Experimental Real Time Optimization of a Continuous Membrane Separation Plant [⋆]

Anwesh Reddy Gottu Mukkula [*] Petra Valiauga [**] Miroslav Fikar [**] Radoslav Paulen [**] Sebastian Engell [*]

[*] *Process Dynamics and Operations Group, Technische Universität Dortmund, Germany*
(E-mail: {anweshreddy.gottumukkula, sebastian.engell}@tu-dortmund.de)
[**] *Faculty of Chemical and Food Technology, Slovak University of Technology in Bratislava, Slovakia*
(E-mail: {petra.valiauga, miroslav.fikar, radoslav.paulen}@stuba.sk)

Abstract: This paper deals with the optimal operation of a continuously operated laboratory membrane separation plant. The goal is to find an economically optimal regime of operation using the transmembrane pressure (TMP) and the operating temperature as adjustable set-points for the low-level controllers. The main challenge is to identify the optimum in the absence of an accurate process model. We employ an iterative real-time optimization scheme, modifier adaptation with quadratic approximation (MAWQA), to identify the plant optimum in the presence of the plant-model mismatch and measurement noise. Two experiments are performed; one with and one without a productivity constraint. The experimental results show the capabilities of the MAWQA scheme to identify the process optimum in real-world scenarios. The optimum identified by the MAWQA scheme coincides with the optimum of a surrogate model that was built using a larger data set.

Keywords: Iterative real-time optimization, Modifier adaptation, Plant-model mismatch.

1. INTRODUCTION

Membrane separation processes are widely used in the pharmaceutical, food (e.g., dairy), and beverage industries because of their capability to separate, concentrate, clarify and purify solutions consisting of one or more solutes with different particle sizes dissolved in a solvent (Scott and Hughes, 2012). Nanofiltration is a membrane separation process in which a nanofilter membrane is used to retain solutes with a molecular mass ranging from $[200, 1000]$ g/mol. It is widely used in wastewater treatment, production of beverages, processing of vegetable oil, and for the concentration of lactose from cheese whey (Das et al., 2016), etc.

Membranes are prone to fouling, i.e., degradation of the membrane performance caused by the deposit of filtered particles that block the pores of the membrane (Zondervan and Roffel, 2007). The economics of a membrane process is thus strongly influenced by appropriate cleaning procedures, which strive to maximize the restoration of the membrane performance up to the level of a clean membrane. There are, however, also opportunities that can be exploited during the regular operation of a membrane process to improve the plant economics. This includes minimization of the energy required for the operation, mostly for pumping. The energy consumption is directly attributed to the permeate flow rates and, if minimized, it has a synergistic effect on the rate of fouling, thus improving the overall economics by prolonging the production periods before cleaning has to be performed.

The optimal operating regime of a continuous membrane process can be found by solving a model based optimization problem if an accurate plant model is available, similar to previous work on batch membrane separation processes (Paulen and Fikar, 2019; Sharma et al., 2019). Such a model however usually is difficult to obtain and one has typically two choices; to resort to empirical correlations based on data or to build a rigorous (dynamic) model. The latter involves a significant effort; from the selection of functional form of the model, through conducting possibly costly experiments and performing as-accurate-as-possible identification of all model parameters, to model validation. Such a rigorous modeling procedure necessitates to compromise accuracy of prediction and complexity of the model, and the model will not capture the evolution of the plant over time due to fouling and other phenomena. So plant-model mismatch will always exist and the optimum that was determined using the fitted model (data-based or rigorous) will differ from the true plant optimum.

Iterative real-time optimization techniques have gained popularity in recent years to identify the optimal operating condition of a process despite parametric and structural

[⋆] 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).

Copyright lies with the authors

plant-model mismatch. Initially, the two-step approach to handle parametric plant-model mismatch was proposed in Jang et al. (1987). In this approach, the model parameters are updated using the plant measurements, usually using least-squares estimation, then the adapted model based optimization problem is solved and this procedure is iterated. In Roberts (1979), the so-called integrated system optimization and parameter estimation (ISOPE) approach was proposed, where in addition to updating the model parameters the objective function of the model-based optimization problem is modified by adding bias and gradient correction terms which are also updated iteratively. The ISOPE scheme handles both structural and parametric plant-model mismatch and the inputs obtained from the ISOPE scheme iteratively converge to a KKT point of the real plant. To avoid the estimation of the model parameters in each iteration, in Tatjewski (2002), the redesigned-ISOPE scheme was proposed. It was shown (Tatjewski, 2002) that the uncertain parameters do not have to be updated in each iteration to converge to a KKT point of the plant. The redesigned-ISOPE scheme was extended to handle process-dependent constraints and termed iterative gradient-modification optimization (IGMO) in Gao and Engell (2005). This approach was further analyzed in Marchetti et al. (2009) and the name modifier adaptation (MA) was proposed. In addition to model-parameter adaptation and modifier adaptation methods, there are also direct input adaptation (e.g. extremum seeking control (ESC) (Krstić and Wang, 2000)), necessary conditions of optimality (NCO)-tracking and self optimizing control (SOC) (Jäschke and Skogestad, 2011) schemes to handle plant-model mismatch. A comprehensive survey of the existing iterative real-time optimization schemes can be found in Marchetti et al. (2016).

The ISOPE, IGMO and MA schemes require the knowledge of the true process gradients which are not trivial to obtain (Roberts, 2000). Finite differences are used to compute the process gradients in the ISOPE and in the redesigned-ISOPE schemes. In the presence of measurement noise, the gradients approximated using finite differences are, however, prone to errors. Additionally, there is the disadvantage of requiring additional plant perturbations around each input (Roberts, 2000). In Gao and Engell (2005), Broyden's formula is used to approximate the plant gradients which does not require additional plant perturbations. Additional trial points are added and optimized if the condition number of the data matrix renders the gradient computation unreliable (Gao and Engell, 2005). In Gao et al. (2016), modifier adaptation with quadratic approximation (MAWQA) was proposed as a combination of MA, quadratic approximation (QA) and elements from derivative free optimization (DFO) (Conn et al., 2009). It was demonstrated in Gao et al. (2016) that the MAWQA scheme is less vulnerable to the measurement noise and is capable of decreasing the gradient approximation error compared to the aforementioned methods.

Although there exists a rich literature on iterative real-time optimization methods, they are still not widely used in the industry. According to Darby et al. (2011), most successful applications of iterative real-time optimization are in ethylene plants. An application of dynamic RTO at an industrial polymerization process has been reported in Pontes et al. (2015). Recently, a successful implementation of MAWQA at a hydroformylation miniplant was reported in (Hernandez et al., 2018).

In this contribution, we address the application of MAWQA to a real lab-scale membrane separation process. First, an a priori surrogate model of the plant is built using process measurements that were obtained for inputs that are distributed over the admissible operating range. MAWQA is then applied to iteratively determine the optimal process inputs at the real plant using the surrogate model of the process and gradient modifiers. The optimum identified by MAWQA in the experiments matches the optimum that was subsequently identified using an a posteriori surrogate quadratic model that was built using a larger data set. So the iterative optimization scheme can speed up the development of an optimized process and moreover can adapt to changes of the plant.

The remainder of the paper is organized as follows: In Section 2, the iterative real-time optimization scheme MAWQA is presented. A description of the membrane separation process is presented in Section 3. The goal of the optimization and the experimental results are presented and discussed in Section 4 for the unconstrained case and for a case when a certain productivity level of the plant is required. Section 5 contains conclusions and recommendations for further work.

2. MODIFIER ADAPTATION WITH QUADRATIC APPROXIMATION

Consider a process for which a steady state mathematical model of the form (1b) has been built. The mapping function $\boldsymbol{F}_m : \mathbb{R}^{n_u} \to \mathbb{R}^{n_y}$ maps the input variables that are represented by the n_u-dimensional vector $\boldsymbol{u}$ to the n_y-dimensional vector of measured variables $\hat{\boldsymbol{y}}$. Let $J_m : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}$ be the function of input and measured variables that we want to minimize and $\boldsymbol{G}_m : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_c}$ be a vector of n_c inequality constraints that have to be respected. We assume that the functions J_m and G_m are at least twice differentiable with respect to $\boldsymbol{u}$.

Optimal inputs of the process between bounds $\boldsymbol{u}^L$ and $\boldsymbol{u}^U$ can be obtained by solving

$$\boldsymbol{u}_m^* = \arg \min_{\boldsymbol{u} \in [\boldsymbol{u}^L, \boldsymbol{u}^U]} J_m(\hat{\boldsymbol{y}}, \boldsymbol{u}) \tag{1a}$$

$$\text{s.t. } \hat{\boldsymbol{y}} = \boldsymbol{F}_m(\boldsymbol{u}), \tag{1b}$$

$$\boldsymbol{G}_m(\hat{\boldsymbol{y}}, \boldsymbol{u}) \leq \boldsymbol{0}. \tag{1c}$$

The identified optimum $\boldsymbol{u}_m^*$ however may differ considerably from the optimum $\boldsymbol{u}_p^*$ of the real process if $\boldsymbol{F}_m$ does not describe the real process (plant) with sufficient accuracy.

MAWQA is an iterative gradient-modification optimization scheme that contains elements from derivative free optimization (DFO) (Gao et al., 2015; Gao et al., 2016). In MAWQA, the plant-model mismatch is handled by modifying and iteratively updating the objective and the constraint functions of the model-based optimization problem (1) as in Gao and Engell (2005); Marchetti et al. (2009).

At the k^{th} iteration of the MAWQA scheme, the objective function is modified by adding a gradient correction term and the constraint function is modified by adding bias and gradient correction terms:

$$J_m^{ad,k} := J_m(\hat{\boldsymbol{y}}, \boldsymbol{u}) + \\ (\nabla J_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k) - \nabla J_m(\hat{\boldsymbol{y}}^k, \boldsymbol{u}^k))^T(\boldsymbol{u} - \boldsymbol{u}^k), \tag{2a}$$

$$\boldsymbol{G}_m^{ad,k} := \boldsymbol{G}_m(\hat{\boldsymbol{y}}, \boldsymbol{u}) + (\boldsymbol{G}_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k) - \boldsymbol{G}_m(\hat{\boldsymbol{y}}^k, \boldsymbol{u}^k)) + \\ (\nabla \boldsymbol{G}_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k) - \nabla \boldsymbol{G}_m(\hat{\boldsymbol{y}}^k, \boldsymbol{u}^k))^T(\boldsymbol{u} - \boldsymbol{u}^k), \tag{2b}$$

and the following optimization problem is solved:

$$\hat{\boldsymbol{u}}^{k+1} = \arg \min_{\boldsymbol{u} \in [\boldsymbol{u}^L, \boldsymbol{u}^U]} J_m^{ad,k} \tag{3a}$$

$$\text{s.t. } \hat{\boldsymbol{y}} = \boldsymbol{F}_m(\boldsymbol{u}), \tag{3b}$$

$$\boldsymbol{G}_m^{ad,k} \leq \boldsymbol{0}. \tag{3c}$$

Solving (3) determines a new input ($\boldsymbol{u}^{k+1}$) at the k^{th} iteration. The vectors $\nabla J_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k)$ and $\nabla J_m(\hat{\boldsymbol{y}}^k, \boldsymbol{u}^k)$ in (2) represent the gradient of the plant objective function $J_p(\boldsymbol{y}_p, \boldsymbol{u})$ and of the nominal-model objective function $J_m(\hat{\boldsymbol{y}}, \boldsymbol{u})$, with respect to the process inputs in the k^{th} iteration. Similarly, the matrices $\nabla \boldsymbol{G}_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k)$, $\nabla \boldsymbol{G}_m(\hat{\boldsymbol{y}}^k, \boldsymbol{u}^k)$ represent the gradients of the plant constraint functions $\boldsymbol{G}_p(\boldsymbol{y}_p, \boldsymbol{u})$ and the constraint functions of the nominal model $\boldsymbol{G}_m(\hat{\boldsymbol{y}}, \boldsymbol{u})$.

In MAWQA, $\nabla J_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k)$ and $\nabla \boldsymbol{G}_p(\boldsymbol{y}_p^k, \boldsymbol{u}^k)$ are computed from the available data using a surrogate quadratic model (Q). A general quadratic function can be expressed as

$$Q(\boldsymbol{p}, \boldsymbol{u}) = \sum_{i=1}^{n_u} \sum_{j=1}^{i} a_{i,j} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c, \tag{4}$$

where $\boldsymbol{p} := \{a_{1,1}, \ldots, a_{n_u,n_u}, b_1, \ldots, b_{n_u}, c\}$ is the vector of parameters of the quadratic function. To fit Q to the data, a minimum of $\frac{(n_u+1)(n_u+2)}{2}$ data points is required including the input point $\boldsymbol{u}^k$ at which the gradient is approximated. This is referred to as the cardinality condition. As long as the cardinality condition is not fulfilled, finite differences with a step length Δh or the past input moves obtained from IGMO are used to approximate the plant gradients. If the cardinality condition is satisfied, a subset of $\frac{(n_u+1)(n_u+2)}{2}$ input data points ($\mathcal{U}^k$) is selected from the set of all evaluated plant inputs up to the k^{th} iteration ($\mathbb{U}^k$). The data set $\mathcal{U}^k$ is used to approximate the plant objective function $J_p(\cdot)$ and each of the constraint functions $\boldsymbol{G}_p(\cdot)$ using quadratic functions. The plant gradients are obtained by analytically differentiating the fitted quadratic functions and evaluating them at $\boldsymbol{u}^k$.

The set $\mathcal{U}^k$ is identified by screening all the available data points in $\mathbb{U}^k$. A comparative study of screening algorithms was reported in Wenzel et al. (2017). In general, it is attempted to select $\mathcal{U}^k$ such that it consists of well distributed distant data points $\mathcal{U}_{dist}^k$ which act as anchor points and neighboring points $\mathcal{U}_{nb}^k$ which lie in the vicinity of $\boldsymbol{u}^k$. $\mathcal{U}_{nb}^k$ is defined by a tuning parameter $\boldsymbol{\Delta u}$. All input points in the set $\mathbb{U}^k$ lying in the inner circle, i.e. not farther than $\boldsymbol{\Delta u}$ from $\boldsymbol{u}^k$ are considered as neighboring points. The inverse of the condition number of $\boldsymbol{s}^k$, defined as

$$\boldsymbol{s}^k = [\boldsymbol{u}^k]^{n_u \times 1} \otimes 1^{\text{cardinality}(\mathcal{U}_{dist}^k) \times 1} - [\mathcal{U}_{dist}^k], \tag{5}$$

is used to assess the quality of the distribution of the data points (Gao et al., 2016). In (5), $[\boldsymbol{u}^k]$ and $[\mathcal{U}_{dist}^k]$ are matrix representations of $\boldsymbol{u}^k$ and of the set $\mathcal{U}_{dist}^k$. If the inverse of the condition number of $\boldsymbol{s}^k$ ($\kappa^{-1}(\boldsymbol{s}^k)$) is less than a desired value (δ), additional plant perturbations are performed and added to $\mathcal{U}^k$ to improve the distribution of the points. This condition is referred to as conditionality.

As the fitted quadratic functions are only local approximations of $J_p(\cdot)$ and $G_p(\cdot)$, they are valid only in the vicinity of $\boldsymbol{u}^k$. In MAWQA, this is taken into account by restricting the process input $\hat{\boldsymbol{u}}^{k+1}$ obtained by the optimization (3) to lie inside an ellipsoidal trust region by adding the following constraint to the modified optimization problem in (3):

$$(\boldsymbol{u} - \boldsymbol{u}^k)^T \text{cov}(\mathcal{U}^k)(\boldsymbol{u} - \boldsymbol{u}^k) \leq \gamma^2, \tag{6}$$

where γ is a tuning parameter which scales the size of the trust region (Gao et al., 2016).

In addition to QA, there are also other elements of DFO (Conn et al., 2009) present in MAWQA. This includes the criticality-check, the quality check and a possible switch to an optimization based on the quadratic approximation model.

The criticality-check is used to ensure that all the input points in $\mathcal{U}_{dist}^k$ lie not further than $2\boldsymbol{\Delta u}$ from $\boldsymbol{u}^k$ if the new iteration input $\boldsymbol{u}^{k+1}$ from the optimization problem (3) or the optimization problem based on the quadratic approximation model (8) is less than $\boldsymbol{\Delta u}$ apart from $\boldsymbol{u}^k$, where $\boldsymbol{\Delta u}$ is a tuning parameter. If any of the points used in fitting the quadratic function lie further than $2\boldsymbol{\Delta u}$ from the input point at which the gradient has to be approximated ($\boldsymbol{u}^k$), instead of applying the input from the optimization problem ($\boldsymbol{u}^{k+1}$) to the plant, the plant is probed with an input which replaces the farthest point from $\boldsymbol{u}^k$ in $\mathcal{U}_{dist}^k$. The algorithm for the criticality-check is shown in Algorithm 1.

In the quality-check procedure, the quality of $J_m^{ad,k}(\cdot)$ and $\boldsymbol{G}_m^{ad,k}(\cdot)$ is compared with the fitted quadratic models for the objective function $J_Q^k(\cdot)$ and the constraint functions $\boldsymbol{G}_Q^k(\cdot)$. The prediction accuracies of the adapted model ρ_m^k and of the fitted quadratic model ρ_Q^k are computed according to

$$\rho_m^k := \max \left\{ \begin{array}{r} \left| 1 - \dfrac{J_m^{ad,k} - J_m^{ad,k-1}}{J_p^k - J_p^{k-1}} \right|, \left| 1 - \dfrac{G_{m,1}^{ad,k} - G_{m,1}^{ad,k-1}}{G_{p,1}^k - G_{p,1}^{k-1}} \right|, \\ \ldots, \left| 1 - \dfrac{G_{m,n_c}^{ad,k} - G_{m,n_c}^{ad,k-1}}{G_{p,n_c}^k - G_{p,n_c}^{k-1}} \right| \end{array} \right\}, \tag{7a}$$

$$\rho_Q^k := \max \left\{ \begin{array}{r} \left| 1 - \dfrac{J_Q^k - J_Q^{k-1}}{J_p^k - J_p^{k-1}} \right|, \left| 1 - \dfrac{G_{Q,1}^k - G_{Q,1}^{k-1}}{G_{p,1}^k - G_{p,1}^{k-1}} \right|, \\ \ldots, \left| 1 - \dfrac{G_{Q,n_c}^k - G_{Q,n_c}^{k-1}}{G_{p,n_c}^k - G_{p,n_c}^{k-1}} \right| \end{array} \right\}. \tag{7b}$$

The minimum of ρ_m^k and ρ_Q^k in (7) determines the better among the adapted and the quadratic models (Gao et al., 2015). If the quality of the quadratic model is better than the adapted model, i.e. $\rho_Q^k < \rho_m^k$, instead of solving the optimization problem in (3) using the modified objective and constraint functions, an optimization problem based on the fitted quadratic approximation model is solved to determine the next input:

$$\boldsymbol{u}^{k+1} = \arg \min_{\boldsymbol{u}\in[\boldsymbol{u}^L,\boldsymbol{u}^U]} J_Q^k(\boldsymbol{p}^{k,J},\boldsymbol{u}) := Q(\boldsymbol{p}^{k,J},\boldsymbol{u}) \tag{8a}$$

$$\text{s.t. } \boldsymbol{G}_Q^k(\boldsymbol{p}^{k,\boldsymbol{G}},\boldsymbol{u}) := \boldsymbol{Q}(\boldsymbol{p}^{k,\boldsymbol{G}},\boldsymbol{u}) \leq \boldsymbol{0}, \tag{8b}$$

$$(\boldsymbol{u}-\boldsymbol{u}^k)^T \mathrm{cov}(\mathcal{U}^k)(\boldsymbol{u}-\boldsymbol{u}^k) \leq \gamma^2. \tag{8c}$$

A flowsheet representation of the MAWQA scheme is shown in Fig. 1. In this work, the termination criterion (see Fig. 1) is not used. Instead, the iterative optimization is run for a fixed number of iterations.

Algorithm 1 Criticality-check

```
Require: $\boldsymbol{u}^{k+1}, \boldsymbol{u}^k, \mathcal{U}^k_{dist}, \varDelta\boldsymbol{u}$
  if $||\boldsymbol{u}^{k+1}-\boldsymbol{u}^k||_2 < \varDelta\boldsymbol{u}$ then
    Solve
        $\boldsymbol{u}^* = \arg \max_{\boldsymbol{u}_i\in\mathcal{U}^k_{dist}} ||\boldsymbol{u}_i-\boldsymbol{u}^k||_2$
    while $||\boldsymbol{u}^*-\boldsymbol{u}^k||_2 > 2\varDelta\boldsymbol{u}$ do
      $\boldsymbol{u}^* := \frac{\boldsymbol{u}^*+\boldsymbol{u}^k}{2}$
      $\boldsymbol{u}^{k+1} := \boldsymbol{u}^*$
    end while
  end if
```

The performance of the MAWQA scheme is highly dependent on the tuning factors γ, finite-differences perturbation factor ($\varDelta h$), conditionality limit (δ) and the radius of the inner circle $\varDelta\boldsymbol{u}$ in MAWQA. $\varDelta\boldsymbol{u}$ influences the size of the trust-region and also plays a key role in the criticality-check step. γ scales the trust-region which restricts the input moves by the modifier-adaptation problem in (3) or by the optimization problem in (8) based on the quadratic approximation model and also has an indirect effect in the criticality-check step. The quadratic approximation using a regression set with small value of $\varDelta\boldsymbol{u}$ is more sensitive to process noise and will lead to more input moves (or iterations) which do not improve the plant profit. A large value for $\varDelta\boldsymbol{u}$ is less sensitive to noise, but leads to a loose fit which decreases the accuracy of the identified final optimum (Gao et al., 2016).

3. THE MEMBRANE SEPARATION PROCESS

A process flow diagram of the nanofiltration membrane process that is considered in this study is illustrated in Fig. 2. The nanofiltration membrane was manufactured by Synder Filtration, USA. It has an area of $0.465\,\mathrm{m}^2$ and a cut-off range of $300-500\,\mathrm{Da}$. The pump forces the feed from the feed tank through the nanofilter membrane at a certain flow rate. The feed through the nanofiltration membrane is split into the retentate stream and the permeate stream. The transmembrane pressure ($\varDelta P$) is an important operating parameter. It is defined as

$$\varDelta P = \frac{P_i + P_r}{2} - P_p, \tag{9}$$

where P_i is the pressure at the inlet of the membrane module, P_r is the pressure at the retentate side, and P_p is the pressure at the permeate side, which is atmospheric. A desired transmembrane pressure ($\varDelta P$) is maintained using a pressure controller, which restricts the flow rate of the retentate, building up the pressure on the retentate side of the membrane. The temperature of the feed in the feed tank can also be regulated. The conductivity (κ_p [$\mu\mathrm{S\,cm}^{-1}$]) and the volumetric flow rate ($\dot{V}_p$ [$\mathrm{L\,h}^{-1}$]) of the permeate are measured.

4. EXPERIMENTAL RESULTS

The feed to the membrane process is a solution of lactose monohydrate with molar mass $360.31\,\mathrm{g\,mol}^{-1}$ and sodium chloride (NaCl) with molar mass $58.44\,\mathrm{g\,mol}^{-1}$ in water. The concentration of lactose monohydrate and NaCl in the feed is $66.6\,\mathrm{g\,L}^{-1}$ and $0.68\,\mathrm{g\,L}^{-1}$, respectively. The feed is separated by the nanofiltration membrane into the (retentate) stream with the retained lactose monohydrate and NaCl, and the permeate stream that contains only NaCl and water. The goal is to remove as much salt as possible from the feed solution using the membrane plant. The transmembrane pressure ($\varDelta P$) and the feed inlet temperature (T) are the manipulated variables. The upper and lower limits of the manipulated variables are $[4, 22]$ bar and $[20, 30]$ °C for $\varDelta P$ and T. The concentration of NaCl in the permeate is inferred from κ_p. The objective function to be maximized is defined as

$$J = \dot{V}_p\kappa_p - 50\varDelta P^2 - 5\varDelta P(T_{\mathrm{amb}} - T), \tag{10}$$

where $T_{\mathrm{amb}} := 30\,°\mathrm{C}$ is the ambient temperature. The first term of the objective function rewards for having a high flow rate of permeate with high concentration of NaCl. The second and the third term of the objective function penalize operating at high transmembrane pressure ($\varDelta P$) and low feed temperature (T), which increase the production costs related to pumping and cooling energy.

To develop an a priori (nominal) model of the plant, the operating region was gridded using a coarse 6×4 grid with 24 points and the plant was operated with each combination of inputs in the grid. A nominal model was built by fitting a surrogate quadratic function to the measurements that were obtained with the probing inputs. The values of the plant profit function and the constraint function are computed for each probing input using the plant measurements. Later, a quadratic approximation function is fitted to the computed values of the profit and constraint functions to obtain their respective surrogate models. Figure 3 illustrates the contour plot of the profit function of the nominal model and its constraint.

In this study, we consider two cases. In the unconstrained case, the goal is to simply identify the operating input which maximizes the objective function. For the constrained case, the objective function is maximized such that the volumetric flow rate of the permeate is maintaned at least at $18\,\mathrm{L\,h}^{-1}$. The solution of the model-based optimization problem for both the constrained and the unconstrained cases using the nominal (surrogate) model is [$15.2\,\mathrm{bar}, 30\,°\mathrm{C}$] for [$\varDelta P, T$]. The normalized model optimum is $[0.62, 1]$ for [$\varDelta P, T$].

4.1 Case 1: Optimization without productivity constraint

Figure 4 shows the input values applied during the iterations of the MAWQA scheme and the corresponding evolution of the plant profit function evaluated using the plant measurements for the unconstrained case. The legend of the figure is provided in Table 1. The tuning parameters $\gamma, \varDelta\boldsymbol{u}, \varDelta h$ and δ were set to 3.0, 0.1, 0.1 and 0.1, respectively. The iterative optimization scheme was initialized at $\boldsymbol{u}^0 := [0.5, 0.5]$ (normalized) for [$\varDelta P, T$] in the 0th iteration.

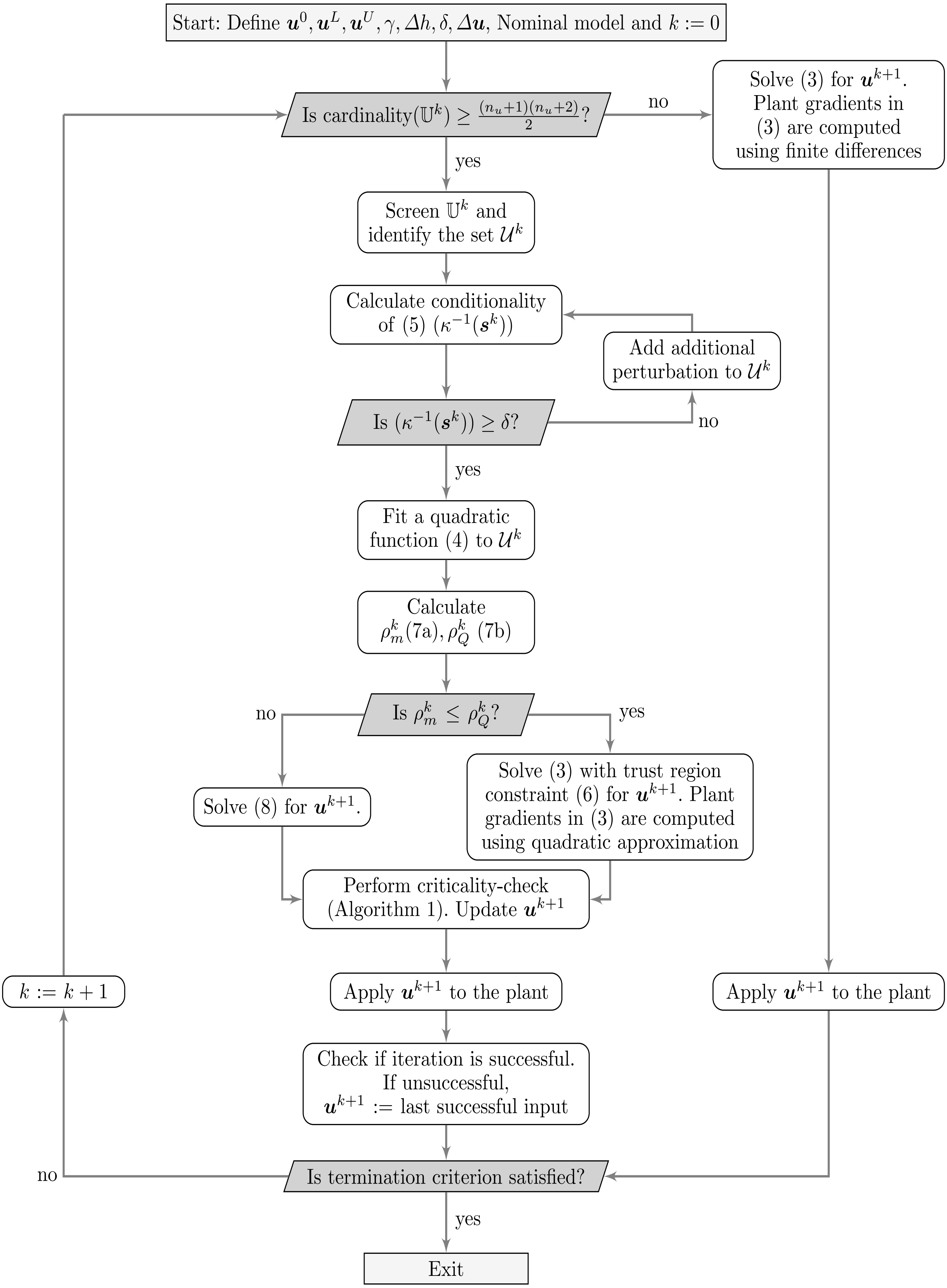
Start: Define $\boldsymbol{u}^0, \boldsymbol{u}^L, \boldsymbol{u}^U, \gamma, \Delta h, \delta, \Delta\boldsymbol{u}$, Nominal model and $k := 0$
Is cardinality$(\mathbb{U}^k) \geq \frac{(n_u+1)(n_u+2)}{2}$?
no
Solve (3) for $\boldsymbol{u}^{k+1}$. Plant gradients in (3) are computed using finite differences
yes
Screen $\mathbb{U}^k$ and identify the set $\mathcal{U}^k$
Calculate conditionality of (5) $(\kappa^{-1}(\boldsymbol{s}^k))$
Add additional perturbation to $\mathcal{U}^k$
Is $(\kappa^{-1}(\boldsymbol{s}^k)) \geq \delta$?
no
yes
Fit a quadratic function (4) to $\mathcal{U}^k$
Calculate ρ_m^k(7a), ρ_Q^k (7b)
no
Is $\rho_m^k \leq \rho_Q^k$?
yes
Solve (8) for $\boldsymbol{u}^{k+1}$.
Solve (3) with trust region constraint (6) for $\boldsymbol{u}^{k+1}$. Plant gradients in (3) are computed using quadratic approximation
Perform criticality-check (Algorithm 1). Update $\boldsymbol{u}^{k+1}$
$k := k + 1$
Apply $\boldsymbol{u}^{k+1}$ to the plant
Apply $\boldsymbol{u}^{k+1}$ to the plant
Check if iteration is successful. If unsuccessful, $\boldsymbol{u}^{k+1}$:= last successful input
no
Is termination criterion satisfied?
yes
Exit

Fig. 1. Flowsheet representation of the modifier adaptation with quadratic approximation (MAWQA) scheme.

Table 1. Legend for Figs. 4, 5. Input markers in red and black color refer to the transmembrane pressure; Blue and magenta input markers refer to the inlet temperature.

Marker	Description	Marker	Description
[▮, ▮]	Successful-iteration input from model-based optimization	[▯, ▯]	Perturbation input
[★, ★]	Successful-iteration input from optimization using quadratic model	-○-	Evolution of the profit function
[☆, ☆]	Explorative-iteration input from optimization using quadratic model		
[▶, ▶]	Input due to criticality-check, which improved the value of profit function		
[▷, ▷]	Input due to criticality-check, which did not improve the value of profit function		

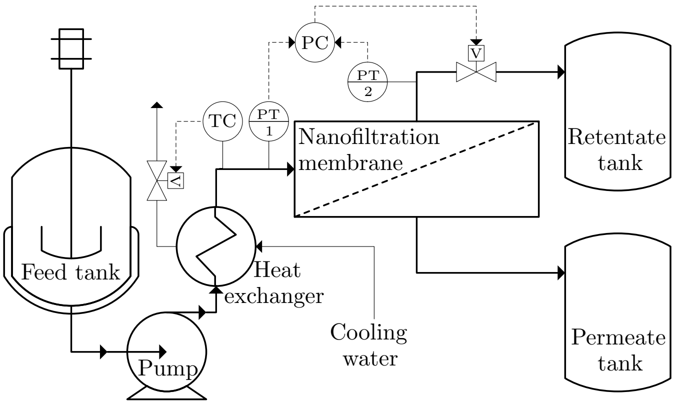

Fig. 2. Illustration of the process flow diagram of the continuously operated membrane separation process.

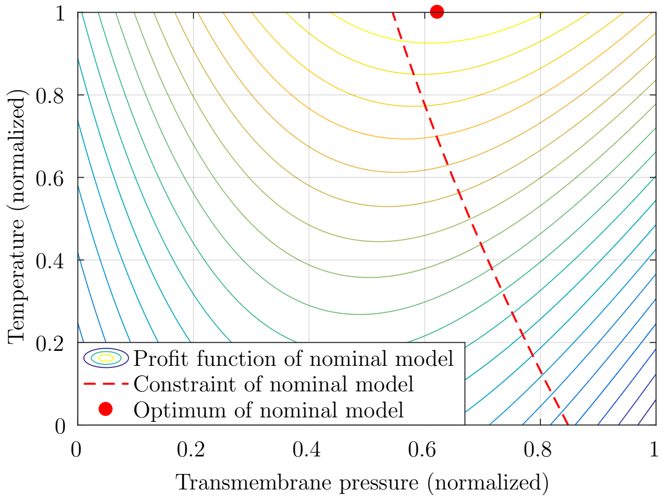

Fig. 3. Contour plot of the profit function of the nominal model, constraint and its optimum for both the constrained and the unconstrained cases.

In order to estimate the plant gradients using quadratic approximation, steady-state plant measurements are required for at least 6 different input values (since $n_u = 2$) to satisfy the cardinality condition. As the cardinality condition fails in the beginning (0th iteration), finite differences are used to estimate the plant gradients. Therefore two additional perturbations ([▯, ▯]), one for each input with a step length of $\Delta h := 0.1$ are performed after applying the input $\boldsymbol{u}^0$ ([▮, ▮]) to the plant. As it takes some time (approximately 15 min) for the plant to reach a steady state, the value of the plant profit function (○) evaluated using the obtained steady-state plant measurements is shown in the figure with a delay. Once the plant measurements for all applied inputs are available, the modifiers are computed and the modifier adaptation problem (3) is solved to compute the next iteration input, i.e. the 1st iteration input ($\boldsymbol{u}^1$). As the cardinality condition fails also in the 1st iteration, the iteration input for the 2nd iteration ($\boldsymbol{u}^2$) is computed in the same manner as $\boldsymbol{u}^1$. From the 2nd iteration on, the cardinality condition is satisfied. Therefore, in each iteration a quadratic approximation model is used to approximate the plant gradients. In order to identify the steady-state data points that are used for fitting a quadratic function, a screening algorithm is used to identify the set $\mathcal{U}^2$ from all the available data points in the 2nd iteration ($\mathbb{U}^2$). Later, the quality check is performed by comparing the quality of the modified nominal model and of the fitted quadratic model. In the quality check step of the second iteration $\rho_m^2 < \rho_Q^2$. Therefore the optimization problem based on the quadratic model (8) is solved and the obtained input $\boldsymbol{u}^3$ is checked for the satisfaction of the criticality-check criterion in Algorithm 1. As $||\boldsymbol{u}^3 - \boldsymbol{u}^2||_2 > \Delta\boldsymbol{u}$, $\boldsymbol{u}^3$ is not updated in the criticality-check step and is applied directly to the plant. As $\boldsymbol{u}^3$ does not improve the plant profit, it is considered as an explorative move and thus represented by [☆, ☆]. The iteration input obtained from the 4th iteration, i.e. $\boldsymbol{u}^5$, does not satisfy the criticality-check criterion; therefore, it is updated in the criticality-check step. As the updated input from the criticality-check does not improve the plant profit, it is marked by [▷, ▷].

In the 5th iteration, $\mathcal{U}^5$ satisfies the conditionality condition and the input $\boldsymbol{u}^6$ is computed by solving (8) since $\rho_m^6 < \rho_Q^6$. As $\boldsymbol{u}^6$ satisfies the criticality-check criterion, it is not updated and is applied to the plant. The iteration input $\boldsymbol{u}^6$ improves the plant profit; therefore, it is considered as a successful iteration and is represented by [★, ★]. The inputs from $\boldsymbol{u}^7$ to $\boldsymbol{u}^{23}$ are oscillating. The oscillations of the inputs are caused by the input update in the criticality-check algorithm since the data points in $\mathcal{U}$ are farther away than $2\Delta\boldsymbol{u}$ from $\boldsymbol{u}^k$. From the 24th iteration onwards the criticality-check criterion is always satisfied and the inputs from the MAWQA iterations converge to $[0.5, 0.99]$ (normalized). Although the input from the 5th iteration, i.e., $\boldsymbol{u}^6$ yielded the highest profit, MAWQA did not converge precisely to $\boldsymbol{u}^6$. This can be attributed to measurement and approximation errors. Nevertheless, MAWQA converged to an input $[0.5, 0.99]$ which is in close vicinity of $\boldsymbol{u}^6$.

4.2 Case 2: Optimization with productivity constraints

In this case, the volumetric flow rate of the permeate stream is restricted to be at least 18 lh^{-1}. Figure 5 shows all the input moves made by the MAWQA scheme and the evolution of the plant profit function computed using the

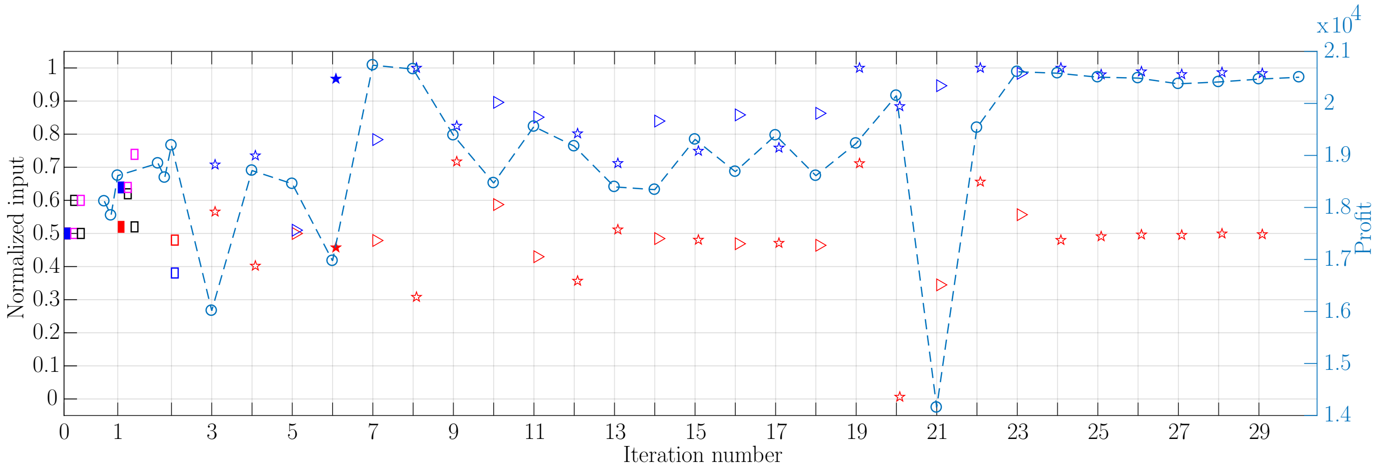

Fig. 4. Experimental results for the unconstrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The description of the markers used can be found in Table 1.

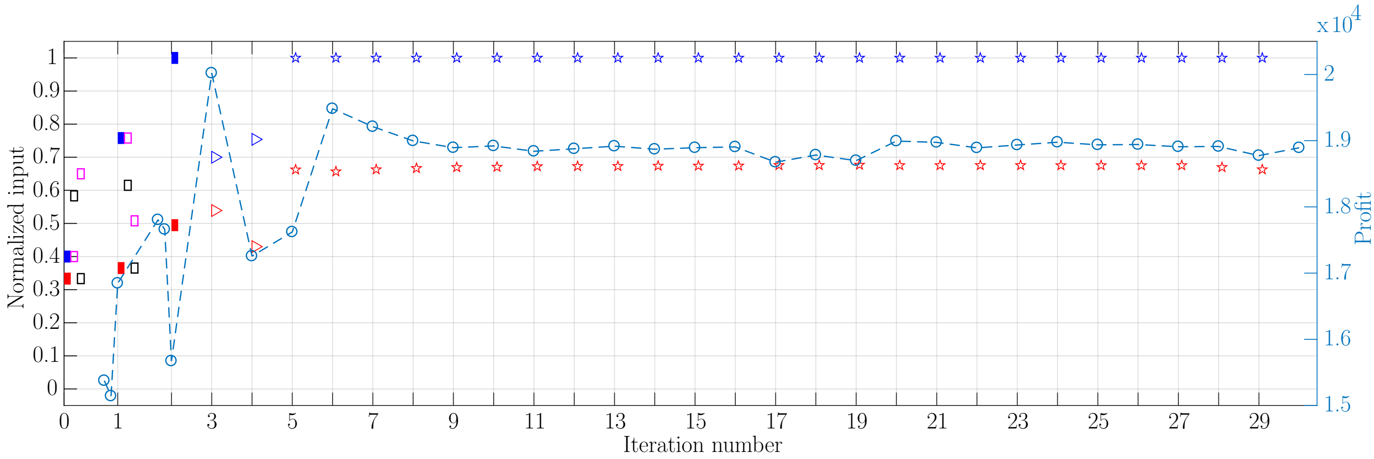

Fig. 5. Experimental results for the constrained case: Evolution of the inputs (normalized) transmembrane pressure, temperature and the profit function obtained using process measurements. The description of the markers used is provided in Table 1.

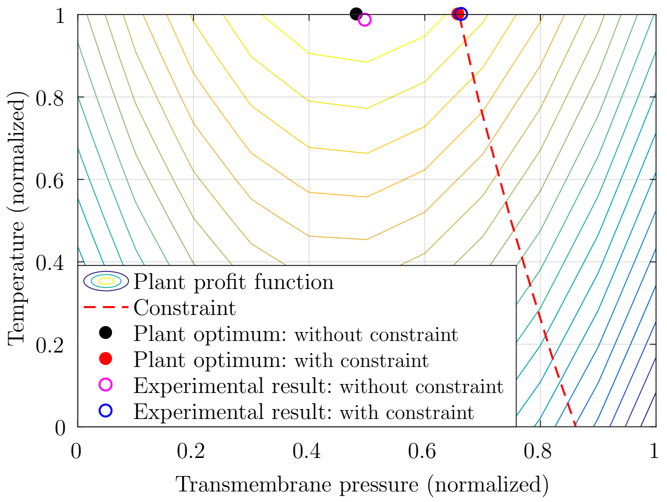

Fig. 6. Contour plot of the plant profit function, its optimum and the optimum identified by the MAWQA experiments for both the constrained and the unconstrained cases.

plant measurements for the constrained case. The values of the tuning parameters $\gamma, \Delta\boldsymbol{u}, \Delta h$ and δ used in the MAWQA scheme are set to 3.0, 0.25, 0.25 and 0.25. A higher value for $\Delta\boldsymbol{u}, \Delta h$ and δ is chosen to increase the speed of convergence. The iterative optimization scheme is initialized at $\boldsymbol{u}^0 := [0.33, 0.4]$ (normalized) for $[\Delta P, T]$ in the beginning (0^{th} iteration). Similar to the unconstrained case, during the initial iterations of the constrained case, finite differences with step length $\Delta h := 0.25$ are used for plant gradient approximation. Although the input from the 1^{st} iteration, i.e., $\boldsymbol{u}^2$ improved the plant profit significantly, it violates the productivity constraint (see Fig 6). Unlike in the unconstrained case, the iteration input is updated in the criticality-check step in only two iterations, i.e., in $\boldsymbol{u}^3$ and $\boldsymbol{u}^4$. Also in this case, $\rho_m < \rho_Q$ for all iterations from the 3^{rd} iteration. Therefore, from the 3^{rd} iteration onwards, the quadratic model based optimization problem (8) is solved to compute the iteration inputs. Due to the choice of Δh and Δu, the trust region (6) is larger than in the unconstrained case, therefore the iteration inputs converged in fewer iterations to $[0.67, 1]$ (normalized).

In order to verify the experimental results, a surrogate plant model was built using the plant measurements from a larger (a posteriori) data set containing 237 combinations of inputs. The values of the plant profit function and the constraint function are computed for each input using the plant measurements from the a posteriori data set. Later, a quadratic approximation function is fitted to the computed values of the profit and constraint functions to obtain surrogate models. Figure 6 shows the constraint and the normalized plant optimum for the constrained and the unconstrained cases. For the unconstrained case, the MAWQA scheme converged to $[0.5, 0.99]$ (normalized) close to the normalized plant optimum ($[0.48, 1]$). The region to the right of the constraint is feasible. In the constrained case, the MAWQA scheme converged to a $[0.67, 1]$ (normalized) which is close to the normalized plant optimum ($[0.66, 1]$) and lies in the feasible operating region. The MAWQA results were obtained experimentally, the deviations reflect the fluctuations in the behaviour of the real plant.

5. CONCLUSION

In this contribution, the operation of a continuously operated membrane plant with a nanofiltration membrane is considered. A novel iterative optimization scheme, MAWQA, which uses a plant model in combination with measured data, was chosen for the iterative online optimization of the membrane plant. A surrogate nominal model with measurement data from a coarse grid and a surrogate plant model with measurements from a large historical data set were built. 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. In the unconstrained case, the algorithm needed more than 20 iterations to converge to the plant optimum due to the criticality-check. There obviously is a potential for improvement in this element of the algorithm. The experiments validated that the combination of the plant measurements with the iterative optimization algorithm MAWQA can drive a real plant to an optimal operation, despite deviations between the nominal model and the true plant behavior, without the need for building a highly accurate model. In our future work, we will focus on reducing the inefficient input moves and on developing a standardized approach for choosing the tuning parameters.

REFERENCES

Conn, A.R., Scheinberg, K., and Vicente, L.N. (2009). *Introduction to derivative-free optimization.* SIAM.

Darby, M.L., Nikolaou, M., Jones, J., and Nicholson, D. (2011). Rto: An overview and assessment of current practice. *Journal of Process Control*, 21(6), 874–884.

Das, B., Sarkar, S., Sarkar, A., Bhattacharjee, S., and Bhattacharjee, C. (2016). Recovery of whey proteins and lactose from dairy waste: A step towards green waste management. *Process Safety and Environmental Protection*, 101, 27–33.

Gao, W., Wenzel, S., and Engell, S. (2015). Integration of gradient adaptation and quadratic approximation in real-time optimization. In *2015 34th Chinese Control Conference (CCC)*, 2780–2785.

Gao, W. and Engell, S. (2005). Iterative set-point optimization of batch chromatography. *Computers & Chemical Engineering*, 29(6), 1401–1409.

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

Hernandez, R., Dreimann, J., Vorholt, A., Behr, A., and Engell, S. (2018). Iterative real-time optimization scheme for optimal operation of chemical processes under uncertainty: Proof of concept in a miniplant. *Industrial & Engineering Chemistry Research*, 57(26), 8750–8770.

Jang, S.S., Joseph, B., and Mukai, H. (1987). On-line optimization of constrained multivariable chemical processes. *AIChE Journal*, 33(1), 26–35.

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

Krstić, M. and Wang, H.H. (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4), 595–601.

Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *Industrial & engineering chemistry research*, 48(13), 6022–6033.

Marchetti, A.G., François, G., Faulwasser, T., and Bonvin, D. (2016). Modifier adaptation for real-time optimization–methods and applications. *Processes*, 4(4), 55.

Paulen, R. and Fikar, M. (2019). Dynamic real-time optimization of batch processes using pontryagin's minimum principle and set-membership adaptation. *Computers & Chemical Engineering*, 128, 488–495.

Pontes, K.V., Wolf, I.J., Embiruçu, M., and Marquardt, W. (2015). Dynamic real-time optimization of industrial polymerization processes with fast dynamics. *Industrial & Engineering Chemistry Research*, 54(47), 11881–11893.

Roberts, P. (1979). An algorithm for steady-state system optimization and parameter estimation. *International Journal of Systems Science*, 10(7), 719–734.

Roberts, P. (2000). Broyden derivative approximation in isope optimising and optimal control algorithms. *IFAC Proceedings Volumes*, 33(16), 293–298.

Scott, K. and Hughes, R. (2012). *Industrial membrane separation technology.* Springer Science & Business Media.

Sharma, A., Valo, R., Kalúz, M., Paulen, R., and Fikar, M. (2019). Implementation of optimal strategy to economically improve batch membrane separation. *Journal of Process Control*, 76, 155–164.

Tatjewski, P. (2002). Iterative optimizing set-point control–the basic principle redesigned. *IFAC Proceedings Volumes*, 35(1), 49–54.

Wenzel, S., Yfantis, V., and Gao, W. (2017). Comparison of regression data selection strategies for quadratic approximation in RTO. In *Computer Aided Chemical Engineering*, volume 40, 1711–1716.

Zondervan, E. and Roffel, B. (2007). Evaluation of different cleaning agents used for cleaning ultra filtration membranes fouled by surface water. *Journal of Membrane Science*, 304(1), 40–49.