

**This document must be cited according to its final version  
which is published in a conference proceeding as:  
K. Abid<sup>1</sup>, P. Dufour<sup>1</sup>, I. Bombard<sup>1</sup>, P. Laurent<sup>1</sup>,  
« Model predictive control of a powder coating curing process:  
an Application of the MPC@CB© Software »,  
Proceedings of the 26th IEEE Chinese Control Conference (CCC) 2007,  
Zhangjiajie, China, vol. 2, pp. 630-634,  
July 27-29 2007.  
<http://dx.doi.org/10.1109/CHICC.2006.4346973>**

**All open archive documents of Pascal Dufour are available at:  
<http://hal.archives-ouvertes.fr/DUFOUR-PASCAL-C-3926-2008>**

**The professional web page (Fr/En) of Pascal Dufour is:  
<http://www.lagep.univ-lyon1.fr/signatures/dufour.pascal>**

1

Université de Lyon, Lyon, F-69003, France; Université Lyon 1;  
CNRS UMR 5007 LAGEP (Laboratoire d'Automatique et de GENie des Procédés),  
43 bd du 11 novembre, 69100 Villeurbanne, France  
Tel +33 (0) 4 72 43 18 45 - Fax +33 (0) 4 72 43 16 99  
<http://www-lagep.univ-lyon1.fr/> <http://www.univ-lyon1.fr> <http://www.cnrs.fr>



# Model Predictive Control of a Powder Coating Curing Process: an Application of the MPC@CB<sup>®</sup> Software

by: Kamel Abid, **Pascal Dufour**, Isabelle  
Bombard, Pierre Laurent

CCC'07, Zhangjiajie, July, 27-29 2007

# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

# Outline

1. *Control problem statement*
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

# Control problem statement

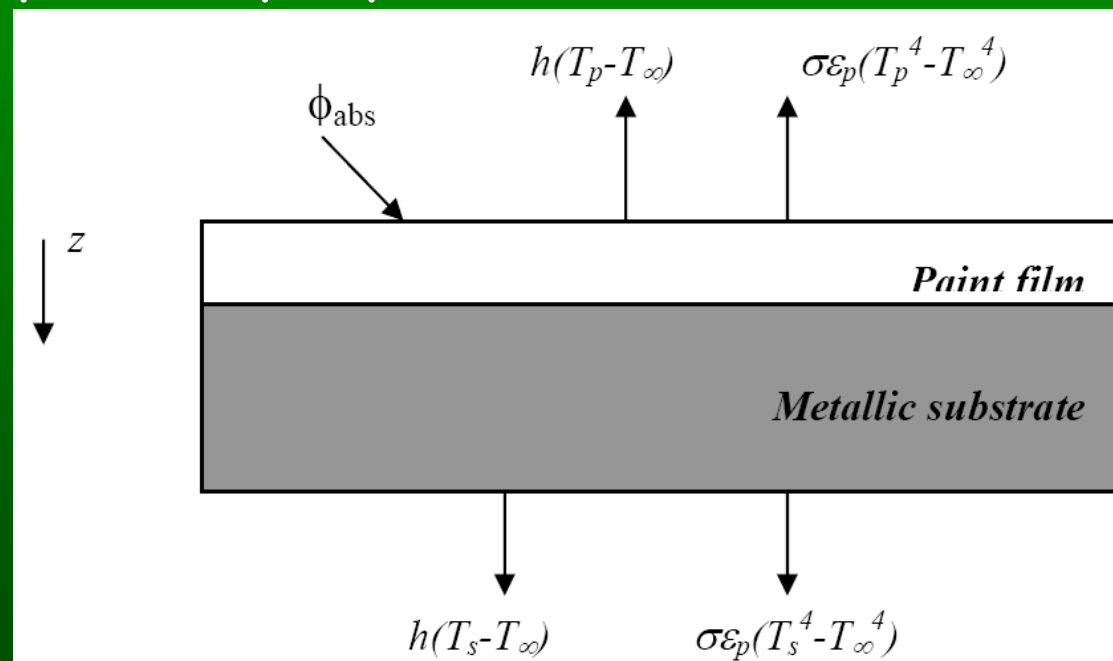
- **Painting** applications in: building (outdoor and indoor), furniture, cars accessories ...
- Need to **decrease pollution** due to the painting: organic solvent based coating are replaced by powder coatings
- Quite recently, **UV-curable powder coatings** and low-temperature coatings designed for heat sensitive substrates have appeared on the coatings market.
- But few studies on curing kinetics, thermal modelling and **control of such powder coating curing process**: trajectory tracking or minimization of curing time or ...

# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

# First principle PDE model

**Powder coatings** = fine particles of: resin + cross-linker in thermosetting or thermoplastic powder coatings + pigments + extenders + flow additives and fillers to achieve specific properties (**color**, ....)



Schematic drawing of the "substrat+powder" sample

# First principle PDE model

1. Thermal model based on the Fourier law of heat conduction uses:
  1. the **temperature** variable varying in the thickness of the powder coated metal sample
  2. the **degree of cure conversion** variable (ranging from 0+ to 1 at the end)
2. A **non linear PDE Boundary control problem** has to be tackled



# First principle PDE model *[Bombard et al, 2006]*

$$\frac{\partial T_p(z,t)}{\partial t} = \frac{\lambda_{c,p}}{\rho_p C_{pp}} \frac{\partial^2 T_p(z,t)}{\partial z^2} - \frac{e_p \Delta H_0}{C_{pp}} k_0 e^{\left(\frac{-E_a}{RT_p(z,t)}\right)} x^m (1-x)^n \quad \forall z \in ]0, e_p[ \quad \forall t > 0$$

$T_p(z,t)$  = temperature across the powder film thickness

$e_p$  = film thickness (~0.1 mm)

$x(z,t)$  = degree of cure

$$\frac{\partial T_s(z,t)}{\partial t} = \frac{\lambda_{c,s}}{\rho_s C_{ps}} \frac{\partial^2 T_s(z,t)}{\partial z^2} \quad \forall z \in ]e_p, e_p + e_s[ \quad \forall t > 0$$

$T_s(z,t)$  = temperature across the substrate

$e_s$  = film thickness (~1 mm)

# First principle PDE model [Bombard et al, 2006]

3 boundary conditions for the temperature:

$$-\lambda_p \frac{\partial T_p(z,t)}{\partial z} = \alpha_{ir} \phi_{ir}(t) - \sigma \varepsilon_p (T_p^4(z,t) - T_{ext}^4) - h_p (T_p(z,t) - T_{ext})$$

at  $z = 0, \forall t > 0$

**Manipulated variable**

$$-\lambda_{c,p} \frac{\partial T_p(z,t)}{\partial z} = -\lambda_{c,s} \frac{\partial T_p(z,t)}{\partial z} \quad \text{at } z = e_p, \forall t > 0$$

$$-\lambda_s \frac{\partial T_s(z,t)}{\partial z} = -\sigma \varepsilon_s (T_s^4(z,t) - T_{ext}^4) - h_s (T_s(z,t) - T_{ext})$$

at  $z = e_p + e_s, \forall t > 0$

# First principle PDE model *[Bombard et al, 2006]*

The degree of cure  $x(z,t)$  of the powder:

$$\frac{\partial x(z,t)}{\partial t} = k_0 e^{\left(\frac{-E_a}{RT_p(z,t)}\right)} x^m (1-x)^n \quad \forall z \in [0, e_p], \forall t > 0$$

Initial conditions:

$$T_p(z,t) = T_s(z,t) = T_{ext} \quad \forall z \in [0, e_p + e_s], t = 0$$

$$x(z,t) = 0^+ \quad \forall z \in [0, e_p], t = 0$$

# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

# Model predictive control strategy

## Advantages:

- **constraints** (such as manipulated variables physical limitations, constraints due to operating procedures or safety reasons...) may be specified
- a model aims to **predict** the future behavior of the process and the best one is chosen by a correct **optimal control** of the manipulated variables.

## Drawbacks:

- **computational time** needed may limit online use
- **suboptimal solutions**
- **how to handle unfeasibilities**

# Model predictive control strategy

$$\min_u J(u) = \sum_{j=k+1}^{j=k+N_p} [f(y_{ref}(j), y_p(j))]$$

subject to constraints on the manipulated variable :

$$u_{min} \leq u(j) \leq u_{max} \forall j \in \{k+1, k+N_p\}$$
$$\Delta u_{min} \leq u(j) - u(j-1) \leq \Delta u_{max} \forall j \in \{k+1, k+N_p\}$$

subject to constraints on the controlled variable :

$$c(y_p(j), y_{ref}(j)) \leq 0 \forall j \in \{k+1, k+N_p\}$$

The function **f** means: trajectory tracking, processing time minimization, productivity function ...

# Model predictive control strategy

*[Dufour et al, IEEE TCST 11(5) 2003]*

- Originally developed for nonlinear PDE model control
- Main idea: decrease the online time needed to compute the PDE model based control

## Approach:

- Input constraints: hyperbolic transformation
- Output constraints: exterior penalty method
- Linearization + sensitivities computed off line
- On line use of a time varying linear model
- On line resolution of a penalized (and so unconstrained) optimization control problem : a modified Levenberg Marquardt Algorithm

# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. **MPC@CB<sup>©</sup> software main features**
5. Simulation results
6. Experimental results
7. Conclusions & perspectives



# MPC@CB© software main features

1. Developed under Matlab, MPC@CB© solves any user defined :
  - trajectory tracking problem
  - operating time minimization problem
  - any cost function
  - input/output constraint handled
2. Any user defined continuous model (SISO, MISO, SIMO, MIMO model), including large scale PDE model
3. Easy to introduce a user defined observer
4. Easy to apply the software for simulation or real time application

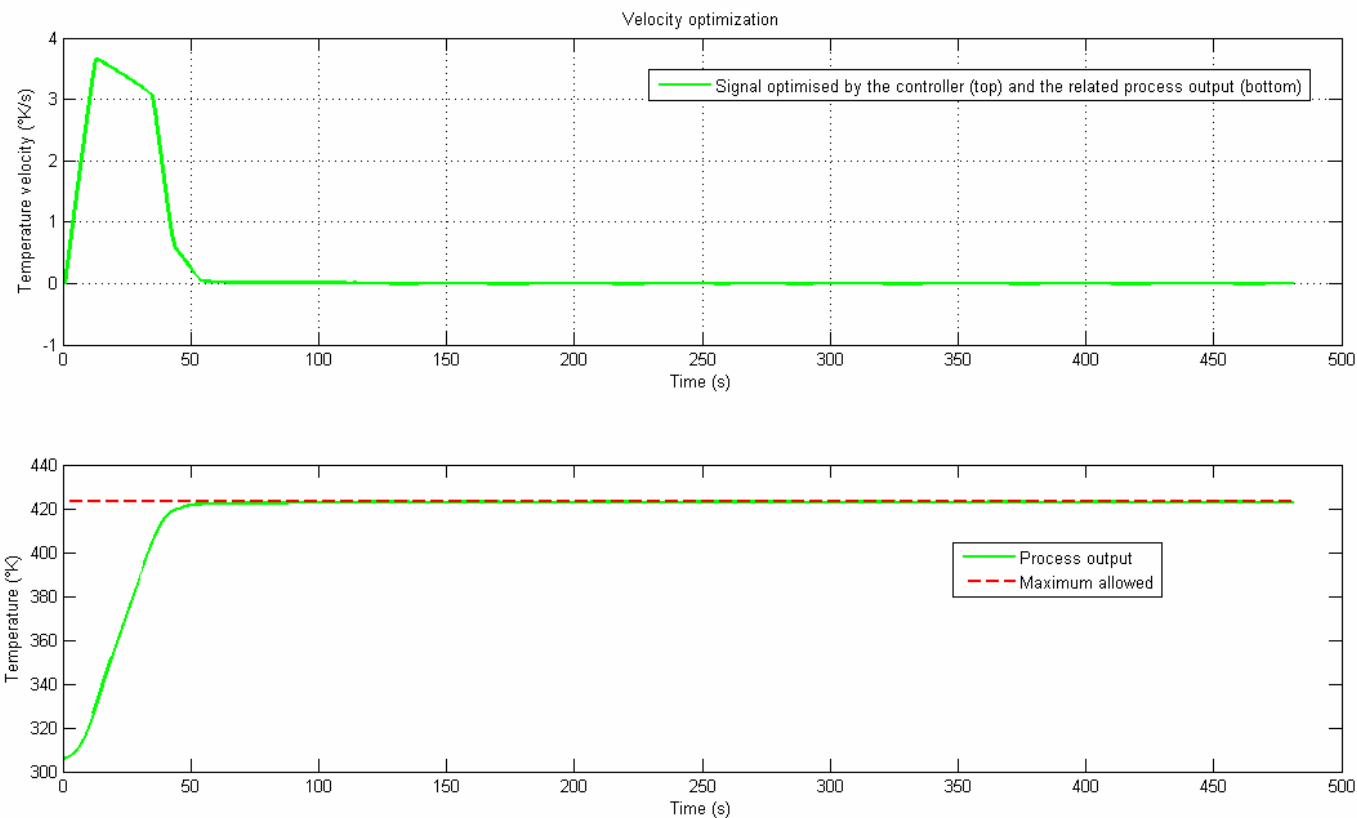
**MPC@CB ©: flexibility/ease for a quick use in control !**

# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

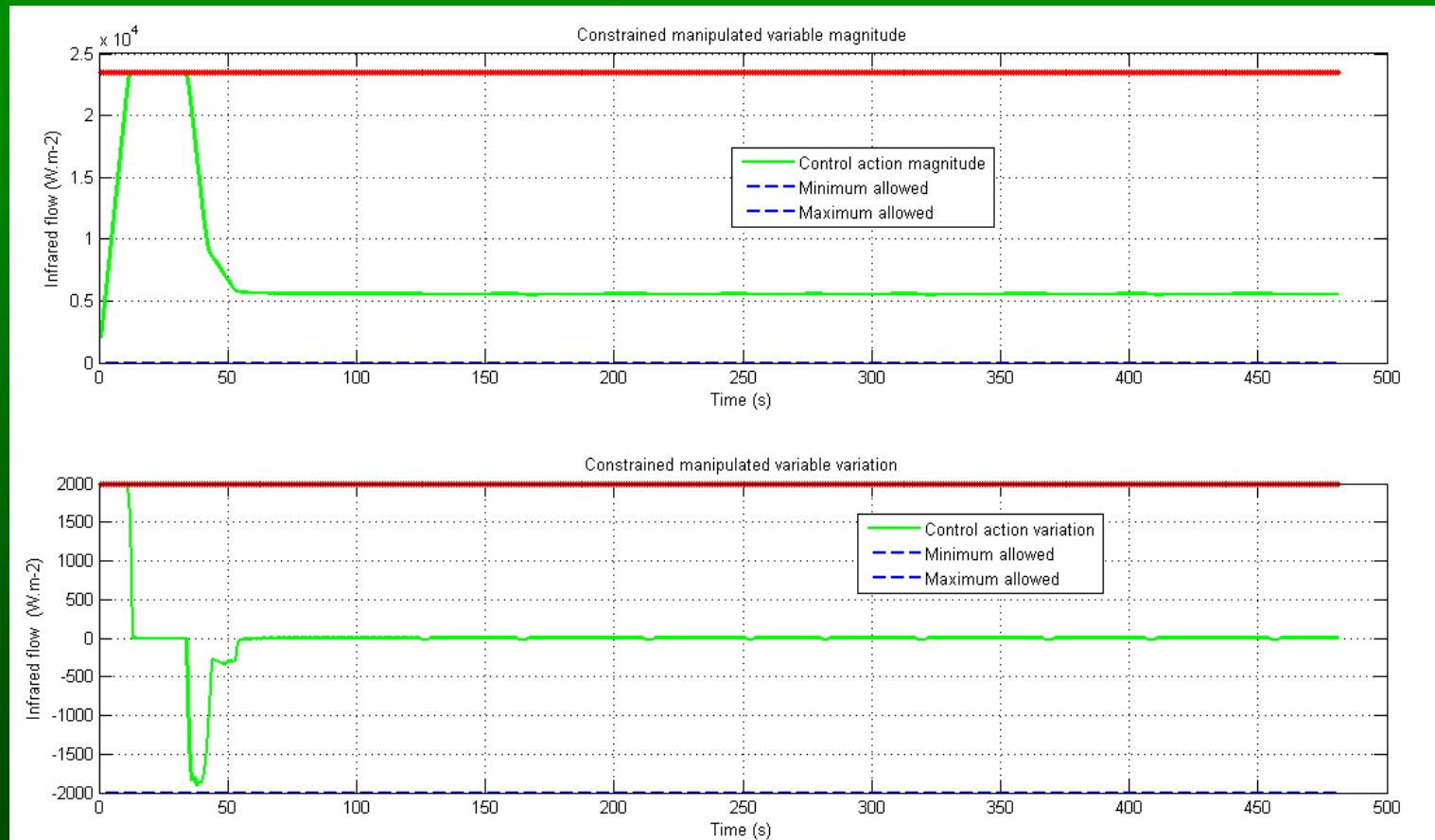
# Simulation results

Minimization of the curing time, with magnitude+velocity constraints on  $u(t)$  + output constraint  $y_p(t)$  : horizon=6, sampling time = 1s



# Simulation results

Minimization of the curing time, with magnitude+velocity constraints on  $u(t)$  + output constraint  $y_p(t)$  : horizon=6, sampling time = 1s

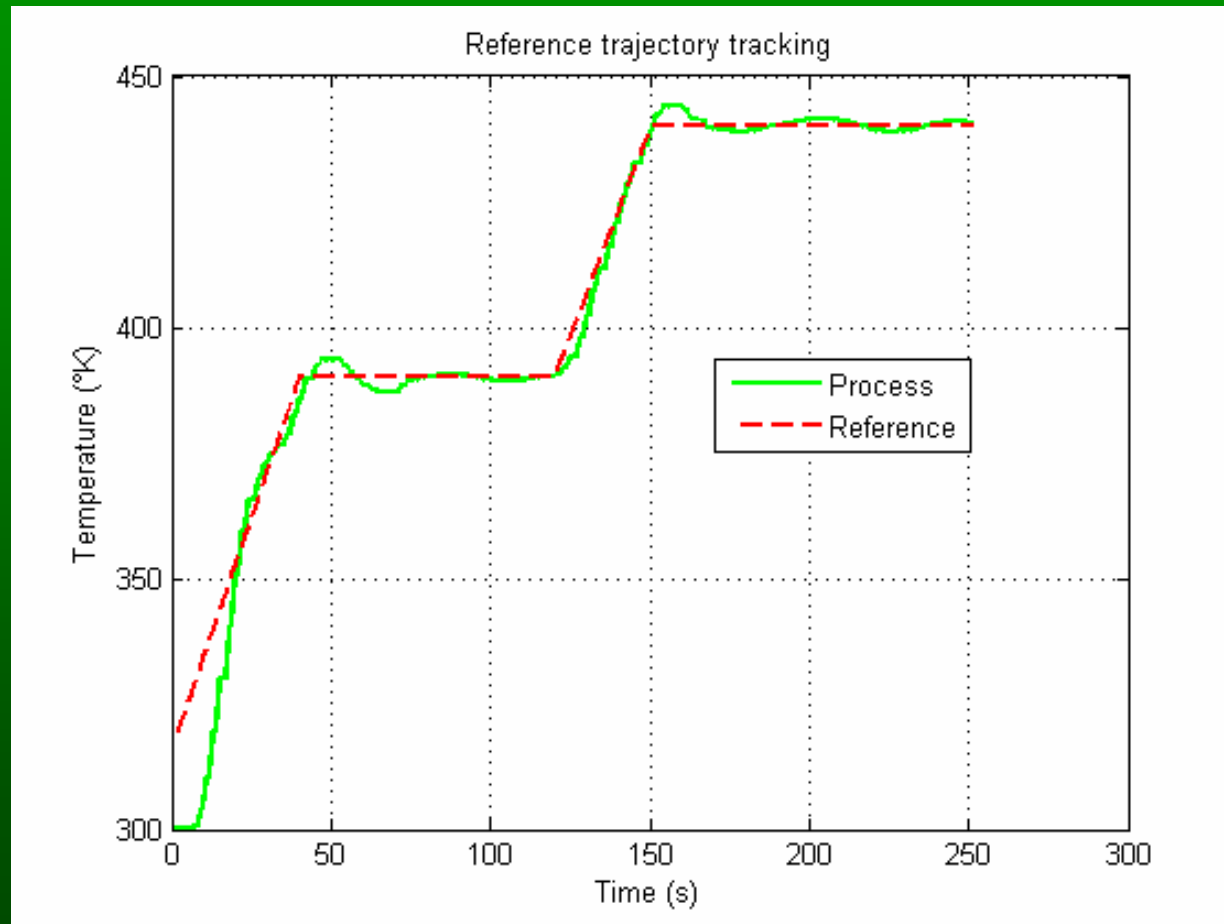


# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. Conclusions & perspectives

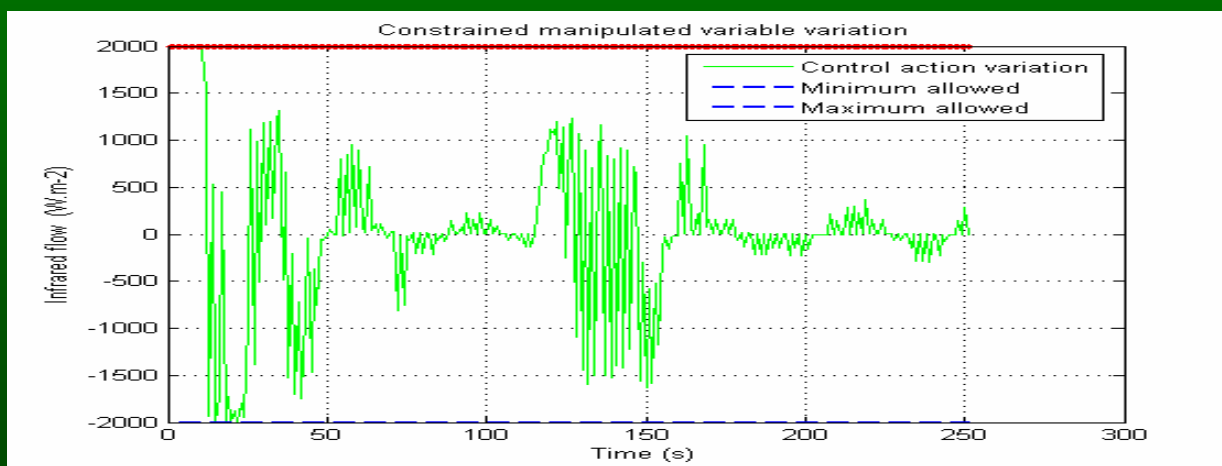
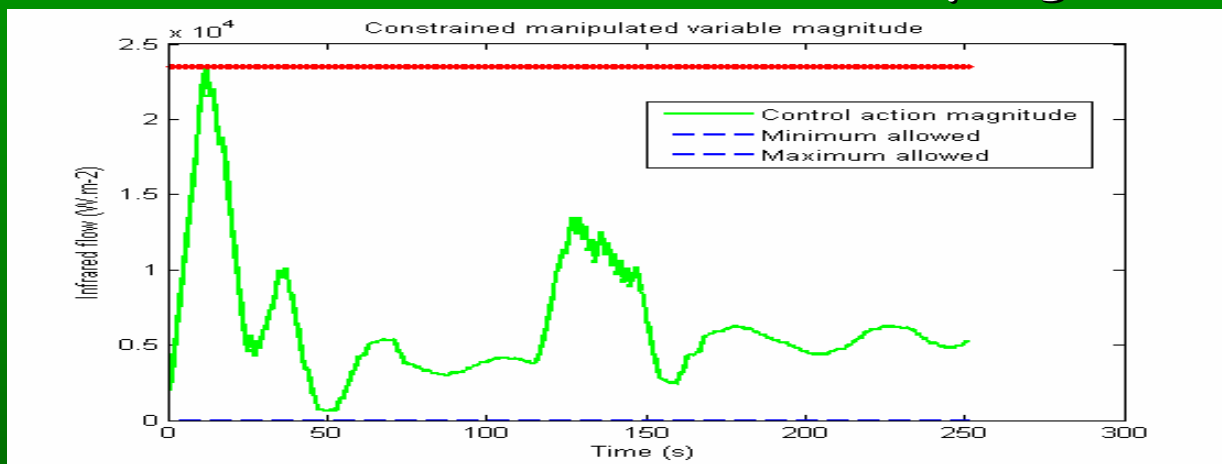
# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=6, sampling time = 1s



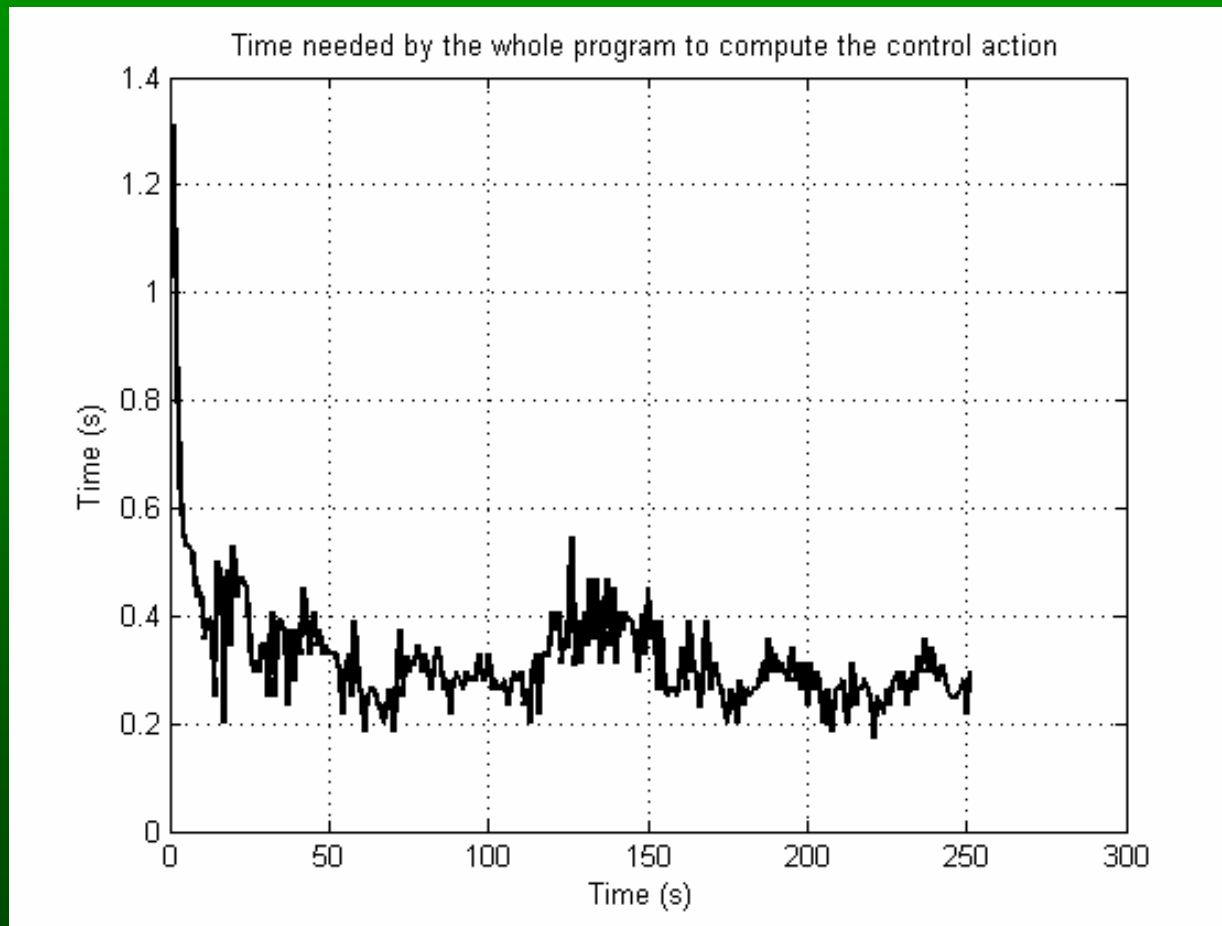
# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=6, sampling time = 1s



# Experimental results

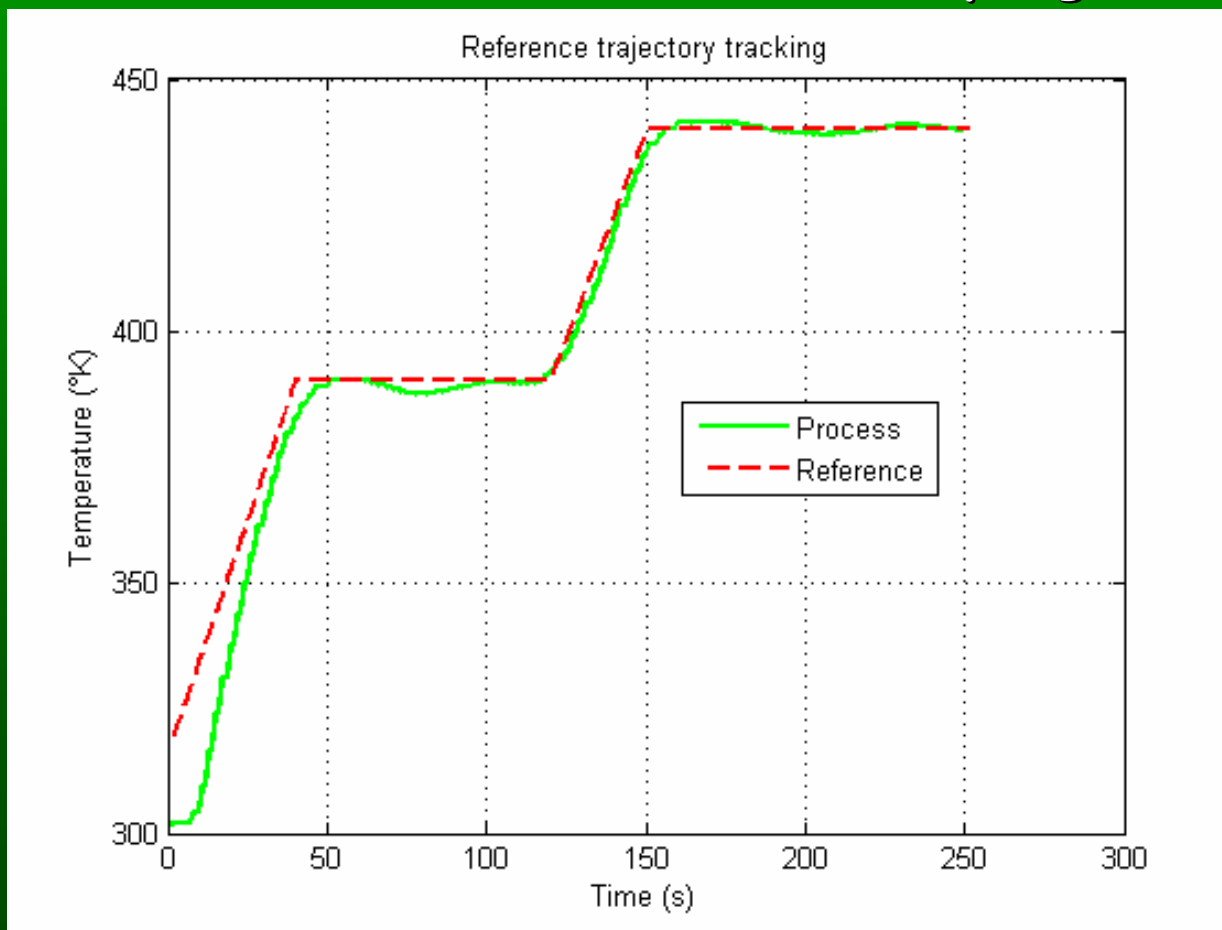
Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=6, sampling time = 1s





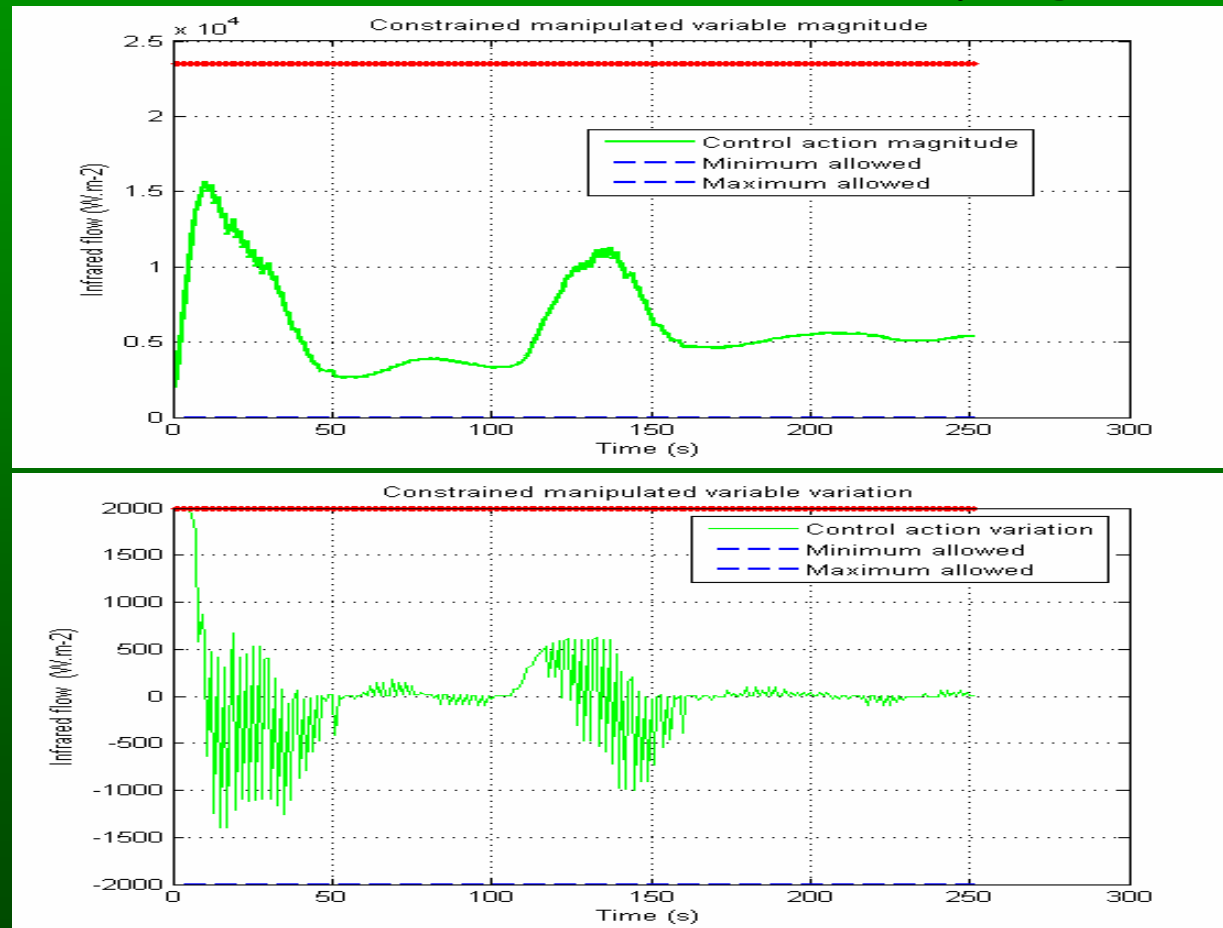
# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=14, sampling time = 1s



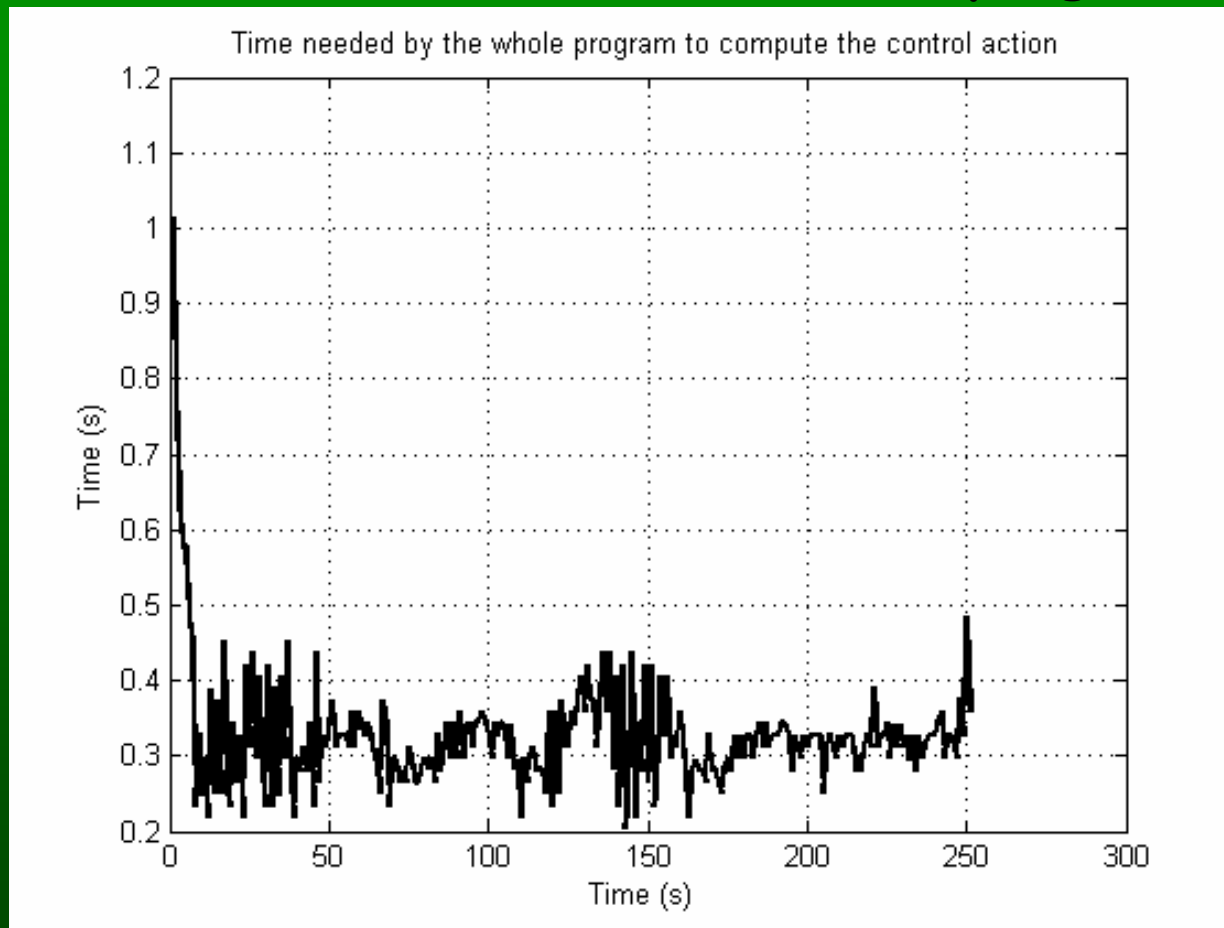
# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=14, sampling time = 1s



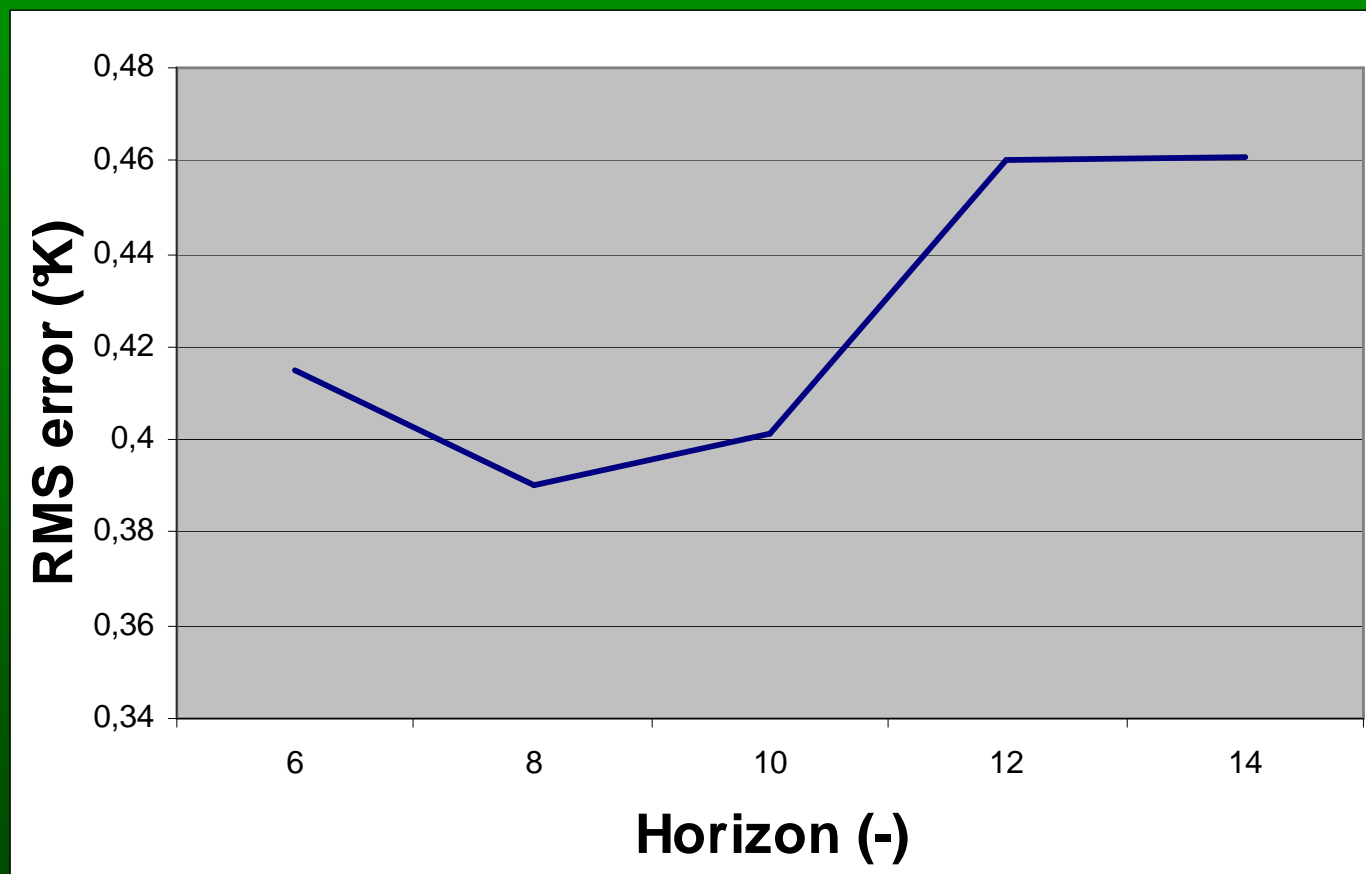
# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=14, sampling time = 1s



# Experimental results

Temperature trajectory tracking: influence of the horizon over the results



# Outline

1. Control problem statement
2. First principle PDE model
3. Model predictive control strategy
4. MPC@CB© software main features
5. Simulation results
6. Experimental results
7. **Conclusions & perspectives**

## Conclusions

- The real time control of powder curing is possible : temperature trajectory tracking
- Experimental control of PDE system by a general MPC@CB© software has been shown

## Perspectives

- Minimization of the powder curing time under constraints: an observer is under development
- MPC@CB© may be used for any process: since its development, it is currently used for control of polymer production, vial lyophilisation, pasta drying.
- To use MPC@CB©: [dufour@lagep.univ-lyon1.fr](mailto:dufour@lagep.univ-lyon1.fr)



Thank you



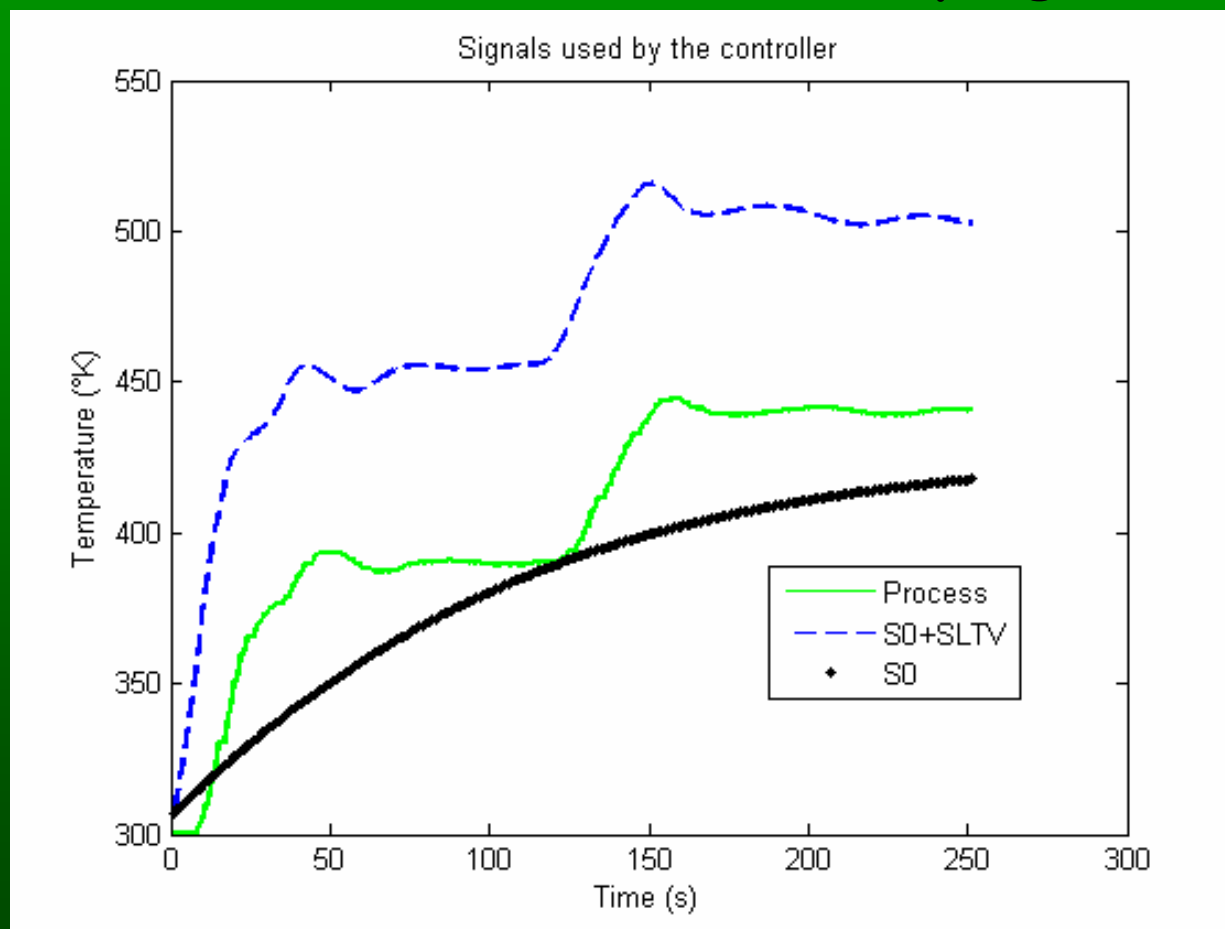
Any questions ?





# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=6, sampling time = 1s





# Experimental results

Temperature trajectory tracking, with magnitude+velocity constraints on  $u(t)$  : horizon=14, sampling time = 1s

