4.2 Steady-State Real-Time Optimization

Prof. Gabriele Pannocchia

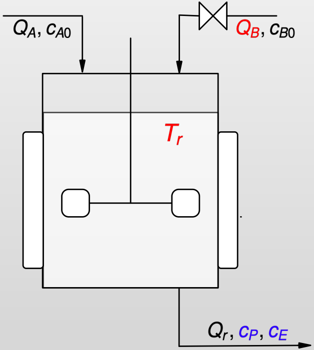
University of Pisa, Italy
gabriele.pannocchia@unipi.it

GRICU PhD School 2021
Digitalization Tools for the Chemical and Process Industries
March 19, 2021

Outline

A simple example to set the stage

Williams-Otto non-isothermal continuous-stirred tank reactor: introduction



Inputs: Q_B, T_r

Outputs: c_E, c_P

Actual reaction scheme:

$$
\begin{aligned}
&\mathrm{A} + \mathrm{B} \xrightarrow{k_1} \mathrm{C}, && r_1 = k_1(T_r)c_A c_B,\\
&\mathrm{B} + \mathrm{C} \xrightarrow{k_2} \mathrm{P} + \mathrm{E}, && r_2 = k_2(T_r)c_B c_C,\\
&\mathrm{C} + \mathrm{P} \xrightarrow{k_3} \mathrm{G}, && r_3 = k_3(T_r)c_C c_P.
\end{aligned}
$$

Model reaction scheme:

$$
\begin{aligned}
&\mathrm{A} + 2\mathrm{B} \xrightarrow{k_1^*} \mathrm{P} + \mathrm{E}, && r_1^* = k_1^*(T_r)c_A c_B^2,\\
&\mathrm{A} + \mathrm{B} + \mathrm{P} \xrightarrow{k_2^*} \mathrm{G}, && r_2^* = k_2^*(T_r)c_A c_B c_P.
\end{aligned}
$$

A simple example to set the stage

Williams-Otto non-isothermal continuous-stirred tank reactor: system dynamics

Plant:

$$\frac{dc_A}{dt} = \frac{Q_A c_{A0} - Q_r c_A}{V_r} - r_1,$$
$$\frac{dc_B}{dt} = \frac{Q_B c_{A0} - Q_r c_B}{V_r} - r_1 - r_2,$$
$$\frac{dc_C}{dt} = -\frac{Q_r c_C}{V_r} + r_1 - r_2 - r_3,$$
$$\frac{dc_P}{dt} = -\frac{Q_r c_P}{V_r} + r_2 - r_3,$$
$$\frac{dc_E}{dt} = -\frac{Q_r c_E}{V_r} + r_2,$$
$$\frac{dc_G}{dt} = -\frac{Q_r c_G}{V_r} + r_3.$$

Model:

$$\frac{dc_A}{dt} = \frac{Q_A c_{A0} - Q_r c_A}{V_r} - r_1^* - r_2^*,$$
$$\frac{dc_B}{dt} = \frac{Q_B c_{A0} - Q_r c_B}{V_r} - 2r_1^* - r_2^*,$$
$$\frac{dc_P}{dt} = -\frac{Q_r c_P}{V_r} + r_1^* - r_2^*,$$
$$\frac{dc_E}{dt} = -\frac{Q_r c_E}{V_r} + r_1^*,$$
$$\frac{dc_G}{dt} = -\frac{Q_r c_G}{V_r} + r_2^*,$$

A simple example to set the stage

Optimizing steady-state operation: what is desired?

- Operating cost: $\ell_c = Q_A c_{A0} p_A + Q_B c_{B0} p_B - Q_r c_P p_P - Q_r c_E p_E$
- Operating (manipulated) variables: $u = \begin{bmatrix} Q_B & T_r \end{bmatrix}$

The plant-based solution

- Plant state vector: $x_p = \begin{bmatrix} c_A & c_B & c_C & c_P & c_E & c_G \end{bmatrix}$
- Plant dynamics:

$$\frac{dx_p}{dt} = f_p(x_p, u)$$

- The plant optimal steady state:

$$(x^\star, u^\star) = \arg\min_{u, x_p} \ell_c$$

$$\text{subject to:} \qquad 0 = f_p(x_p, u)$$

A simple example to set the stage

Optimizing steady-state operation: what can be computed?

The model-based solution

- Model state vector: $x = \begin{bmatrix} c_A & c_B & c_P & c_E & c_G \end{bmatrix}$
- Model dynamics:

$$\frac{dx}{dt} = f(x, u)$$

- The model optimal steady-state:

$$(\bar{x}, \bar{u}) = \arg\min_{u,x} \ell_c$$

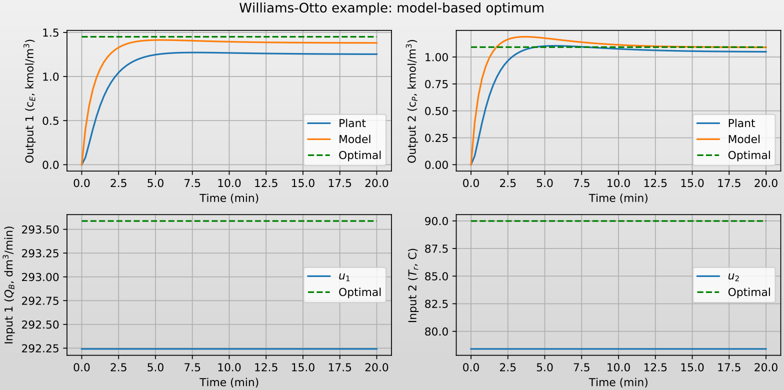
$$\text{subject to:} \qquad 0 = f(x, u)$$

- The model-based solution is only approximate because the model is not exact due to plant-model mismatch.

Question: Can the plant still be driven to true plant optimality?

A simple example to set the stage

Results: implementing the nominal model-based solution



Actual profit: 151.3 vs. True optimal profit: 193.2. Hence 22% profit loss!

A simple example to set the stage

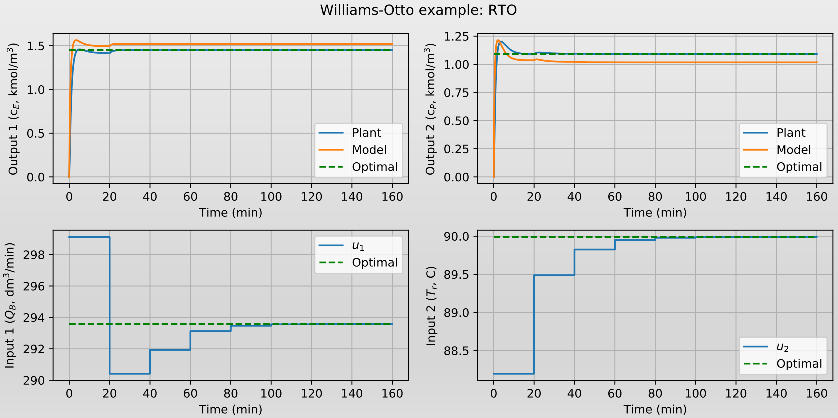
How to reach plant optimality?

Real-Time Optimization

1. Implement the model-based solution
2. Wait for the system to reach a steady state
3. Use steady-state data (input and output) to modify model
4. Solve the modified model-based optimization problem
5. Go to 1

Results: Does RTO work?

RTO scheme with bias and gradient correction



A simple example to set the stage

RTO pros vs cons

Pros

- Convergence to the correct (... **unknown**) steady state is achieved when plant gradients can be evaluated/estimated can be estimated accurately
- Constraints are fulfilled during iterations

Cons

- Need to wait for the plant to reach steady state at each iteration –> Dynamic RTO (next module)
- Estimation of plant gradients may not be simple –> Various approaches (discussed next)

Outline

General formulation of an optimization problem

The three ingredients

1. $x \in \mathbb{R}^n$, vector of variables
2. $f : \mathbb{R}^n \to \mathbb{R}$, scalar objective function to be minimized
3. $g : \mathbb{R}^n \to \mathbb{R}^m$, vector function of m **inequality** constraints that the variables must satisfy
   $h : \mathbb{R}^n \to \mathbb{R}^p$ vector function of p **equality** constraints that the variables must satisfy

The optimization problem

$$\min_{x \in \mathbb{R}^n} f(x) \qquad \text{subject to} \begin{cases} g_i(x) \leq 0 & i = 1, \ldots, m \\ h_j(x) = 0 & j = 1, \ldots, p \end{cases}$$

Example of an optimization problem (1/2)

Example and standard form

- Starting problem

$$\min\ (x_1 - 2)^2 + (x_2 - 1)^2 \qquad \text{subject to} \begin{cases} x_1^2 - x_2 & \leq 0 \\ x_1 + x_2 & \leq 2 \end{cases}$$

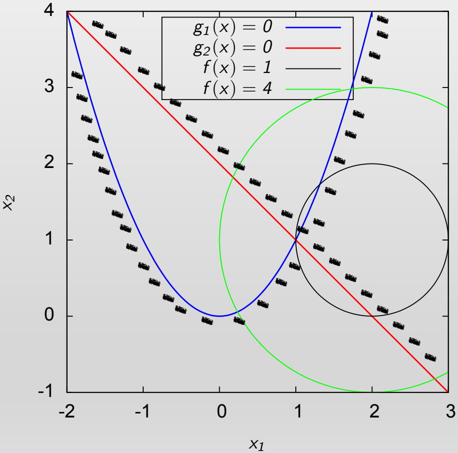
- Rewritten in standard form

$$f(x) = (x_1 - 2)^2 + (x_2 - 1)^2, \qquad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$g(x) = \begin{bmatrix} g_1(x) \\ g_2(x) \end{bmatrix} = \begin{bmatrix} x_1^2 - x_2 \\ x_1 + x_2 - 2 \end{bmatrix}, \qquad h(x) = []$$

Example of an optimization problem (2/2)

Feasible region and objective function level curves



Constrained optimization: example 1

$$\min \; x_1 + x_2 \qquad \text{s. t.} \; 2 - x_1^2 - x_2^2 = 0$$

Standard notation, feasibility region and solution

- In standard notation: $f(x) = x_1 + x_2$, $g(x) = []$, $h_1(x) = 2 - x_1^2 - x_2^2$
- Feasibility region: circle of radius $\sqrt{2}$, only the border
- Solution: $x^\star = [-1, -1]^T$

Observation

$$\nabla f(x^\star) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad \nabla h_1(x^\star) = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

The constraint normal vector $\nabla h_1(x^\star)$ is parallel to the cost function gradient $\nabla f(x^\star)$:

$$\nabla f(x^\star) + \nu_1^\star \nabla h_1(x^\star) = 0 \qquad \text{with } \nu_1^\star = -\frac{1}{2}$$

Constrained optimization: example 2

$$\min \; x_1 + x_2 \qquad \text{s. t.} \;\; 2 - x_1^2 - x_2^2 \geq 0$$

Standard notation, feasibility region and solution

- In standard notation: $f(x) = x_1 + x_2$, $g_1(x) = x_1^2 + x_2^2 - 2$
- Feasibility region: circle of radius $\sqrt{2}$, including the interior
- Solution: $x^\star = [-1, -1]^T$

Observation

$$\nabla f(x^\star) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad \nabla g_1(x^\star) = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$$

The constraint normal vector $\nabla g_1(x^\star)$ is parallel to the cost function gradient $\nabla f(x^\star)$:

$$\nabla f(x^\star) + \lambda_1^\star \nabla g_1(x^\star) = 0 \qquad \text{with } \lambda_1^\star = \frac{1}{2}$$

Constrained optimality conditions (KKT)

Lagrangian function

$$\mathcal{L}(x, \mu, \nu) = f(x) + \sum_{i=1}^{m} \mu_i g_i(x) + \sum_{j=1}^{p} \nu_j h_j(x)$$

Karush-Kuhn-Tucker optimality conditions (necessary)

- If $x^\star$ is a local solution to the standard problem, there exist $\mu^\star \in \mathbb{R}^m$ and $\nu^\star \in \mathbb{R}^p$:

$$\begin{aligned}
\nabla_x \mathcal{L}(x^\star, \mu^\star, \nu^\star) &= 0 & & \\
h_j(x^\star) &= 0 & & j = 1, \dots, p \\
g_i(x^\star) &\leq 0 & & i = 1, \dots, m \\
\mu_i^\star &\geq 0 & & i = 1, \dots, m \\
\mu_i^\star g_i(x^\star) &= 0 & & i = 1, \dots, m
\end{aligned}$$

- A multiplier μ_i is zero when the corresponding constraint is **inactive**, i.e. $g_i(x^\star) < 0$.

Elaborating the KKT conditions

Observations

- Exploiting the definition of $\mathcal{L}(x, \mu, \nu)$, we obtain:

$$\nabla_x \mathcal{L}(x, \mu, \nu) = \nabla f(x) + \nabla g(x)\mu + \nabla h(x)\nu$$

- Note that: $\nabla h(x) \in \mathbb{R}^{n \times m}$ and $\nabla g(x) \in \mathbb{R}^{n \times p}$

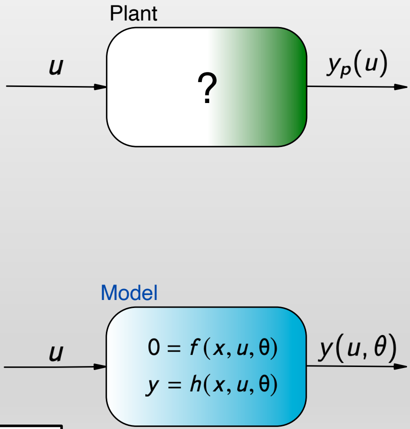
Revised KKT conditions

$$\begin{aligned}
\nabla f(x^\star) + \nabla h(x^\star)\mu^\star + \nabla g(x^\star)\nu^\star &= 0 \\
h_j(x^\star) &= 0 \qquad j = 1, \dots, p \\
g_i(x^\star) &\leq 0 \qquad i = 1, \dots, m \\
\mu_i^\star &\geq 0 \qquad i = 1, \dots, m \\
\mu_i^\star g_i(x^\star) &= 0 \qquad i = 1, \dots, m
\end{aligned}$$

Outline

RTO problem formulation

Plant and model



- Plant cost function to be minimized:

$$\Phi_p(u) = \phi(y_p(u), u)$$

- Constraints to be fulfilled:

$$G(y_p(u), u) \leq 0$$

- **Model-based** optimization problem:

$$\min_u \Phi(u, \theta) := \phi(y(u, \theta), u)$$
$$\text{s.t.} \qquad G(y(u, \theta), u) \leq 0$$

RTO problem formulation

KKT optimality conditions and matching

- Model-based optimization problem KKTs:

$$
\begin{aligned}
\nabla \Phi(u^\star, \theta) + \nabla G(y(u^\star, \theta), u^\star)\mu^\star &= 0 \\
G_i(y(u^\star, \theta), u^\star) &\leq 0 && i = 1, \dots, m \\
\mu_i^\star &\geq 0 && i = 1, \dots, m \\
\mu_i^\star G_i(y(u^\star, \theta), u^\star) &= 0 && i = 1, \dots, m
\end{aligned}
$$

- Plant optimization problem KKTs:

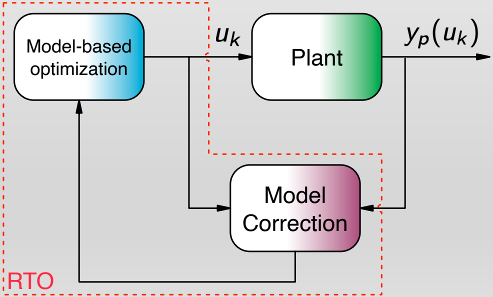
$$
\begin{aligned}
\nabla \Phi_p(u^\star) + \nabla G(y_p(u^\star, \theta), u^\star)\mu^\star &= 0 \\
G_i(y_p(u^\star), u^\star) &\leq 0 && i = 1, \dots, m \\
\mu_i^\star &\geq 0 && i = 1, \dots, m \\
\mu_i^\star G_i(y_p(u^\star), u^\star) &= 0 && i = 1, \dots, m
\end{aligned}
$$

RTO iterative scheme

KKT matching

- For the plant and model KKT conditions to match, i.e. to achieve the true optimum:

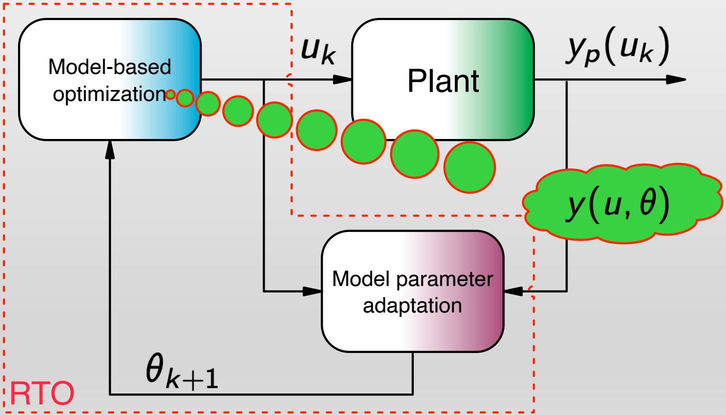
$$y_p(u^\star) - y(u^\star, \theta) = 0 \qquad \text{output value matching}$$
$$\nabla y_p(u^\star) - \nabla y(u^\star, \theta) = 0 \qquad \text{output gradient matching}$$



Outline

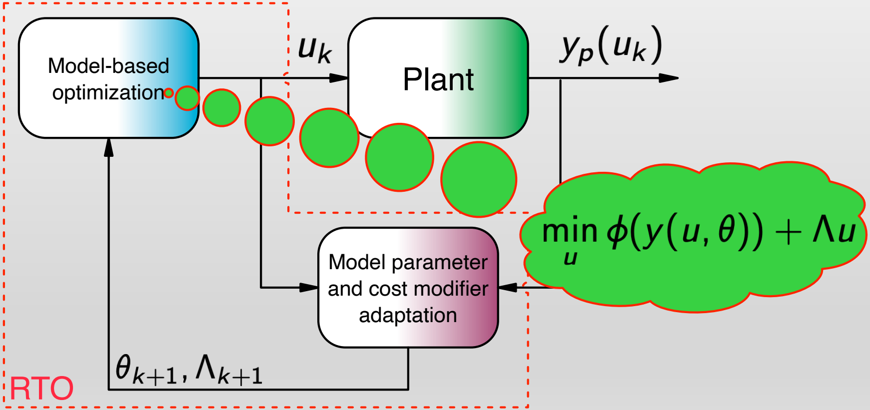
Different RTO approaches

Two-step approach



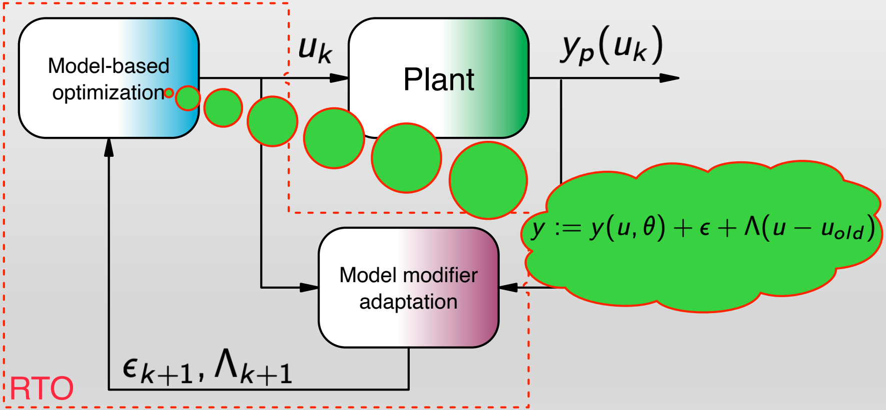
Different RTO approaches

ISOPE (Integrated System Optimization and Parameter Estimation)



Different RTO approaches

Modifier Adaptation (MA)



Different RTO approaches

Some details about the correction terms

Model parameter adaptation

- The model parameters θ are updated via data-reconciliation tools

Modifier adaptation

- 0-order modifier:

$$\epsilon_{k+1} = y_p(u_k) - y(u_k, \theta)$$

- 1-st modifier (MA):

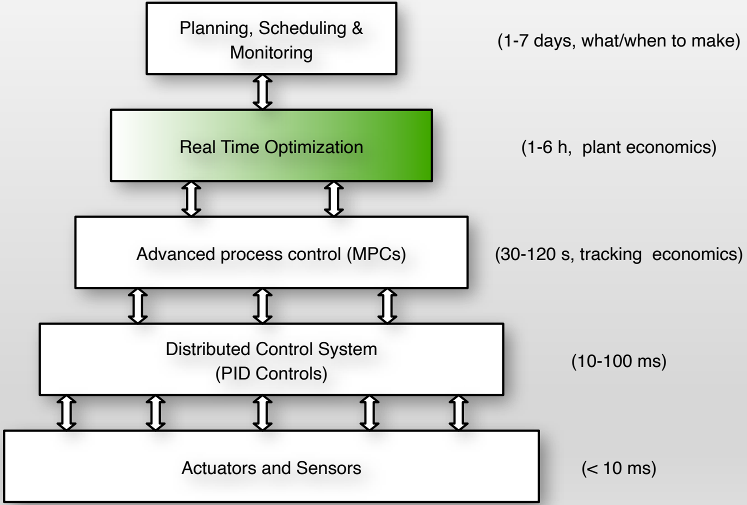
$$\Lambda_{k+1} = \nabla y_p(u_k) - \nabla y(u_k, \theta)$$

- 1-st modifier (ISOPE):

$$\Lambda_{k+1} = \nabla_y \phi(y, u) \left[\nabla y_p(u_k) - \nabla y(u_k, \theta)\right]$$

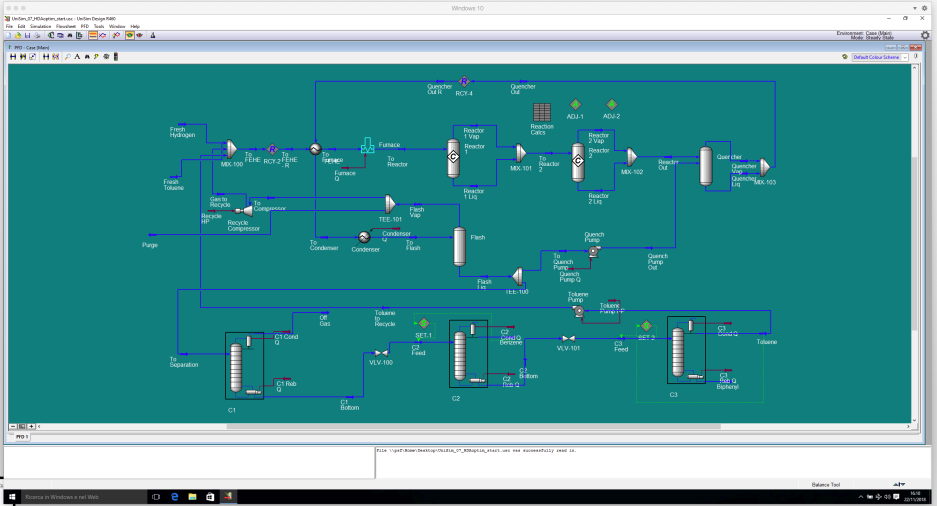
Outline

Hierarchical scheme of optimization, monitoring and control



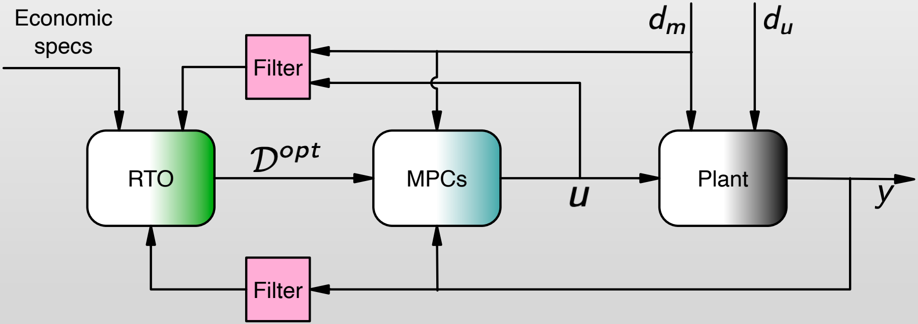
RTO first-principles models

Steady-state process simulators



RTO and MPC

Bi-directional connection



RTO and MPC

Examples of RTO and MPC variables

RTO variables		**MPC variables**	
Constraints	Decisions to MPC	Constraints	Manipulated setpoints (in DCS)
Reactor conversion	Desired targets	Temperature	Flow
Production rates	Min/max limits	Level	Temperature
MPC constraints	Costs/economic priorities	Composition	Pressure
		Column DP	Valve positions
		Compressor power	
		Valve positions (PID outputs)	

Outline

An introductory example

Some theory on numerical optimization

Steady-state RTO: Problem formulation and iterative approach

Various RTO approaches

Industrial implementations

Conclusions

RTO: Conclusions

- Optimal operation typically resides at the intersection of multiple constraints
- The basic structure of RTO in cascade to MPC has become the standard approach for implementing steady-state optimization in plants that operate around nominal steady states
- The advent of open equation modeling and SQP optimization techniques has enabled rigorous steady state optimizations to be formulated and reliably solved
- Ensuring plant optimality needs input-output gradient estimation: Broyden, RLS, etc. may overcome current limits of FD

I'd like to thank Prof. Dominique Bonvin (EPFL) for his comments on this lecture and for sharing with me his (infinite) knowledge on RTO

References I

Maurício M Câmara, André D Quelhas, and José Carlos Pinto. Performance evaluation of real industrial RTO systems. *Processes*, 4(4):44, 2016.

Benoît Chachuat, Balasubramaniam Srinivasan, and Dominique Bonvin. Adaptation strategies for real-time optimization. *Computers & Chemical Engineering*, 33(10):1557–1567, 2009.

Tianyou Chai, S Joe Qin, and Hong Wang. Optimal operational control for complex industrial processes. *Annual Reviews in Control*, 38(1):81–92, 2014.

Mark L Darby, Michael Nikolaou, James Jones, and Doug Nicholson. RTO: An overview and assessment of current practice. *Journal of Process Control*, 21(6):874–884, 2011.

Johannes Jäschke and Sigurd Skogestad. NCO tracking and self-optimizing control in the context of real-time optimization. *Journal of Process Control*, 21(10):1407–1416, 2011.

Johannes Jäschke, Yi Cao, and Vinay Kariwala. Self-optimizing control–a survey. *Annual Reviews in Control*, 43:199–223, 2017.

André D Quelhas, Normando José Castro de Jesus, and José Carlos Pinto. Common vulnerabilities of RTO implementations in real chemical processes. *The Canadian Journal of Chemical Engineering*, 91(4):652–668, 2013.

Sen Xie, Chunhua Yang, Xiaofeng Yuan, Xiaoli Wang, and Yongfang Xie. A novel robust data reconciliation method for industrial processes. *Control Engineering Practice*, 83:203–212, 2019.