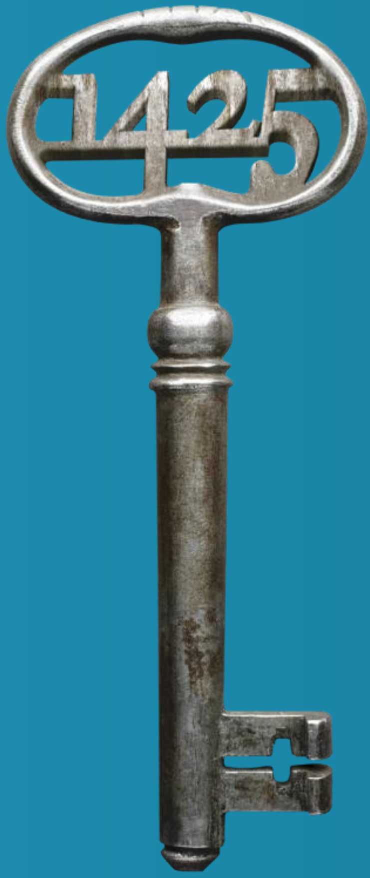
KU LEUVEN



Robustness Analysis of Model Predictive Control applied to a Hybrid Ground Coupled Heat Pump System

Stefan Antonov

Outline

- Motivation
- Introduction to a Robustness Analysis algorithm
- How it is supposed to work
- How does it fail
- Conclusions
- Questions I have to you
- Questions you would have to me ☺

KU LEUVEN

Motivation

- Why am I here with this strange topic?
- What do you gain if you care for my research?
  - A sufficient condition for your simulated MPC to work stable in practice.
  - Can be used as a fast check for robustness

It’s about an off-line algorithm, used to analyze a given system with MPC.

- It computes the maximum allowed uncertainty, for witch the controlled system remains robust
- It achieves that by using:
  - the controller model
  - the optimization problem of the MPC
  - the definition of uncertainty

KU LEUVEN

The main principle is that the objective function should decrease in time.

Given an optimal trajectory

$$|x_0 \quad u_o^* \quad u_1^* \quad \cdots \quad u_{N-1}^*|^T$$

define that the objective function decreases in time

$$J_N(x) - J_N(Ax + Bu^*(\hat{x})) \geq \epsilon||x||_2^2$$

KU LEUVEN

Then make sure that you stay close to the optimal trajectory.

Pre-solve the MPC optimization problem.

Take the objective function value $J_N(x)$

Divide it by a chosen initial state $\frac{J_N(x)}{x^T U_I x}$

Define that the objective function is bounded from above

$$J_N(x) \leq x^T U x$$

Finally, force the optimal trajectory to imply the objective function decrease

So, force

$$J_N(x) \leq x^T U x$$

to imply

$$J_N(x) - J_N(Ax + Bu^*(\hat{x})) \geq \epsilon \|x\|_2^2$$

given the uncertainty

$$\|\hat{x} - x\|_2 = err * \|x\|_2$$

KU LEUVEN

Or, in a matrix form

Force

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T \Pi_i \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} \geq 0, \qquad i = 0 \dots N$$

to imply

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T \Pi_s \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} \geq 0$$

Use S-procedure to do that

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T \Pi_i \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} \geq 0, \quad i = 0 \dots N \qquad \text{implies} \qquad \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T \Pi_s \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} \geq 0$$

if

$$\sum_{i=0}^{N} \tau_i \Pi_i - \Pi_s \preccurlyeq 0, \qquad \tau_i \geq 0, \qquad i = 0 \dots N$$

KU LEUVEN

So, just solve the LMI

$$\max(err)$$

s.t.

$$\tau_i \geq 0$$

$$\epsilon \geq 0$$

$$M \preccurlyeq 0$$

If the LMI was feasible, it holds that

$$\begin{vmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{vmatrix}^T M \begin{vmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{vmatrix} \leq 0$$

for all $|x_0 \quad u_0 \quad u_1 \quad \cdots \quad u_N|^T$

Then the system with MPC is robust for $err \in [0 \quad err^*]$

However, so far it was all about zero-tracking

In such case the MPC solves

$$\min\{x^T Qx + u^T Ru\}$$

In result I demand

$$\mathrm{M} \preccurlyeq 0$$

so that

$$\begin{vmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{vmatrix}^T M \begin{vmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{vmatrix} \leq 0$$

KU LEUVEN

But I need reference tracking

In such case the MPC solves

$$\min\left\{\left(x - x^{ref}\right)^T Q\left(x - x^{ref}\right) + u^T R u\right\}$$

In result I demand

$$\mathrm{M} \preccurlyeq 0, \qquad p^T M^{-1} p - 4q \leq 0$$

so that

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T M \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + p^T \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + q \leq 0$$

To avoid matrix inversion I reformulate

So, represent $\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}^T M \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + p^T \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + q \leq 0$

as

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \\ 1 \end{bmatrix}^T \begin{bmatrix} M & \frac{1}{2}p \\ \frac{1}{2}p^T & q \end{bmatrix} \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \\ 1 \end{bmatrix} \leq 0$$

And I try to solve the LMI

$$\max(err)$$

s.t.

$$\tau_i \geq 0$$

$$\epsilon \geq 0$$

$$\begin{vmatrix} M & \frac{1}{2}p \\ \frac{1}{2}p^T & q \end{vmatrix} \preccurlyeq 0$$

KU LEUVEN

If the LMI was feasible, it holds that

$$\begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \\ m \end{bmatrix}^T \begin{bmatrix} M & \frac{1}{2}p \\ \frac{1}{2}p^T & q \end{bmatrix} \begin{bmatrix} x_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \\ m \end{bmatrix} \leq 0$$

for all $\begin{bmatrix} x_0 & u_0 & u_1 & \cdots & u_N & m \end{bmatrix}^T$

But it is not feasible.

It is perhaps too much for all m. I only need for $m = 1$.

KU LEUVEN

Conclusions

1. (Imagine that it worked…)
   - A convenient way to quickly check the robustness of a system with MPC
   - Extended to the case for setpoint tracking

2. (…but it doesn't work yet.)
   - Difficulty to reach feasibility
   - Risky approach: an LMI is great! …when it is feasible…

I ask your opinion about:

1. How to relax $M \preccurlyeq 0$
   such that $\begin{vmatrix} x \\ 1 \end{vmatrix}^T M \begin{vmatrix} x \\ 1 \end{vmatrix} \leq 0$ holds for all x?
2. Is that relaxation needed at all?
3. Theory of Moments?
4. Sum of Squares?
5. Positive Polynomials?
6. Other trick?

KU LEUVEN

Stefan Antonov

stefan.antonov@kuleuven.be

KU LEUVEN