Real-Time Optimization Based on Adaptation of Surrogate Models

Martand Singhal *, **Alejandro G. Marchetti** *, **Timm Faulwasser** *,** **and Dominique Bonvin** *

* *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*
e-mail: {martand.singhal, alejandro.marchetti, timm.faulwasser, dominique.bonvin}@epfl.ch
** *Institute for Applied Informatics, Karlsruhe Institute of Technology, D-76131 Karlsruhe, Germany*

Abstract: Recently, different real-time optimization (RTO) schemes that guarantee feasibility of all RTO iterates and monotonic convergence to the optimal plant operating point have been proposed. However, simulations reveal that these schemes converge very slowly to the plant optimum, which may be prohibitive in applications. This note proposes an RTO scheme based on second-order surrogate models of the objective and the constraints, which enforces feasibility of all RTO iterates, i.e., plant constraints are satisfied at all iterations. In order to speed up convergence, we suggest an online adaptation strategy of the surrogate models that is based on trust-region ideas. The efficacy of the proposed RTO scheme is demonstrated in simulations via both a numerical example and the steady-state optimization of the Williams-Otto reactor.

Keywords: real-time optimization, plant-model mismatch, plant feasibility, online adaptation

1. INTRODUCTION

To ensure profitability in the process industries, one typically maximizes the economic performance, while respecting safety and environmental constraints. Hence, maximizing plant performance in real time, known as real-time optimization (RTO), has gathered commendable industrial attention. The goal of RTO is to enforce plant optimality in the presence of uncertainty such as plant-model mismatch and process disturbances. RTO methods rely on the available (not necessarily very accurate) plant model and measurements to push the *plant* towards optimality.

In the design of RTO schemes, it is desirable to achieve the following properties: (i) plant optimality and feasibility upon convergence, (ii) acceptable number of RTO iterations, and (iii) plant feasibility throughout the optimization process. The works of Gao and Engell (2005), Chachuat et al. (2009), Marchetti et al. (2009, 2010) are tailored to (i). However, the proposed schemes depend on tuning parameters to enforce (ii) and cannot guarantee (iii). Furthermore, it has been proposed to estimate gradients via surrogate models (Bunin et al., 2013a; Gao et al., 2015).

Recently, Bunin et al. (2013b) suggested to rely on Lipschitz constants of the constraints and a Hessian upper bound of the cost to enforce plant feasibility and monotonic convergence in some kind of post-processing of RTO iterates. Furthermore, it has been shown in Singhal et al. (2015) that similar ideas can also be used to design data-driven RTO schemes based on linear-quadratic surrogate models, where the plant constraints are approximated in a linear-affine fashion and the plant objective is modeled as a quadratic function. Both Bunin et al. (2013b) and Singhal et al. (2015) guarantee plant feasibility of all RTO iterates, that is, they enforce (iii). However, simulations have shown that the (conservative) first-order approximation of plant constraints using Lipschitz constants often leads to slow convergence of the RTO algorithm, which may be prohibitive in applications.

The present paper investigates data-driven RTO based on surrogate models. We propose an RTO scheme based on quadratically-constrained quadratic programs (QCQP), which can be regarded as an extension of the scheme proposed in Singhal et al. (2015) based on quadratic programs (QP). The contributions of the present work are as follows: We sketch the feasibility and optimality properties of the proposed RTO scheme. Furthermore, we analyze why first-order constraint approximations based on Lipschitz constants may lead to overly slow convergence. Finally, we present a scheme for adapting second-order surrogate models and illustrate via simulation that this leads to considerably faster RTO convergence.

The paper is structured as follows. Section 2 briefly formulates the RTO problem. Section 3 introduces the QCQP-based surrogate model and draws a comparison between the QCQP and the QP surrogate models. The performance of the QCQP-based surrogate model is tested on a numerical example. Section 4 introduces the online adaptation algorithm for the upper bounding term and presents the gradient-estimation method. Section 5 presents the Williams-Otto reactor case study, illustrating the performance of the proposed RTO scheme. Finally, the conclusions are presented in section 6.

2. PLANT OPTIMIZATION PROBLEM

Steady-state performance improvement can be formulated mathematically as a nonlinear program

$$\min_{u} \phi(u, d) \tag{1a}$$
$$\text{s.t.} \quad g_j(u, d) \leq 0, \qquad j{=}1, .., n_g \tag{1b}$$
$$u^L \preceq u \preceq u^U, \tag{1c}$$

where u is the n_u-dimensional input vector, d is the n_d-dimensional vector of disturbances, $\phi : \mathbb{R}^{n_u} \times R^{n_d} \to \mathbb{R}$ is the plant cost, $g_j : \mathbb{R}^{n_u} \times R^{n_d} \to \mathbb{R}$ is the j^{th} plant constraint. The symbol ($\prec$) denotes component-wise inequality of vectors. The disturbance d models the fact that the plant cost and constraints are affected by disturbances. In this paper, however, we do not deal with the disturbance d explicitly. Instead, we assume that the plant cost ϕ and the plant constraints g_j are not exactly known.

3. RTO BASED ON SURROGATE MODELS

Next, we focus on RTO based on surrogate models to find the plant optimum. The surrogate models will enforce plant feasibility and guarantee monotonic plant cost decrease provided that the available measurements are noise free. To this end, we recall two technical lemmas presented in Bunin et al. (2013b).

Lemma 1. (Lipschitz upper bound). Let $f : \mathbb{R}^{n_u} \to \mathbb{R}$ be continuously differentiable over the compact set $\mathcal{U} \subset \mathbb{R}^{n_u}$ such that

$$-\lambda_i < \left.\frac{\partial f}{\partial u_i}\right|_u < \lambda_i, \quad \forall u \in \mathcal{U}, \quad i = 1, ..., n_u, \tag{2}$$

where λ are the univariate Lipschitz constants of f. Then, the evolution of f between the two successive inputs u_k and u_{k+1} is bounded by

$$f(u_{k+1}) \leq f(u_k) + \sum_{i=1}^{n_u} \lambda_i |u_{k+1,i} - u_{k,i}|. \tag{3}$$

□

Lemma 2. (Hessian upper bound). Let $f : \mathbb{R}^{n_u} \to \mathbb{R}$ be twice continuously differentiable over the compact set $\mathcal{U} \subset \mathbb{R}^{n_u}$ such that

$$-M_{ij} < \left.\frac{\partial^2 f}{\partial u_i \partial u_j}\right|_u < M_{ij}, \quad \forall u \in \mathcal{U}, \quad i, j = 1, ..., n_u.$$

Let $\Delta_{k+1} := u_{k+1} - u_k$. Then, the change in f between u_k and u_{k+1} can be bounded as

$$f(u_{k+1}) \leq f(u_k) + \nabla f(u_k)^T \Delta_{k+1} + \frac{1}{2} \Delta_{k+1}^T \bar{Q} \Delta_{k+1}, \tag{4}$$

where $\bar{Q} \succ 0$ is a diagonal matrix with the diagonal elements $\bar{Q}_{ii} = \sum_{j=1}^{n_u} M_{ij},\ i = 1, ..., n_u$. □

While the first lemma provides a first-order affine upper bound for any continuously differentiable function over any compact set, the second lemma gives a quadratic approximation that upper bounds twice continuously differentiable functions. Note that, in Lemma 2, the matrix $\bar{Q}$ can be seen as a Hessian upper bounding matrix for the function f over the compact set $\mathcal{U}$. We refer to Bunin et al. (2013b) for the proofs of these two lemmas.

3.1 RTO with QP surrogate model

Based on these two lemmas, the following QP-based surrogate model is proposed in Singhal et al. (2015)

$$\min_{\Delta_{k+1}} \quad \nabla \phi_k^T \Delta_{k+1} + \frac{1}{2} \Delta_{k+1}^T \bar{Q} \Delta_{k+1} \tag{5a}$$

subject to

$$g_j(u_k) + \sum_{i=1}^{n_u} \lambda_{i,j} |\Delta_{k+1,i}| \leq 0, \quad j = 1, ..., n_g, \tag{5b}$$
$$\nabla g_{j,k}^T \Delta_{k+1} \leq -\delta_j, \ \forall j \in \mathcal{J}_k, \tag{5c}$$
$$u^L - u_k \preceq \Delta_{k+1} \preceq u^U - u_k, \tag{5d}$$

where $\nabla \phi_k$ is the gradient of the plant cost function ϕ at u_k and $\nabla g_{j,k}$ is the gradient of the plant constraint g_j at u_k. The positive definite matrix $\bar{Q}$ is the Hessian upper bounding matrix for the cost function (computed as in Lemma 2 for $f := \phi$), $\lambda_{i,j}$ are the univariate Lipschitz constants of the plant constraint g_j. The set $\mathcal{J}_k$ is the set of ϵ-active constraints defined as (Bunin et al., 2013b)

$$\mathcal{J}_k = \{ j \in \{1, \ldots, n_g\} : -\epsilon_j \leq g_j(u_k) \leq 0 \},$$

where ϵ is a small positive scalar. The next RTO iterate u_{k+1} is given by

$$u_{k+1} = u_k + \Delta_{k+1}^\star, \tag{6}$$

where $\Delta_{k+1}^\star$ is the solution to Problem (5). The constraints (5b) ensure that u_{k+1} is a feasible point for the plant.

Singhal et al. (2015) showed that the RTO scheme based on (5)-(6) guarantees plant feasibility at all RTO iterations. However, this scheme may converge very slowly because of the conservatism induced by both the Hessian upper bound $\bar{Q}$ and the univariate Lipschitz constants $\lambda_{i,j}$. Furthermore, note that the surrogate model does not use information on the constraint gradient for ensuring feasibility in (5b), but only for keeping a distance from the active constraints in (5c), which may also contribute to the slow convergence.

3.2 RTO with QCQP surrogate model

In order to achieve faster convergence, we propose the following RTO scheme based on a QCQP surrogate model

$$\min_{\Delta_{k+1}} \nabla \phi_k^T \Delta_{k+1} + \frac{1}{2} \Delta_{k+1}^T \bar{Q} \Delta_{k+1} \tag{7a}$$

subject to

$$g_j(u_k) + \nabla g_{j,k}^T \Delta_{k+1} + \frac{1}{2} \Delta_{k+1}^T \bar{Q}_j \Delta_{k+1} \leq 0, \quad j = 1, ..., n_g, \tag{7b}$$
$$u^L - u_k \preceq \Delta_{k+1} \preceq u^U - u_k, \tag{7c}$$

where $\bar{Q}_j$ is the Hessian upper bounding matrix of the plant constraint function g_j. The input update is also given by (6) with $\Delta_{k+1}^\star$ being in this case the solution to Problem (7).

Remark 1. (Nominal plant feasibility).
The scheme (7) can be understood as an RTO-specific translation of the nominal optimization method suggested in (Auslender et al., 2010). Therein, it is shown that the QCQP formulation leads to recursively feasible iterates provided that the gradient information is exact and the following assumptions hold:

- The cost ϕ and the constraints $g_j, \quad j = 1,..,n_g$, are twice continuously differentiable on an open set containing the input space $\mathcal{U} = \{u \in \mathbb{R}^{n_u} : u^L \preceq u \preceq u^U\}$.
- The initial input u_0 is feasible with respect to the constraint functions g_j, i.e., $g_j(u_0) \leq 0$, $j = 1, ..., n_g$.
- The Mangasarian-Fromovitch constraint qualification (MFCQ) holds:
$\forall u \in \mathcal{U} : \exists d \in \mathbb{R}^n : \nabla g_j^T d < 0 \quad \forall j \in \mathcal{J}(u) := \{j \in [1, .., n_g] : g_j(u) = 0\}$.

Hence, using the surrogate model (7) is similar to the approach used in Auslender et al. (2010), and the feasibility proofs follow the same line of arguments. Note however that, in Auslender et al. (2010), gradient Lipschitz constants are used instead of the Hessian upper bound matrices $\bar{Q}$.

3.3 Matching of Lagrangian gradients

Both surrogate models (5) and (7) guarantee plant constraint feasibility. Yet, it not easy to tell which of the schemes will lead to faster convergence. To this end, consider the Lagrangian of the plant (1) at the input iterate u_k

$$L_k = \phi_k + \sum_{j=1}^{n_g} \mu_{j,k} g_{j,k} \, ,$$

where $\phi_k := \phi(u_k), g_{j,k} := g_j(u_k)$, and $L_k := L(u_k)$. The gradient of the plant Lagrangian is

$$\nabla L_k = \nabla \phi_k + \sum_{j=1}^{n_g} \mu_{j,k} \nabla g_{j,k}.$$

Next, consider the gradient of the Lagrangian of the QP surrogate model (5) evaluated at u_k,

$$\nabla L_k^{qp} = \nabla \phi_k + \sum_{j \in \mathcal{J}_k} \mu'_{j,k} \nabla g_{j,k} + \sum_{j=1}^{n_g} \sum_{m=1}^{2^{n_u}} \nu_{j,k}^m \Lambda_j S(m),$$

where $\mu'_{j,k}$, $\nu_{j,k}^m \geq 0$ are the Lagrange multipliers and $\Lambda_j = diag(\lambda(i,j))$, $i = 1, ..., n_u$. Here, $S(m)$ is the mth column of the $n_u \times 2^{n_u}$ matrix S defined as $S(i,m) := (-1)^{\lfloor \frac{m-1}{2^{(n_u - i)}} \rfloor}$, with $\lfloor \rfloor$ being the floor function. The Lagrangian gradient ∇L_k^{qp} does not match that of the plant (1).

Finally, consider the Lagrangian of the QCQP surrogate model (7) evaluated at u_k,

$$\nabla L_k^{qcqp} = \nabla \phi_k + \sum_{j=1}^{n_g} \mu_{j,k} \nabla g_{j,k}.$$

Note that all the terms depending on $\bar{Q}_j$ vanish, since $u = u_k$ implies $\Delta_{k+1} = 0$. Hence, we conclude that $\nabla L_k^{qcqp} = \nabla L_k$. In other words, the advantage of the QCQP surrogate model (7) is that its Lagrangian gradient matches the Lagrangian gradient of the plant problem (1) at u_k. Hence, using the QCQP surrogate model (7) gives a better representation of the plant at u_k, which may lead to faster convergence. Next, we compare the performances of the two surrogate models via a numerical example.

Example 1. (RTO with QP and QCQP models).
Consider the following plant optimization problem

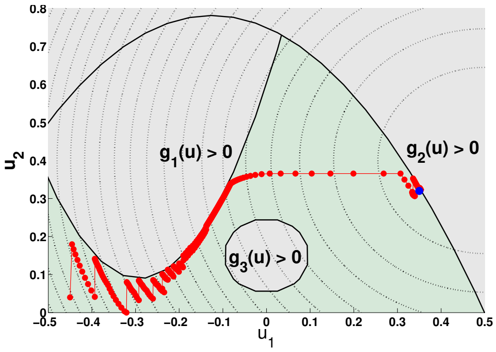

Fig. 1. RTO using the QP surrogate model (5) to optimize Problem (8). The RTO iterates are in red and remain strictly inside the feasible region shaded in green. The plant optimum is highlighted in blue.

$$\begin{aligned}
&\min_{u_1,u_2} (u_1 - 0.5)^2 + (u_2 - 0.4)^2 \\
&\text{subject to } g_1 = -6u_1^2 - 3.5u_1 + u_2 - 0.6 \leq 0 \\
&\qquad g_2 = 2u_1^2 + 0.5u_1 + u_2 - 0.75 \leq 0 \\
&\qquad g_3 = -u_1^2 - (u_2 - 0.5)^2 + 0.01 \leq 0 \\
&\qquad\quad -0.5 \leq u_1 \leq 0.5, \; 0 \leq u_2 \leq 0.8
\end{aligned} \tag{8}$$

The plant optimum is $u^\star = [0.35, 0.32]^T$. Starting from the initial feasible point $u_0 = [-0.5, 0.05]^T$, the RTO algorithms using the QP and QCQP surrogate models are implemented. For simplicity, we assume that the plant cost and constraint values, as well as their respective gradients, are available or can be measured at u_k for each k. The Lipschitz constants of the plant constraints taken from Singhal et al. (2015) are $\lambda_{1,1} = 10.45$, $\lambda_{2,1} = \lambda_{2,2} = \lambda_{1,3} = 1.1$, $\lambda_{1,2} = 2.75$, $\lambda_{2,3} = 1.43$. Initially, we take the cost Hessian upper bound matrix as $\bar{Q} = \text{diag}(2.2, 2.2)$. In order to find the plant optimum using the QP surrogate model (5), we use Algorithm 1 given in Singhal et al. (2015) and also take the same tuning parameters for all three constraints, $\bar{\epsilon} = 0.11$, $\bar{\delta} = 0.0002$, $\eta_\epsilon = 10^{-3}$ and $\eta_\delta = 10^{-10}$. The RTO result is shown in Figure 1, where it is seen that the iterates remain strictly inside the feasible region highlighted in green. Clearly, the plant optimum is reached, but only after a large number of iterations.

Next, we apply RTO based on the QCQP surrogate model (7). The Hessian upper bound of the objective is the same as the one for the QP surrogate model (5). The constraint Hessian upper bound matrices used are $\bar{Q}_1 = \bar{Q}_3 = \text{diag}(2.2, 2.2)$ and $\bar{Q}_2 = \text{diag}(4.2, 2.2)$. The results are shown in Figure 2. It can be seen that, as before, all RTO iterations remain strictly inside the feasible region. However, the number of iterations taken to reach the plant optimum has been significantly reduced to 7 iterations. This simple numerical example supports our analysis that the convergence speed is likely to increase by using the QCQP surrogate model (7) instead of the QP surrogate model (5).

However, note that the convergence speed depends on the values of the Hessian upper bound matrices. If the Hessian upper bounds are too conservative, then many more RTO iterations are required to reach the plant KKT point. For instance, let us take the cost and constraint Hessian upper bounds as $\bar{Q} = \bar{Q}_1 = \bar{Q}_2 = \bar{Q}_3 = \text{diag}(8, 8)$. In this case, the QCQP algorithm (7) requires

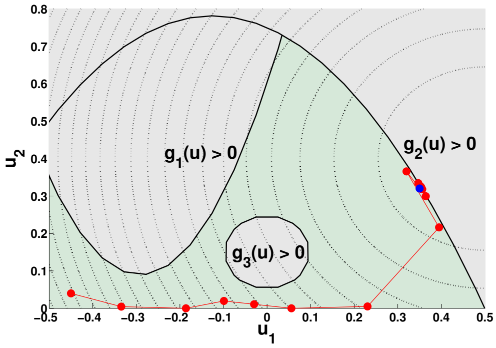

Fig. 2. RTO using the QCQP surrogate model (7) to optimize Problem (8). The RTO iterations are in red and remain strictly inside the feasible region shaded in green. The plant optimum is highlighted in blue.

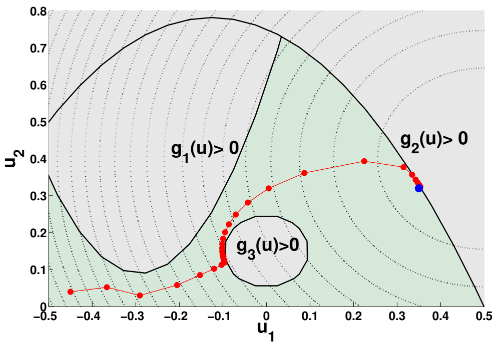

Fig. 3. RTO using the QCQP surrogate model (7) and conservative Hessian upper bounds to optimize Problem (8).

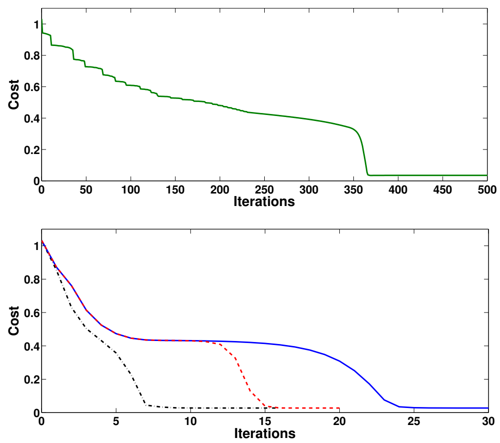

Fig. 4. Cost improvement over RTO iterations: the green line corresponds to the QP surrogate model (5); the black line is for the QCQP surrogate model (7) with non-conservative Hessian upper bound; the blue and red lines correspond to the conservative upper bound, without and with online adaptation, respectively, as described in Example 2 below.

a larger number of iterations to find the plant KKT point (24 iterations), as shown in Figure 3. The cost function evolutions corresponding to Figures 1, 2 and 3 are shown in Figure 4.

Note that the Hessians in Problem (8) are constant. In general, however, the Hessians of the cost and constraint functions depend on the operating. Hence, if the Hessian upper bounds are chosen to be globally valid over the entire input space, they may be overly conservative for the current RTO input.

4. ONLINE ADAPTATION OF HESSIAN BOUNDS

The preceding section motivates the idea of solving the QCQP surrogate model (7) using tight Hessian upper bounds that are valid locally in a neighborhood of the current input iterate u_k. A natural way of getting local approximations of the Hessian upper bounds is to use the existing plant model over a reduced input space. In order to account for plant-model mismatch, these bounds can be multiplied by an appropriate factor. However, if the confidence in the model is low, then an on-line adaptation strategy for the Hessian upper bounds might be more appropriate. We sketch next an intuitive way of adapting the Hessian upper bounds based on the trust-region ideas.

4.1 Adaptation of trust-region-like Hessian upper bounds

We suggest obtaining positive-definite Hessian upper bounds by fitting convex quadratic functions to past operating data. The Hessian upper bound of the fitted quadratic model, labeled D_k, is then combined with the previous Hessian upper bound, $\bar{Q}_{k-1}$, to give $\bar{Q}_k = \alpha D_k + (1-\alpha)\bar{Q}_{k-1}$. The weighing factor α can be updated at each iteration based on the agreement between the quadratic model and the plant, thereby giving α_k.

For this adaptation, consider the optimal value of the cost function of the surrogate quadratic model, $m_k := \nabla\phi_{k-1}^T \Delta_k^\star + \frac{1}{2}\Delta_k^{\star T}\bar{Q}\Delta_k^\star$, with $\Delta_k^\star$ representing the optimal input of Problem (7). Let m_k^{avg} and $\Delta\phi_k^{avg}$ denote the moving averages of the improvement in the cost function values of the surrogate model and the plant, respectively. These moving averages are taken over the latest N iterations. If the ratio

$$\rho_k := \frac{m_k^{avg}}{\Delta\phi_k^{avg}}$$

is small, then the Hessian upper bound is too conservative and the value of α can be increased. In contrast, if ρ_k is close to unity, the Hessian upper bound is rather close to the local value of the plant Hessian, and the value of α can be decreased. Note that such an adaptation is similar to (or inspired by) trust-region methods, since the Hessian upper bounds implicitly define the step length of the RTO scheme.

In order to formalize the RTO scheme based on the QCQP and the adaptation of the Hessian bounds, let $\mathcal{S}_k^{reg}$ and ϕ_k^{reg} represent the regression set and the convex quadratic regression function at the k^{th} iteration of the RTO algorithm. Let H_k be the Hessian of the regression function ϕ_k^{reg}. Consider the constants $0 < \underline{\alpha} < \alpha_0 < \bar{\alpha} < 1$, $0 < N, \beta$, and $0 < \eta_1 < \eta_2$. Define the maximal allowable step size in each input direction as $h_i, i = 1, ..., n_u$, and the initial Hessian upper bound on the cost, $\bar{Q}_0$. The procedure is defined in Algorithm 1.

The Hessian upper bounds of the constraints can be updated similarly by replacing the data points for the cost with the constraint values in the regression set $\mathcal{S}_k^{reg}$ and by computing the ratio ρ_k for the constraint function.

Algorithm 1 Adaptation algorithm

Step 0 (Initialization). Perform at least $((n_u+1)(n_u+2)/2)-1$ RTO iterations with the QCQP surrogate model (7) and generate the regression set $\mathcal{S}_k^{reg}$. Fit the convex quadratic function to the regression set and obtain ϕ_k^{reg}. Extract H_k from ϕ_k^{reg}. Set $\bar{Q}_k = \bar{Q}_0$ and $\alpha_k = \alpha_0$.

Step 1 (Choose α_{k+1}). Compute ρ_k by calculating the moving averages m_k^{avg} and $\Delta\phi_k^{avg}$ over the past N RTO iterations.

- If $\rho_k < \eta_1$, then $\alpha_{k+1} = \min\{2\alpha_k, \bar{\alpha}\}$,
- If $\rho_k > \eta_2$, then $\alpha_{k+1} = \max\{0.5\alpha_k, \underline{\alpha}\}$,
- Else $\alpha_{k+1} = \alpha_k$.

Step 2 (Hessian upper bound of the fitted model). Compute the diagonal matrix

$$D_{ii,k} = \sum_{j=1}^{n_u} H_{ij,k}, \quad i = 1, ..., n_u.$$

Step 3 (Hessian upper bound update).

- If $\rho_k < 0$ and $\phi_k - \phi_{k-1} < \beta$, then $\bar{Q}_{k+1} = \bar{Q}_0, \quad \alpha_{k+1} = \alpha_0$.
- Else $\bar{Q}_{k+1} = \alpha_{k+1} D_k + (1-\alpha_{k+1})\bar{Q}_k$.

Step 4 (RTO step size). Update the allowable input space as $u_{i,k} - h_i \le u_{i,k+1} \le u_{i,k} + h_i, \ i = 1, ..., n_u$.

Step 5: Increment k by one and run the RTO scheme with the updated Hessian upper bound. Update the regression set by adding the new data point to obtain the new set $\mathcal{S}_k^{reg}$, and compute the Hessian H_k. Return to Step 1.

Example 2. (RTO with QCQP model and adaptation of Hessian upper bounds) Consider the numerical example of Problem (8). Starting with $\bar{Q} = \bar{Q}_1 = \bar{Q}_2 = \bar{Q}_3 = \mathrm{diag}(8, 8)$, the Hessian upper bounds of the cost and constraint functions are adapted using Algorithm 1. The result is shown in Figure 5. In this case, it takes 15 iterations compared to the 24 iterations required when the Hessian upper bounds are not adapted, as shown in Figure 4.

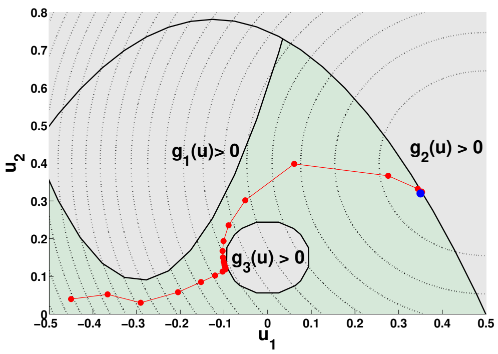

Fig. 5. RTO performance for Problem (8) when the Hessian upper bounds are adapted online.

4.2 Combining gradient estimation and Hessian update

Note that the presence of measurement noise may cause difficulties in accurately capturing the Hessian information. To ensure that the regression set is well-poised, and thus the curvature information is well captured by the noisy data, we can introduce additional constraints in the optimization Problem (7). These constraints were originally designed to estimate the gradient information from past RTO iterations in the presence of noise (Marchetti et al., 2010). Including them in the RTO problem ensures that the next RTO iterate is located such that accurate gradient information can be extracted. Hence, in addition to extracting Hessian information via regression, we use the past RTO data to estimate the gradient for the next RTO iteration. For the sake of completeness, we briefly recall the gradient estimation that is used in Marchetti et al. (2010).

The use of the QCQP surrogate model (7) requires plant gradient information for the cost and constraints. The classical way of extracting gradient information is by performing forward finite differences at each RTO iteration. Alternatively, and in order to avoid these additional plant perturbations, the data recorded during the previous RTO iterations can also be used to estimate the gradient in the following way:

$$\nabla\hat{y}_k = {U_k}^{-1} Y_k,$$

where $\nabla\hat{y}_k$ is the gradient estimate at u_k, and the matrices Y_k and U_k are defined as:

$$Y_k := [y(u_k) - y(u_{k-1}) \quad \dots \quad y(u_k) - y(u_{k-n_u})]^T \in \mathbb{R}^{n_u},$$
$$U_k := [u_k - u_{k-1} \quad \dots \quad u_k - u_{k-n_u}] \in \mathbb{R}^{n_u \times n_u}.$$

The matrix U_k, which is a function of u_k, is required to be non-singular. Note that y can represent the cost or the constraint functions.

To ensure well-poisedness of the data, bounds on the gradient error norm are introduced. The gradient error consists of truncation error and measurement error. The following constraint can then be enforced to ensure that u_{k+1} is located such that the gradient error norm is bounded by $\varepsilon_{upper} > 0$:

$$\frac{\sigma_{max}}{2} ||U_k^{-1} \,\mathrm{diag}(U_k U_k^T)|| + \frac{\delta_{noise}}{l_k^{min}} \le \varepsilon_{upper}, \qquad (9)$$

where the first term on the left-hand side is an upper bound on the truncation error, with σ_{max} being an upper bound on the spectral radius of the Hessian of the function y whose gradient is estimated. The second term on the left-hand side is an upper bound on the noise error, with l_k^{min} being the shortest distance between all possible pairs of complement affine subspaces generated from $S := \{u_k, u_{k-1}, ..., u_{k-n_u}\}$ and δ_{noise} the accepted range of measurement noise. Note that ε_{upper} is user defined. Note also that the non-convex constraint (9) is added to the otherwise convex optimization Problem (7). It turns out that an additional constraint can be added to maintain convexity of the optimization problem, as illustrated in Marchetti et al. (2010).

5. ILLUSTRATIVE EXAMPLE

The Williams-Otto reactor (Williams and Otto, 1960) has been used as a benchmark problem to gauge the performance of different RTO schemes in (Roberts and Williams, 1981; Marchetti et al., 2010; Navia, 2012). The isothermal reactor consists of an ideal continuous-stirred tank reactor, where the reactants A and B are fed with the mass flowrates F_A and F_B, respectively, the

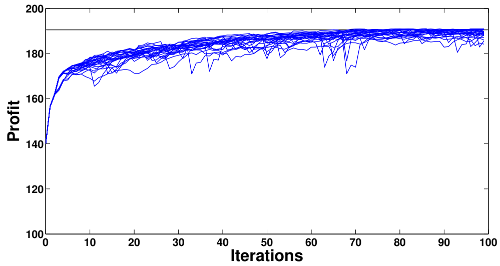

Fig. 6. Profit as a function of the iteration number when the matrix $\bar{Q}$ is not updated. 25 simulation runs are shown, each one with a different noise realization.

desired products P and E are formed together with the intermediate product C and the undesired product G. The mass flowrate of the product is $F = F_A + F_B$. The following reactions occur in the plant (here simulated reality):

$$\begin{aligned} &A + B \rightarrow C, && \bar{k}_1 = \bar{\eta}_1 e^{-\frac{\bar{\nu}_1}{T_r}} \\ &B + C \rightarrow P + E, && \bar{k}_2 = \bar{\eta}_2 e^{-\frac{\bar{\nu}_2}{T_r}} \\ &C + P \rightarrow G, && \bar{k}_3 = \bar{\eta}_3 e^{-\frac{\bar{\nu}_3}{T_r}}. \end{aligned}$$

The optimization objective considers profit maximization at steady state. The decision variables are the flowrate F_B and the reactor temperature T_R. The RTO problem takes the following form:

$$\max_{F_B, T_R} \phi = (1143.38 X_P + 25.92 X_E)(F_A + F_B) - 114.34 F_B,$$

$$3 \leq F_B(kg/s) \leq 6, \quad 70 \leq T_R(^\circ C) \leq 100,$$

where X_P and X_E are the steady-state concentrations of the products P and E, respectively. These concentrations are functions of the decision variables F_B and T_R. Note that the flowrate of reactant A is fixed at $F_A = 1.8275$ kg/s.

We initialize the problem at $F_B = 4.78$ kg/s and $T_R = 77^\circ C$. We calculate the Hessian upper bound for the process model taken from Marchetti (2009) to approximate the Hessian upper bound of the plant. For this purpose, we restrict the input space over which the cost function is optimized. The values of the diagonal elements found for the Hessian upper bound are $\bar{Q}(1,1) = 120$ for the feedrate F_B and $\bar{Q}(2,2) = 20$ for the temperature T_R.

In order to test the performance of the QCQP surrogate model (7), we add random noise to the plant cost values, with Gaussian distribution, zero mean and standard deviation of 0.5. The gradients are estimated using the method described in Section 4.2. The values of the parameters required to evaluate the constraint (9) are taken from Marchetti et al. (2010). The two inputs F_B, and T_R are scaled with the scaling factors $\frac{1}{3}$ and $\frac{1}{30}$, respectively. The Hessian upper bound $\bar{Q}$ is scaled accordingly. Result of the RTO scheme without adaptation of $\bar{Q}$ is shown in Figure 6. The result consists of 25 simulation runs for different noise realizations. We do not apply any stopping criterion for the simulations. It takes close to 60 iterations on average to reach and stay in the vicinity of the plant optimum. Figure 7 shows the performance of the RTO scheme with adaptation of $\bar{Q}$. Clearly, the vicinity of the plant optimum is reached faster, in about 20 iterations on average. The parameter values taken for the online adaptation algorithm are $N = 4$, $\eta_1, \underline{\alpha} = 0.25$, $\eta_2 = 0.75$, $\bar{\alpha} = 0.5$, $\alpha_0 = 0.4$, $\beta = -1.5$ and the scaled step sizes are $h_1 = 0.08$, $h_2 = 0.043$.

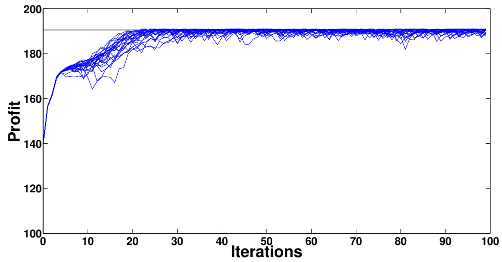

Fig. 7. Profit as a function of the iteration number when the matrix $\bar{Q}$ is updated. 25 simulation runs are shown, each one with a different noise realization.

6. CONCLUSIONS

This paper has presented an RTO scheme based on quadratic surrogate models that are built to upper bound the plant cost and constraint functions. Simulations have shown that this scheme ensures plant feasibility and speeds up convergence significantly. Furthermore, we have introduced a trust-region-like adaptation of the Hessian upper bounds that achieves even faster convergence. Future work will focus on the formal analysis of the feasibility properties.

REFERENCES

Auslender, A., Shefi, R., and Teboulle, M. (2010). A moving balls approximation method for a class of smooth constrained minimization problems. *SIAM Journal on Optimization*, 20(6), 3232–3259.

Bunin, G., François, G., and Bonvin, D. (2013a). From discrete measurements to bounded gradient estimates: A look at some regularizing structures. *I&EC Research*, 52(35), 12500–12513.

Bunin, G., François, G., and Bonvin, D. (2013b). Sufficient conditions for feasibility and optimality of real-time optimization schemes - i. Theoretical foundations. ArXiv:1308.2620.

Chachuat, B., Srinivasan, B., and Bonvin, D. (2009). Adaptation strategies for real-time optimization. *Comput. Chem. Engng*, 33(10), 1557–1567.

Gao, W. and Engell, S. (2005). Iterative set-point optimization of batch chromatography. *Comput. Chem. Engng*, 29(6), 1401–1409.

Gao, W., S, W., and S, E. (2015). Modifier adaptation with quadratic approximation in iterative optimizing control. In *Proceedings of the 2015 European Control Conference*.

Marchetti, A. (2009). *Modifier-Adaptation Methodology for Real-Time Optimization*. EPFL thesis, no 4449, Ecole Polytechnique Fédérale de Lausanne. doi:10.5075/epfl-thesis-4449.

Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *I&EC Research*, 48(13), 6022–6033. doi:10.1021/ie801352x.

Marchetti, A., Chachuat, B., and Bonvin, D. (2010). A dual modifier-adaptation approach for real-time optimization. *J. Process Control*, 20(9), 1027–1037.

Navia, D. (2012). *Handling uncertainties in process optimization*. Ph.D. thesis, Universidad de Valladolid.

Roberts, P. and Williams, T. (1981). On an algorithm for combined system optimisation and parameter estimation. *Automatica*, 17(1), 199–209.

Singhal, M., Faulwasser, T., and Bonvin, D. (2015). On handling cost gradient uncertainty in real-time optimization. *IFAC-PapersOnLine*, 48(8), 176 – 181.

Williams, T. and Otto, R. (1960). A generalized chemical processing model for the investigation of computer control. *Trans. American Institute of Electrical Engineers, Part I: Communication and Electronics*, 79(5), 458–473.