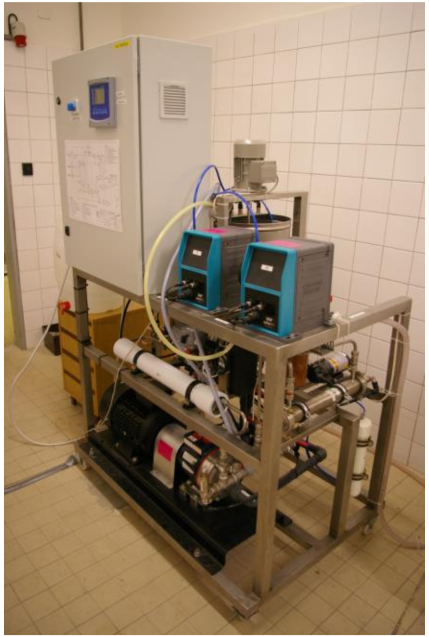
Experimental Real Time Optimization of a Continuous Membrane Separation Plant

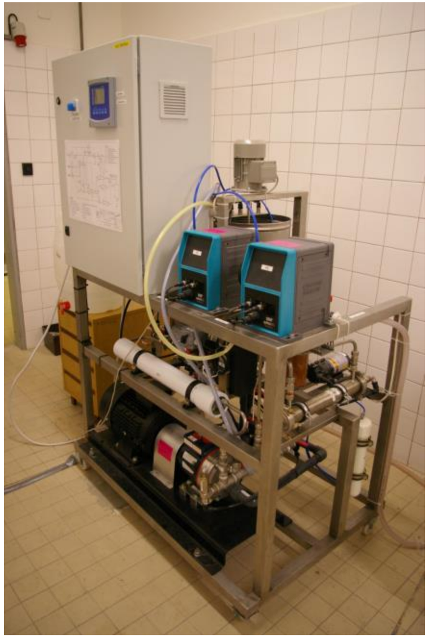
Anwesh Reddy Gottu Mukkula[1], Petra Valiauga[2], Miroslav Fikar[2], Radoslav Paulen[2], Sebastian Engell[1]

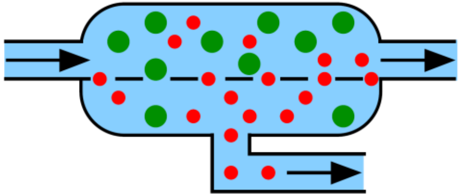
[1] Process Dynamics and Operations Group, Department of Biochemical and Chemical Engineering, Technical University of Dortmund, Germany
[2] Faculty of Chemical and Food Technology, Slovak University of Technology in Bratislava, Slovakia

Motivation

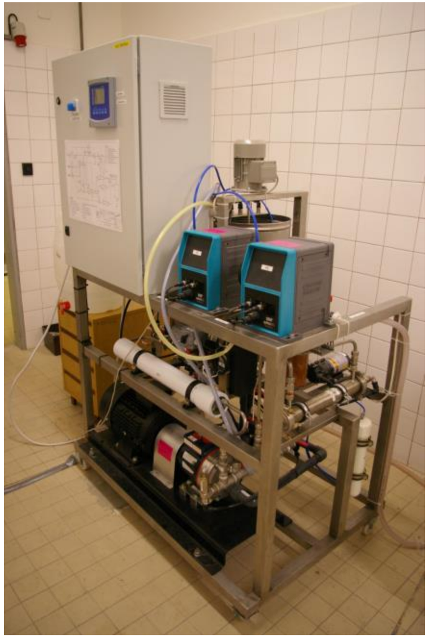


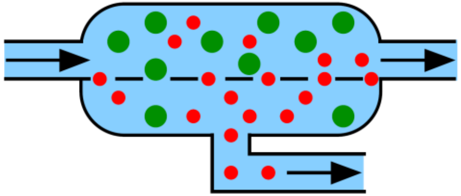
Motivation





Motivation



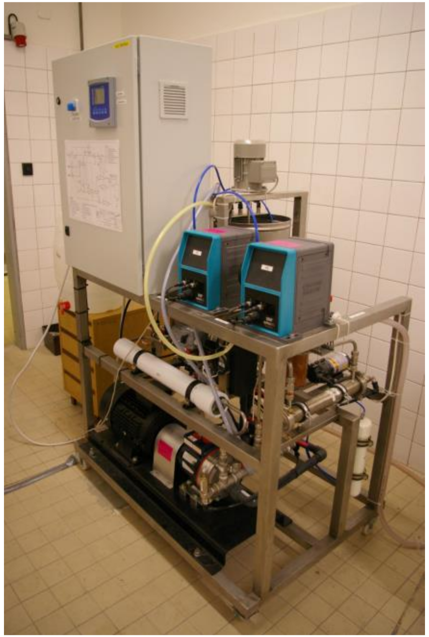


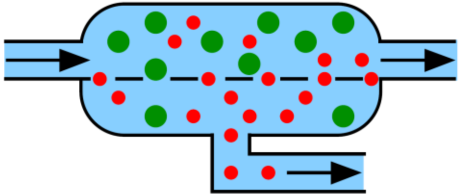
Goal

- Identify the optimal operating input for a nanofiltration membrane separation process
- Process and productivity constraints have to be taken into account

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

Motivation





Goal

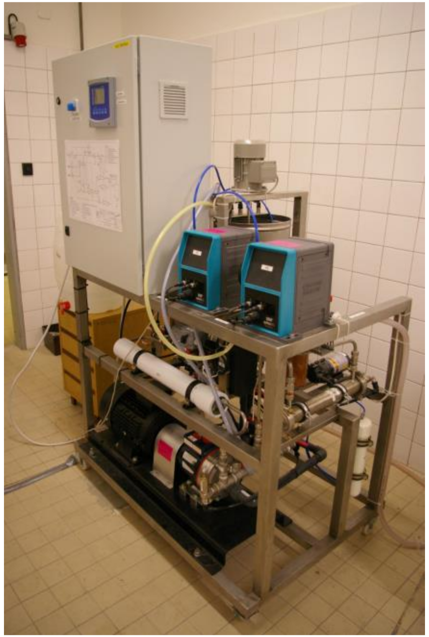
- Identify the optimal operating input for a nanofiltration membrane separation process
- Process and productivity constraints have to be taken into account

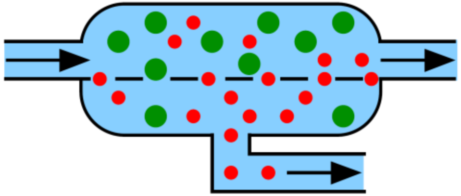
Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t. } \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

Motivation





Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t. } \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

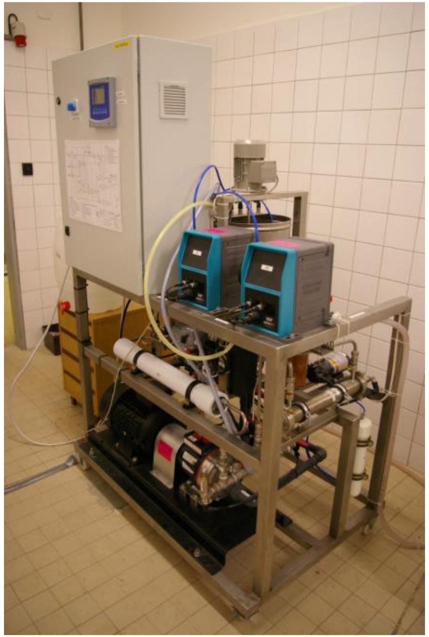
Goal

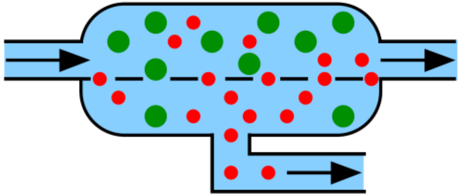
- Identify the optimal operating input for a nanofiltration membrane separation process
- Process and productivity constraints have to be taken into account

Challenge

- Unknown process model

Motivation





Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t.} \;\; \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

Goal

- Identify the optimal operating input for a nanofiltration membrane separation process
- Process and productivity constraints have to be taken into account

Challenge

- Unknown process model

Proposed solution

- Real-time optimization methods

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

Real-time optimization

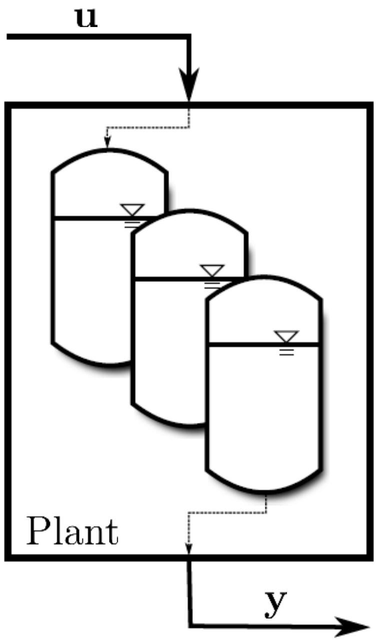


Figure: Illustration of a general plant.

Real-time optimization

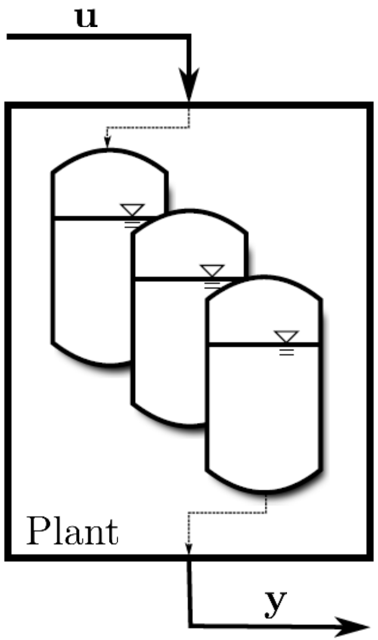


Figure: Illustration of a general plant.

Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t. } \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

Model:

$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{u})$$

$$\mathbf{u}_m^* \neq \mathbf{u}_p^*$$

Real-time optimization

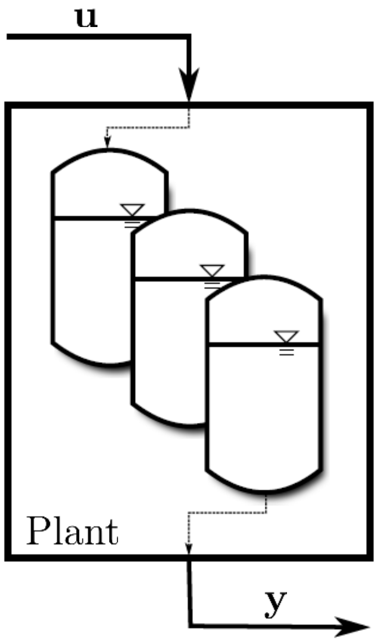


Figure: Illustration of a general plant.

Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t.} \;\; \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

Model:

$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t.} \;\; \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{u})$$

$$\mathbf{u}_m^* \neq \mathbf{u}_p^*$$

RTO method

- Modifier adaptation with quadratic approximation (MAWQA)

Real-time optimization

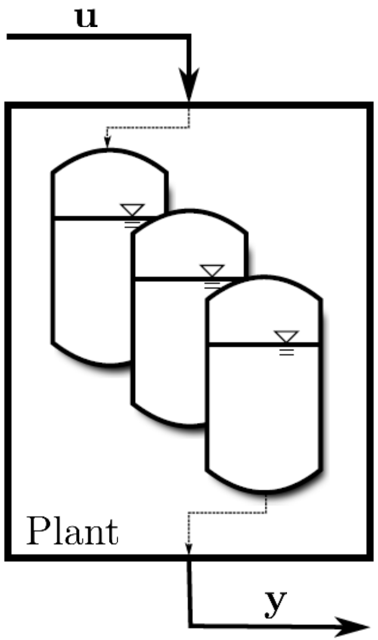


Figure: Illustration of a general plant.

Plant:

$$\mathbf{u}_p^* = \min_{\mathbf{u}} \mathcal{J}_p(\mathbf{y}, \mathbf{u})$$
$$\text{s.t. } \mathbf{y} = \mathbf{f}_p(\mathbf{u})$$

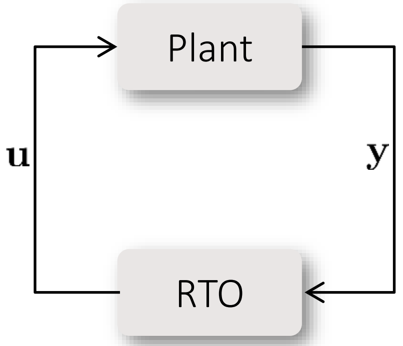
Model:

$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{u})$$

$$\mathbf{u}_m^* \neq \mathbf{u}_p^*$$

RTO method

- Modifier adaptation with quadratic approximation (MAWQA)



Modifier adaptation (MA)

Model-based optimization

$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m(\mathbf{x}, \mathbf{u}) \leq 0$$

Modifier adaptation (MA)

$$\hat{\mathbf{u}}_m^{*k+1} = \min_{\mathbf{u}} \mathcal{J}_m^{ad,k}(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m^{ad,k}(\mathbf{x}, \mathbf{u}) \leq 0$$

Modifier adaptation (MA)

Model-based optimization

$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m(\mathbf{x}, \mathbf{u}) \leq 0$$

Modifier adaptation (MA)

$$\hat{\mathbf{u}}_m^{*k+1} = \min_{\mathbf{u}} \mathcal{J}_m^{ad,k}(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m^{ad,k}(\mathbf{x}, \mathbf{u}) \leq 0$$

Modified objective function and constraints:

$$\mathcal{J}_m^{ad,k}(\mathbf{x}, \mathbf{u}) = \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u}) + (\nabla \mathcal{J}_p^k - \nabla \mathcal{J}_m^k)^T (\mathbf{u} - \mathbf{u}^k)$$
$$\mathbf{g}_m^{ad,k}(\mathbf{x}, \mathbf{u}) = \mathbf{g}_m(\mathbf{x}, \mathbf{u}) + (\mathbf{g}_p^k - \mathbf{g}_m^k) + (\nabla \mathbf{g}_p^k - \nabla \mathbf{g}_m^k)^T (\mathbf{u} - \mathbf{u}^k)$$

Modifier adaptation (MA)

Model-based optimization

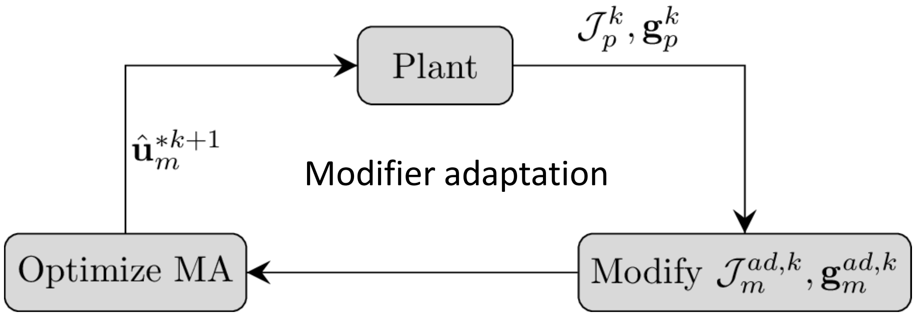
$$\mathbf{u}_m^* = \min_{\mathbf{u}} \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m(\mathbf{x}, \mathbf{u}) \leq 0$$

Modifier adaptation (MA)

$$\hat{\mathbf{u}}_m^{*k+1} = \min_{\mathbf{u}} \mathcal{J}_m^{ad,k}(\hat{\mathbf{y}}, \mathbf{u})$$
$$\text{s.t. } \hat{\mathbf{y}} = \mathbf{f}_m(\mathbf{x}, \mathbf{u})$$
$$\mathbf{g}_m^{ad,k}(\mathbf{x}, \mathbf{u}) \leq 0$$

Modified objective function and constraints:

$$\mathcal{J}_m^{ad,k}(\mathbf{x}, \mathbf{u}) = \mathcal{J}_m(\hat{\mathbf{y}}, \mathbf{u}) + (\nabla \mathcal{J}_p^k - \nabla \mathcal{J}_m^k)^T (\mathbf{u} - \mathbf{u}^k)$$
$$\mathbf{g}_m^{ad,k}(\mathbf{x}, \mathbf{u}) = \mathbf{g}_m(\mathbf{x}, \mathbf{u}) + (\mathbf{g}_p^k - \mathbf{g}_m^k) + (\nabla \mathbf{g}_p^k - \nabla \mathbf{g}_m^k)^T (\mathbf{u} - \mathbf{u}^k)$$



Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA

Finite differences

- Used for gradient approximation when not enough points are available for fitting a quadratic function
- When number of available points are less than $\frac{(n_u+1)(n_u+2)}{2}$

[1] Gao, W., Wenzel, S., and Engell, S. (2016), “A reliable modifier-adaptation strategy for real-time optimization”, Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA

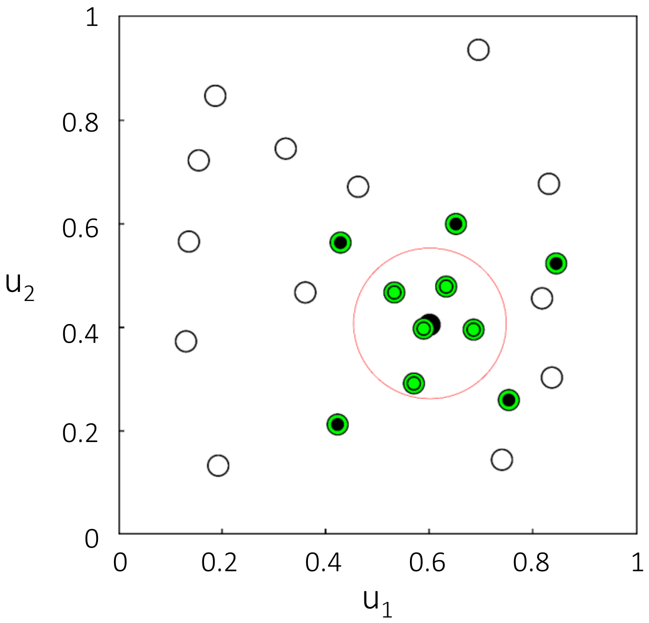


Figure: Illustration of screening algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), “A reliable modifier-adaptation strategy for real-time optimization”, Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

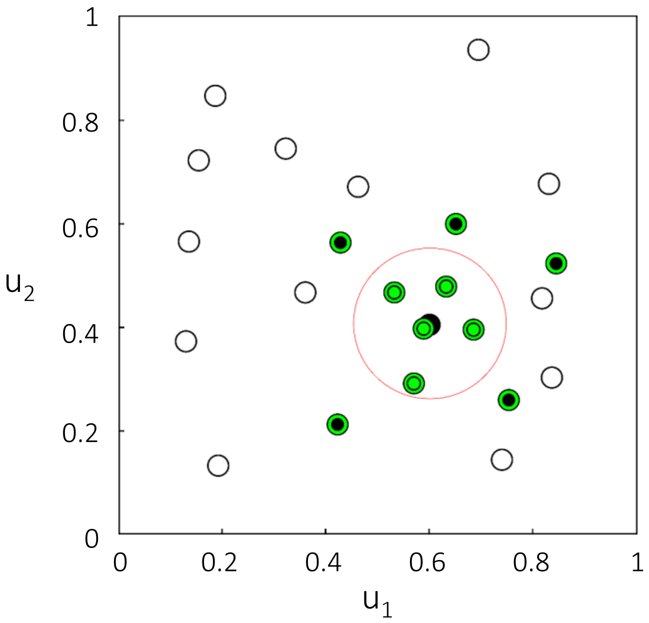


Figure: Illustration of screening algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

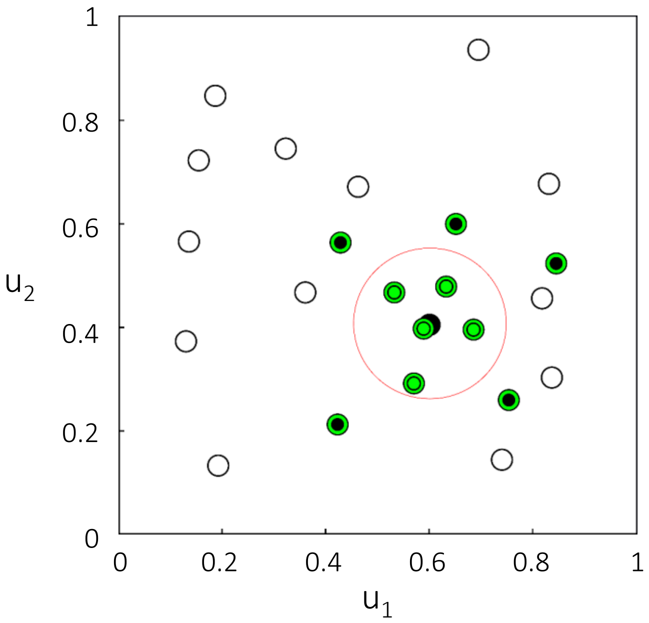


Figure: Illustration of screening algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

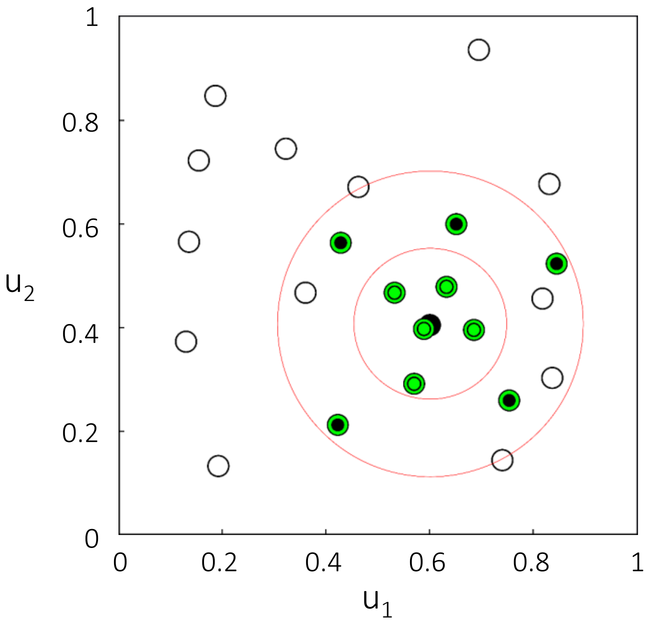


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

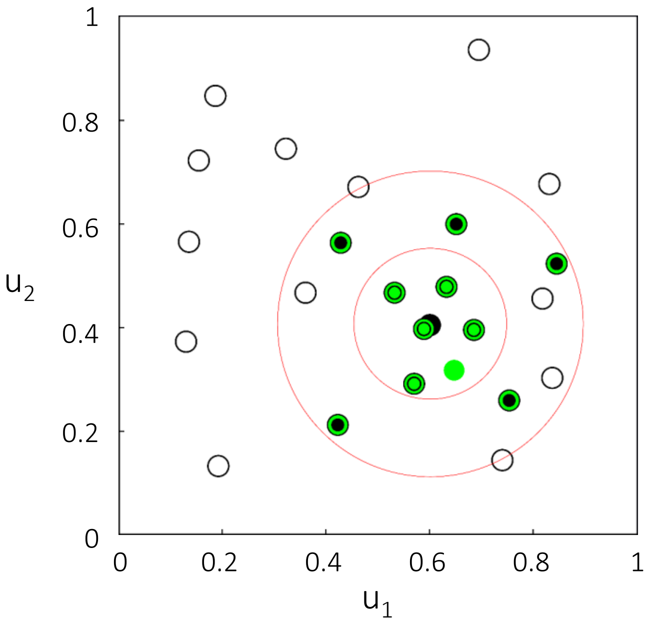


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

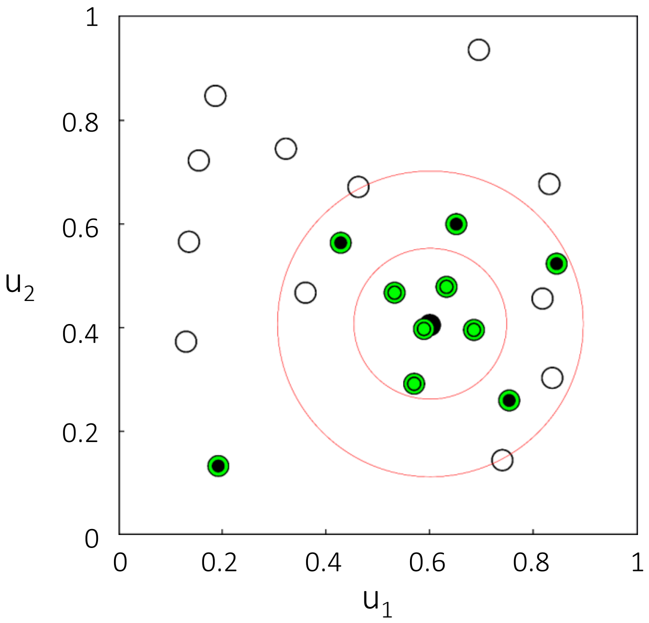


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), “A reliable modifier-adaptation strategy for real-time optimization”, Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

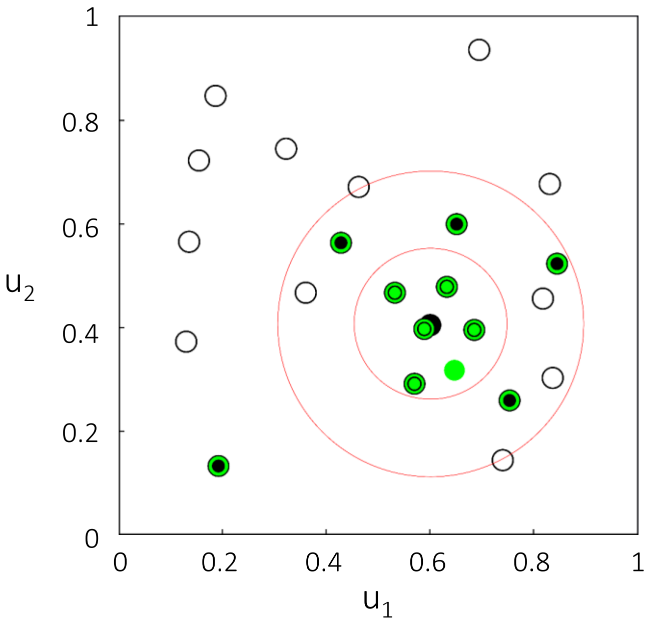


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

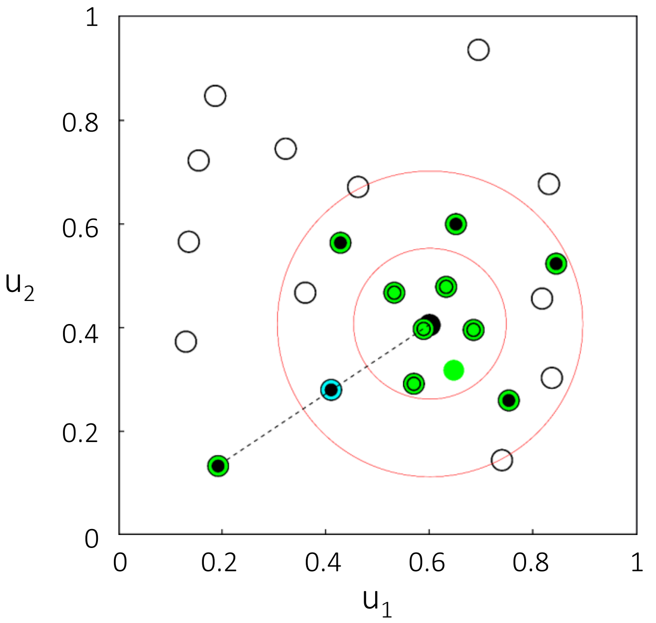


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), "A reliable modifier-adaptation strategy for real-time optimization", Computers & Chemical Engineering, 91, 318 - 328.

Modifier adaptation with quadratic approximation (MAWQA)[1]

Quadratic approximation (QA)

$$\mathcal{Q}(\mathcal{P}, \mathbf{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c$$

$$\mathcal{P} = \{a_{1,1}, \cdots, a_{n_u,n_u}, b_1, \cdots, b_{n_u}, c\}$$

- Screening algorithm to choose points for QA → **Criticality Check**

- Additional trust-region constraint:

$$(\mathbf{u} - \mathbf{u}^k)^T cov(\mathcal{U}^k)(\mathbf{u} - \mathbf{u}^k) - \gamma^2 \leq 0$$

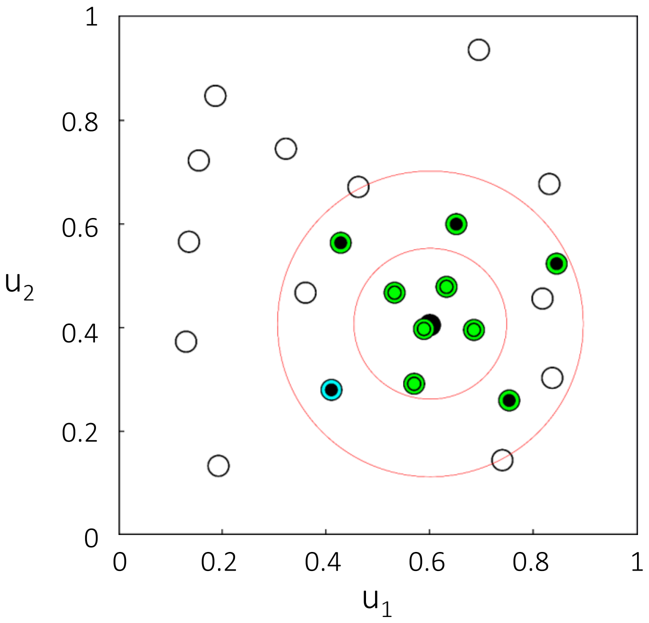


Figure: Illustration of criticality check algorithm[1].

[1] Gao, W., Wenzel, S., and Engell, S. (2016), “A reliable modifier-adaptation strategy for real-time optimization”, Computers & Chemical Engineering, 91, 318 - 328.

Membrane separation process

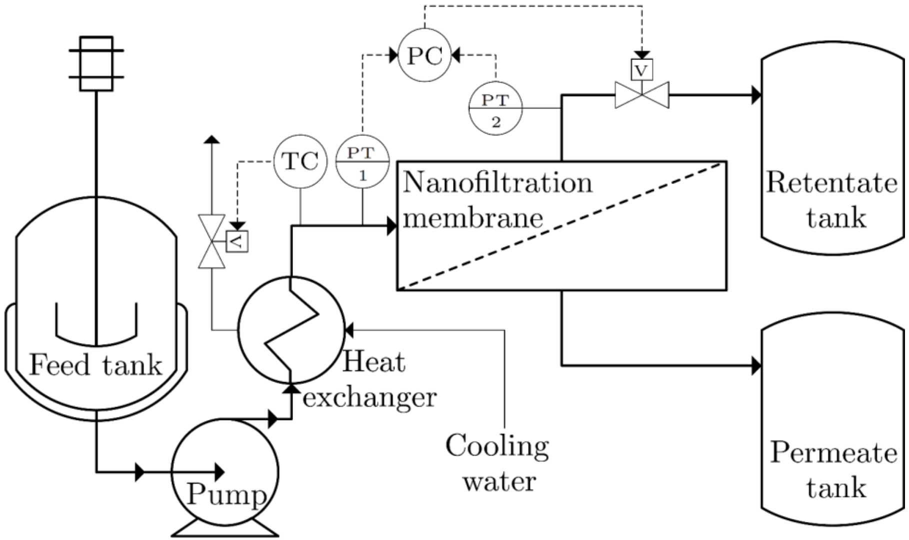


Figure: Illustration of the process flow diagram of the continuously operated membrane separation process.

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

Membrane separation process

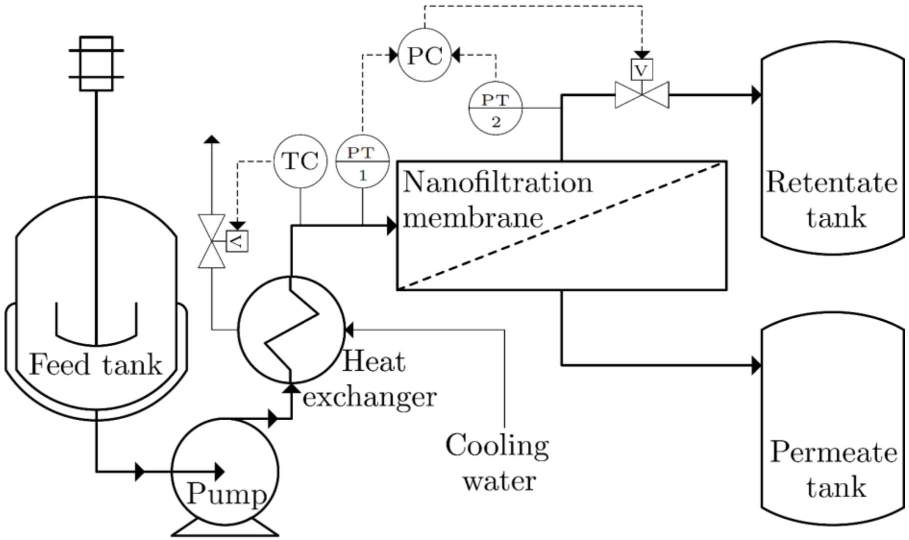


Figure: Illustration of the process flow diagram of the continuously operated membrane separation process.

$$\max_{\boldsymbol{u}=\{\Delta P, T\}} \quad \dot{V}_p \kappa_p - 50\Delta P^2 - 5\Delta P(T_{\text{amb}} - T)$$

$$\text{s.t. nominal model,}$$

$$18 - \dot{V}_p \leq 0,$$

$$4 \leq \Delta P \leq 22,$$

$$20 \leq T \leq 30.$$

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

Membrane separation process

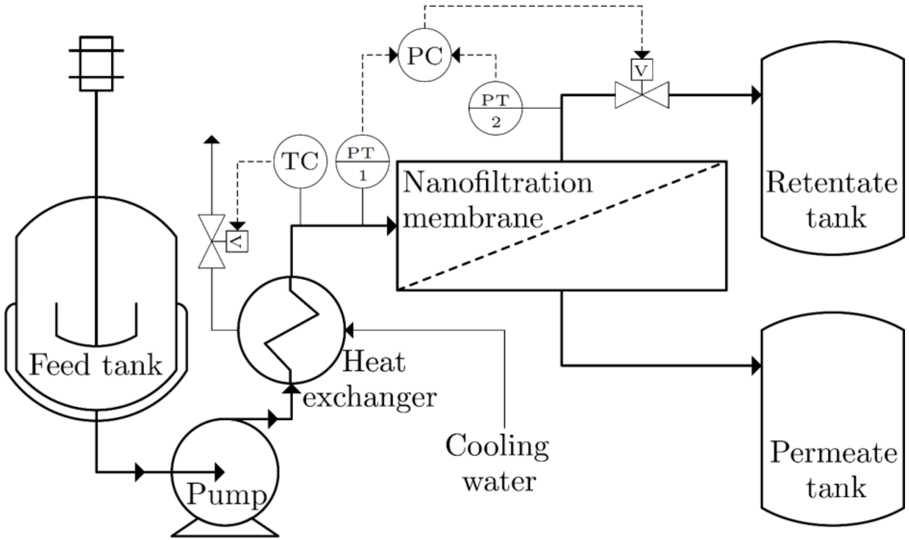
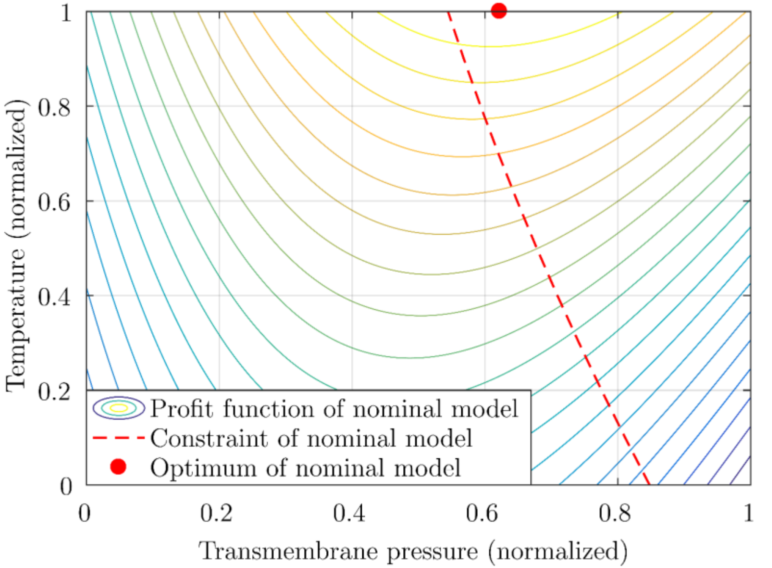


Figure: Illustration of the process flow diagram of the continuously operated membrane separation process.



$$\max_{\boldsymbol{u}=\{\Delta P, T\}} \quad \dot{V}_p \kappa_p - 50\Delta P^2 - 5\Delta P(T_{\text{amb}} - T)$$

$$\text{s.t. nominal model,}$$

$$18 - \dot{V}_p \leq 0,$$

$$4 \leq \Delta P \leq 22,$$

$$20 \leq T \leq 30.$$

Gottu Mukkula, A.R., et al., Experimental Real Time Optimization of a Continuous Membrane Separation Plant
21st IFAC World Congress, July 12th-17th, 2020

MAWQA Experiment result without productivity constraint

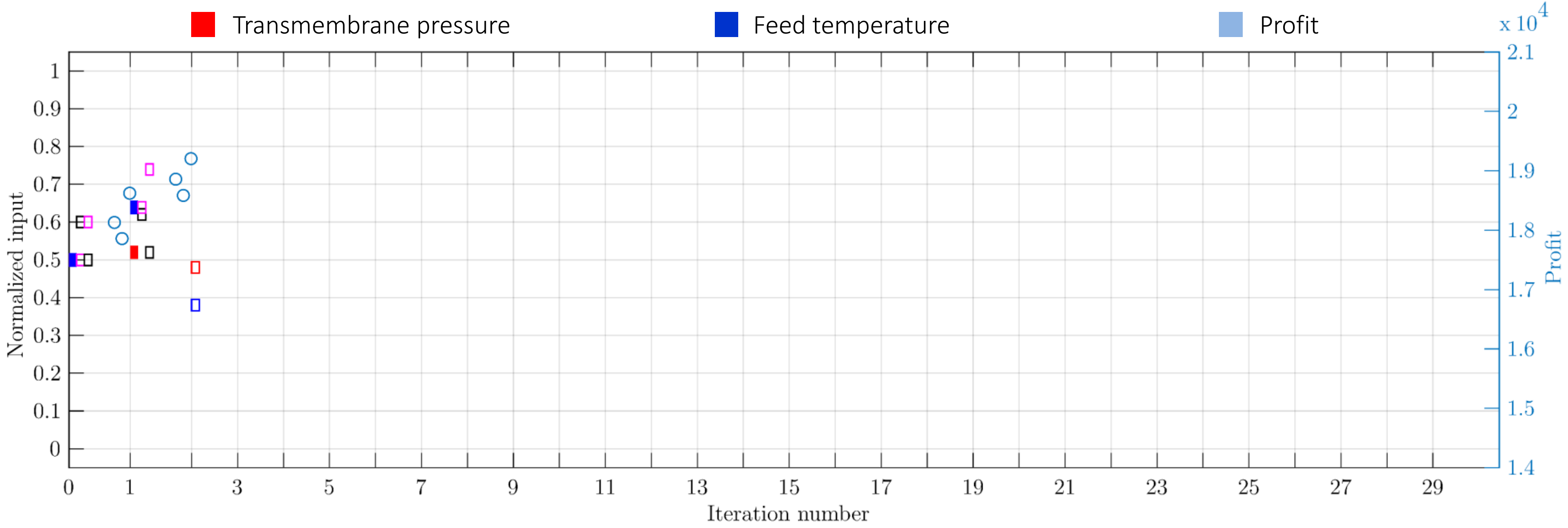


Figure: Experimental results for the unconstrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The tuning parameters γ, Δu, Δh and δ were set to 3.0, 0.1, 0.1 and 0.1, respectively.

MAWQA Experiment result without productivity constraint

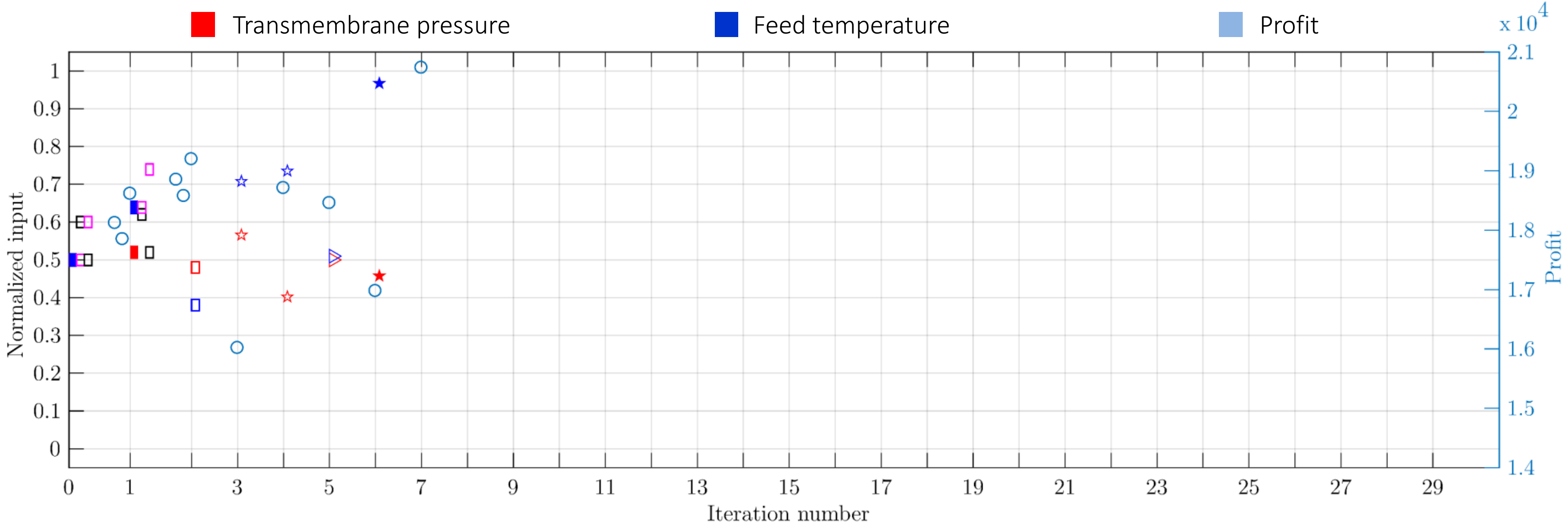


Figure: Experimental results for the unconstrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The tuning parameters γ, Δu, Δh and δ were set to 3.0, 0.1, 0.1 and 0.1, respectively.

MAWQA Experiment result without productivity constraint

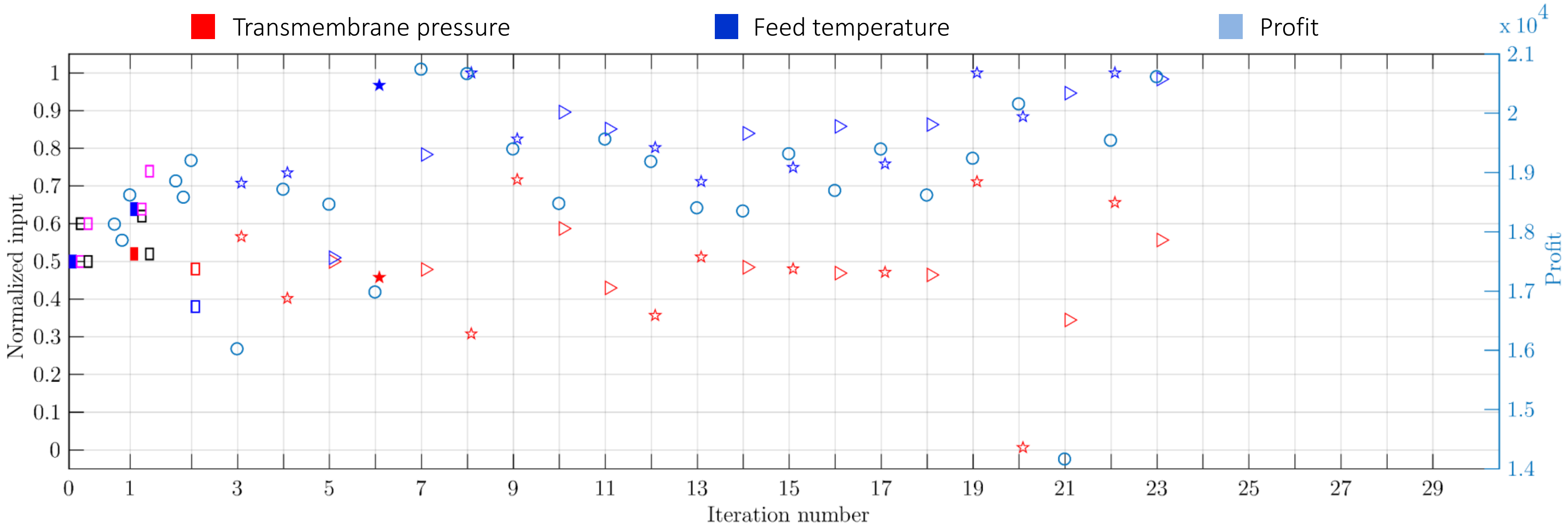


Figure: Experimental results for the unconstrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The tuning parameters γ, Δu, Δh and δ were set to 3.0, 0.1, 0.1 and 0.1, respectively.

MAWQA Experiment result without productivity constraint

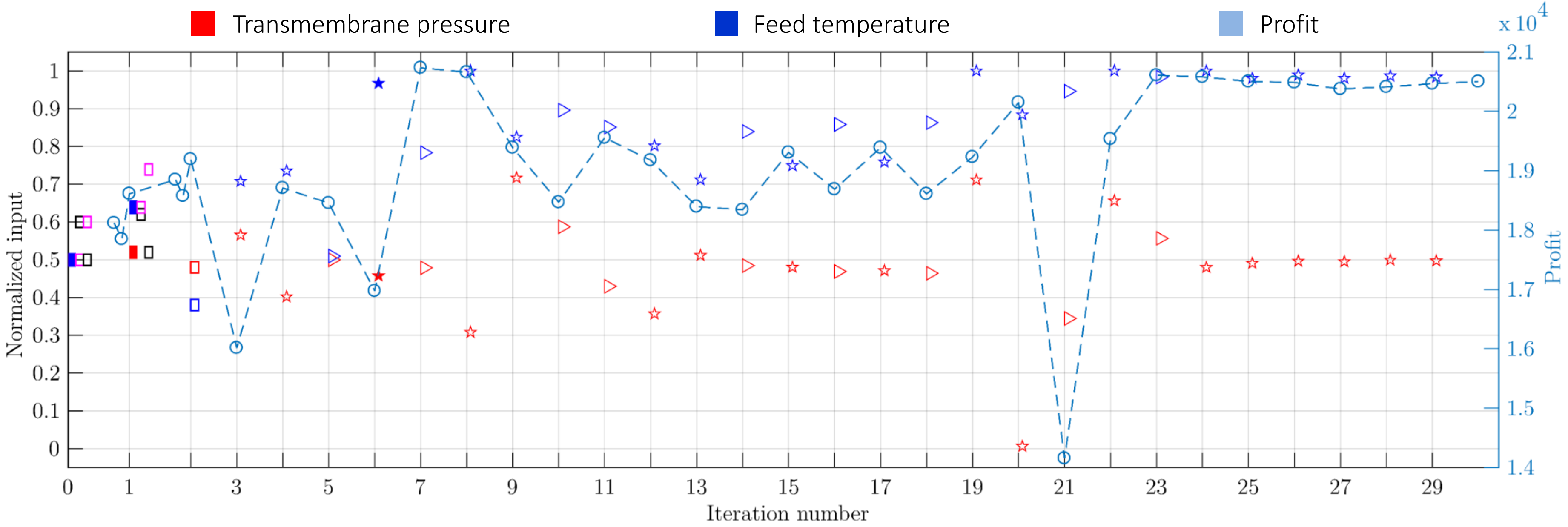


Figure: Experimental results for the unconstrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The tuning parameters γ, Δu, Δh and δ were set to 3.0, 0.1, 0.1 and 0.1, respectively.

MAWQA Experiment result with productivity constraint

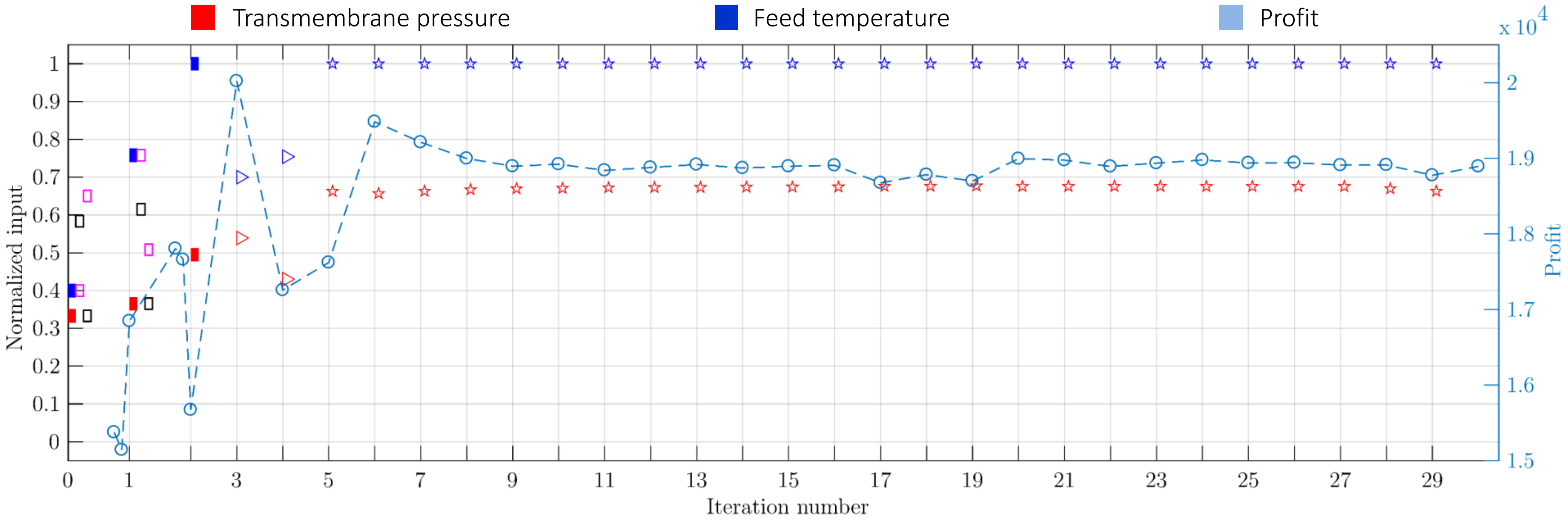
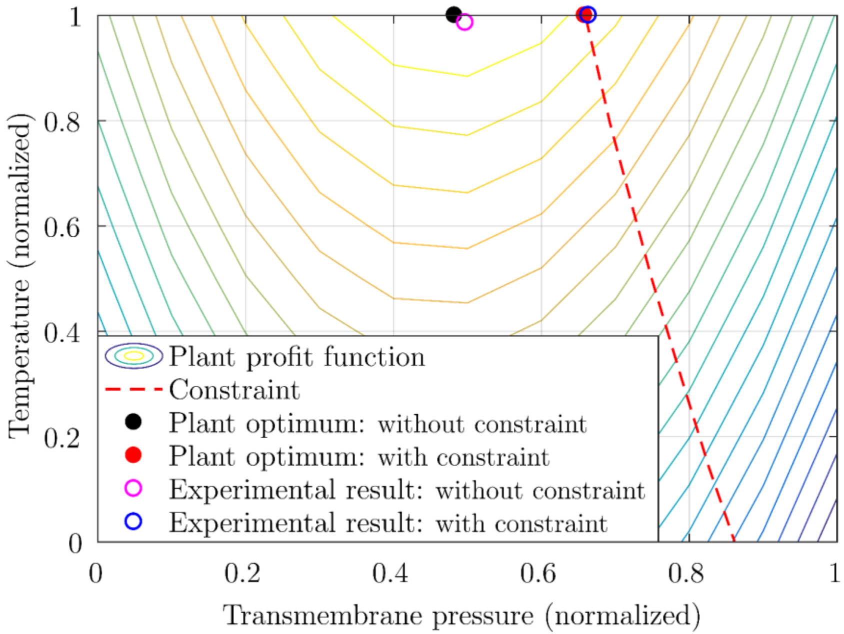


Figure: Experimental results for the constrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The values of the tuning parameters γ, Δu, Δh and δ used in the MAWQA scheme are set to 3.0, 0.25, 0.25 and 0.25.

Validation of experiment results



Top figure: Contour plot of the profit function of the nominal models and of the plant, its optimum and the optimum identified by the MAWQA experiments for both the constrained and the unconstrained cases.

Summary & Outlook

- We reported the development of an online real-time optimization solution (MAWQA) for the optimal operation of a continuously operated membrane plant with a nanofiltration membrane.
- Two experiments, one without constraints and another one with a productivity constraint were performed.
- In both experiments, the MAWQA scheme converged to an input close to the plant optimum that is predicted by the surrogate model for a large data set.
- The experiments validated that the combination of the plant measurements with MAWQA can drive a real plant to an optimal operation despite plant-model mismatch
- In our future work, we will focus on reducing the inefficient input moves and on developing a standardized approach for choosing the tuning parameters.

Thank you

anweshreddy.gottumukkula@tu-dortmund.de

The authors acknowledge the German Academic Exchange Service (DAAD) and The Ministry of Education, Science, Research and Sport of the Slovak Republic under the Exchange involving projects (PPP) project "Reliable and Real-time Feasible Estimation and Control of Chemical Plants". STU acknowledges the contribution of the Slovak Research and Development Agency (project APVV 15 – 0007) and of the European Commission (grant 790017).