Real-Time Optimization of Chemical Processes

Dominique Bonvin, Grégory François and Gene Bunin
Laboratoire d'Automatique
EPFL, Lausanne

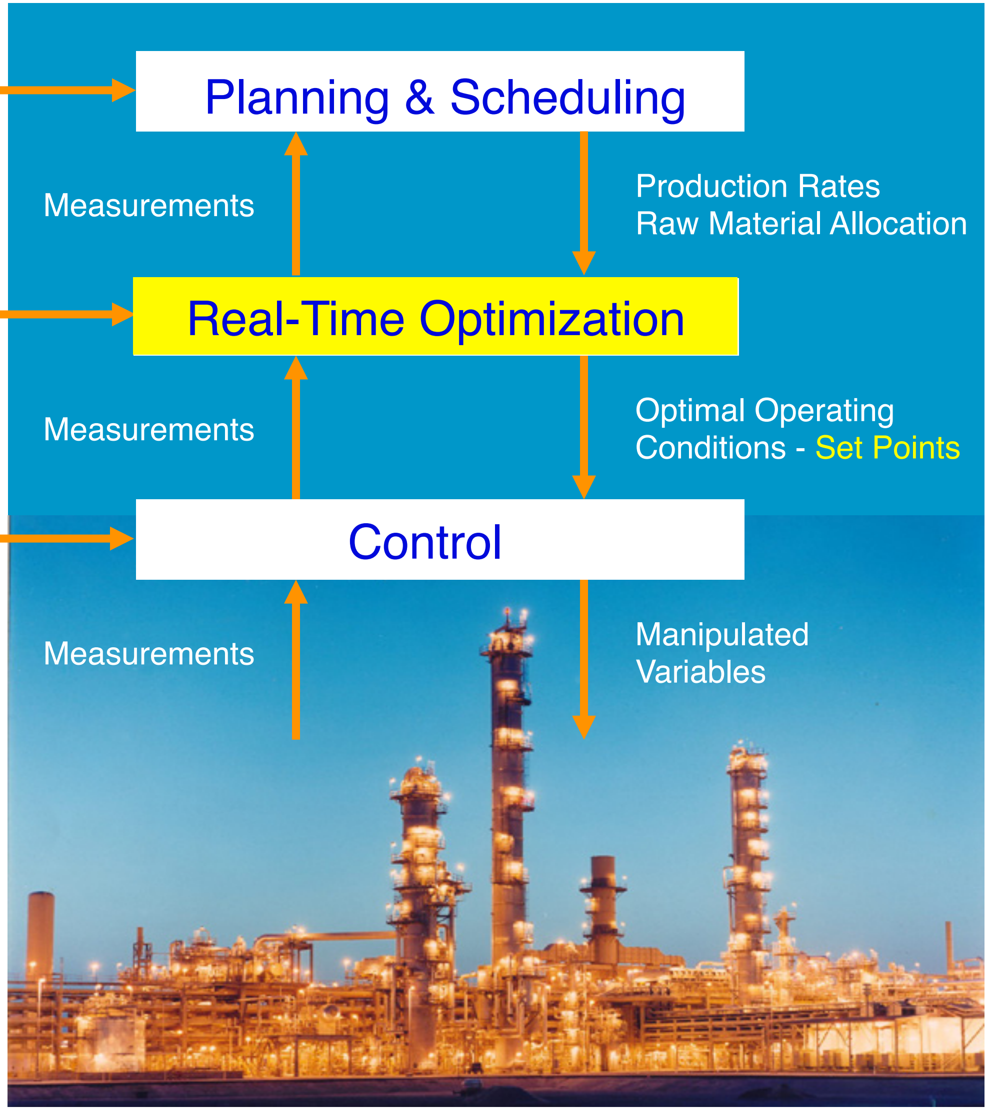
SFGP, Lyon 2013

Real-Time Optimization of a Continuous Plant

Laboratoire d'Automatique

	Disturbances
Long term week/month	Market Fluctuations, Demand, Price
Medium term day	Catalyst Decay, Changing Raw Material Quality
Short term second/minute	Fluctuations in Pressure, Flowrates, Compositions

Decision Levels

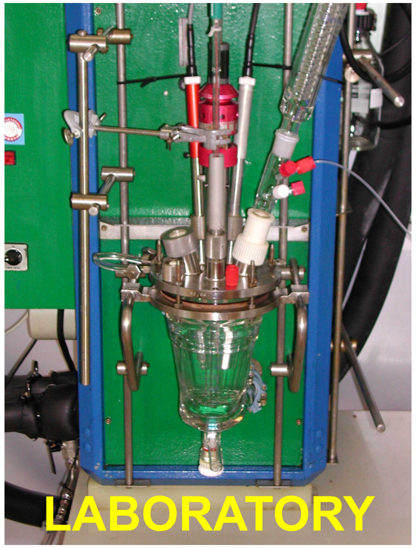


Large-scale complex processes

Changing conditions
➔ Real-time adaptation

Optimization of a Discontinous Plant

RECIPE → BATCH PLANT → PRODUCTS



Scale-up

Differences in Equipment and Scale

- mass- and heat-transfer characteristics
- surface-to-volume ratios
- operational constraints

Production Constraints

- meet product specifications
- meet safety and environmental constraints
- adhere to equipment constraints

Different conditions ➔ Run-to-run adaptation

Outline

What is real-time optimization

- Goal: Optimal plant operation
- Tool: Model-based numerical optimization, experimental optimization
- Key feature: use of real-time measurements

Real-time optimization framework

- Three approaches
- Key issues: Which measurements? How to best exploit them?
- Simulated comparison

Experimental case studies

- Fuel-cell stack
- Batch polymerization

Static Optimization Problem

Optimize the steady-state performance of a (dynamic) process while satisfying a number of operating constraints

Plant

$$\min_{\mathbf{u}} \quad \Phi_p(\mathbf{u}) := \phi_p(\mathbf{u}, \mathbf{y}_p)$$

$$\text{s. t.} \quad \mathbf{G}_p(\mathbf{u}) := \mathbf{g}_p(\mathbf{u}, \mathbf{y}_p) \le \mathbf{0}$$

Optimal plant operation

Inputs $\mathbf{u}$?

(set points)

Plant Outputs $\mathbf{y_p}$

Model-based Optimization

$$\mathbf{F}(\mathbf{u}, \mathbf{y}, \boldsymbol{\theta}) = \mathbf{0}$$

$$\min_{\mathbf{u}} \quad \Phi(\mathbf{u}) := \phi(\mathbf{u}, \mathbf{y})$$

$$\text{s. t.} \quad \mathbf{G}(\mathbf{u}) := \mathbf{g}(\mathbf{u}, \mathbf{y}) \le \mathbf{0}$$

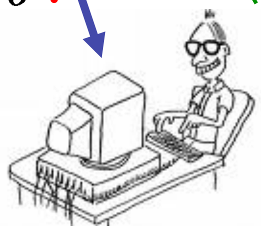
NLP

Model-based numerical optimization

Model Parameters $\boldsymbol{\theta}$?

Inputs $\mathbf{u}$?

(set points)



Predicted Outputs $\mathbf{y}$

Dynamic Optimization Problem

Optimize the dynamic performance of a (dynamic) process while satisfying a number of operating constraints

Plant

$$\min_{\mathbf{u}[0,t_f]} \Phi := \phi\left(\mathbf{x}_p(t_f)\right)$$

$$\text{s.t.} \quad \mathbf{S}(\mathbf{x}_p,\mathbf{u}) \le \mathbf{0}$$

$$\mathbf{T}\left(\mathbf{x}_p(t_f)\right) \le \mathbf{0}$$

Optimal plant operation

$\mathbf{u}(t)$

$\mathbf{x}_p(t_f)$

Model-based Optimization

$$\min_{\mathbf{u}[0,t_f]} \quad \Phi := \phi\left(\mathbf{x}(t_f),\boldsymbol{\theta}\right)$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \quad \mathbf{x}(0) = \mathbf{x}_0$$

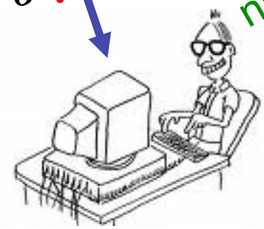
$$\mathbf{S}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \le \mathbf{0}$$

$$\mathbf{T}\left(\mathbf{x}(t_f),\boldsymbol{\theta}\right) \le \mathbf{0}$$

Model-based numerical optimization

Model Parameters $\boldsymbol{\theta}$?

Inputs $\mathbf{u}(t)$?



Predicted States $\mathbf{x}(t)$

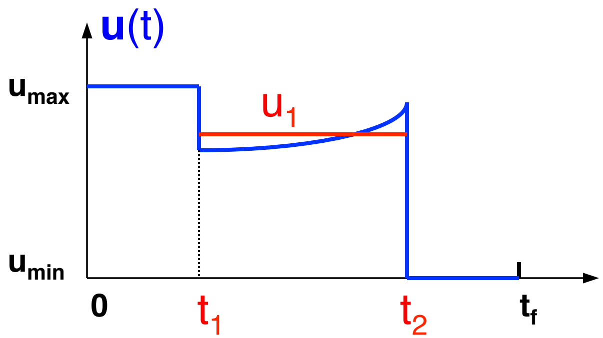
Run-to-Run Optimization of a Batch Plant

Batch plant with
finite terminal time

$$\min_{\mathbf{u}[0,t_f]} \quad \Phi := \phi\big(\mathbf{x}(t_f),\boldsymbol{\theta}\big)$$
$$\text{s. t.} \quad \dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \quad \mathbf{x}(0) = \mathbf{x}_0$$
$$\mathbf{S}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \le \mathbf{0}$$
$$\mathbf{T}\big(\mathbf{x}(t_f),\boldsymbol{\theta}\big) \le \mathbf{0}$$

Input Parameterization

$$\mathrm{u}[0,t_f] = \mathrm{U}(\pi)$$



Batch plant
viewed as a static map

$$\min_{\pi} \quad \Phi(\pi,\boldsymbol{\theta})$$
$$\text{s. t.} \quad \mathbf{G}(\pi,\boldsymbol{\theta}) \le \mathbf{0}$$

NLP

Outline

What is real-time optimization

- Goal: Optimal plant operation
- Tool: Model-based numerical optimization, experimental optimization
- Key feature: use of real-time measurements

Real-time optimization framework

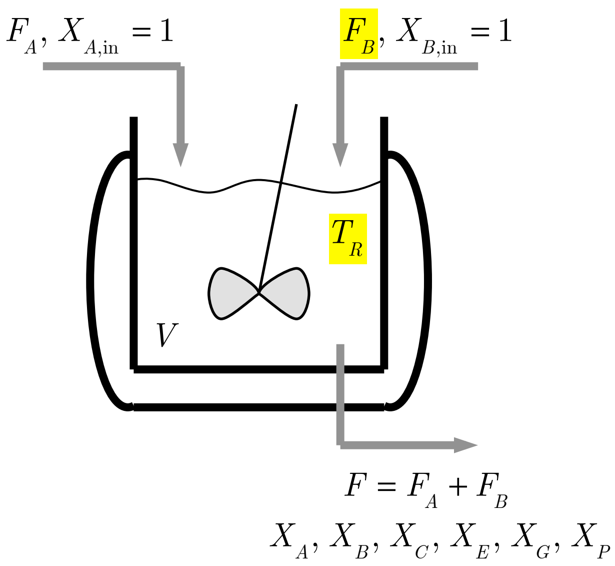
- Three approaches
- Key issues: Which measurements? How to best exploit them?
- Simulated comparison

Experimental case studies

- Fuel-cell stack
- Batch polymerization

Example of Plant-Model Mismatch

Williams-Otto reactor



3-reaction system

A + B → C
B + C → P + E
C + P → G

2-reaction model

A + 2B $\xrightarrow{k_1}$ P + E

A + B + P $\xrightarrow{k_2}$ G

Objective: maximize operating profit

Model

- 4^{th}-order model
- 2 inputs
- 2 adjustable parameters (k_{10}, k_{20})

Three RTO Approaches
How to best exploit the measurements?

Laboratoire d'Automatique

Optimization in the presence of Uncertainty

No Measurement:
Robust Optimization

Measurements:
Adaptive Optimization

input update: $\delta\mathbf{u}$

parameter update: $\delta\boldsymbol{\theta}$

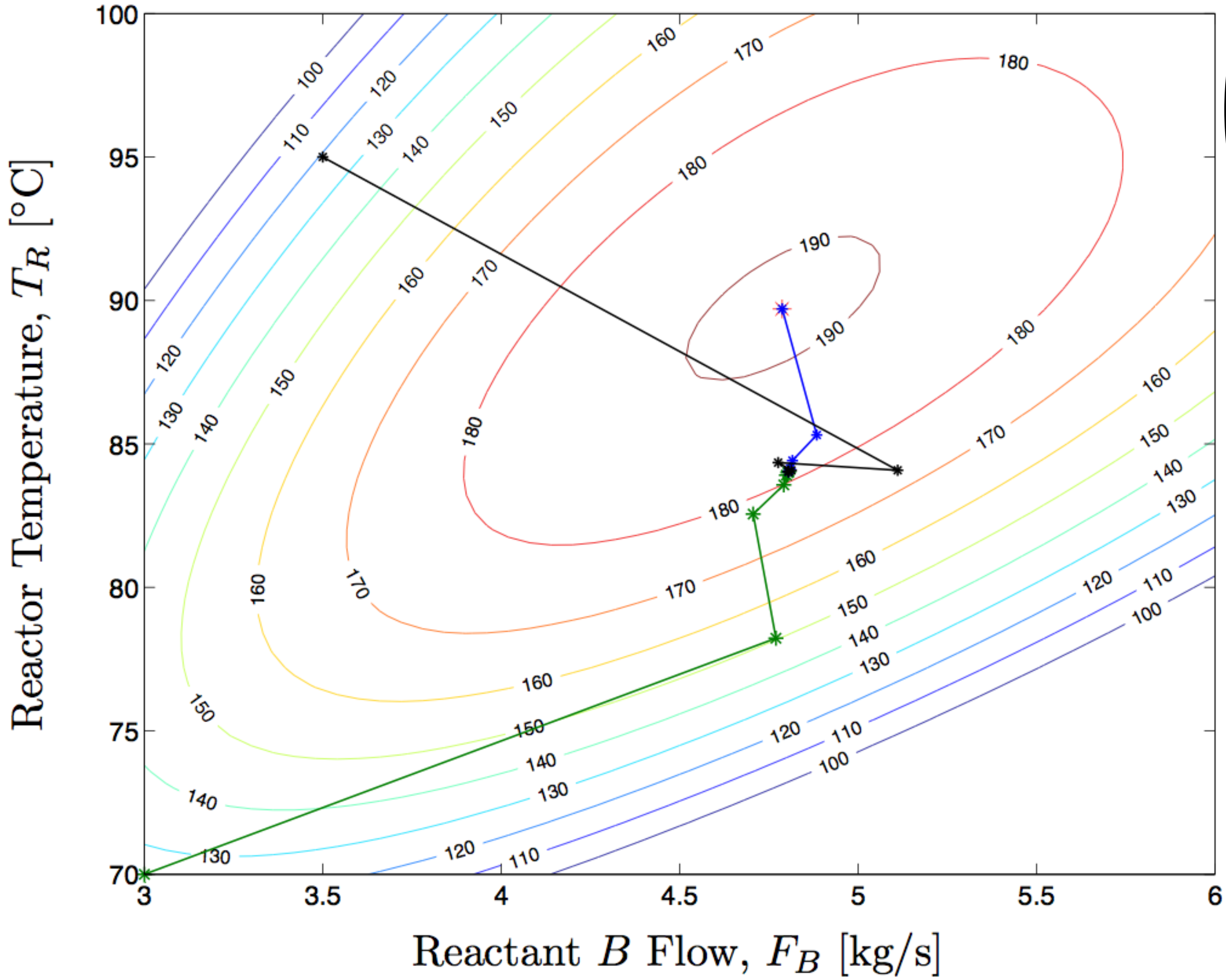
$$\mathbf{u}^* \in \arg\min_{\mathbf{u}} \ \phi(\mathbf{u},\mathbf{y}) \quad \text{s.t.} \quad \mathbf{F}(\mathbf{u},\mathbf{y},\boldsymbol{\theta}) = \mathbf{0}, \quad \mathbf{g}(\mathbf{u},\mathbf{y}) \le \mathbf{0}$$

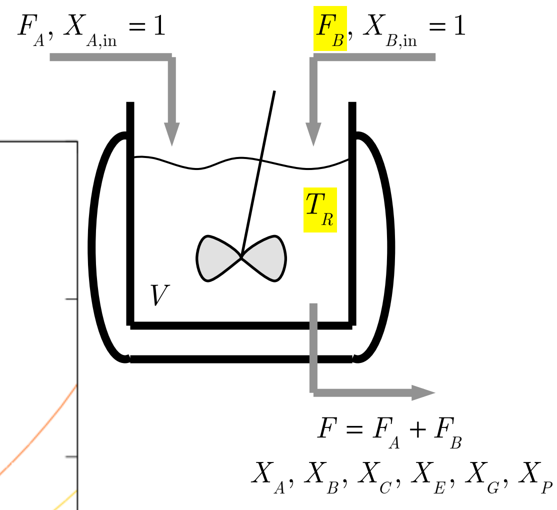
cost & constraint update: $\delta\mathbf{g}, \delta\phi$

Adaptation of Model Parameters	Adaptation of Cost & Constraints	Adaptation of Inputs
- two-step approach (repeated identification and optimization)	*- bias update* *- constraint update* *- gradient correction* *- modifier adaptation*	*- tracking active constraints* *- NCO tracking* *- extremum-seeking control* *- self-optimizing control*

1. Adaptation of Model Parameters

Two-step approach





Williams-Otto Reactor

- 4^{th}-order model
- 2 inputs
- 2 adjustable par.

Does not converge to plant optimum

Two-step approach

Parameter Estimation Problem

$$\theta_k^* \in \arg\min_{\theta} \quad J_k^{\mathrm{id}}$$

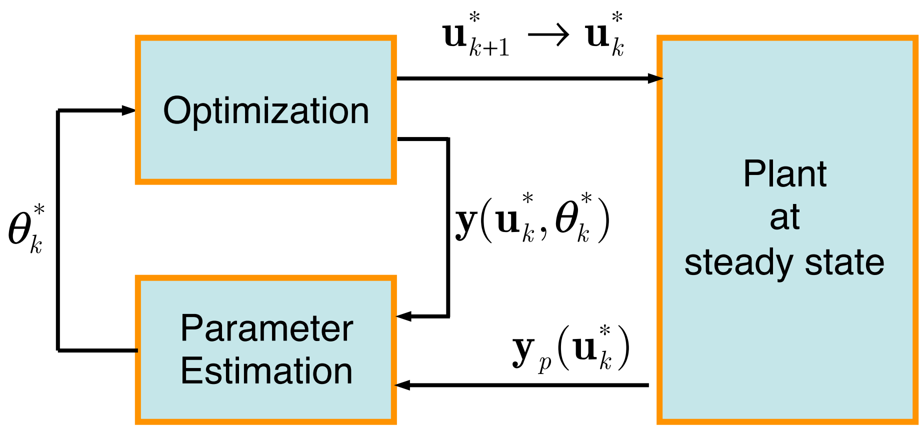
$$J_k^{\mathrm{id}} = \left[\mathbf{y}_p(\mathbf{u}_k^*) - \mathbf{y}(\mathbf{u}_k^*, \theta)\right]^{\mathrm{T}} \mathrm{Q} \left[\mathbf{y}_p(\mathbf{u}_k^*) - \mathbf{y}(\mathbf{u}_k^*, \theta)\right]$$

Optimization Problem

$$\mathbf{u}_{k+1}^* \in \arg\min_{\mathbf{u}} \quad \phi\left(\mathbf{u}, \mathbf{y}(\mathbf{u}, \theta_k^*)\right)$$

$$\text{s.t.} \quad \mathbf{g}\left(\mathbf{u}, \mathbf{y}(\mathbf{u}, \theta_k^*)\right) \leq \mathbf{0}$$

$$\mathbf{u}^{\mathrm{L}} \leq \mathbf{u} \leq \mathbf{u}^{\mathrm{U}}$$

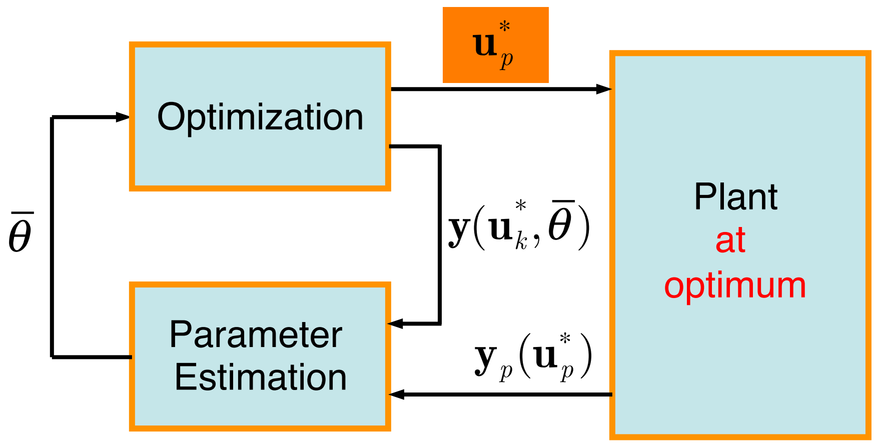


Current Industrial Practice
for tracking the changing optimum
in the presence of disturbances

T.E. Marlin, A.N. Hrymak. Real-Time Operations Optimization of Continuous Processes, *AIChE Symposium Series - CPC-V,* **93**, 156-164, 1997

Model Adequacy for Two-Step Approach

A process model is said to be adequate for use in an RTO scheme if it is capable of producing a fixed point for that RTO scheme at the plant optimum



$\bar{\theta}$ converged value

Model-adequacy conditions

SOSC / Par. Est.:

$$\frac{\partial J^{\mathrm{id}}}{\partial \theta}\Big(\mathbf{y}_p(\mathbf{u}_p^*), \mathbf{y}(\mathbf{u}_p^*, \bar{\theta})\Big) = \mathbf{0},$$

$$\frac{\partial^2 J^{\mathrm{id}}}{\partial \theta^2}\Big(\mathbf{y}_p(\mathbf{u}_p^*), \mathbf{y}(\mathbf{u}_p^*, \bar{\theta})\Big) > 0,$$

Opt.:

$$G_i(\mathbf{u}_p^*, \bar{\theta}) = 0, \quad i \in A(\mathbf{u}_p^*)$$

$$G_i(\mathbf{u}_p^*, \bar{\theta}) < 0, \quad i \notin A(\mathbf{u}_p^*)$$

$$\nabla_r \Phi(\mathbf{u}_p^*, \bar{\theta}) = \mathbf{0},$$

$$\nabla_r^2 \Phi(\mathbf{u}_p^*, \bar{\theta}) > 0$$

J.F. Forbes, T.E. Marlin. Design Cost: A Systematic Approach to Technology Selection for Model-Based Real-Time Optimization Systems. *Comp. Chem. Eng.*, **20**(6/7), 717-734, 1996

2. Adaptation of Cost & Constraints

Input-Affine Correction to the Model

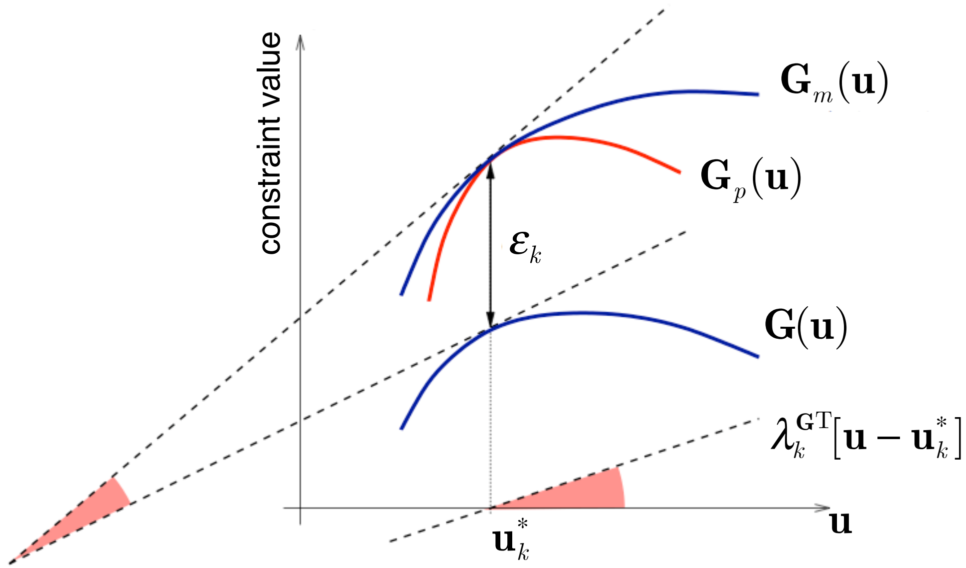
Modified Optimization Problem

$$\mathbf{u}_{k+1}^* \in \arg\min_{\mathbf{u}} \quad \Phi_m(\mathbf{u}) := \Phi(\mathbf{u}) + \lambda_k^{\Phi \mathrm{T}}[\mathbf{u} - \mathbf{u}_k^*]$$

$$\text{s.t.} \quad \mathbf{G}_m(\mathbf{u}) := \mathbf{G}(\mathbf{u}) + \varepsilon_k + \lambda_k^{\mathrm{GT}}[\mathbf{u} - \mathbf{u}_k^*] \leq \mathbf{0}$$

$$\mathbf{u}^{\mathrm{L}} \leq \mathbf{u} \leq \mathbf{u}^{\mathrm{U}}$$

Affine corrections of cost and constraint functions



Force the modified problem to satisfy the optimality conditions of the **plant**

P.D. Roberts and T.W. Williams, On an Algorithm for Combined System Optimization and Parameter Estimation, *Automatica*, **17**(1), 199–209, 1981

Input-Affine Correction to the Model

Laboratoire d'Automatique

Modified Optimization Problem

$$\mathbf{u}_{k+1}^* \in \arg\min_{\mathbf{u}} \quad \Phi_m(\mathbf{u}) := \Phi(\mathbf{u}) + \lambda_k^{\Phi\mathrm{T}}[\mathbf{u} - \mathbf{u}_k^*]$$

$$\text{s.t.} \quad \mathbf{G}_m(\mathbf{u}) := \mathbf{G}(\mathbf{u}) + \varepsilon_k + \lambda_k^{\mathbf{G}\mathrm{T}}[\mathbf{u} - \mathbf{u}_k^*] \leq \mathbf{0}$$

$$\mathbf{u}^{\mathrm{L}} \leq \mathbf{u} \leq \mathbf{u}^{\mathrm{U}}$$

- KKT Elements: $\mathcal{C}^{\mathrm{T}} = \left(G_1, \cdots, G_{n_g}, \frac{\partial G_1}{\partial \mathbf{u}}, \cdots, \frac{\partial G_{n_g}}{\partial \mathbf{u}}, \frac{\partial \Phi}{\partial \mathbf{u}} \right) \in \mathbb{R}^{n_K}$ $\quad n_K = n_g + n_u(n_g + 1)$
- KKT Modifiers: $\Lambda^{\mathrm{T}} = \left(\varepsilon_1, \cdots, \varepsilon_{n_g}, \lambda^{G_1\mathrm{T}}, \cdots, \lambda^{G_{n_g}\mathrm{T}}, \lambda^{\Phi\mathrm{T}} \right) \in \mathbb{R}^{n_K}$

Modifier Adaptation (without filter)

$$\Lambda_k = \mathrm{C}_p(\mathbf{u}_k^*) - \mathrm{C}(\mathbf{u}_k^*)$$

Requires evaluation of KKT elements of plant

Modifier Adaptation (with filter)

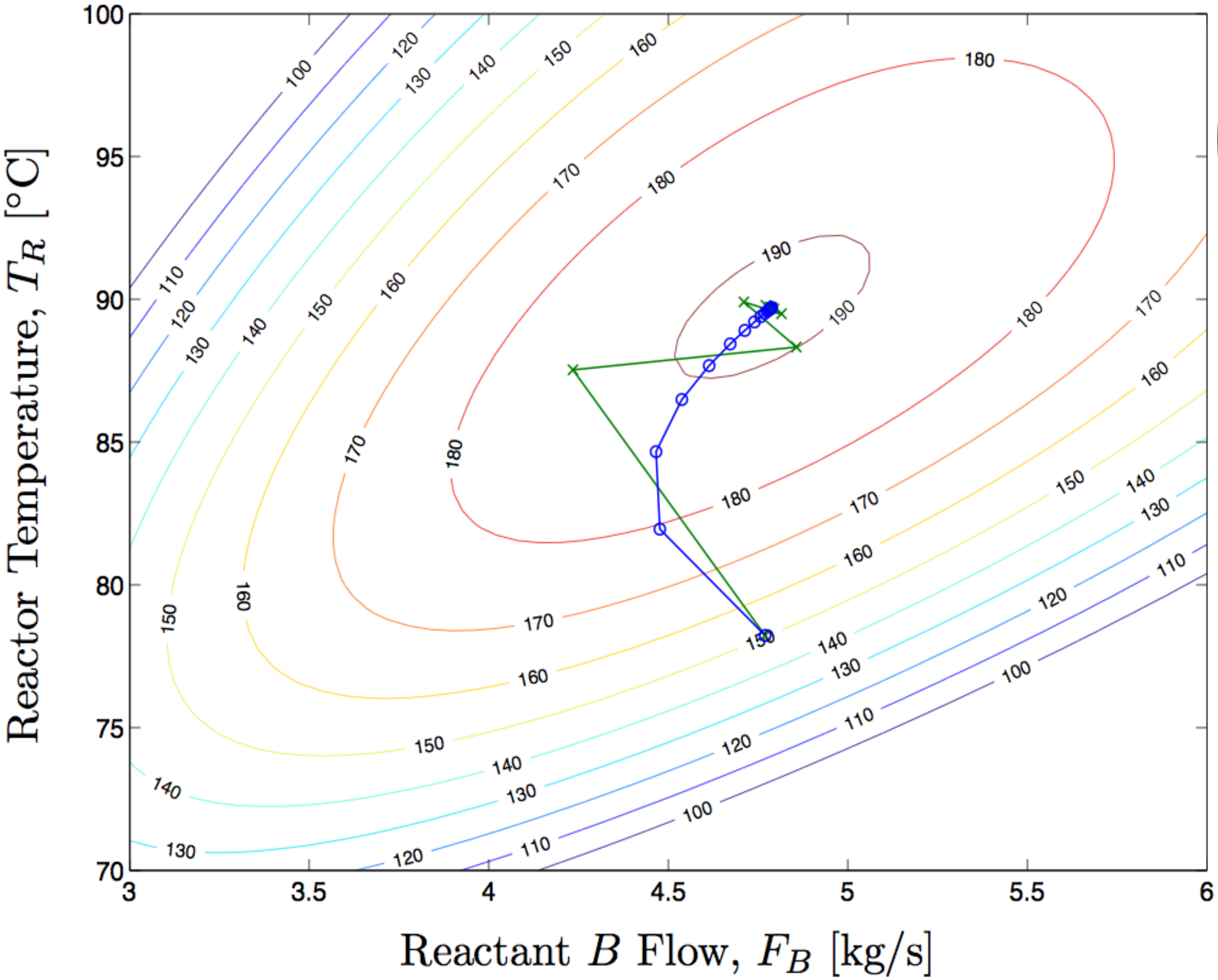
$$\Lambda_k = (\mathbf{I} - \mathrm{K})\,\Lambda_{k-1} + \mathrm{K}\left[\mathrm{C}_p(\mathbf{u}_k^*) - \mathrm{C}(\mathbf{u}_k^*)\right]$$

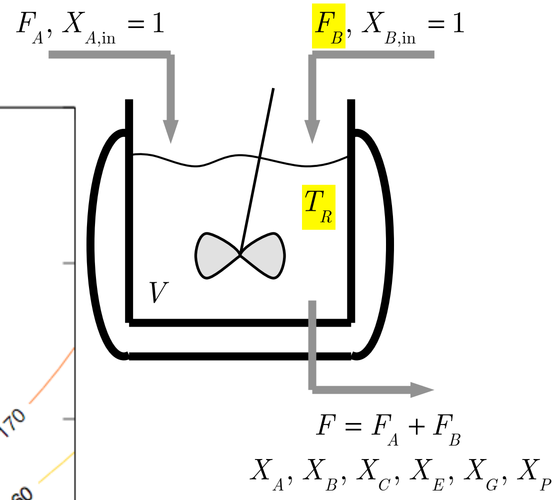
W. Gao and S. Engell, Iterative Set-point Optimization of Batch Chromatography, *Comput. Chem. Eng.*, **29**, 1401–1409, 2005

A. Marchetti, B. Chachuat and D. Bonvin, Modifier-Adaptation Methodology for Real-Time Optimization, I&EC Research, **48**(13), 6022-6033 (2009)

Example Revisited

Modifier adaptation





Williams-Otto Reactor

- 4th-order model
- 2 inputs
- 2 adjustable par.

Converges to plant optimum

A. Marchetti, PhD thesis, EPFL, Modifier-Adaptation Methodology for Real-Time Optimization, 2009

Modeling for Optimization

Laboratoire d'Automatique

Features of a "good" model

- Must be able to predict the optimality conditions of the plant:
  active constraints and (reduced) gradients
- Focuses on the optimal solution
  → "solution model" rather than "plant model"

Need to be able to estimate the plant gradients

- From cost and constraint values at previous operating points
- Must be able to use the key measurements (active constraints and gradients)

Run-to-Run Optimization
of Semi-Batch Reactor

- **Industrial Reaction System** Lonza

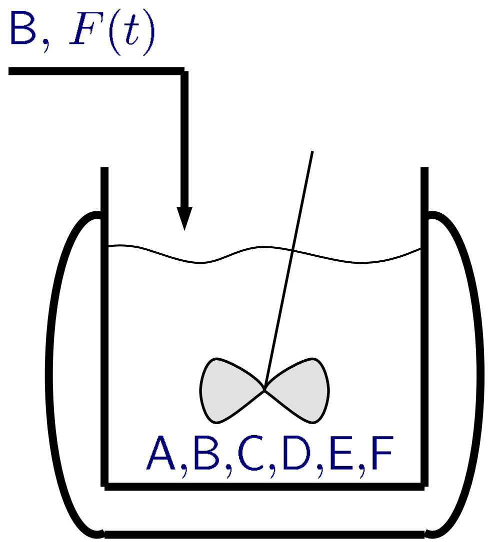
$$\mathsf{A} + \mathsf{B} \xrightarrow{k_1} \mathsf{C}$$
$$2\mathsf{B} \xrightarrow{k_2} \mathsf{D}$$

Simulated Reality

~~$\mathsf{B} \xrightarrow{k_3} \mathsf{E}$~~

~~$\mathsf{C} + \mathsf{B} \xrightarrow{k_4} \mathsf{F}$~~

Model



- Manipulated Variables: $F(t)$ (feed flow rate of B)

- Objective: Maximize $n_{\mathsf{C}}(t_{\mathrm{f}})$ (production of C)

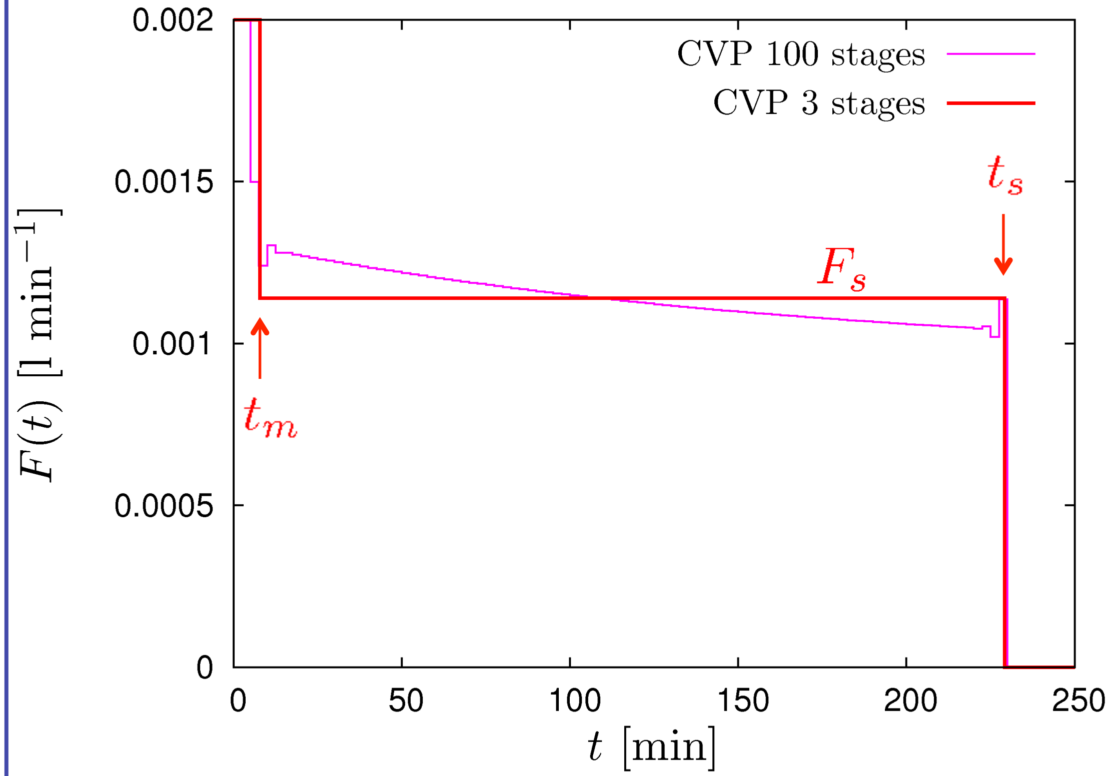
- Constraints:

Input bounds: $0 \leq F(t) \leq 0.002 \text{ l min}^{-1}$

Terminal constraints: $c_{\mathsf{B}}(t_{\mathrm{f}}) \leq 0.025 \text{ mol l}^{-1}$ (max. residual concentration)

$c_{\mathsf{D}}(t_{\mathrm{f}}) \leq 0.15 \text{ mol l}^{-1}$ (max. by-product concentration)

Nominal Optimal Input



Plant model

- 3 nonlinear balance equations
- 2 uncertain parameters k_1 and k_2
- Measurements to adjust k_1 and k_2

A solution model

- 3 arcs: F_{max}, F_s and F_{min}
- 3 adjustable parameters t_m, t_s and F_s
- Measurements to adjust t_m, t_s and F_s

◦ Optimal Solution

3 arcs, 2 active terminal constraints

$J^\star \approx 0.5081$ mol

◦ Approximate Solution

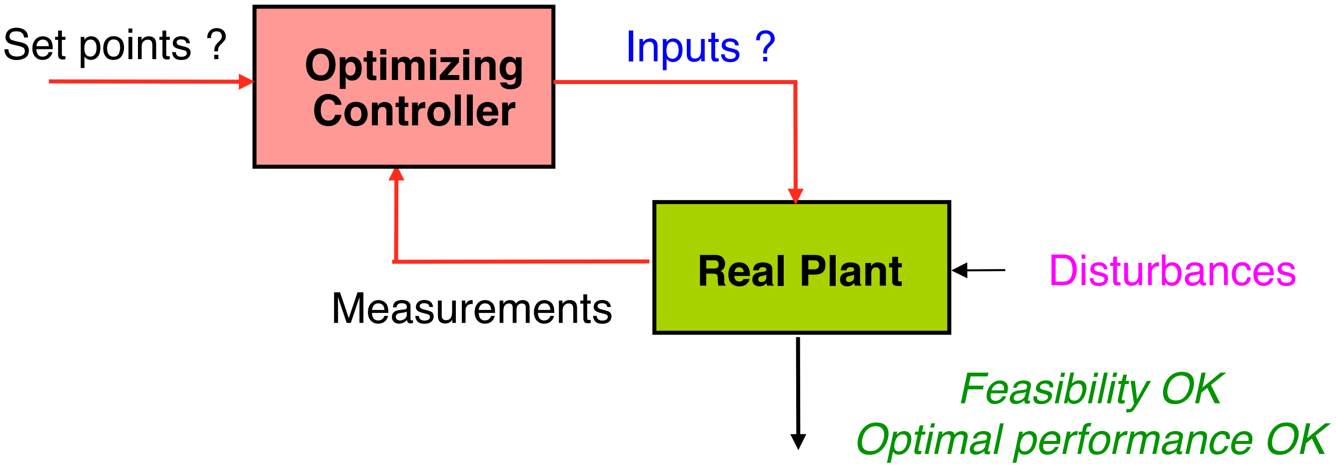
Parameterization: $\mathbf{u} = (t_m, t_s, F_s)$

$J^\star \approx 0.5079$ mol

3. Adaptation of Inputs
NCO tracking

NCO	Available degrees of freedom Input parameters
$c_B(t_f)=0.025$ $c_D(t_f)=0.15$	t_s, F_s
CV ?	**MV ?**

Solution model



Control problem

B. Srinivasan and D. Bonvin, Real-Time Optimization of Batch Processes by Tracking the Necessary Conditions of Optimality, I&EC Research, 46, 492-504 (2007).

Comparison of RTO Schemes

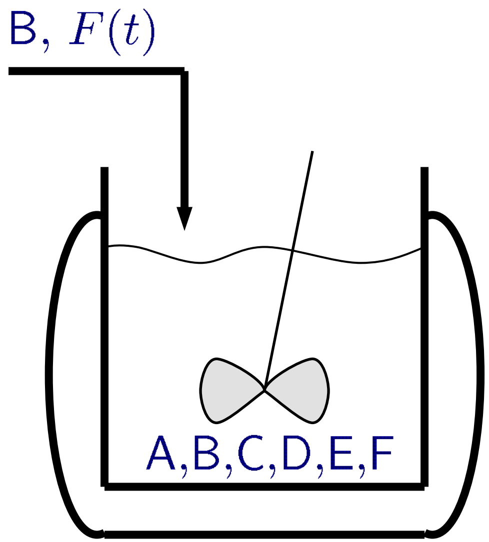
Run-to-Run Optimization of Semi-Batch Reactor

- **Industrial Reaction System** Lonza

Simulated Reality

$$\mathrm{A} + \mathrm{B} \xrightarrow{k_1} \mathrm{C}$$
$$2\mathrm{B} \xrightarrow{k_2} \mathrm{D}$$
~~$\mathrm{B} \xrightarrow{k_3} \mathrm{E}$~~
~~$\mathrm{C} + \mathrm{B} \xrightarrow{k_4} \mathrm{F}$~~

Model



- Manipulated Variables: $F(t)$ (feed flow rate of B)
- Objective: Maximize $n_{\mathrm{C}}(t_{\mathrm{f}})$ (production of C)
- Constraints:

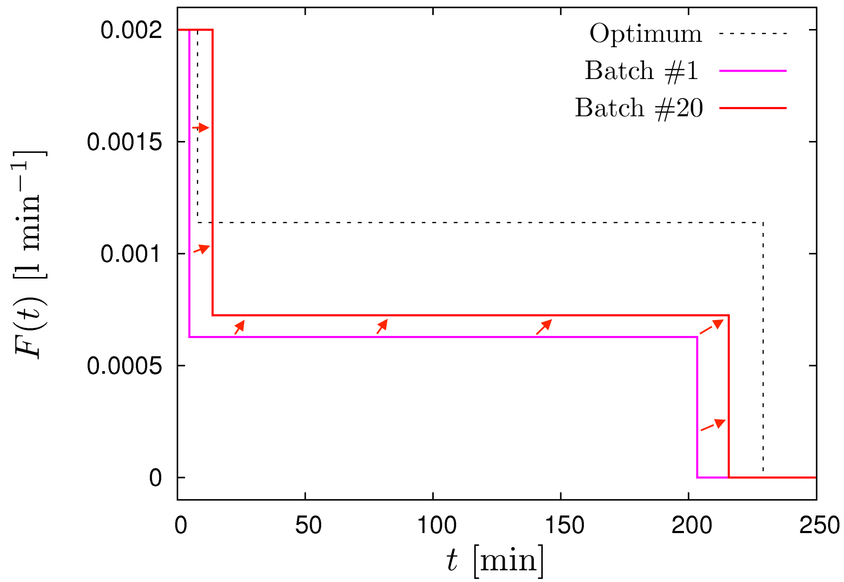
Input bounds: $0 \leq F(t) \leq 0.002 \text{ l min}^{-1}$

Terminal constraints: $c_{\mathrm{B}}(t_{\mathrm{f}}) \leq 0.025 \text{ mol l}^{-1}$ (max. residual concentration)

$c_{\mathrm{D}}(t_{\mathrm{f}}) \leq 0.15 \text{ mol l}^{-1}$ (max. by-product concentration)

Adaptation of Model Parameters k_1 and k_2

Laboratoire d'Automatique

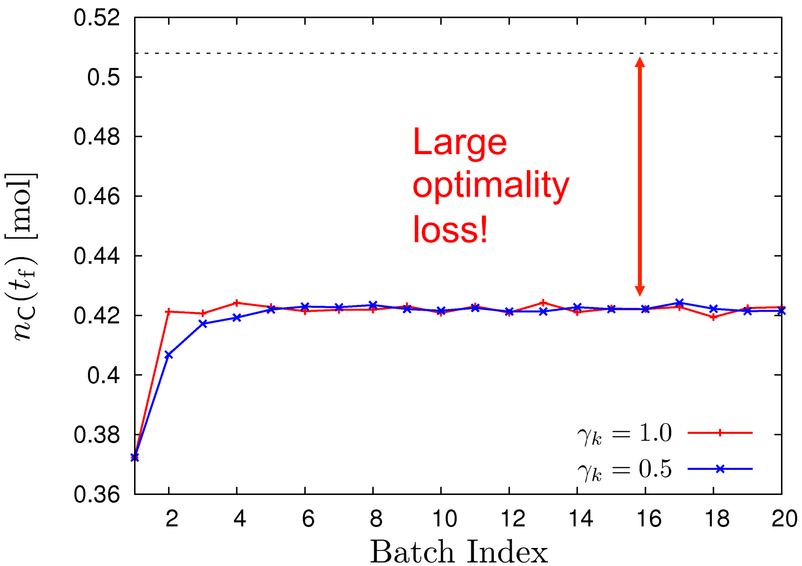


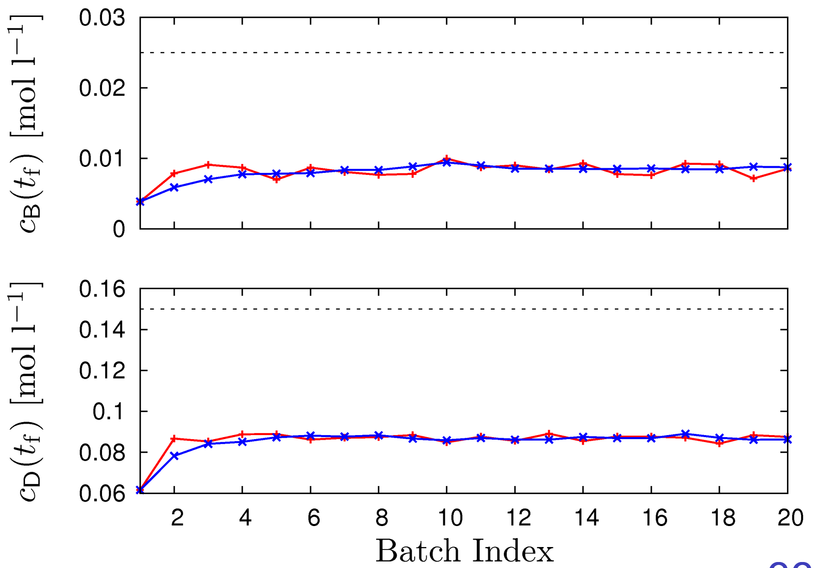
- Measurement Noise: $\sigma_y = 5\%$
  (10% constraint backoffs)
- Identification Objective:

$$J^{\text{id}} = \sum_{k=1}^{n^{\text{meas}}} \left[\frac{y - y^{\text{meas}}}{\overline{y}} \right]^2_{t=t_k}, \quad y = (c_\text{B}, c_\text{C}, c_\text{D}), \quad n^{\text{meas}} = 6$$

- Exponential Filter for k_1, k_2:

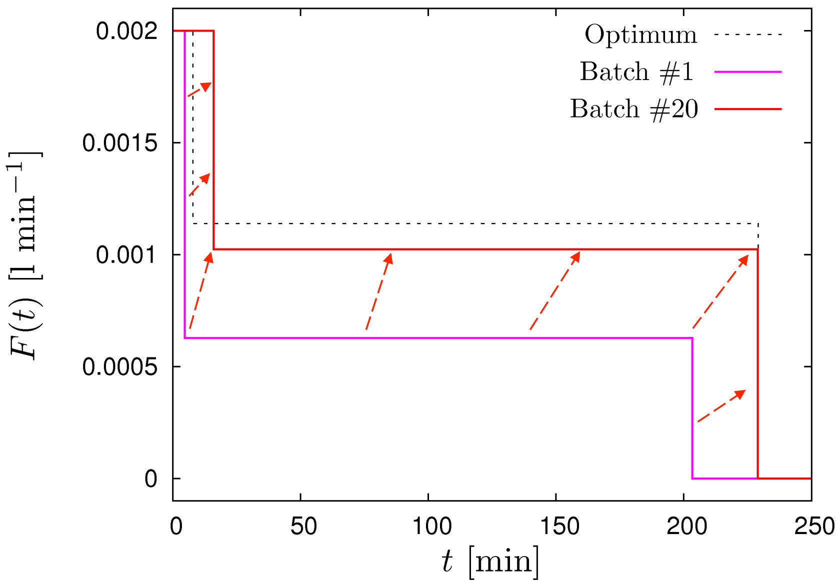
$$\begin{pmatrix} k_1^i \\ k_2^i \end{pmatrix} = (1 - \gamma_k) \begin{pmatrix} k_1^{i-1} \\ k_2^{i-1} \end{pmatrix} + \gamma_k \begin{pmatrix} k_1^\star \\ k_2^\star \end{pmatrix}$$





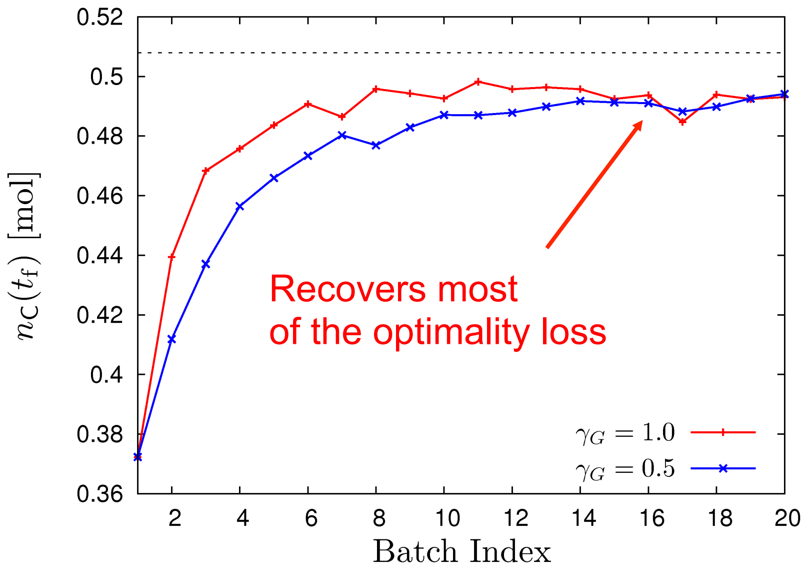
Adaptation of Constraint Modifiers ε_G

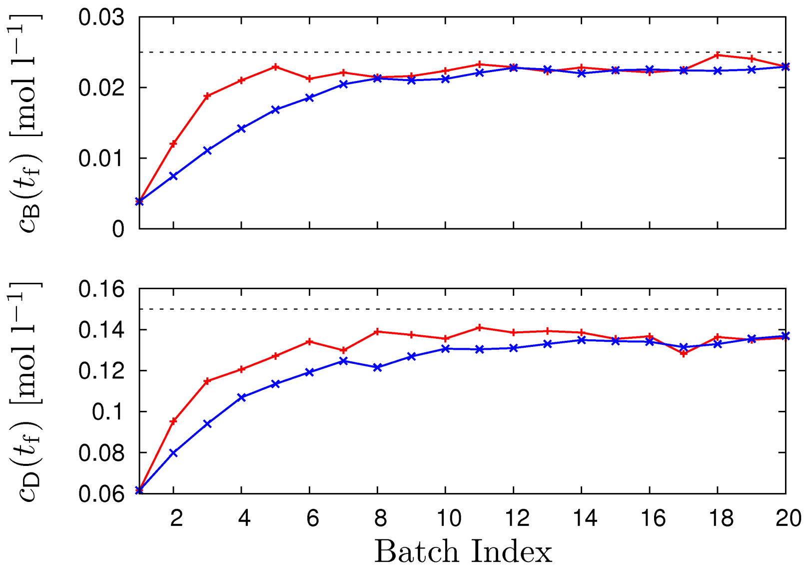
Laboratoire d'Automatique



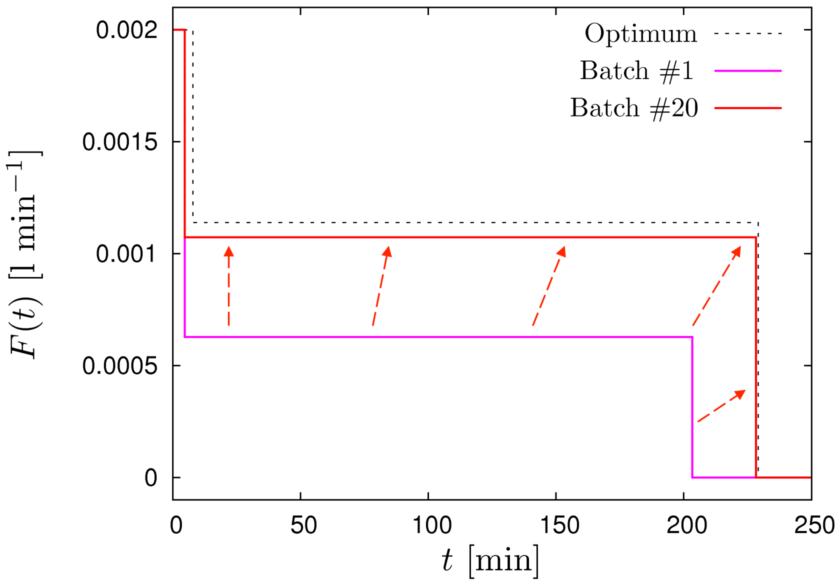
- Measurement Noise: $\sigma_y = 5\%$
  (10% constraint backoffs)
- No Gradient Correction
- Exponential Filter for Modifiers:

$$\begin{pmatrix} \varepsilon_{G,1}^{i} \\ \varepsilon_{G,2}^{i} \end{pmatrix} = (1-\gamma_G)\begin{pmatrix} \varepsilon_{G,1}^{i-1} \\ \varepsilon_{G,2}^{i-1} \end{pmatrix} + \gamma_G \begin{pmatrix} c_{\mathrm{B}}^{\mathrm{meas}}(t_{\mathrm{f}}) - c_{\mathrm{B}}(t_{\mathrm{f}}) \\ c_{\mathrm{D}}^{\mathrm{meas}}(t_{\mathrm{f}}) - c_{\mathrm{D}}(t_{\mathrm{f}}) \end{pmatrix}_{\boldsymbol{\pi}=\boldsymbol{\pi}^{i-1}}$$





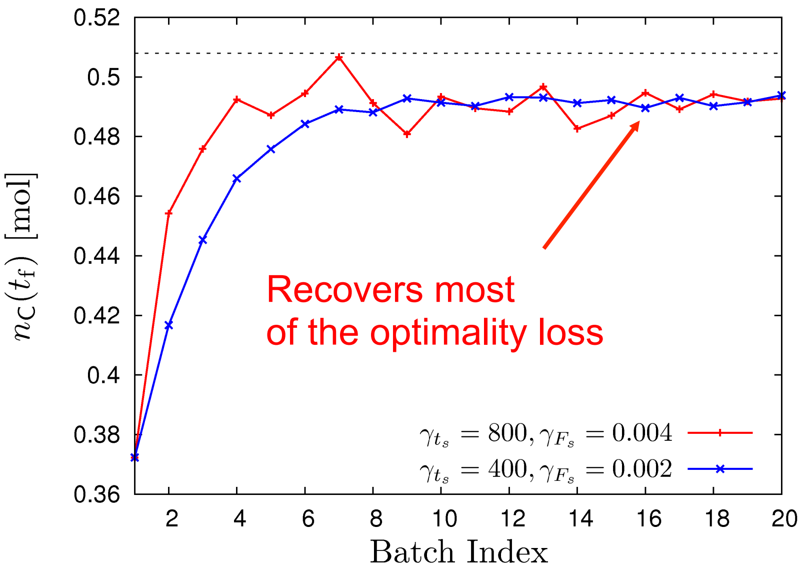
Adaptation of Input Parameters t_s and F_s

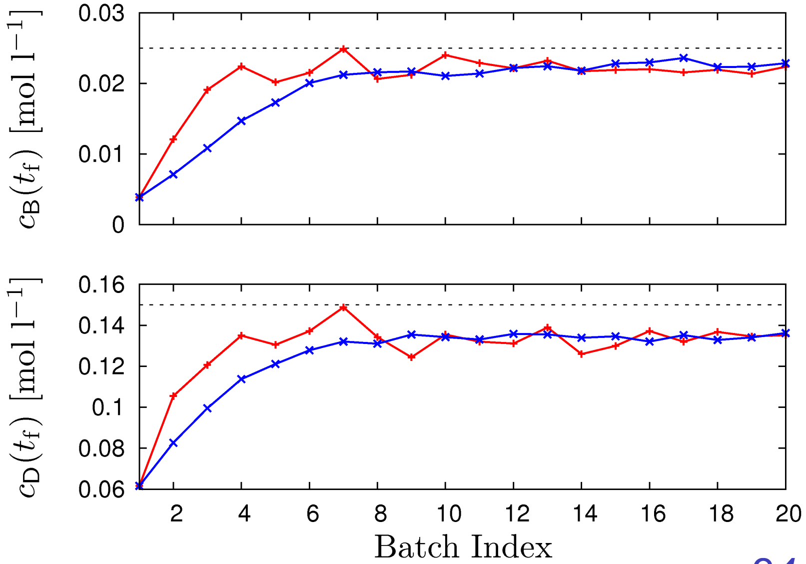


- Measurement Noise: $\sigma_y = 5\%$
  (10% constraint back-offs)
- No Gradient Correction
- Controller Design:

$$t_m = 4.71 \text{ min (fixed)}$$

$$\begin{pmatrix} t_s^k \\ F_s^k \end{pmatrix} = \begin{pmatrix} t_s^{k-1} \\ F_s^{k-1} \end{pmatrix} + \begin{pmatrix} \gamma_{t_s} & \\ & \gamma_{F_s} \end{pmatrix} \begin{pmatrix} c_B^{\text{meas}}(t_f) - 0.025 \\ c_D^{\text{meas}}(t_f) - 0.15 \end{pmatrix}_{\pi = \pi^{k-1}}$$





Outline

What is real-time optimization

- Goal: Optimal plant operation
- Tool: Model-based numerical optimization, experimental optimization
- Key feature: use of real-time measurements

Real-time optimization framework

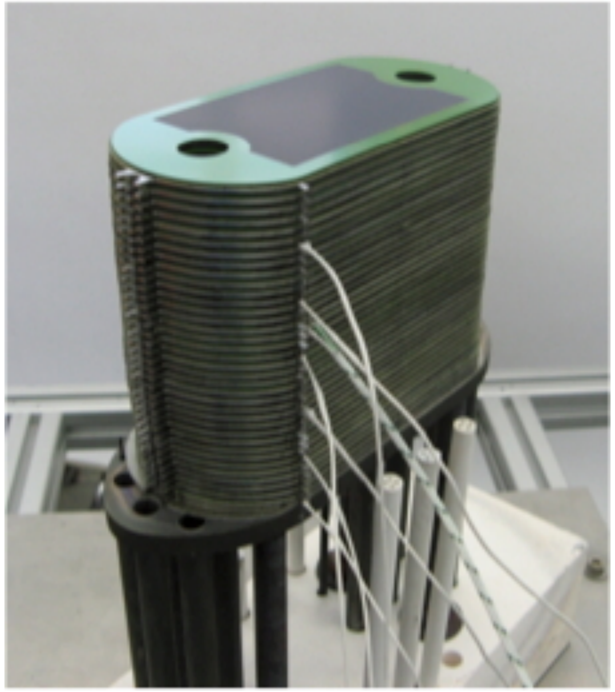
- Three approaches
- Key issues: Which measurements? How to best exploit them?
- Simulated comparison

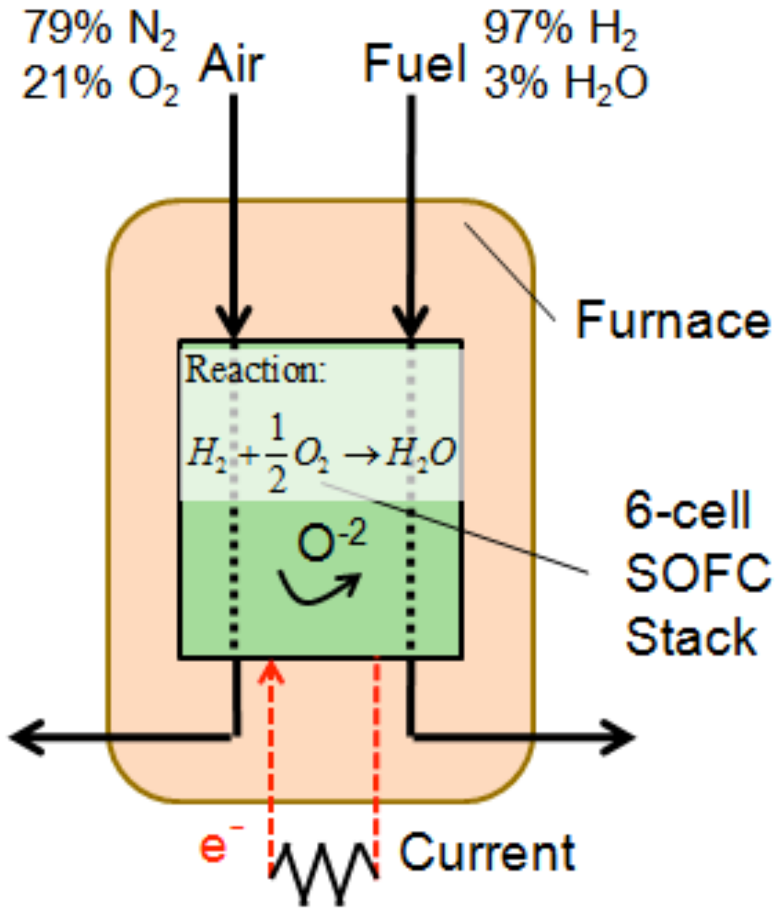
Experimental case studies

- Fuel-cell stack
- Batch polymerization

Laboratoire d'Automatique

Solid Oxide Fuel Cell Stack





- Stack of 6 cells, active area of 50 cm^2, metallic interconnector
- Anodes : standard nickel/yttrium stabilized-zirconia (Ni-YSZ)
- Electrolyte : dense YSZ.
- Cathodes: screen-printed (La, Sr)(Co, Fe)O_3
- Operation temperatures between 650 and 850°C.

G.A. Bunin, Z. Wuillemin, G. François, A. Nakajo, L. Tsikonis and D. Bonvin, Experimental real-time optimization of a solid oxide fuel cell stack via constraint adaptation, *Energy*, **39**(1), 54-62 (2012).

RTO via Constraint Adaptation

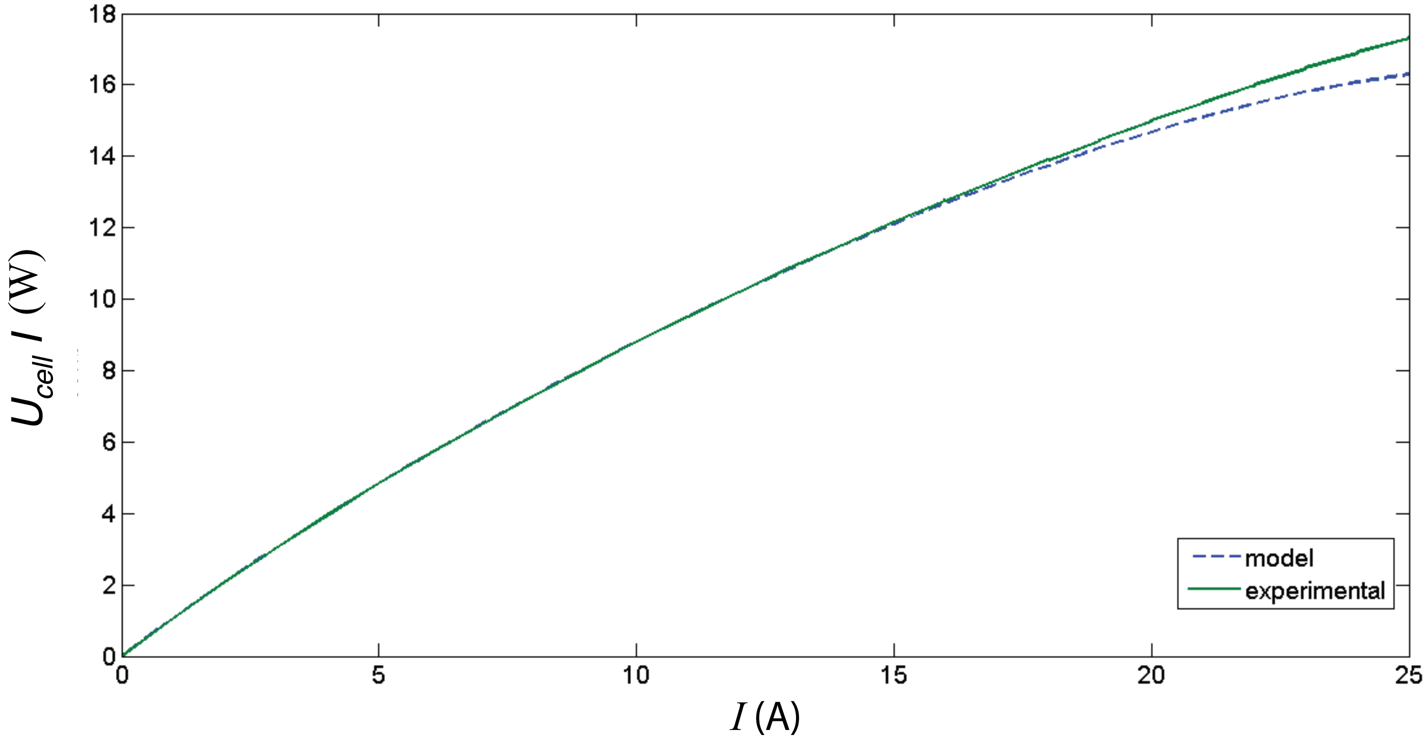
Laboratoire d'Automatique

Experimental features

- Inputs: flowrates (H_2, O_2), current (or load)
- Outputs: power density, cell potential, electrical efficiency
- Time-scale separation
    - *slow temperature dynamics, treated as process drift !*
    - *static model (for the rest)*
- Power demand changes without prior knowledge
- Inaccurate model in the operating region (power, cell)

RTO via Constraint Adaptation

Laboratoire d'Automatique



Challenge: Implement optimal operation with changing power demand

RTO via Constraint Adaptation

Laboratoire d'Automatique

Problem Formulation

At each RTO instant k, solve a static optimization problem, with a zeroth-order modifier in the constraints, regardless of the fact that T has reached steady state or not

$$
\begin{aligned}
\max_{u_k} \quad & \eta(\mathbf{u}_k, \mathbf{\Theta}) \\
\text{s.t.} \quad & p_{el}(\mathbf{u}_k, \mathbf{\Theta}) + \varepsilon_{k-1}^{p_{el}} = p_{el}^S \\
& U_{cell}(\mathbf{u}_k, \mathbf{\Theta}) + \varepsilon_{k-1}^{U_{cell}} \geq 0.75V \\
& \nu(\mathbf{u}_k) \leq 0.75 \\
& 4 \leq 2\frac{u_{2,k}}{u_{1,k}} = \lambda_{air}(\mathbf{u}_k) \leq 7 \\
& u_{1,k} \geq 3.14\,\text{mL/(min cm}^2) \\
& u_{3,k} \leq 30\text{A}
\end{aligned}
$$

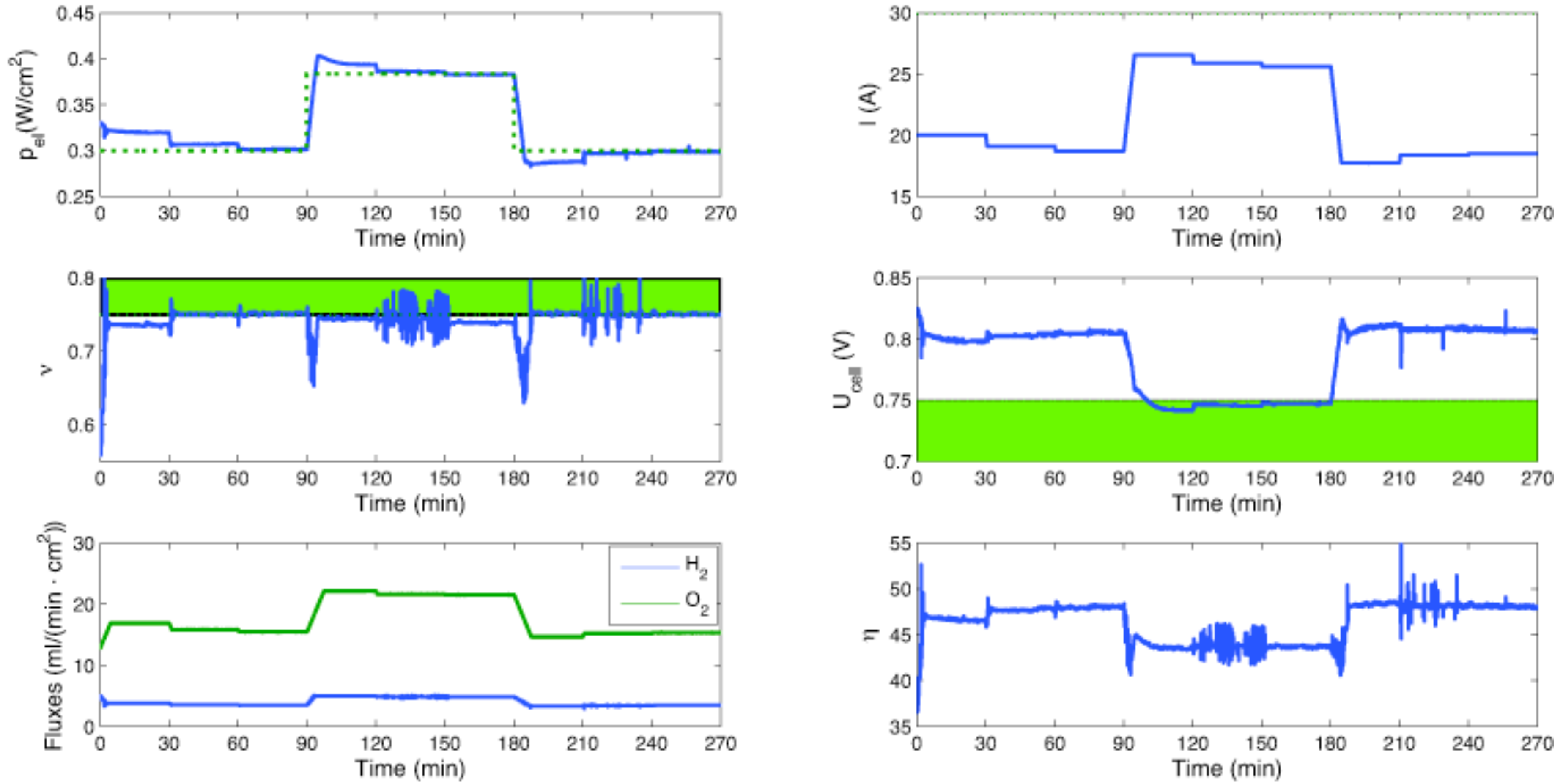
$$
u_k = \begin{bmatrix} u_{1,k} = \dot{n}_{H_2,k} \\ u_{2,k} = \dot{n}_{O_2,k} \\ u_{2,k} = I_k \end{bmatrix}
$$

$$
\begin{aligned}
\varepsilon_k^{p_{el}} &= \left(1\text{-}K_{p_{el}}\right)\varepsilon_{k-1}^{p_{el}} + \\
& \quad K_{p_{el}}\left[p_{el,p,k} - p_{el}(\mathbf{u}_k, \mathbf{\Theta})\right] \\
\varepsilon_k^{U_{cell}} &= \left(1\text{-}K_{U_{cell}}\right)\varepsilon_{k-1}^{U_{cell}} + \\
& \quad K_{U_{cell}}\left[U_{cell,p,k} - U_{cell}(\mathbf{u}_k, \mathbf{\Theta})\right]
\end{aligned}
$$

Slow RTO ("Wait for Steady State")

Laboratoire d'Automatique

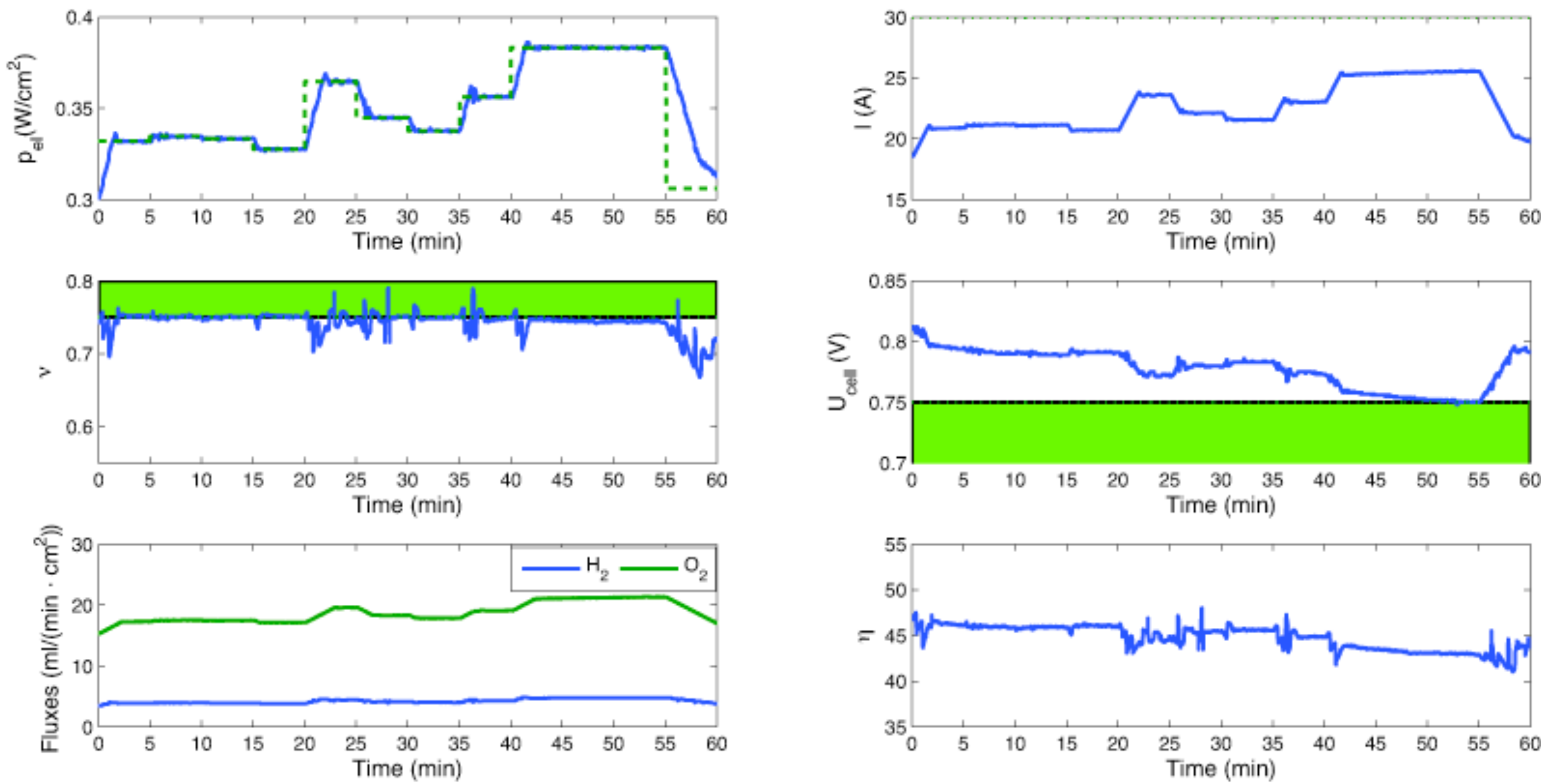
- RTO very 30 min
- Unknown power changes every 90 min



Fast RTO with Random Power Changes

Laboratoire d'Automatique

- Use steady-state model for predicting temperature
- RTO every 10 s, load changes every 5 min

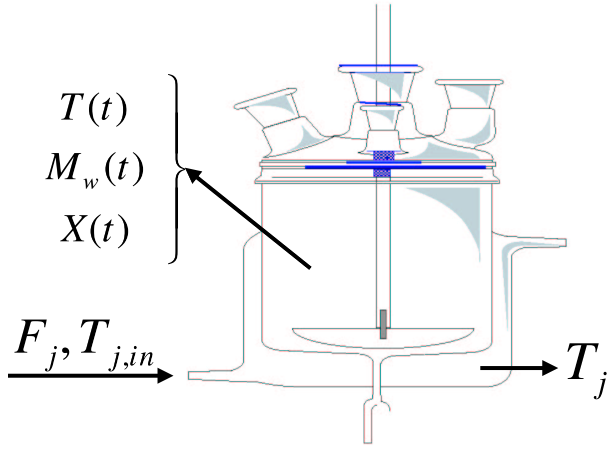


Emulsion Copolymerization Process

Laboratoire d'Automatique

- Industrial process
  - 1-ton reactor, risk of runaway
  - Initiator efficiency can vary considerably
  - Several recipes
    - *different initial conditions*
    - *different initiator feeding policies*
    - *use of chain transfer agent*

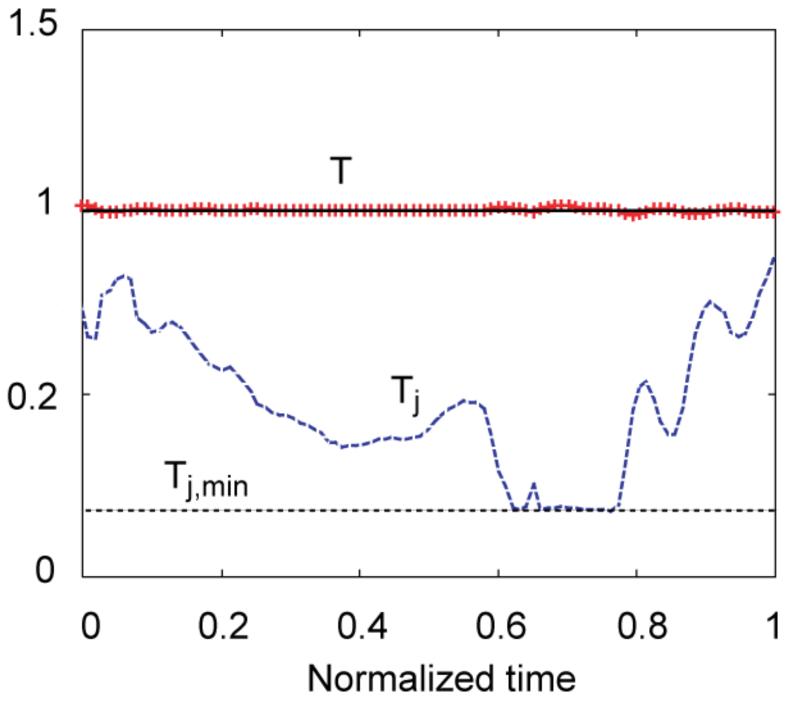
AQUA+TECH SPECIALTIES SA

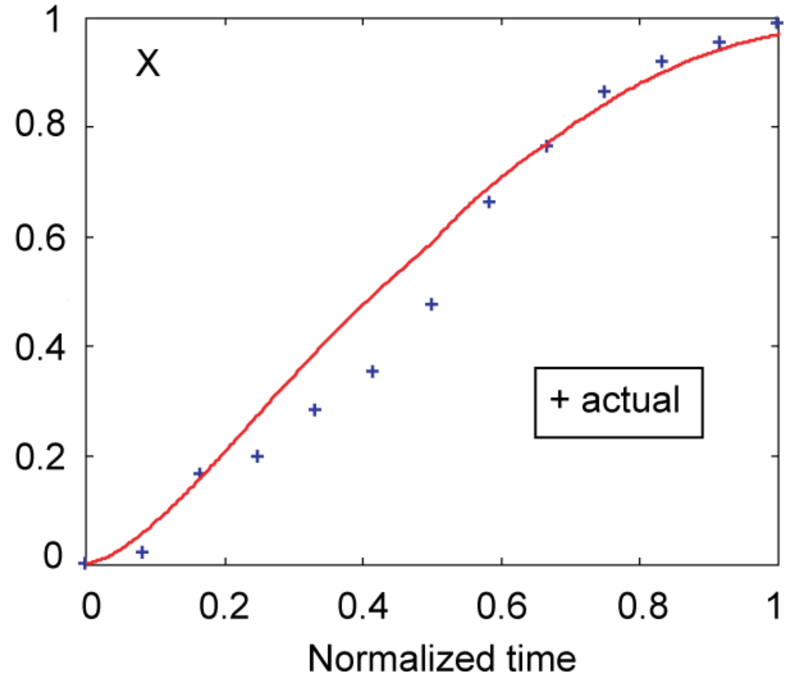


→ • Modeling difficulties • Uncertainty

- Objective: Minimize batch time by adjusting the reactor temperature
  - Temperature and heat removal constraints
  - Quality constraints at final time

Industrial Practice

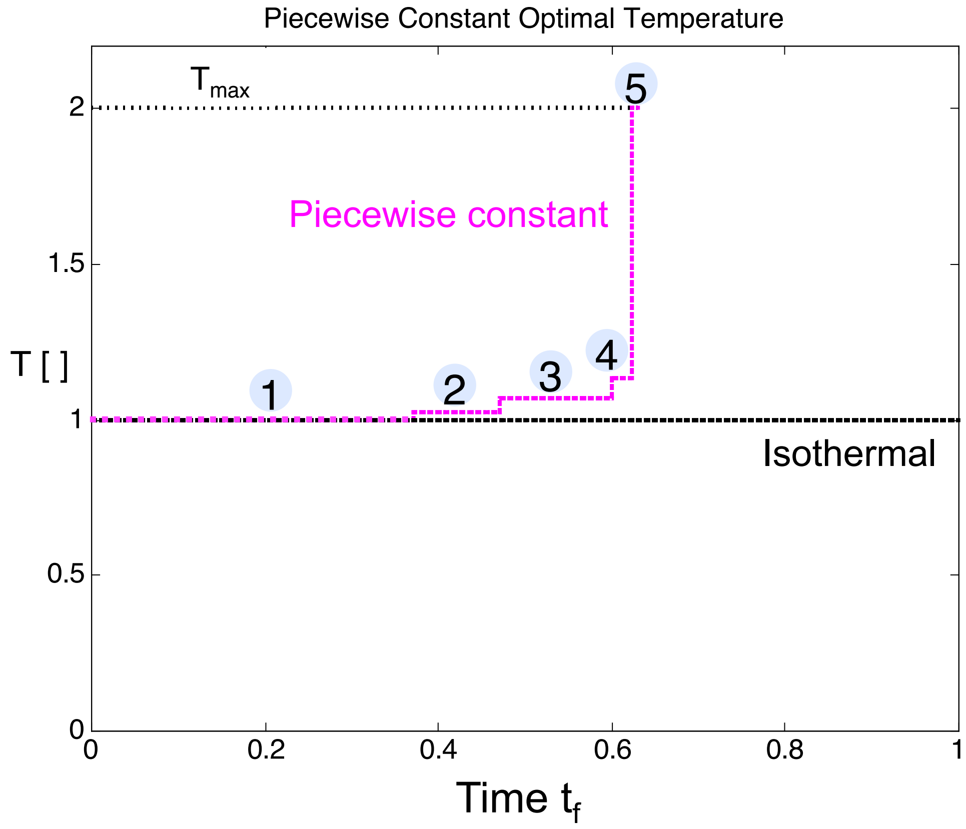




Optimal Temperature Profile

Numerical Solution using a Tendency Model

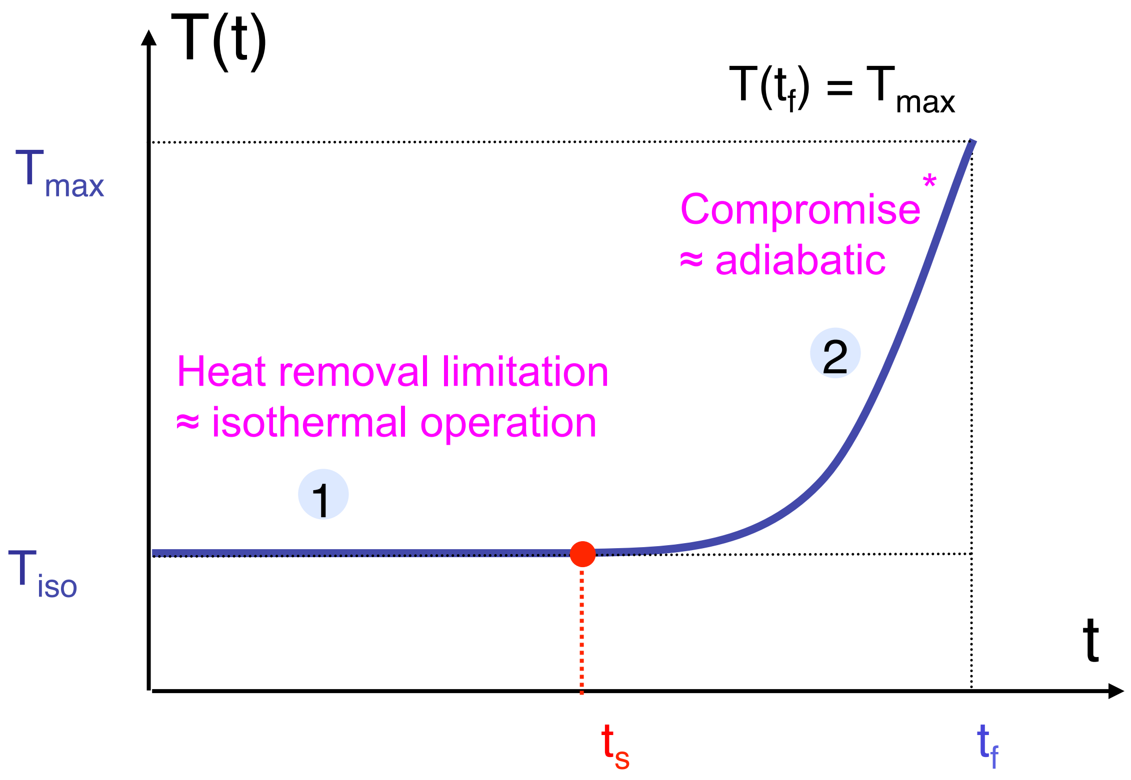
Laboratoire d'Automatique



- Current practice: isothermal
- Numerical optimization
  - ✓ Piecewise-constant input
  - ✓ 5 decision variables (T_2-T_5, t_f)
  - ✓ Fixed relative switching times
- Active constraints
  - ✓ Interval 1: heat removal
  - ✓ Interval 5: T_{max}

Model of the Solution

Semi-adiabatic Profile

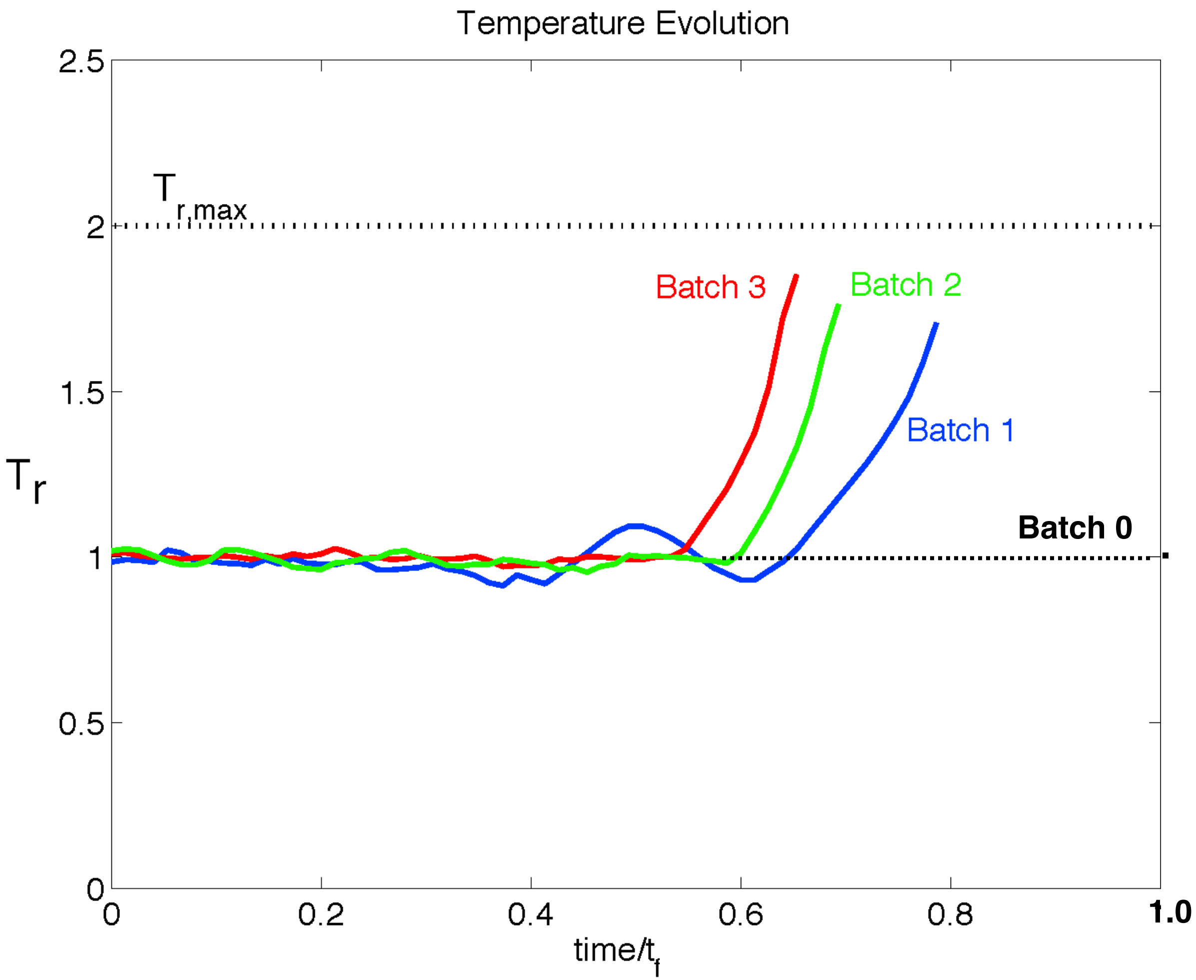


*Compromise between conversion and quality

t_s enforces $T(t_f) = T_{max}$

run-to-run adjustment of t_s

Industrial Results with NCO Tracking



1-ton reactor

Final time

- Isothermal: 1.00
- Batch 1: 0.78
- Batch 2: 0.72
- Batch 3: 0.65

Francois *et al.*, Run-to-run Adaptation of a Semi-adiabatic Policy for the Optimization of an Industrial Batch Polymerization Process, *I&EC Research*, **43**(23), 7238-7242, 2004

Conclusions

Laboratoire d'Automatique

Process optimization is difficult in practice

- Models are often inaccurate → use real-time measurements
- Repeated estimation and optimization lacks model adequacy
- Which measurements? How to best exploit them?

  → NCO (active constraints and reduced gradients)

Two appoaches involving the NCO

- Input-affine corrections to cost and constraints
- NCO tracking (optimization via a multivariable control problem)
- Key challenge is estimation of plant gradient

NCO tracking
New Paradigm for RTO

Operator-friendly approach

- Start with best current operation (nominal model-based solution) and push the process until constraints are reached
- Know what to manipulate → solution model
- Determine how much to change from measurements

Important features

- Two steps: offline (model-based), online (data-driven)
- Can test robustness offline by using model perturbations
- Approach converges to plant optimum, not model optimum
- Complexity depends on the number of inputs (not system order)
- Solution is partly determined by active constraints → easy tracking
- Price to pay: need to estimate experimental gradients