09_Explicit MPC

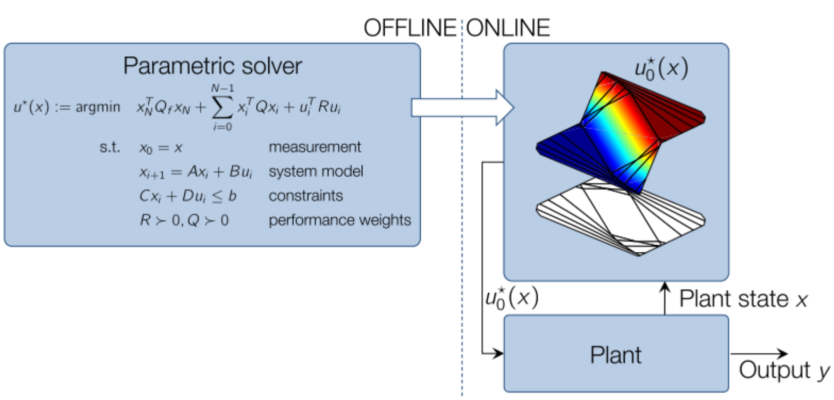
1. Model and Motivation of LQR

Motivation

Traditional MPC cannot guarantee **real-time**. And a classical MPC problem has following properties:

- Optimization problem is function parameterized by state
- Control law piecewise affine for linear systems/constraints
- Pre-compute control law as function of state x

So we can build a **pre-compute control law**.



2. MPC=Parametric Quadratic Programming

MPC to Parametric Quadratic Programming

A classical MPC can be translated from

$$
\begin{aligned}
J^\star(x) = \min x_N^T Q_f x_N + \sum_{i=0}^{N-1} x_i^T Q x_i + u_i^T R u_i \\
\text{s.t. } x_0 = x \qquad\qquad \\
x_{i+1} = A x_i + B u_i \\
C x_i + D u_i \leq b
\end{aligned}
$$

to

$$
\begin{aligned}
J^\star(x) = \min_u \frac{1}{2} u^T Q u + (Fx + f)^T u \\
\text{s.t. } Gu \leq Ex + e
\end{aligned}
$$

KKT Optimality Condition

For optimization problem:

$$
\begin{aligned}
\min_x f(x) \\
\text{s.t. } g(x) \leqslant 0 \\
h(x) = 0
\end{aligned}
$$

The optimum always meet:

Proposition (Karush-Kuhn-Tucker conditions)

$$
\begin{aligned}
\nabla f(x) + \nabla g(x)\mu + \nabla h(x)\lambda = 0 \\
\mu^T g(x) = 0 \\
\mu \geqslant 0 \\
h(x) = 0 \\
g(x) \leqslant 0
\end{aligned} \tag{1}
$$

KKT for ConvexQPs

For Convex QPs:

$$
\begin{aligned}
\min f(z) := \frac{1}{2} z^\top Q z \\
\text{s.t.} \quad Az \leq b
\end{aligned}
$$

We have

$$
\begin{aligned}
&Qz + A^\top \lambda = 0, \lambda \geq 0 && \text{Gradient is in the normal cone} \\
&Az \leq b && \text{Optimal point must be feasible} \\
&\lambda^\top (Az - b) = 0 && \text{Normal cone contains only active constraints}
\end{aligned}
$$

3. Parametric Linear Complementarity Problems

Parametric Linear Complementarity

Definition (Parametric Linear Complementarity Problem)

Given matrices M, q and Q, find functions $w(x), z(x)$ such that

$$
\begin{gathered}
w - Mz = q + Qx \\
w^\top z = 0 \\
w, z \geq 0
\end{gathered}
$$

we can always convert a pQP problem to pLCP problem by the following way:

$$
\begin{gathered}
J^\star(x) := \min_u \frac{1}{2} u^T Q u + (Fx + f)^T u \\
\text{s.t. } Gu \geq Ex + e \\
u \geq 0
\end{gathered}
$$

1. derives the KKT Conditions

$$
\begin{aligned}
&Qu + Fx + f - G^T \lambda - \nu = 0 && \text{Stationarity} \\
&-s + Gu = Ex + e, u \geq 0 \quad s \geqslant 0 && \text{Primary feasibility} \\
&\lambda, \nu \geq 0 && \text{Dual feasibility} \\
&\nu^T u = 0, \lambda^T s = 0 && \text{Complementarity}
\end{aligned}
$$

2. Then transform to Matrix Format

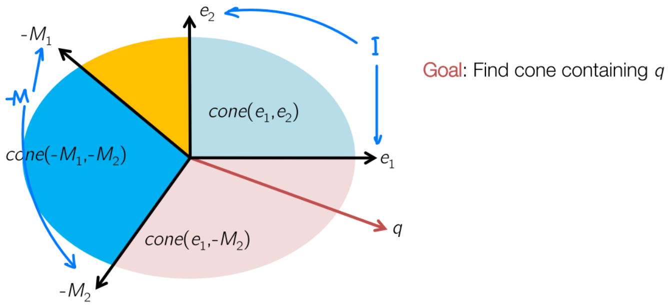
$$
\begin{gathered}
\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \nu \\ s \end{pmatrix} - \begin{bmatrix} Q & -G^T \\ G & 0 \end{bmatrix} \begin{pmatrix} u \\ \lambda \end{pmatrix} = \begin{bmatrix} F \\ -E \end{bmatrix} x + \begin{bmatrix} f \\ -e \end{bmatrix} \\
\nu, s, u, \lambda \geq 0 \\
\nu^T u = s^T \lambda = 0
\end{gathered}
$$

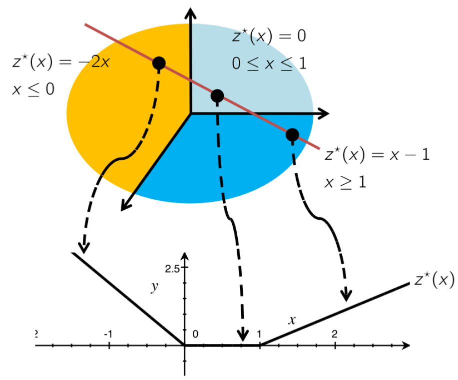
The Geometry Perspective

we have

- $w^\top z = 0, w, z \geq 0$: either w_i or z_i is zero for all i

- $Iw - Mz = q$: q is in the cone of non-zero variables





The Algebra Perpsective

Define the matrix $A := [l - M] \in \mathbb{R}^{n \times 2n}$ $\quad A \cdot [w, z] = Q_x + q$

Definition: Basis

The index set $B \subset \{1, \cdots, 2n\}$ is a **basis** if

- B contains n elements $|B| = n$
- Columns of A indexed by B are full-rank, rank $A_B = n$

Definition: Complementary Basis

B is a **complementary basis** if

$$i \in B \Leftrightarrow i + n \notin B \text{ for all } i \in \{1, \ldots, n\}$$

Notes:

Complementary bases define complementary cones

$$\mathcal{C}(B) := \left\{q \in \mathbb{R}^n \mid A_B^{-1} q \geq 0\right\}$$

Theorem

Basis B 'solves' the LCP (M, q) if and only if $q \in \mathcal{C}(B)$

Method: Simple Parametric LCP Solver

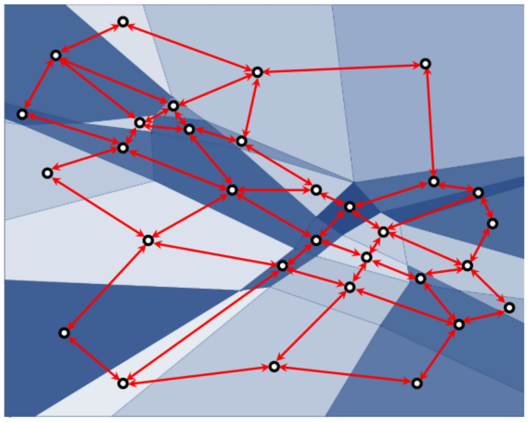
Simple Parametric LCP Solver

- For each complementary basis B
  - If $CR(B)$ is non-empty

$$\begin{bmatrix} w(x) \\ z(x) \end{bmatrix}_B = A_B^{-1}(q + Qx) \quad \text{for all } x \in CR(B)$$

Efficient Solution Methods

We can define a graph G, in which vertices are non-empty CRs and edges are adjacent CRs, then we can find all critical regions by standard graph enumeration



Basic Idea

Definition (Sufficient Matrix)

A matrix $M \in {}^{n\times n}$ is called **column sufficient** if it satisfies the implication

$$[z_i(Mz)_i \leq 0 \text{ for all } i] \implies [z_i(Mz)_i = 0 \text{ for all } i]\,.$$

The matrix M is called **row sufficient** if its transpose is column sufficient. If M is both column and row sufficient, then it is called **sufficient.**

Proposition

Positive semi-definite matrices are sufficient. (applies to non-symmetric PSD matrices too)

Proposition (LCPs with Sufficient Matrices)

- If $M$is a sufficient matrix, then the relative interiors of any two distinct complementary cones are disjoint (Cones cannot overlap)
- If M is a sufficient matrix, then the union of all complementary cones $K(M)$ is a convex polyhedral cone $K(M) = cone([I{-}M])$ (Neighbour graph is connected)

Computing Adjacent Regions

* for each row $(A_B^{-1}(q+Qx))_i \geq 0$ which forms a facet of CR(B)

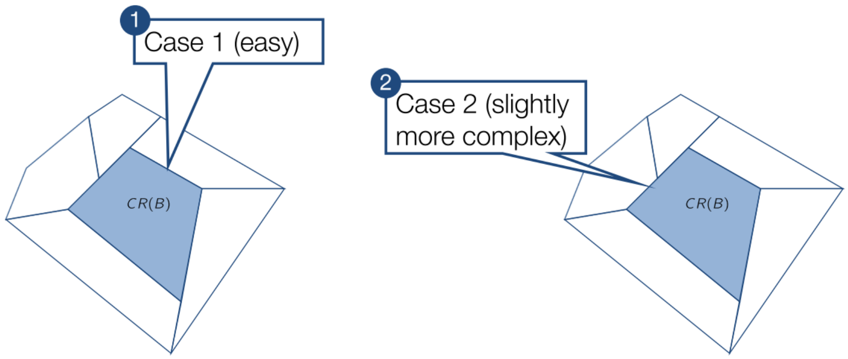
1 Case 1 : Diagonal pivot

return $B \backslash \{i\} \cup \{\bar{i}\}$

2 Case 2 : Exchange pivot

for each j if $CR(B \backslash \{i,j\} \cup \{\bar{i},\bar{j}\}) \cap CR(B) \neq \emptyset$

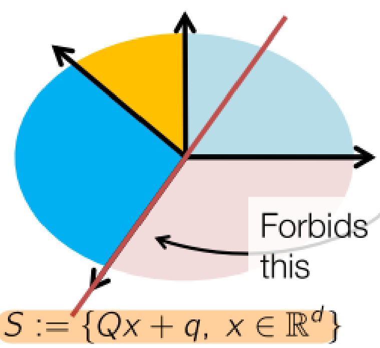
return $B \backslash \{i,j\} \cup \{\bar{i},\bar{j}\}$

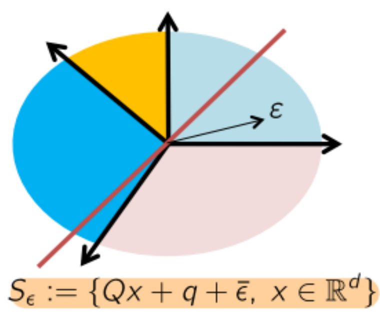


Degeneracy

All previous statements rely on S being in **general position:** S intersects CR(B) $\Rightarrow$ S intersects int(C(B))

Remain To Check





- Multiple solutions for same parameter
- We can simulate general position artificially

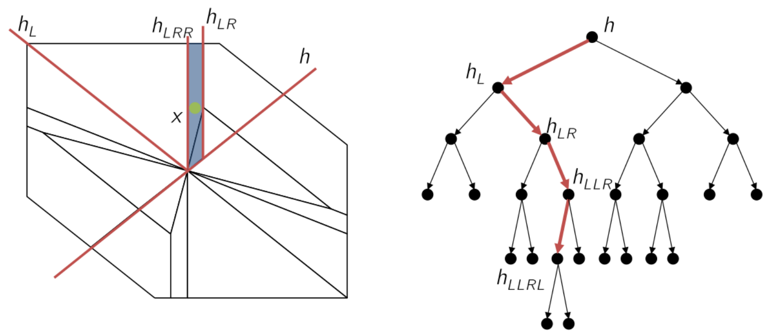
4. Point Location Problem

4.1. Sequential Search

Method: Sequential Search

Test each region one by one whether the given point meets its constraints

4.2. Bisection Search



<u>Method: Bisection Sea</u>

1. Find hyperplane that separates regions into two equal sized sets
2. Repeat for left and right sets

Summary

- implement MPC in **real-time scenario**s: by using **Explicit MPC**. That is we calculated all input-output pair **offline and store the mapping** into the controller.
- In order to do that,
  - need to transfer normal MPC problem to **parametric Quadratic Programming Problem,**
  - then translated to **Parametric Linear Complementarity** based on KKT condition.
  - Then we can use **geometry or algebra method** to solve it.
- When facing complex problem, we need to **optimize the framework** in order to improve the performance. We then use **some advanced graph methods** to present and store the mapping.