

Lecture 2 - Mathematical Foundations

Sets, Reachability, and Set Operators

Kjell Machalowsky

Department of Engineering Cybernetics, NTNU

Fall 2026

Why we analyze sets in MPC

(1) Recursive feasibility:

If the optimization is feasible at time k , will it still be feasible at time $k + 1$, and at every future time?

If not, the controller can simply stop! $\implies$ unacceptable

(2) Stability:

Does the closed-loop system converge to the desired set-point (or terminal set), and does it stay there?

Both aspects are **not** about single trajectories. We would like to know:

- ▶ from which states our OCP is feasible,
- ▶ whether this feasibility is preserved in future steps,
- ▶ and if there are regions we will never leave.

This leads us to the analysis of sets of states.

- Plan:**
- (1) Build a mathematical toolbox for set operations
 - (2) Express properties of discrete systems via set operations

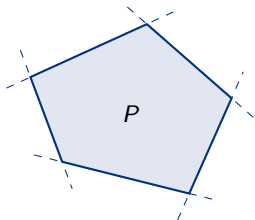
Polyhedrons and Polytopes

Polyhedron, Polytope

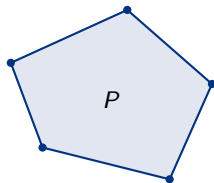
A **polyhedron** $P \subseteq \mathbb{R}^n$ is the intersection of a finite number of half-spaces:

$$P := \{x \in \mathbb{R}^n \mid Ax \leq b\}, \quad A \in \mathbb{R}^{q \times n}, \quad b \in \mathbb{R}^q.$$

A *bounded* polyhedron is called a **polytope**. We can describe a polytope either by the intersection of half-spaces ($\mathcal{H}$ -representation) or by its vertices ($\mathcal{V}$ -representation).



$\mathcal{H}$ -representation



$\mathcal{V}$ -representation

Common Set Operations

► Intersection:

$$P \cap Q := \{x \mid x \in P \wedge x \in Q\}$$

► Linear image:

$$MP := \{Mx \mid x \in P\}$$

► Pre-image:

$$M^{-1}P := \{x \mid Mx \in P\}$$

► Minkowski sum:

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

► Pontryagin difference:

$$P \ominus Q := \{x \mid x + q \in P \ \forall q \in Q\}$$

► Projection:

$$\text{proj}_x(P) := \{x \mid \exists u : (x, u) \in P\}$$

Remarks

Minkowski sum and Pontryagin difference are **not** inverse operations:

$$(P \ominus Q) \oplus Q \subseteq P \quad \text{but not necessarily} \quad (P \ominus Q) \oplus Q = P.$$

The intersection, linear image, pre-image, and projections of polyhedra are polyhedra as well.

Minkowski sum and Pontryagin difference of polytopes are polytopes.

Types of Dynamic Systems

- ▶ Autonomous system:

$$x_{k+1} = f_a(x_k), \quad f_a : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$$

- ▶ System with external inputs:

$$x_{k+1} = f(x_k, u_k), \quad f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_x}$$

In general, we apply polyhedral state and input constraints:

$$x_k \in \mathcal{X} \subseteq \mathbb{R}^{n_x}, \quad u_k \in \mathcal{U} \subseteq \mathbb{R}^{n_u} \quad \forall k \geq 0.$$

Key observation:

For a linear system with fixed feedback $u = Kx$, the closed loop is autonomous:

$$x_{k+1} = f(x_k, u_k) = Ax_k + Bu_k = Ax_k + BKx_k = (A + BK)x_k$$

One-step Reachable Sets

Successor set

For $\mathcal{S} \subset \mathbb{R}^{n_x}$, the successor set $\text{Suc}(\mathcal{S})$ is the set of states that are reachable from $\mathcal{S}$ in one step:

- ▶ autonomous: $\text{Suc}(\mathcal{S}) := \{x \mid \exists x_k \in \mathcal{S} : x = f_a(x_k)\}$
- ▶ ext. inputs: $\text{Suc}(\mathcal{S}) := \{x \mid \exists x_k \in \mathcal{S}, u_k \in \mathcal{U} : x = f(x_k, u_k)\}$

Also called the one-step **forward** reachable set.

Precursor set

Let $\mathcal{S} \subset \mathbb{R}^{n_x}$. The precursor set $\text{Pre}(\mathcal{S})$ contains all states that reach $\mathcal{S}$ in one step:

- ▶ autonomous: $\text{Pre}(\mathcal{S}) := \{x \mid f_a(x) \in \mathcal{S}\}$
- ▶ ext. inputs: $\text{Pre}(\mathcal{S}) := \{x \mid \exists u \in \mathcal{U} : f(x, u) \in \mathcal{S}\}$

Every state in $\text{Pre}(\mathcal{S})$ can be driven to $\mathcal{S}$ by a suitable controller, subject to constraints $\Rightarrow$ one-step **backward** reachable set.

Example: One-step reachable sets for an autonomous system

- ▶ Consider the 2D system: $x_{k+1} = Ax_k = \begin{bmatrix} 0.5 & 0 \\ 1 & -0.5 \end{bmatrix} x_k$, $x_k \in \mathcal{X}$
- ▶ with the constraints $\mathcal{X} := \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix} \right\}$

We derive obtain the $\mathcal{H}$ -representation of the constraint set

$$\mathcal{X} = \{x \in \mathbb{R}^2 \mid Hx \leq h\}, \quad \text{with} \quad H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad h = \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \end{bmatrix}$$

and use it to compute the Pre-set

$$\text{Pre}(\mathcal{X}) = \{x \in \mathbb{R}^2 \mid f_a(x) \in \mathcal{X}\} = \{x \in \mathbb{R}^2 \mid HAx \leq h\}$$

Example: One-step reachable sets for an autonomous system

$$\text{Pre}(\mathcal{X}) = \{x \in \mathbb{R}^2 \mid f_a(x) \in \mathcal{X}\} = \{x \in \mathbb{R}^2 \mid HA x \leq h\}$$

then simplifies to a simple matrix-multiplication

$$HAx = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 \\ 1 & -0.5 \end{bmatrix} x = \begin{bmatrix} 0.5 & 0 \\ 1 & -0.5 \\ -0.5 & 0 \\ -1 & 0.5 \end{bmatrix} x \leq \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \end{bmatrix} = h.$$

Next, we unify the computed Pre-set with the feasible set to **exclude infeasible states**

$$\text{Pre}(\mathcal{X}) \cap \mathcal{X} = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \\ -1 & -0.5 \\ -1 & 0.5 \end{bmatrix} x \leq \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \end{bmatrix} \right\}$$

Example: One-step reachable sets for an autonomous system

Analogous, we can compute the one-step forward reachable set

$$\text{Suc}(\mathcal{X}) := \{x \in \mathbb{R}^2 \mid \exists x_k \in \mathcal{X} : x = f_a(x_k)\} = \{x \in \mathbb{R}^2 \mid HA^{-1}x \leq h\}.$$

For 2D systems, calculating the inverse is rather easy.

$$HA^{-1}x = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} 2 & 0 \\ 4 & -2 \end{bmatrix}}_{=A^{-1}} x = \begin{bmatrix} 2 & 0 \\ 4 & -2 \\ -2 & 0 \\ -4 & 2 \end{bmatrix} x \leq \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \end{bmatrix} = h$$
$$\iff \begin{bmatrix} 1 & 0 \\ 1 & -0.5 \\ -1 & 0 \\ -1 & 0.5 \end{bmatrix} x \leq \begin{bmatrix} 5 \\ 2.5 \\ 5 \\ 2.5 \end{bmatrix}$$

If we have higher dimensions, the $\mathcal{V}$ -representation becomes useful:

$$\text{Suc}(\mathcal{X}) = \text{conv}(AV) = A \circ \mathcal{X}$$

Example: One-step reachable sets for an input-system

- ▶ Now, we also include external inputs
- ▶ Consider another the 2D system:

$$x_{k+1} = Ax_k + Bu_k = \begin{bmatrix} 1.5 & 0 \\ 1 & -1.5 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad x_k \in \mathcal{X}, \quad u_k \in \mathcal{U}$$

- ▶ Constraints $\mathcal{X} := \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix} \right\}$
 $\mathcal{U} := \{ u \in \mathbb{R} \mid -5 \leq u \leq 5 \}$

Again, we rewrite the constraints as polytopes in $\mathcal{H}$ -representation:

$$\mathcal{X} = \{ x \in \mathbb{R}^2 \mid Hx \leq h \}, \quad \mathcal{U} = \{ u \in \mathbb{R} \mid H_u u \leq h_u \} \quad \text{with}$$

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad h = \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \end{bmatrix}, \quad H_u = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad h_u = \begin{bmatrix} 5 \\ 5 \end{bmatrix}$$

Example: One-step reachable sets for an input-system

Looking one step forward, the successor set can be computed using the introduced notation:

$$\text{Suc}(\mathcal{X}) = (A \circ \mathcal{X}) \oplus (B \circ \mathcal{U}), \quad \text{then} \quad \text{Suc}(\mathcal{X}) \cap \mathcal{X}.$$

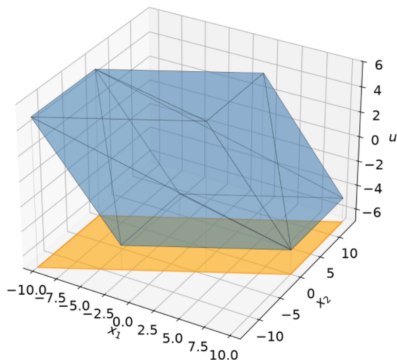
Exemplary, we will take a look at the precursor set for this example.

$$\begin{aligned} \text{Pre}(\mathcal{X}) &= (\mathcal{X} \oplus (-B \circ \mathcal{U})) \circ A \\ &= \{x \in \mathbb{R}^2 \mid \exists u \in \mathcal{U} : \underbrace{f(x, u)}_{\substack{HA & HB \\ \mathbf{0} & H_u}} \in \mathcal{X}\} \\ &= \begin{bmatrix} HA & HB \\ \mathbf{0} & H_u \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq \begin{bmatrix} h \\ h_u \end{bmatrix} \end{aligned}$$

$$\Rightarrow \begin{bmatrix} 1.5 & 0 & 1 \\ 1 & -1.5 & 0 \\ -1.5 & 0 & -1 \\ -1 & 1.5 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ u \end{bmatrix} \leq \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \\ 5 \\ 5 \end{bmatrix}$$

But this is a 3D polytope!
We need a **projection**
back to the state-space.

Example: One-step reachable sets for an input-system

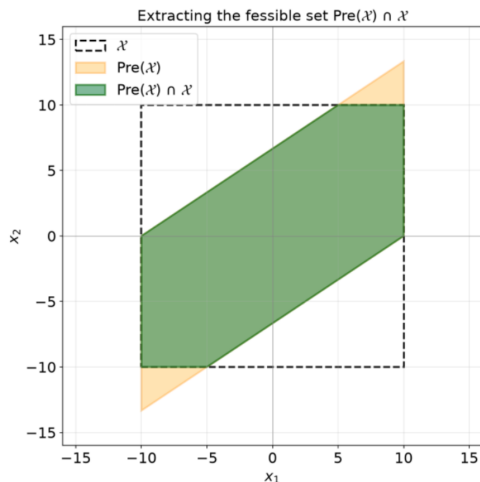


- ▶ Projection is like a shadow
- ▶ Fourier-Motzkin Elimination for polytopes
- ▶ Visualization is limited to low dimensions

We will not focus on this in detail, since many libraries exist for these tasks in practical implementation.

You need to know what is going on and which influences it might have!

Example: One-step reachable sets for an input-system



We still have to take the intersection with $\mathcal{X}$ to extract the feasible part of $\text{Pre}(\mathcal{X})$!

Positive Invariant Sets

A key property of a set in control theory is whether it is invariant:
“once inside, always inside”

Positive invariant set

A set $\mathcal{O} \subseteq \mathcal{X}$ is positively invariant for the autonomous system s.t. $x \in \mathcal{X}$ if

$$x \in \mathcal{O} \implies f_a(x) \in \mathcal{O}.$$

Equivalently: $\mathcal{O} \subseteq \text{Pre}(\mathcal{O})$ or $\text{Suc}(\mathcal{O}) \subseteq \mathcal{O}$.

Maximal positive invariant set

The maximal positive invariant set $\mathcal{O}_\infty \subseteq \mathcal{X}$ is the union of all positively invariant subsets of $\mathcal{X}$. Equivalently, it is the set of all initial states whose entire trajectory stays inside $\mathcal{X}$:

$$\mathcal{O}_\infty := \{x_0 \in \mathcal{X} \mid f_a^k(x_0) \in \mathcal{X} \ \forall k \geq 0\}.$$

Computation of $\mathcal{O}_\infty$

Since $\mathcal{O} \subseteq \text{Pre}(\mathcal{O}) \Leftrightarrow \text{Pre}(\mathcal{O}) \cap \mathcal{O} = \mathcal{O}$, this can be used to compute $\mathcal{O}_\infty$ iteratively.

Iteration for $\mathcal{O}_\infty$ (Geman-Gilbert iteration)

Set $\Omega_0 = \mathcal{X}$, $\Omega_{k+1} = \Omega_k \cap \text{Pre}(\Omega_k)$. Then $\Omega_{k+1} \subseteq \Omega_k$ and $\mathcal{O}_\infty = \bigcap_{k \geq 0} \Omega_k$. If $\Omega_{k+1} = \Omega_k$ for some finite k , the iteration has converged and $\mathcal{O}_\infty = \Omega_k$.

- ▶ This algorithm is **not guaranteed to terminate** in finitely many steps.
- ▶ Testing set equality is, in general, not trivial.
- ▶ Each iteration accumulates the intersection of half-spaces.

Control Invariant Sets

Control invariant set

A set $\mathcal{C} \subseteq \mathcal{X}$ is control invariant for the system with external inputs if

$$x \in \mathcal{C} \implies \exists u \in \mathcal{U} : f(x, u) \in \mathcal{C} \quad \text{or equivalently } \mathcal{C} \subseteq \text{Pre}(\mathcal{C})$$

This is the key ingredient for **terminal sets** in MPC!

Maximal control invariant set

The maximal control invariant set $\mathcal{C}_\infty \subseteq \mathcal{X}$ is the union of all control invariant subsets of $\mathcal{X}$: the set of initial states for which some admissible input sequence keeps the state in $\mathcal{X}$ forever.

Remark

$\mathcal{C}_\infty$ can be computed with the same iteration as $\mathcal{O}_\infty$:

The only difference is that Pre is now evaluated over an admissible control policy rather than a fixed feedback signal.

N -step Controllable Sets

Controllability asks: "Can we reach a target set $\mathcal{S}$ in N steps?"

N -step controllable set

Given a target set $\mathcal{S} \subseteq \mathcal{X}$, the N -step controllable set $\mathcal{K}_N(\mathcal{S})$ is recursively defined as:

$$\mathcal{K}_k(\mathcal{S}) := \text{Pre}(\mathcal{K}_{k-1}(\mathcal{S})) \cap \mathcal{X}, \quad \mathcal{K}_0(\mathcal{S}) := \mathcal{S}, \quad \forall k \in [1, N]$$

All states in $\mathcal{K}_N(\mathcal{S})$ will reach $\mathcal{S}$ in a maximum of N steps, respecting all constraints.

Maximal controllable set

$$\mathcal{K}_\infty(\mathcal{S}) = \bigcup_{N \geq 0} \mathcal{K}_N(\mathcal{S})$$

is the set of states that can be steered into $\mathcal{S}$ in any finite number of steps while staying in $\mathcal{X}$.

Feasible Set of MPC

We now bridge set theory and MPC.

Feasible set of MPC

Consider an MPC problem with horizon N , stage constraints $x \in \mathcal{X}$, $u \in \mathcal{U}$, and terminal set $\mathcal{X}_f \subseteq \mathcal{X}$. The set of states for which the MPC optimization is feasible is exactly the N -step controllable set to $\mathcal{X}_f$:

$$\mathcal{X}_N = \mathcal{K}_N(\mathcal{X}_f).$$

- ▶ The feasible region of MPC is a controllable set.
- ▶ Its size is governed by the **horizon** N and the choice of the **terminal set** $\mathcal{X}_f$.
- ▶ Larger N or a larger control invariant $\mathcal{X}_f$
 $\implies$ larger feasible set $\mathcal{X}_N$.

Introducing Uncertainty to Dynamic Systems

We now consider dynamic systems with bounded additive disturbance:

$$x_{k+1} = f_a(x_k) + w_k, \quad w_k \in \mathcal{W},$$

$$x_{k+1} = f(x_k, u_k) + w_k, \quad w_k \in \mathcal{W}.$$

The disturbance set $\mathcal{W} \subset \mathbb{R}^{n_x}$ is assumed.

Sources of uncertainty:

- ▶ Plant-model mismatch
- ▶ Unmeasured external inputs
- ▶ Linearization error
- ▶ Sensor/process noise

An important property of MPC: give robust guarantees despite disturbances.

Robust versions of the previous set operators require the corresponding property to hold for all admissible disturbances $w \in \mathcal{W}$, not just the nominal case.

Robust one-step Reachable Sets

Robust precursor set

The robust pre-set $\text{Pre}_{\mathcal{W}}(\mathcal{S})$ under additive disturbance is defined as

$$\text{Pre}_{\mathcal{W}}(\mathcal{S}) := \{x \in \mathcal{X} \mid \exists u \in \mathcal{U} : f(x, u) + w \in \mathcal{S} \ \forall w \in \mathcal{W}\}.$$

For a linear system $x^+ = Ax + Bu + w$, $w \in \mathcal{W}$, and target set $\mathcal{S} = \{x \in \mathcal{X} \mid Fx \leq g\}$, the robust precursor set tightens the target set by the disturbance:

$$\text{Pre}_{\mathcal{W}}(\mathcal{S}) = \text{Pre}(\mathcal{S} \ominus \mathcal{W}).$$

This is why the **Pontryagin difference** is the workhorse of robust MPC: it provides a safety margin against every future disturbance realization.

Robust Positive and Control Invariant Sets

Robust positive invariant sets, robust control invariant sets

- $\mathcal{O} \subseteq \mathcal{X}$ is **robust positively invariant** if:

$$x \in \mathcal{O} \implies f_a(x) + w \in \mathcal{O} \quad \forall w \in \mathcal{W}$$

- $\mathcal{C} \subseteq \mathcal{X}$ is **robust control invariant** if:

$$x \in \mathcal{C} \implies \exists u \in \mathcal{U} : f(x, u) + w \in \mathcal{C} \quad \forall w \in \mathcal{W}$$

Equivalently: $\mathcal{C} \subseteq \text{Pre}_{\mathcal{W}}(\mathcal{C})$.

As in the nominal case, the maximal robust positive/control invariant sets can be computed by the analogous fixed-point iteration, now using $\text{Pre}_{\mathcal{W}}$ instead of Pre .

Summary: Set operations for Linear Systems

The set operations for the one-step reachable sets in the linear case can be summarized as follows, using the introduced set operations:

	$x_{k+1} = Ax_k$	$x_{k+1} = Ax_k + Bu_k$
$\text{Pre}(\mathcal{X})$	$\mathcal{X} \circ A$	$(\mathcal{X} \oplus (-B \circ \mathcal{U})) \circ A$
$\text{Suc}(\mathcal{X})$	$A \circ \mathcal{X}$	$(A \circ \mathcal{X}) \oplus (B \circ \mathcal{U})$

And analogous for the uncertain case (additive disturbance):

	$x_{k+1} = Ax_k + w_k$	$x_{k+1} = Ax_k + Bu_k + w_k$
$\text{Pre}_{\mathcal{W}}(\mathcal{X})$	$(\mathcal{X} \ominus \mathcal{W}) \circ A$	$((\mathcal{X} \ominus \mathcal{W}) \oplus (-B \circ \mathcal{U})) \circ A$
$\text{Suc}_{\mathcal{W}}(\mathcal{X})$	$(A \circ \mathcal{X}) \oplus \mathcal{W}$	$(A \circ \mathcal{X}) \oplus (B \circ \mathcal{U}) \oplus \mathcal{W}$

Summary: Stability and Feasibility of MPC

Recursive feasibility via a control-invariant terminal set

Consider MPC with terminal set $\mathcal{X}_f$. If $\mathcal{X}_f$ is control invariant, then the feasible set $\mathcal{X}_N = \mathcal{K}_N(\mathcal{X}_f)$ is positively invariant for the closed-loop: whenever the problem is feasible at x_k , it is feasible at x_{k+1} . Hence feasibility at the initial time implies feasibility for all future times.

Stability via terminal cost as a control Lyapunov function

Suppose the terminal cost V_f is a control Lyapunov function on $\mathcal{X}_f$. Hence, a local feedback $u = \kappa_f(x)$ is admissible and keeps the state in $\mathcal{X}_f$, s.t.

$$V_f(f(x, \kappa_f(x))) - V_f(x) \leq -\ell(x, \kappa_f(x)), \quad \forall x \in \mathcal{X}_f,$$

with pos. def. stage cost ℓ . Then, the optimal MPC value function V_N is a Lyapunov function for the closed-loop on $\mathcal{X}_N$ and the origin is asympt. stable in $\mathcal{X}_N$.

Summary: Stability and Feasibility of MPC

Take-aways

- ▶ $\mathcal{X}_N$ is a control invariant set
 $\implies$ MPC is recursively feasible
- ▶ V_f is a control Lyapunov function on control invariant $\mathcal{X}_N$
 $\implies$ MPC is stable
- ▶ The feasible region is the controllable set $\mathcal{X}_N = \mathcal{K}_N(\mathcal{X}_f)$

Remarks

- ▶ Standard choice is $\mathcal{X}_f = \mathcal{O}_\infty$ under the LQR feedback K
- ▶ Set $V_f(x) = x^\top P x$ with P from the DARE
($\mathcal{X}_f$ automatically pos. inv. with local Lyapunov function V_f)