Model Predictive Control using MATLAB
1: Introduction and Preliminaries

by
Midhun T. Augustine

Overview

Introduction

Introduction

- MPC is a modern control approach which uses model based optimization for computing the control input.
- In MPC a model of the system is used to predict the future behaviour (states) of the system for a control input sequence.
- Also known as Receding Horizon Control (RHC).
- MPC: advantages over optimal control
  1. It gives closed-loop control schemes whereas optimal control mostly results in open-loop control schemes.
  2. MPC can handle complex systems such as nonlinear, higher-order, multi-variable, etc.
  3. MPC can incorporate constraints easily.

MPC: Block diagram

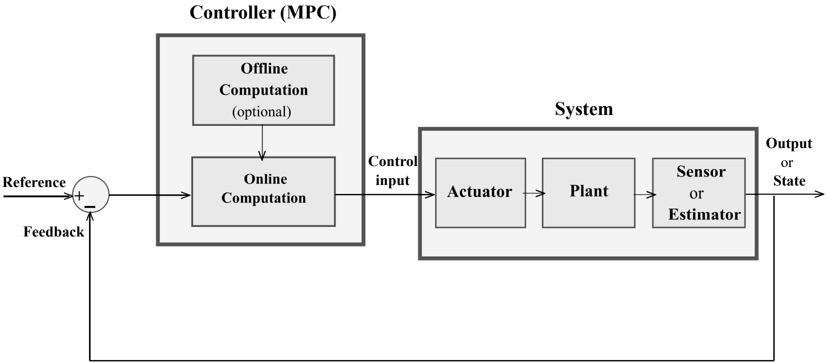


Figure 1: MPC: Block diagram

Notations

- $\mathbb{N}, \mathbb{Z}, \mathbb{R}$: Set of natural numbers, integers and real numbers.
- $\mathbb{R}^n$: $n-$ dimensional Euclidean space.
- $\mathbb{R}^{m \times n}$: Space of $m \times n$ real matrices.
- $\mathbf{A}, \mathbf{a}$: Matrix/vector $\mathbf{A}, \mathbf{a}$.
- A, a : Scalar A, a.
- $\mathbb{A}$: Set $\mathbb{A}$.
- $\mathbf{P} > 0$: Real symmetric positive definite matrix $\mathbf{P}$
- $\mathbf{P} \geq 0$: Real symmetric positive semidefinite matrix $\mathbf{P}$
- $\mathbf{I}, \mathbf{0}$: Identity matrix and zero matrix.

MPC: Terminologies

1. **Sampling time** (T): It is the time difference between two consecutive state measurements or control updates. In general $T \in \mathbb{R}^+$.
2. **Time horizon** (N_T): It is the number of time instants the control input is applied to the system. In general $N_T \in \mathbb{N}$.
3. **Prediction horizon** (N): It is the length of the prediction window over which the states are predicted and optimized. In general $N \in \mathbb{N}$ and usually $2 \leq N \leq N_T$.
4. **Control horizon** (N_C): It is the length of the control window in which the control input is optimized, and normally $N_C \leq N$.

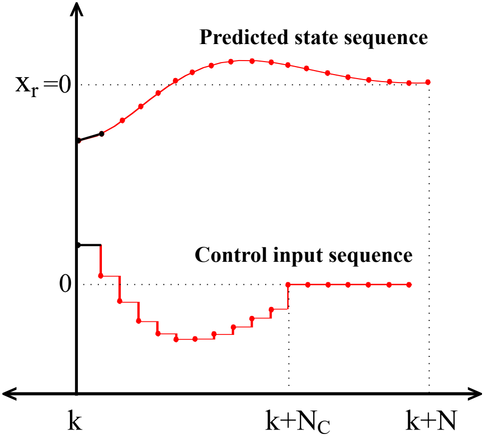


Figure 2: MPC: Basic strategy

Preliminaries

General optimization problem

- Optimization problem

$$\begin{aligned} \underset{\mathbf{z}}{inf} \quad f(\mathbf{z}) \quad & subject\ to \\ \mathbf{z} \in \mathbb{Z} & \subseteq \mathbb{R}^p \end{aligned} \tag{1}$$

- $\mathbf{z}$: Decision vector $\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{bmatrix}$
- f : Cost function. Some of the basic cost functions are
  1. $f(\mathbf{z}) = \mathbf{c}^T\mathbf{z}$: Linear cost function
  2. $f(\mathbf{z}) = \mathbf{z}^T\mathbf{H}\mathbf{z} + \mathbf{q}^T\mathbf{z} + r$: Quadratic cost function
- $\mathbb{Z}$: Constraint set.

General optimization problem

- Constrained set is usually defined using linear inequalities

$$\mathbb{Z} = \left\{ \mathbf{z} \in \mathbb{R}^2 : \begin{matrix} z_1 \geq 0 \\ z_2 \geq 0 \\ z_1 + 2z_2 \leq 10 \\ 2z_1 + z_2 \leq 10 \end{matrix} \right\} \text{ which is equivalent to}$$

$$\mathbb{Z} = \{\mathbf{z} \in \mathbb{R}^2 : \mathbf{Fz} \leq \mathbf{g}\}, \qquad \mathbf{F} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 2 \\ 2 & 1 \end{bmatrix}, \mathbf{g} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

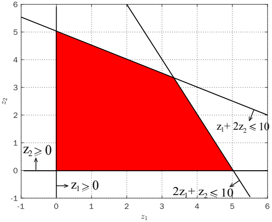


Figure 3: Constraint set (Polyhedron)

Convex optimization problem

- **Convex set**: A set $\mathbb{Z} \in \mathbb{R}^p$ is convex if

$$\lambda \mathbf{z}_1 + (1-\lambda)\mathbf{z}_2 \in \mathbb{Z} \tag{2}$$

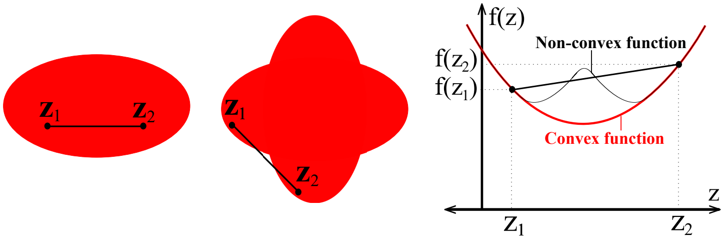
  for all $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{Z}$ and $\lambda \in [0,1]$.

- **Convex function**: A function $f : \mathbb{Z} \to \mathbb{R}$ is convex if $\mathbb{Z}$ is convex and

$$f\big(\lambda \mathbf{z}_1 + (1-\lambda)\mathbf{z}_2\big) \leq \lambda f(\mathbf{z}_1) + (1-\lambda) f(\mathbf{z}_2) \tag{3}$$

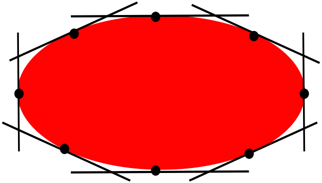
  for all $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{Z}$ and $\lambda \in [0,1]$.

- **Convex optimization problem**: in which the cost function f is a convex function and the constraint set $\mathbb{Z}$ is a convex set.

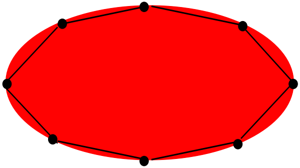


(a) Convex set (b) Non-convex set (c) Convex function

Convex optimization problem



(a) Over approximation



(b) Under approximation

Figure 5: Approximating a convex set with a polytope

Convex optimization problem: Examples

1. **Linear programming problem**: is of the form

$$\begin{aligned} \underset{\mathbf{z}}{inf} \quad & \mathbf{c}^T\mathbf{z} \quad subject\ to \\ & \mathbf{F}\mathbf{z} \leq \mathbf{g} \\ & \mathbf{F}_{eq}\mathbf{z} = \mathbf{g}_{eq} \end{aligned} \tag{4}$$

2. **Quadratic programming problem**: is of the form

$$\begin{aligned} \underset{\mathbf{z}}{inf} \quad & \mathbf{z}^T\mathbf{H}\mathbf{z} + \mathbf{q}^T\mathbf{z} + r \quad subject\ to \\ & \mathbf{F}\mathbf{z} \leq \mathbf{g} \\ & \mathbf{F}_{eq}\mathbf{z} = \mathbf{g}_{eq} \end{aligned} \tag{5}$$

Numerical optimization

- The two major approaches for numerical optimization
  1. **Iterative approach**: In which the elements of the decision vector are optimized together. Here the optimal decision vector is computed iteratively by starting with an initial guess which is then improved in each iterations.
  2. **Recursive approach**: In which the elements of the decision vector are optimized recursively, i.e., one at a time. The popular optimization algorithm which uses the recursive approach is the dynamic programming.

MPC: Classifications

MPC: Classifications

- Based on the type of system model used in optimization
  1. **Linear MPC**: The system model and the constraints are linear. The cost function can be linear or quadratic which results in linear or quadratic programming problems which are convex optimization problems.
  2. **Nonlinear MPC**: The system model is nonlinear and constraints are either linear or nonlinear. The cost function is usually chosen as a linear or quadratic function which results in a nonlinear programming problem which is non-convex.
- Based on the implementation
  1. **Implicit MPC**: This also known as the traditional MPC in which the control input at each time instant is computed by solving an optimization problem online.
  2. **Explicit MPC**: In this the online computation is reduced by transferring the optimization problem offline.

Thank you