MECH 6v29.002 – Model Predictive Control

L9 – MPC Feasibility (continued)

Outline

- Feedback on HW #1
- Feasible Sets for constrained linear MPC
  - Batch approach
  - Projections
  - Example
  - Recursive approach
- Summary

Feedback on HW #1

- Grades are posted
  - Provided feedback only were needed
  - General comments
    - A few minor errors with indexing or number of iterations
    - It is good practice to write code that 'scales' well.
      - For example, think "does this approach work if $N = 5$ and if $N = 100$?"
    - Start assignments early to make sure that you have enough time to ask questions and for me to respond
- Code for an example implementation of solutions is provided on eLearning
  - Can use to debug your code or learn an alternative approach that might work better on future assignments

Constrained Linear Quadratic MPC

- For MPC with input and state/output constraints, it is important to analyze the feasibility of the optimization problem
  - Specifically, what is the set of initial states for which the constrained MPC problem is feasible?

$$J_0^*(x_0) = \min_{U_0} \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + x_N^T P x_N$$

$$s.t.$$

$$x_{k+1} = A x_k + B u_k, \quad k \in \{0, 1, \ldots, N-1\}$$

$$x_k \in \mathcal{X}, \quad u_k \in \mathcal{U}, \quad k \in \{0, 1, \ldots, N-1\}$$

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

$$x_0 = x(0)$$

Feasible Sets

- Feasible Set $\mathcal{X}_0$
  - Set of initial states $x(0)$ for which the optimal control problem is feasible
  - Equivalent to the N-step Controllable Set

$$\mathcal{X}_0 = \mathcal{K}_N\left(\mathcal{X}_f\right)$$

$$J_0^*\left(x_0\right) = \min_{U_0} \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + x_N^T P x_N$$

$$\begin{aligned} & s.t. \\ & x_{k+1} = Ax_k + Bu_k,\ k \in \{0,1,\ldots,N-1\} \\ & x_k \in \mathcal{X},\quad u_k \in \mathcal{U},\quad k \in \{0,1,\ldots,N-1\} \\ & x_N \in \mathcal{X}_f \\ & x_0 = x(0) \end{aligned}$$

  - Specifically:

$$\mathcal{X}_0 = \left\{ \begin{array}{r} x(0) \in \mathbb{R}^n \mid \exists U_0\ s.t.\ x_k \in \mathcal{X}, u_k \in \mathcal{U},\ \forall k = 0,\ldots,N-1 \\ x_N \in \mathcal{X}_f,\ x_{k+1} = Ax_k + Bu_k, \forall k = 0,\ldots,N-1 \end{array} \right\}$$

Batch Approach

- Inputs, state, and terminal constraint sets in H-Rep

$$\mathcal{U} = \left\{ u \in \mathbb{R}^m \mid A_u u \le b_u \right\}$$
$$\mathcal{X} = \left\{ x \in \mathbb{R}^n \mid A_x x \le b_x \right\}$$
$$\mathcal{X}_f = \left\{ x \in \mathbb{R}^n \mid A_f x \le b_f \right\} \subseteq \mathcal{X}$$

- Batch approach state trajectory

$$X = S_x x_0 + S_u U_0$$

$$X = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \quad U_0 = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \quad S_x = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^N \end{bmatrix} \quad S_u = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}$$

Batch Approach (cont.)

- Need to impose state constraints, terminal constraint, and input constraints
- Let's collect all of these values

$$X = S_x x_0 + S_u U_0$$

$$X = \begin{bmatrix} S_x & S_u \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix}$$

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} X \\ U_0 \end{bmatrix} = \begin{bmatrix} S_x & S_u \\ 0 & I \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix}$$

$$\begin{bmatrix} X \\ U_0 \end{bmatrix} \in \mathcal{X} \times \ldots \times \mathcal{X} \times \mathcal{X}_f \times \mathcal{U} \times \ldots \times \mathcal{U}$$

Cartesian Product

$$\mathcal{X} \times \mathcal{U} = \left\{ \begin{bmatrix} x \\ u \end{bmatrix} \in \mathbb{R}^{n+m} \mid x \in \mathcal{X}, u \in \mathcal{U} \right\}$$

Batch Approach (cont.)

- Collect constraints

$$\begin{bmatrix} X \\ U_0 \end{bmatrix} = \begin{bmatrix} S_x & S_u \\ 0 & I \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix}$$

$$\mathcal{U} = \{ u \in \mathbb{R}^m \mid A_u u \le b_u \}$$
$$\mathcal{X} = \{ x \in \mathbb{R}^n \mid A_x x \le b_x \}$$
$$\mathcal{X}_f = \{ x \in \mathbb{R}^n \mid A_f x \le b_f \} \subseteq \mathcal{X}$$

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} X \\ U_0 \end{bmatrix} \in \mathcal{X} \times \ldots \times \mathcal{X} \times \mathcal{X}_f \times \mathcal{U} \times \ldots \times \mathcal{U}$$

$$\Downarrow$$

$$blkdiag\left(A_x, \ldots, A_x, A_f, A_u, \ldots, A_u \right) \begin{bmatrix} S_x & S_u \\ 0 & I \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \le \begin{bmatrix} b_x \\ \vdots \\ b_x \\ b_f \\ b_u \\ \vdots \\ b_u \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} A_0 & A_{U_0} \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \le \begin{bmatrix} b_0 \\ b_{U_0} \end{bmatrix}$$

Batch Approach - Projection

- Now we have a set P that defines the vector of initial states and input trajectories that satisfy all constraints

$$\mathcal{P} = \left\{ \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \in \mathbb{R}^{n+mN} \mid \begin{bmatrix} A_0 & A_{U_0} \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \leq \begin{bmatrix} b_0 \\ b_{U_0} \end{bmatrix} \right\}$$

- To compute the feasible set $\mathcal{X}_0$, we need to project P onto the first n dimensions

$$\mathcal{P} = \left\{ \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \in \mathbb{R}^{n+mN} \mid \begin{bmatrix} A_0 & A_{U_0} \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \leq \begin{bmatrix} b_0 \\ b_{U_0} \end{bmatrix} \right\}$$

Project onto state dimensions

$$\mathcal{P}_x = \left\{ x_0 \in \mathbb{R}^n \mid \bar{A}_0 x_0 \leq \bar{b}_0 \right\}$$

Projections

- Given a polytope

$$\mathcal{P} = \left\{ \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m} \mid \begin{bmatrix} A_x & A_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} b_x \\ b_y \end{bmatrix} \right\}$$

$$\mathcal{P} = \left\{ \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m} \mid A_x x + A_y y \le \begin{bmatrix} b_x \\ b_y \end{bmatrix} \right\}$$

the projection on to the first n dimensions is

$$\text{proj}_{1:n}\left(\mathcal{P}\right) = \left\{ x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^m \;\; s.t.\; A_x x + A_y y \le \begin{bmatrix} b_x \\ b_y \end{bmatrix} \right\}$$

- Using MPT
  - p = projection(P, dims, method)
  - type edit projection to view description of input arguments in Matlab

Projections (cont.)

- Detailed discussion and novel algorithm provided in [1]
- Consider a d-dimensional polytope with q halfspaces to be projected down by k dimensions
- Projection methods:
  - Fourier-Motzkin elimination
    - Analogue of Gaussian elimination (but for inequalities instead of equalities)
    - Iterative approach by recursively projecting by removing one dimension
    - Generates many redundant constraints, may not be practical to remove all at each time step (requires solving a LP)
    - Time complexity is $\mathcal{O}\left(q^{2^k}\right)$
    - Cube in d-dimensions has q = 2d halfspaces. Projecting onto half the dimensions

$$d = 2, 4, 6, 8, \ldots$$

$$q^{2^k} = 16, 4096, 4e^8, 2e^{19}$$

[1] C. N. Jones, E. C. Kerrigan, and J. M. Maciejowski. "Equality Set Projection: A new algorithm for the projection of polytopes in halfspace representation," Technical Report, Department of Engineering, Cambridge University, 2004.

Projections (cont.)

- Detailed discussion and novel algorithm provided in [1]
- Consider a d-dimensional polytope with q halfspaces to be projected down by k dimensions
- Projection methods:
  - Vertex Projection
    - If the set is already in V-Rep, projection is relatively easy
      Just project each of the points
    - Iterative Quickhull algorithm to determine points that are needed for convex hull
    - Only practical with a relatively small number of vertices
    - Conversion from H-rep to V-rep is worst-case exponential

$$\#\text{vertices} = \mathcal{O}\left(q^{\left\lfloor \frac{d}{2} \right\rfloor} \right)$$

[1] C. N. Jones, E. C. Kerrigan, and J. M. Maciejowski. “Equality Set Projection: A new algorithm for the projection of polytopes in halfspace representation,” Technical Report, Department of Engineering, Cambridge University, 2004.

Example

- Consider the unstable 2nd order system

$$x_{k+1} = Ax_k + Bu_k = \begin{bmatrix} 1.5 & 0 \\ 1 & -1.5 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k$$

- Subject to input, state, and terminal constraints

$$u_k \in U = \{u \in \mathbb{R} \mid -5 \leq u \leq 5\}$$

$$x_k \in \mathcal{X} = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix} \right\}$$

$$x_N \in \mathcal{X}_f = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -1 \\ -1 \end{bmatrix} \leq x \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$$

- Compute the *N*-step Controllable Set
  (aka Feasible Set, Region of Attraction) for $N = 4$

Example - Batch Approach

- Algorithm
  - Define system matrices and number of steps

$$A=\begin{bmatrix}1.5 & 0\\ 1 & -1.5\end{bmatrix}, B=\begin{bmatrix}1\\ 0\end{bmatrix},\ N=4$$

$n=2$ states

$m=1$ input

  - Determine H-Rep of constraints

$$u_k \in U = \{u \in \mathbb{R} \mid -5 \le u \le 5\}$$

$$x_k \in \mathcal{X} = \left\{x \in \mathbb{R}^2 \mid \begin{bmatrix}-10\\ -10\end{bmatrix} \le x \le \begin{bmatrix}10\\ 10\end{bmatrix}\right\}$$

$$x_N \in \mathcal{X}_f = \left\{x \in \mathbb{R}^2 \mid \begin{bmatrix}-1\\ -1\end{bmatrix} \le x \le \begin{bmatrix}1\\ 1\end{bmatrix}\right\}$$

$\Rightarrow$

$A_u u \le b_u$

$$u_k \in U = \left\{u \in \mathbb{R} \mid \begin{bmatrix}1\\ -1\end{bmatrix} u \le \begin{bmatrix}5\\ 5\end{bmatrix}\right\}$$

$$x_k \in \mathcal{X} = \left\{x \in \mathbb{R}^2 \mid \begin{bmatrix}I_n\\ -I_n\end{bmatrix} x \le \begin{bmatrix}10\\ 10\\ 10\\ 10\end{bmatrix}\right\}$$

$A_x x \le b_x$

$$x_K \in \mathcal{X}_f = \left\{x \in \mathbb{R}^2 \mid \begin{bmatrix}I_n\\ -I_n\end{bmatrix} x \le \begin{bmatrix}1\\ 1\\ 1\\ 1\end{bmatrix}\right\}$$

$A_f x \le b_f$

Example - Batch Approach (cont.)

- Algorithm
  - Define system matrices and number of steps
  - Determine H-Rep of constraints
  - Compute lifted matrices S_x, S_u such that $X = S_x x_0 + S_u U_0$
  - Compute inequality constraints that define set P (see slide 7)

$$\mathcal{P} = \left\{ \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \in \mathbb{R}^{n+mN} \mid \begin{bmatrix} A_0 & A_{U_0} \end{bmatrix} \begin{bmatrix} x_0 \\ U_0 \end{bmatrix} \le \begin{bmatrix} b_0 \\ b_{U_0} \end{bmatrix} \right\}$$

    - MPT: P = Polyhedron('H',[H f]); $\leftarrow \mathcal{P} = \{ x \mid Hx \le f \}$

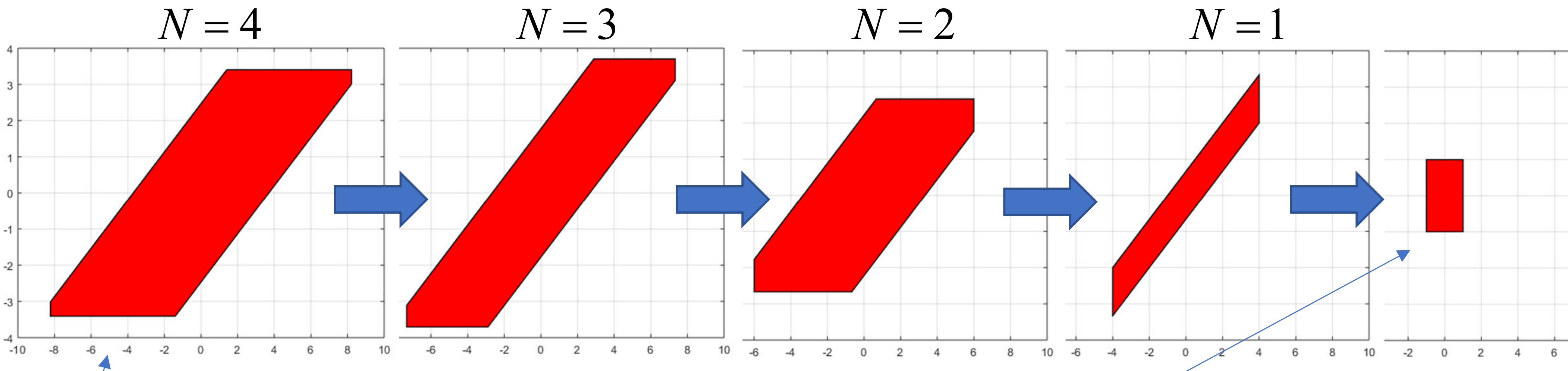
  - Project set P into first n dimensions

$$\mathcal{X}_0 = \text{proj}_{1:n}(\mathcal{P}) = \left\{ x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^m \;\; s.t. \; A_x x + A_y y \le \begin{bmatrix} b_x \\ b_y \end{bmatrix} \right\}$$

    - MPT: X0 = projection(P,1:n);

  - Plot: figure;plot(X0)

Example - Batch Approach (cont.)



$$\mathcal{X}_0 = \left\{ x \in \mathbb{R}^n \,\middle|\, \begin{bmatrix} 0 & -3.8387 \\ -2.2941 & 0 \\ 0.1477 & -0.2216 \\ -1.5681 & 2.3521 \\ 0.0625 & 0 \\ 0 & 0.1526 \end{bmatrix} x \le \begin{bmatrix} 13.0798 \\ 18.8625 \\ 0.5471 \\ 5.8076 \\ 0.5136 \\ 0.5199 \end{bmatrix} \right\}$$

$$\mathcal{X}_f = \left\{ x \in \mathbb{R}^2 \,\middle|\, \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} x \le \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Recursive Approach

- As an alternative to the batch approach, we can use an iterative approach to compute the same feasible set $\mathcal{X}_0$
- Same as computing the N-step Controllable Set $\mathcal{X}_0 = \mathcal{K}_N\left(\mathcal{X}_f\right)$ using the following recursion

$$\mathcal{K}_0\left(\mathcal{X}_f\right) = \mathcal{X}_f$$
$$\mathcal{K}_j\left(\mathcal{X}_f\right) = \mathrm{Pre}\left(\mathcal{K}_{j-1}\left(\mathcal{X}_f\right)\right) \cap \mathcal{X}, \quad j \in \{1,\dots,N\}$$

- For systems with inputs, computing the precursor set requires the projection operation

$$\mathcal{S} = \left\{x \in \mathbb{R}^n \mid A_s x \le b_s\right\}$$

$$\mathrm{Pre}\left(\mathcal{S}\right) = \left\{x_k \in \mathbb{R}^n \mid \exists u_k \in \mathcal{U} \;\; s.t.\; x_{k+1} = Ax_k + Bu_k \in \mathcal{S}\right\}$$

$$\mathrm{Pre}\left(\mathcal{S}\right) = \left\{x_k \in \mathbb{R}^n \mid \begin{bmatrix} A_s A & A_s B \\ 0 & A_u \end{bmatrix}\begin{bmatrix} x_k \\ u_k \end{bmatrix} \le \begin{bmatrix} b_s \\ b_u \end{bmatrix}\right\}$$

- Same as our definition of projection

$$\mathrm{proj}_{1:n}\left(\mathcal{P}\right) = \left\{x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^m \;\; s.t.\; A_x x + A_y y \le \begin{bmatrix} b_x \\ b_y \end{bmatrix}\right\}$$

Recursive Approach

- Algorithm
  - Define system matrices and number of steps
  - Determine H-Rep of constraints
  - Initialize

$$\mathcal{K}_0\left(\mathcal{X}_f\right)=\mathcal{X}_f$$

$$\mathcal{K}_0\left(\mathcal{X}_f\right)=\left\{x\in\mathbb{R}^2 \mid A_{K(0)}x\le b_{K0}\right\}$$

$$\mathcal{X}_f=\left\{x\in\mathbb{R}^2 \mid \underbrace{\begin{bmatrix} I_n \\ -I_n \end{bmatrix} x \le \begin{bmatrix}1\\1\\1\\1\end{bmatrix}}_{A_f x\le b_f}\right\}$$

  - Iterate for N steps
    - Precursor set

$$\mathrm{Pre}\left(\mathcal{K}_{j-1}\left(\mathcal{X}_f\right)\right)=\left\{x\in\mathbb{R}^2 \mid \begin{bmatrix} A_{K(j-1)}A & A_{K(j-1)}B \\ 0 & A_u \end{bmatrix}\begin{bmatrix} x_k \\ u_k \end{bmatrix}\le\begin{bmatrix} b_{k(j-1)} \\ b_u \end{bmatrix}\right\}$$

    - Define Polyhedron in MPT
    - Project onto first n states $\mathrm{proj}_{1:n}\left(\mathrm{Pre}\left(\mathcal{K}_{j-1}\left(\mathcal{X}_f\right)\right)\right)$
    - Intersection with state constraint set
      - Concatenation of inequality constraints $\mathcal{X}=\left\{x\in\mathbb{R}^2 \mid A_x x\le b_x\right\}$
  - Should get same set as batch approach but with computational advantages

Summary

- We can now compute the set of initial conditions for which our MPC problem is feasible

$$J_0^*(x_0) = \min_{U_0} \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + x_N^T P x_N$$

$$s.t.$$

$$x_{k+1} = Ax_k + Bu_k,\ k \in \{0,1,\ldots,N-1\}$$

$$x_k \in \mathcal{X},\quad u_k \in \mathcal{U},\quad k \in \{0,1,\ldots,N-1\}$$

$$x_N \in \mathcal{X}_f = \{0\}$$

$$x_0 = x(0)$$

$$x(0) \in \mathcal{X}_0 = \mathcal{K}_N(\mathcal{X}_f) \xrightarrow{\text{feasible}}$$

- Both the batch and the recursive approach require projection operations but the recursive approach requires the projection of lower-dimensional sets => more computationally efficient

- Next week, we will discuss invariant sets and show how to expand this set of feasible initial conditions

$$J_0^*(x_0) = \min_{U_0} \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + x_N^T P x_N$$

$$s.t.$$

$$x_{k+1} = Ax_k + Bu_k,\ k \in \{0,1,\ldots,N-1\}$$

$$x_k \in \mathcal{X},\quad u_k \in \mathcal{U},\quad k \in \{0,1,\ldots,N-1\}$$

$$x_N \in \mathcal{X}_f = \Omega \longleftarrow \text{Invariant set}$$

$$x_0 = x(0)$$