On the Use of Second-Order Modifiers for Real-Time Optimization

Timm Faulwasser and Dominique Bonvin

Laboratoire d'Automatique, Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne Switzerland,
e-mail: {timm.faulwasser, dominique.bonvin}@epfl.ch

Abstract: We consider the real-time optimization of static plants and propose a generalized version of the modifier-adaptation strategy that relies on second-order adaptation of the cost and constraint functions. We show that second-order adaptation allows checking whether a (local) plant optimum is reached upon convergence. A sufficient convergence condition that is applicable to first- and second-order modifier-adaptation schemes is proposed. We also discuss how second-order updates can lead to SQP-like model-free RTO schemes. The approach is illustrated via the simulated example of a continuous reactor.

Keywords: real-time optimization, modifier adaptation, convergence, Hessian approximation

1. INTRODUCTION

The aim of real-time optimization (RTO) is to enforce plant optimality despite uncertainty in the form of plant-model mismatch and disturbances. Instead of searching for a robust solution to the problem, RTO methods typically rely on measurements to push the *plant* toward optimality. In principle, one can (i) update the uncertain model parameters and repeat the optimization with the updated model (Jang et al., 1987; Chachuat et al., 2009), (ii) compute correction terms to modify the optimization problem and enforce plant optimality (Gao and Engell, 2005; Marchetti et al., 2009), or (iii) use feedback control to adapt the inputs directly (Skogestad, 2000; Srinivasan and Bonvin, 2007).

This paper considers option (ii) as it investigates the use of modifier adaptation for solving static RTO problems. The main idea of modifier adaptation is to perform (affine) corrections to the cost and constraint functions based on appropriate plant measurements (Marchetti et al., 2009, 2010). An appealing feature of modifier adaptation is that the plant reaches a KKT point upon convergence. The concept of modifier adaptation can be regarded as a generalization of previous works (Roberts and Williams, 1981; Brdyś and Tatjewski, 2005; Gao and Engell, 2005; Chachuat et al., 2008).

Although a necessary condition for local asymptotic convergence is given in Marchetti et al. (2009), sufficient conditions are still missing. Some elements toward sufficiency have been proposed, for example by describing the iterative scheme as a dynamical system and looking for a local Lyapunov function (Chachuat et al., 2008). Recently, Bunin (2014) proposed to look at the equivalence between the modifier-adaptation and trust-region frameworks. Modifier-adaptation is shown to be a special case of the trust-region approach. This relation is then exploited to propose a globally convergent modifier-adaptation algorithm using already developed trust-region theory.

This paper proposes the use second-order corrections in the context of modifier adaptation and investigates its contribution in terms of convergence and accuracy. The main contributions are as follows:

(1) It is shown that the presence of second-order correction terms allows assessing whether, upon convergence, a local minimum of the plant is attained. A sufficient condition for convergence of modifier-adaptation schemes of varying orders is presented.
(2) We implement second-order modification by estimating the plant Hessians via Hessian approximations known from quasi-Newton methods and finite-difference approximations of the plant gradients.
(3) We sketch how second-order information can be used to design SQP-like model-free RTO schemes.

The paper is structured as follows. Useful results from fixed-point theory are briefly reviewed in Section 2. Section 3 presents an approach to second-order modifier adaptation, while Section 4 illustrates the approach via a simulated example.

2. TECHNICAL PRELIMINARIES

Notations

The Euclidean norm of the vector $x \in \mathbb{R}^{n_x}$ is written $\|x\|$. The sequence of real vectors x_k, $k \in \mathbb{N}$, is written $(x_k)_{k\in\mathbb{N}}$. I is the identity matrix. For $A \in \mathbb{R}^{n_x \times n_x}$, $\|A\|$ denotes the induced 2-norm. The (minimal) vectorization of the symmetric matrix $A = A^T \in \mathbb{R}^{n_x \times n_x}$ is written $vec\,(A) \in \mathbb{R}^{n_x(n_x+1)/2}$.

A Useful Result from Fixed-Point Theory

The convergence analysis of modifier-adaptation schemes will rely on a well-known result from fixed-point theory. We briefly review the necessary concepts and results here. For a broad overview of fixed-point theory, the reader is referred to Dugundji and Granas (1982). For the sake of simplicity, the special case of fixed-point maps that live on a nonempty convex subset $\mathcal{C}$ of $\mathbb{R}^{n_x}$ are considered.

Definition 1. (Contractive map). The map $\Gamma : \mathcal{C} \to \mathcal{C}$ is called

- strictly contractive, if there exists $k < 1$ such that
$$\forall x, y \in \mathcal{C}: \quad \|\Gamma(x) - \Gamma(y)\| \leq k\|x - y\|, \tag{1a}$$
- nonexpansive (contractive), if
$$\forall x, y \in \mathcal{C}: \quad \|\Gamma(x) - \Gamma(y)\| \leq \ \|x - y\|. \tag{1b}$$

Theorem 1. (Convergence of averaged operators).
Let $\Gamma : \mathcal{C} \to \mathcal{C}$ be a nonexpansive operator with at least one fixed point on $\mathcal{C}$ and $(\alpha_k)_{k\in\mathbb{N}}$ a sequence of real numbers on $[0, 1]$ such that
$$\sum_{k\in\mathbb{N}} \alpha_k(1 - \alpha_k) = +\infty.$$
Furthermore, let $x_0 \in \mathcal{C}$ and set
$$x_{k+1} = (1 - \alpha_k)x_k + \alpha_k\Gamma(x_k). \tag{2}$$
Then, the sequence $(\Gamma(x_k) - x_k)_{k\in\mathbb{N}}$ converges to 0.

This result is known as the Krasnoselski-Mann theorem, see Theorem 5.14 in Bauschke and Combettes (2011). Extensions for specific choices of the averaging sequences $(\alpha_k)_{k\in\mathbb{N}}$ can be found in Johnson (1972); Ishikawa (1974).

3. SECOND-ORDER MODIFIER ADAPTATION

This section will successively present modifier adaptation of various orders, discuss the matching properties between the plant and the modified model for second-order modification, provide a guarantee that the plant has reached a local minimum, provide sufficient conditions for convergence, and discuss an approach for estimating the experimental Hessians. Furthermore, we sketch how second-order adaptation is linked to SQP methods.

3.1 Modifier Adaptation of Various Orders

Real-time optimization attempts to enforce optimal operation for a given plant despite the presence of uncertainty (Chachuat et al., 2009). In the case of a static plant, $\phi_p : \mathcal{U} \to \mathbb{R}$, with $\mathcal{U} \subset \mathbb{R}^{n_u}$, this can be formally described as

$$\underset{u}{\text{minimize}} \quad \phi_p(u) \tag{3a}$$
subject to
$$g_{p,j}(u) \leq 0, \quad j = 1, \ldots, n_g \tag{3b}$$
$$u \in [\underline{u}, \overline{u}] \subset \mathbb{R}^{n_u}. \tag{3c}$$

In real applications, however, neither the plant cost *function* ϕ_p nor the constraint *functions* $g_{p,j}$ are known exactly. One typically relies on the approximations ϕ and g_j, which leads to the following model-based optimization problem:

$$\underset{u}{\text{minimize}} \quad \phi(u) \tag{4a}$$
subject to
$$g_j(u) \leq 0, \quad j = 1, \ldots, n_g \tag{4b}$$
$$u \in [\underline{u}, \overline{u}] \subset \mathbb{R}^{n_u}. \tag{4c}$$

However, due to plant-model mismatch and disturbances, solving problem (4) does not guarantee that a KKT point (local minimum) of the plant will be reached (Forbes and Marlin, 1996; François and Bonvin, 2013a). For this purpose, the optimization problem (4) is modified as follows and solved iteratively:

$$\underset{u}{\text{minimize}} \quad \phi(u) + \sum_{i=0}^{i_\phi} m_\phi^i \left(\lambda_{\phi,k}^i, u - u_k\right) \tag{5a}$$
subject to
$$g_j(u) + \sum_{i=0}^{i_g} m_{g_j}^i \left(\lambda_{g_j,k}^i, u - u_k\right) \leq 0, \quad j = 1, \ldots, n_g \tag{5b}$$
$$u \in [\underline{u}, \overline{u}] \subset \mathbb{R}^{n_u}, \tag{5c}$$

with the subscript $\cdot_k$ indicating a value corresponding to the k^{th} iteration.

The scalar modifiers $m_l^i,\ i \in \{0, 1, 2\},\ l \in \{\phi, g_j\}$ are:

$$m_\phi^0 \left(\lambda_{\phi,k}^0, u - u_k\right) = \lambda_{\phi,k}^0 \tag{6a}$$
$$m_\phi^1 \left(\lambda_{\phi,k}^1, u - u_k\right) = (\lambda_{\phi,k}^1)^T (u - u_k) \tag{6b}$$
$$m_\phi^2 \left(\lambda_{\phi,k}^2, u - u_k\right) = \frac{1}{2}(u - u_k)^T \lambda_{\phi,k}^2 (u - u_k) \tag{6c}$$
$$m_{g_j}^0 \left(\lambda_{g_j,k}^0, u - u_k\right) = \lambda_{g_j,k}^0 \tag{6d}$$
$$m_{g_j}^1 \left(\lambda_{g_j,k}^1, u - u_k\right) = (\lambda_{g_j,k}^1)^T (u - u_k) \tag{6e}$$
$$m_{g_j}^2 \left(\lambda_{g_j,k}^2, u - u_k\right) = \frac{1}{2}(u - u_k)^T \lambda_{g_j,k}^2 (u - u_k), \tag{6f}$$

where the *modifiers* $\lambda_l^i,\ i \in \{0, 1, 2\},\ l \in \{\phi, g_j\}$ are defined as follows: $\lambda_{\phi,k}^0$ is the difference between the plant and model cost functions, [1]

$$\lambda_{\phi,k}^0 = \phi_p(u_k) - \phi(u_k), \tag{7a}$$

λ_ϕ^1 is the difference in cost gradient,

$$\lambda_{\phi,k}^1 := \left.\frac{\partial \phi_p}{\partial u}\right|_{u_k} - \left.\frac{\partial \phi}{\partial u}\right|_{u_k}, \tag{7b}$$

and $\lambda_{\phi,k}^2$ is the difference in cost Hessian,

$$\lambda_{\phi,k}^2 := \left.\frac{\partial^2 \phi_p}{\partial u^2}\right|_{u_k} - \left.\frac{\partial^2 \phi}{\partial u^2}\right|_{u_k}. \tag{7c}$$

Similarly, $\lambda_{g_j,k}^0 := g_{p,j}(u_k) - g_j(u_k)$ is the difference between the measured value of the j^{th} plant constraint and its predicted value, $\lambda_{g_j,k}^1$ is the difference in the j^{th} constraint gradient, and $\lambda_{g_j}^2$ is the difference in the j^{th} constraint Hessian.

To simplify the notation, we collect all the modifiers in one vector $\Lambda \in \mathbb{R}^{n_\Lambda},\ n_\Lambda = (n_g + 1)\left(n_u + \frac{n_u(n_u+1)}{2}\right) + n_g + 1$,

$$\Lambda_k := \Big(\lambda_{\phi,k}^0, \lambda_{\phi,k}^1, vec\left(\lambda_{\phi,k}^2\right), \lambda_{g_j,k}^0, \lambda_{g_j,k}^1, vec\left(\lambda_{g_j,k}^2\right)\Big)^T, j = 1, \ldots, n_g. \tag{8}$$

Note that the optimal solution to the modified problem (5) is not applied as such to the plant but filtered, or averaged, at each RTO iteration to account for noisy measurements:

[1] The offset in cost function does not affect the solution to Problem (5). It is merely introduced here to unify the presentation.

$$u_{k+1} = (1-\alpha)u_k + \alpha\, u^\star(u_k, \Lambda_k), \quad \alpha \in (0,1], \tag{9}$$

where $u^\star$ denotes the solution to the modified problem (5) for given values of u_k and Λ_k. In order to simplify the analysis to come, we use the scalar filter gain $\alpha \in (0,1]$. [2]

We are now ready to formally define the order of a modifier-adaptation scheme.

Definition 2. (Order of modifier adaptation). The 2-tuple

$$\Omega = (i_\phi, i_g), \qquad i_\phi, i_g \in \{0,1,2\},$$

which defines the summation indices in (5a) and (5b), is called the *order of the modifier-adaptation scheme* (9).

With the adaptation order $\Omega = (0,0)$, the only difference between Problems (4) and (5) are the cost and constraint offsets $\lambda^0_{\phi,k}$ and $\lambda^0_{g_j,k}$. This case is similar to (Forbes and Marlin, 1994) and (Chachuat et al., 2008). Using first-order adaptation with $\Omega = (1,1)$ leads to affine corrections to the cost and constraints as proposed in Gao and Engell (2005) and (Marchetti et al., 2009). The case $\Omega = (2,2)$ brings second-order corrections to the cost and constraints. Hence, we propose to call this scheme *second-order modifier adaptation*. Furthermore, one could also consider a *mixed second- and first-order adaptation* with $\Omega = (2,1)$, that is, one can use $i_\phi = 2$ for the cost function adaptation and $i_g = 1$ for the constraint adaptation.

3.2 Matching Properties for Second-Order Adaptation

To investigate the properties of second-order adaptation, the following assumptions are made.

Assumption 1. (Cost and constraint functions). The plant and model cost and constraint functions $\{\phi_p, g_{p,j}, \phi, g_j\}, j \in \{1,\ldots,n_g\}$ are twice continuously differentiable.

Recall that the static optimization problem (5) is called *feasible* if the set of inputs that satisfies the constraints (5b-c) is non-empty. A set of inputs is called *admissible* for Problem (5) if it satisfies the constraints. The feasible set of Problem (5) at the $(k+1)^{st}$ iteration can be written as

$$\mathcal{U}(u_k) = \Bigg\{ u \in [\underline{u}, \overline{u}] \mid j \in \{1,\ldots,n_g\} \\ g_j(u) + \sum_{i=0}^{i_g} m^i_{g_j}\left(\lambda^i_{g_j,k}(u_k), u - u_k\right) \le 0 \Bigg\}. \tag{10}$$

Since the set $\mathcal{U}(u_k)$ will typically change at each RTO iteration, we require the following assumption.

Assumption 2. (Feasibility). Problem (5) is globally feasible, that is, $\forall u \in [\underline{u}, \overline{u}] : \mathcal{U}(u) \neq \emptyset$.

This assumption does not imply that the RTO iterates u_k are admissible with respect to the plant constraints. [3]

[2] Similarly to Marchetti et al. (2009), one could also use the filter gain matrix $K \in \mathbb{R}^{n_u \times n_u}$ or filter the modifiers instead of the inputs.
[3] Guaranteeing strict admissibility of the RTO iterates with respect to the plant constraints is in general difficult. Bunin et al. (2013b) presents one way of enforcing admissibility on the basis of Lipschitz bounds for the plant.

A simple way of satisfying Assumption 2 is to consider the constraints (5b) as soft constraints in the numerical solution to Problem (5).

Assumption 3. (Uniqueness of solution). For all $u \in \mathcal{U}([\underline{u}, \overline{u}])$, Problem (5) has a unique solution.

This ensures that, for all $u \in \mathcal{U}([\underline{u}, \overline{u}])$, the optimal solution $u^\star(u, \Lambda(u))$ is a singleton. Note that the use of strictly convex models for first-order modifier adaptation as suggested in François and Bonvin (2013b) ensures the validity of Assumption 3.

It is well known that first-order modifier adaptation has the property of meeting the plant KKT conditions upon convergence (Marchetti et al., 2009). It is not surprising that the second-order scheme has similar properties, which are formalized in the next proposition.

Proposition 2. (KKT and Hessian matching). Assume that the second-order modifier-adaptation scheme given by Problem (5) with $\Omega = (2,2)$ has converged with $\lim_{k\to\infty} u_k = u_\infty$. Let a linear independence constraint qualification hold at u_∞. Then, the following matching properties hold:

(i) u_∞ satisfies the KKT conditions for the plant optimization problem (3);
(ii) the cost and constraint Hessians of the modified problem (5) match those of the plant (3).

Proof. We recall the main steps presented in Marchetti et al. (2009) for convergence of a first-order modifier-adaptation scheme. Let us denote the converged values of the modifiers as

$$\Lambda_\infty := \left(\lambda^0_{\phi,\infty}, \lambda^1_{\phi,\infty}, vec\left(\lambda^2_{\phi,\infty}\right), \right. \\ \left. \lambda^0_{g_j,\infty}, \lambda^1_{g_j,\infty}, vec\left(\lambda^2_{g_j,\infty}\right)\right)^T, j = 1,\ldots,n_g \tag{11}$$

and rewrite the cost and constraints of the modified problem as

$$\phi_m(u) := \phi(u) + \sum_{i=0}^{i_\phi} m^i_\phi\left(\lambda^i_{\phi,k}, u - u_k\right)$$
$$g_{m,j}(u) := g_j(u) + \sum_{i=0}^{i_g} m^i_{g_j}\left(\lambda^i_{g_j,k}, u - u_k\right).$$

Using the definitions given in (6) and (7), one can write for the converged point u_∞:

$$\left.\frac{\partial \phi_m}{\partial u}\right|_{u_\infty} = \left.\frac{\partial \phi}{\partial u}\right|_{u_\infty} + \lambda^1_{\phi,\infty} = \left.\frac{\partial \phi_p}{\partial u}\right|_{u_\infty}$$

$$g_{m,j}(u_\infty) = g_j(u_\infty) + \lambda^0_{g_j,\infty} = g_{p,j}(u_\infty)$$

$$\left.\frac{\partial g_{m,j}}{\partial u}\right|_{u_\infty} = \left.\frac{\partial g_j}{\partial u}\right|_{u_\infty} + \lambda^1_{g_j,\infty} = \left.\frac{\partial g_{p,j}}{\partial u}\right|_{u_\infty}.$$

From this and the assumption that u_∞ satisfies a linear independence constraint qualification, it can be inferred that u_∞ meets the KKT conditions of the plant (Bazaraa et al., 2006; Marchetti et al., 2009). This proves part (i).

Part (ii) follows similarly from

$$\left.\frac{\partial^2\phi_m}{\partial u^2}\right|_{u_\infty} = \left.\frac{\partial^2\phi}{\partial u^2}\right|_{u_\infty} + \lambda^2_{\phi,\infty} = \left.\frac{\partial^2\phi_p}{\partial u^2}\right|_{u_\infty}$$
$$\left.\frac{\partial^2 g_{m,j}}{\partial u^2}\right|_{u_\infty} = \left.\frac{\partial^2 g_j}{\partial u^2}\right|_{u_\infty} + \lambda^2_{g_j,\infty} = \left.\frac{\partial^2 g_{p,j}}{\partial u^2}\right|_{u_\infty}.$$

□

Remark 1. (Active set). As a consequence of Proposition 2, we can infer that Problem (5) and the plant optimum share the same set of active constraints. Furthermore, the Lagrange multipliers that solve the KKT conditions for Problem (5) at u_∞ are also solution to the KKT conditions of Problem (3).

3.3 Convergence to Plant Optimum

We show next that second-order adaptation allows checking whether, upon convergence, the plant indeed reaches a (local) optimum. For this, consider the restricted Lagrangian function $L_m : \mathbb{R}^{n_u} \times \mathbb{R}^{\hat{j}} \to \mathbb{R}$

$$L_m(u,\mu_j) = \phi_m(u) + \sum_{j\in J} \mu_j\, g_{m,j}(u)\,,$$

where $J \subset \mathbb{N}$ denotes the set of active constraints at u_∞ and $\hat{j}$ is the cardinality of J at u_∞.

Next, let J^+ and J^0 denote the sets of strongly active and weakly active constraints, respectively. The cone $\mathcal{C}_m(u) \subset \mathbb{R}^{n_u}$ is defined as

$$\mathcal{C}_m(u) = \Bigg\{ d \in \mathbb{R}^{n_u} \mid d \neq 0, \forall j \in J^+ : \left(\frac{\partial g_{m,j}}{\partial u}\right)^T d = 0, \quad \forall j \in J^0 : \left(\frac{\partial g_{m,j}}{\partial u}\right)^T d \leq 0, \Bigg\}.$$

The corresponding function $L_p : \mathbb{R}^{n_u} \times \mathbb{R}^{\hat{j}} \to \mathbb{R}$ and the cone $\mathcal{C}_p(u) \subset \mathbb{R}^{n_u}$ are defined mutatis mutandis for the plant optimization problem (3).

Proposition 3. (Convergence to a local plant minimum). Assume that the second-order modifier-adaptation scheme given by Problem (5) with $\Omega = (2,2)$ has converged with $\lim_{k\to\infty} u_k = u_\infty$. Let the converged value be a strict local minimum that satisfies

$$\forall d \in \mathcal{C}_m(u_\infty): \quad d^T \left.\frac{\partial^2 L_m}{\partial u^2}\right|_{u_\infty} d > 0. \qquad (12)$$

Then, u_∞ is a local minimizer of the plant optimization problem (3), and $\phi_p(u_\infty)$ is a strict local minimum of $\phi_p(u)$.

Proof. Condition (12) is sufficient to guarantee that $\phi_m(u_\infty)$ is a strict local minimum of Problem (5) (Bazaraa et al., 2006, Thm. 4.4.2). We know from Proposition 2 that, upon convergence, the modified optimization problem and the plant will have matching cost and constraint gradients and Hessians. Furthermore, the set of active constraints and the Lagrange multipliers will also match, and thus $\mathcal{C}_m(u_\infty) = \mathcal{C}_p(u_\infty)$. It can be inferred from (12) that an equivalent condition holds for the plant, i.e.

$$\forall d \in \mathcal{C}_p(u_\infty): \quad d^T \left.\frac{\partial^2 L_p}{\partial u^2}\right|_{u_\infty} d > 0,$$

and thus u_∞ is a strict local minimum of the plant. □

3.4 Sufficient Conditions for Convergence

A limited number of results are available regarding the convergence of modifier-adaptation schemes. A necessary condition based on a linearization of the algorithm around a fixed point is presented in Marchetti et al. (2009). A general framework that allows enforcing feasibility and optimality (and thus also convergence) of a broad class of RTO schemes is presented in Bunin et al. (2013b,c). Next, we propose to use a well-known result from fixed-point theory to state a fairly general sufficient condition for the convergence of modifier-adaptation schemes.

Proposition 4. (Convergence of modifier adaptation). Consider the modifier-adaptation scheme (9) with any order $\Omega \in \{(0,0),(1,1),(1,2),(2,2),(2,1),(2,0)\}$. Let Assumptions 1–3 hold and the filter gain $\alpha \in (0,1]$ be used.

If the map $u^\star : u \mapsto u^\star(u,\Lambda(u))$ is nonexpansive in the sense of Definition 1 and has at least one fixed point on $[\underline{u},\overline{u}]$, then the sequence $(u_k)_{k\in\mathbb{N}}$ of RTO iterates defined by (9) converges to a fixed point, i.e.

$$\lim_{k\to\infty} \|u^\star(u_k,\Lambda(u_k)) - u_k\| = 0.$$

Proof. The main idea of the proof is to show that the stated assumptions allow using Theorem 1. Recall that the modifier-adaptation scheme with input filtering given by (9) can be understood as an averaged iteration of the operator $u^\star : u \mapsto u^\star(u,\Lambda(u))$.

Next, we know from (10) that, for all $k \in \mathbb{N}$, $\mathcal{U}(u_k) \subset [\underline{u},\overline{u}]$. Assumptions 2 and 3 ensure that, for all $k \in \mathbb{N}$, $u^\star(u_k,\Lambda_k) \in [\underline{u},\overline{u}]$. Furthermore, the modifier-adaptation scheme (5) computes at each step a convex combination of u_k and $u^\star(u_k,\Lambda_k)$. Hence, we have $(u_k)_{k\in\mathbb{N}} \in [\underline{u},\overline{u}]$. It follows that Theorem 1 applies, and the sequence $(u)_{k\in\mathbb{N}}$ converges to a fixed point. □

Remark 2. (Reasons for filtering). This result holds for modifier-adaptation schemes of zeroth-, first-, second- and mixed orders. Since Theorem 1 indicates that the averaged (or filtered) iteration of an operator increases its domain of attraction, filtering can be understood as a way to increase the domain of attraction of modifier-adaptation schemes. This comes in addition to dealing with noisy measurements and the fact that large correction steps based on local information should be avoided.

Note that the value of Proposition 4 is mainly conceptual since it can be difficult to verify its assumptions and conditions. Subsequently, we discuss a simple unconstrained case and show how the main assumptions of Proposition 4 can be verified explicitly.

Illustrative Example: Consider uncertain quadratic programs. Assume that the plant is $\min_u \frac{1}{2}u^T H_p u + F_p u + c_p$. And the model is $\min_u \frac{1}{2}u^T H u + F u + c$. This means we consider unconstrained quadratic programs. Assume furthermore that the Hessians H_p, H are positive definite symmetric matrices.

Firstly, note that $u = -H_p^{-1}F_p^T$ is a fixed point of the first- and second-order modifier-adaptation schemes. Secondly, we show how nonexpansiveness of the *argmin*-operator can be verified. In the case of first-order modifiers, the modified cost function $\phi_m(u)$ reads

$$\phi_m(u) = \frac{1}{2}u^T Hu + Fu + c + \big(u_k^T(H_p - H) + F_p - F\big)(u - u_k).$$

It follows that

$$\frac{\partial \phi_m}{\partial u} = u^T H + F + \Big(u_k^T(H_p - H) + F_p - F\Big),$$

which leads to

$$u^\star(u_k) = H^{-1}\Big((H - H_p)\,u_k - F_p^T\Big).$$

Since

$$\|u^\star(w) - u^\star(v)\| \leq \|I - H^{-1}H_p\|\|w - v\|,$$

$\|I - H^{-1}H_p\| \leq 1$ is sufficient to ensure convergence of the first-order modifier-adaptation scheme (5) with input filtering.

Now consider the application of a second-order modifier-adaptation scheme. The modified cost function is

$$\phi_m(u) = \frac{1}{2}u^T Hu + Fu + c + \frac{1}{2}(u - u_k)^T(H_p - H)(u - u_k) + \big(u_k^T(H_p - H) + F_p - F\big)(u - u_k).$$

It is easy to show that

$$\frac{\partial \phi_m}{\partial u} = u^T H + F + (u - u_k)^T(H_p - H) + \big(u_k^T(H_p - H) + F_p - F\big) = u^T H_p + F_p$$

which yields

$$u^\star(u_k) = -H_p^{-1}F_p^T.$$

Since this is a constant, we immediately see that $\|u^\star(w) - u^\star(v)\| = 0$. Hence, the second-order modifier-adaptation scheme converges with input filtering. Furthermore, without input filtering ($\alpha = 1$), it converges in one step for any choice of positive definite symmetric matrices H_p and H.

3.5 Hessian Approximation

So far, we have implicitly assumed that the plant gradients and the Hessians are known. In practice, however, it is difficult to obtain the plant gradients from measurements (Bunin et al., 2013a). For the purpose of this paper, we will assume that the gradients of the plant cost and constraints are estimated via finite differences. For second-order adaptation, we consider Hessian approximation formulas that are well-known in numerical optimization (Nocedal and Wright, 2000).

Consider the two successive RTO iterations $k-1$ and k. Denote the differences between two successive RTO inputs and the corresponding gradients of the general function $l \in \{\phi, g_j\}$ as

$$s_k := u_k - u_{k-1}$$
$$t_{l,k} := \lambda_{l,k}^1 - \lambda_{l,k-1}^1.$$

For each of the functions l, we use the SR1 update formula

$$B_{l,k+1} = B_{l,k} + \frac{(t_{l,k} - B_{l,k}s_k)(t_{l,k} - B_{l,k}s_k)^T}{(t_{l,k} - B_{l,k}s_k)^T s_k} \qquad (13)$$

to approximate the corresponding Hessian matrix. Note that the definition of $t_{l,k}$ implies that we approximate the Hessian of the *difference* between the plant and model cost and constraints. In other words, we set $\lambda_{l,k}^2 := B_{l,k},\ l \in \{\phi, g_j\}$ in the modifier-adaptation scheme represented by (5) and (9).

The SR1 formula is well known in the context of quasi-Newton methods, where it is used to compute Hessian approximations to the cost functions (Nocedal and Wright, 2000, Chap. 6). In order to avoid singularities in (13), a skipping rule is used. This means that the update (13) is only applied if

$$\big\|(t_{l,k} - B_{l,k}s_k)^T s_k\big\| \geq \rho\|s_k\|\,\|t_{l,k} - B_{l,k}s_k\|\,, \qquad (14)$$

where $\rho \in (0, 1]$ is a small positive number.

Note that one could as well consider other Hessian approximation formulas such as the BFGS update. Here, we use the SR1 formula since the convergence of the Hessian estimates $B_{l,k}$ to the true Hessian can be guaranteed under certain conditions (Nocedal and Wright, 2000, Chap. 6).

3.6 Link to SQP and Trust-Region Methods

The results of Propositions 2-4 also hold for the model $\phi(u) = 0, g_j(u) = 0,\ j = 1, \ldots, n_g$, that is, the case of no model. Hence, it is fair to ask whether second-order corrections allow implementing model-free RTO schemes. If one considers the case of no model and mixed second- and first-order adaptation with $\Omega = (2, 1)$, each RTO iteration then amounts to solving a QP based on the Hessian approximation (13) and subject to affine approximations of the plant constraints. In other words, at the $(k+1)^{st}$ iteration, one would solve the following QP:

$$\min_u \ \frac{1}{2}(u - u_k)^T B_{\phi,k}\,(u - u_k) + t_{\phi,k}^T\,(u - u_k) \qquad (15a)$$

subject to

$$t_{g_j,k}^T(u - u_k) + g_j(u_k) \leq 0, \quad j = 1, \ldots, n_g \qquad (15b)$$
$$u \in [\underline{u}, \overline{u}]\ . \qquad (15c)$$

Obviously, such an approach leads to an SQP-like RTO scheme. Since the approximations of the cost and constraints are of local nature, a trust-region constraint is added,

$$\|u - u_k\| \leq \Delta_k, \qquad (15d)$$

where the radius Δ_k is adjusted according to the achieved progress at each iteration. [4]

[4] Different procedures exist for updating the trust-region radius, cf. (Nocedal and Wright, 2000; Conn et al., 2000).

Algorithm 1. SQP-like RTO scheme.

DATA: $u_0, B_0, t_{\phi,0}, t_{g_j,0}, \Delta_0, \varepsilon, \eta, \hat{\Delta}$

0. *INITIALIZE* $u_k = u_0, \Delta_k = \Delta_0, k = 0$;
1. *SOLVE* the QP (15);
2. *APPLY* $u_{k+1} = u^\star$ to plant and *EVALUATE* the resulting plant gradients $t_{l,k},\ l \in \{\phi, g_j\}$;
3. *COMPUTE* the gradient and Hessian updates (13);
4. *UPDATE* the trust-region radius
$$\Delta_{k+1} = f(\Delta_k, \hat{\Delta}, \phi_p(u_k), \phi_p(u_{k+1}));$$
5. *IF* $\|\phi_p(u_{k+1}) - \phi_p(u_k)\| \leq \eta$
THEN APPLY $u_{k+2} = u_k$, $k \to k+2$ *GOTO* 1.
ELSE $k \to k+1$ *GOTO* 1.

An SQP-like RTO scheme is described in Algorithm 1. The algorithm is initialized with the guesses $u_0, B_0, t_{\phi,0}, t_{g_j,0}, \Delta_0$ for the inputs, the Hessian, the gradients and the trust-region radius, respectively. Additionally, a solution tolerance $\varepsilon > 0$, an improvement requirement $\eta > 0$, and a maximum trust-region radius $\hat{\Delta} > 0$ have to be provided.

Note that the Hessians and gradients are updated even along non-decreasing directions, which improves the quality of the Hessian approximation (Nocedal and Wright, 2000, Chap. 6). Furthermore, there is no need for Algorithm 1 to filter the inputs, since the trust-region constraint already accounts for the local nature of the available plant data.

At this point, it is justified to mention several issues and open questions:

(1) Does the consideration of second-order adaptation bring any benefit in terms of convergence?
(2) Provided that sufficiently accurate gradient information is available, does second-order adaptation bring any advantage over the SQP-like RTO scheme given in Algorithm 1?
(3) In a noise-free setting, the SR1-based Hessian updates converge to the true Hessian (Nocedal and Wright, 2000). However, one might wonder how SQP-like RTO performs in the presence of measurement noise. And more generally, how do the three schemes compare in the presence of noise?

4. SIMULATED EXAMPLE

Subsequently, we investigate the first two questions via a simulated example. An answer to the third question is beyond the scope of this paper and subject of future work.

We consider the Williams-Otto reactor (Williams and Otto, 1960) that is often used as a test problem for real-time optimization techniques (Marchetti et al., 2010). The *plant* (or simulated reality) is a continuous stirred-tank reactor with the following reactions:

$$A + B \rightarrow C, \qquad \bar{k}_1 = \bar{\eta}_1 e^{\frac{-\bar{\nu}_1}{T_R}} \tag{16a}$$

$$B + C \rightarrow P + E, \qquad \bar{k}_2 = \bar{\eta}_2 e^{\frac{-\bar{\nu}_2}{T_R}} \tag{16b}$$

$$C + P \rightarrow G, \qquad \bar{k}_3 = \bar{\eta}_3 e^{\frac{-\bar{\nu}_3}{T_R}} \tag{16c}$$

The species P and E are the desired products, while C is an intermediate species. The reactor mass holdup is $W = 2105$ kg. The two inputs, namely the reactor temperature T_R in K and the inlet mass flowrate F_B in kg/s, are constrained to $[348, 368] \times [3.5, 5]$.

To simulate plant-model mismatch, the *model* considers a simplified reaction scheme that does not involve the intermediate species C:

$$A + 2B \rightarrow P + E, \quad k_1 = \eta_1 e^{\frac{-\nu_1}{T_R}} \tag{17a}$$

$$A + B + P \rightarrow G, \qquad k_2 = \eta_2 e^{\frac{-\nu_2}{T_R}}. \tag{17b}$$

Since the reaction schemes for the plant and the model are different, the kinetic parameters $\bar{\nu}_i, \bar{\eta}_i$, $i \in \{1, 2, 3\}$ and ν_j, η_j, $j \in \{1, 2\}$ also differ. Here, we consider the plant values given in Marchetti (2009), whereas the model values are $\eta_1 = 1.3 \cdot 10^8$, $\eta_2 = 1.1 \cdot 10^{13}$, $\nu_1 = 8.3 \cdot 10^3$, $\nu_2 = 1.28 \cdot 10^4$.

Table 1. Results 1^{st}-order adaptation

Gain α	1^{st} Order $\bar{i}_{conv}$	$\sigma(i_{conv})$	$\bar{\delta}$
0.25	55.37	5.74	$4.13 \cdot 10^{-3}$
0.5	27.30	3.22	$4.04 \cdot 10^{-3}$
0.75	17.47	1.74	$4.01 \cdot 10^{-3}$
1.0	13.43	0.89	$4.06 \cdot 10^{-3}$

Table 2. Results 2^{nd}-order adaptation

Gain α	2^{nd} Order $\bar{i}_{conv}$	$\sigma(i_{conv})$	$\bar{\delta}$
0.25	41.37	10.63	$3.65 \cdot 10^{-2}$
0.5	20.86	1.97	$4.07 \cdot 10^{-3}$
0.75	13.53	1.5	$4.05 \cdot 10^{-3}$
1.0	9.07	1.26	$4.06 \cdot 10^{-3}$

Table 3. Results SQP-like RTO

Gain α	Algorithm 1 $\bar{i}_{conv}$	$\sigma(i_{conv})$	$\bar{\delta}$
1.0	9.09	1.8	0.339

The plant profit to be maximized is

$$\phi_p(u) = (c_1 \bar{X}_P + c_2 \bar{X}_E)(F_A + F_B) - c_3 F_A - c_4 F_B,$$

where $\bar{X}_P$ and $\bar{X}_E$ are the mass fractions of species P and E in the plant, and F_A is the inlet mass flowrate of A in kg/s. The constants are $c_1 = 1143.38, c_2 = 25.92, c_3 = 76.23, c_4 = 114.34$.

We will apply three RTO schemes to the Williams-Otto reactor and compare their performance:

i) First-order modifier adaptation with finite-difference approximation of the gradients;
ii) Second-order modifier adaptation based on finite-difference approximation of the gradients and Hessian approximation according to (13);
iii) The SQP-like RTO scheme of Algorithm 1 based on finite-difference approximation of the gradients, Hessian approximation according to (13) and trust region update according to (Nocedal and Wright, 2000, Chap. 6).

For all three cases, a set of 100 randomly chosen initial guesses $u_0 \in [348, 368] \times [3.5, 5]$ and the filter gains $\alpha \in \{0.25, 0.5, 0.75, 1.0\}$ are considered. The SQP-like scheme is simulated without any input filtering since the trust-region constraint accounts for the local nature of the available plant data. The schemes are stopped when the difference between two successive RTO iterates is less than 10^{-4}. The Hessian approximations are initialized with $B_0 = -diag(1, 10)$.

Tables 1 and 2 summarize the result for first- and second-order modifier adaptation: $\bar{i}_{conv}$ is the mean of the number of iterations and $\bar{\delta}$ is the mean of the difference $\delta = \|u(i_{conv}) - u^\star\|$, where $u^\star = (362.85, 4.79)^T$ is the true plant optimum.

For this example, the second-order adaptation leads to slightly faster convergence. Furthermore, except for the case of $\alpha = 0.25$, the achieved accuracies are comparable.

Table 3 lists the result for the SQP-like scheme. We see that convergence is typically faster than with nearly all modifier-adaptation schemes. The price to pay for fast convergence is that the achieved average accuracy, as measured by $\bar{\delta}$, is not as good. The reason for this is that, in several of the 100 cases, the fixed initial Hessian guess B_0 did not allow the computation of a direction with sufficient cost decrease.

5. CONCLUSIONS

This paper has generalized first-order modifier adaptation to include second-order corrections to the cost and constraint functions. A sufficient condition for convergence of modifier-adaptation schemes of various orders has been presented. The consideration of Hessian terms allows checking whether the plant has indeed reached (local) optimality. Provided that sufficiently accurate gradient information is available, the use of second-order correction terms can speed up convergence. However, further research is required to assess the applicability of second-order updates in the presence of measurement noise.

This study conveys an additional, more abstract message, namely the interplay between model quality and measurements quality for the purpose of reaching the plant optimum. The weight between the two sources of information has to represent their relative quality. If sufficiently accurate gradient information is available, one can even discard the model completely and rely solely on local QP approximations that are obtained from measurements. Hence, in the presence of reliable measurements, one can use approximate first- and second-oder data in an SQP-like model-free RTO scheme. It turns out that, in the case of modifier-adaptation schemes, the adaptation order may well be a useful tuning knob that lets the designer weight the relative importance to give to the model and to the measurements.

REFERENCES

Bauschke, H.H. and Combettes, P.L. (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer.

Bazaraa, M.S., Sherali, H.D., and Shetty, C.M. (2006). *Nonlinear Programming: Theory and Algorithms*. John Wiley and Sons, New Jersey, 3rd edition.

Brdyś, M. and Tatjewski, P. (2005). *Iterative Algorithms for Multilayer Optimizing Control*. Imperial College Press, London UK.

Bunin, G. (2014). On the equivalence between the modifier-adaptation and trust-region frameworks. *Comput. Chem. Engng (submitted)*.

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.

Bunin, G., François, G., and Bonvin, D. (2013c). Sufficient conditions for feasibility and optimality of real-time optimization schemes - ii. Implementation issues. ArXiv:1308.2625.

Chachuat, B., Marchetti, A., and Bonvin, D. (2008). Process optimization via constraints adaptation. *J. Process Control*, 18(3), 244–257.

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

Conn, A., Gould, N., and Toint, P. (2000). *Trust Region Methods*, volume 1. MOS-SIAM Series on Optimization.

Dugundji, J. and Granas, A. (1982). *Fixed Point Theory*, volume 1. PWN-Polish Scientific Publishers.

Forbes, J. and Marlin, T. (1994). Model accuracy for economic optimizing controllers: The bias update case. *I&EC Research*, 33(8), 1919–1929.

Forbes, J. and Marlin, T. (1996). Design cost: A systematic approach to technology selection for model-based real-time optimization systems. *Comput. Chem. Engng*, 20(6), 717–734.

François, G. and Bonvin, D. (2013a). Measurement-based real-time optimization of chemical processes. In S. Pushpavanam (ed.), *Control and Optimisation of Process Systems*, volume 43, 1–50. Academic Press, Oxford.

François, G. and Bonvin, D. (2013b). Use of convex model approximations for real-time optimization via modifier adaptation. *I&EC Research*, 52(33), 11614–11625.

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

Ishikawa, S. (1974). Fixed points by a new iteration method. *Proc. of the American Mathematical Society*, 44(1), 147–150.

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

Johnson, G.G. (1972). Fixed points by mean value iterations. *Proc. of the American Mathematical Society*, 34(1), 193–194.

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.

Nocedal, J. and Wright, S. (2000). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, 2nd edition.

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

Skogestad, S. (2000). Plantwide control: The search for the self-optimizing control structure. *J. Process Control*, 10(5), 487–507.

Srinivasan, B. and Bonvin, D. (2007). Real-time optimization of batch processes by tracking the necessary conditions of optimality. *I&EC Research*, 46(2), 492–504.

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.