AA203 Optimal and Learning-based Control
Lecture 11
Introduction to Model Predictive Control

Autonomous Systems Laboratory
Daniele Gammelli
Outline of the next two lectures

MPC: Basic setting and key ideas

Main design choices:
• Persistent feasibility
• Stability

Implementation aspects of MPC

Further reading:
Let’s consider the problem of controlling a F1 such that:

**Objective:** Minimize lap time

**Constraints:**
- Avoid other cars
- Stay on road
- Don’t skid
- Limited acceleration

An intuitive approach would be to use formulate this as an optimization problem and resort to open-loop approaches to compute a full trajectory.

What if something unexpected happens (e.g., unseen obstacle)?
Model Predictive Control (MPC)

Model predictive control (or, more broadly, receding horizon control) entails solving finite-time optimal control problems in a receding horizon fashion.

Specifically, given a model of the system:
- Obtain a state measurement
- Generate a plan by solving a finite-time open-loop problem for a pre-specified planning horizon
- Execute the first control action
- Repeat

Receding horizon introduces **feedback**
Model Predictive Control (MPC)

**Key steps:**
- At each sampling time $t$, solve an open-loop optimal control problem over a finite horizon.
- Apply optimal input signal during the following sampling interval $[t, t + 1)$.
- At the next time step $t + 1$, solve new optimal control problem based on new measurements of the state over a shifted horizon.
MPC in the wild

Trajectory Optimization

Library of Template Behaviors

Perception Driven

Model Predictive Control

Slide by Scott Kuindersma (Boston Dynamics)
Basic formulation - Linear System

• Consider the problem of regulating to the origin the discrete-time linear time-invariant system

\[ x(t + 1) = Ax(t) + Bu(t), \quad x(t) \in \mathbb{R}^n, \quad u(t) \in \mathbb{R}^m \]

Subject to constraints

\[ x(t) \in X, \quad u(t) \in U, \quad t \geq 0 \]

Where the sets \( X \) and \( U \) are polyhedra

• Historical note: MPC was originally developed in the context of chemical plant control
Notation

- \( x(t) \) is the state of the system at time \( t \)
- \( x_{t+k|t} \) is the state of the model at time \( t + k \), predicted at time \( t \) obtained by starting from the current state \( x_{t|t} = x(t) \) and applying to the system model
  \[
  x_{t+1|t} = Ax_{t|t} + Bu_{t|t},
  \]
the input sequence \( u_{t|t}, \ldots, u_{t+k-1|t} \)
- \( u_{t+k|t} \) to denote the input \( u \) at time \( t + k \) computed at time \( t \)

Note: \( x_{3|1} \neq x_{3|2} \)
Notation

Let $U^*_{t\rightarrow t+N|t} := \left\{ u^*_{t|t}, u^*_{t+1|t}, \ldots, u^*_{t+N-1|t} \right\}$ be the optimal solution to the short-term problem. The first element of $U^*_{t\rightarrow t+N|t}$ is applied to the system

$$u(t) = u^*_{t|t}(x(t)).$$

The optimization problem is then repeated at time $t + 1$ based on the new state $x_{t+1|t+1} = x(t + 1)$

Thus, we define the receding horizon control law as

$$\pi_t(x(t)) := u^*_{t|t}(x(t))$$

Which results in the following closed-loop systems:

$$x(t + 1) = Ax(t) + B\pi_t(x(t)) := f_{cl}(x(t), t)$$

(Preview: a central question will be to characterize the behavior of the closed-loop system)
Basic formulation - OCP

Assume that a full measurement of the state $x(t)$ is available at the current time $t$

The finite-time optimal control problem solved at each stage is

$$
J^*_t(x(t)) = \min_{u_t, \ldots, u_{t+N-1}|t} \left( l_T \left( x_{t+N|t} \right) + \sum_{k=0}^{N-1} l \left( x_{t+k|t}, u_{t+k|t} \right) \right)
$$

s.t 
- $x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t}$, $k = 0, \ldots, N - 1$
- $x_{t+k|t} \in X$, $k = 0, \ldots, N - 1$
- $u_{t+k|t} \in U$, $k = 0, \ldots, N - 1$
- $x_{t+N|t} \in X_f$
- $x_t|t = x(t)$

Why add a terminal cost and terminal constraints if what I really care about is the long-horizon problem?

$l_T$ and $X_f$ are key design decisions

Goal: Ensure that the short-horizon problem models the long-horizon problem
- $l_T$ approximates the “tail” of the cost
- $X_f$ approximates the “tail” of the constraints
Simplifying the notation: time-invariant systems

Note that the system, the constraints, and the cost function are time-invariant, hence, to simplify the notation, we (i) remove \( t \) and (ii) set \( t = 0 \), in the finite-time optimal control problem, namely

\[
J_0^*(x(t)) = \min_{u_0, \ldots, u_{N-1}} l_T(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k)
\]

\[
\text{s.t} \quad x_{k+1} = Ax_k + Bu_k, \quad k = 0, \ldots, N - 1
\]
\[
x_k \in X, \quad k = 0, \ldots, N - 1
\]
\[
u_k \in U, \quad k = 0, \ldots, N - 1
\]
\[
x_N \in X_f
\]
\[
x_0 = x(t)
\]

- Denote the optimal solution to the short-term problem \( U_0^*(x(t)) = \{ u_0^*, \ldots, u_{N-1}^* \} \)
- With the new notation, the closed-loop system becomes

\[
x(t + 1) = Ax(t) + B\pi(x(t)) := f_{cl}(x(t))
\]
Typical cost function

- 2-norm (i.e., constrained LQR)

\[ l_T(x_N) = x_N^T P x_N, \quad c(x_k, u_k) = x_k^T Q x_k + u_k^T R u_k, \quad P \succeq 0, Q \succeq 0, R > 0 \]

- 1-norm

\[ l_T(x_N) = \| P x_N \|_p, \quad l(x_k, u_k) = \| Q x_k \|_p + \| R u_k \|_p, \quad p = 1 \text{ or } \infty \]

where \( P, Q, R \) are full column ranks
Online model predictive control (MPC v0)

repeat

measure the state $x(t)$ at time instant $t$

obtain $U^*_0(x(t))$ by solving finite-time optimal control problem

if $U^*_0(x(t)) = \emptyset$ then ‘problem infeasible’ stop

apply the first element $u^*_0$ of $U^*_0(x(t))$ to the system

wait for the new sampling time $t + 1$
MPC Features

Pros:
- Any model
  - Linear
  - Nonlinear
  - Single/Multivariable
  - Constraints
- Any objective
  - Sum of squared errors
  - Sum of absolute errors
  - Economic objective
  - Minimum time

Cons:
- Computationally demanding (important when embedding controller on hardware)
- May or may not be feasible
- May or may not be stable
Example: Loss of feasibility

Consider the double-integrator

\[
x(t + 1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)
\]

Consider a receding horizon controller that solves the optimization problem

\[
J_0^*(x(t)) = \min_{u_0, \ldots, u_{N-1}} l_T(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k)
\]

with

\[
l_T(x_N) = x_N^T P x_N, \quad l(x_k, u_k) = x_k^T Q x_k + u_k^T R u_k, \quad N = 3, \quad P = Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 10, \quad X_f = \mathbb{R}^2
\]

Subject to input and state constraints

\[
-0.5 \leq u(k) \leq 0.5, \quad k = 0, \ldots, 3
\]

\[
\begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x(t) \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix}, \quad k = 0, \ldots, 3
\]
Example: Loss of feasibility
Example: Dependency on parameters

**Question:** can we tune parameters and solve this issue?

Consider the unstable system

$$x(t + 1) = \begin{bmatrix} 2 & 1 \\ 0 & 0.5 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t)$$

Consider a receding horizon controller that solves the optimization problem $J^*_0(x(t)) = \min_{u_0, \ldots, u_{N-1}} l_T(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k)$, with

$$l_T(x_N) = x_N^T P x_N, \quad l(x_k, u_k) = x_k^T Q x_k + u_k^T R u_k, \quad Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad X_f = \mathbb{R}^2, P = 0$$

Subject to input and state constraints

$$-1 \leq u(k) \leq 1, \quad k = 0, \ldots, N - 1$$

$$\begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x(t) \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix}, \quad k = 0, \ldots, N - 1$$
**Example: Dependency on parameters**

1. $R = 10$, $N = 2$: all trajectories unstable.
2. $R = 2$, $N = 3$: some trajectories stable.
3. $R = 1$, $N = 4$: more stable trajectories.

* Initial points with convergent trajectories
  - Initial points that diverge

**Take-away:**
Parameters for receding horizon control influence the behavior of the resulting closed-loop trajectories in a complex manner.
Main implementation issues

1. The controller may lead us into a situation where after a few steps the finite-time optimal control problem is infeasible \(\rightarrow\) \textit{persistent feasibility} issue

2. Even if the feasibility problem does not occur, the generated control inputs may not lead to trajectories that converge to the origin (i.e., closed-loop system is unstable) \(\rightarrow\) \textit{stability} issue

\textbf{Key question:} how do we guarantee that such a “short-sighted” strategy leads to effective long-term behavior?

One could consider two distinct approaches for doing this:

- Analyze closed-loop behavior directly \(\rightarrow\) generally very difficult
- Derive conditions on
  - terminal function \(l_f\) so that closed-loop stability is guaranteed
  - terminal constraint set \(X_f\) so that persistent feasibility is guaranteed
Outline of the next two lectures

MPC: Basic setting and key ideas

Main design choices:
• Persistent feasibility
• Stability

Implementation aspects of MPC

Further reading:
Addressing persistent feasibility

**Goal:** design MPC controller so that feasibility for all future times is guaranteed

**Approach:** leverage tools from *invariant set theory*

\[
J_0^*(x(t)) = \min_{u_0, \ldots, u_{N-1}} l_T(x_N) + \sum_{k=0}^{N-1} l(x_k, u_k)
\]

**s.t.**

\[
x_{k+1} = Ax_k + Bu_k, \quad k = 0, \ldots, N - 1
\]

\[
x_k \in X, \quad k = 0, \ldots, N - 1
\]

\[
u_k \in U, \quad k = 0, \ldots, N - 1
\]

\[
x_N \in X_f
\]

\[
x_0 = x(t)
\]

**Def:** Set of feasible initial states

\[
X_0 := \{ x_0 \in X \mid \exists (u_0, \ldots, u_{N-1}) \text{ such that } x_k \in X, u_k \in U, k = 0, \ldots, N - 1, x_N \in X_f \text{ where } x_{k+1} = Ax_k + Bu_k, k = 0, \ldots, N - 1 \}
\]

A control input can be found only if \( x(0) \in X_0 \)
Controllable sets

For the autonomous system $x(t + 1) = \phi(x(t))$ with constraints $x(t) \in X, u(t) \in U$, the one-step controllable set to set $S$ is defined as

$$\text{Pre}(S) := \{ X \in \mathbb{R}^n : \phi(X) \in S \}$$

For the system $x(t + 1) = \phi(x(t), u(t))$ with constraints $x(t) \in X, u(t) \in U$, the one-step controllable set to set $S$ is defined as

$$\text{Pre}(S) := \{ x \in \mathbb{R}^n : \exists u \in U \text{ such that } \phi(X, U) \in S \}$$
Control invariant sets

A set $C \subseteq X$ is said to be a **control invariant set** for the system $x(t + 1) = \phi(x(t), u(t))$ with constraints $x(t) \in X, u(t) \in U$, if:

$$x(t) \in C \Rightarrow \exists u \in U \text{ such that } \phi(x(t), u(t)) \in C \text{ for all } t$$

The set $C_\infty \subseteq X$ is said to be the **maximal control invariant set** for the system $x(t + 1) = \phi(x(t), u(t))$ with constraints $x(t) \in X, u(t) \in U$, if it is control invariant and contains all control invariant sets contained in $X$

Let’s define the equivalent for autonomous systems:

• a set $A \subseteq X$ is said to be a **positive invariant set** for the system $x(t + 1) = \phi(x(t))$ if $x(t) \in A \Rightarrow \phi(x(t)) \in A$

• the **maximal positive invariant set** contains all other positive invariant sets

Note on implementation: these sets can be computed by using the MPT toolbox (multi-parametric toolbox) [https://www.mpt3.org/](https://www.mpt3.org/)
Persistent feasibility lemma

Define the “truncated” feasibility set:

\[ X_1 := \{ x_1 \in X \mid \exists (u_1, \ldots, u_{N-1}) \text{ such that } x_k \in X, u_k \in U, k = 1, \ldots, N - 1 \text{ for } x_N \in X_f \text{ where } x_{k+1} = Ax_k + Bu_k, k = 1, \ldots, N - 1 \} \]

**Feasibility lemma:**

If set \( X_1 \) is a control invariant set for system \( x(t + 1) = Ax(t) + Bu(t), \quad x(t) \in X, \quad u(t) \in U, \quad t \geq 0 \), then the MPC law is persistently feasible
Persistent feasibility lemma

**Proof:**
1. Consider the preimage of $X_1$, $\text{Pre}(X_1) = \{ x \in \mathbb{R}^n : \exists u \in U \text{ such that } Ax + Bu \in X_1 \}$
2. Since $X_1$ is control invariant, it means that $\forall x \in X_1, \exists u \in U$ such that $Ax + Bu \in X_1$
3. Thus $X_1 \subseteq \text{Pre}(X_1) \cap X$
4. One can write $X_0 = \{ x_0 \in X | \exists u_0 \in U \text{ such that } Ax_0 + Bu_0 \in X_1 \} = \text{Pre}(X_1) \cap X$
5. Thus, $X_1 \subseteq X_0$
6. Pick some $x_0 \in X_0$. Let $U_0^*$ be the solution to the finite-time optimization problem, and $u_0^*$ be the first control. Let $x_1 = Ax_0 + Bu_0^*$
7. Since $U_0^*$ is clearly feasible, one has $x_1 \in X_1$. Since $X_1 \subseteq X_0$, one has $x_1 \in X_0$
8. Hence the next optimization problem is feasible!
Practical significance

- For $N = 1$, we can set $X_f = X_1$. If we choose the terminal set to be control invariant, then MPC will be persistently feasible independent of chosen control objectives and parameters.
- Designer can choose the parameters to affect performance (e.g., stability).

- How to extend this result to $N > 1$?
Persistent feasibility theorem

Feasibility theorem:
If set $X_f$ is a control invariant set for system $x(t+1) = Ax(t) + Bu(t)$, $x(t) \in X$, $u(t) \in U$, $t \geq 0$, then the MPC law is persistently feasible

Proof:
1. Define the “truncated” feasibility set:
   $$X_{N-1} := \{ x_{N-1} \in X \mid \exists u_{N-1} \text{ such that } x_{N-1} \in X, u_{N-1} \in U x_N \in X_f \text{ where } x_N = Ax_{N-1} + Bu_{N-1} \}$$
2. Due to the terminal constraint, we know that $Ax_{N-1} + Bu_{N-1} = x_N \in X_f$
3. Since $X_f$ is a control invariant set, there exists a $u \in U$ such that $x^+ = Ax_N + Bu_N \in X_f$
4. The above is exactly the requirement to belong to set $X_{N-1}$
5. Thus, $Ax_{N-1} + Bu_{N-1} = x_N \in X_{N-1}$
6. We have just proved that $X_{N-1}$ is control invariant
7. Repeating this argument, one can recursively show that $X_{N-2}, X_{N-3}, \ldots, X_1$ are control invariant
8. The persistent feasibility lemma then applies
Practical aspects of persistent feasibility

- The terminal set $X_f$ is introduced artificially for the sole purpose of leading to a sufficient condition for persistent feasibility.
- We want it to be large so that it does not compromise closed-loop performance.
- Though it is simplest to choose $X_f = \{0\}$, this is generally undesirable.
- We’ll discuss better choices in the next lecture.
Next time

• Stability of MPC
• Explicit MPC
• Practical considerations