Computing the explicit MPC solution using the Hasse diagram of the lifted feasible domain

Florin Stoican, Ștefan S. Mihai

POLITEHNICA University of Bucharest (UPB) - Automatic Control and Systems Engineering Department

25.05.2023 - RTE workshop (CentraleSupelec)

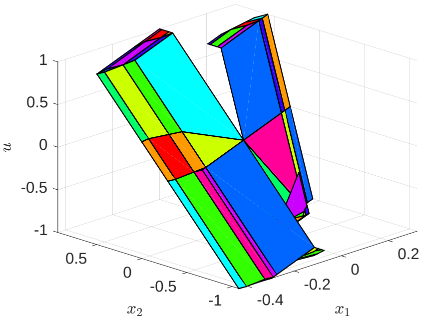
Outline

1. Motivation
2. Preliminaries
3. Geometrical interpretation of the explicit MPC representation
4. Explicit MPC toolbox

Outline

Motivation

- The Explicit MPC allows to solve the optimization problem off-line.
- The optimal control is an "explicit" function of the state $\rightarrow$ the on-line operations become simple function evaluations.
- Usually, the control law is a piecewise affine (PWA) function $\rightarrow$ the controller is stored in a lookup table of affine gains.
- Since both the control law and the cost surface are known (piecewise affine and, respectively, quadratic), stability and performance can be analyzed offline.



Idea

Exploit the geometrical structure of the problem to reduce the computation time of the eMPC!

Outline

Polyhedron – general notions

- Given $A \in \mathbb{R}^{d_H \times d}$, $b \in \mathbb{R}^{d_H}$ and $V \in \mathbb{R}^{d \times d_V}$, $R \in \mathbb{R}^{d \times d_R}$, a polyhedral set can be represented using the[1]:
  - **half-space representation**, i.e., the intersection of d_H linear inequalities

$$P(A, b) = \left\{ x \in \mathbb{R}^d : \ A_i x \leq b_i, \forall i \in \{1, \ldots, d_H\} \right\};$$

  - **generator representation**, i.e., the convex sum of d_V vertices added to the linear sum of d_R rays

$$P(V, R) = \Big\{ x \in \mathbb{R}^d : \ x = \sum_{j=1}^{d_V} \alpha_j V_j + \sum_{k}^{d_R} \beta_k R_k, \ \alpha_j, \beta_k \geq 0; \\ \sum_{j=1}^{d_V} \alpha_j = 1, \ \forall j \in \{1, \ldots, d_V\}, k \in \{1, \ldots, d_R\} \Big\}.$$

- A bounded polyhedron is called a **polytope**.

[1] Fukuda, K., "Polyhedral computation", în (2020). DOI: 10.3929/ethz-b-000426218.

.

Polytope – faces/ f-vector

- A face F may be represented in[2]:
  - **half-space form**, by a combination of active/inactive constraints, indexed by sets $\mathcal{A}$ – active, $\mathcal{I}$ – inactive, where $\mathcal{A} \cup \mathcal{I} = \{1, \dots, d_H\}$ and $\mathcal{A} \cap \mathcal{I} = \emptyset$, as

$$F(\mathcal{A}) = \left\{ A_i x = b_i, \forall i \in \mathcal{A},\ A_i x \leq b_i, \forall i \in \mathcal{I} \right\};$$

  - **generator form**, by a collection of vertices indexed by the set $\mathcal{V} \subset \{1 : d_V\}$, as

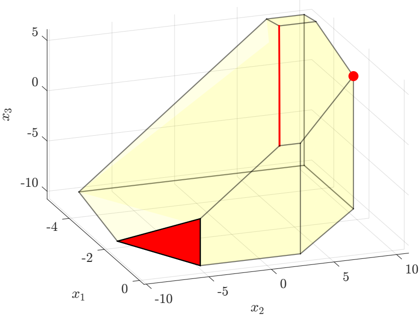
$$F(\mathcal{V}) = \left\{ x = \sum_{j \in \mathcal{V}} \alpha_j V_j,\ \sum_{j \in \mathcal{V}} \alpha_j = 1,\ \alpha_j \geq 0, \forall j \in \mathcal{V} \right\}.$$

- A face F is called a $k-$face if it is embedded in a $k-$subspace in $\mathbb{R}^d$.
- The f-vector of P is given as:

$f(P) = (f_{-1}, f_0, f_1, \dots, f_d)$

f_k is the number of $k-$dimensional faces.

the convention is that $f_{-1} = f_d = 1$



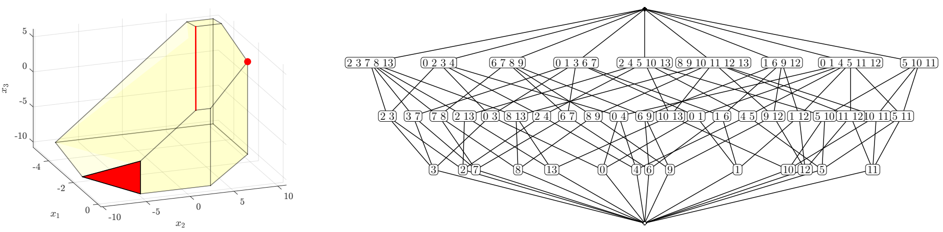
[2] Fukuda, K., "Polyhedral computation", în (2020). DOI: 10.3929/ethz-b-000426218.

.

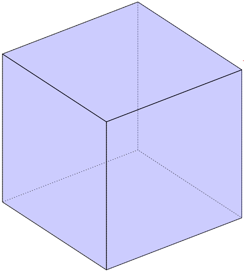
Polytope – Hasse diagram

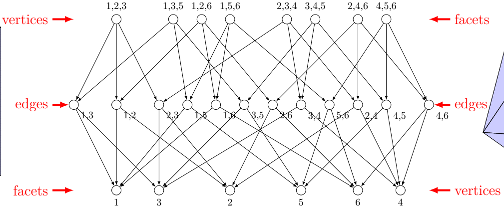
- The collection of all faces of a polytope P is called its face lattice.
- It enjoys a partial ordering relation $\rightarrow$ graph representation, the so-called Hasse diagram.

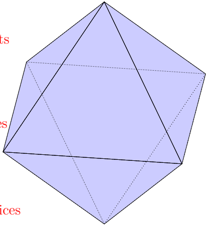
The Hasse diagram (right) of the polytope (left)



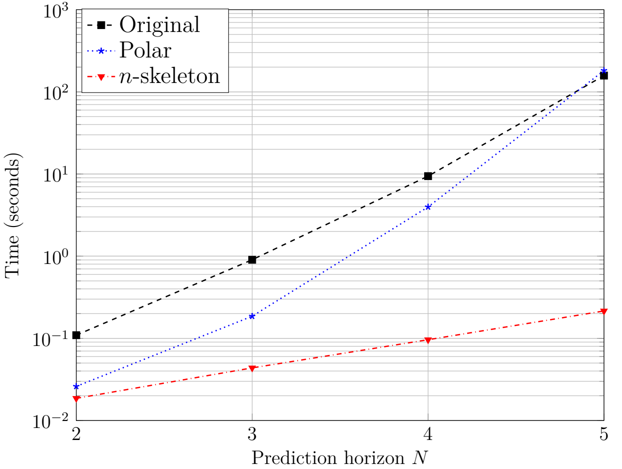
The cube and its polar, the cross-polytope







Comparison times for Hasse diagram computation



The MPC problem

- Consider the linear time-invariant (LTI) discrete system:

$$x_{k+1} = Ax_k + Bu_k, \quad y_k = Cx_k.$$

- Then the typical (quadratic cost and linear constraints) MPC problem is:

$$\begin{aligned}
\mathbf{u}_N^\star = \arg\min_{\mathbf{u}_N} x_N^\top S x_N + \sum_{k=0}^{N-1} \left(x_k^\top Q x_k + u_k^\top R u_k \right) \,, & \longleftarrow \text{quadratic cost} \\
\text{s.t.} \quad x_{k+1} = Ax_k + Bu_k \,, & \longleftarrow \text{state equation} \\
y_k = Cx_k \,, & \longleftarrow \text{output equation} \\
x_k \in \mathcal{X} \,, u_k \in \mathcal{U} \,, y_k \in \mathcal{Y} \,, & \longleftarrow \text{state, input, output constraints} \\
x_N \in \mathcal{X}_f. & \longleftarrow \text{terminal state constraint}
\end{aligned}$$

- The equivalent multi-parametric quadratic program (mp-QP):

$$\begin{aligned}
\mathbf{u}_N^\star(x_0) = \arg\min_{\mathbf{u}_N} \frac{1}{2} \mathbf{u}_N^\top \tilde{Q} \mathbf{u}_N + x_0{}^\top \tilde{H} \mathbf{u}_N \\
\text{s.t.} \ A\mathbf{u}_N \leq b + Ex_0,
\end{aligned}$$

The compact form

For further use, we introduce auxiliary notation:

$$
\mathbf{x}_N = \begin{bmatrix} x_1^\top & \dots & x_N^\top \end{bmatrix}^\top, \mathbf{y}_N = \begin{bmatrix} y_1^\top & \dots & y_N^\top \end{bmatrix}^\top,
$$

$$
\Theta_N = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \ \Phi_N = \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix},
$$

$$
\mathbf{X}_N = \mathrm{diag}\{\underbrace{X, \dots, X}_{N}\}, \forall X \in \{Q, R, C\}, \ {}_N = [\underbrace{0, \dots, 0}_{N-1}, I].
$$

This allows to rewrite the MPC problem compactly:

$$
\begin{aligned}
\mathbf{u}_N^\star = \arg\min_{\mathbf{u}_N} \mathbf{u}_N^\top & \left[\Phi_N^\top (\mathbf{Q}_N + {}_N^\top S \, {}_N)\Phi_N + \mathbf{R}_N\right] \mathbf{u}_N \\
& + 2{x_0}^\top \Theta_N^\top (\mathbf{Q}_N + {}_N^\top S \, {}_N)\Phi_N \mathbf{u}_N + {x_0}^\top \Theta_N^\top (\mathbf{Q}_N + {}_N^\top S \, {}_N)\Theta_N x_0, \\
\text{s.t.} \quad & \Theta_N x_0 + \Phi_N \mathbf{u}_N \in \mathcal{X}_N, \\
& \mathbf{u}_N \in \mathcal{U}_N, \\
\mathbf{C}_N(\Theta_N x_0 & + \Phi_N \mathbf{u}_N) \in \mathcal{Y}_N, \\
{}_N(\Theta_N x_0 & + \Phi_N \mathbf{u}_N) \in \mathcal{X}_f.
\end{aligned}
$$

Outline

The KKT conditions

The usual approach is to rewrite the initial problem in its dual form via the Karush-Kuhn-Tucker (KKT) optimality conditions

$$\begin{aligned} \tilde{Q}\mathbf{u}_N^\star + \tilde{H}^\top x_0 + A^\top \lambda^\star &= 0, \\ A\mathbf{u}_N^\star - Ex_0 &\leq b, \\ \lambda^\star &\geq 0, \\ \lambda^\star \times (A\mathbf{u}_N^\star - Ex_0 - b) &= 0. \end{aligned}$$

Taking a particular subset $\mathcal{A} \subset \{1, 2, \dots\}$ of inequalities to be active, we arrive at

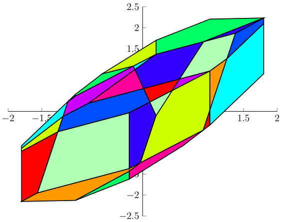
$$\begin{aligned} \tilde{Q}\mathbf{u}_N^\star + \tilde{H}^\top x_0 + A_{\mathcal{A}}^\top \lambda^\star &= 0, \\ A_{\mathcal{A}}\mathbf{u}_N^\star - E_{\mathcal{A}}x_0 &= b_{\mathcal{A}}, \quad A_{\mathcal{I}}\mathbf{u}_N^\star - E_{\mathcal{I}}x_0 \leq b_{\mathcal{I}} \\ \lambda_{\mathcal{A}}^\star &\geq 0, \quad \lambda_{\mathcal{I}}^\star = 0 \end{aligned}$$

where $\mathcal{I} = \{1, 2, \dots\} \setminus \mathcal{A}$.

The explicit MPC solution

The mp-QP may be exploited (using the KKT conditions) to explicitly and offline give the input in terms of the current value of the state (piecewise formulation[3]):

- for a given x_0, a subset of constraints, $\mathcal{A}$, is active;
- to it, corresponds a critical region: $CR_{\mathcal{A}} = \{Z_{\mathcal{A}} x_0 \leq z_{\mathcal{A}}\}$
- over which, the constrained optimum is defined: $\mathbf{u}_N^{\star}(x_0) = L_{\mathcal{A}} x_0 + l_{\mathcal{A}}$



To find all critical regions/control laws, we need to iterate all possible candidate sets of active constraints $\mathcal{A}$!

Issue

- the number of critical regions increases exponentially with prediction horizon.

[3] Bemporad, A., M. Morari, V. Dua și E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems", în *Automatica* 38.1 (2002), pp. 3–20, 2002.

.

Geometrical interpretation – I

Define

- The subspace of all the unconstrained solutions of the mp-QP:

$$\overline{\mathcal{U}} = \left\{ \begin{bmatrix} \mathbf{u}_N \\ x_0 \end{bmatrix} : \begin{bmatrix} -\tilde{Q}^{-1}\tilde{H}^\top \\ \mathbf{I} \end{bmatrix} x_0, \ \forall x_0 \in \mathbb{R}^n \right\};$$

- The lifted feasible domain:

$$P(\begin{bmatrix} A & -E \end{bmatrix}, b).$$

Main idea

- Inspired by the work of Seron[4][5], we state that a subset of faces of the lifted feasible domain correspond to the critical regions.
- The faces which do not correspond to the critical regions are hidden → we developed a visibility test.

[4] Seron, M. M., G. C. Goodwin și J. A. De Doná, "Characterisation of receding horizon control for constrained linear systems", în *Asian Journal of Control* 5.2 (2003), pp. 271–286, 2003.

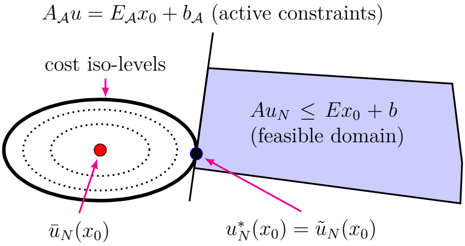
.

[5] Seron, M. M., J. A. De Dona și G. C. Goodwin, "Global analytical model predictive control with input constraints", în *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 1, 2000, pp. 154–159, 2000.

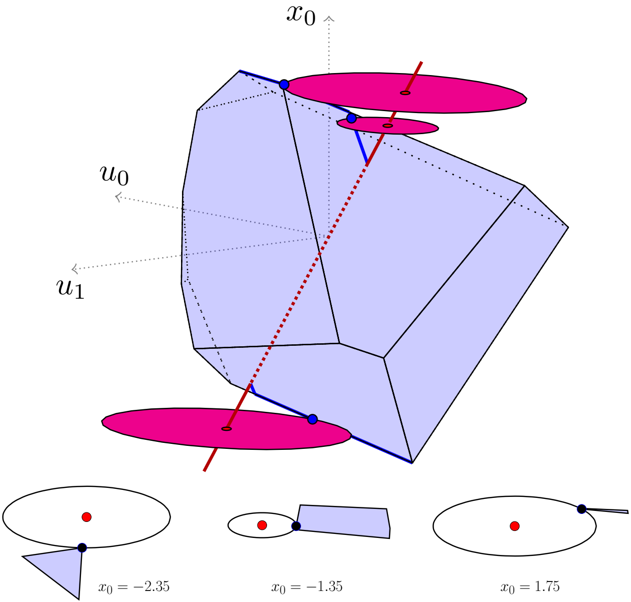
.

Geometrical interpretation – II

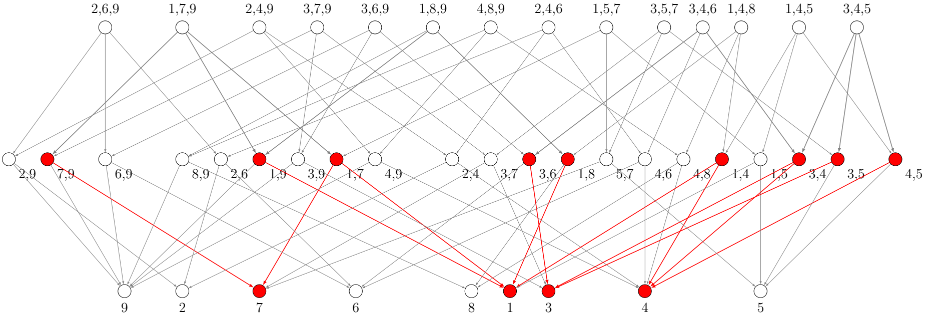
- At each value of x_0 a different instance of the feasible domain is active
- We can look at the problem in a lifted space (input plus initial state)
- We now have three elements:
  - the subspace of the unconstrained optimum
  - the lifted feasible domain
  - the piecewise constrained optimal solution



Geometrical interpretation of the problem – III



Associated Hasse diagram



Complexity bounds

- The candidate set $\mathcal{A}$ of active constraints is in fact an intersection of faces ($\mathcal{F}(\mathcal{A}) = \{h_i x = k_i, \forall i \in \mathcal{A}\} \cap \{h_i x \leq k_i, \forall i \notin \mathcal{A}\}$).

- The upper bound for the number of candidate sets / critical regions is given by the complexity of the feasible domain (its number of faces):

$$I(P) = \sum_{\mathcal{A} \subset \{1,\ldots,N_h\}, \mathcal{F}(\mathcal{A}) \neq \emptyset} 1 \quad = \quad \sum_{k=0}^{d-1} f_k(P),$$

  where $f_k(P)$ is the number of k-order faces of the polyhedron P.

- The bound for the k-order face for a polyhedron is [Fukuda 2020]:

$$f_k(P) \leq f_{d-k-1}(c(d, N_h)), \forall k = 0, 1, \ldots, d-2,$$

  where:

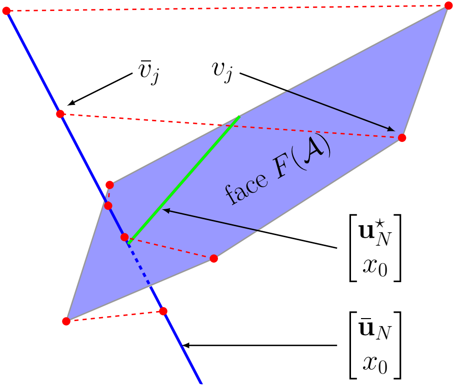
$$f_k(c(d, N_h)) = \sum_{r=0}^{\lfloor d/2 \rfloor} \binom{r}{d-k-1} \binom{N_h - d + r - 1}{r} + \sum_{r=\lfloor d/2 \rfloor + 1}^{d} \binom{r}{d-k-1} \binom{N_h - r - 1}{d-r},$$

  and $c(d, N_h)$ denotes the cyclic polytope.

The algorithm sketch

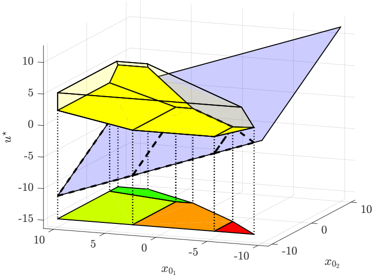
- compute the k-skeleton of the Hasse diagram for the lifted feasible domain (in its polar form!, using cdd and Kaibel's algorithm)
- as stop conditions in the Hasse construction, use geometric/algebraic (Truffet's algorithm) methods
- for all active sets of constraints construct the critical regions and associated affine control laws
- use the graph structure to efficiently store and retrieve the active region at runtime (the point location problem)

Face visibility idea



Visibility test

- Each face of $P(\begin{bmatrix} A & -E \end{bmatrix}, b)$ is either:
  - hidden by the other faces of the polytope, or
  - projects onto the sub-space $\overline{\mathcal{U}}$ into a region which corresponds bijectively with a critical region of the explicit solution.
- A face is visible if no segment $[v_j, \overline{v}_j]$ intersects the feasible domain $P(\begin{bmatrix} A & -E \end{bmatrix}, b)$; here, v_j is a vertex of P and $\overline{v}_j$ is its projection on $\overline{\mathcal{U}}$.



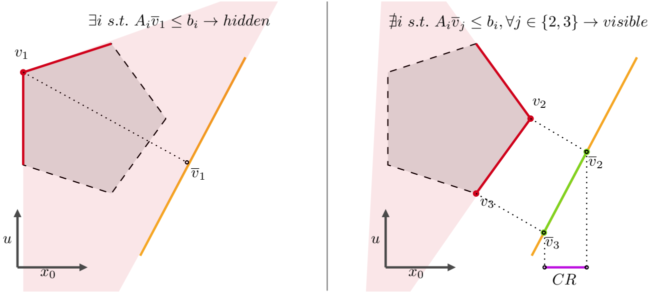
Face visibility test

Consider a face $F \subset P(\begin{bmatrix} A & -E \end{bmatrix}, b)$ given in both half-space – $F(\mathcal{A})$, and generator – $F(\mathcal{V})$, forms. Then, verifying:

$$\exists i \in \mathcal{A} \text{ s.t. } \begin{bmatrix} A & -E \end{bmatrix}_i \overline{v}_j \leq b_i, \ \forall j \in \mathcal{V},$$

is a sufficient condition for the visibility of the face F w.r.t. the subspace $\overline{\mathcal{U}}$.

Visibility test



Face visibility test

Consider a face $F \subset P(\begin{bmatrix} A & -E \end{bmatrix}, b)$ given in both half-space – $F(\mathcal{A})$, and generator – $F(\mathcal{V})$, forms. Then, verifying:

$$\exists i \in \mathcal{A} \text{ s.t. } \begin{bmatrix} A & -E \end{bmatrix}_i \overline{v}_j \leq b_i,\ \forall j \in \mathcal{V},$$

is a sufficient condition for the visibility of the face F w.r.t. the subspace $\overline{\mathcal{U}}$.

Algebraic test for critical region emptiness

Rationale

- The process of selecting the candidate polyhedral sets from the face lattice continues after the visibility test.
- Thus, the remaining candidate sets have to be non-empty.
- We used an algebraic method to tackle this problem[6].

Assumptions

- the matrix $Z_{\mathcal{A}}$ has no null row;
- that $q > n$ and that $Z_{\mathcal{A}} \in \mathbb{R}^{q \times n}$ has full column rank, i.e., $\mathrm{rank}(Z_{\mathcal{A}}) = n$.

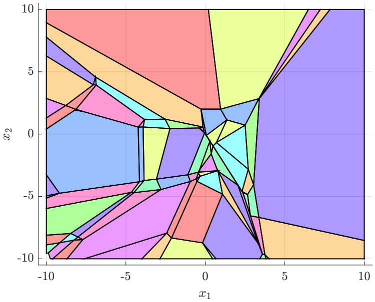
[6] Truffet, L., "Some Ideas to Test if a Polyhedron is Empty", în *arXiv preprint arXiv:2004.12818* (2020)

.

Illustrative example

Approach

- compute the face lattice and export the Hasse graph using Polymake – a powerful tool "designed for the algorithmic treatment of polytopes and polyhedra";
- apply successively the test procedures to retrieve the non-empty critical regions and their associated affine control laws.
- we validate our results by comparing with the Parametric Optimization toolbox (POP) – a MATLAB toolbox with efficient implementations of mp-QP problem solvers (problem 82 from POP dataset).



	Time (seconds)			
POP problem	POP toolbox	Current approach		
		Polymake	*MATLAB*	*Total*
82	13.27	3.06	2.87	5.93

Tools: Polymake[7], POP Toolbox[8].

[7] Gawrilow, E. și M. Joswig, "Polymake: a framework for analyzing convex polytopes", în *Polytopes—combinatorics and computation*, Springer, 2000, pp. 43–73, 2000.

.

[8] Oberdieck, R., N. A. Diangelakis, M. M. Papathanasiou, I. Nascu și E. N. Pistikopoulos, "Pop–parametric optimization toolbox", în *Industrial & Eng. Chemistry Research* 55.33 (2016), ACS Publications, pp. 8979–8991, 2016.

.

Outline

Mpqp-lattice toolbox

- We are working on a toolbox in C++ specialized in solving multi-parametric quadratic programs with linear constraints (stable release via QR)
- It uses `Eigen 3` and `libigl` for linear algebra and the `cddlib` [Fukuda 2003] library for polyhedral computations (vertex/ facet enumeration and nonemptyness check)
- Algebraic test (experimental) for nonemptyness check [Truffet 2020]
- Containerized application with Docker (can be compiled on different platforms)

A medium-sized example

- The state-space model

$$x^{+} = \begin{bmatrix} I_3 & 0.1I_3 \\ O_3 & I_3 \end{bmatrix} x + \begin{bmatrix} 0.005I_3 \\ 0.1I_3 \end{bmatrix} u$$

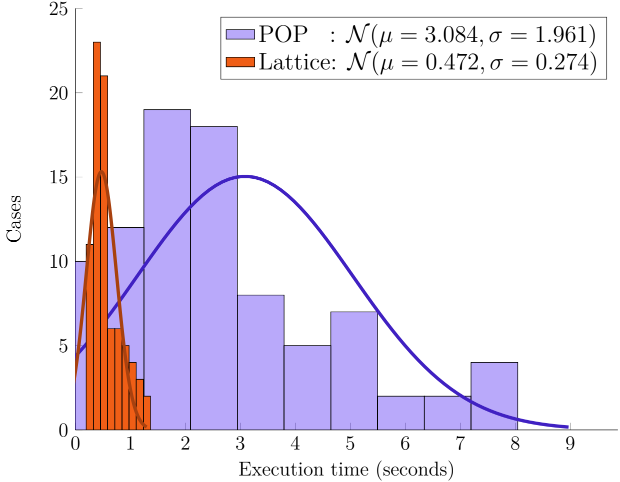
- Constraints: $\left| \begin{bmatrix} I_3 & O_3 \end{bmatrix} x \right| \leq [1.5\ 1.5\ 1.5]^\top$ and $|u| \leq [0.8154\ 0.8154\ 3.2700]^\top$
- Prediction horizon $N = 3$ and weights $Q = diag(50I_3, 5I_3)$, $R = 5I_3$
- We used the POP Toolbox [Oberdieck, Diangelakis, Papathanasiou, Nascu și Pistikopoulos 2016b] to generate the mpQP and to solve the problem for reference

Results

	Execution time (sec)	Number of nonempty regions
POP Toolbox	3611	11838
mpqp-lattice	1044	18016

- We generated more solutions than POP; all POP solutions can be found within ours

Comparison between POP/our toolbox (simple example)



References

[1] Bemporad, A., M. Morari, V. Dua și E. N. Pistikopoulos,"The explicit linear quadratic regulator for constrained systems",în *Automatica* 38.1 (2002), pp. 3–20, 2002.

[2] Fukuda, K.,"Cddlib reference manual",în *Report version 093a, McGill University, Montréal, Quebec, Canada* (2003).

[3] Fukuda, K.,"Polyhedral computation",în (2020).

[4] Gawrilow, E. și M. Joswig,"Polymake: a framework for analyzing convex polytopes",în *Polytopes—combinatorics and computation*,Springer, 2000,pp. 43–73, 2000.

[5] Oberdieck, R., N. A. Diangelakis, M. M. Papathanasiou, I. Nascu și E. N. Pistikopoulos,"Pop–parametric optimization toolbox",în *Industrial & Eng. Chemistry Research* 55.33 (2016), ACS Publications, pp. 8979–8991, 2016.

[6] Oberdieck, R., N. A. Diangelakis, M. M. Papathanasiou, I. Nascu și E. N. Pistikopoulos,"Pop–parametric optimization toolbox",în *Industrial & Engineering Chemistry Research* 55.33 (2016), ACS Publications, pp. 8979–8991, 2016.

[7] Seron, M. M., J. A. De Dona și G. C. Goodwin,"Global analytical model predictive control with input constraints",în *Proceedings of the 39th IEEE Conference on Decision and Control*,vol. 1,2000,pp. 154–159, 2000.

[8] Seron, M. M., G. C. Goodwin și J. A. De Doná,"Characterisation of receding horizon control for constrained linear systems",în *Asian Journal of Control* 5.2 (2003), pp. 271–286, 2003.

[9] Truffet, L.,"Some Ideas to Test if a Polyhedron is Empty",în *arXiv preprint arXiv:2004.12818* (2020).