Model Predictive Control

Lecture: Model Predictive Control

Colin Jones

Laboratoire d'Automatique, EPFL

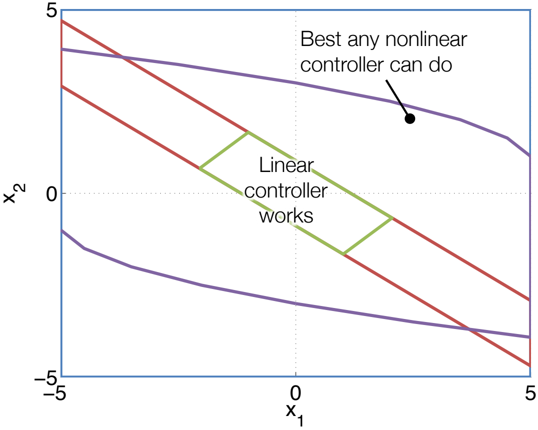
Recap: Objectives of Constrained Control

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

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

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

Limitations of Linear Controllers



System:

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u$$

Constraints:

$$\mathbb{X} := \{x \mid \|x\|_\infty \leq 5\}$$
$$\mathbb{U} := \{u \mid \|u\|_\infty \leq 1\}$$

Consider an LQR controller, with $Q = I$, $R = 1$.

Does linear control work?

Yes, but the region where it works is very small

Use nonlinear control (MPC) to increase the region of attraction

Invariance and Controlled Invariance

- Invariant set
  - Region where **autonomous** system satisfies constraints **for all time**
- Control invariant set
  - Region where there **exists a controller** so that the system satisfies the constraints **for all time**

If we have a controlled invariant set $C \subset \mathbb{X}$, we can generate a controller

$$\kappa(x) := \operatorname{argmin} \{ g(x,u) \mid f(x,u) \in C, \ \ u \in \mathbb{U} \}$$

so that for all $x \in C$, $f(x, \kappa(x)) \in C \subset \mathbb{X}$ and $\kappa(x) \in \mathbb{U}$.

Control invariant sets are almost always too complex to compute.

- MPC is a method of **implicitly** describing a control invariant set that is easy to represent and compute!

Outline

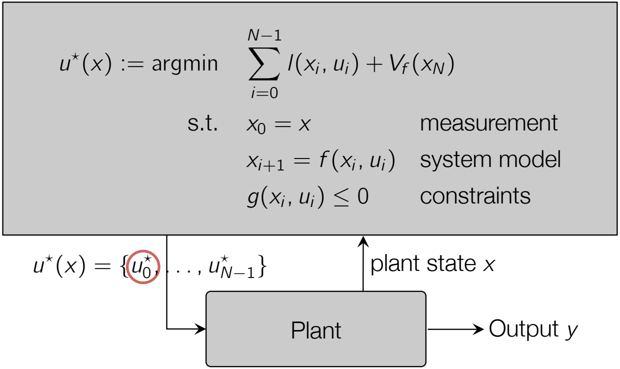
1. MPC: Key Points Illustrated

2. Stability and Invariance of MPC

3. Designing MPC to be Stabilizing and Invariant

4. Implementation of Linear MPC

MPC: Optimization in the loop



At each sample time:

- Measure /estimate current state
- Find the optimal input sequence for the entire planning window N
- Implement only the first control action

History of MPC

- Original concept: Propoi in 1963
- Mid-70's: Richalet proposed the MPC technique (called it "Model Predictive Heuristic Control (MPHC)")
- Late 70's: Cutler and Ramaker introduced Dynamic Matrix Control (DMC). Hugely successful in the petro-chemical industry.
  - Many methods followed: e.g., Quadratic Dynamic Matrix Control (QDMC), Adaptive Predictive Control (APC), Generalized Predictive Control (GPC), Sequential Open Loop Optimization (SOLO), ...
  - Constraints were generally treated in an ad-hoc fashion
- Mid-90's: an extensive theoretical effort devoted to provide conditions for guaranteeing feasibility and closed-loop stability
- 00's: development of tractable robust MPC approaches; nonlinear and hybrid MPC; MPC for very fast systems
- 10's: stochastic MPC; distributed large-scale MPC; economic MPC

Receding Horizon Control : The Motivation

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

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

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

In this lecture, we will demonstrate that these objectives can be met in a predictive control framework.

(and later)

5. Is robust to noise
6. Can be computed efficiently and reliably for a wide range of systems

Optimal Control (Want we'd like to solve)

Infinite horizon optimal control

$$
\begin{aligned}
V^\star(x_0) = \min \; & \sum_{i=0}^{\infty} l(x_i, u_i) \\
\text{s.t. } & x_{i+1} = f(x_i, u_i) \\
& (x_i, u_i) \in \mathbb{X}, \mathbb{U}
\end{aligned}
$$

- **Stage cost** $l(x, u)$ describes "cost" of being in state x and applying input u
- Optimizing over a trajectory provides a **tradeoff between short- and long-term benefits** of actions
- We'll see that such a control law has many beneficial properties...
  ... but we can't compute it: there are an **infinite number of variables**

Predictive Control (What we can sometimes solve)

Finite-time optimal control

$$V_N^\star(x_0) = \min \sum_{i=0}^{N-1} l(x_i, u_i) + V_f(x_N) \tag{1}$$

$$\text{s.t. } x_{i+1} = f(x_i, u_i)$$

$$(x_i, u_i) \in \mathbb{X}, \mathbb{U}$$

$$x_N \in \mathcal{X}_f$$

Truncate after a finite horizon:

- V_f : Approximates the 'tail' of the cost
- $\mathcal{X}_f$: Approximates the 'tail' of the constraints

Optimal control law: $\kappa_N(x) := u_0^\star$
where $u^\star := \left\{u_0^\star, \ldots, u_{N-1}^\star\right\}$ is the optimizer of (1)

Nonlinear MPC (NMPC) Properties

Pros

- Any model
  - linear
  - nonlinear
  - single/multivariable
  - time delays
  - constraints
  - etc
- Any objective:
  - sum of squared errors
  - sum of absolute errors (i.e., integral)
  - worst error over time
  - economic objective
  - etc

Cons

- Very computationally demanding in the general case
- May or may not be stable
- May or may not be invariant

This lecture:
- Conditions ensuring invariance and stability **by design**
- Systems for which optimization is computationally tractable

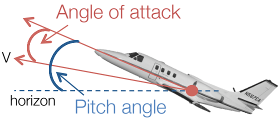
Example: Cessna Citation Aircraft

Linearized continuous-time model:

(at altitude of 5000m and a speed of 128.2 m/sec)

$$\dot{x} = \begin{bmatrix} -1.2822 & 0 & 0.98 & 0 \\ 0 & 0 & 1 & 0 \\ -5.4293 & 0 & -1.8366 & 0 \\ -128.2 & 128.2 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} -0.3 \\ 0 \\ -17 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x$$



- Input: elevator angle
- States: x_1: angle of attack, x_2: pitch angle, x_3: pitch rate, x_4: altitude
- Outputs: pitch angle and altitude
- Constraints:elevator angle ± 0.262rad ($\pm 15°$), elevator rate ± 0.524rad ($\pm 60°$), pitch angle ± 0.349 ($\pm 39°$)

Open-loop response is unstable (open-loop poles: $0, 0, -1.5594 \pm 2.29i$)

LQR and Linear MPC with Quadratic Cost

- Quadratic performance measure
- Linear system dynamics
- Linear constraints on inputs and states

LQR	**MPC**
$$\begin{aligned} J^\infty(x) = \min_{x,u} \ & \sum_{i=0}^{\infty} x_i^T Q x_i + u_i^T R u_i \\ \text{s.t.} \ & x_{i+1} = A x_i + B u_i \\ & x_0 = x \end{aligned}$$	$$\begin{aligned} J^\star(x) = \min_{x,u} \ & \sum_{i=0}^{N-1} {x_i}^T Q x_i + {u_i}^T R u_i \\ \text{s.t.} \ & x_{i+1} = A x_i + B u_i \\ & x_0 = x \\ & b \geq C x_i + D u_i \end{aligned}$$

Assume: $Q = Q^T \succeq 0$, $R = R^T \succ 0$

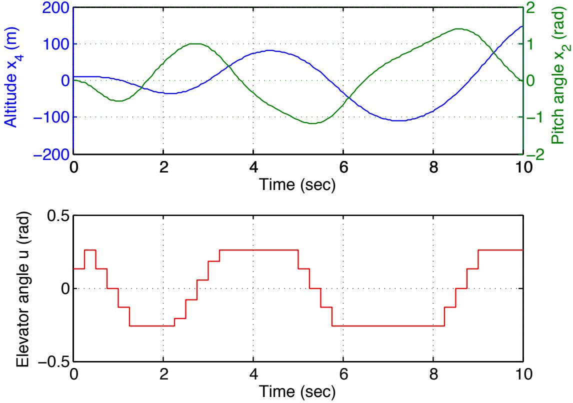
Example: LQR with saturation

Linear quadratic regulator with saturated inputs.

At time $t = 0$ the plane is flying with a deviation of $10m$ of the desired altitude, i.e. $x_0 = [0; 0; 0; 10]$

Problem parameters:

$$\boxed{\text{Sampling time } 0.25\text{sec}, \; Q = I, \; R = 10}$$



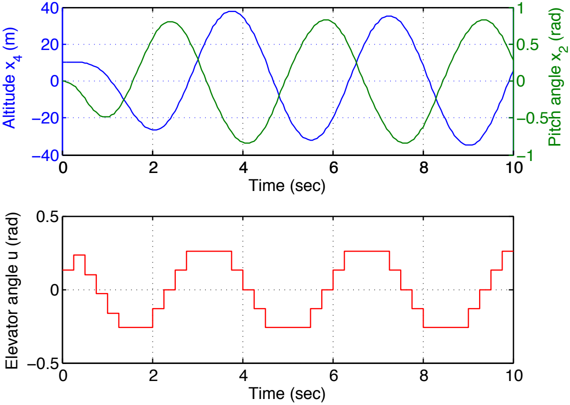
- Closed-loop system is unstable
- Applying LQR control and saturating the controller can lead to instability!

Example: MPC with Bound Constraints on Inputs

MPC controller with input constraints $|u_i| \leq 0.262$

Problem parameters:

$\boxed{\text{Sampling time 0.25sec, } Q = I,\ R = 10,\ N = 10}$



The MPC controller uses the knowledge that the elevator will saturate, but it does not consider the rate constraints.

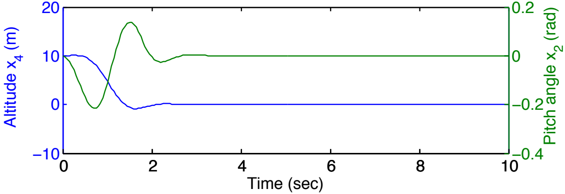
$\Rightarrow$ System does not converge to desired steady-state but to a limit cycle

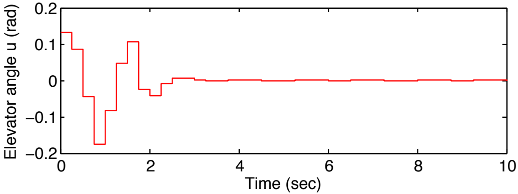
Example: MPC with all Input Constraints

MPC controller with input constraints $|u_i| \leq 0.262$
and rate constraints $|\dot{u}_i| \leq 0.349$
approximated by $|u_k - u_{k-1}| \leq 0.349\,T_s$

Problem parameters:

$$\boxed{\begin{array}{l}\text{Sampling time } 0.25\text{sec}, \\ Q = I,\ R = 10,\ N = 10\end{array}}$$





The MPC controller considers all constraints on the actuator

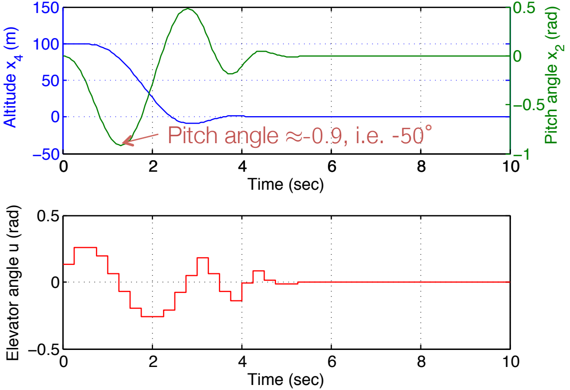
- Closed-loop system is stable
- Efficient use of the control authority

Example: Inclusion of state constraints

MPC controller with input constraints $|u_i| \leq 0.262$
and rate constraints $|\dot{u}_i| \leq 0.349$
approximated by $|u_k - u_{k-1}| \leq 0.349\,T_s$

Problem parameters:

$$\boxed{\begin{array}{l}\text{Sampling time 0.25sec,}\\ Q = I,\ R = 10,\ N = 10\end{array}}$$



Increase step:

At time $t = 0$ the plane is flying with a deviation of 100m of the desired altitude, i.e.
$x_0 = [0; 0; 0; 100]$

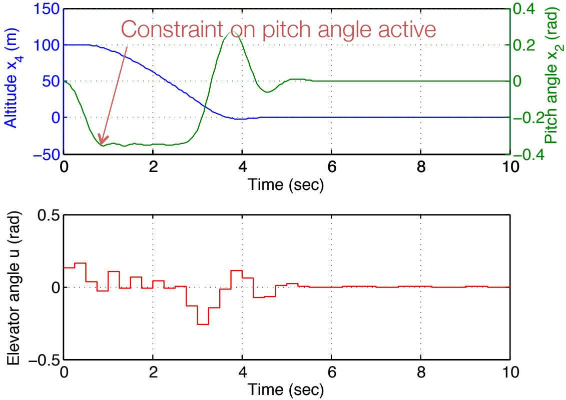
- Pitch angle too large during transient

Example: Inclusion of state constraints

MPC controller with input constraints $|u_i| \leq 0.262$
and rate constraints $|\dot{u}_i| \leq 0.349$
approximated by $|u_k - u_{k-1}| \leq 0.349\, T_s$

Problem parameters:

$$\boxed{\begin{array}{l}\text{Sampling time } 0.25\text{sec},\\ Q = I,\ R = 10,\ N = 10\end{array}}$$



Add state constraints for passenger comfort:

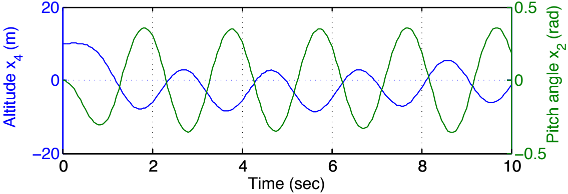
$$|x_2| \leq 0.349$$

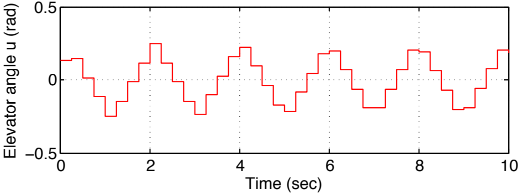
Example: Short horizon

MPC controller with input constraints $|u_i| \leq 0.262$ and rate constraints $|\dot{u}_i| \leq 0.349$ approximated by $|u_k - u_{k-1}| \leq 0.349\,T_s$

Problem parameters:

Sampling time 0.25sec, $Q = I$, $R = 10$, $N = 4$





Decrease in the prediction horizon causes loss of the stability properties

Next: How to ensure stability and constraint satisfaction for all choices of Q, R and N.

Outline

Loss of Feasibility and Stability

What can go wrong with "standard" MPC?

- No feasibility guarantee, i.e., the MPC problem may not have a solution
- No stability guarantee, i.e., trajectories may not converge to the origin

$$
\begin{aligned}
\min_{x,u} \; & \sum_{i=0}^{N-1} {x_i}^T Q x_i + {u_i}^T R u_i \\
\text{s.t. } \; & x_{i+1} = A x_i + B u_i \\
& b \geq C x_i + D u_i
\end{aligned}
$$

Definition: Feasible set

The **feasible set** $\mathcal{X}_N$ is defined as the set of initial states x for which the MPC problem with horizon N is feasible, i.e.

$\mathcal{X}_N := \{x \mid \exists [u_0, \ldots, u_{N-1}]$ such that $Cu_i + Dx_i \leq b, i = 1, \ldots, N\})$

Example: Loss of feasibility

Consider the double integrator $\quad x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$

subject to the input constraints $\quad -0.5 \le u \le 0.5$

and the state constraints $\quad \begin{bmatrix} -5 \\ -5 \end{bmatrix} \le x \le \begin{bmatrix} 5 \\ 5 \end{bmatrix}$

Parameters: $N = 3$, $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $R = 10$

Time step 1:

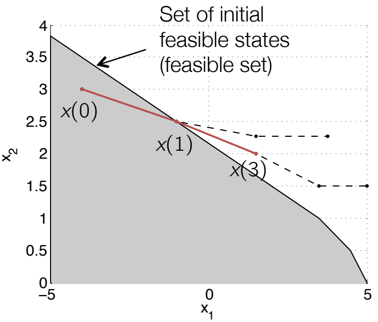
$$x_0 = [-4; 4], \quad u_0^\star(x) = -0.5$$

Time step 2:

$$x_0 = [0; 3], \quad u_0^\star(x) = -0.5$$

Time step 3:

$$x_0 = [3; 2], \quad \text{Problem infeasible}$$



Example: Loss of stability

Consider the unstable system $$x^+ = \begin{bmatrix} 2 & 1 \\ 0 & 0.5 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

subject to the input constraints $-1 \leq u \leq 1$

and the state constraints $$\begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix}$$

Parameters: $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

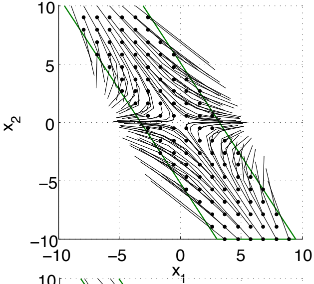
Investigate the stability properties for different horizons N and weights R by solving the finite-horizon MPC problem in a receding horizon fashion...

Example: Loss of stability

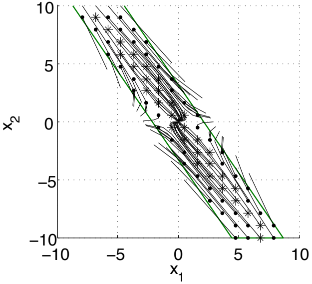
1. $R = 10,\ N = 2$
2. $R = 2,\ N = 3$
3. $R = 1,\ N = 4$

- $*$ Initial points with convergent trajectories
- $\circ$ Initial points that diverge

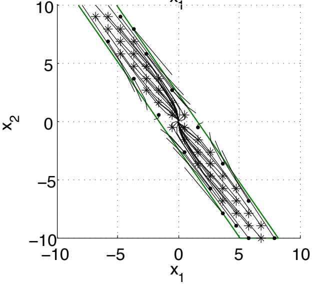
①



②



③



Parameters have complex effect on closed-loop trajectory

Feasibility and stability in MPC - Main Idea

Main idea: Introduce terminal cost and constraints to explicitly ensure stability and feasibility:

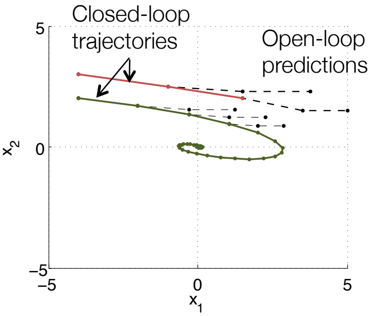
$$
\begin{aligned}
J^*(x) = \min_{\mathrm{x,u}} \quad & \sum_{i=0}^{N-1} x_i^T Q x_i + u_i^T R u_i + \boxed{x_N^T P x_N} \quad \text{Terminal cost} \\
\text{s.t.} \quad & x_{i+1} = A x_i + B u_i \\
& C x_i + D u_i \leq b \\
& \boxed{x_N \in \mathcal{X}_f} \quad \text{Terminal constraint} \\
& x_0 = x
\end{aligned}
$$

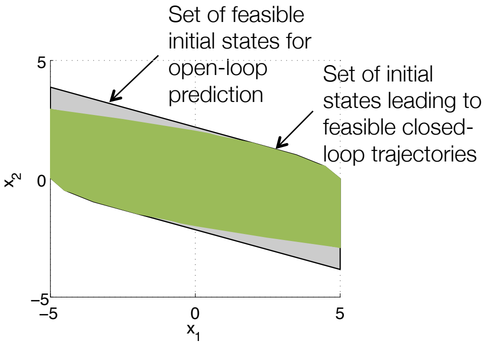
The values of P and $\mathcal{X}_f$ are chosen to **simulate an infinite horizon**.

Terminal set and cost: Main idea

Problems originate from the use of a 'short sight' strategy

⇒ Finite horizon causes deviation between the open-loop prediction and the closed-loop system:





Ideally we would solve the MPC problem with an infinite horizon, but that is computationally intractable

Design finite horizon problem such that it approximates the infinite horizon

How to choose terminal cost

We can split the infinite horizon problem into two subproblems:

① Up to time $k = N$, where the constraints may be active

$$
\begin{aligned}
J^*(x) = \min_{x,u} \quad & \sum_{i=0}^{N-1} x_i^T Q x_i + u_i^T R u_i \\
\text{s.t.} \quad & x_{i+1} = A x_i + B u_i \\
& C x_i + D u_i \leq b \\
& x_0 = x
\end{aligned}
$$

② For $k > N$, where there are no constraints active

$$
\underbrace{+ \min_{x,u} \sum_{i=N}^{\infty} x_i^T Q x_i + u_i^T R u_i \quad \text{s.t.} \quad x_{i+1} = A x_i + B u_i}_{\text{Unconstrained LQR starting from state } x_N} \ ,
$$

$$
\boxed{+ \ {x_N}^T P x_N}
$$

Unconstrained LQR starting from state x_N

- Bound the tail of the infinite horizon cost from N to ∞ using the LQR control law $u = K_{LQR} x$
- ${x_N}^T P x_N$ is the corresponding infinite horizon cost P is the solution of the discrete-time algebraic Riccati equation

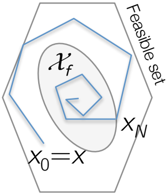
Choice of N such that constraint satisfaction is guaranteed?

How to choose terminal set

Terminal constraint provides a sufficient condition for constraint satisfaction :

$$
\begin{aligned}
J^*(x) = \min_{x,u} \quad & \sum_{i=0}^{N-1} x_i^T Q x_i + u_i^T R u_i + \boxed{x_N^T P x_N} \\
\text{s.t.} \quad & x_{i+1} = A x_i + B u_i \\
& C x_i + D u_i \leq b \\
& \boxed{x_N \in \mathcal{X}_f} \\
& x_0 = x
\end{aligned}
$$

Infinite horizon cost starting from x_N



- All input and state constraints are satisfied for the closed-loop system using the LQR control law for $x \in \mathcal{X}_f$
- Terminal set is often defined by linear or quadratic constraints

→ The bound holds in the **terminal set** and is used as a **terminal cost**

→ The terminal set defines the **terminal constraint**

In the following: Show that this problem setup provides feasibility and stability

Outline

Formalize Goals: Feasibility and Stability

Goal 1: Feasibility for all time

Definition: Recursive feasibility

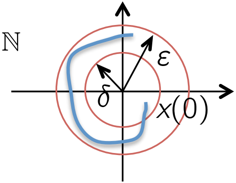
The MPC problem is called **recursively feasible**, if for all feasible initial states feasibility is guaranteed at every state along the closed-loop trajectory.

Goal 2: Stability

Definition: Lyapunov stability

The equilibrium point at the origin of system $x_{k+1} = Ax_k + B\kappa(x_k) = f_\kappa(x_k)$ is said to be **(Lyapunov) stable** in $\mathcal{X}$ if for every $\epsilon > 0$, there exists a $\delta(\epsilon) > 0$ such that, for every $x(0) \in \mathcal{X}$:

$$\|x(0)\| \leq \delta(\epsilon) \Rightarrow \|x(k)\| < \epsilon \; \forall k \in \mathbb{N}$$



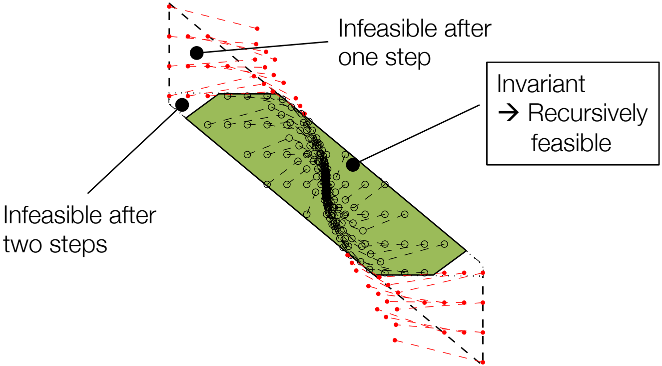
Reminder: Invariant sets

Definition: Invariant set

A set $\mathcal{O}$ is called **positively invariant** for system $x(k+1) = f_\kappa(x(k))$, if

$$x(k) \in \mathcal{O} \Rightarrow f_\kappa(x(k)) \in \mathcal{O}, \quad \forall k \in \mathbb{N}$$

The positively invariant set that contains every closed positively invariant set is called the maximal positively invariant set $\mathcal{O}_\infty$.

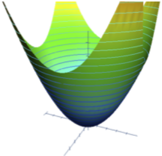


Reminder: Lyapunov Stability

Lyapunov function

Let $\mathcal{X}$ be a positively invariant set for system $x(k+1) = f_\kappa(x(k))$ containing a neighborhood of the origin in its interior. A function $V : \mathcal{X} \to \mathbb{R}_+$[1] is called a **Lyapunov function** in $\mathcal{X}$ if for all $x \in \mathcal{X}$:

$$V(x) > 0 \, \forall x \neq 0, \; V(0) = 0,$$
$$V(x(k+1)) - V(x(k)) \leq 0$$



Theorem: *(e.g., [Vidyasager, 1993])*

If a system admits a Lyapunov function in $\mathcal{X}$, then the equilibrium point at the origin is **(Lyapunov) stable** in $\mathcal{X}$.

---

[1] For simplicity it is assumed that $V(x)$ is continuous. This assumption can be relaxed by requiring an additional state dependent upper bound on $V(x)$, see e.g. [Rawlings & Mayne, 2009]

Stability and Feasibility of MPC : The Proof

Main steps:

- Prove recursive feasibility by showing the existence of a feasible control sequence at all time instants when starting from a feasible initial point
- Prove stability by showing that the optimal cost function is a Lyapunov function

We will discuss two main cases in the following:

1. Terminal constraint at zero: $x_N = 0$
2. Terminal constraint in some (convex) set: $x_N \in \mathcal{X}_f$

For simplicity, we use the more general notation:

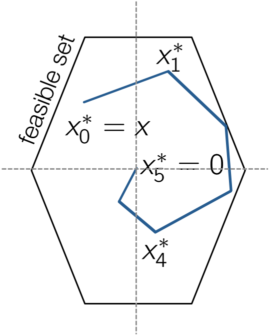
$$J^*(x) = \min_{\mathbf{x},\mathbf{u}} \sum_{i=0}^{N-1} \underbrace{l(x_i, u_i)}_{\text{stage cost}} + \underbrace{V_f(x_N)}_{\text{terminal cost}}$$

(In the quadratic case: $l(x_i, u_i) = x_i^T Q x_i + u_i^T R u_i,\ V_f(x_N) = x_N^T P x_N$)

Stability of MPC - Zero terminal state constraint

Terminal constraint $x_N = 0$

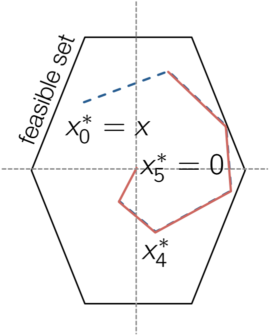
- Assume feasibility of x and let $[u_0^*,\ u_1^*,\ \dots,\ u_{N-1}^*]$ be the optimal control sequence computed at x



Stability of MPC - Zero terminal state constraint

Terminal constraint $x_N = 0$

- Assume feasibility of x and let $[u_0^*,\ u_1^*,\ \ldots,\ u_{N-1}^*]$ be the optimal control sequence computed at x
- At x^+ the control sequence $[u_1^*,\ u_2^*,\ \ldots,\ u_{N-1}^*,\ 0]$ is feasible (apply 0 control input $\Rightarrow x_{N+1} = 0$)



$\Rightarrow$ **Recursive feasibility ✔**

Stability of MPC - Zero terminal state constraint

Terminal constraint $x_N = 0$

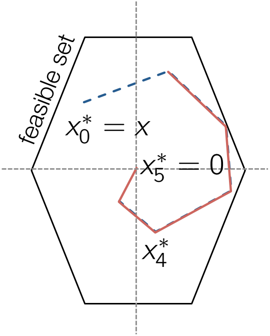
Goal: Show $J^*(x) - J^*(x^+) < 0$

$$J^*(x_0) = \sum_{i=0}^{N-1} l(x_i^*, u_i^*)$$

$$J^*(x_1) \leq \tilde{J}(x_1) = \sum_{i=1}^{N} l(x_i^*, u_i*)$$

$$= \sum_{i=0}^{N-1} l(x_i^*, u_i^*) - l(x_0, u_0^*) + l(x_N, u_N)$$

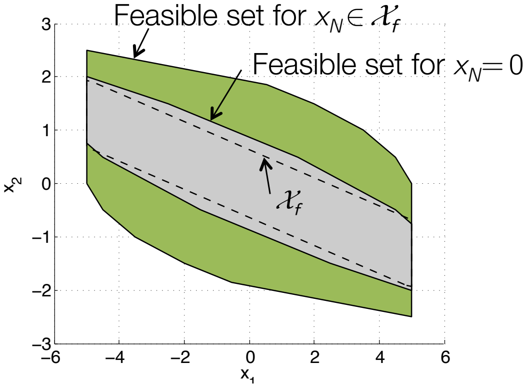
$$= J^*(x_0) - \underbrace{l(x, u_0^*)}_{\text{Subtract cost at stage 0}} + \underbrace{l(0,0)}_{\text{Add cost for staying at 0}}$$



$\Rightarrow$ $J^\star(x)$ **is a Lyapunov function** $\rightarrow$ **(Lyapunov) Stability ✔**

Extension to More General Terminal Sets

Problem: The terminal constrain $x_N = 0$ reduces the size of the feasible set
Goal: Use convex set for $\mathcal{X}_f$ to increase the region of attraction



Double integrator

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$\begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix}$$

$$-0.5 \leq u \leq 0.5$$

$$N = 5,\ Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ R = 10$$

Goal: Generalize proof to the constraint $x_N \in \mathcal{X}_f$

Stability of MPC - Main Result

Standing assumptions hold:

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** under the local control law $\kappa_f(x)$:

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

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

   $$\mathcal{X}_f \subseteq \mathbb{X},\ \kappa_f(x) \in \mathbb{U} \text{ for all } x \in \mathcal{X}_f$$

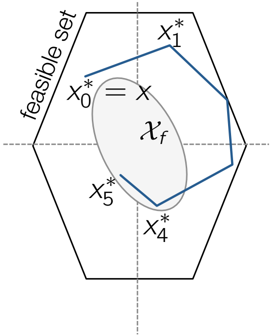
3. Terminal cost is a continuous **Lyapunov function** in the terminal set $\mathcal{X}_f$:

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

Thm: The closed-loop system under the MPC control law $u_0^\star(x)$ is stable and the system $x^+ = Ax + Bu_0^\star(x)$ is invariant in the feasible set $\mathbb{X}_N$.

Stability of MPC - Outline of the Proof

- Assume feasibility of x and let $[u_0^\star,\ u_1^\star,\ \ldots,\ u_{N-1}^\star]$ be the optimal control sequence computed at x



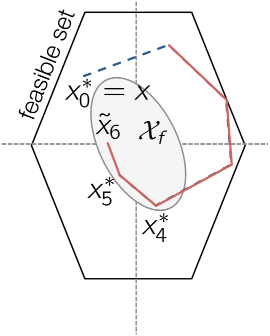
Stability of MPC - Outline of the Proof

- Assume feasibility of x and let $[u_0^\star,\ u_1^\star,\ \dots,\ u_{N-1}^\star]$ be the optimal control sequence computed at x

- At x^+, $[u_1^\star,\ u_2^\star,\ \dots,\ \kappa_f(x_N^\star)]$ is feasible:

$$x_N \text{ is in } \mathcal{X}_f \rightarrow \kappa_f(x_N^\star) \text{ is feasible}$$
$$\text{and } x_{N+1} = Ax_N^\star + B\kappa_f(x_N^\star) \text{ in } \mathcal{X}_f$$

$\Rightarrow$ **Terminal constraint provides recursive feasibility**



Stability of MPC - Outline of the Proof

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

Feasible, sub-optimal sequence for x_1 : $[u_1^\star,\ u_2^\star,\ \dots,\ \kappa_f(x_N^\star)]$

$$\begin{aligned}
J^*(x_1) &\leq \sum_{i=1}^{N} l(x_i^*, u_i^*) + V_f(\tilde{x}_{N+1}) \\
&= \sum_{i=0}^{N-1} l(x_i^*, u_i^*) + V_f(x_N^*) - l(x_0^*, u_0^*) + V_f(\tilde{x}_{N+1}) - V_f(x_N^*) + l(x_N^*, \kappa_f(x_N^*)) \\
&= J^*(x_0) - l(x, u_0^*) + \underbrace{V_f(\tilde{x}_{N+1}) - V_f(x_N^*) + l(x_N^*, \kappa_f(x_N^*))}_{V_f(x) \text{ is a Lyapunov function: } \leq 0}
\end{aligned}$$

$J^\star(x)$ **is a Lyapunov function $\rightarrow$ (Lyapunov) Stability**

Stability of MPC - Remarks

- The terminal set $\mathcal{X}_f$ and the terminal cost ensure recursive feasibility and stability of the closed-loop system.
  But: the terminal constraint reduces the region of attraction.
  (Can extend the horizon to a sufficiently large value to increase the region)

Are terminal sets used in practice?

- Generally not...
  - Not well understood by practicioners
  - Requires advanced tools to compute (polyhedral computation or LMI)
- Reduces region of attraction
  - A 'real' controller must provide *some* input in *every* circumstance
- Often unnecessary
  - Stable system, long horizon $\rightarrow$ will be stable and feasible in a (large) neighbourhood of the origin

Proof of Asymptotic Stability

Definition: Asymptotic stability

Given a positively invariant set $\mathcal{X}$ including the origin as an interior-point, the equilibrium point at the origin of system $x_{k+1} = f_\kappa(x_k)$ is said to be **asymptotically stable** in $\mathcal{X}$ if it is

- **(Lyapunov) stable**
- **attractive in** $\mathcal{X}$, i.e. $\lim_{k\to\infty} \|x_k\| = 0$ for all $x(0) \in \mathcal{X}$

Extension of Lyapunov's direct method: (see e.g. [Vidyasagar, 1993])
If the continuous Lyapunov function additionally satisfies

$$V(x_{k+1}) - V(x_k) < 0 \; \forall x \neq 0$$

then the closed loop system converges to the origin and is hence asymptotically stable.

Recall: Decrease of the optimal MPC cost was given by

$$J^*(x_{k+1}) - J^*(x_k) \leq -l(x_k, u_0^*)$$

where the stage cost was assumed to be positive and only 0 at 0.

$\Rightarrow$ The closed-loop system under the MPC control law is asymptotically stable

Extension to Nonlinear MPC

Consider the nonlinear system dynamics: $x^+ = f(x, u)$

Nonlinear MPC problem

$$
\begin{aligned}
J^*(x) = \min_{\mathbf{x},\mathbf{u}} \quad & \sum_{i=0}^{N-1} l(x_i, u_i) + V_f(x_N) \\
\text{s.t.} \quad & x_{i+1} = f(x_i, u_i) \\
& g(x_i, u_i) \leq 0 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x
\end{aligned}
$$

- Presented assumptions on the terminal set and cost did not rely on linearity
- Lyapunov stability is a general framework to analyze stability of nonlinear dynamic systems

$\rightarrow$ Results can be directly extended to nonlinear systems.

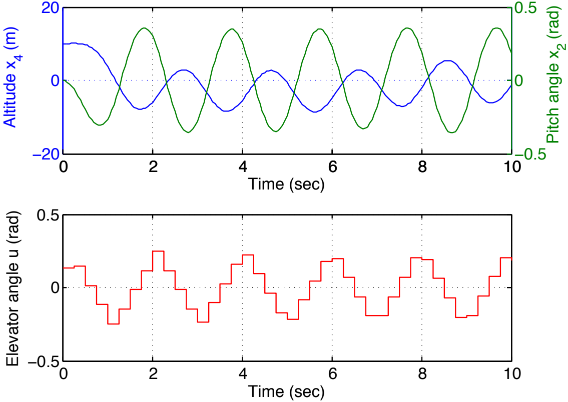
However, computing the sets $\mathcal{X}_f$ and function V_f can be very difficult!

Example: Short horizon

MPC controller with input constraints $|u_i| \leq 0.262$ and rate constraints $|\dot{u}_i| \leq 0.349$ approximated by $|u_k - u_{k-1}| \leq 0.349 T_s$

Problem parameters:

$\boxed{\text{Sampling time } 0.25\text{sec}, \; Q = I, \; R = 10, \; N = 4}$



Decrease in the prediction horizon causes loss of the stability properties

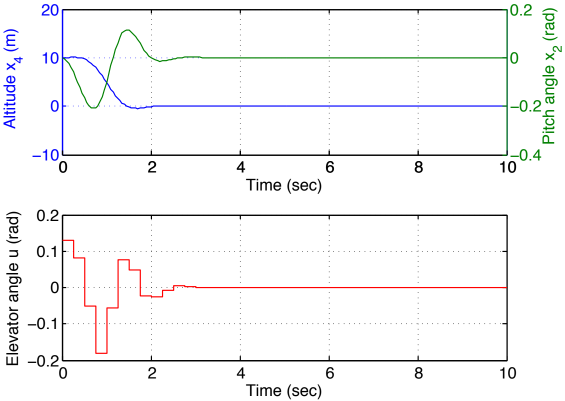
Next: How to ensure stability and constraint satisfaction for all choices of Q, R and N.

Example: Short horizon

MPC controller with input constraints $|u_i| \leq 0.262$ and rate constraints $|\dot{u}_i| \leq 0.349$ approximated by $|u_k - u_{k-1}| \leq 0.349\,T_s$

Problem parameters:

Sampling time 0.25sec, $Q = I,\ R = 10,\ N = 4$



Inclusion of terminal cost and constraint provides stability

Summary

Finite-horizon MPC may not be stable!
Finite-horizon MPC may not satisfy constraints for all time!

- An infinite-horizon provides stability and invariance.

- We ‘fake’ infinite-horizon by forcing the final state to be in an invariant set for which there exists an invariance-inducing controller, whose infinite-horizon cost can be expressed in closed-form.

- These ideas extend to non-linear systems, but the sets are difficult to compute.

Outline

1. MPC: Key Points Illustrated

2. Stability and Invariance of MPC

3. Designing MPC to be Stabilizing and Invariant

4. Implementation of Linear MPC

Linear MPC with Quadratic Cost

Standard formulation:

- Quadratic performance measure
- Linear system dynamics
- $\mathbb{X}$, $\mathcal{X}_f$ and $\mathbb{U}$ are polyhedra

$$
\begin{aligned}
\min_{\mathbf{u}} \quad & \sum_{i=0}^{N-1} x_i{}^T Q x_i + u_i{}^T R u_i + x_N{}^T Q_f x_N \\
\text{s.t.} \quad & x_i \in \mathbb{X} && i \in \{1, \ldots, N-1\} \\
& u_i \in \mathbb{U} && i \in \{0, \ldots, N-1\} \\
& x_N \in \mathcal{X}_f \\
& x_{i+1} = A x_i + B u_i
\end{aligned}
$$

Assumptions: $Q = Q^T \succeq 0$, $Q_f = Q_f^T \succ 0$, $R = R^T \succ 0$

Next: How to write the MPC problem as a quadratic program

QP Formulation of MPC

Standard input form for QP software:

$$
\begin{aligned}
\min_{\mathbf{z}} \quad & \frac{1}{2}\mathbf{z}^T H \mathbf{z} \\
\text{s.t.} \quad & G\mathbf{z} \le g \\
& T\mathbf{z} = t
\end{aligned}
$$

Generate matrices H, G and T and vectors g and t from the optimization problem:

$$
\begin{aligned}
\min_{\mathbf{u}} \quad & \sum_{i=0}^{N-1} {x_i}^T Q x_i + {u_i}^T R u_i + {x_N}^T Q_f x_N \\
\text{s.t.} \quad & x_i \in \mathbb{X} && i \in \{1, \dots, N-1\} \\
& u_i \in \mathbb{U} && i \in \{0, \dots, N-1\} \\
& x_N \in \mathcal{X}_f \\
& x_{i+1} = A x_i + B u_i
\end{aligned}
$$

QP Formulation of MPC

Formulation of matrices H, G and T and vectors g and t:

- Define variables:

$$\mathbf{z} := \begin{bmatrix} x_1^T & \dots & x_N^T & u_0^T & \dots & u_{N-1}^T \end{bmatrix}^T$$

- Equalities (T, t) from system dynamics $x_{i+1} = Ax_i + Bu_i$:

$$T := \left[\begin{array}{ccccc:cccc} I & & & & & -B & & & \\ -A & I & & & & & -B & & \\ & -A & I & & & & & -B & \\ & & \ddots & \ddots & & & & & \ddots \\ & & & -A & I & & & & & -B \end{array}\right]$$

$$t := \begin{bmatrix} A \\ 0 \\ \vdots \\ 0 \end{bmatrix} x_0 \qquad t \text{ is a linear function of the current state } x_0!$$

QP Formulation of MPC

Inequalities $G\mathbf{z} \leq g$:

- Assume $\mathbb{X}$ and $\mathbb{U}$ given by:

$$\mathbb{X} := \{x \mid Fx \leq f\} \qquad \mathbb{U} := \{u \mid Mu \leq m\} \qquad \mathcal{X}_f := \{x \mid F_f x \leq f_f\}$$

- Form matrices G and g

$$G := \left[\begin{array}{ccccc|ccccc} F & & & & & 0 & & & & \\ & F & & & & & 0 & & & \\ & & \ddots & & & & & \ddots & & \\ & & & F & & & & & 0 & \\ & & & & F_f & & & & & 0 \\ \hline 0 & & & & & M & & & & \\ & 0 & & & & & M & & & \\ & & \ddots & & & & & \ddots & & \\ & & & 0 & & & & & M & \\ & & & & 0 & & & & & M \end{array}\right] \qquad g := \left[\begin{array}{c} f \\ f \\ \vdots \\ f \\ f_f \\ \hline m \\ m \\ \vdots \\ m \\ m \end{array}\right]$$

QP Formulation of MPC

Build cost function $\mathbf{z}^T H \mathbf{z}$ from MPC cost $\sum_{i=0}^{N-1} {x_i}^T Q x_i + {u_i}^T R u_i + {x_N}^T Q_f x_N$

$$H := \left[\begin{array}{cccc|ccc} Q & & & & & & \\ & \ddots & & & & & \\ & & Q & & & & \\ & & & Q_f & & & \\ \hline & & & & R & & \\ & & & & & \ddots & \\ & & & & & & R \end{array}\right]$$

Matlab hint:

```
H = blkdiag(kron(eye(N-1),Q), Qf, kron(eye(N),R))
```