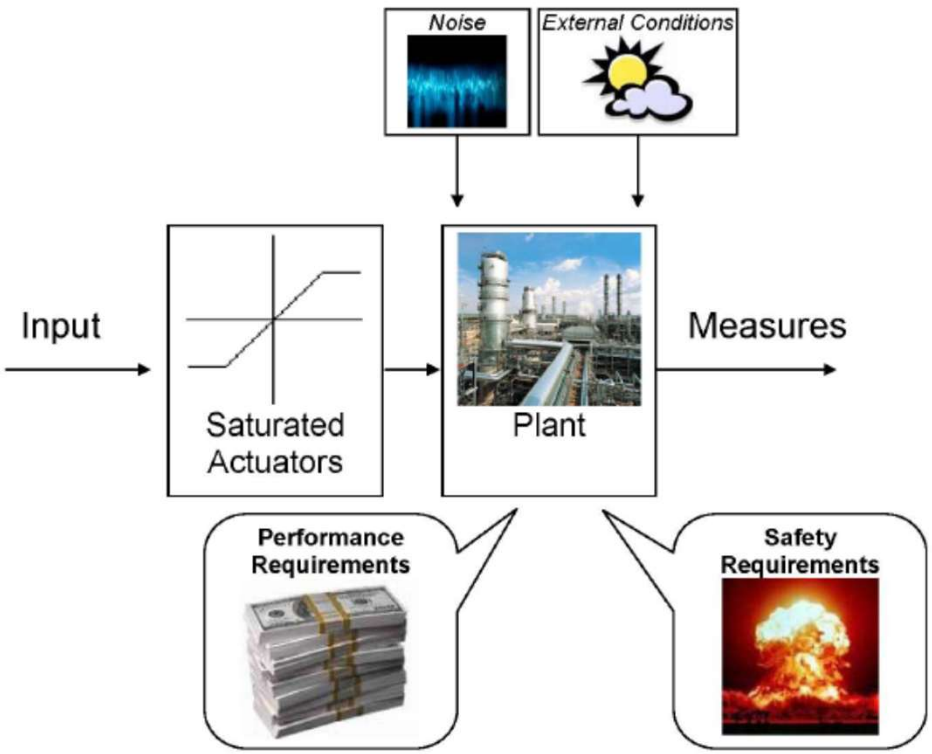
Control at large



Real-world plants are typically:

1. nonlinear systems,
2. subject to time-varying uncertainties (due to environmental conditions, aging, unmodeled dynamics, etc.)
3. Subject to noise and disturbances
4. Subject to input saturations

To control ideally we want:

1. To maximize performances
2. To satisfy constraints

Control at large – My personal experience

ULB

Theory

Adaptive Control
Networked Control
Control Allocation
Linear Parametric Varying Systems
Geometric Approach
Robust Control
Nonlinear Control
Sensor Selection
Distributed Control
Synchronization
Decentralized Control
Constrained Control
Cyber-Physical Security
Nonlinear Control
Extremum Seeking
Hybrid systems
Estimation

Applications

Aerospace
Robotics
Naval
Automotive
Batteries
Smart Grids
Wind-turbines
Wireless Sensor Networks
Agriculture
Medicine
Structures
Epidemics
Data Centers

Constrained Control

Every real-world system is subject to constraints:
- On the inputs (e.g. saturations)
- On the state

Whenever you need to push on performances, you will hit the constraints

Typical Constrained Control Problem

Control the system

$$x(t+1) = f\big(x(t), u(t)\big)$$

Subject to

$$x(t) \in \mathbb{X}$$
$$u(t) \in \mathbb{U}$$

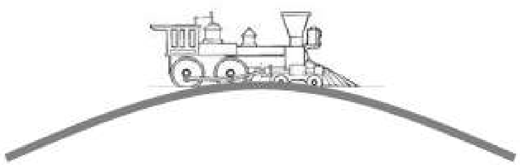
so that some control performance is maximized.

is this problem really so important ?

Is Constrained Control Relevant ?

The answer is **YES !**

Example: Let p, v be the position and velocity of a vehicle over a track with parabolic shape



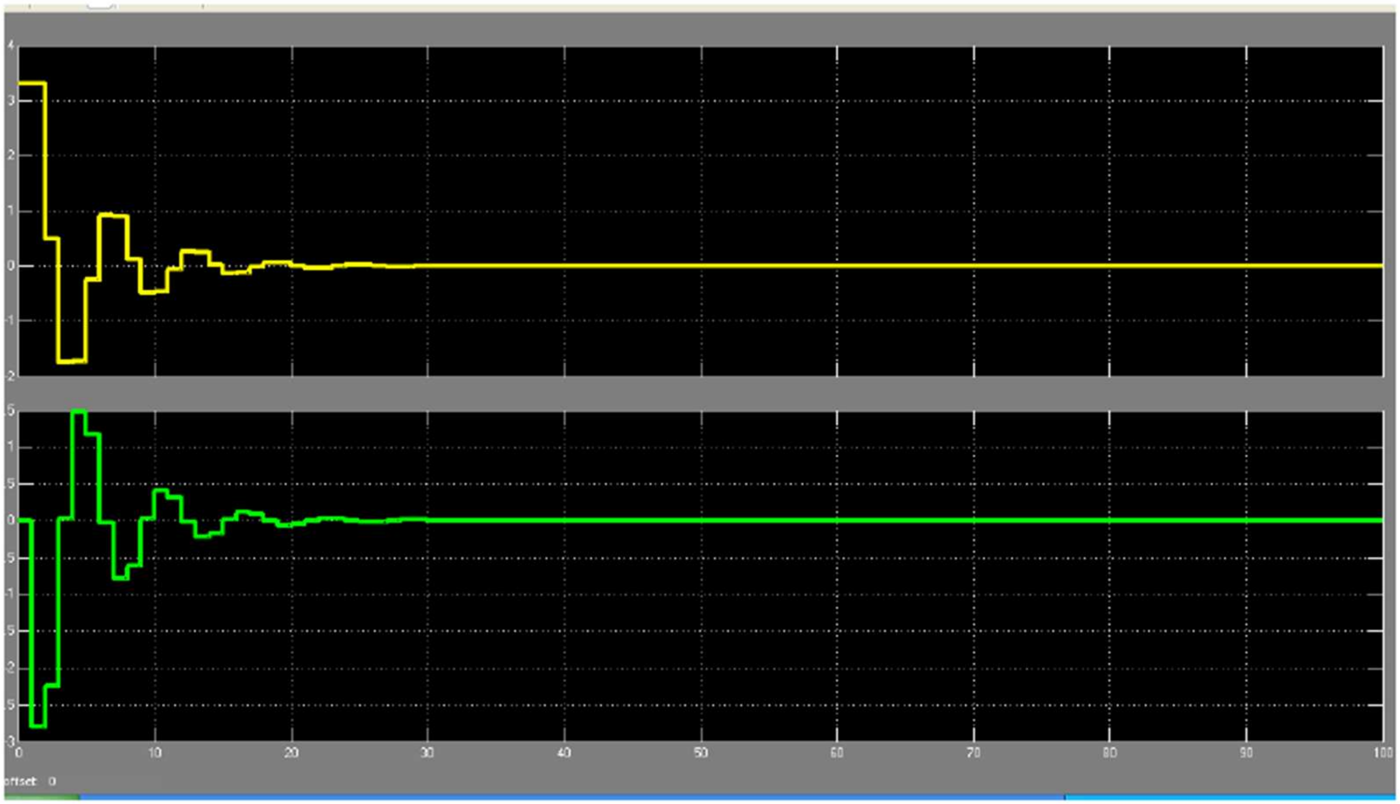
and let its model be the following (normalized) model

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

- The actuator is saturated (maximum force provided by the engine) $|u(t)| \leq 10$
- The system is controlled by a stabilizing control $u(t) = -2.85p(t) + 1.2v(t) + 0.85r_p(t)$ which stabilizes the linear system

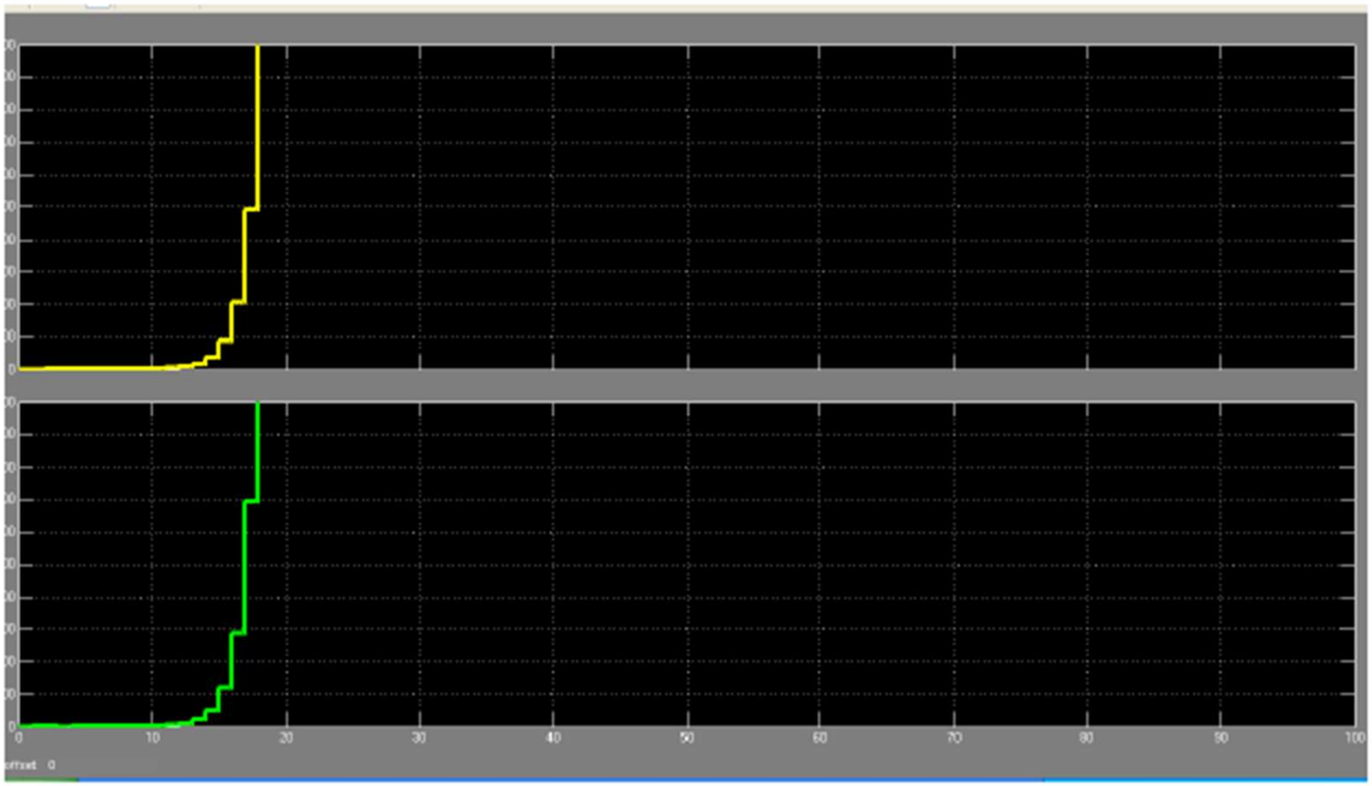
Is Constrained Control Relevant ?

if $p(0) = 3.295$, $v(0) = 0$ and the reference in position is $r_p(0) = 0$



Is Constrained Control Relevant ?

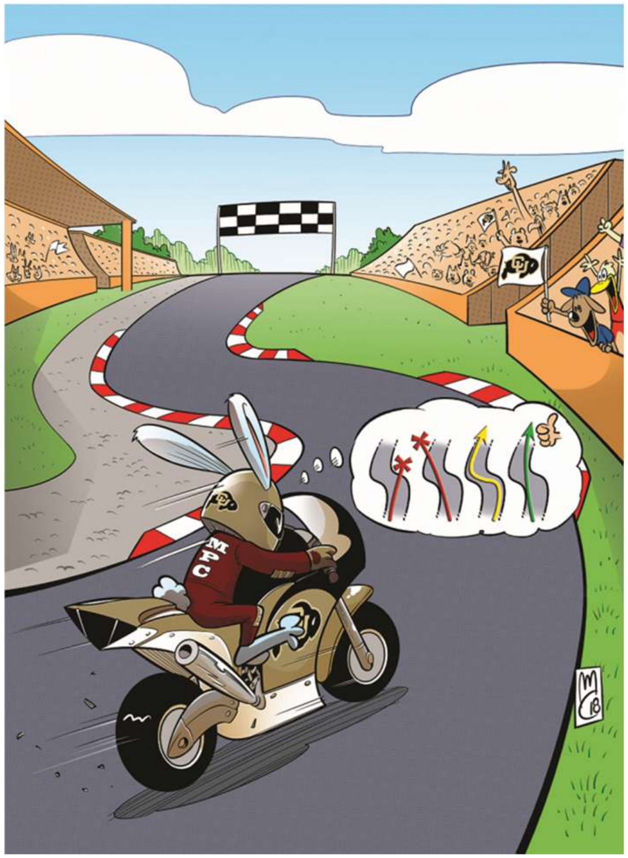
if $p(0) = 0$, $v(0) = 0$ and the reference in position is $r_p(0) = 3.295$



So yes, constrained control is VERY IMPORTANT

How to tackle constrained control?

ULB



The most used solution is

Model Predictive Control

Who invented MPC ?

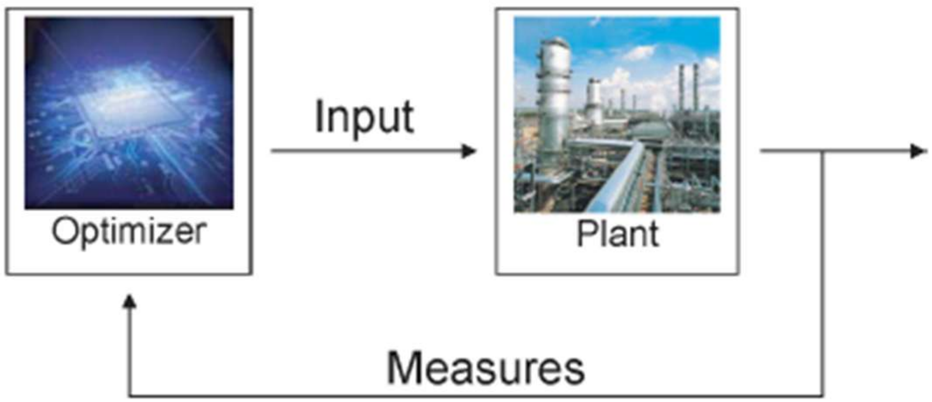
Model Predictive Control (MPC), a.k.a. *Receding Horizon Control (RHC)* and more rarely as *Model Based Predictive Control (MBPC)* and *Moving Horizon Optimal Control* is a **class** of control algorithms **widely used in the industry**

The Model Predictive Control ideas arose in the **petrochemical industry** :
- in 1978 a predictive technique (IDCOM) was developed by ADERSA, a small French company
- in 1979 Shell Oil (Houston, TX) described the Dynamic Matrix Control (DMN)

In the last 3 decades academia gave solid **theoretical foundation** to MPC and extended its use in many directions

What is MPC ?

MPCs computes, at each time instant, a control action as the result of an **optimization**



At each time t a general MPC does the following operations

1. **Compute a control strategy** $\hat{u}(t+k|t) \in \mathbb{U}, k = 0, \dots, N-1$ such that
   a. The state prediction satisfies the constraints $\hat{x}(t+k|t) \in \mathbb{X}, \quad k = 1, \dots, N$
   b. A performance index is optimized
2. **Apply** $\mathrm{u(t)} = \hat{u}(t+k|t)$

More Formally

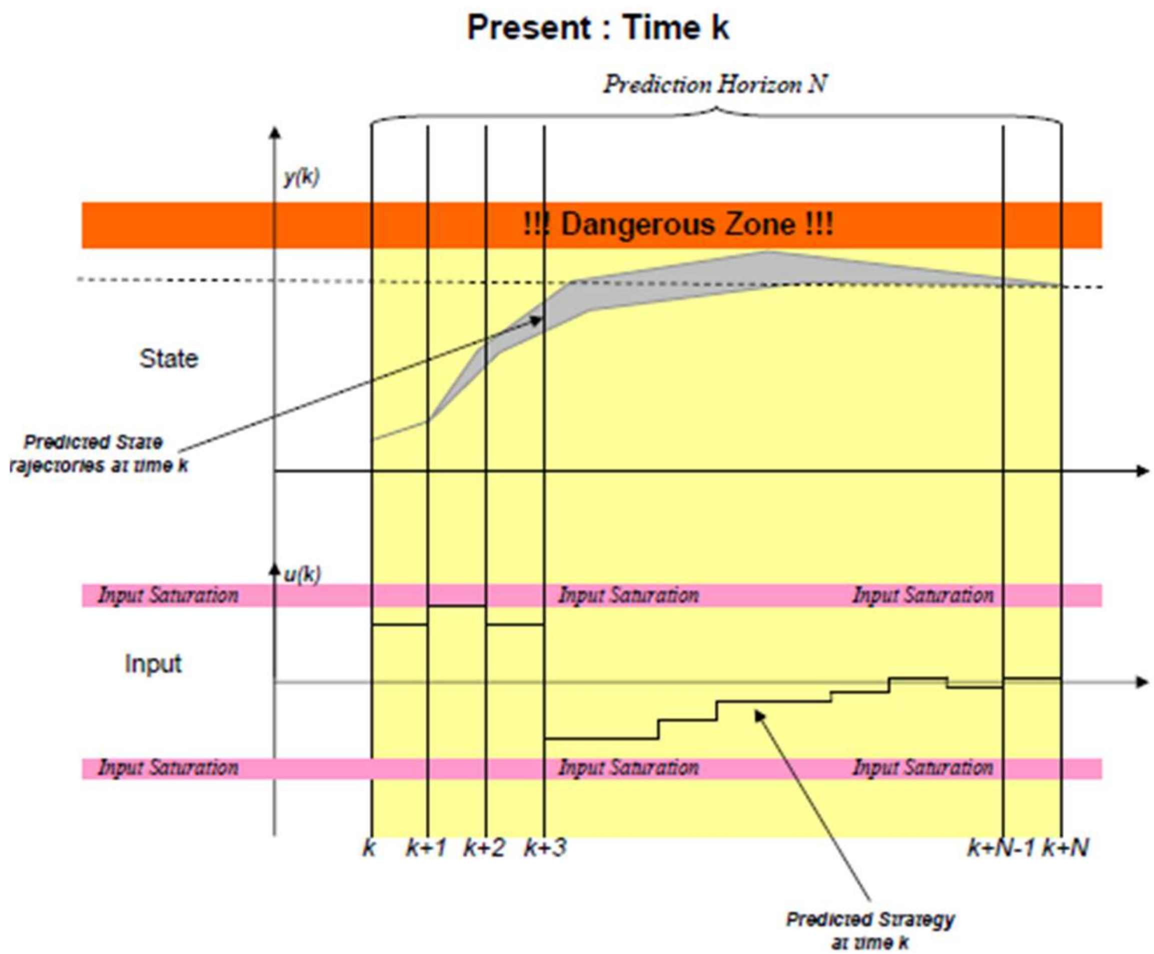
At each time t

1. Optimize

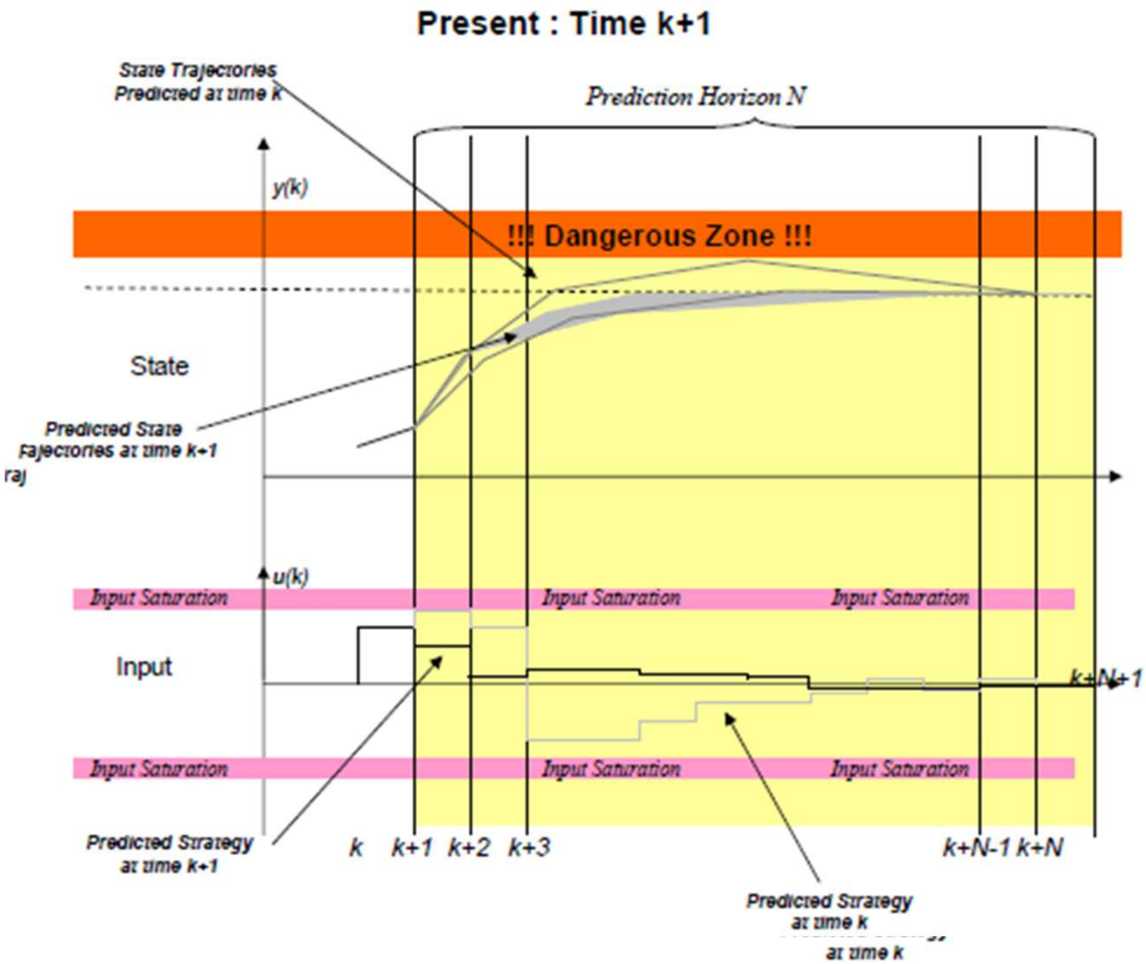
$$\begin{cases} \min\limits_{\hat{u}(\cdot|t),\hat{x}(\cdot|t)} J(\hat{u}(\cdot\,|t),\hat{x}(\cdot\,|t)) \\ \hat{u}(t+k|t) \in \mathbb{U}, \qquad k = 0,\dots,N-1 \\ \hat{x}(t+k|t) \in \mathbb{X}, \qquad k = 1,\dots,N \end{cases}$$

2. Apply $\mathrm{u(t)} = \hat{u}(t+k|t)$

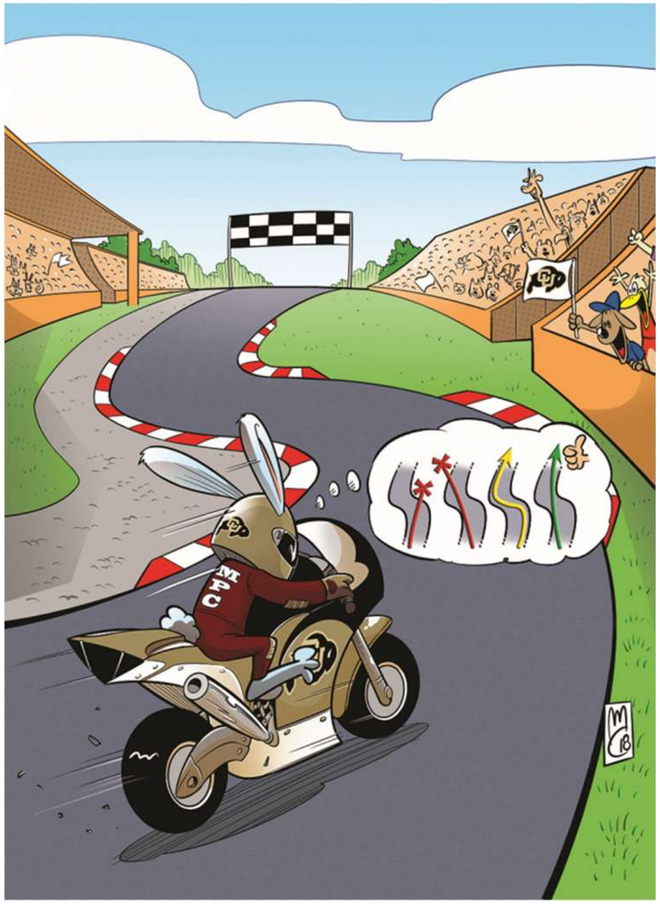
Receding Horizon Idea



Receding Horizon Idea



Analogies



Does it always work ?

ULB

At each time t

1. Optimize

$$\begin{cases} \min\limits_{\hat{u}(\cdot|t),\hat{x}(\cdot|t)} J(\hat{u}(\cdot\,|t),\hat{x}(\cdot\,|t)) \\ \hat{u}(t+k|t) \in \mathbb{U}, \qquad k=0,\dots,N-1 \\ \hat{x}(t+k|t) \in \mathbb{X}, \qquad k=1,\dots,N \end{cases}$$

2. Apply $\mathrm{u(t)} = \hat{u}(t+k|t)$

Question: Does MPC work properly for any $\mathrm{N}, \mathbb{X}, \mathbb{U}, J(\cdot)$?

The answer is no !

Feasibility Problem

FEASIBILITY PROBLEM: The existence of a feasible solution at time t, does not assure that in the future constraints will be satisfied

An example: Let us consider the following double integrator system

$$\begin{bmatrix} p(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p(k+1) \\ v(k+1) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k)$$

subject to constraints $|u(k)| < 1$ and $p(k) < 1$ and the cost function

$$J(\cdot) = \sum_{\tau=0}^{N-1} ||\hat{u}(k+\tau|k)||_I^2$$

Please, see what happens for the initial state $p(0) = -2, v(0) = 2$ and the horizon $N = 1$ and $N = 2$

Feasibility Problem

ULB

ORIGINAL : *C'est l'histoire d'un homme qui tombe d'un immeuble de 50 étages. Le mec, au fur et à mesure de sa chute, il se répète sans cesse pour se rassurer : " Jusqu'ici tout va bien... Jusqu'ici tout va bien... Jusqu'ici tout va bien. (La Haine, Kassovitz, 1995)*

ENGLISH : *Heard about the guy who fell off a skyscraper? On his way down past each floor, he kept saying to reassure himself: So far so good... so far so good... so far so good. (La Haine, Kassovitz, 1995)*

Stability Problem

STABILITY PROBLEM: Since we always apply only the first sequence of the strategy computed at each instant, the overall cost does not necessarily converge.

Computability Problem

ULB

COMPUTABILITY PROBLEM: It is not possible to solve the optimization problem within one sampling time

In many application **this is the key limitation of MPC**

Note that **computability**, unlike feasibility and stability, **is not strictly a property of the algorithm** but is a technological issues

Ensuring Reachability

How to build MPC schemes ensuring feasibility and stability ?

The right answer to the above question is : **it depends.** In particular it strongly depends on

- the class of system into consideration
- the form of the chosen moves (they can be both single values or functions of the state)

However there are general directions.

- **To ensure feasibility**, the usual approach consists in building algorithms ensuring that, if at time k a sequence of moves $\hat{u}(k|k), ..., \hat{u}(k+N-1|k)$ compatible with constraints is known, then at time $k+1$ a feasible (tough non optimal) sequence can be build as $\hat{u}(k+1|k), ..., \hat{u}(k+N-1|k), \hat{u}(k+N|k+1)$.

  Using this idea, the problem translates into ensuring the existence of an feasible $\hat{u}(k+N|k)$. A general case in which this thing is always true is *if the prediction horizon were* **infinite** the problem would be solved. *(Keerthi and Gilbert 1988, Rawlings and Muske 1993, Zheng and Morari 1995)*

Ensuring Stability

For what regard the **stability issue** the two main approaches used in literature are:

- To explicitly add **contraction constraints** by imposing that $x(k+1|k)$ is decreasing in some norm $||\hat{x}(k+1|k)|| \leq \alpha||\hat{x}(k)||, \alpha < 1$
- To use $V(x(k), N) = J(x(k), \hat{u}(k|k), ..., \hat{u}(k+N-1|k))$ as a **Lyapunov function and to guarantee that** $V(x(k+1), N) < V(x(k), N)$

Let us focus on the second approach. The standard way to prove the seen contraction is by proving for all possible x the following two steps:

1) $V(\hat{x}(k+1|k), N-1) < V(x(k|k), N)$
2) $V(x(k+1|k+1), N) \leq V(\hat{x}(k+1|k), N-1)$

While the first step is quite obvious (unless we use strange and typically meaningless cost functions), to guarantee the second some expedient have to be used.

Again note that *if the horizon N were* **infinite**, the second step would always be proved.

The idea of Infinite Horizon

The previous slides shows that, if it were possible to deal with an infinite horizon, feasibility and stability problems could be managed.

It is clear that in practice *it is not possible to* **explicitly** *compute infinite horizon sequences* !

However by means of simple considerations, an infinite horizon can be used **implicitly** by means of **state feedback**, **invariance properties** and **stability ideas**.

The main ideas are

1. To assume that **from** $k + N$ **onward the virtual commands results from a stabilizing control law** in the form $\hat{u}(k + \tau|k) = g(\hat{x}(k + \tau|k)), \tau \geq N$;
2. To **constraints the terminal state** $\hat{x}(k + N|k)$ to lie in a subset of the admissible set $\Omega_g \subseteq X$ such that:
   - is a **positive invariant set** for $x(t + 1) = f(x(t), g(x(t)))$, i.e. if $x(t) \in \Omega_g$ then $x(t + 1) = f(x(t), g(x(t))) \in \Omega_g$
   - $g(x) \in U, \forall x \in \Omega_g$
3. To find an **implicit cost function** $c(\hat{x}(k + N|k))$ such that $V(x(k), \infty) = V(x(k), N) + c(x(k + N|k))$

For several class of system (Linear, Polytopic Uncertain System, Polytopic LPV systems, etc...) such an idea is successful.

In the next slide we will discuss the linear case by partially following (Skokaert and Rawlings, IFAC 1996)

(Skokaert and Rawlings, IFAC 1996) - The Idea

Let us consider a linear system in the form

$$x(k+1) = Ax(k) + Bu(k)$$

subject to constraints:

$$x(k) \in X$$
$$u(k) \in U$$

with X, U intersections of ellipsoidal set containing 0. Moreover, once a prediction horizon N is fixed, let assume that

1. from $k+N$ onward the virtual commands will be $\hat{u}(k+\tau|k) = K\hat{x}(k+\tau|k), \tau \geq N$;
2. $\hat{x}(k+N|k)$ is constrained to lie in a set $\Omega_K \subseteq X$ where $\Omega_F = \{x : x^T W x \leq 1\}, \;\; W \geq 0$
3. W is such that:
   - $(A+BK)^T W(A+BK) - W \leq 0$
   - $Kx \in U$ for all $x \in \Omega_K$
4. P is a matrix such that $P = (A+BK)^T P(A+BK) + \Psi_x + K^T \Psi_u K$

Under the above assumption we can prove that for the MPC scheme that at each time k optim the following constrained problem

$$min \;\; J(\cdot) = \sum_{\tau=1}^{N-1} ||\hat{x}(k+\tau|k)||^2_{\Psi_x} + ||\hat{u}(k+\tau|k)||^2_{\Psi_u} + ||\hat{x}(k+N|k)||^2_P$$

$$subject \quad to : \begin{cases} \hat{x}(k+\tau|k) \in X, & \tau = 1, ..., N-1 \\ \hat{u}(k+\tau|k) \in U, & \tau = 0, ..., N-1 \\ \hat{x}(k+N|k) \in \Omega_K \end{cases}$$

feasibility and stability are ensured.

Remark: it is possible to make the choice of matrices P, W and K be variable of the convex

Conclusions

In conclusion an effective MPC scheme has to ensure:
- feasibility
- (possibly) **stability**

So, before using an MPC, be sure that it has these properties.

Moreover every time we use an MPC algorithm over a plant we have to be sure that the **computational time** complies with the sampling time of the chosen plant.

Check in the literature !

Constrained Control

MPC – Huge literature. Many variants for different kind of systems/problems

Linear/Nonlinear/Hybrid with different kind of constraints
With different types of model uncertainties and disturbance
etc.

Mayne; Rawlings, Rao, Scokaert (2000). ”Constrained model predictive control: stability and optimality”. Automatica

Reference Governor Framework – a special type MPC allowing to solve the problem of **tracking under constraints**

Garone, Di Cairano, Kolmanovsky, “Reference and Command Governors for Systems with Constraints: A Survey on Theory and Applications“, *Automatica*, 2017.

It's almost the end...

ULB

A huge thanks to Leo Ortolani for authorizing the (ab)use of his artwork and of the Ratman character.
No rat has been harmed during the realization of this presentation.