A closed loop identification software for dynamic systems: ODOE4OPE (Optimal Design Of Experiments for Online Parameter Estimation)

Jun QIAN[1,2], Pascal DUFOUR[1,3], Madiha NADRI[1]

[1]Laboratory of Process Control and Chemical Engineering (LAGEP), UMR5007, CNRS, University Claude Bernard Lyon 1
[2]Acsystème company (IT and Control engineering), Rennes, France
[1,2]Emails: jun.qian@acsysteme.com or qian@lagep.univ-lyon1.fr; dufour@lagep.univ-lyon1.fr; nadri@lagep.univ-lyon1.fr;
[3]Project leader and contact. Software website: http://odoe4ope.univ-lyon1.fr

Journée nationale des logiciels de modélisation et de calcul scientifique (LMCS):
07/12/2012

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.
- Combines observer design theory and an on-line predictive controller (MPC).

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.
- Combines observer design theory and an on-line predictive controller (MPC).
- Input and output constraints may be specified to keep the process in a desired operating zone.

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.
- Combines observer design theory and an on-line predictive controller (MPC).
- Input and output constraints may be specified to keep the process in a desired operating zone.
- For simulations and real applications.

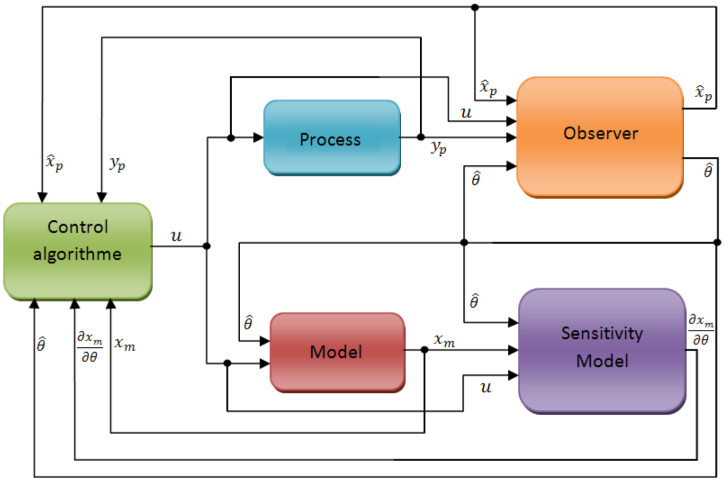
Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.
- Combines observer design theory and an on-line predictive controller (MPC).
- Input and output constraints may be specified to keep the process in a desired operating zone.
- For simulations and real applications.
- No existing similar software available on the market.

Main features of ODOE4OPE

- Synthesizes the online design of the optimal experiment (DOE) and online closed-loop identification.
- For linear and nonlinear dynamic model based systems.
- Online optimal input design which optimizes the sensitivities of the measurements with respect to the unknown constant model parameters.
- Combines observer design theory and an on-line predictive controller (MPC).
- Input and output constraints may be specified to keep the process in a desired operating zone.
- For simulations and real applications.
- No existing similar software available on the market.

Closed loop control structure



The components

Model (linear or nonlinear)

$$(M)\left\{\begin{array}{rcl}\dot{x}(t) & = & f(x(t),\theta,u(t))\\ y(t) & = & h(x(t),\theta,u(t))\end{array}\right. \quad (1)$$

where $x \in \mathbb{R}^n$ is the state vector, $y \in \mathbb{R}^p$ is the output vector, $u \in \mathcal{U} \subset \mathbb{R}^m$ is the input vector, $\theta \in \mathbb{R}^q$ is the unknown constant parameters vector.

Observer

- system augmented with the unknown constant model parameters.
- synthesis of an observer for the system augmented: high gain observer, EKF, adaptive-gain observer, ...

The components

Model (linear or nonlinear)

$$(M)\begin{cases} \dot{x}(t) &= f(x(t),\theta,u(t)) \\ y(t) &= h(x(t),\theta,u(t)) \end{cases} \quad (1)$$

where $x \in \mathbb{R}^n$ is the state vector, $y \in \mathbb{R}^p$ is the output vector, $u \in \mathcal{U} \subset \mathbb{R}^m$ is the input vector, $\theta \in \mathbb{R}^q$ is the unknown constant parameters vector.

Observer

- system augmented with the unknown constant model parameters.
- synthesis of an observer for the system augmented: high gain observer, EKF, adaptive-gain observer, ...

Sensitivity model

$$(M_\theta)\begin{cases} \frac{\partial}{\partial t}\tilde{x}_\theta = \frac{d}{\partial t}\left(\frac{\partial x}{\partial \theta}\right) = \frac{\partial}{\partial \theta}\left(\frac{\partial x}{\partial t}\right) = \frac{\partial f}{\partial \theta} + \frac{\partial f}{\partial x}\frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} = \tilde{y}_\theta = \frac{\partial h}{\partial \theta} + \frac{\partial h}{\partial x}\frac{\partial x}{\partial \theta} \end{cases} \quad (2)$$

The components

Model (linear or nonlinear)

$$(M)\begin{cases} \dot{x}(t) = f(x(t), \theta, u(t)) \\ y(t) = h(x(t), \theta, u(t)) \end{cases} \quad (1)$$

where $x \in \mathbb{R}^n$ is the state vector, $y \in \mathbb{R}^p$ is the output vector, $u \in \mathcal{U} \subset \mathbb{R}^m$ is the input vector, $\theta \in \mathbb{R}^q$ is the unknown constant parameters vector.

Observer

- system augmented with the unknown constant model parameters.
- synthesis of an observer for the system augmented: high gain observer, EKF, adaptive-gain observer, ...

Sensitivity model

$$(M_\theta)\begin{cases} \frac{\partial}{\partial t}\tilde{x}_\theta = \frac{d}{\partial t}\left(\frac{\partial x}{\partial \theta}\right) = \frac{\partial}{\partial \theta}\left(\frac{\partial x}{\partial t}\right) = \frac{\partial f}{\partial \theta} + \frac{\partial f}{\partial x}\frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} = \tilde{y}_\theta = \frac{\partial h}{\partial \theta} + \frac{\partial h}{\partial x}\frac{\partial x}{\partial \theta} \end{cases} \quad (2)$$

Optimal control law design

- Sensitivity matrix:

$$Z_k = \begin{bmatrix} \frac{\partial x_1}{\partial \theta_1}\Big|_k & \frac{\partial x_1}{\partial \theta_2}\Big|_k & \cdots & \frac{\partial x_1}{\partial \theta_p}\Big|_k \\ \frac{\partial x_2}{\partial \theta_1}\Big|_k & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial x_n}{\partial \theta_1}\Big|_k & \cdots & \cdots & \frac{\partial x_n}{\partial \theta_p}\Big|_k \end{bmatrix} \quad (3)$$

- Fisher Information Matrix (FIM): $M_k = Z_k^T Z_k$
- Cost function

$$\begin{cases} J = \phi(F(x_{j|k}, u_{j|k}, \theta_{k|k})), \\ \text{with } F(x_{j|k}, u_{j|k}, \theta_{k|k}) = \frac{1}{N_p} \sum_{j=k+1}^{k+N_p} M_{j|k} \\ c_i(y, x, \theta, u) < 0 \end{cases} \quad (4)$$

- Criterion: A-optimality

$$\begin{cases} u = & arg\max_{u \in [u_{min}, u_{max}]} J_A(u) \\ \text{with} & J_A(u) = trace(F) \end{cases} \quad (5)$$

Stage of development

- History: created in 2009
- Fundamentals Aspects:
  - 2006-2010: Saida Flila's PhD thesis
  - Mars 2012: PhD thesis CIFRE (J. QIAN) between Acsystème and LAGEP (UMR5007, CNRS, UCBL1)
- User interface: under Matlab, GUI under development

The nonlinear model of Bio-reactor: $X + S \rightarrow X$

The nonlinear dynamical model of the process is:

$$(\Sigma)\begin{cases} \dot{X}(t) = \dfrac{\mu_{max} S(t)}{S(t)+K} X(t) - D(t)X(t) \\ \dot{S}(t) = -\alpha \dfrac{\mu_{max} S(t)}{S(t)+K} X(t) - D(t)\left(S(t) - S_{in}\right) \\ y(t) = X(t). \end{cases} \quad (6)$$

where:

- Inputs: a scalar controllable dilution rate $D(t)$ (h^{-1}) and an substrate concentration S_{in} (g/L).
- Output: a biomass $X(t)$ (g/L)
- Unknowns constants parameters: μ_{max} and α.
- input constraints: $0\,h^{-1} \leqslant D(j) \leqslant 0.2\,h^{-1}$
- output constraint: $X(j) \leqslant 1.95\,g/L$

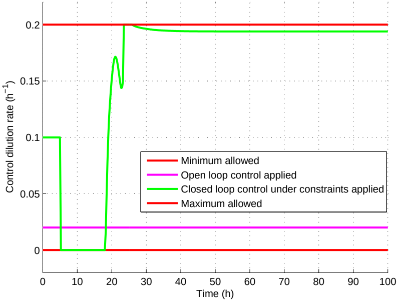
Objective: based on (Σ) online identify the unknowns parameters.

Simulation results

- Input applied: $D(t)$

Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$

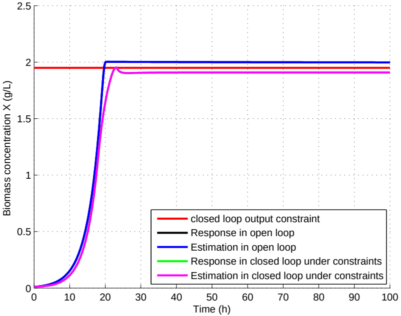


Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$

Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}

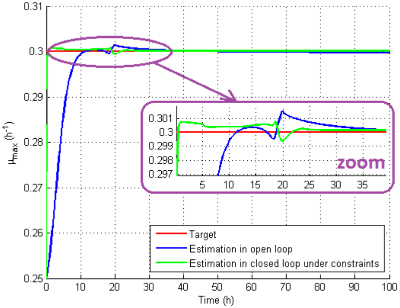


Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}

Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}
- Parameter estimation: α

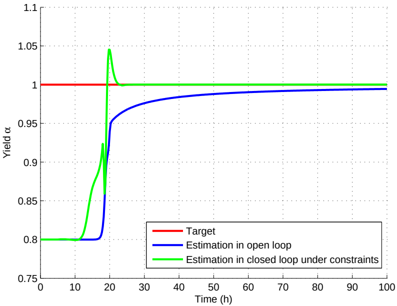


Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}
- Parameter estimation: α

Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}
- Parameter estimation: α
- Sensitivities of the model states with respect to the estimated parameters $\frac{\partial x_i}{\partial \theta_j}(t)$

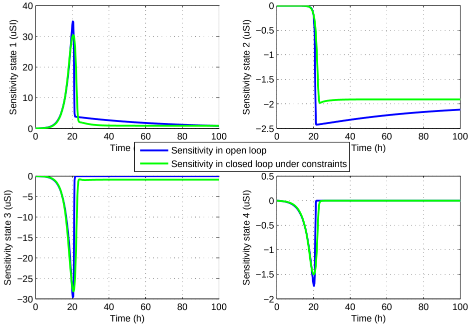


Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}
- Parameter estimation: α
- Sensitivities of the model states with respect to the estimated parameters $\frac{\partial x_i}{\partial \theta_j}(t)$

Simulation results

- Input applied: $D(t)$
- Process output: Biomass $X(t)$
- Parameter estimation: μ_{max}
- Parameter estimation: α
- Sensitivities of the model states with respect to the estimated parameters $\frac{\partial x_i}{\partial \theta_j}(t)$



Conclusion

- ODOE4OPE is able to design online the optimal experiment under constraints.
- ODOE4OPE is able to identify online model parameters.
- The combination of an observer and a predictive control in closed loop improve the speed of the parameter estimation.
- The sensitivity criteria improve the accuracy of parameter estimation and leads to an optimal control at the same time.
- The input and output constraints specify the physical limitations imposed by the system and ensure the efficiency of the DOE.
- The software may be adapted and tuned for any user defined dynamic model.

A Model Predictive Control software: MPC@CB

Jun QIAN[1,2], Pascal DUFOUR[1,3], Madiha NADRI[1]

[1]Laboratory of Process Control and Chemical Engineering (LAGEP), UMR5007, CNRS, University Claude Bernard Lyon 1
[2]Acsystème company (IT and Control engineering), Rennes, France
[1,2]Emails: jun.qian@acsysteme.com or qian@lagep.univ-lyon1.fr; dufour@lagep.univ-lyon1.fr; nadri@lagep.univ-lyon1.fr;
[3]Project leader and contact. Software website: http://MPCatCB.univ-lyon1.fr

Journée nationale des logiciels de modélisation et de calcul scientifique (LMCS): 07/12/2012

Main features of the MPC@CB software

- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.

Main features of the MPC@CB software

- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.
- Open loop or PID may also be applied by the software before using MPC@CB (to compare these different control approaches).

Main features of the MPC@CB software

- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.
- Open loop or PID may also be applied by the software before using MPC@CB (to compare these different control approaches).
- A software sensor (observer) based on model can be introduced.

Main features of the MPC@CB software

- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.
- Open loop or PID may also be applied by the software before using MPC@CB (to compare these different control approaches).
- A software sensor (observer) based on model can be introduced.
- Industrial application domains: chemistry/chemical engineering, electrical engineering, food, materials, mechanics, pharmaceuticals,...

Main features of the MPC@CB software

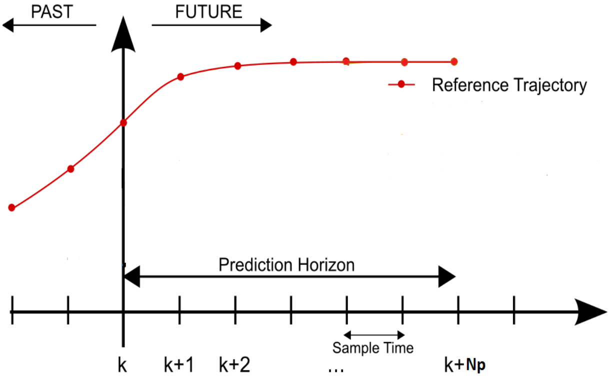
- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.
- Open loop or PID may also be applied by the software before using MPC@CB (to compare these different control approaches).
- A software sensor (observer) based on model can be introduced.
- Industrial application domains: chemistry/chemical engineering, electrical engineering, food, materials, mechanics, pharmaceuticals,...
- For simulation (training) or real time application.

Main features of the MPC@CB software

- A control software for dynamic systems based on any kind of model: SISO or MIMO (S=single, M=multiple), linear or nonlinear, time variant or time invariant, with ordinary differential equations (ODE) and/or partial differential equations (PDE).
- A model predictive control (MPC) strategy for solving an optimal control problem (trajectory tracking, processing time minimization, any user defined criteria ...) with input constraints and with (or without) output constraints.
- Open loop or PID may also be applied by the software before using MPC@CB (to compare these different control approaches).
- A software sensor (observer) based on model can be introduced.
- Industrial application domains: chemistry/chemical engineering, electrical engineering, food, materials, mechanics, pharmaceuticals,...
- For simulation (training) or real time application.

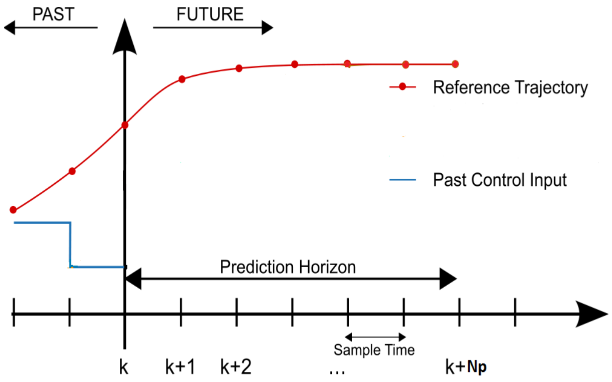
MPC: general framework

- MPC scheme



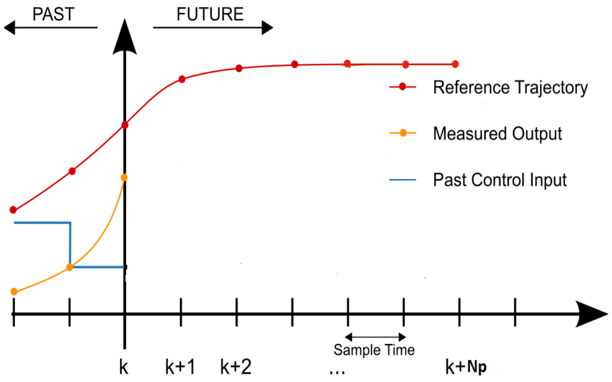
MPC: general framework

- MPC scheme



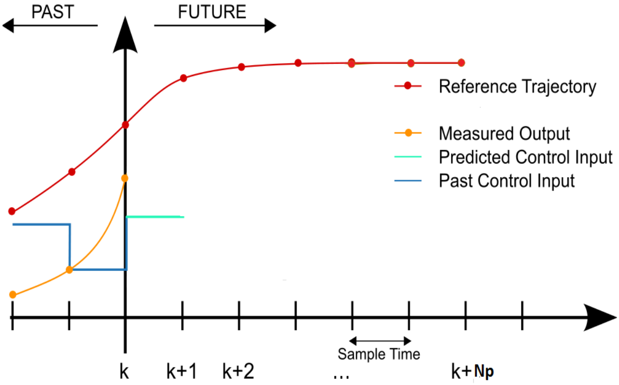
MPC: general framework

- MPC scheme



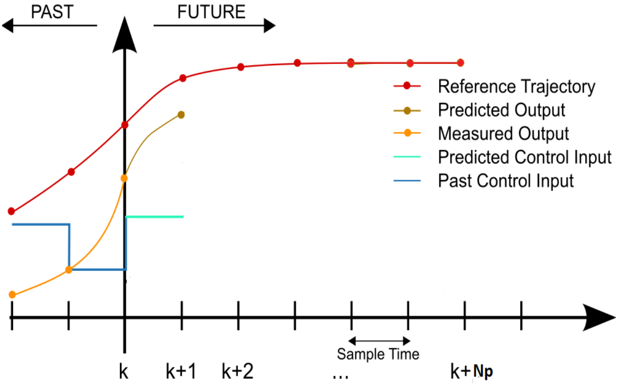
MPC: general framework

- MPC scheme



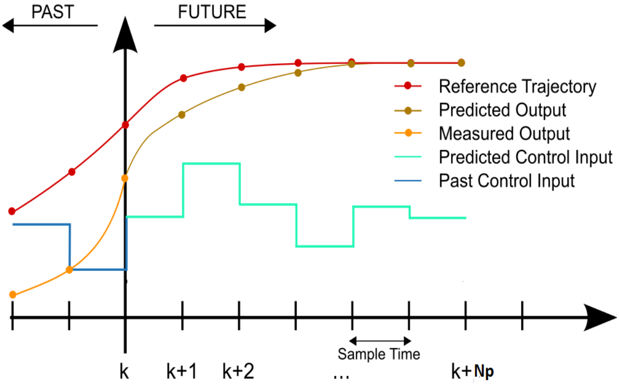
MPC: general framework

- MPC scheme

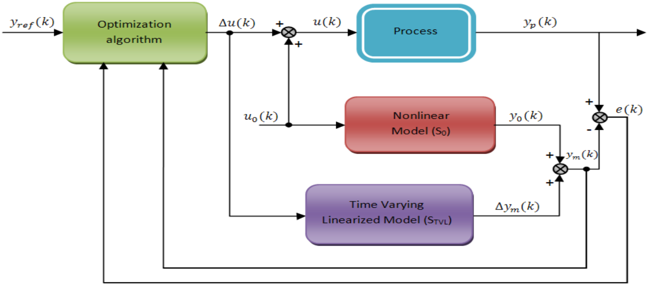


MPC: general framework

- MPC scheme



Linearized IMC-MPC structure



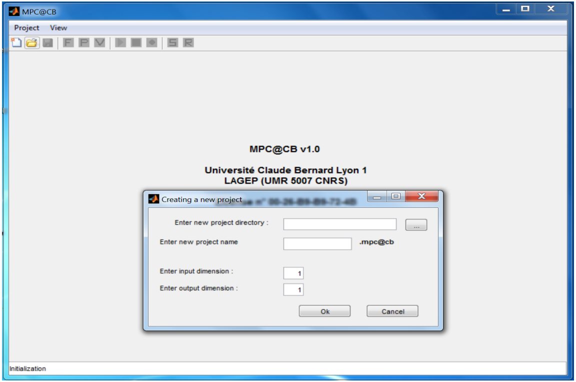
MPC@CB is based on an internal model control (IMC) structure where:

- Nonlinear model S_0 is solved off-line.
- Time-varying linearized model S_{TVL} (obtained from S_0) is solved on-line.
- Off-line open loop results are used on-line for the correct closed loop optimal constrained tuning of the control action.

Formulation of the optimization problem solved in a MPC approach

$$
\left\{
\begin{array}{l}
\min_p J_{tot} = J(p) + J_{ext}(p) \\
J(p) = \sum_{j=k+1}^{k+N_p} g(y_{ref}(j), \Delta y_m(j), \Delta u(p(j)), e(k)) \\
J_{ext}(p) = \sum_{j=k+1}^{k+N_p} (\sum_{i=1}^{N_c} w_i max^2(0, c_i(y_{ref}(j), \Delta y_m(j), \Delta u(p(j)), e(k)))) \\
p\text{: unconstrained input parameter} \\
c_i\text{: output constraints for the controlled variables} \\
\text{Input constraints handling: hyperbolic transformation} \\
\text{Ouput constraints handling: exterior penalty method} \\
\text{Control algorithm: Levenberg-Maquardt's algorithm}
\end{array}
\right.
\tag{1}
$$

Stage of development for MPC@CB



- History: created in 2007, under Matlab, with GUI
- Today: a standalone application without Matlab is available

The nonlinear model of CSTR

A continuous stirred tank reactor (CSTR): $A \rightarrow B$ is described as follows:

$$(\Sigma)\begin{cases} \dot{c}_A(t) = \frac{q}{V}(c_A^f - c_A(t)) - k_0 exp\left(-\left(\frac{E}{R}\right)/T(t)\right) c_A(t) \\ \dot{T}(t) = \frac{q}{V}(T_f - T(t)) + \frac{\Delta H}{\rho C_p} k_0 exp\left(-\left(\frac{E}{R}\right)/T(t)\right) c_A + \frac{UA}{\rho V C_p}(T_c - T(t)) \\ y(t) = c_A(t) \end{cases} \tag{2}$$

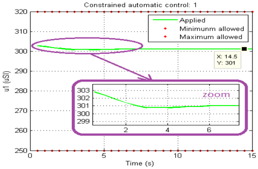
Where:

- Input: the controllable temperature of cooling jacket $T_c(t)$ (K).
- Output: the concentration of A $c_A(t)$ (mol/m^3)
- Input constraints: $250K < T_c < 320K$.
- Output constraint $y > y_{min} = 0.87$.

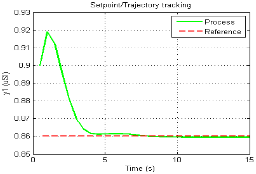
Objective: use MPC@CB (with or without the output constraint) for the set-point tracking of a reference value 0.86.

Simulation results

- without the output constraint
  - optimal input applied

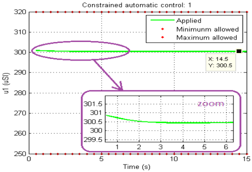


  - setpoint trajectory tracking

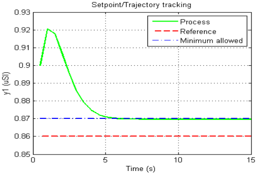


Conclusion: setpoint regulation, OK!

- with the output constraint: $y > y_{min} = 0.87$
  - optimal input applied



  - trajectory tracking



Conclusion: constrained setpoint regulation, OK!

Conclusion

- MPC@CB is easily tunable for any new dynamic process.
- The specified user defined constrained control objectives are well achieved by the online closed loop control with MPC@CB.
- With the off-line and on-line IMC-MPC structure, the on-line computational time of optimization is decreased by MPC@CB.
- More case studies are discussed on the website.
- MPC@CB is available: short time evaluation, commercial licence or embedded in a complete turnkey solution for the customer.

Contact for the softwares

Pascal DUFOUR

Associate professor
University Claude Bernard Lyon 1, Campus de la Doua
Ecole CPE, bât 308G, étage 3
LAGEP UMR 5007, Bureau G322
3 rue Victor Grignard
69100 Villeurbanne, France
Tel: +33 4 72 43 18 78
dufour@lagep.univ-lyon1.fr
http://www.tinyurl.com/dufourpascal

ODOE4OPE

Email: odoe4ope@univ-lyon1.fr
Website: http://odoe4ope.univ-lyon1.fr

MPC@CB

Email: MPCatCB@univ-lyon1.fr
Website: http://mpcatcb.univ-lyon1.fr/

References

ODOE4OPE

[1] S. Flila, P. Dufour and H. Hammouri, *Identification optimale en boucle fermée pour les systèmes non linéaires*, 6th IEEE Conference International Francophone Automatic (CIFA), Nancy, France, 2012.

[2] S. Flila, P. Dufour, H. Hammouri and Nadri M., *A combined closed loop optimal design of experiments and online identification control approach*,in 29th IEEE C 2010, Beijing, China.

MPCatCB

[1] N. Daraoui, P. Dufour, H. Hammouri, A. Hottot, *Model predictive control during the primary drying stage of lyophilisation*, Control Engineering Practice, 2010, 18(5), pp. 483-494.

[2] I. Bombard, B. Da Silva, P. Dufour and P. Laurent, *Experimental predictive control of the infrared cure of a powder coating: a non-linear distributed parameter model based approach*, Chemical Engineering Science Journal, 2010, 65(2), pp. 962-975.

[3] P. Dufour, Y. Touré, D. Blanc, P. Laurent,*On nonlinear distributed parameter model predictive control strategy: On-line calculation time reduction and application to an experimental drying process*, Computers and Chemical Engineering, 2003, 27(11), pp.1533-1542.

Annex A: ODOE4OPE

- Simulation condition
  - Parameters in the model of the bio-reactor

Parameter	Symbol	Value
The maximal specific rate of the biomass (h^{-1})	μ_{max}	0.3
The yield (-)	α	1
The constant of the saturation (g/L)	K	0.05
The substrate concentration in the feed (g/L)	S_{in}	2

  - Initial conditions and parameters value for the simulation

Initial conditions and Parameters	Symbol	Value (Unit)
Target values of parameters	$[\theta_1\ \theta_2]_p$	[0.3 1]
Initial estimates of parameters	$[\hat{\theta}_1(0)\ \hat{\theta}_2(0)]$	[0.25 0.8]
Initial values of model states	$[x_{m1}(0)\ x_{m2}(0)]$	[0.01 2]
Initial estimates of states	$[\hat{x}_1(0)\ \hat{x}_2(0)]$	[0.01 1.5]
Initial estimate of covariance	$P(0)$	$50 \times I$
Time of the simulation	T_{fin}	100 h
Sampling period	T_s	0.25 h
Prediction horizon	N_p	8

Annex A: ODOE4OPE

- Observer for bio-reactor
  - System augmented:

$$(M)\begin{cases} \dot{x}_1(t) = \dfrac{\theta_1 x_2(t)}{x_2(t)+a_1} x_1(t) - u(t)x_1(t) \\ \dot{x}_2(t) = -\theta_2 \dfrac{\theta_1 x_2(t)}{x_2(t)+a_1} x_1(t) - u(t)\left(x_2(t) - a_2\right) \\ \dot{\theta}_1 = 0 \\ \dot{\theta}_2 = 0 \\ y(t_k) = x_1(t_k), \end{cases} \tag{3}$$

where $t_k - t_{k-1}$ is the sampling time measurements.

Annex A: ODOE4OPE

- Extended Kalman Filter (EKF)

Model	$\dot{x}(t) = f(x(t), u(t)) + w(t), w(t) \in N(0, Q(t))$
	$y_k = h(x_k) + v_k, v_k \in N(0, R_k)$ where $x_k = x(t_k)$
Initialize	$\hat{x}_{0\|0} = E[x(t_0)],\ P_{0\|0} = Var[x(t_0)]$
Predict	$\begin{cases} \dot{\hat{x}}(t) = f(\hat{x}(t), u(t)) \\ \dot{P}(t) = F(t)P(t) + P(t)F(t)^T + Q(t) \end{cases}$
	with $\begin{cases} \hat{x}(t_{k-1}) = \hat{x}_{k-1\|k-1} \\ P(t_{k-1}) = P_{k-1\|k-1} \end{cases}$
	$\Rightarrow \begin{cases} \hat{x}_{k\|k-1} = \hat{x}(t_k) \\ P_{k\|k-1} = P(t_k) \end{cases}$
Update	$K_k = P_{k\|k-1}H_k^T(H_kP_{k\|k-1}H_k^T + R_k)^{-1}$
	$\hat{x}_{k\|k} = \hat{x}_{k\|k-1} + K_k(y_k - h(\hat{x}_{k\|k-1}))$
	$P_{k\|k} = (I - K_kH_k)P_{k\|k-1}$
	where $F(t) = \frac{\partial f}{\partial x}\Big\|_{\hat{x}(t),u(t)},\ H(t) = \frac{\partial h}{\partial x}\Big\|_{\hat{x}_{k\|k-1}}$

Annex A: ODOE4OPE

- Sensitivity model of the bio-reactor

$$\begin{cases} \dot{x}_{1\theta 1}(t) = \dfrac{x_2(t)x_1(t)}{x_2(t)+a_1} - u(t)x_{1\theta 1}(t) + \hat{\theta}_1(t)\dfrac{x_2(t)x_{1\theta 1}(t)(x_2(t)+a_1)+a_1x_1(t)x_{2\theta 1}(t)}{(x_2(t)+a_1)^2} \\ \dot{x}_{1\theta 2}(t) = -u(t)x_{1\theta 2} + \hat{\theta}_1(t)\dfrac{x_2(t)x_{1\theta 2}(t)(x_2(t)+a_1)+a_1x_1(t)x_{2\theta 2}(t)}{(x_2(t)+a_1)^2} \\ \dot{x}_{2\theta 1}(t) = -\dfrac{\hat{\theta}_2(t)x_2(t)x_1(t)}{x_2(t)+a_1} - u(t)x_{2\theta 1}(t) - \hat{\theta}_1(t)\hat{\theta}_2(t)\dfrac{x_2(t)x_{1\theta 1}(t)(x_2(t)+a_1)+a_1x_1(t)x_{2\theta 1}(t)}{(x_2(t)+a_1)^2} \\ \dot{x}_{2\theta 2}(t) = -\dfrac{\hat{\theta}_1(t)x_2(t)x_1(t)}{x_2(t)+a_1} - u(t)x_{2\theta 2}(t) - \hat{\theta}_1(t)\hat{\theta}_2(t)\dfrac{x_2(t)x_{1\theta 2}(t)(x_2(t)+a_1)+a_1x_1(t)x_{2\theta 2}(t)}{(x_2(t)+a_1)^2} \\ x_{1\theta 1}(0) = x_{1\theta 2}(0) = x_{2\theta 1}(0) = x_{2\theta 2}(0) = 0 \end{cases} \tag{4}$$

where $x_{i\theta_j} = \frac{\partial x_i}{\partial \theta_j}$.

Annex B: MPC@CB

- The constrained optimization problem based on IMC-MPC structure is described as followed:

$$\left\{\begin{array}{l} \min_{\tilde{u}} J(\tilde{u}) = \sum_{j \in \mathcal{J}_1^{N_p}} g(y_{ref}(j), \Delta y_m(j), \Delta u(j-1), e(k)) \\ \Delta \tilde{u} = [\cdots \Delta u(j) \cdots]^T \quad \forall j \in \mathcal{J}_0^{N_c-1} \\ \Delta u(j) = \Delta u(k+N_c-1) \quad \forall j \in \mathcal{J}_{N_c}^{N_p-1} \\ u_{min} - u_0(j) \leq \Delta u(j) \leq u_{min} - u_0(j) \quad \forall j \in \mathcal{J}_0^{N_p-1} \\ \Delta u'_{min} \leq \Delta u(j) - \Delta u(j-1) \leq \Delta u'_{max} \quad \forall j \in \mathcal{J}_0^{N_p-1} \\ \Delta u'_{min} = \Delta u_{min} - (u_0(j) - u_0(j-1)) \quad \forall j \in \mathcal{J}_0^{N_p-1} \\ \Delta u'_{max} = \Delta u_{max} - (u_0(j) - u_0(j-1)) \quad \forall j \in \mathcal{J}_0^{N_p-1} \\ c_i(y_{ref}(j), \Delta y_m(j), \Delta u(j-1), e(k)) \leq 0 \; \forall j \in \mathcal{J}_0^{N_p}, \; \forall i \in \mathcal{I}_1^n \\ \text{and subjet to the resolution of the model } (S_{TVL}). \end{array}\right. \tag{5}$$

Annex B: MPC@CB

- Input constraints handling: hyperbolic transformation:

$$\begin{cases} u(j) = f(p(j)) = f_{moy} + f_{amp} tanh(\frac{p(j) - f_{moy}}{f - amp}) \quad \forall j \in \mathcal{J}_0^{N_c - 1} \\ p(j) \in \mathbb{R} \; \forall j \in \mathcal{J}_0^{N_c - 1} \text{ (unconstrained input parameter)} \\ f_{moy} = \frac{f_{max} + f_{min}}{2} \\ f_{amp} = \frac{f_{max} - f_{min}}{2} \\ f_{min} = \max(u_{min}, u(j-1) + \Delta u_{min}) \quad \forall j \in \mathcal{J}_0^{N_c - 1} \\ f_{max} = \max(u_{max}, u(j-1) + \Delta u_{max}) \quad \forall j \in \mathcal{J}_0^{N_c - 1} \end{cases} \tag{6}$$

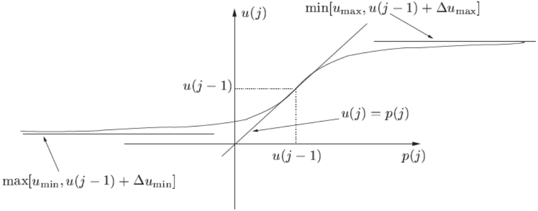


Fig. Mapping from unconstrained variable p into constrained variable u

Annex B: MPC@CB

- Control algorithm: Levenberg-Maquardt

$$\Delta\tilde{p}^{n+1} = \Delta\tilde{p}^{n} - (\bigtriangledown^2 J_{tot}^{n} + \lambda I)^{-1} \bigtriangledown J_{tot}^{n} \tag{7}$$

where the argument $\Delta\tilde{p}$ is determined at each sample instant k by this iteration procedure, $\bigtriangledown^2 J_{tot}^{n}$ and $\bigtriangledown J_{tot}^{n}$ are the criteria gradient and criteria hessain with respect to $\Delta\tilde{p}^{n}$ at the iteration n.