Handling Measurement Delay in Iterative Real-Time Optimization Methods

Anwesh Reddy Gottu Mukkula * and Sebastian Engell

Process Dynamics and Operations Group, Technische Universität Dortmund, 44221 Dortmund, Germany; sebastian.engell@tu-dortmund.de
* Correspondence: anweshreddy.gottumukkula@tu-dortmund.de

Abstract: This paper is concerned with the real-time optimization (RTO) of chemical plants, i.e., the optimization of the steady-state operating points during operation, based on inaccurate models. Specifically, modifier adaptation is employed to cope with the plant-model mismatch, which corrects the plant model and the constraint functions by bias and gradient correction terms that are computed from measured variables at the steady-states of the plant. This implies that the sampling time of the iterative RTO scheme is lower-bounded by the time to reach a new steady-state after the previously computed inputs were applied. If analytical process measurements (PAT technology) are used to obtain the steady-state responses, time delays occur due to the measurement delay of the PAT device and due to the transportation delay if the samples are transported to the instrument via pipes. This situation is quite common because the PAT devices can often only be installed at a certain distance from the measurement location. The presence of these time delays slows down the iterative real-time optimization, as the time from the application of a new set of inputs to receiving the steady-state information increases further. In this paper, a proactive perturbation scheme is proposed to efficiently utilize the idle time by intelligently scheduling the process inputs taking into account the time delays to obtain the steady-state process measurements. The performance of the proposed proactive perturbation scheme is demonstrated for two examples, the Williams–Otto reactor benchmark and a lithiation process. The simulation results show that the proposed proactive perturbation scheme can speed up the convergence to the true plant optimum significantly.

Keywords: iterative optimization; real-time optimization; modifier adaptation; plant-model mismatch; time delay; active perturbation strategy; quadratic approximation; gradient estimation; guaranteed model adequacy

1. Introduction

There is a strong interest in the process industries to identify the best steady-state operating conditions of their processes such that they are operated at their economic optimum while meeting constraints on product qualities, emissions, and equipment capabilities. Optimal operating conditions of a process can be identified by formulating and solving an optimization problem where a cost (or profit) function is minimized (or maximized) respecting all process limitations on the basis of a mathematical model of the plant. As the parameters of the optimization problem (e.g., prices for raw materials, energy, and products) and the behavior of the plant vary over time, such optimization is performed repeatedly during plant operation, which is called RTO [1,2] (real-time optimization). At each call of the optimizer, the problem at hand is a constrained nonlinear program (NLP) which usually is solved by gradient-based algorithms, e.g., SQP solvers [3,4] or interior point algorithms [5,6]. These solvers compute local optima, which is sufficient under the tacit assumption that only minor adaptations are made and the problem is convex in the region of interest. As far as the model-based optimization is concerned, in principle also other optimization methods can be applied, as e.g., derivative free methods [7,8].
population-based methods [9], or nature-inspired algorithms [10]. While for the solution of the optimization problems efficient algorithms are available, a major practical problem is the mismatch between the model and the true behavior of the plant. No matter how the optimum is computed, if the model and the behavior of the plant do not match, the computed optimum won’t be the true optimum for the plant and the solution may even violate the constraints of the plant. Of course the model could be discarded completely and a brute force optimization could be performed using any derivative-free or meta-heuristic algorithm, but this implies performing a large number of exploratory moves with the real plant, which is highly undesirable. In particular, large constraint violations of internal variables must be avoided, which is impossible if the optimization is completely model-free. Therefore, a combination of model-based and data-based optimization is advantageous. The classical approach to this problem is to add a parameter estimation element, called the two-stage approach [11].

In the two-step approach, some model parameters are updated repeatedly using least-squares estimation and the optimization problem with updated model parameters is solved to identify the process optimum. The performance of the two-step approach is discussed in detail in Chen and Joseph [12]. The two-step approach is designed to handle parametric plant-model mismatch, which means that the model can, with the correct parameter values, represent the true plant exactly. However, there always is also structural plant-model mismatch, i.e., even with optimized parameters, the predictions of the model are inaccurate. This issue was addressed in the “Integrated system optimization and parameter estimation” (ISOPE) approach proposed in Roberts [13]. In ISOPE scheme, in addition to updating the model parameters in each iteration, the objective function of the model based optimization problem is modified iteratively by adding bias and gradient correction terms so that the solution converges iteratively to the optimum of the real plant. These correction terms (also called modifiers) are estimated from the available measurements.

Later, by Tatjewski [14], it was shown that the model parameters do not have to be updated in each iteration to converge to a process optimum. Based on this insight, Tatjewski [14] simplified the ISOPE approach and proposed the redesigned-ISOPE scheme without the parameter-update step from the ISOPE approach. The redesigned-ISOPE scheme was extended in Gao and Engell [15] to handle process-dependent constraints, by adding bias and gradient correction terms also to the constraint functions, and was termed iterative gradient modification optimization (IGMO). The IGMO scheme in Gao and Engell [15] was analyzed in detail in Marchetti et al. [16] and there was given the name modifier adaptation (MA).

The efficiency of MA-based iterative RTO methods depends on the accuracy of the gradient modifiers, which contain the gradients of the response of the cost function and of the constraints to the inputs. As the plant gradients usually cannot be measured directly, they have to be computed from the available measurements. The traditional way of doing this is to use finite differences, preferably employing the past input moves as in Gao and Engell [15] where this was enhanced with probing moves if the gradient estimation problem becomes ill-conditioned. However, in finite differences-based schemes, a trade-off between the approximation error and the effect of measurement noise has to be found. Therefore, Gao et al. [17] proposed to use quadratic approximation (QA) for the approximation of the plant gradients, which is more robust at measuring noise compared to that of finite differences (FD) and Broyden’s formula [18]. The resulting modifier adaptation with quadratic approximation (MAWQA) scheme combines elements of MA, QA, and derivative free optimization [7] (DFO).

For all adaptation-based iterative RTO methods, it is a prerequisite that the adapted process model is adequate at the—unknown—optimum of the real plant [1,16,19–21]. A model is considered adequate if the Lagrangian of the model-based optimization problem satisfies the necessary second-order conditions of optimality at the optimum of the real process. In Faulwasser and Bonvin [22], it was proposed to use second-order correction terms (modifiers) in the objective and constraint functions to match the Hessian of the nominal model to that of the plant. However, it is difficult to estimate the plant Hessian [23] from
real data. Model adequacy was addressed in François and Bonvin [24] by using convex model approximation of the objective and constraint functions. This may however slow down the rate of convergence to the process optimum. Ahmad et al. [25] addressed model adequacy by updating some model parameters only if the model adequacy conditions are satisfied at the current operating input. In Gottu Mukkula and Engell [26], a guaranteed model adequacy (GMA) scheme for MAWQA was proposed where model adequacy is ensured by strictly convex quadratic approximations of the objective and constraint functions if the goal of the optimization is to minimize an objective function. The MAWQA with GMA scheme may converge to the process optimum in fewer iterations than MAWQA [26].

No matter how the optimization model is corrected, by parameter adaptation or by modifier adaptation of by a combination of both [23], the solution of the resulting model-based optimization problem can be performed by any algorithm. In our work, we use gradient-based algorithms, as the cost functions and the constraints are smooth, no discrete variables are involved, and guaranteed (local) optimality is important.

Modifier adaptation-based iterative RTO methods require steady-state process measurements to compute a new input that provides a lower limit to the time between iterates. This can be alleviated by the use of transient process measurements [27–31], but this is outside the scope of this paper. In addition to the time required to reach a steady-state, the convergence to a process optimum may also be slowed down due to delays in obtaining measurement information. Considerable measurement delays may occur due to the time required for the measurement device (e.g., gas chromatography) to analyze the sample and due to the remote positioning of a measurement device. Figure 1 illustrates a situation when the time delay is caused by the remote positioning of the measurement device for safety reasons, e.g., due to harsh conditions at the plant. Measurement devices like NMR have to be placed in a certified (e.g., ATEX, IECEx) enclosure located at a distance from the process equipment. The thin, long tubings that carry the sample from plant to the remotely positioned measurement device can cause a significant delay, and this is quite common in the process industries.

![Figure 1. Illustration of a general process with a measurement device at a remote location.](image)

Gottu Mukkula et al. [32,33] proposed two schemes where additional plant perturbations are performed during the waiting period. The purpose was to gain additional information about the process instead of remaining idle until the effect of a new operating input propagated through the measurement device setup. Here, we propose a strategy in the context of MA that proactively schedules the input moves and thereby significantly reduces the effect of the time delay caused due to the remote positioning of a measurement device. The input perturbations in the proposed method are computed by solving an optimization problem using the latest measurement information.
The rest of the paper is organized as follows. Firstly, the MA scheme used in this paper is presented in detail. Then, the previously proposed approaches to handle the time delay due to the positioning of a measurement device are presented. Thereafter, the new proactive perturbation scheme to effectively handle the time delay is described. Finally, the performance of the proposed scheme is analyzed using the Williams–Otto reactor benchmark and a lithiation reaction case study.

2. Preliminaries

2.1. Model

Consider the steady-state mathematical models \( y = F_p(u) \) and \( \hat{y} = F_m(u) \) as the exact representation of a process and as the nominal model that was built to represent the underlying process. The mapping functions \( F_p : \mathbb{R}^{n_u} \to \mathbb{R}^{n_y} \) and \( F_m : \mathbb{R}^{n_u} \to \mathbb{R}^{n_y} \) map the \( n_u \)-dimensional vector of manipulated variables \( u \) to the \( n_y \)-dimensional vector of measured variables \( y \) and \( \hat{y} \). Let \( J : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R} \) represent the objective function, which is continuous and twice differentiable with respect to \( u \), that should be minimized, and let \( G : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_c} \) be a continuous and twice differentiable with respect to \( u \) \( n_c \)-dimensional vector of constraint functions, which may include process, safety, and quality restrictions.

An optimal input of the process lying within \( u^L \) and \( u^U \) can be obtained theoretically by solving

\[
\begin{align}
  u^*_p &= \arg \min_{u \in [u^L, u^U]} J(y, u), \quad (1a) \\
  \text{s.t. } y &= F_p(u), \quad (1b) \\
  G(y, u) &\leq 0, \quad (1c)
\end{align}
\]

but the mapping function (1b) is not available. The optimum \( u^*_m \) results from solving the optimization problem using the nominal model:

\[
\begin{align}
  u^*_m &= \arg \min_{u \in [u^L, u^U]} J(\hat{y}, u), \quad (2a) \\
  \text{s.t. } \hat{y} &= F_m(u), \quad (2b) \\
  G(\hat{y}, u) &\leq 0, \quad (2c)
\end{align}
\]

and may differ from \( u^*_p \) considerably if \( F_m \neq F_p \). To simplify the notation, we from here on replace \( J(y, u), G(y, u) \) by \( J_p(u), G_p(u) \) and \( J(\hat{y}, u), G(\hat{y}, u) \) by \( J_m(u), G_m(u) \).

2.2. Modifier Adaptation

Modifier adaptation handles plant-model mismatch by adding and iteratively updating gradient correction term to the objective function, and by adding and iteratively updating bias and gradient correction terms to the constraint functions of the model based optimization problem (2) as described in Gao and Engell [15], Marchetti et al. [16]. In the \( k \)th iteration, the modified objective and constraint functions of the model based optimization problem, i.e., the MA problem, are

\[
\begin{align}
  j_{m, k}^{ad}(u) &= j_m(u) + (\nabla J_p^k - \nabla J_m^k)^T(u - u^k), \quad (3a) \\
  G_{m, k}^{ad}(u) &= G_m(u) + (G_p^k - G_m^k) + (\nabla G_p^k - \nabla G_m^k)^T(u - u^k), \quad (3b)
\end{align}
\]

where \( \nabla J_p^k, \nabla J_m^k \) are the gradients of the plant objective function \( J_p(u) \) and of the nominal model objective function \( j_m(u) \) with respect to the process input for the \( k \)th iteration. Similarly, \( \nabla G_p^k, \nabla G_m^k \) represent the gradients of the plant constraint functions \( G_p(u) \) and
the constraint functions of the nominal model \( G_m(u) \) with respect to the process input for the \( k \)th iteration. The modified optimization problem that is solved in the \( k \)th iteration is:

\[
\hat{u}^{k+1} = \arg \min_{u \in [u^*, m^*]} f_m^{ad,k}(u) \quad \text{s.t.} \quad G_m^{ad,k}(u) \leq 0, \tag{4a}
\]

where the determined optimum \( (\hat{u}^{k+1}) \) is the \( k + 1 \)th input which is applied to the plant.

The gradients of the plant objective function \( (\nabla f_p^k) \) and of the plant constraint functions \( (\nabla G_p^k) \) can be obtained for example using finite differences for which additional plant perturbations are required. Methods to approximate the plant gradients using the available past \( \ell \) number of inputs and steady-state measurements, without the need for additional plant perturbations, are proposed in Roberts [18], Brdyš and Tatjewski [34], and Gao et al. [35].

In MAWQA, the gradients of the plant objective function \( (\nabla f_p^k) \) and of the plant constraint functions \( (\nabla G_p^k) \) are obtained using surrogate QA models as long as the cardinality condition, i.e., the minimum number of data points required to identify all the parameters of the QA function, is fulfilled. A minimum of \( \ell := \frac{(n_u + 1)(n_u + 2)}{2} \) data points, i.e., process inputs and their corresponding steady-state process measurements, are required to fit a surrogate QA function \( (Q) \) for the plant objective function and for each of the plant constraint functions. Upon satisfying the cardinality condition, \( \nabla f_p^k \) and \( \nabla G_p^k \) are then obtained by evaluating the analytical derivatives of the surrogate QA models at \( u^k \). A surrogate QA model can be defined as

\[
Q(p, u) = \sum_{i=1}^{n_u} \sum_{j=1}^{n_u} a_{ij} u_i u_j + \sum_{i=1}^{n_u} b_i u_i + c, \tag{5}
\]

where \( p := \{a_{1,1}, \ldots, a_{n_u,n_u}, b_1, \ldots, b_{n_u}, c\} \) are the parameters of the surrogate QA model.

To obtain a surrogate QA model for each of the objective and constraint functions, a subset of at least \( \ell \) data points \( (U^k) \) is selected from the set of all data points available until the \( k \)th iteration \( (U^k) \). Wenzel et al. [36] reported a comparative study of existing methods to screen \( U^k \) from \( U^k \). In general, the methods for screening \( U^k \) attempt to include distant data points which are well distributed around \( u^k \) \((U_{dist}^k)\) and can act as anchor points for the QA. They also include all neighboring data points \( (U_{nb}^k) \), which lie within the inner-circle around \( u^k \). The inner-circle is defined as an \( n \)-dimensional sphere around \( u^k \) with the tuning parameter \( \Delta u \) as its radius. As proposed in Gao et al. [17], the quality of the distribution of data points in \( U_{dist}^k \) can be evaluated using a so-called conditionality-check criterion, the inverse of the condition number of \( s^k \) \((s^{-1}(s^k))\) has to be greater than a desired value \( (\delta) \). \( s^k \) in the conditionality-check criterion is defined as

\[
s^k = [u^k]_{n_u \times 1} \otimes 1_{\text{cardinality}(U_{dist}^k) \times 1} - [U_{dist}^k], \tag{6}
\]

where \( [U_{dist}^k] \) and \( [u^k] \) are matrix representations of the set \( U_{dist}^k \) and of \( u^k \). If the conditionality-check criterion fails, additional plant perturbations are performed and added to \( U^k \) to improve the distribution of the anchor points used for fitting the QA-function.

As the data points in \( U^k \) consist of input variables positioned locally around \( u^k \), the QA functions fitted using \( U^k \) are also local approximations of \( f_p \) and \( G_p \). In MAWQA, this is taken into account by restricting the process input for the next iteration \( (\hat{u}^{k+1}) \) obtained by solving the optimization (4) to lie inside an ellipsoidal trust region by adding the following constraint to the modifier adaptation problem in (4)
\[
(u - u^k)^T \text{cov}(U^k)(u - u^k) \leq \gamma^2, \quad (7)
\]

where \(\gamma\) is a tuning parameter that scales the trust region \([17]\). In addition, there are also elements of DFO \([7]\) that include the criticality-check, the quality-check and a possible switch to an optimization based on the quadratic approximation model. We refer the readers to Gao et al. \([17,37]\) for a detailed discussion on MAWQA.

Convergence to the plant optimum is assumed when the evaluation function \(C\) defined as
\[
C_i \overset{\text{def}}{=} \sum_{j=1}^{n_u} |(u'_i - u_{i-1}^j)||\nabla J_{i,p,j}|
\]
for all \(i = \{k + 1 - N_c, \ldots, k\}\) where \(\nabla J_{i,p,j}\) is the gradient of the plant objective function of the \(j\)th input-variable of \(k\)th input \(u^k\) \([38]\). The tuning parameter \(N_c\) is the number of iterations for which the condition \(C \leq |\varepsilon_{J_{k,p,best}}|\) has to be satisfied continuously. \(\varepsilon_{J_{k,p,best}}\) is the best value of the objective function realized by the plant up to the \(k\)th-iteration and \(\varepsilon\) is a tuning parameter that must be chosen taking into account the measurement noise and the required tolerance of the value of the optimum.

3. Active Perturbation Strategies

Figure 1 illustrates a situation that occurs typically in the process industries while implementing MA based RTO methods to optimize their processes. Consider a process that reaches a steady-state for constant inputs within \(\tau_p\) time units and the measurement device that requires a maximum of \(\tau_m\) time units to analyze a sample. An additional \(\tau_d\) time units is required for the sample to be transported to the measurement device making the total time to obtain steady-state measurements after a change of the inputs \(\tau_t := \tau_p + \tau_d + \tau_m\). If the measurement device is located at a remote position, this leads to a waiting period. Figure 2 illustrates the waiting period in an iteration of a MA algorithm. If an input \(u^{k+1}\) is given to the plant at time \(t\), its corresponding steady-state plant measurement becomes available at time \(t + \tau_t\). The time between iterations in MA is thus at least \(\tau_t\) time units if the plant gradients are computed using past inputs and measurements, and even larger if additional perturbations are performed. During the waiting period the plant remains at a steady-state and no additional process information is gained during this period, hence the convergence to the optimum steady-state is slowed down. Gottu Mukkula et al. \([32,33]\) proposed active perturbation strategies to perform input perturbations during the waiting period to gain additional process information that is used later to drive the process to its optimum faster. Figure 2 illustrates the active perturbation scheme where additional process perturbing inputs are given to the process, for example, between \(t + \tau_t + \tau_p\) and \(t + 2\tau_t\) to gain additional process information that can be used in the subsequent iterations, once the measurements of the effects of the input perturbations arrive. In general, the maximum possible number of input perturbations that can be given to a process within the waiting period can be computed as \([32,33]\)

\[
P_{\text{max}} := \min \left\{ \frac{\tau_d + \tau_m}{\tau_p}, \frac{\tau_d + \tau_p}{\tau_m} \right\}. \quad (8)
\]

In the following subsections, we present the active perturbation strategies proposed in Gottu Mukkula et al. \([32,33]\). Both active perturbation strategies are illustrated using an example case with 2 input variables and by assuming the maximum possible perturbations per iteration \(P_{\text{max}}\) as 4.
3.1. Active Perturbation around the Current Input (APCI)

According to the method proposed in Gottu Mukkula et al. [32], perturbing inputs around the input in the current iteration, i.e., $u^k$, are given to the process during the waiting period. Figure 3 illustrates this active perturbation method. In the figure, □ represents the inputs from MA. Let $u^k$ be the input given to the plant in the $k$th iteration at time $t$. The plant reaches its steady-state at $t + \tau_p$, and the steady-state measurements for $u^k$ are available at $t + \tau_t$. During the waiting period, i.e., from $t + \tau_p$ to $t + \tau_t$, $P_{\text{max}} = 4$ additional perturbations represented by circles (○) are performed around $u^k$. Redundant probing is avoided by suppressing perturbation inputs that are closer than a threshold $\xi$ from one of the earlier inputs. For example, the perturbing input represented by ○ is suppressed as it is in close vicinity to $u^k - 1$, and therefore, would be redundant. During the remaining waiting period, which is available due to suppressing of some perturbations, the plant is given the best known input ($u^k_{\text{best}}$) to avoid a loss of performance. As the perturbing inputs are not computed optimally, not all of them provide useful information. For example, in Figure 3, only the perturbation inputs represented by ⊕ are used by the QA scheme for computing the gradient at $u^{k+1}$.

3.2. Active Perturbation around an Estimate of the Next Input (APENI)

Here, the process is perturbed around an estimate of the next input, i.e., $\hat{u}^{k+1}$, instead of perturbing around current input $u^k$. Due to the unavailability of the measurements for the input $u^k$ during the waiting period, the plant gradients at $u^k$ cannot be computed. So, the next input $u^{k+1}$ cannot be calculated precisely without the plant gradients at $u^k$, and therefore is estimated using the following assumption: If $u^k$ is close to $u^{k-1}$, the difference between the plant and the model gradients at $u^k$ is approximately equal to the gradient difference at $u^{k-1}$, i.e., $(\nabla J^p_k - \nabla J^m_k) \approx (\nabla J^p_{k-1} - \nabla J^m_{k-1})$ and $(\nabla G^p_k - \nabla G^m_k) \approx (\nabla G^p_{k-1} - \nabla G^m_{k-1})$. The
same is assumed for the bias correction term \((G_p^k - G_m^k)\). In other words, the optimization problem is unchanged. The input in the next iteration can then be estimated by solving the following modified MA problem:

\[
\hat{u}^{k+1} = \arg \min_{u \in [u^k, u^{k+1}]} f_{\text{ind}, e}^{m,k}(u) 
\]

s.t.
\[
G_{m,e}^{m,k}(u) \leq 0, 
\]

\[
(u - u^k)^T \text{cov}(u^k)(u - u^k) \leq \gamma^2, 
\]

(9a)

(9b)

(9c)

where

\[
f_{\text{ind},e}^{m,k}(u) = f_m(u) + (\nabla f_p^{k-1} - \nabla f_m^{k-1})^T (u - u^k), 
\]

\[
G_{m,e}^{m,k}(u) = G_m(u) + (G_p^{k-1} - G_m^{k-1}) + (\nabla G_p^{k-1} - \nabla G_m^{k-1})^T (u - u^{k-1}). 
\]

(10a)

(10b)

The problem in (9) for computing \(\hat{u}^{k+1}\) is the same as the problem in (3) for computing \(u^{k+1}\), except that the trust-region in (7) for computing \(\hat{u}^{k+1}\) is now centered around \(u^k\).

In Figure 4a, an ideal scenario for an active perturbation around next input is illustrated. In the ideal case, the assumption on the modifiers is valid, i.e., \((\nabla f_p^{k-1} - \nabla f_m^{k-1}) \approx (\nabla G_p^{k-1} - \nabla G_m^{k-1}) \approx (\nabla G_p^{k-1} - \nabla G_m^{k-1}) \approx (\nabla G_p^{k-1} - \nabla G_m^{k-1})\). Then, \(\hat{u}^{k+1}\) is useful, and therefore, the measurement information from the perturbed inputs are used to compute the inputs in upcoming iterations. Figure 4b illustrates a nonideal scenario where the assumption about the modifiers does not hold, so \(\hat{u}^{k+1} \neq u^{k+1}\). Therefore, as shown in Figure 4b, not all measurement information from the perturbing inputs may be useful in the upcoming iterations. Redundant perturbation inputs are suppressed similar to the active perturbation around the current input (APCI) method.

4. Proactive Perturbation Scheme

Although the perturbation schemes proposed in Gottu Mukkula et al. [32,33] gain additional measurement information by perturbing the process during the waiting period, they are suboptimal as they are based on heuristics. Also, the measurement information gained from the additional input perturbations is not used immediately. Therefore, in this paper, we propose a proactive perturbation scheme where the additional perturbation inputs are computed by solving an optimization problem using all available process information. The key idea of the proactive perturbation scheme is illustrated in Figure 5 using an example case with \(n_u = 1\), \(P_{\text{max}} = 5\), and where the plant gradients are estimated using QA. At \(t = 0\), an input \(u^0\) is given to the process, the corresponding measurements become available after time \(\tau\). In the initial (0th) iteration, FD is used for gradient approximation as there are no past data points for gradient approximation. Therefore, after applying \(u^0\)
for \( \max\{\tau_p, \tau_m\} \) units of time, the process is operated with \( u_0^0 \), for gradient approximation, the corresponding measurements are available at time \( \zeta := \tau_1 + (n_u \times \max\{\tau_p, \tau_m\}) \) and a new iteration input \( u^k \) is computed by solving the MA problem in (3). During the waiting period in the 0th iteration, i.e., from \( (n_u + 1) \times \max\{\tau_p, \tau_m\} \) to \( \zeta \), \( P_{max} \) additional perturbation moves (from \( u^1 \) to \( u^5 \)) around \( u^0 \) are performed to gain additional knowledge about the process. For \( n_u := 1 \), steady-state measurements for at least \( \ell := 3 \) different inputs are required for the process gradients to be approximated using QA (cardinality condition). We note that \( \ell \) varies if other methods are used for plant gradient estimation using past data. In this example, after time \( \zeta + \max\{\tau_p, \tau_m\} \), there are already measurements for \( \ell \) inputs. From here on, an MA problem is solved each time a new measurement is available as the plant gradients are computed using past inputs and measurements. From this point onward, each input to the process is optimal with respect to the available process knowledge, thereby making the proposed proactive perturbation scheme more efficient than the active perturbation methods proposed in Gottu Makkula et al. [32, 33].

A flowsheet for the implementation of the proposed proactive perturbation scheme is presented in Figure 6. In the flowsheet, \( \phi \) is the total number of inputs given to the plant, also including the inputs whose steady-state measurements are not yet available and \( \psi \) is the total number of inputs given to the plant for whom the steady-state plant measurements are already available. The major steps in the flowsheet are:

1. Perturbation input for gradient estimation using FD
2. Perturbation input for gradient estimation using FD, also considered as an input \( (u^k) \) for MA problem formulation
3. Perturbation input to gain additional information during the waiting period
4. Perturbation input to gain additional information during the waiting period, also considered as an input \( (u^k) \) for MA problem formulation
5. Input \( (u^{k+1}) \) computed by solving a MA problem (4) with plant gradients approximated using FD
6. Perturbation input considered as an input \( (u^k) \) for MA problem formulation, to ensure a continuous flow of steady-state measurements every \( \max\{\tau_p, \tau_m\} \) units of time
7. Input \( (u^{k+1}) \) computed by solving MA problem with gradients approximated from past (at least \( \ell \)) measurements

In all the active perturbation strategies discussed in this section, to avoid frequent probing of the plant with unsuccessful inputs, and thus, deteriorating the plant performance, a list of unsuccessful moves is stored. An input from MA \( u^k \) is applied to the plant only if the number of past inputs on the list that are not farther than \( \phi \) from \( u^k \) is less than a
predefined number $\chi_{count}$. The parameters $\varphi$, $\chi_{count}$ are defined depending on the expected level of measurement noise.

Initialize with $u^0, k = 0$

Figure 6. Flowsheet for proactive perturbation scheme for MA.

5. Williams–Otto Reactor Case Study

5.1. Process Description

The Williams–Otto reactor case study [39] is a benchmark case study widely used to analyze the performance of RTO methods. Reactants A and B react in a CSTR to produce products E and P, and a side product G. While the reaction in the plant follows a three-step reaction mechanism, the nominal model considers a two-step reaction mechanism, ignoring the formation of an intermediate component C, leading to a structural plant-model mismatch. The reaction mechanisms for the plant and in the nominal model are:
Plant: Nominal model:

\[ \begin{align*}
A + B & \xrightarrow{k_1} C, \\
C + B & \xrightarrow{k_2} P + E, \\
P + C & \xrightarrow{k_3} G.
\end{align*} \]

\[ \begin{align*}
A + 2B & \xrightarrow{k_1} P + E, \\
A + B + P & \xrightarrow{k_1} G + E.
\end{align*} \]

The goal is to maximize the profit function \( J(y, u) := (1143.38x_{P,ss} + 25.92x_{E,ss})F_R - 76.23F_A - 114.34F_B, \) where \( x_{P,ss}, x_{E,ss} \) are the measured steady-state concentrations of products \( P, E, \) and \( F_R \) is the summation of the feed flowrates of reactants \( A \) and \( B, \) i.e., \( F_R := F_A + F_B. \) \( F_B \) and the reaction temperature \( T_R \) are the manipulated variables \( (u = [F_B, T_R]) \) with a range \([3, 6] \) kg s\(^{-1}\) and \([343.15, 373.15] \) K. The steady-state plant model \( (F_p(u)) \) and the nominal model \( (F_m(u)) \) are:

\[
F_p(u) := \begin{pmatrix}
F_A - F_R x_A - m_R r_1 \\
F_B - F_R x_B - m_R (r_1 + r_2) \\
-F_R x_C + m_R (2r_1 - 2r_2 - r_3) \\
-F_R x_E + 2m_R r_2 \\
-F_R x_G + 1.5m_R r_3 \\
-F_R x_P + m_R (r_2 - 0.5r_3) \\
F_R - F_A - F_B
\end{pmatrix} = 0, \tag{11}
\]

and

\[
F_m(u) := \begin{pmatrix}
F_A - F_x x_A - m_R \tilde{r}_1 \\
F_B - F_x x_B - m_R (2\tilde{r}_1 + \tilde{r}_2) \\
-F_R x_C + m_R \tilde{r}_1 \\
-F_R x_E + 2m_R \tilde{r}_1 \\
-F_R x_G + 3m_R \tilde{r}_2 \\
-F_R x_P + m_R (\tilde{r}_1 - \tilde{r}_2) \\
F_R - F_A - F_B
\end{pmatrix} = 0, \tag{12}
\]

where \( x_i \) is the concentration of component \( i \) and \( m_R := 2105 \) kg is the mass of the reaction mixture in the reactor. \( r := \{r_1, r_2, r_3\}, \tilde{r} := \{\tilde{r}_1, \tilde{r}_2\} \) represent the reaction rates in the plant, and in the nominal model. The reaction rates and their kinetic parameters are defined as:

\[
\begin{align*}
r_1 &= k_1 x_A x_B, & k_1 &= 1.660 \times 10^6 e^{-6666.7/(T_R + 273.15)}, \\
r_2 &= k_2 x_B x_C, & k_2 &= 7.212 \times 10^6 e^{-8333.3/(T_R + 273.15)}, \\
r_3 &= k_3 x_C x_P, & k_3 &= 2.675 \times 10^6 e^{-11111/(T_R + 273.15)}, \\
\tilde{r}_1 &= \tilde{k}_1 x_A x_B^2, & \tilde{k}_1 &= 2.19 \times 10^9 e^{-8077.6/(T_R + 273.15)}, \\
\tilde{r}_2 &= \tilde{k}_2 x_A x_B x_P, & \tilde{k}_2 &= 4.31 \times 10^{13} e^{-12438.5/(T_R + 273.15)}.
\end{align*}
\]

For the purpose of illustration of the performance of the proposed proactive perturbation scheme, we consider that \( \tau_p, \tau_p, \tau_m \) are 3, 12 and 3 min.

5.2. Simulation Results

The modifier adaptation scheme MAWQA with GMA [26] is used in the simulation study to compare the methods active perturbation around the current input, active perturbation around an estimate of the next input, and the proposed proactive perturbation scheme using the tuning parameters listed in Table 1. In MAWQA with GMA, the plant gradients are estimated from past inputs and steady-state measurements using QA. All schemes are initialized at \( u^0 := [3, 100] \). For the simulation study, the measured plant profit is corrupted by Gaussian noise with zero mean and 1.0 standard deviation.
Table 1. Tuning parameters for simulation study.

| No. | Parameter                                          | Williams–Otto | Lithiation Reaction |
|-----|----------------------------------------------------|---------------|---------------------|
| 1.  | trust-region scaling ($\gamma$)                   | 3             | 3                   |
| 2.  | radius of inner circle ($\Delta u$)               | 0.1           | 0.125               |
| 3.  | step length for finite differences ($\Delta h_u$) | 0.1           | 0.2                 |
| 4.  | conditionality limit ($\delta$)                   | 0.2           | 0.25                |
| 5.  | threshold to stop active perturbations ($\zeta$)   | 0.05          | 0.1                 |
| 6.  | threshold for unsuccessful iteration ($\varphi$)   | 0.005         | 0.005               |
| 7.  | number of times an unsuccessful input is probed ($\chi_{count}$) | 3            | 3                   |
| 8.  | number of inputs looked at for convergence ($N_c$) | 5             | 5                   |
| 9.  | accepted variance of $J$ for convergence ($\varepsilon$) | 0.01          | –                   |

Figure 7 illustrates the evolution of the inputs (scaled) and the plant profit function from MAWQA with GMA, MAWQA with GMA with active perturbation schemes APCI and APENI, and MAWQA with GMA with the proposed proactive perturbation scheme. Descriptions of all markers in the figure are given in Table 2. The axis for the value of the profit function ($\mathcal{O}$) can be found on the right-side of the plot and the axis for the inputs (scaled) to the plant is on the left. Each vertical grid line in the plot represents the completion of an iteration, which includes all steps from giving input to a process to computing a new input by solving a modifier-adaptation problem. For MAWQA with GMA and MAWQA with GMA with active perturbation schemes APCI and APENI, after applying $u^0 ([\bullet, \bullet])$ at $t = 0$, two input steps (perturbations $[\bullet, \bullet]$) with a scaled step length $\Delta h_u$ are performed to compute the plant gradients using finite differences. Each input is applied to the plant until it reaches a steady-state, i.e., for $\tau_p := 3$ min. The steady-state plant measurement ($\mathcal{O}$) for each applied input are obtained after $\tau_i := 18$ min from applying an input. Upon computing the plant gradients using the steady-state plant measurements, the modifier adaptation problem in (4) is solved to compute the the input $u^1$. As the condition for cardinality is not met, i.e., $\|U^1\| < \ell$, $n_u := 2$ additional perturbations ($[\bullet, \bullet]$) are performed after $u^1 ([\bullet, \bullet])$ to compute the plant gradients. Similar to the 0th iteration, the MA problem in (4) is solved in the 1st iteration to compute $u^2$. From here on, the cardinality condition is always satisfied. Therefore, QA is used for gradient approximation for all further iterations. In MAWQA with GMA, additional input perturbations are made only if the criticality-check criterion is not satisfied, for example in the 2nd iteration after applying $u^2$ to the plant.

Table 2. Description of markers in Figure 7. Input markers in red and black color refer to the input variable $u_1$; blue and magenta input markers refer to the input variable $u_2$.

| Marker | Description                                      |
|--------|--------------------------------------------------|
| [red, black] | successful-iteration input ($[u_1, u_2]$)       |
| [red, blue]  | unsuccessful-iteration input ($[u_1, u_2]$)      |
| [red, magenta]  | perturbation input ($[u_1, u_2]$)           |
| [red, magenta]  | value of the plant profit function           |

In MAWQA with GMA with active perturbation schemes APCI and APENI, additional perturbations are made once the cardinality condition is satisfied, i.e., from the 2nd iteration onwards, and if there are no past inputs lying closer than $\zeta$ to the scaled planned perturbation inputs. In MAWQA with GMA with APCI, the additional perturbations are stopped from the 11th iteration, as from $k := 11$, there are no past inputs lying farther than $\zeta$ from all the planned perturbation inputs. In MAWQA with GMA with APENI, the
perturbation inputs in kth iteration are chosen around an estimate of iteration input for k + 1th iteration. From the iteration k := 10 onwards, all the planned perturbation inputs are not farther than \( \xi \) from earlier inputs, so all active perturbations are stopped.

Figure 7. Williams–Otto reactor case study: evolution of the plant profit function and the inputs from MAWQA with GMA with different active perturbation schemes. The vertical red-dashed line represents the time of convergence. Note that the time axis is scaled differently in the different subplots.

In the new scheme, MAWQA with GMA with proactive perturbation, after applying the input \( u^0 \) to the plant, as \( \phi := 1 \) and \( \phi < \ell \), two input perturbations (\( u^0_{f1}, u^0_{f2} \)) are performed to compute the plant gradients using finite differences. All necessary steady-state measurements for gradient correction, i.e., \( u^0 \) and two additional
perturbations $u^0_{fd_1}, u^0_{fd_2}$ are available after $\zeta := 24$ min. During the waiting period, i.e., from $(n_u + 1) \times \max\{\tau_p, \tau_m\} := 9$ min to $\zeta := 24$ min, $P_{\max} := 5$ additional input perturbations $u^0_{ap_1}, \ldots, u^0_{ap_5}$ are performed to gain additional plant information. As the input perturbations when $\phi \geq \ell$ are considered as inputs from MA (according to the flowsheet in Figure 6), among the 5 additional perturbations, $u^0_{ap_3}, \ldots, u^0_{ap_5}$ are renamed as $u^1, u^2, u^3$. At 24 min, steady-state measurements for $u^0$ and $u^0_{fd_1}, u^0_{fd_2}$ are available. Therefore, an MA problem is formulated and solved to compute the next input $u^4$. For the next iteration, as $\phi := 9$ and $\phi \not< \ell$, FD is no more used for gradient approximation. However, there are not enough measurements for QA, i.e., $\psi := 4$, after applying $u^4$. As $(\ell - \psi) := 2$, input perturbations $u^5$ and $u^6$ are added to smoothly switch from using FD for gradient approximation to QA. After $u^6$, $\psi = \ell$ and a new steady-state measurement is always available every $\max\{\tau_p, \tau_m\}$. From here on, a new iteration input is computed every $\max\{\tau_p, \tau_m\} := 3$ min taking into account the latest measurement information available.

Although all the schemes converged to an input in the neighborhood of the plant optimum, the MAWQA with GMA scheme took the most time, 375 min and 20 iterations to converge. The time of convergence is indicated with a red-dashed vertical line in Figure 7. The MAWQA with GMA, APCI, and APENI gained additional plant information from suboptimal input perturbations, and therefore converged in 303, 249 min and 16, 13 iterations. Finally, MAWQA with GMA and proactive perturbation scheme converged in only 123 min with 31 iterations. It gained more information per time from additional perturbations than that of the earlier proposed active perturbation schemes and also by using the measurement information immediately after it is available.

The mean profit function for 100 simulations for all schemes is shown in Figure 8. The mean and standard deviation of the profit function did not vary significantly upon convergence. The mean and standard deviation of the times of convergence for all modifier adaptation schemes is also shown. In the proactive perturbation scheme, the profit function converges earlier to the plant optimum, followed by the active perturbations schemes APCI, APENI, and the standard modifier adaptation scheme. The mean time of convergence for the MAWQA with GMA if $\tau_d = 0$, shown in green, is longer than the mean time of convergence of the proposed proactive perturbation scheme as the proposed scheme also overcomes $\tau_m$ in addition to $\tau_d$. This illustrates that the proposed scheme overcame the effect of time delay ($\tau_d >> \tau_p, \tau_m$) in this case.

The Friedman ranking test [40] was performed for a ranking comparison of the proposed proactive perturbation scheme with standard MAWQA, and the APCI and
APENI schemes based on the time of convergence for 100 realizations of the measurement noise. The Friedman test is a nonparametric statistical test to recognize performance differences among different methods across multiple test samples. For each realization of the measurement noise, the methods are ranked based on their time of convergence. The lowest mean rank for a method implies its consistently superior performance, and vice versa for the highest mean rank. The mean rank of the standard scheme with no additional perturbations is 3.87, for APCI it is 2.6, for APENI 2.51, and the proposed proactive perturbation scheme has a mean rank of 1.02. Thus, the proposed scheme clearly showed a superior performance when compared to that of the other methods.

6. Lithiation Process Case Study

6.1. Process Description

The lithiation case study is a real case where $\tau_d \gg \{\tau_p, \tau_m\}$ [41]. The iterative real-time optimization method MAWQA was successfully applied to the real process by Gottu Mukkula et al. [41] but there is still scope for improvement in the time of convergence to the real process optimum due to the large waiting period in the process. In this simulation study, we aim at proving that the plant optimum could be identified much earlier with the proposed proactive perturbation scheme.

A lithiation reaction [41] is performed to produce Lithium 2-Nitrodiphenylamine (component H) in a containerized reactor module developed within the F$^3$-factory project [42] at the INVITE GmbH facility. All chemical components involved in the lithiation reaction are listed in Table 3 and the reaction mechanism is shown in Figure 9. The main modules of the containerized reactor, i.e., the coiled tubular reactor, a NIR flow cell, an online-NMR sensor [43] and a product filter, are shown in Figure 10. The byproduct Lithium fluoride (component G) precipitates along the length of the tubular reactor, and the solid G is filtered out before the reaction mixture from the outlet of the coiled reactor is collected in a product tank. A filtered sample of the reaction mixture is continuously fed to an online-NMR measurement system and to a NIR flow cell to measure the concentration of the product in the reaction mixture at the reactor outlet [43].

![Reaction Mechanism](image)

Figure 9. Detailed reaction mechanism of lithiation reaction. Names of the chemical components are given in Table 3.
Table 3. List of chemical components involved in lithiation reaction.

| Label | Chemical Name (Abbreviation)                | Role          |
|-------|--------------------------------------------|---------------|
| A     | Aniline (An)                               | reactant      |
| B     | Lithium bis(trimethylsilyl)amide (Li-HMDS) | reactant      |
| C     | Lithium phenylazanide (Li-An)              | intermediate  |
| D     | Hexamethyldisilazane (HMDS)                | by-product    |
| E     | 1-Fluoro-2-nitrobenzene (FNB)              | reactant      |
| F     | 2-Nitrodiphenylamine (NDPA)                | intermediate  |
| G     | Lithium fluoride (LiF)                     | by-product    |
| H     | Lithium 2-Nitrodiphenylamine (Li-NDPA)     | product       |

Additional solvent: Tetrahydrofuran (THF)

Figure 10. (A) picture of modular pilot plant for continuous production of Li-NDPA. (B) close-up of tubular reactor (D, inner diameter = 12.4 mm) and filter section (E). (C) picture of integrated compact NMR spectrometer (43 MHz) with ATEX certified pressurized housing for online concentration measurements. (F) indicates location of the NIR flow cell.

The following steady-state plant model was developed assuming a negligible value of the dispersion coefficients for concentrations and temperature:

\[
\frac{\partial C_i}{\partial z} = \frac{r_i}{v} \quad \forall i \in \{A, B, C, D, E, F, G, H\}, \quad (13a)
\]

\[
\frac{\partial T_R}{\partial z} = \frac{1}{v \rho_c} \left( K_A (T_R - T_{Envi}) - \Delta H_1 k_1 C_A C_B - \right.
\]

\[
\left. \Delta H_2 k_2 C_A C_E - \Delta H_3 k_3 C_C C_F - \Delta H_4 k_4 C_B C_F \right), \quad (13b)
\]

where \( r_i \) is the reaction rate of component \( i \), described by:

\[
r_A = -k_1 C_A C_B + k_3 C_C C_F, \quad r_E = -k_2 C_C C_E, \quad (14a)
\]

\[
r_B = -k_1 C_A C_B + k_4 C_B C_F, \quad r_F = k_2 C_C C_E - k_3 C_C C_F - k_4 C_B C_F, \quad (14b)
\]

\[
r_C = k_1 C_A C_B - k_2 C_C C_E - k_3 C_C C_F, \quad r_G = k_2 C_C C_E, \quad (14c)
\]

\[
r_D = k_1 C_A C_B + k_4 C_F C_B, \quad r_H = k_3 C_C C_F + k_4 C_F C_B. \quad (14d)
\]

The model parameters are provided in Table 4, and the reaction rates constants \( k_i \forall i = 1, \ldots, 4 \) depend on the temperature via an Arrhenius-type equation:

\[
k_i = k_{i0} \exp\left(\frac{-E}{RT_R}\right). \quad (15)
\]

Although a four-step reaction takes place in the reactor, the nominal model instead considers a single reaction:

\[
A + 2B + E \rightarrow 2D + G + H, \quad (16)
\]
leading to a structural plant-model mismatch. The steady-state nominal model is defined as follows:

\[
\begin{align*}
\frac{\partial C_i}{\partial z} &= r_i v_z \quad \forall i \in \{A, B, D, E, G, H\}, \\
\frac{\partial T_R}{\partial z} &= \frac{1}{v_z \rho c_p} (KA(T_R - T_{Envi}) - \Delta H kC_A C_B C_E).
\end{align*}
\] (17a) (17b)

The reaction rates \(r_i\) in the nominal model follow the elementary rate law with the rate constant \(k\) defined by the Arrhenius equation with \(k = 17.8 \exp(\frac{E}{RT})\).

Table 4. Parameter values used in plant model.

| Parameter | Value |
|-----------|-------|
| \(\Delta H_1, \ldots, \Delta H_4\) | \(-6 J/mol\) |
| \(k_{10}, \ldots, k_{40}\) | \(17.8 s^{-1}\) |
| \(K A\) | \(0.001 W/K\) |
| \(T_{Envi}\) | \(293.15 K\) |
| \(\rho\) | \(900 kg/m^3\) |
| \(T_{in}\) | \(283.15 K\) |
| \(E\) | \(25 \times 10^3 J/mol\) |
| \(\Delta H\) | \(-19 J/mol\) |
| \(c_p\) | \(0.123 J/(mol K)\) |
| \(R\) | \(8.314 J/(mol K)\) |

The goal is to maximize the plant profit that is computed by the function

\[
J(y, u) = \frac{\bar{w}_H C_{H,ss} M_H}{\rho} \left( 8 + \sum_{i=1}^{2} u_i \right) - \bar{w}_A u_A - \bar{w}_E u_E,
\] (18)

where \(u_A\) and \(u_E\) are the feed flowrates of Aniline and FNB in kg h\(^{-1}\). The feed flow rate of LiHMDS is set to the maximum of 8 kg h\(^{-1}\). The manipulated variables \(u_A\) and \(u_E\) can take values between 3 and 8 kg h\(^{-1}\). \(M_H = 0.1752 \text{ kg mol}^{-1}\) is the molecular weight of the component H, \(\{\bar{w}_H, \bar{w}_A, \bar{w}_E\}\) reflect the component prices, their values are \(\{450,000, 10,000, 12,000\} \text{ $ kg}^{-1}\) and \(C_{H,ss}\) is the measured steady state concentration of component H in mol m\(^{-3}\). \(\{\tau_p, \tau_d, \tau_m\}\) are \(\{3, 12, 3\}\) min.

6.2. Simulation Results

All tuning parameters used in the simulation study are listed in Table 1. The RTO schemes are initialized at \(u^0 = [3.58, 3.58]\) and the measured variable \(C_H\) is corrupted by Gaussian noise with zero mean and 1.0 standard deviation. Similar to the Williams–Otto reactor case study, model adequacy is enforced using the guaranteed model adequacy (GMA) proposed in Gottu Mukkula and Engell [26]. Convergence to the process optimum, in this case study, is assumed when the profit from \(N_c\) consecutive iterations is greater than \(1.175 \times 10^5\). Figure 11 illustrates the mean and the 10 to 90 percentile range of the profit function, and the mean time of convergence for 100 simulations of MAWQA with GMA, MAWQA with GMA with active perturbation schemes APCI and APENI, and MAWQA with GMA with the proposed proactive perturbation scheme. The deviations in the profit function are due to the measurement noise, which affects the computation of the next inputs via the calculation of the quadratic approximation. The figure shows that the proposed proactive perturbation scheme converges faster when compared to that of MAWQA with GMA and the two active perturbation variants proposed earlier. The reason for this is that the idle time is used better by intelligently scheduling the process inputs taking into account the time delays to obtain the steady-state process measurements. Figure 12 shows the mean value of the profit function from MAWQA with GMA, MAWQA with GMA with APCI and APENI, and MAWQA with GMA with the proposed proactive perturbation scheme. The mean time of convergence and the standard deviation of the time of convergence for 100 simulations are shown as error bars. In addition to the faster convergence of the proposed proactive perturbation scheme, the standard deviation is also smaller when compared to the other schemes. Similar to the Williams–Otto reactor case
study, the mean time of convergence of the proposed proactive perturbation scheme is even shorter than that of the MAWQA with GMA scheme with $\tau_d = 0$. The mean Friedman rank [40] of the standard scheme with no additional perturbations is 3.64, for APCI it is 2.82, for APENI 2.5, and for the proposed proactive perturbation scheme it is 1.06.

Figure 11. Lithiation reaction case study: mean and standard deviation of plant profit function and mean time of convergence for 100 simulations. Convergence is assumed when profit for inputs from five consecutive iterations is greater than $1.175 \times 10^5$. (a) MAWQA with GMA. (b) MAWQA with GMA with APCI. (c) MAWQA with GMA with APENI. (d) MAWQA with GMA with proactive perturbation.

Figure 12. Lithiation reaction case study: evolution of mean profit function (solid lines), mean time of convergence (vertical dashed lines), and standard deviations of time of convergence.
The time to compute a new input in an Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz × 4 using a 64 bit Windows 10 operating system for both case studies ranged between 0.01 and 0.02 s. Such short computation time would enable the usage of the proposed proactive perturbation scheme also for processes with very short settling time.

7. Conclusions

In this paper, we address iterative RTO with plant-model mismatch in the presence of a time delay caused due to the positioning of an analytic measurement device at some distance from the place where the process stream is sampled. The time delay leads to a longer time of convergence to the real process optimum for modifier adaptation methods, during which the plant is not operated optimally. Perturbation schemes to handle this time delay in the sense that the period where one must wait for the result of the transfer of the sample and the analytic measurement is used for further probing of the plant were proposed earlier, but they rely on heuristics for the choice of the plant perturbations. A new proactive perturbation scheme is proposed in this paper to handle the time delay where additional perturbation inputs are computed by solving an optimization problem using all available process information, which contrasts the earlier proposed active perturbation schemes.

In the proposed proactive perturbation approach, the measurement delay affects the time of convergence only once at the beginning, until a number of plant measurements for gradient approximation using quadratic approximation are available. Later, the proactive perturbation scheme computes a new input by solving a modifier adaptation problem as soon as new measurement information is available, with a sampling time equal to the maximum of the settling time of the plant and the time taken by the measurement device to analyze a sample. The performance of the proposed scheme was evaluated for two examples and compared with that of other active perturbation schemes using two case studies. The proposed proactive perturbation scheme drives the real plant to its optimum significantly faster than the other approaches, and it significantly reduces the effect of the measurement time delay for both case studies because the scheme collects more useful information from the perturbations. An important issue in the application of MA schemes to real plants is to freeze and restart the adaptation when the input moves are mainly caused by measurement noise. Initial proposals for this can be found in [38], but this requires more research in the future.

Author Contributions: Conceptualization, formal analysis, methodology, software, writing—original draft preparation, validation, visualization, A.R.G.M.; conceptualization, funding acquisition, supervision, writing—review and detailed editing, S.E. Both authors have read and agreed to the published version of the manuscript.

Funding: The research leading to these results was funded the European Union’s Horizon 2020 research and innovation program under grant agreement number 636942 “CONSENS—Integrated Control and Sensing”. Gefördert durch die Deutsche Forschungsgemeinschaft (DFG)—TRR 63 “Integrierte chemische Prozesse in flüssigen Mehrphasensystemen” (Teilprojekt D4)—56091768 (supported by Deutsche Forschungsgemeinschaft in the context of the Interdisciplinary Research Centre TRR 63, Integrated Chemical Processes in Liquid Multiphase Systems, Subproject D4).

Conflicts of Interest: The authors declare no conflict of interest.
Symbols

The following symbols are used in this manuscript:

- \( u \) input variables
- \( n_u \) number of input variables
- \( F_p(u) \) true process model
- \( F_m(u) \) nominal process model
- \( y \) measured variables
- \( \hat{y} \) values of the measured variables estimated using a nominal process model
- \( n_y \) number of measured variables
- \( n_c \) number of constraint functions in the optimization problem
- \( u^*_p \) true process optimum
- \( u^*_m \) optimum computed using a nominal process model
- \( u^L \) lower bound of the input variables
- \( u^U \) upper bound of the input variables
- \( J(y, u) \) cost function in the optimization problem
- \( J_p(u) \) cost function computed using true process measurements
- \( J_m(u) \) cost function computed using estimated values of the measured variables
- \( J_{ad,k}^{m} \) cost function of the modifier adaptation problem in the \( k \)th iteration
- \( G(y, u) \) constraint functions of the optimization problem
- \( G_p(u) \) constraint functions evaluated using the true process measurements
- \( G_m(u) \) constraint functions evaluated using estimated values of the measured variables
- \( G_{ad,k}^{m} \) constraint functions of the modifier adaptation problem in the \( k \)th iteration
- \( \nabla \) gradient operator
- \( Q(p, u) \) quadratic function
- \( p \) parameters of the quadratic function
- \( \ell \) number of parameters in the quadratic function
- \( \mathbb{U}^k \) set of all data points available until the \( k \)th iteration of MA
- \( \mathbb{U}_k^{dist} \) selected data points for quadratic approximation in the \( k \)th iteration of MA
- \( \mathbb{U}_k^{nb} \) anchor points in \( \mathbb{U}_k^{k} \)
- \( \Delta u \) radius of the inner circle (tuning parameter)
- \( \Delta h_u \) step length for finite differences (tuning parameter)
- \( \delta \) minimum value for the inverse of the condition number (tuning parameter)
- \( s^k \) inverse of the condition number in the \( k \)th iteration of MA
- \( \hat{u}^{k+1} \) input variables for the \( k + 1 \) iteration computed by solving the MA problem in the \( k \)th iteration
- \( \hat{u}_c^{k+1} \) estimated input variables for the \( k + 1 \) iteration
- \( \gamma \) tuning parameter to scale the trust region (tuning parameter)
- \( \text{cov} \) covariance operator
- \( C \) evaluation function to identify the convergence of MA to true process optimum
- \( \varepsilon \) accepted variance of \( J_p(u) \) for convergence (tuning parameter)
- \( J_{p,best}^k \) best value of cost function computed using the plant measurements until the \( k \)th iteration of MA
- \( N_c \) number of iterations the convergence criterion has to be satisfied (tuning parameter)
- \( \tau_p \) time taken by the process to reach steady-state after a change of the inputs
- \( \tau_d \) time required for the sample to reach (or to be transported to) the measurement device
- \( \tau_m \) time required for the measurement device to analyze the sample
- \( \tau_l \) total time to obtain steady-state measurements after a change in the input
- \( F_{\text{max}} \) maximum possible number of input perturbations in the waiting period
- \( \phi \) number of inputs given to the process, including the inputs whose measurements are not available
- \( \psi \) number of inputs given to the process for whom the measurements are already available
- \( \xi \) threshold to stop active perturbations (tuning parameter)
- \( \zeta \) threshold for unsuccessful iteration (tuning parameter)
- \( \chi_{\text{count}} \) number of times an unsuccessful input can be probed (tuning parameter)
Abbreviations
The following abbreviations are used in this manuscript:

- RTO: Real-time optimization
- SQP: Sequential quadratic programming
- ISOPE: Integrated system optimization and parameter estimation
- IGMO: Iterative gradient modification optimization
- MA: Modifier adaptation
- FD: Finite differences
- QA: Quadratic approximation
- DFO: Derivative free optimization
- MAWQA: Modifier adaptation with quadratic approximation
- APCI: Active perturbation around the current input
- APENI: Active perturbation around an estimate of the next input
- GMA: Guaranteed model adequacy
- NMR: Nuclear magnetic resonance
- ATEX: Atmosphere explosibles
- IECEx: International electrotechnical commission explosive

References
1. Marchetti, A.G.; François, G.; Faulwasser, T.; Bonvin, D. Modifier Adaptation for Real-Time Optimization—Methods and Applications. *Processes* 2016, 4, 55. [CrossRef]
2. Marlin, T.E.; Hrymak, A.N. Real-time operations optimization of continuous processes. In *Chemical Process Control-V*; AIChE Symposium Series; American Institute of Chemical Engineers: New York, NY, USA, 1997; Volume 93, pp. 156–164.
3. Philip, E.; Murray, W.; Saunders, M.A.; Wright, M.H. User’s Guide for NPSOL 5.0: A Fortran Package for Nonlinear Programming; Technical Report SOL 86–6; Stanford University: Stanford, CA, USA, 2001.
4. Gill, P.E.; Murray, W.; Saunders, M.A. SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization. *SIAM Rev.* 2005, 47, 99–131. [CrossRef]
5. Wächter, A.; Biegler, L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* 2006, 106, 25–57. [CrossRef]
6. Kaustuv. IPSOL: An Interior Point Solver for Nonconvex Optimization Problems. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 2009.
7. Conn, A.R.; Scheinberg, K.; Vicente, L.N. *Introduction to Derivative-Free Optimization*; SIAM: Philadelphia, PA, USA, 2009.
8. Rios, L.M.; Sahinidis, N.V. Derivative-free optimization: A review of algorithms and comparison of software implementations. *J. Glob. Optim.* 2013, 56, 1247–1293. [CrossRef]
9. Jedrzejowicz, P. Current Trends in the Population-Based Optimization. In Proceedings of the 11th International Conference on Computational Collective Intelligence, Hendaye, France, 4–6 September 2019; Springer International Publishing: Cham, Switzerland, 2019; pp. 523–534.
10. Wang, Z.; Qin, C.; Wan, B.; Song, W.W. A Comparative Study of Common Nature-Inspired Algorithms for Continuous Function Optimization. *Entropy* 2021, 23, 874. [CrossRef]
11. Jang, S.S.; Joseph, B.; Mukai, H. On-line optimization of constrained multivariable chemical processes. *AIChE J.* 1987, 33, 26–35. [CrossRef]
12. Chen, C.Y.; Joseph, B. On-line optimization using a two-phase approach: An application study. *Ind. Eng. Chem. Res.* 1987, 26, 1924–1930. [CrossRef]
13. Roberts, P. An algorithm for steady-state system optimization and parameter estimation. *Int. J. Syst. Sci.* 1979, 10, 719–734. [CrossRef]
14. Tatjewski, P. Iterative optimizing set-point control—The basic principle redesigned. *IFAC Proc. Vol.* 2002, 35, 49–54. [CrossRef]
15. Gao, W.; Engell, S. Iterative set-point optimization of batch chromatography. *Comput. Chem. Eng.* 2005, 29, 1401–1409. [CrossRef]
16. Marchetti, A.; Chachuat, B.; Bonvin, D. Modifier-adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.* 2009, 48, 6022–6033. [CrossRef]
17. Gao, W.; Wenzel, S.; Engell, S. A reliable modifier-adaptation strategy for real-time optimization. *Comput. Chem. Eng.* 2016, 91, 318–328. [CrossRef]
18. Roberts, P. Broyden derivative approximation in ISOPE optimising and optimal control algorithms. *IFAC Proc. Vol.* 2000, 33, 293–298. [CrossRef]
19. Forbes, J.; Marlin, T.; MacGregor, J. Model adequacy requirements for optimizing plant operations. *Comput. Chem. Eng.* 1994, 18, 497–510. [CrossRef]
20. Forbes, J.; Marlin, T. Design cost: A systematic approach to technology selection for model-based real-time optimization systems. *Comput. Chem. Eng.* 1996, 20, 717–734. [CrossRef]
21. Bonvin, D.; Srinivasan, B. On the role of the necessary conditions of optimality in structuring dynamic real-time optimization schemes. *Comput. Chem. Eng.* 2013, 51, 172–180. [CrossRef]
22. Faulwasser, T.; Bonvin, D. On the Use of Second-Order Modifiers for Real-Time Optimization. *IFAC Proc. Vol.* 2014, 47, 7622–7628. [CrossRef]
23. Bunin, G.A.; François, G.; Bonvin, D. From Discrete Measurements to Bounded Gradient Estimates: A Look at Some Regularizing Structures. *Ind. Eng. Chem. Res.* 2013, 52, 12500–12513. [CrossRef]
24. François, G.; Bonvin, D. Use of Convex Model Approximations for Real-Time Optimization via Modifier Adaptation. *Ind. Eng. Chem. Res.* 2013, 52, 11614–11625. [CrossRef]
25. Ahmad, A.; Gao, W.; Engell, S. A study of model adaptation in iterative real-time optimization of processes with uncertainties. *Comput. Chem. Eng.* 2019, 122, 218–227. [CrossRef]
26. Gottu Mukkula, A.R.; Engell, S. Guaranteed Model Adequacy for Modifier Adaptation with Quadratic Approximation. In Proceedings of the 2020 European Control Conference (ECC), St. Petersburg, Russia, 12–15 May 2020; pp. 1037–1042.
27. Gao, W.; Hernández, R.; Engell, S. Real-time optimization of a novel hydroformylation process by using transient measurements in modifier adaptation. *IFAC-PapersOnLine* 2017, 50, 5731–5736. [CrossRef]
28. Ferreira, T.D.A.; François, G.; Marchetti, A.G.; Bonvin, D. Use of Transient Measurements for Static Real-Time Optimization. *IFAC-PapersOnLine* 2017, 50, 5737–5742. [CrossRef]
29. Rodriguez-Blanco, T.; Sarabia, D.; Pitarch, J.; de Prada, C. Modifier Adaptation methodology based on transient and static measurements for RTO to cope with structural uncertainty. *Comput. Chem. Eng.* 2017, 106, 480–500. [CrossRef]
30. Cadavid, J.; Hernández, R.; Engell, S. Speed-up of Iterative Real-Time Optimization by Estimating the Steady States in the Transient Phase using Nonlinear System Identification. *IFAC-PapersOnLine* 2017, 50, 11269–11274. [CrossRef]
31. Krishnamoorthy, D.; Jahanshahi, E.; Skogestad, S. Feedback Real-Time Optimization Strategy Using a Novel Steady-state Gradient Estimate and Transient Measurements. *Ind. Eng. Chem. Res.* 2018, 58, 207–216. [CrossRef]
32. Gottu Mukkula, A.R.; Wenzel, S.; Engell, S. Active Perturbation in Modifier Adaptation for Real Time Optimization to Cope with Measurement Delays. *IFAC-PapersOnLine* 2018, 51, 124–129. [CrossRef]
33. Gottu Mukkula, A.R.; Wenzel, S.; Engell, S. Active Perturbations Around Estimated Future Inputs in Modifier Adaptation to Cope with Measurement Delays. *IFAC-PapersOnLine* 2018, 51, 839–844. [CrossRef]
34. Brdyš, M.; Tatjewski, P. An Algorithm for Steady-State Optimizing Dual Control of Uncertain Plants. *IFAC Proc. Vol.* 2019, 37, 215–220. [CrossRef]
35. Gao, W.; Wenzel, S.; Engell, S. Modifier adaptation with quadratic approximation in iterative optimizing control. In Proceedings of the 2015 IEEE European Control Conference (ECC), Linz, Austria, 15–17 July 2015; pp. 2527–2532.
36. Wenzel, S.; Yfantis, V.; Gao, W. Comparison of regression data selection strategies for quadratic approximation in RTO. *Comput. Aided Chem. Eng.* 2017, 40, 1711–1716.
37. Gao, W.; Wenzel, S.; Engell, S. Integration of gradient adaptation and quadratic approximation in real-time optimization. In Proceedings of the 2015 34th Chinese Control Conference (CCC), Hangzhou, China, 28–30 July 2015; pp. 2780–2785.
38. Gottu Mukkula, A.R.; Ahmad, A.; Engell, S. Start-up and Shut-down Conditions for Iterative Real-Time Optimization Methods. In Proceedings of the 2019 Sixth Indian Control Conference (ICC), Hyderabad, India, 18–20 December 2019; pp. 158–163.
39. Williams, TJ.; Otto, R.E. A generalized chemical processing model for the investigation of computer control. *Trans. Am. Inst. Electr. Eng. Part I Commun. Electron.* 1960, 79, 458–473. [CrossRef]
40. Friedman, M. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Am. Stat. Assoc.* 1937, 32, 675–701. [CrossRef]
41. Gottu Mukkula, A.; Kern, S.; Salge, M.; Holtkamp, M.; Guhl, S.; Fleischer, C.; Meyer, K.; Remelhe, M.; Maiwald, M.; Engell, S. An Application of Modifier Adaptation with Quadratic Approximation on a Pilot Scale Plant in Industrial Environment. *IFAC-PapersOnLine* 2020, 53, 11773–11779. [CrossRef]
42. Bieringer, T.; Buchholz, S.; Kockmann, N. Future production concepts in the chemical industry: Modular–small-scale–continuous. *Chem. Eng. Technol.* 2013, 36, 900–910. [CrossRef]