraint sets are:

$$\begin{aligned}
\mathcal{X} \ominus \mathcal{E} = \{x \mid x + e \in \mathcal{X} \ \forall e \in \mathcal{E}\} &= \left\{ x \,\middle|\, \begin{bmatrix} I \\ -I \end{bmatrix} x + \begin{bmatrix} I \\ -I \end{bmatrix} e \leq \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \end{bmatrix} \ \forall e \in \mathcal{E} \right\} \\
&= \left\{ x \,\middle|\, \begin{bmatrix} I \\ -I \end{bmatrix} x \leq \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} e \ \forall e \in \mathcal{E} \right\} \\
&= \left\{ x \,\middle|\, \begin{bmatrix} I \\ -I \end{bmatrix} x \leq \min \left\{ \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \end{bmatrix} + \begin{bmatrix} -I \\ I \end{bmatrix} e \,\middle|\, e \in \mathcal{E} \right\} \right\} \\
&= \left\{ x \,\middle|\, \begin{bmatrix} I \\ -I \end{bmatrix} x \leq \begin{bmatrix} 1 + \min \left\{ \begin{bmatrix} -1 & 0 \end{bmatrix} e \,\middle|\, Fe \leq f \right\} \\ 1 + \min \left\{ \begin{bmatrix} 0 & -1 \end{bmatrix} e \,\middle|\, Fe \leq f \right\} \\ 1 + \min \left\{ \begin{bmatrix} 1 & 0 \end{bmatrix} e \,\middle|\, Fe \leq f \right\} \\ 1 + \min \left\{ \begin{bmatrix} 0 & 1 \end{bmatrix} e \,\middle|\, Fe \leq f \right\} \end{bmatrix} \right\} \quad \text{Linear programs, can compute offline}
\end{aligned}$$

The result is a polytope with smaller RHS.

Offline Design - Tighten State Constraints



Blue : Original constraint set $\mathcal{X}$
Red : Error set $\mathcal{E}$
Green : Tightened constraints $\mathcal{X} \ominus \mathcal{E}$

Offline Design - Tighten Input Constraints

We compute $\mathcal{U} \ominus K\mathcal{E}$ in the same manner:

$$
\begin{aligned}
\mathcal{U} \ominus K\mathcal{E} &= \left\{ u \,\middle|\, \begin{bmatrix} 1 \\ -1 \end{bmatrix} u \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\} \ominus \{ Ke \mid e \in \mathcal{E} \} \\
&= \left\{ u \,\middle|\, \begin{bmatrix} 1 \\ -1 \end{bmatrix} u \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \end{bmatrix} Ke \; \forall e \in \mathcal{E} \right\} \\
&= \left\{ u \,\middle|\, \begin{bmatrix} 1 \\ -1 \end{bmatrix} u \leq \min \left\{ \begin{bmatrix} 1 - Ke \\ 1 + Ke \end{bmatrix} \,\middle|\, Fe \leq f \right\} \right\} \\
&= \left\{ u \,\middle|\, \begin{bmatrix} 1 \\ -1 \end{bmatrix} u \leq \begin{bmatrix} 1 + \min \{ -Ke \mid Fe \leq f \} \\ 1 + \min \{ Ke \mid Fe \leq f \} \end{bmatrix} \right\}
\end{aligned}
$$

Offline Design - Terminal Weights and Constraints

We need to find a function V_f and a set $\mathcal{X}_f$ that satisfy the conditions on slide 88:

1. The terminal set is invariant **for the nominal system** under the local control law $\kappa_f(z)$:

   $$z^+ = Az + B\kappa_f(z) \in \mathcal{X}_f \quad \text{for all } z \in \mathcal{X}_f$$

   All **tightened state and input constraints** are satisfied in $\mathcal{X}_f$:

   $$\mathcal{X}_f \subseteq \mathbb{X} \ominus \mathcal{E},\ \kappa_f(z) \in \mathbb{U} \ominus K\mathcal{E} \text{ for all } z \in \mathcal{X}_f$$

2. Terminal cost is a continuous Lyapunov function in the terminal set $\mathcal{X}_f$:

   $$V_f(Az + B\kappa_f(z)) - V_f(z) \leq -l(z, \kappa_f(z)) \text{ for all } z \in \mathcal{X}_f$$

Offline Design - Terminal Constraint

We base our terminal weights and constraints on the LQR controller (many other choices possible).

Choose the terminal control law to the the LQR control law: $\kappa_f(x) = Kx$ where the weights Q and R are taken the same as for our MPC problem.

We need a set $\mathcal{X}_f$ that is invariant under this controller and contained in the tightened constraints:

$$\text{pre}(\mathcal{X}_f) \subseteq \mathcal{X}_f \quad \text{and} \quad \mathcal{X}_f \subseteq \mathcal{X} \ominus \mathcal{E} \quad \text{and} \quad K\mathcal{X}_f \subseteq \mathcal{U} \ominus K\mathcal{E}$$

We know how to compute the maximal invariant set for linear systems with polytopic constraints (Lecture: Introduction to Constrained Systems)

Offline Design - Terminal Cost

We need to find a function V_f with the property:

$$V_f(Az + B\kappa_f(z)) - V_f(z) \leq -l(z, \kappa_f(z)) \text{ for all } z \in \mathcal{X}_f$$

where we've chosen $\kappa_f(z) = Kz$ (the optimal LQR controller)

Recall the the optimal cost of the LQR control law is:

$$V^\star(z_0) = \sum_{i=0}^{\infty} {z_i}^\top (Q + K^T RK) z_i = z_0^T P z_0$$

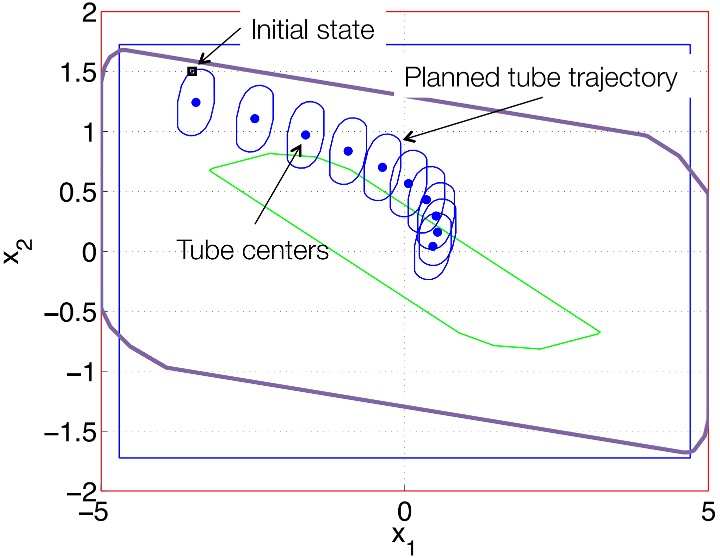
where P is the solution to a discrete-time Riccati equation.

We know that $V^\star(z)$ is a Lyapunov function for the system $z^+ = (A + BK)z$:

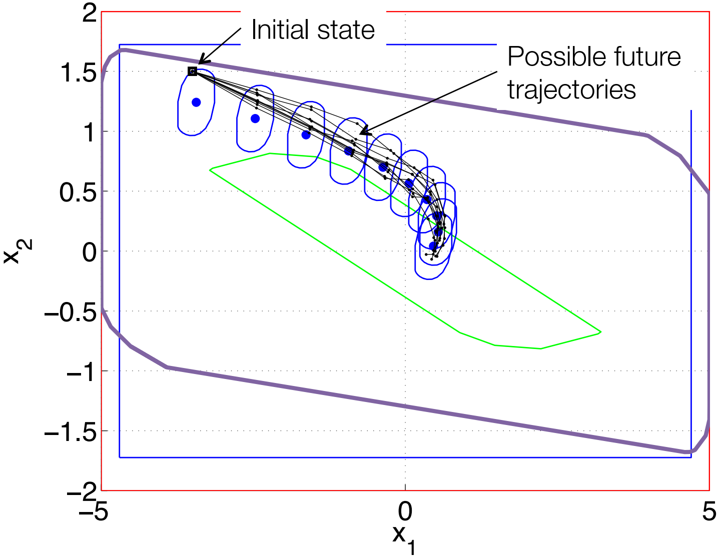
$$\begin{aligned} V^\star(z_1) - V^\star(z_0) &= \sum_{i=1}^{\infty} {z_i}^\top (Q + K^T RK) z_i - \sum_{i=0}^{\infty} {z_i}^\top (Q + K^T RK) z_i \\ &= -{z_0}^\top (Q + K^T RK) z_0 = -l(z_0, \kappa_f(z_0)) \end{aligned}$$

which is exactly what we need, and therefore, we can take $V_f(z) = z^T P z$.

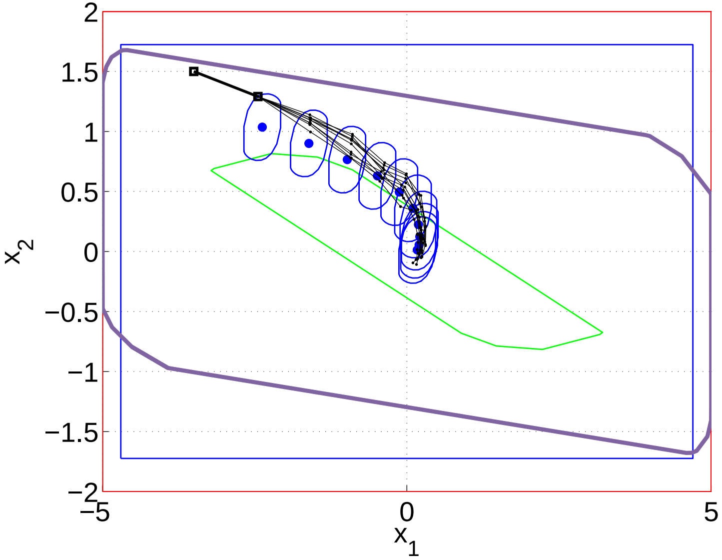
Tubes - Example



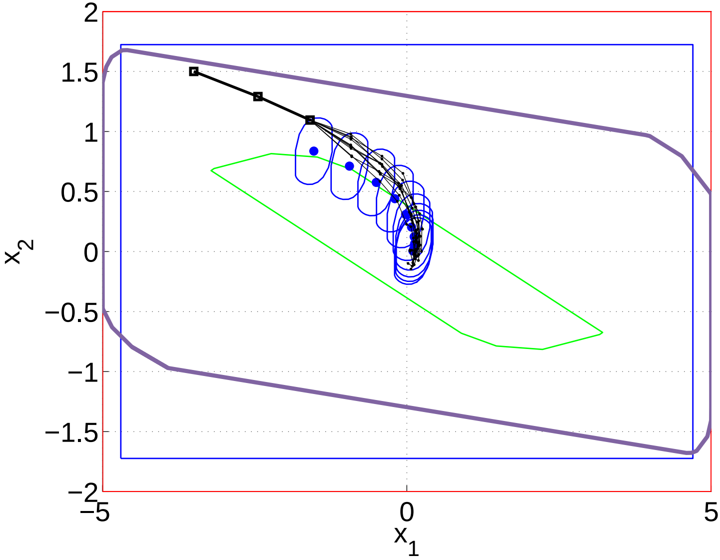
Tubes - Example



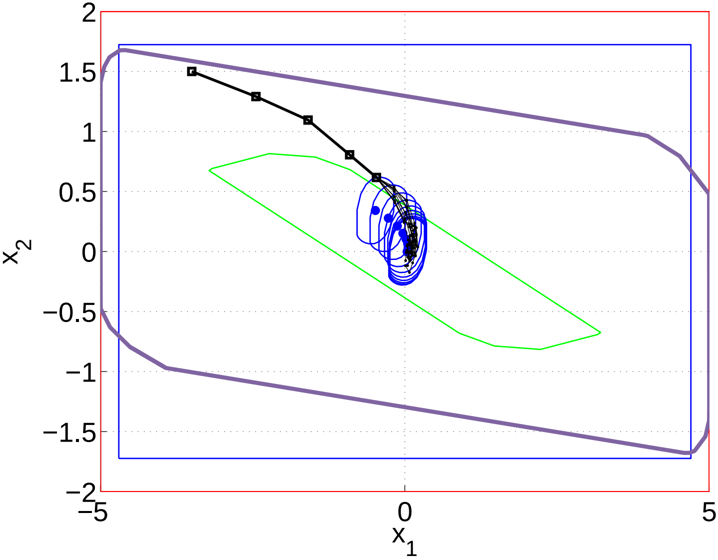
Tubes - Example



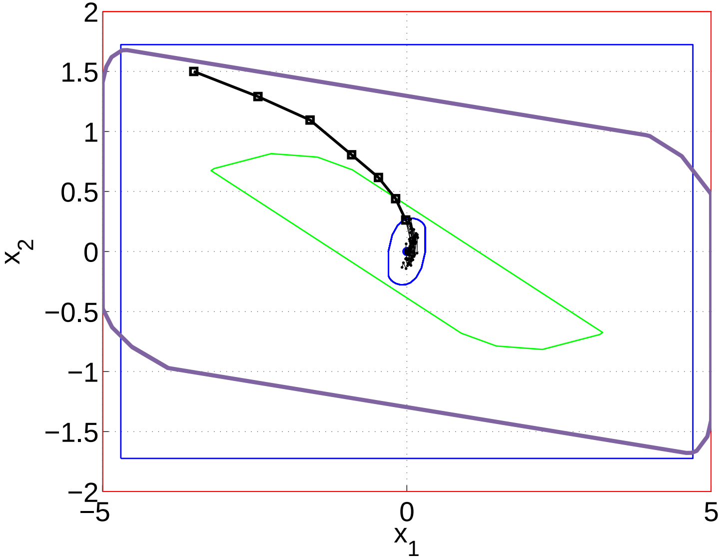
Tubes - Example



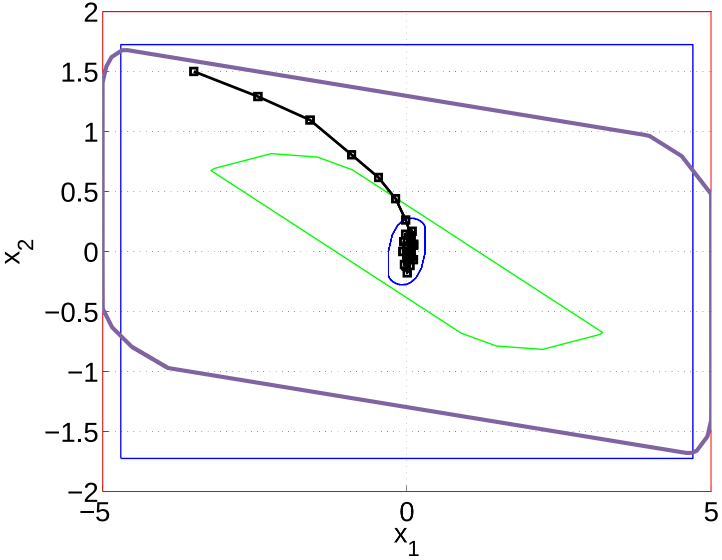
Tubes - Example



Tubes - Example



Tubes - Example



Tube MPC - Summary

Idea:

- Split input into two parts: One to steer system (v), one to compensate for the noise (Ke)

$$u = Ke + v$$

- Optimize for the nominal trajectory, ensuring that any deviations stay within constraints

Benefits:

- Less conservative than open-loop robust MPC (we're now actively compensating for noise in the prediction)
- Works for unstable systems
- Optimization problem to solve is simple

Cons:

- Sub-optimal MPC (optimal is extremely difficult)
- Reduced feasible set when compared to nominal MPC
- We need to know what $\mathbb{W}$ is (this is usually not realistic)

Outline

Nominal MPC with Noise

We want to control the noisy system:

$$x^+ = Ax + Bu + w$$

What happens if we just ignore the noise and hope for the best?

Setup and solve a standard MPC problem:

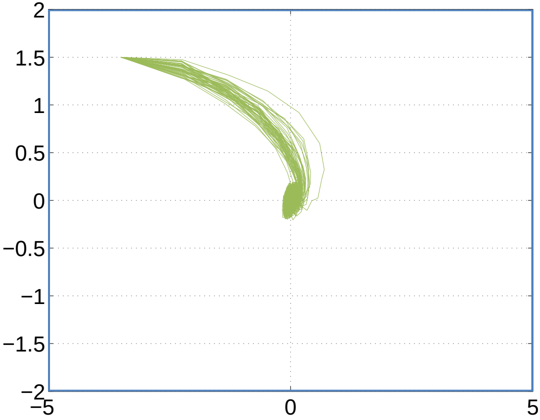
$$\begin{aligned} V^\star(x_0) = \min_{\vec{u}} \; & \sum_{i=0}^{N-1} l(x_i, u_i) + V_f(x_N) \\ \text{subj. to} \quad & x_{i+1} = Ax_i + Bu_i \\ & (x_i, u_i) \in \mathcal{X} \times \mathcal{U} \\ & x_N \in \mathcal{X}_f \end{aligned}$$

Our closed-loop system is now:

$$x^+ = Ax + Bu_0^\star(x) + w$$

Example

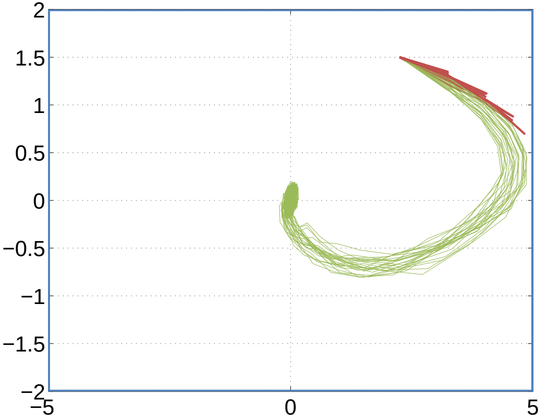
Consider the same example again, with the same noise, but now we just pretend it's not there in the controller.



- 100 trajectories with different noise realizations
- Seems to work fine?!

Example

Consider the same example again, with the same noise, but now we just pretend it's not there in the controller.



- 100 trajectories with different noise realizations
- Seems to work fine?!
- Can no longer be certain it will work!
- For some states it will work sometimes

How do we formalize this idea?

What Happens to Our Lyapunov Function?

Recall: The optimal cost $V^\star(x)$ is a Lyapunov function for the nominal system

$$V^\star(Ax + Bu^\star(x)) - V^\star(x) \leq -l(x, u^\star(x))$$

However, our state at the next point in time is now

$$x^+ = Ax + Bu^\star(x) + w$$

Do we still have a Lyapunov decrease?

What Happens to Our Lyapunov Function?

Assume: Optimal cost $V^\star$ is continuous[2]

$$|V^\star(Ax + Bu^\star(x) + w) - V^\star(Ax + Bu^\star(x))| \\ \leq \gamma \|Ax + Bu^\star(x) + w - (Ax + Bu^\star(x))\| = \gamma \|w\|$$

Our Lyapunov decrease can be bounded as:

$$\begin{aligned} & V^\star(Ax + Bu^\star(x) + w) - V^\star(x) \\ & = V^\star(Ax + Bu^\star(x) + w) - V^\star(x) - V^\star(Ax + Bu^\star(x)) + V^\star(Ax + Bu^\star) \\ & \leq V^\star(Ax + Bu^\star(x)) - V^\star(x) + \gamma \|w\| \\ & \leq -l(x, u^\star(x)) + \gamma \|w\| \end{aligned}$$

- Amount of decrease grows with $\|x\|$
- Amount of increase is upper bounded by $\max \{\|w\| \mid w \in \mathbb{W}\}$

Therefore we will move towards the origin until there is a balance between the size of x and the size of w

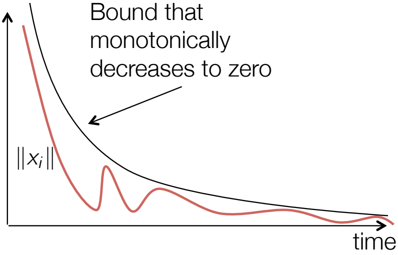
[2]True for linear systems, convex constraints and continuous stage costs.

Input-to-State Stability

What we have shown is that our system is **Input-to-State Stable**.

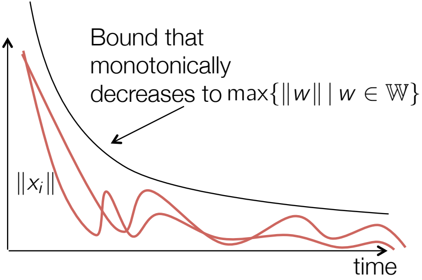
Much more general theory than what is given here[3]

Asymptotic stability



System converges to zero

ISS stability



Converges to set around zero, who's size is determined by size of the noise

[3] Limon, D., Alamo, T., Raimondo, D. M., Muñoz de la Peña, D., Bravo, J. M., Ferramosca, A., and Camacho, E. F. (2009). Input-to-State Stability: A Unifying Framework for Robust Model Predictive Control. In L. Magni, D. M. Raimondo, & F. Allgöwer (Eds.), Nonlinear Model Predictive Control (Vol. 384, pp. 1-26). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-01094-1

Nominal MPC for Uncertain Systems - Summary

Idea

- Ignore the noise and hope it works

Benefits

- Simple
- No knowledge of the noise set $\mathbb{W}$ required - 'just works'
- Often very effective in practice (this is what most practitioners do anyway)
- Feasible set is large (we can find a solution, but it may be garbage)
- Region of attraction may be larger than other approaches

Cons

- Very difficult to determine region of attraction (set of states in which the controller works)
- Hard to tune - no obvious way to tradeoff robustness against performance

Robust MPC for Uncertain Systems - Summary

Idea

- Compensate for noise in prediction to ensure all constraints will be met

Cons

- Complex (some schemes are simple to implement, like tubes, but complex to understand)
- Must know the largest noise $\mathbb{W}$
- Often very conservative
- Feasible set may be small

Benefits

- Feasible set is invariant - we know exactly when the controller will work
- Easier to tune - knobs to tradeoff robustness against performance