AMPyC

Chapter 1: Introduction

30. May 2025

Authors: Prof. Melanie Zeilinger
Dr. Lukas Hewing

ETH Zurich

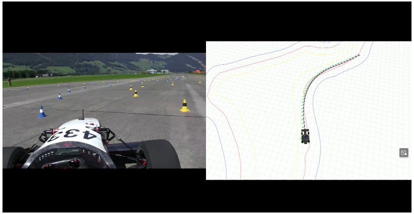
Outline

1. Basic MPC ideas
2. Motivation & Outlook
3. The ideal optimal control problem & approximations
4. Module Overview and Motivation
5. Examples
6. Brief recap of basic MPC results

Outline

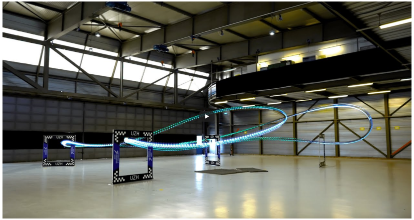
1. Basic MPC ideas
2. Motivation & Outlook
3. The ideal optimal control problem & approximations
4. Module Overview and Motivation
5. Examples
6. Brief recap of basic MPC results

Examples

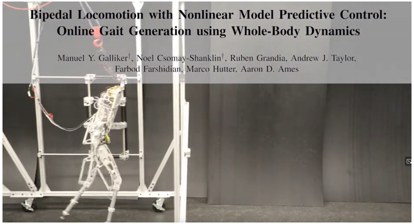


AMZ Driverless
https://www.youtube.com/watch?v=FbKLE7uar9Y

RSL ANYmal, ETHZ
https://www.youtube.com/watch?v=_rPvKlvyw2w



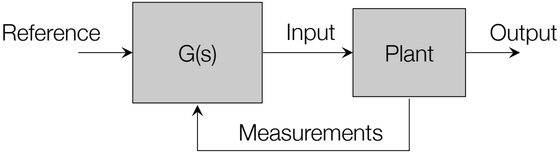
Robotics and Perception Group, UZH
https://www.youtube.com/watch?v=mHDQcckqdg4



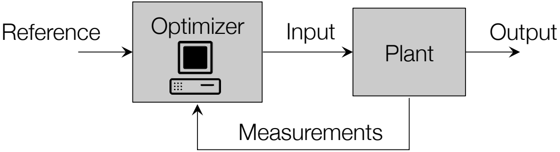
AMBER Lab, Caltech
https://www.youtube.com/watch?v=3g8ZNsCWd0A

Optimization in the loop

Classical control loop:



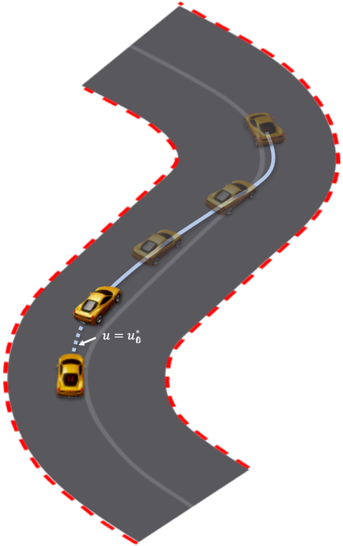
The classical controller is replaced by an optimization algorithm:



The optimization uses predictions based on a model of the plant.

Basic MPC problem formulation

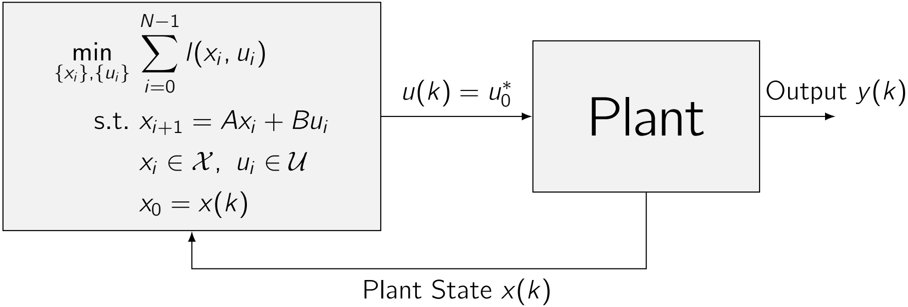
$$
\begin{aligned}
\min_{\{x_i\},\{u_i\}} \ & \sum_{i=0}^{N-1} l(x_i, u_i) & \\
\text{s.t. } & x_{i+1} = f(x_i, u_i) & \text{system model} \\
& x_i \in \mathcal{X} & \text{state constraints} \\
& u_i \in \mathcal{U} & \text{input constraints} \\
& x_0 = x(k) & \text{measurement/initialization}
\end{aligned}
$$



Problem is defined by

- **Objective** that is minimized
- Internal **system model** to predict system behavior
- **Constraints** that have to be satisfied

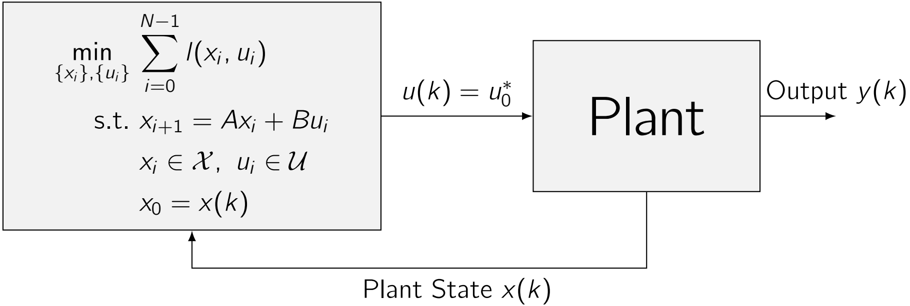
Optimization in the loop

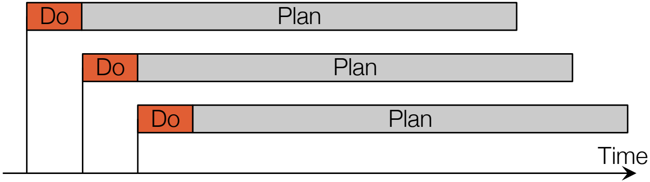


At each sample time:

- Measure / estimate current state $x(k)$
- Find the optimal input sequence for the entire planning window N: $U^* = \{u_0^*, u_1^*, \ldots, u_{N-1}^*\}$
- Implement only the **first** control action $u(k) = u_0^*$

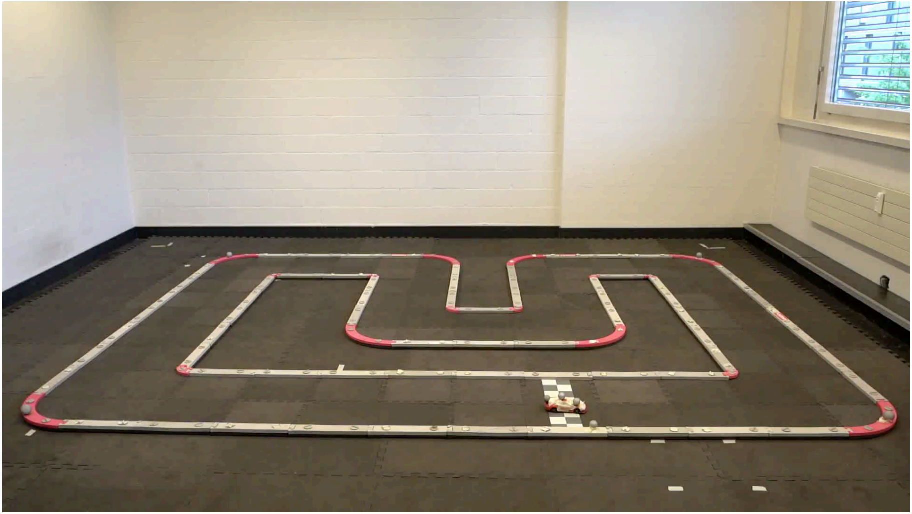
Optimization in the loop: Receding horizon control





Receding horizon strategy introduces **feedback**.

Examples: MPC for Autonomous Racing



Autonomous racing, CRS @ IDSC, ETH Zurich

Basic MPC assumptions

Assumptions:

- Regularization, set point tracking
- Linear systems
- No model uncertainties or bounded additive uncertainties

$\rightarrow$ MPC requires good model and suitable cost function.

$$
\begin{aligned}
\min_{\{x_i\},\{u_i\}} \; & \sum_{i=0}^{N-1} l(x_i, u_i) & \\
\text{s.t. } & x_{i+1} = Ax_i + Bu_i & \text{system model} \\
& x_i \in \mathcal{X} & \text{state constraints} \\
& u_i \in \mathcal{U} & \text{input constraints} \\
& x_0 = x(k) & \text{measurement/initialization}
\end{aligned}
$$

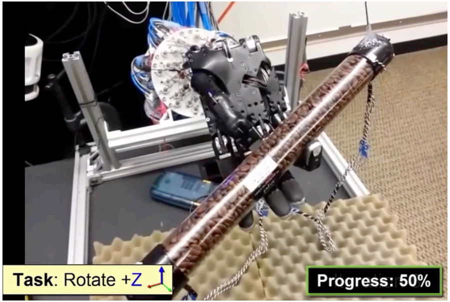
Outline

1. Basic MPC ideas
2. Motivation & Outlook
3. The ideal optimal control problem & approximations
4. Module Overview and Motivation
5. Examples
6. Brief recap of basic MPC results

Challenges

Which problems can we not address with basic MPC yet?

Following complex references / goals



[*ADROIT Manipulation Platform*]
https://www.youtube.com/watch?v=IN7VFquI7ow

[*Boston Dynamics*]
https://www.youtube.com/watch?v=kHBcVlqpvZ8

Optimizing a true **cost**

Example: Process control
Maximize yield, efficiency, ...

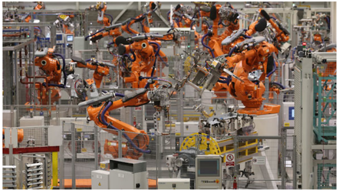
Example: Manufacturing
Maximize quality

Uncertainties and variation

Large-scale systems of systems

Horns Rev Wind Farm, Jha et. al, Wind Energy Symposium 2018

Future power grid (Photograph: Krunja / Dreamstime.com)



Factory (Credit: Oli Scarff/Getty Images)

Satellite formation flight (Credit: DARPA artist's concept)

Problem formulation?

$$
\begin{aligned}
\operatorname{argmin} \sum_{i=0}^{N-1} \quad & ? & \\
\text{s.t. } x_0 = x(k) \qquad & & \text{measurement} \\
? \qquad & & \text{system model} \\
? \qquad & & \text{state constraints} \\
? \qquad & & \text{input constraints} \\
? \qquad & & \text{optimization variables}
\end{aligned}
$$

Problem is defined by

- **Objective** that is minimized
- Internal **system model** to predict system behavior
- **Constraints** that have to be satisfied

Outline

Outline

3. The ideal optimal control problem & approximations

   System dynamics & uncertainties

   Cost function

   Constraints

The Real World

Simplifying assumption:

$$x(k+1) = f(x(k), u(k))$$

- System evolves in a predictable fashion

The real world:

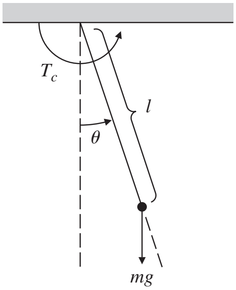
$$x(k+1) = f(x(k), u(k), w(k); \theta)$$

- Random noise w changes the evolution of the system
- Model structure is unknown (and potentially also state dimension)
- Unknown parameters θ impact the dynamics

Parametric uncertainties

Example: Pendulum with unknown weight or length

$$m\, l^2\, \ddot{\theta} = T_c - m\, g\, l \sin(\theta)$$



$\rightarrow$ Uncertain dynamics

Source of parametric uncertainties in practice:

- Production tolerances/variations
- Parameters cannot be measured directly (e.g. tire model)
- Parameters may change over time (e.g. tires)

$\rightarrow$ Parameter uncertainties can often be learned (in principle)

- $\theta \sim \mathcal{Q}^\theta$ (stochastic treatment)
- $\theta \in \Theta$ (robust treatment)

Model Mismatch

True system model is typically not available

- accurate modeling too time/cost intensive
- complex system model not suitable for controller design
- state used in control typically reduced

Structural model uncertainty can be addressed using robust bounds

$$\|f_t(x, u) - f(x, u)\| \leq \bar{w} \quad \forall x, u$$

$\rightarrow$ Global bound difficult to obtain and can be very conservative

Disturbances

$$x(k+1) = f(x(k), u(k), \mathbf{w(k)})$$

Typical forms:

- Additive: $f(x,u) + \mathbf{w}$
- Multiplicative: $f(x,u) + \mathbf{g}(\mathbf{x},\mathbf{u})\mathbf{w}$

Robust Approach

$$f(x,u) + \mathbf{w}, \text{ with } \mathbf{w} \in \mathcal{W}$$

Blanket term for all uncertainties (disturbances, model mismatch etc.!), but robust controller can be overly cautious/conservative

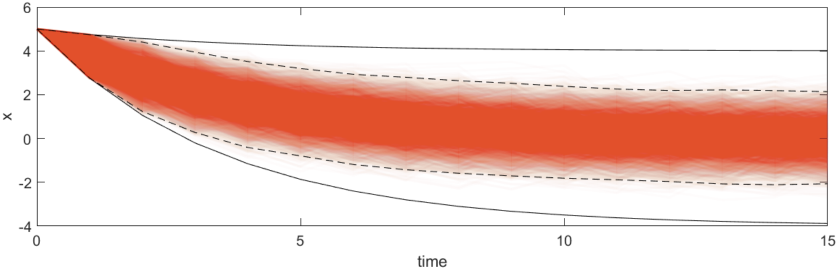
Stochastic Disturbances

$$f(x,u) + \mathbf{w}, \text{ with } \mathbf{w} \sim \mathcal{Q}^w, \text{ i.i.d.}$$

Can enable good practical solutions, **but** subject to specific assumptions

Example: Robust vs. Stochastic Treatment

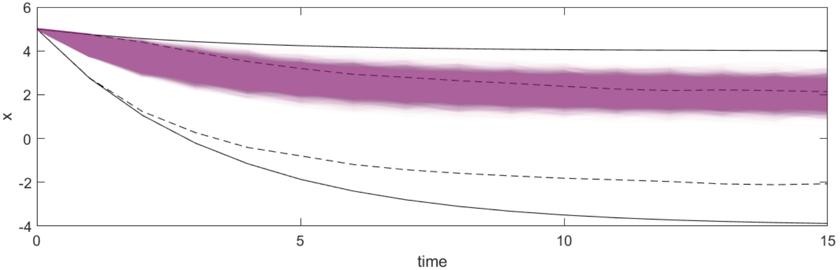
Scalar Linear system $\quad x(k+1) = 0.75x(k) + w(k)$
Additive Disturbance $\quad$ i.i.d. $w(k) \sim \mathcal{U}(-1, 1)$



- Robust bound encloses all trajectories, but 99% within dashed lines.
- Stochastic statement "more informative", but subject to specific assumptions

Example: Robust vs. Stochastic Treatment

Scalar Linear system $x(k+1) = 0.75x(k) + w(k)$
Additive Disturbance i.i.d. $\mathbf{w(k) \sim \mathcal{U}(0,1)}$ **("$\mathcal{U}(-0.5, 0.5) + 0.5$", steady-state offset)**

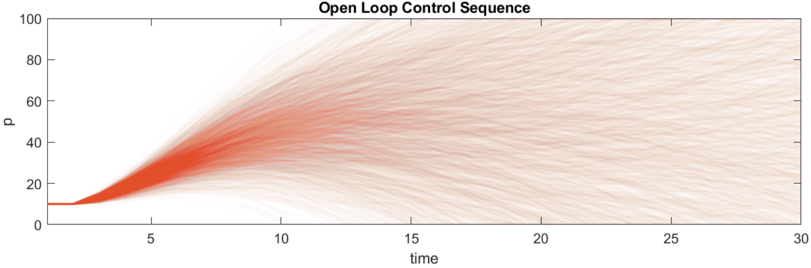


- Robust bound encloses all trajectories, but 99% within dashed lines.
- Stochastic statement "more informative", but subject to specific assumptions
- Robust bound applies whenever $x(k+1) - 0.75x(k) \in [-1, 1]$
  (model uncertainties, steady-state offset, state reduction, time-delays, ...)

Optimization over Feedback Policies

When considering model uncertainties, state evolution is not fully determined by input $u(k)$

- Simple control sequence $U = \{u(0), \dots, u(\bar{N}-1)\}$ is suboptimal and may be insufficient



- Optimization over policies $u(k) = \pi_k(\cdot)$ (with access to all past states & inputs)

$$\Pi = \{\pi_0, \dots, \pi_{\bar{N}-1}\}, \text{ with } \pi_k(x(0), \dots, x(k), u(0), \dots, u(k-1))$$

Special case: time-invariant state feedback $\pi_k = \pi(x)$

Outline

3. The ideal optimal control problem & approximations

    System dynamics & uncertainties

    Cost function

    Constraints

Cost function

Nominal cost: $$J(x(0), U) = \sum_{k=0}^{\bar{N}} l_k(x(k), u(k)), \text{ where } x(k+1) = f(x(k), u(k))$$

or $$J(x(0), U) = L(X, U) \text{ with } U = \{u(0), \dots, u(\bar{N}-1)\}, X = \{x(0), \dots, x(\bar{N})\}$$

- Optimize over complete task horizon $\bar{N}$ (possibly infinite)
- L can in principle represent complex objectives

Example:

Enforce terminal constraint $x(\bar{N}) \in \mathcal{X}_f$ via indicator function

$$L(X, U) = \mathbf{I}_{\mathcal{X}_f}(x(\bar{N}))$$

Cost function

What is a suitable cost for a "real" system, where

$$
\begin{aligned}
x(k+1) &= f(x(k), u(k), w(k), \theta) \\
u(k) &= \pi_k(x(0), \dots, x(k), u(0), \dots, u(k-1))
\end{aligned}
$$

and both X and U are functions of x_0, Π, $W = \{w(0), \dots, w(\bar{N}-1)\}$ and θ?

- Minimize the expected value (requires some assumption on the distribution of w, θ)

$$J(x(0), \Pi) := \mathbb{E}\left[L(X(x(0), \Pi, W, \theta), U(x(0), \Pi, W, \theta))\right]$$

- Take the worst-case

$$J(x(0), \Pi) := \max_{W \in \mathcal{W}^{\bar{N}}, \theta \in \Theta} L(X(x(0), \Pi, W, \theta), U(x(0), \Pi, W, \theta))$$

- Take the nominal case

$$J(x(0), \Pi) := L(X(x(0), \Pi, 0, \theta_{\text{nom}}), U(x(0), \Pi, 0, \theta_{\text{nom}})) = L(X_{\text{nom}}, U_{\text{nom}})$$

Cost function - Remarks

"True" cost function often not suited for direct optimization, e.g. due to

- Optimization over policies
- Expected value / inner maximization
- Differentiability
- Nonconvexity / Multimodality

Common approximation:

1. Formalize true objective
2. Compute reference
3. Use optimal control to track reference, e.g. LQR or nominal MPC with quadratic cost

Active research field: Economic MPC (not part of this lecture)

- Directly optimize "true" or "economic" cost in MPC, e.g. linear cost
- Significant theoretical challenges

$\Rightarrow$ Minimize "surrogate" objective (often stability)

Outline

3. The ideal optimal control problem & approximations

    System dynamics & uncertainties

    Cost function

    Constraints

Constraints

- Nominal control:

$$x(k) \in \mathcal{X}, u(k) \in \mathcal{U} \quad \forall k$$

- Robust constraint satisfaction for bounded uncertainties/disturbances:

$$x(k) \in \mathcal{X}, u(k) \in \mathcal{U} \quad \forall k, \forall w(k) \in \mathcal{W}, \theta \in \Theta$$

$\rightarrow$ satisfy constraints for all disturbance realizations
$\rightarrow$ generally achieved by using the concept of robust invariance

Constraints

- Nominal control:

$$x(k) \in \mathcal{X}, u(k) \in \mathcal{U} \quad \forall k$$

- Robust constraint satisfaction for bounded uncertainties/disturbances:

$$x(k) \in \mathcal{X}, u(k) \in \mathcal{U} \quad \forall k, \forall w(k) \in \mathcal{W}, \theta \in \Theta$$

$\rightarrow$ satisfy constraints for all disturbance realizations
$\rightarrow$ generally achieved by using the concept of robust invariance

- Probabilistic constraint satisfaction for stochastic uncertainties/disturbances:

$$Pr\left(x(k) \in \mathcal{X}, u(k) \in \mathcal{U}\right) \geq p \quad \forall k,\ w \sim \mathcal{Q}^w, \theta \sim \mathcal{Q}^\theta$$

$\rightarrow$ also called chance constraints (will be discussed in detail in the class)

Putting things together: The "true" optimal control problem

Optimal control problem that we ideally would like to solve – the robust case

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \max_{W,\theta} L(X, U) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(x(0), \dots, x(k), u(0) \dots, u(k-1)), \\
& X \in \mathcal{X}^{\bar{N}}, U \in \mathcal{U}^{\bar{N}} \ \forall W \in \mathcal{W}^{\bar{N}}, \theta \in \Theta, \\
& x(0) = x_{\text{init}}
\end{aligned}
$$

- State sequence $X = [x(0)^\top, \dots, x(\bar{N}-1)^\top]^\top$
- Input sequence $U = [u(0)^\top, \dots, u(\bar{N}-1)^\top]^\top$
- Disturbance sequence $W = [w(0)^\top, \dots, w(\bar{N}-1)^\top]^\top$

> But: we usually don't know f, $\mathcal{W}$, Θ, etc. exactly

Putting things together: The "true" optimal control problem

Optimal control problem that we ideally would like to solve – the stochastic case

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(L(X, U)\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(x(0), \dots, x(k), u(0) \dots, u(k-1)), \\
& W \sim \mathcal{Q}^W,\ \theta \sim \mathcal{Q}^\theta, \\
& Pr\left(x(k) \in \mathcal{X}, u(k) \in \mathcal{U}\right) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
$$

- State sequence $X = [x(0)^\mathsf{T}, \dots, x(\bar{N})^\mathsf{T}]^\mathsf{T}$
- Input sequence $U = [u(0)^\mathsf{T}, \dots, u(\bar{N})^\mathsf{T}]^\mathsf{T}$
- Disturbance sequence $W = [w(0)^\mathsf{T}, \dots, w(\bar{N})^\mathsf{T}]^\mathsf{T}$

But: we usually don't know f, $\mathcal{Q}^W$, $\mathcal{Q}^\theta$, etc. exactly

Two common approximations

Optimal control problem that we ideally would like to solve

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(L(X,U)\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(x(0), \ldots, x(k), u(0) \ldots, u(k-1)), \\
& W \sim \mathcal{Q}^W,\ \theta \sim \mathcal{Q}^\theta, \\
& Pr\left(x(k) \in \mathcal{X}, u(k) \in \mathcal{U}\right) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
$$

Model Predictive Control:

- Approximate objective, model, constraints
- Solve optimization problem numerically (in receding horizon)

Reinforcement learning:

- Evaluate objective empirically (typically in episodes)
- Iterative improvements (often model-free, no constraints)

Outline

1. Basic MPC ideas
2. Motivation & Outlook
3. The ideal optimal control problem & approximations
4. Module Overview and Motivation
5. Examples
6. Brief recap of basic MPC results

Module Overview

Optimal control problem that we ideally would like to solve

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(L(X, U)\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(x(0), \ldots, x(k), u(0) \ldots, u(k-1)), \\
& W \sim \mathcal{Q}^W,\ \theta \sim \mathcal{Q}^\theta, \\
& Pr\left(x(k) \in \mathcal{X}, u(k) \in \mathcal{U}\right) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
$$

Part I: MPC under uncertainty

Part II: Learning-based MPC

MPC under Uncertainty

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k), u(k))\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(\cdot), \\
& w(k) \sim \mathcal{Q}^w, \; \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x(k) \in \mathcal{X}) \geq p, \\
& \Pr(u(k) \in \mathcal{U}) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\min_{\{\pi_i\}} \quad & \mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i, u_i)\right) \\
\text{s.t.} \quad & x_{i+1} = f(x_i, u_i, w_i; \theta), \\
& u_i = \pi_i(\cdot), \\
& w_i \sim \mathcal{Q}^w, \; \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x_i \in \mathcal{X}) \geq p, \\
& \Pr(u_i \in \mathcal{U}) \geq p, \\
& x_N \in \mathcal{X}_f, \\
& x_0 = x(k)
\end{aligned}
$$

- Solve over shortened horizon
- Restrict policy class (open-loop sequence, state feedback,...)

MPC under Uncertainty – Outlook

1. Nominal MPC: No explicit uncertainty modelling
   + Often works in practice, often theoretical justification
   - Does not always work, hard to quantify "robustness"

2. Robust MPC: Explicitly consider worst-case disturbances
   + Quantifyable robustness properties (by design), blanket term for all disturbances/model mismatch
   - Can be conservative, can be computationally expensive, potentially increased design effort

3. Stochastic MPC: Explicitly consider disturbances according to (i.i.d.) distribution
   + Quantifiably probabilistic constraint satisfaction (by design), allows risk-performance tradeoff
   - Subject to specific model assumptions, guarantees (currently) only for linear systems, potentially increased design effort, (can be computationally expensive)

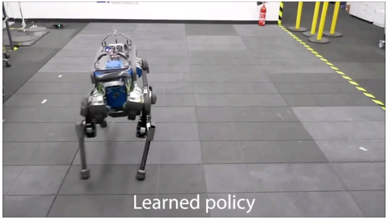
Learning-based Control

Various learning-based concepts have been considered in the literature.

For example: Adaptive Control, Iterative Learning Control, Reinforcement Learning

Example Demonstrations of Learning-based Control

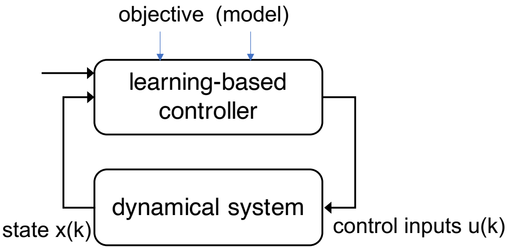
[*Google Deep Mind*]
`https://www.youtube.com/watch?v=gn4nRCC9TwQ`



[*Robotic Systems Lab, ETH Zurich*]
`https://www.youtube.com/watch?v=aTDkYFZFWug&feature=emb_logo`

[*UZH Robotics and Perception Group*]
`https://www.youtube.com/watch?v=2N_wKXQ6MXA`

Challenge of Learning-based Control: Safety



Dilemma:

Need to act to collect data & explore

But: Every action can be safety-critical

Goal of Learning-based MPC:

Combine constraint satisfaction, i.e. safety, with performance gained through learning

Where Learning is used in MPC

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k), u(k))\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(\cdot), \\
& w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x(k) \in \mathcal{X}) \geq p, \\
& \Pr(u(k) \in \mathcal{U}) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\min_{\{\pi_i\}} \quad & \mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i, u_i)\right) \\
\text{s.t.} \quad & x_{i+1} = f(x_i, u_i, w_i; \theta), \\
& u_i = \pi_i(\cdot), \\
& w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x_i \in \mathcal{X}) \geq p, \\
& \Pr(u_i \in \mathcal{U}) \geq p, \\
& x_N \in \mathcal{X}_f, \\
& x_0 = x(k)
\end{aligned}
$$

- Solve over shortened horizon
- Restrict policy class (open-loop sequence, state feedback,...)

Learning-based MPC: Improve MPC approximation by learning problem components from data

Where Learning is used in MPC

$$\begin{aligned} \min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k), u(k))\right) \\ \text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\ & u(k) = \pi_k(\cdot), \\ & w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\ & \Pr(x(k) \in \mathcal{X} \mid x(0)) \geq p, \\ & \Pr(u(k) \in \mathcal{U} \mid x(0)) \geq p, \\ & x(0) = x_{\text{init}} \end{aligned}$$

$$\Rightarrow$$

$$\begin{aligned} \min_{\{\pi_i\}} \quad & \mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i, u_i)\right) \\ \text{s.t.} \quad & x_{i+1} = f(x_i, u_i, w_i; \theta), \\ & u_i = \pi_i(\cdot), \\ & w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\ & \Pr(x_i \in \mathcal{X} \mid x_0) \geq p, \\ & \Pr(u_i \in \mathcal{U} \mid x_0) \geq p, \\ & x_N \in \mathcal{X}_f, \\ & x_0 = x(k) \end{aligned}$$

- **Learning the system dynamics**
  Improve model using (online) measurements

$\rightarrow$ Stochastic or robust models

$\rightarrow$ Parametric or non-parametric regression

Where Learning is used in MPC

$$\min_{\{\pi_k\}} \quad \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k), u(k))\right)$$
$$\text{s.t.} \quad x(k+1) = f(x(k), u(k), w(k); \theta),$$
$$u(k) = \pi_k(\cdot),$$
$$w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta,$$
$$\Pr(x(k) \in \mathcal{X}) \geq p,$$
$$\Pr(u(k) \in \mathcal{U}) \geq p,$$
$$x(0) = x_{\text{init}}$$

$$\Rightarrow$$

$$\min_{\{\pi_i\}} \quad \mathbb{E}\left(l_f(x_N; \theta_l) + \sum_{i=0}^{N-1} l_i(x_i, u_i; \theta_l)\right)$$
$$\text{s.t.} \quad x_{i+1} = f(x_i, u_i, w_i; \theta),$$
$$u_i = \pi_i(\cdot),$$
$$w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta,$$
$$\Pr(x_i \in \mathcal{X}) \geq p,$$
$$\Pr(u_i \in \mathcal{U}) \geq p,$$
$$x_N \in \mathcal{X}_f,$$
$$x_0 = x(k)$$

- **Learning the controller design**
  Optimize for closed-loop performance (usually episodic)

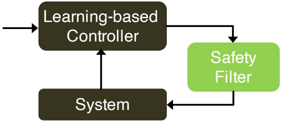
$\rightarrow$ Performance-driven learning: Bayesian / Convex optimization, Terminal components

$\rightarrow$ Inverse optimal control

Where Learning is used in MPC

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k),u(k))\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k),u(k),w(k);\theta), \\
& u(k) = \pi_k(\cdot), \\
& w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x(k) \in \mathcal{X}) \geq p, \\
& \Pr(u(k) \in \mathcal{U}) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\min_{\{\pi_i\}} \quad & \cancel{\mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i,u_i)\right)} \\
\text{s.t.} \quad & x_{i+1} = f(x_i,u_i,w_i;\theta), \\
& u_i = \pi_i(\cdot), \\
& w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x_i \in \mathcal{X}) \geq p, \\
& \Pr(u_i \in \mathcal{U}) \geq p, \\
& x_N \in \mathcal{X}_f, \\
& x_0 = x(k)
\end{aligned}
$$

- **MPC for safe learning**
  MPC only for constraint satisfaction



Where Learning is used in MPC

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k), u(k))\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k), u(k), w(k); \theta), \\
& u(k) = \pi_k(\cdot), \\
& w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x(k) \in \mathcal{X}) \geq p, \\
& \Pr(u(k) \in \mathcal{U}) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\color{red}{\tilde{\pi}(x(k)) \approx \arg} \min_{\{\pi_i\}} \quad & \mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i, u_i)\right) \\
\text{s.t.} \quad & x_{i+1} = f(x_i, u_i, w_i; \theta), \\
& u_i = \pi_i(\cdot), \\
& w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x_i \in \mathcal{X}) \geq p, \\
& \Pr(u_i \in \mathcal{U}) \geq p, \\
& x_N \in \mathcal{X}_f, \\
& x_0 = x(k)
\end{aligned}
$$

- **Approximating the MPC control law**
  Learn control law offline
  (speed up evaluation)

Where Learning is used in MPC

$$
\begin{aligned}
\min_{\{\pi_k\}} \quad & \mathbb{E}\left(\sum_{k=0}^{\bar{N}} l_k(x(k),u(k))\right) \\
\text{s.t.} \quad & x(k+1) = f(x(k),u(k),w(k);\theta), \\
& u(k) = \pi_k(\cdot), \\
& w(k) \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x(k) \in \mathcal{X}) \geq p, \\
& \Pr(u(k) \in \mathcal{U}) \geq p, \\
& x(0) = x_{\text{init}}
\end{aligned}
$$

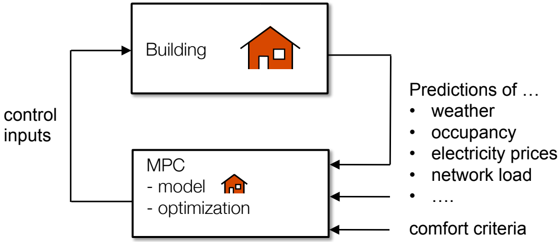
$$\Rightarrow$$

$$
\begin{aligned}
\min_{\{\pi_i\}} \quad & \mathbb{E}\left(l_f(x_N) + \sum_{i=0}^{N-1} l_i(x_i,u_i)\right) \\
\text{s.t.} \quad & x_{i+1} = f(x_i,u_i,w_i;\theta), \\
& u_i = \pi_i(\cdot), \\
& w_i \sim \mathcal{Q}^w,\ \theta \sim \mathcal{Q}^\theta, \\
& \Pr(x_i \in \mathcal{X}) \geq p, \\
& \Pr(u_i \in \mathcal{U}) \geq p, \\
& x_N \in \mathcal{X}_f, \\
& x_0 = x(k)
\end{aligned}
$$

- Learning the system dynamics
  Improve model using (online) measurements
- Learning the controller design
  Optimize for closed-loop performance
- MPC for safe learning
  MPC only for constraint satisfaction
- Approximating the MPC control law
  Learn control law offline (speed up evaluation)

Outline

Stochastic MPC for energy-efficient building control

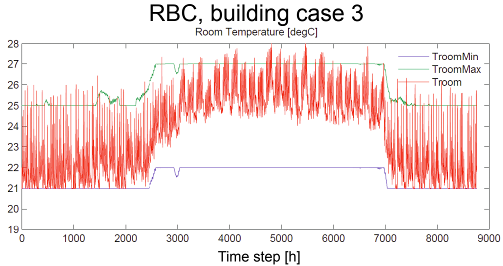


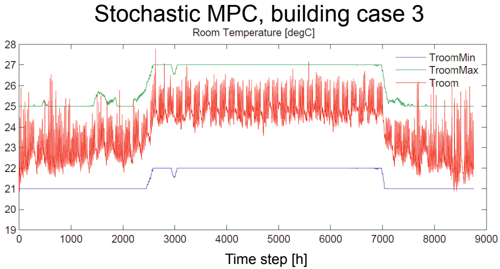
MPC opens the possibility to

- exploit building's **thermal storage capacity**
- use **predictions** of future disturbances, e.g. weather, for better planning
- take into account **stochastic prediction uncertainties**
- offer grid-balancing services to the power network
- ...

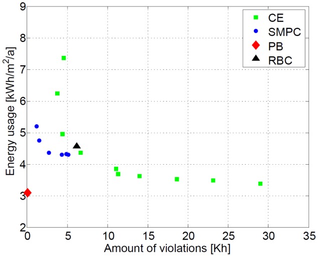
while respecting requirements for building usage (temperature, light, ...)

Stochastic MPC for energy-efficient building control



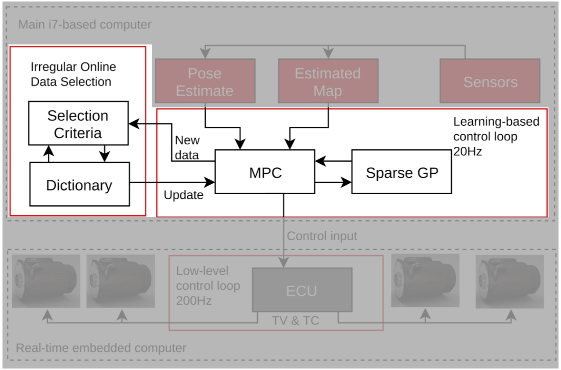


Stochastic MPC offers improved tradeoff between energy and constraint violation



Source: Oldewurtel et. al, "Energy Efficient Building Climate Control using Stochastic Model Predictive Control and Weather Predictions", American Control Conference 2010

Learning-based MPC for AMZ driverless race car

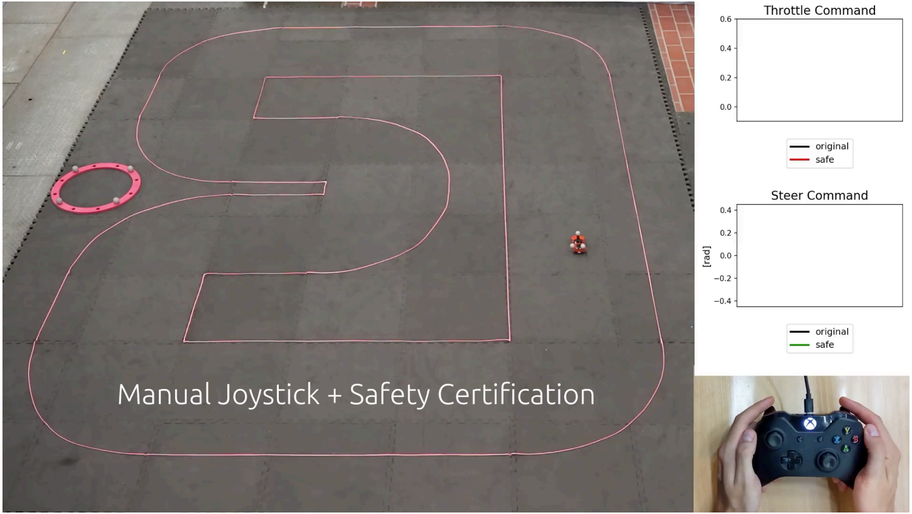


→ Continuous model improvement while racing

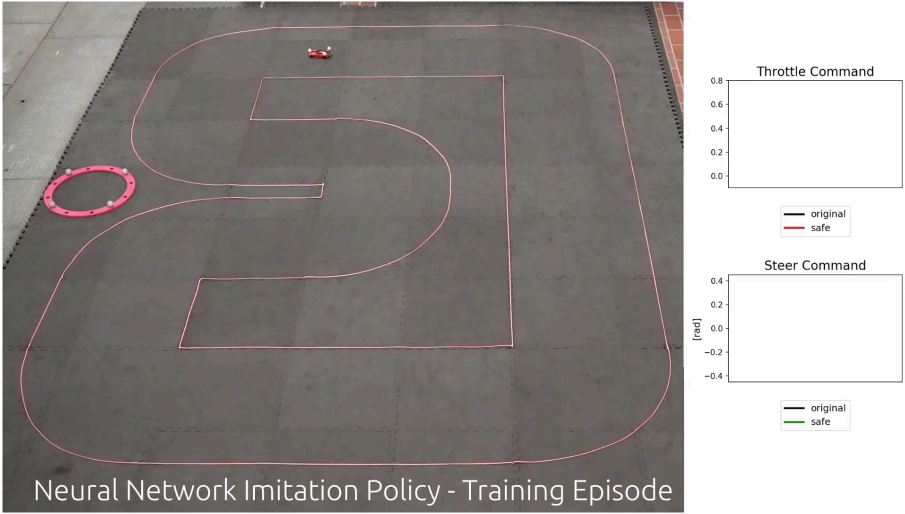
Learning-based MPC for AMZ driverless race car

https://www.youtube.com/watch?v=aCDPwZZm9C4

MPC as a safeguard for a human: Learning to race



MPC as a safeguard for RL: Learning to race



Outline

1. Basic MPC ideas
2. Motivation & Outlook
3. The ideal optimal control problem & approximations
4. Module Overview and Motivation
5. Examples
6. Brief recap of basic MPC results

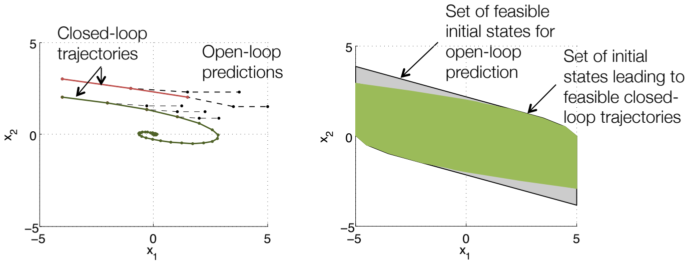
Basic MPC results

Assumptions:

- Regularization, set point tracking
- Linear systems
- No model uncertainties or bounded additive uncertainties

$\rightarrow$ MPC requires good model and suitable cost function.

Challenge for theory: Deviation between open-loop prediction and closed-loop control.



Basic nominal MPC problem with guarantees

$$
\begin{aligned}
J^\star(x(k)) = \min_{U}\ & l_f(x_N) + \sum_{i=0}^{N-1} l(x_i, u_i) \\
\text{s.t. } & x_{i+1} = Ax_i + Bu_i,\ \ i = 0, \dots, N-1 \\
& x_i \in \mathcal{X}, u_i \in \mathcal{U}, i = 0, \dots, N-1 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x(k)
\end{aligned}
$$

where $U = \{u_0, \dots, u_{N-1}\}$.

Truncate after a finite horizon:

- $l_f(x_N)$: Approximates the 'tail' of the cost
- $\mathcal{X}_f$: Approximates the 'tail' of the constraints

Basic MPC results

$$
\begin{aligned}
J^\star(x(k)) = \min_U \ & l_f(x_N) + \sum_{i=0}^{N-1} l(x_i, u_i) \\
\text{s.t. } & x_{i+1} = Ax_i + Bu_i, \ \ i = 0, \dots, N-1 \\
& x_i \in \mathcal{X}, u_i \in \mathcal{U}, i = 0, \dots, N-1 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x(k)
\end{aligned}
$$

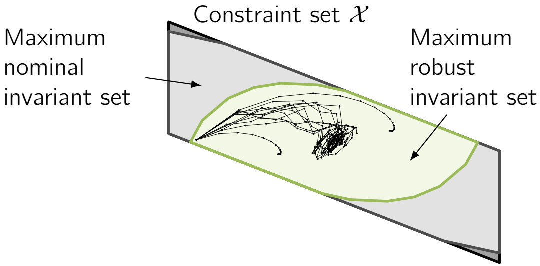
Nominal or Robust MPC theory:

- (Robust) Invariance

Robust Invariance

Robust constraint satisfaction, for **autonomous** system $x(k+1) = f(x(k), w(k))$, or **closed-loop** system $x(k+1) = f(x(k), \pi(x(k)), w(k))$ for a **given** controller π.

Robust Positive Invariant set
A set $\mathcal{O}^{\mathcal{W}}$ is said to be a robust positive invariant set for the autonomous system $x(k+1) = f(x(k), w(k))$ if $$x \in \mathcal{O}^{\mathcal{W}} \;\Rightarrow\; f(x, w) \in \mathcal{O}^{\mathcal{W}}\;, \quad \text{for all } w \in \mathcal{W}$$



If we have a (robust) invariant set $\mathcal{X}_f \subseteq \mathcal{X}$ and $\pi(\mathcal{X}_f) \subseteq \mathcal{U}$, then it provides a set of initial states from which the trajectory will never violate the system constraints if we apply the controller π.

Basic MPC results

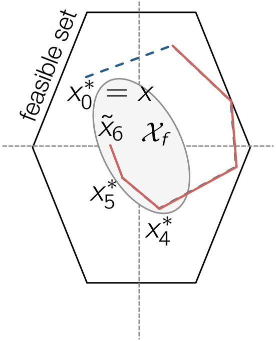
$$J^\star(x(k)) = \min_U \; l_f(x_N) + \sum_{i=0}^{N-1} l(x_i, u_i)$$

$$\text{s.t. } x_{i+1} = Ax_i + Bu_i,\;\; i = 0, \dots, N-1$$

$$x_i \in \mathcal{X},\, u_i \in \mathcal{U},\, i = 0, \dots, N-1$$

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

$$x_0 = x(k)$$

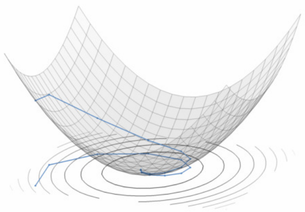


Nominal or Robust MPC theory:

- (Robust) Invariance
- Recursive feasibility –> Constraint satisfaction

Basic MPC results

$$
\begin{aligned}
J^\star(x(k)) = \min_U \; & l_f(x_N) + \sum_{i=0}^{N-1} l(x_i, u_i) \\
\text{s.t. } & x_{i+1} = Ax_i + Bu_i, \; i = 0, \ldots, N-1 \\
& x_i \in \mathcal{X}, u_i \in \mathcal{U}, i = 0, \ldots, N-1 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x(k)
\end{aligned}
$$



Nominal or Robust MPC theory:

- (Robust) Invariance
- Recursive feasibility –> Constraint satisfaction
- Asymptotic stability (via Lyapunov theory) –> Convergence