

# An Analysis of the Directional-Modifier Adaptation Algorithm Based on Optimal Experimental Design

## **Authors:**

Sébastien Gros

*Date Submitted:* 2018-07-31

*Keywords:* optimality loss function, experimental design, directional-modifier adaptation, modifier approach

## **Abstract:**

The modifier approach has been extensively explored and offers a theoretically-sound and practically-useful method to deploy real-time optimization. The recent directional-modifier adaptation algorithm offers a heuristic to tackle the modifier approach. The directional-modifier adaptation algorithm, supported by strong theoretical properties and the ease of deployment in practice, proposes a meaningful compromise between process optimality and quickly improving the quality of the estimation of the gradient of the process cost function. This paper proposes a novel view of the directional-modifier adaptation algorithm, as an approximation of the optimal trade-off between the underlying experimental design problem and the process optimization problem. It moreover suggests a minor modification in the tuning of the algorithm, so as to make it a more genuine approximation.

*Record Type:* Published Article

*Submitted To:* LAPSE (Living Archive for Process Systems Engineering)

*Citation (overall record, always the latest version):*

LAPSE:2018.0207

*Citation (this specific file, latest version):*

LAPSE:2018.0207-1

*Citation (this specific file, this version):*

LAPSE:2018.0207-1v1

*DOI of Published Version:* <https://doi.org/10.3390/pr5010001>

*License:* Creative Commons Attribution 4.0 International (CC BY 4.0)

Article

# An Analysis of the Directional-Modifier Adaptation Algorithm Based on Optimal Experimental Design

Sébastien Gros

Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Göteborg, Sweden; grosse@chalmers.se; Tel.: +46-31-772-1555

Academic Editor: Dominique Bonvin

Received: 1 November 2016; Accepted: 15 December 2016; Published: 22 December 2016

**Abstract:** The modifier approach has been extensively explored and offers a theoretically-sound and practically-useful method to deploy real-time optimization. The recent directional-modifier adaptation algorithm offers a heuristic to tackle the modifier approach. The directional-modifier adaptation algorithm, supported by strong theoretical properties and the ease of deployment in practice, proposes a meaningful compromise between process optimality and quickly improving the quality of the estimation of the gradient of the process cost function. This paper proposes a novel view of the directional-modifier adaptation algorithm, as an approximation of the optimal trade-off between the underlying experimental design problem and the process optimization problem. It moreover suggests a minor modification in the tuning of the algorithm, so as to make it a more genuine approximation.

**Keywords:** modifier approach; directional-modifier adaptation; experimental design; optimality loss function

---

## 1. Introduction

Real-Time Optimization (RTO) aims at improving the performance and safety of industrial processes by means of continually-adjusting their inputs, i.e., the degrees of freedom defining their operating conditions, in response to disturbances and process variations. RTO makes use of both model-based and model-free approaches. The model-free approaches have the clear advantage of being less labor intensive, as a model of the process is not needed, but the increasing number of inputs that can be adjusted when running the process has made them decreasingly attractive.

Model-based techniques have received an increasing interest as the capability of running a large amount of computations online has become standard. Arguably, the most natural approach to model-based RTO is the two-step approach, where model parameter estimation and model-based optimization are alternated so as to refine the process model and adapt the operational parameters accordingly [1,2]. Unfortunately, the two-step approach requires the process model to satisfy very strict criteria in order for the scheme to reach optimality [3,4]. This issue is especially striking in the case of structural mismatch between the model and the process and can make the two-step scheme ineffective or even counterproductive [5–7].

The idea of not only adapting the model parameters, but also the gradient of the cost function can be traced back to [8] and allows for guaranteeing that the resulting scheme reaches optimality upon convergence [7,9,10]. Unlike the two-step approach, adapting the gradient of the cost function allows one to tackle structural model-plant mismatches efficiently, which cannot be efficiently addressed via the adaptation of the model parameters alone. The original idea has been further improved; see, e.g., [7,9–13]. These contributions have converged to the modern Modifier Adaptation (MA) approach, which has been successfully deployed on several industrial processes; see [14–17]. The MA approach has been recently further developed along a number of interesting directions; see [16,18–21].

In a run-to-run scenario where estimations of the uncertain parameters are carried out after every run, the input for any run does not only maximize the process performance for the coming run, but also influences the performance of the subsequent runs through the estimation of the process parameters. This observation is generally valid when parameter estimation is performed between runs and pertains to the MA approach. Taking this influence into account leads one to possibly depart from applying to the process an input that is optimal according to the best available estimation of the parameters at the time and adopt an input that strikes a compromise between process optimality and gathering relevant information for the next parameter estimation. In that sense, the MA approach can be construed as a mix of an optimization problem and an experimental design problem. The problem of tailoring experimental design specifically for optimization in a computationally-tractable way has been recently studied in [22], where the problem of designing inputs for a process so as to gather relevant information for achieving process optimality is tackled via an approximate optimality loss function.

The recently-proposed Directional-Modifier Adaptation (DMA) algorithm [23,24] and its earlier variant the dual-modifier adaptation approach [25] offer a practical way for the MA approach to deal with the compromise between process optimality and gaining information. Indeed, at each process run, the DMA algorithm delivers an input that seeks a compromise between maximizing the process performance and promoting the quality of the estimation of the process gradients. The DMA approach handles this compromise by adopting inputs that depart from the nominal ones in directions corresponding to the largest covariance in the estimated gradients of the process Lagrange function. The DMA algorithm is easy to deploy and has strong theoretical properties, e.g., it converges rapidly and with guarantees to the true process optimum. The directional-modifier adaptation algorithm additionally makes use of iterative schemes to update the modifiers used in the cost model, so as to reduce the computational burden of performing classical gradient estimations.

In this paper, we propose to construct the DMA algorithm from a different angle, based on a modification of the optimality loss function [22]. This construction delivers new theoretical insights into the DMA algorithm and suggests minor modifications that make the DMA algorithm a more genuine approximation of the optimal trade-off between process optimality and excitation. For the sake of simplicity, we focus on the unconstrained case, though the developments can arguably be naturally extended to constrained problems.

The paper is structured as follows. Section 2 proposes some preliminaries on the selection of an optimality loss function for the considered experimental design problem and proposes a computationally-tractable approximation, following similar lines as [22]. Section 3 investigates the MA approach as a special case of the previous developments, proposes to tackle it within the proposed theoretical framework and shows that the resulting algorithm has the same structure as the DMA algorithm, but with some notable differences. Simple examples are presented throughout the text to illustrate and support the concepts presented.

## 2. Optimal Experimental Design

In this paper, we consider the problem of optimizing a process in a run-to-run fashion. The process is described via the cost function  $\phi(u, p)$ , where  $u$  gathers the set of inputs, or degrees of freedom, available to steer the process, and  $p$  gathers the parameters available to adjust the cost function using the measurements gathered on the plant. Function  $\phi$  is assumed to be everywhere defined and smooth. This assumption is arguably not required, but will make the subsequent analysis less involved. The  $N$ -run optimization problem can then be formulated as:

$$\min_{u_0, \dots, u_{N-1}} \frac{1}{N} \sum_{k=0}^{N-1} \phi(u_k, p), \quad (1)$$

where  $u_k$  is the vector of decision variables applied at run  $k$ . Here, we seek the minimization of the average process performance over the  $N$  runs. The cost function  $\phi(u, p)$  associated with the process is not available in practice, such that at any run  $k$ , the input  $u_k$  is typically chosen according to the best parameter estimation  $\hat{p}_k$  available at that time. It is important to observe here that the parametric cost function (1) encompasses parametric mismatch between the plant and the model, but also any structure adjusting the cost function according to the data, such as the MA approach; see Section 3.1. Ideally, one ought to seek solving the optimization model:

$$\min_{u_{0,\dots,N-1}} \frac{1}{N} \sum_{k=0}^{N-1} \mathbb{E}_{\hat{p}_k} [\phi(u_k, \hat{p}_k)], \quad (2)$$

where  $\mathbb{E}_{\hat{p}_k}$  stands for the expected value over  $\hat{p}_k$ . For the sake of simplicity, we will focus in this paper on the two-run problem, i.e., using  $N = 2$  in Problem (2). In the following, we will assume that there exists a vector of parameter  $p_{\text{real}}$  for which  $\phi(u_k, p_{\text{real}})$  captures effectively the cost function of the real process. This assumption is locally fulfilled, up to a constant term, by the MA approach.

When estimations of the parameters  $\hat{p}_k$  are conducted between the runs using the latest measurements gathered on the process, a difficulty in using (2) stems from the fact that it can yield an inadequate sequence of decisions  $u_{0,\dots,N-1}$ . We motivate this statement next, via a simple example.

### 2.1. Failure of Problem (2): An Example

Consider the optimization model  $\phi(u) = u^2 + p^2$  yielding the two-run problem:

$$\min_{u_{0,1}} \frac{1}{2} \sum_{k=0}^1 \mathbb{E}_{\hat{p}_k} \{u_k^2 + \hat{p}_k^2\} = \min_{u_{0,1}} \frac{1}{2} \sum_{k=0}^1 u_k^2 + \Sigma_k + \mu_k^2, \quad (3)$$

where  $\Sigma_k$  is the covariance of the estimation of parameter  $\hat{p}_k$  and  $\mu_k$  its expected value. If the distribution of the estimated parameter  $\hat{p}_1$  is independent of the input  $u_0$ , then Problem (3) takes the trivial solution  $u_{0,1} = 0$ , which yields the best performance on the real cost function  $\phi(u, p_{\text{real}})$ , regardless of the actual parameter value  $p_{\text{real}}$  or of its estimated value  $\hat{p}_0$  available for deciding the input  $u_0$ . However, since the estimated parameter  $\hat{p}_1$  is obtained from the run based on  $u_0$ , it is in fact not independent of the decision variables. Indeed, let us assume that the estimation of  $\hat{p}_1$  is provided between the two runs via the least-square fitting problems:

$$\hat{p}_1 = \arg \min_p \frac{1}{2} \|p - \hat{p}_0\|_{\Sigma_0^{-1}}^2 + \frac{1}{2} \|y(u_0, p) - y_0^{\text{meas}}\|_{\Sigma_{\text{meas}}^{-1}}^2, \quad (4)$$

where  $y_0^{\text{meas}} \in \mathbb{R}^m$  is the measurements taken on the process during or after the run based on  $u_0$ ,  $y(u, p)$  is the corresponding measurement model,  $\Sigma_{\text{meas}}$  is the covariance of the measurement noise and  $\Sigma_0$  the covariance associated with the parameter estimation  $\hat{p}_0$ . Consider then the measurement model:

$$y(u, p) = pu. \quad (5)$$

The solution to (4) is then explicitly given by:

$$\hat{p}_1 = \left( \Sigma_0^{-1} + \Sigma_{\text{meas}}^{-1} u_0^2 \right)^{-1} \left( \Sigma_0^{-1} \hat{p}_0 + \Sigma_{\text{meas}}^{-1} u_0 y_0^{\text{meas}} \right). \quad (6)$$

Assuming that  $\mu_{\hat{p}_0} = p_{\text{real}} = 0$  and  $\mathbb{E}\{y_0^{\text{meas}}\} = 0$ , we then observe that if the measurement noise is independent between the various runs, we have:

$$\Sigma_1 = \left( \Sigma_0^{-1} + \Sigma_{\text{meas}}^{-1} u_0^2 \right)^{-1}. \quad (7)$$

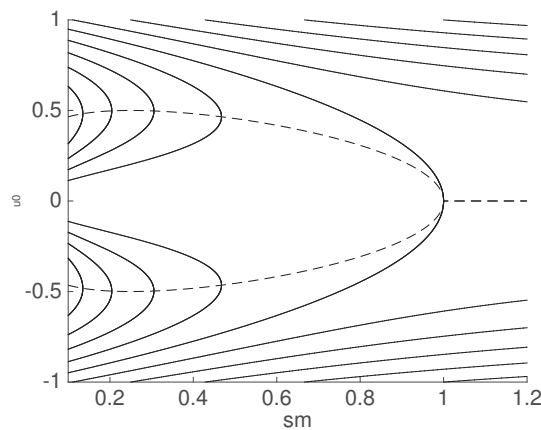
After removing the constant terms, Problem (3) becomes:

$$\min_{u_0, u_1} \frac{1}{2} (u_0^2 + u_1^2) + \frac{1}{2} (\Sigma_0^{-1} + \Sigma_{\text{meas}}^{-1} u_0^2)^{-1}. \quad (8)$$

An interesting situation occurs for  $\Sigma_{\text{meas}} \leq \Sigma_0^2$ , i.e., when the covariance of the measurements is sufficiently low; see Figure 1. The solution to (8) then reads as:

$$u_0 = \pm \left( \Sigma_{\text{meas}}^{\frac{1}{2}} - \frac{\Sigma_{\text{meas}}}{\Sigma_0} \right)^{\frac{1}{2}}, \quad u_1 = 0, \quad (9)$$

while the sequence  $u_0 = u_1 = 0$  should clearly be used in order to minimize the cost of the real two-run process, even in the sense of the expected value. This trivial example illustrates a fundamental limitation of Problem (2) in successfully achieving the goal of minimizing the cost over a two- or  $N$ -run process.



**Figure 1.** Illustration for Problem (8). The level curves report the cost of (8) as a function of  $u_0$  and  $\Sigma_{\text{meas}}$ , with  $\Sigma_0 = 1$ . The dashed lines report the optimal input  $u_0$  for various values of  $\Sigma_{\text{meas}}$ . For  $\Sigma_{\text{meas}}$  low enough, the problem has two non-zero solutions.

## 2.2. Modified Optimality Loss Function

A sensible approach inspired from the work presented in [22] consists of selecting the input  $u_0$  according to:

$$u_0 = \arg \min_u \mathbb{E}_e [\phi(u, p_{\text{real}}) + \phi(u^*(\hat{p}_1), p_{\text{real}})] = \arg \min_u \mathbb{E}_e [ \phi(u, p_{\text{real}}) + \underbrace{\phi(u^*(\hat{p}_1), p_{\text{real}}) - \phi(u^*(p_{\text{real}}), p_{\text{real}})}_{\Delta_0} ], \quad (10)$$

where  $\Delta_0$  is labeled the optimality loss function and  $e$  gathers the noise on the estimation of the process parameters and the measurement noise, i.e.:

$$\hat{p}_0 = p_{\text{real}} + e_0, \quad y_{\text{meas}} = y(u_0, p_{\text{real}}) + e_1. \quad (11)$$

Problem (10) seeks a compromise between the expected process performance at the coming run via the first term in (10) and the expected process performance at the subsequent run via the second term. The performance of the second term depends on the input selected in the first run via the parameter

estimation performed between the two runs. We assume hereafter that  $e$  follows a normal, centered distribution, and we use for the estimated parameter  $\hat{p}_1$  the least-square fitting problem:

$$\hat{p}_1 = \arg \min_p \frac{1}{2} \|p - (p_{\text{real}} + e_0)\|_{\Sigma_0^{-1}}^2 + \frac{1}{2} \|y(u_0, p_{\text{real}}) + e_1 - y(u_0, p)\|_{\Sigma_1^{-1}}^2, \quad (12)$$

where  $\hat{p}_1$  is the parameter estimation following the first run. The optimality loss function  $\Delta_0$  proposed in [22] was designed for the specific purpose of performing experimental design dedicated to capturing the process parameters most relevant for process optimization. However, it was not designed to be used within the two-run problem considered here. In this paper, we propose to use a slightly modified version of (10), so as to avoid a potential difficulty it poses. For the sake of brevity and in order to skip elaborate technical details, let us illustrate this difficulty via the following simple example. Consider the cost function and measurement model:

$$\phi(u, p) = \frac{1}{2} (u - p)^2, \quad y(u, p) = up. \quad (13)$$

The least-square problem (12) reads as:

$$\hat{p}_1 = \arg \min_p \frac{1}{2} \|p - (p_{\text{real}} + e_0)\|_{\Sigma_0^{-1}}^2 + \frac{1}{2} \|u_0 p_{\text{real}} + e_1 - u_0 p\|_{\Sigma_1^{-1}}^2, \quad (14)$$

which takes the explicit form:

$$\hat{p}_1 = p_{\text{real}} + \frac{e_0 \Sigma_1 + e_1 \Sigma_0 u_0}{\Sigma_0 u_0^2 + \Sigma_1}. \quad (15)$$

The optimality loss function  $\Delta_0$  then reads as:

$$\Delta_0(u_0) = \phi(u^*(\hat{p}_1), p_{\text{real}}) - \phi(u^*(p_{\text{real}}), p_{\text{real}}) = \frac{1}{2} \left( \frac{e_0 \Sigma_1 + e_1 \Sigma_0 u_0}{\Sigma_0 u_0^2 + \Sigma_1} \right)^2, \quad (16)$$

and has the expected value:

$$\mathbb{E}_e [\Delta_0] = \frac{1}{2} \frac{\Sigma_0 \Sigma_1}{\Sigma_0 u_0^2 + \Sigma_1}. \quad (17)$$

It is worth observing that a similar optimality loss function has also been used in [25] in order to quantify the loss of optimality resulting from uncertain parameters. Problem (10) can then be equivalently written as:

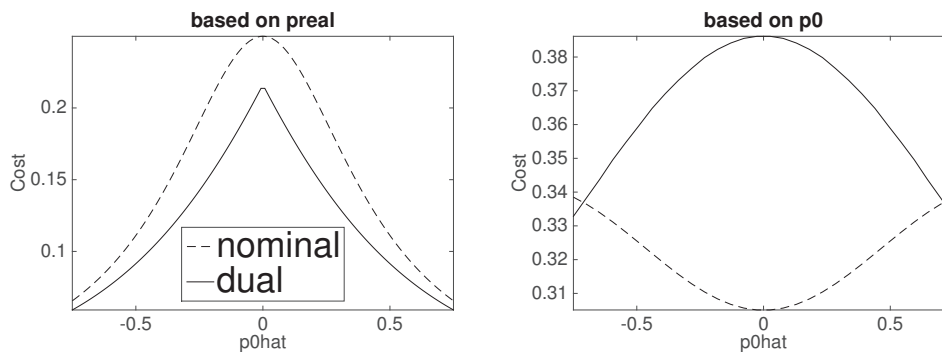
$$u_0 = \arg \min_u \phi(u, p_{\text{real}}) + \mathbb{E}_e [\Delta_0]. \quad (18)$$

However, since in practice,  $p_{\text{real}}$  is not available to solve Problem (18), a surrogate problem must be solved, using  $p_{\text{real}} \approx \hat{p}_0$ . It reads as:

$$u_0 = \arg \min_u \phi(u, \hat{p}_0) + \mathbb{E}_e [\Delta_0] = \arg \min_u \phi(u, p_{\text{real}} + e_0) + \mathbb{E}_e [\Delta_0]. \quad (19)$$

An issue occurs here, which is illustrated in Figure 2. Because the expected value of the optimality loss function computed in a stand-alone fashion in (17) misses the correlation between the control input  $u_0$  and the initial estimation error  $e_0$  that arises via the optimization problem (19), using (19) as a surrogate for (18) can be counterproductive in the sense that the performance of Problem (18) is worse than the one of the nominal problem.

In this paper, we address this issue by taking an approach to the optimality loss function that departs slightly from (16).



**Figure 2.** Comparison of the performance resulting from using the nominal input  $u_0 = u^*(\hat{p}_0)$  and the one resulting from (18) or (19) on the proposed example. The displayed cost is calculated according to (10) and reads as  $\mathbb{E}_e \left[ \frac{1}{2} u_0^2 + \frac{1}{2} u_1^*(\hat{p}_1)^2 \right]$ . The left graph displays the cost resulting from using (18), which delivers a better expected performance than using the nominal input. The right graph displays the cost resulting from using (19), where  $p_{\text{real}} \approx \hat{p}_0$  is used. In this example, this approximation is detrimental to the performance of Problem (10), resulting in a worse performance than the nominal one.

### 2.3. Problem Formulation

For a given initial estimation  $\hat{p}_0$ , initial estimation error  $e_0$  and measurement error  $e_1$  and using  $p_{\text{real}} = \hat{p}_0 - e_0$ , the estimation problem solved after the first run can be formulated as:

$$\hat{p}_1(u_0, \Sigma, \hat{p}_0, e) = \arg \min_p \frac{1}{2} \|p - \hat{p}_0\|_{\Sigma_0^{-1}}^2 + \frac{1}{2} \|y(u_0, p) - (y(u_0, \hat{p}_0 - e_0) + e_1)\|_{\Sigma_1^{-1}}^2, \quad (20)$$

where we use the notation:

$$e = \begin{bmatrix} e_0 \\ e_1 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_0 & 0 \\ 0 & \Sigma_1 \end{bmatrix}, \quad (21)$$

and consider  $e_0, e_1$  to be uncorrelated. Defining:

$$\hat{u}_1^*(u_0, \Sigma, \hat{p}_0, e) = u^*(\hat{p}_1(u_0, \Sigma, \hat{p}_0, e)), \quad (22)$$

the modified optimality loss function can be formulated as:

$$\Delta(u_0, \Sigma, \hat{p}_0, e) = \phi(\hat{u}_1^*(u_0, \Sigma, \hat{p}_0, e), \hat{p}_0 - e_0) - \phi(u^*(\hat{p}_0 - e_0), \hat{p}_0 - e_0). \quad (23)$$

This reformulation allows for construing the optimality loss function from the point of view of the experimenter, by considering  $\hat{p}_0$  as a fixed variable arising as a realization of the estimation of the unknown parameter  $p_{\text{real}}$  rather than a stochastic one. In (20) and (23), the actual parameter  $p_{\text{real}}$  is then, from the experimenter point of view, a stochastic variable, reflecting the uncertainty of the experimenter concerning the real parameter. The resulting two-run problem reads as:

$$u_0 = \arg \min_u \mathbb{E}_e [\phi(u, \hat{p}_0 - e_0) + \Delta(u, \Sigma, \hat{p}_0, e)]. \quad (24)$$

We observe here that the cost function proposed in (24) is different from the original one in (19). From the optimality principle, Problem (24) delivers an expected performance that is better or no worse than the expected performance yielded by applying the nominal input  $u_0 = u^*(\hat{p}_0)$ . A simple

example of the proposed optimality-loss approach is provided in Section 2.5. Unfortunately, solving Problem (24) is in general difficult. In the next section, we consider a second-order approximation instead, following a line also adopted in [22].

#### 2.4. Second-Order Approximation of the Modified Optimality Loss Function

The optimality loss function (23) is difficult to use in practice. A second-order approximation of (23) can be deployed as a tractable surrogate problem in (24). We develop this second-order approximation next. We observe that the following equality trivially holds:

$$\hat{p}_1(u_0, \Sigma, \hat{p}_0, 0) = \hat{p}_0. \quad (25)$$

The sensitivity of the parameter estimations  $\hat{p}_1$  to the errors  $e$  can be obtained via the implicit function theorem applied to the fitting problem (20); it reads as:

$$\left. \frac{\partial \hat{p}_1(u_0, \Sigma, \hat{p}_0, e)}{\partial e} \right|_{e=0} = F(u_0, \Sigma, \hat{p}_0)^{-1} M(u_0, \hat{p}_0), \quad (26)$$

where:

$$F(u_0, \Sigma, \hat{p}_0) = \Sigma_0^{-1} + y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} y_p(u_0, \hat{p}_0) \quad (27)$$

is the Fisher information matrix of (20), and:

$$\begin{aligned} M(u_0, \hat{p}_0) &= \begin{bmatrix} -y_p(u_0, \hat{p}_0) \Sigma_1^{-1} y_p(u_0, \hat{p}_0)^\top & y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} \\ \Sigma_0^{-1} - F(u_0, \Sigma, \hat{p}_0) & y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} \end{bmatrix} \\ &= \begin{bmatrix} \Sigma_0^{-1} - F(u_0, \Sigma, \hat{p}_0) & y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} \end{bmatrix}. \end{aligned} \quad (28)$$

We note that from optimality that  $\Delta \geq 0$  always holds and:

$$\Delta(u_0, \Sigma, \hat{p}_0, 0) = 0, \quad \left. \frac{\partial \Delta(u_0, \Sigma, \hat{p}_0, e)}{\partial e} \right|_{e=0} = 0, \quad (29)$$

which motivates a second-order approximation of  $\Delta$  at  $e = 0$ . The Taylor expansion of  $\Delta$  in  $e$  reads as:

$$\Delta(u_0, \Sigma, \hat{p}_0, e) = \frac{1}{2} e^\top \frac{\partial^2 \Delta(u_0, \Sigma, \hat{p}_0, 0)}{\partial e^2} e + r_3(u_0, \Sigma, \hat{p}_0, e). \quad (30)$$

We can then form the second-order approximation of the modified optimality loss function  $\Delta$ .

**Lemma 1.** *The following equality holds:*

$$\frac{\partial^2 \Delta(u_0, \Sigma, \hat{p}_0, 0)}{\partial e^2} = \left( \frac{\partial \hat{p}_1}{\partial e} + \frac{\partial e_0}{\partial e} \right)^\top \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* \left( \frac{\partial \hat{p}_1}{\partial e} + \frac{\partial e_0}{\partial e} \right), \quad (31)$$

where we note  $\phi_{xx}^* = \phi_{xx}(u^*(\hat{p}_0), \hat{p}_0)$ , and all partial derivatives are evaluated at  $e = 0$ .

**Proof.** We observe that:

$$\begin{aligned} \frac{\partial^2 \Delta(u_0, \Sigma, \hat{p}_0, e)}{\partial e^2} &= \frac{\partial}{\partial e} \left( \phi_u(\hat{u}_1^*, \hat{p}_0 - e_0) u_p^*(\hat{p}_1) \frac{\partial \hat{p}_1}{\partial e} - \phi_p(\hat{u}_1^*, \hat{p}_0 - e_0) \frac{\partial e_0}{\partial e} \right. \\ &\quad \left. + \phi_u(u^*(\hat{p}_0 - e_0), \hat{p}_0 - e_0) u_p^*(\hat{p}_0 - e_0) \frac{\partial e_0}{\partial e} + \phi_p(u^*(\hat{p}_0 - e_0), \hat{p}_0 - e_0) \frac{\partial e_0}{\partial e} \right), \end{aligned} \quad (32)$$



where for the sake of clarity, the arguments are omitted when unambiguous. Using the fact that  $\phi_u(u^*(\hat{p}_0), \hat{p}_0) = 0$ , it follows that:

$$\begin{aligned} \frac{\partial^2 \Delta(u_0, \Sigma, \hat{p}_0, 0)}{\partial e^2} &= \frac{\partial \hat{p}_1^\top}{\partial e} (u_p^*)^\top \phi_{uu}^* u_p^* \frac{\partial \hat{p}_1}{\partial e} - \frac{\partial e_0^\top}{\partial e} \phi_{pu}^* u_p^* \frac{\partial \hat{p}_1}{\partial e} - \frac{\partial \hat{p}_1^\top}{\partial e} (u_p^*)^\top \phi_{up}^* \frac{\partial e_0}{\partial e} + \frac{\partial e_0^\top}{\partial e} \phi_{pp}^* \frac{\partial e_0}{\partial e} \\ &\quad - \frac{\partial e_0^\top}{\partial e} (u_p^*)^\top \phi_{uu}^* u_p^* \frac{\partial e_0}{\partial e} - \frac{\partial e_0^\top}{\partial e} \phi_{pu}^* u_p^* \frac{\partial e_0}{\partial e} - \frac{\partial e_0^\top}{\partial e} (u_p^*)^\top \phi_{up}^* \frac{\partial e_0}{\partial e} - \frac{\partial e_0^\top}{\partial e} \phi_{pp}^* \frac{\partial e_0}{\partial e}, \end{aligned} \quad (33)$$

where all functions are evaluated at  $e = 0$ . We use then the equality  $u_p^* = -(\phi_{uu}^*)^{-1} \phi_{up}^*$  to get (31).  $\square$

In the following, it will be useful to write  $\Delta(u_0, \Sigma, \hat{p}_0, e)$  as:

$$\Delta(u_0, \Sigma, \hat{p}_0, e) = \frac{1}{2} \text{Tr} \left( \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* V \right) + r_3(u_0, \Sigma, \hat{p}_0, e), \quad (34)$$

where we note:

$$V(u_0, \Sigma, \hat{p}_0, e) = \left( \frac{\partial \hat{p}_1(u_0, \Sigma, \hat{p}_0, e)}{\partial e} \Big|_{e=0} + \frac{\partial e_0}{\partial e} \right) e e^\top \left( \frac{\partial \hat{p}_1(u_0, \Sigma, \hat{p}_0, e)}{\partial e} \Big|_{e=0} + \frac{\partial e_0}{\partial e} \right)^\top.$$

Using (28), we observe that:

$$\frac{\partial \hat{p}_1(u, \Sigma, \hat{p}_0, e)}{\partial e} \Big|_{e=0} + \frac{\partial e_0}{\partial e} = F(u_0, \Sigma, \hat{p}_0)^{-1} \begin{bmatrix} \Sigma_0^{-1} & y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} \end{bmatrix}, \quad (35)$$

such that:

$$\mathbb{E}_e [V(u_0, \Sigma, \hat{p}_0, e)] = F(u_0, \Sigma, \hat{p}_0)^{-1} \begin{bmatrix} \Sigma_0^{-1} & y_p(u_0, \hat{p}_0)^\top \Sigma_1^{-1} \end{bmatrix} \Sigma_1 (\star) = F(u_0, \Sigma, \hat{p}_0)^{-1}. \quad (36)$$

It follows that the expected value of the optimality loss function reads as:

$$\mathbb{E}_e [\Delta(u, \Sigma, \hat{p}_0, e)] = \frac{1}{2} \text{Tr} \left( \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* F(u_0, \Sigma, \hat{p}_0)^{-1} \right) + \mathbb{E}_e [r_3(u_0, \Sigma, \hat{p}_0, e)]. \quad (37)$$

It is useful to observe that even though a modified optimality loss function has been selected here, its approximation (37) is nonetheless very similar to the one proposed in [22]. Hence, the real difference lies in its interpretation as an approximation of the modified function (23) rather than (16). Here, it is useful to introduce the following lemma:

**Lemma 2.** *If the following conditions hold:*

1. *the noise  $e$  has a multivariate normal and centered distribution*
2. *for all  $p \in \mathbb{P}$ ,  $u^*(p)$  exists, is smooth, unique and satisfies the Second-Order Sufficient Condition (SOSC) condition of optimality.*
3. *the parameter estimation problem (20) has a unique solution  $\hat{p}_1(u_0, \Sigma, \hat{p}_0, e)$  satisfying SOSC for any  $e$  and is smooth and polynomially bounded in  $e$*
4. *functions  $u^*(p)$ ,  $\hat{p}_1(u_0, \Sigma, \hat{p}_0, e)$  and  $\phi_{up}$ ,  $\phi_{uu}$  are all bounded by polynomials on their respective domains.*

*Then, the inequality:*

$$|\mathbb{E}_e [r_3(u_0, \Sigma, \hat{p}_0, e)]| \leq c \|\Sigma\|^2 \quad (38)$$

*holds locally for some constant  $c > 0$ , where  $\|\cdot\|$  is the matrix two-norm.*

**Proof.** Because all functions are smooth and bounded by polynomials, the function  $\Delta$  is also smooth and bounded by polynomials. It follows that:

$$r_3(u_0, \Sigma, \hat{p}_0, e) = \Delta(u_0, \Sigma, \hat{p}_0, e) - \frac{1}{2} \text{Tr} \left( \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* V \right) \quad (39)$$

is also smooth and polynomially bounded. Additionally, the bound:

$$|r_3(u_0, \Sigma, \hat{p}_0, e)| \leq c \|e\|^3 \quad (40)$$

holds locally for some  $c > 0$  as a result of Taylor's theorem. Then, Inequality (38) follows directly from Lemma 3.  $\square$

Lemma 2 appears to be a special case of the delta method [26]. We can now approximate (24) as:

$$\min_{u_0} \mathbb{E}_e [\phi(u_0, \hat{p}_0 - e_0)] + \frac{1}{2} \text{Tr} \left( \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* F(u_0, \Sigma, \hat{p}_0)^{-1} \right), \quad (41)$$

using  $\phi_{uu}^* = \phi_{uu}(u^*(\hat{p}_0), \hat{p}_0)$  and  $\phi_{up}^* = \phi_{up}(u^*(\hat{p}_0), \hat{p}_0)$ .

---

**Algorithm 1:** 2-run nominal optimal experimental design.

---

**Input** : Current parameter estimation  $\hat{p}_0$ , covariance  $\Sigma$ .

- 1 Compute  $u^*(\hat{p}_0)$
- 2 Evaluate  $\phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^*$  and  $F$  at  $u^*(\hat{p}_0), \hat{p}_0$
- 3 Solve:

$$\min_{u_0} \mathbb{E}_e [\phi(u_0, \hat{p}_0 - e_0)] + \frac{1}{2} \text{Tr} \left( \phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* F(u_0, \Sigma, \hat{p}_0)^{-1} \right)$$

- 4 Apply  $u_0$  to the process, gather measurements, perform parameter estimation update
- return** updated  $\hat{p}_0$  and  $\Sigma$ , repeat
- 

For the sake of clarity, the deployment of Problem (41) in a run-to-run algorithm is detailed in Algorithm 1.

### 2.5. Illustrative Example: Observability Problem

We consider again the example (13), i.e.:

$$\phi(u, p) = \frac{1}{2} (u - p)^2, \quad y(u, p) = up. \quad (42)$$

where we consider  $\hat{p}_0 = p_{\text{real}} + e_0$  as known a priori with  $\mathbb{E}[e_0] = 0$ , and  $\hat{p}_1$  is provided by the estimation problem:

$$\hat{p}_1 = \arg \min_p \frac{1}{2} \|p - \hat{p}_0\|_{\Sigma_0^{-1}}^2 + \frac{1}{2} \underbrace{\|u_0(\hat{p}_0 - e_0) + e_1 - u_0 p\|_{\Sigma_1^{-1}}^2}_{y^{\text{meas}}}, \quad (43)$$

and takes the explicit solution:

$$\hat{p}_1 = \hat{p}_0 + \frac{e_1 \Sigma_0 u_0 - e_0 \Sigma_0 u_0^2}{\Sigma_0 u_0^2 + \Sigma_1}. \quad (44)$$

The optimality loss for the second run then reads as:

$$\Delta = \frac{1}{2} (u_1^* (\hat{p}_1) - (\hat{p}_0 - e_0))^2 - \underbrace{\phi^*}_{=0} = \frac{1}{2} \frac{(e_0 \Sigma_1 + e_1 \Sigma_0 u_0)^2}{(\Sigma_0 u_0^2 + \Sigma_1)^2}, \tag{45}$$

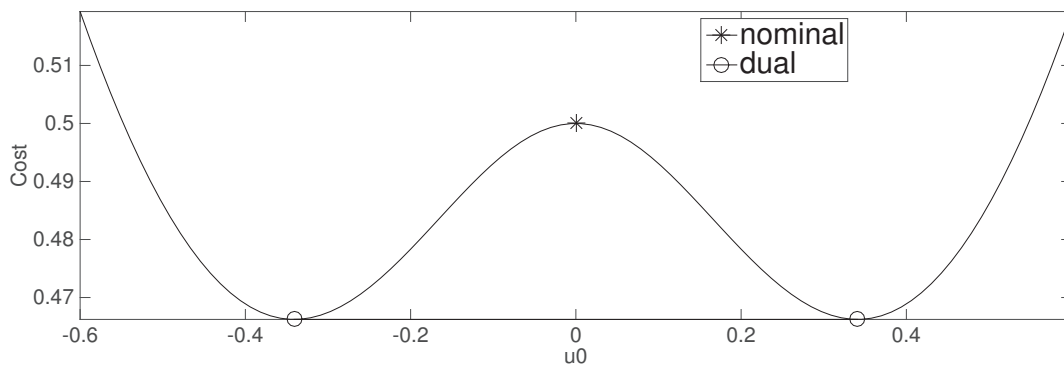
and its expected value takes the form:

$$\mathbb{E}_e [\Delta] = \frac{1}{2} \frac{\Sigma_0 \Sigma_1}{\Sigma_0 u_0^2 + \Sigma_1} = \frac{1}{2} F(u_0, \Sigma)^{-1}. \tag{46}$$

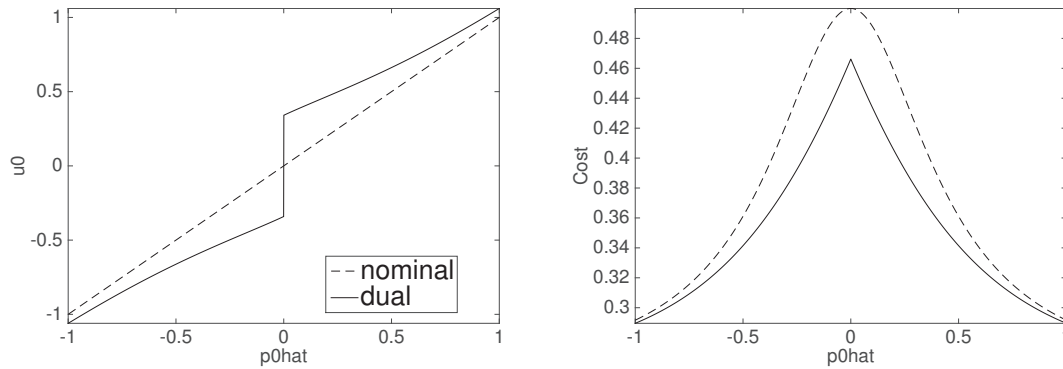
Ignoring the constant terms and since  $\mathbb{E}[e_0] = 0$ , the two-stage optimal experimental design then picks the input  $u_0$  according to:

$$u_0 = \arg \min_u \frac{1}{2} (u - \hat{p}_0)^2 + \frac{1}{2} F(u, \Sigma)^{-1}. \tag{47}$$

We observe that in this simple case, the proposed approximation (41) is identical to the original problem (24) and to Problem (19). This equivalence does not hold in general. The behavior of Problem (41) in this simple case is reported in Figures 3 and 4. In particular, we observe that the expected performance of Problem (41) on this example is consistently better than the one of the nominal approach. It is important to understand here that in this specific example, the difference between Figures 2 and 4 lies in the cost function that evaluates the performance of the nominal and proposed approach. Indeed, because of the approximation  $p_{\text{real}} = \hat{p}_0$ , the original approach (19) appears potentially counterproductive under its targeted performance metric (10). Instead, the proposed performance metric (24) is the one that can be minimized via exploiting measurements for subsequent optimizations. In general, however, the inputs selected by (10) and (24) are different.



**Figure 3.** Comparison of the nominal and optimal experimental design on the proposed example for  $\hat{p}_0 = 0$ . The displayed cost is calculated according to the cost proposed in (24), which reads as  $\mathbb{E}_e \left[ \frac{1}{2} (u_0 - (\hat{p}_0 - e_0))^2 + \frac{1}{2} (u_1^* (\hat{p}_1) - (\hat{p}_0 - e_0))^2 \right]$  in this example. It can be observed that the optimal experimental design approach has two solutions, due to the non-convexity of the problem.



**Figure 4.** The left graph illustrates the nominal and optimal experimental design performance on the proposed example. The displayed cost is calculated according to the cost proposed in (24), which reads as  $\mathbb{E}_e \left[ \frac{1}{2}(u_0 - (\hat{p}_0 - e_0))^2 + \frac{1}{2}(u_1^* (\hat{p}_1) - (\hat{p}_0 - e_0))^2 \right]$  in this example. The right graph displays the corresponding inputs. Observe that the right-hand graph ought to be compared to the right-hand graph of Figure 2.

### 3. Link to the Modifier Approach and the DMA Approximation

In this section, we draw a connection between the proposed developments and the well-proven modifier approach tackled via the recent Directional-Modifier Adaptation (DMA) algorithm [23,24]. In particular, we show that the DMA approach can be construed as an approximation of Problem (41).

#### 3.1. The Modifier Approach

In the context of RTO, instead of considering uncertain model parameters, the Modifier Approach (MA) tackles the difficulty of working with uncertain process models by introducing a modification of the gradient of the cost function in the optimization problem. The MA then considers a model of the cost function in the form:

$$\phi(u, p) = \phi_0(u) + p^T u, \tag{48}$$

where  $p$  is a set of parameters that modifies the gradient of the process model. Hence, instead of refining the process model, the MA focuses on adjusting the cost gradient at the solution in order to reach optimality for the real process. At each run, measurements of the cost function can be used to improve the estimation of the process gradients via numerical differences. The measurements obtained at each run can be written as:

$$y^{\text{real}} = \frac{\phi(u_0, p^{\text{real}}) - \phi(u_{-1}, p^{\text{real}})}{\|u_0 - u_{-1}\|}, \tag{49}$$

while the measurement model reads as:

$$y(u_0, p) = \frac{\phi(u_0, p) - \phi(u_{-1}, p)}{\|u_0 - u_{-1}\|} = p^T \frac{u_0 - u_{-1}}{\|u_0 - u_{-1}\|} + \frac{\phi_0(u_0) - \phi_0(u_{-1})}{\|u_0 - u_{-1}\|}. \tag{50}$$

Here, we consider the inputs prior to  $u_0$  as fixed, since they are already realized, and we consider that a parameter estimation  $\hat{p}_0$  is available from these previous measurements, with associated covariance  $\Sigma_0$ . It can be verified that:

$$\phi_{pu}^* (\phi_{uu}^*)^{-1} \phi_{up}^* = \nabla^2 \phi_0^{-1}, \quad F(u_0, \Sigma, \hat{p}_0) = \Sigma_{\hat{p}_0}^{-1} + \Sigma_{\text{meas}}^{-1} S(u_0), \tag{51}$$

where  $\Sigma_{\text{meas}} \in \mathbb{R}$  is the covariance of the measurements of the numerical gradients of the process cost function and where we have defined:

$$S(u_0) = \frac{(u_0 - u_{-1})(u_0 - u_{-1})^\top}{\|u_0 - u_{-1}\|^2}. \quad (52)$$

Hence, Problem (41) deployed on the MA approach solves the problem:

$$\min_{u_0} \phi(u_0, \hat{p}_0) + \frac{1}{2} \text{Tr} \left[ \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} \left( I + \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^{-1} \right]. \quad (53)$$

### 3.2. DMA as an Approximation of (41)

We will consider next a 1<sup>st</sup>-order Neumann expansion to approximate Problem (53) for  $u_0 \approx u_{-1}$ . We observe that:

$$\left( I + \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^{-1} = I - \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) + R, \quad (54)$$

where:

$$R = \left( I + \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^{-1} \left( \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^2. \quad (55)$$

If the covariance  $\Sigma_\phi$  associated with the measurements of the cost function is fixed, then  $\Sigma_{\text{meas}}^{-1} = \frac{1}{2} \Sigma_\phi^{-1} \|u_0 - u_{-1}\|^2$ . It follows that for  $\|u_0 - u_{-1}\|$  small, the following approximation is asymptotically exact:

$$\text{Tr} \left[ \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} \left( I + \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^{-1} \right] \approx \text{Tr} \left[ \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} - \Sigma_{\text{meas}}^{-1} \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} \Sigma_{\hat{p}_0} S(u_0) \right] \quad (56)$$

$$= \text{Tr} \left[ \nabla^2 \phi_0^{-1} \Sigma_{\hat{p}_0} \right] - \Sigma_{\text{meas}}^{-1} (u_0 - u_{-1})^\top \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} \Sigma_{\hat{p}_0} (u_0 - u_{-1}). \quad (57)$$

One can then consider the following approximation of Problem (53):

$$u_0 = \arg \min_{u_0} \phi_0(u_0) + \hat{p}_0^\top u_0 - \frac{1}{2} \Sigma_{\text{meas}}^{-1} (u_0 - u_{-1})^\top \Sigma_{\hat{p}_0} \nabla^2 \phi_0^{-1} \Sigma_{\hat{p}_0} (u_0 - u_{-1}), \quad (58)$$

which is valid for  $\|u_0 - u_{-1}\|$  small. The DMA approach computes a direction  $\delta u$  in the input space according to:

$$\max_{\delta u} \delta u^\top \Sigma_{\nabla \phi} \delta u \quad (59a)$$

$$\text{s.t. } \|\delta u\| = 1, \quad \delta u \in C(U_r), \quad (59b)$$

where  $U_r = I$  trivially holds in the unconstrained case and then solves the problem:

$$u_0 = \arg \min_{u_0} \phi_0(u) + \hat{p}_0^\top (u_0 - u_{-1}) - \frac{c}{2} \left( \delta u^\top (u_0 - u_{-1}) \right)^2, \quad (60)$$

which is equivalent to:

$$u_0 = \arg \min_{u_0} \phi_0(u_0) + \hat{p}_0^\top u_0 - \frac{c}{2} (u_0 - u_{-1})^\top Q (u_0 - u_{-1}) \quad (61)$$

for the semi-positive, rank-one weighting matrix  $Q = c\delta u\delta u^T$ . The close resemblance of the DMA problem (61) to Problem (58) offers a deeper understanding of the procedure at play in the DMA algorithm. More specifically, Problem (58) is identical to the DMA problem (61) if:

$$\Sigma_{\text{meas}}^{-1}\Sigma_{\hat{p}_0}\nabla^2\phi_0^{-1}\Sigma_{\hat{p}_0} = cQ. \quad (62)$$

We observe here that  $\nabla\phi = \nabla\phi_0 + \hat{p}_0$ , such that  $\Sigma_{\nabla\phi} \equiv \Sigma_{\hat{p}_0}$  mathematically holds. Since  $\delta u$  is the dominant unitary eigenvector of  $\Sigma_{\nabla\phi}$  and is therefore also the dominant unitary eigenvector of  $\Sigma_{\hat{p}_0}^2$ , it follows that matrix  $Q$  is given by:

$$\max \quad \left\langle \Sigma_{\hat{p}_0}^2, Q \right\rangle \quad (63a)$$

$$\text{s.t.} \quad \|Q\| = 1, \quad \text{rank}(Q) = 1. \quad (63b)$$

Observing (62) and (63), it follows that the classical DMA method picks an input using:

- the approximation  $\nabla^2\phi_0 \approx \gamma I$  for some  $\gamma > 0$
- a rank-one approximation of  $\Sigma_{\hat{p}_0}^2$

According to these observations, a reasonable choice for the scaling constant  $c$  can be:

$$c = \Sigma_{\nabla\phi}^{-1} \left\| \Sigma_{\hat{p}_0} \nabla^2\phi_0^{-1} \Sigma_{\hat{p}_0} \right\|. \quad (64)$$

It is useful to remark here that dismissing the information provided by  $\nabla^2\phi_0$  may be advantageous when  $\phi_0$  does not reflect adequately the curvature of the cost function of the real process. In such a case, the weighting provided by  $\nabla^2\phi_0$  in (58) can arguably be misleading. Including estimations of the 2<sup>nd</sup>-order sensitivities in the MA approach has been investigated in [19].

### 3.3. Illustrative Example

We illustrate here the developments proposed above via a simple quadratic example, which nonetheless captures a number of observations that ought to be made. Consider the cost model:

$$\phi_0 = \frac{1}{2}u_0^T R u_0 + f^T u_0, \quad (65)$$

such that the nominal optimal input is trivially given by:

$$u_0 = -R^{-1}f. \quad (66)$$

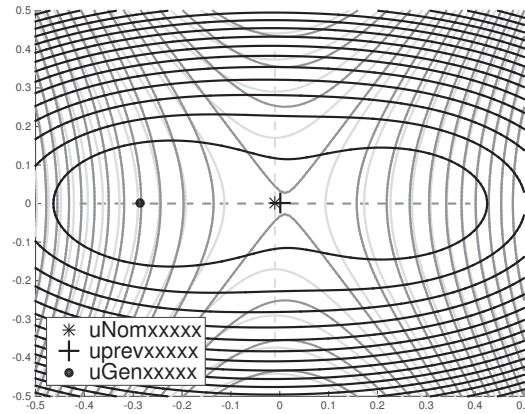
Problem (53) then reads as:

$$\min_{u_0} \quad \frac{1}{2}u_0^T R u_0 + f^T u_0 + \frac{1}{2}\text{Tr} \left[ \Sigma_{\hat{p}_0} R^{-1} \left( I + \Sigma_{\nabla\phi}^{-1} \Sigma_{\hat{p}_0} S(u_0) \right)^{-1} \right], \quad (67)$$

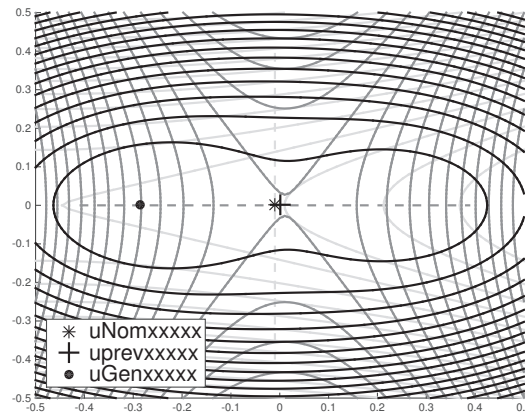
while the approximate problem (58) reads as:

$$u_0 = \arg \min_{u_0} \quad \frac{1}{2}u_0^T \left( R - \Sigma_{\nabla\phi}^{-1} \Sigma_{\hat{p}_0} R^{-1} \Sigma_{\hat{p}_0} \right) u_0 + \left( \hat{p}_0 + f - u_{-1}^T \Sigma_{\nabla\phi}^{-1} \Sigma_{\hat{p}_0} R^{-1} \Sigma_{\hat{p}_0} \right)^T u_0. \quad (68)$$

Note that Problem (68) is unbounded for  $\Sigma_{\nabla\phi} I < (R^{-1}\Sigma_{\hat{p}_0})^2$ , while (67) can have a well-defined solution; see Figures 5 and 6 for an illustration. This situation occurs here when the measurement noise is small while the current parameter estimation is highly uncertain and is discontinued when the parameter estimation becomes reliable, such that  $\Sigma_{\hat{p}_0}$  becomes small. Note that this can be addressed in practice via an ad hoc regularization or by, e.g., bounding the input correction  $\|u_0 - u^*(\hat{p}_0)\|$  in Problem (58).



**Figure 5.** Example of the problem where the quadratic approximation (58) is unbounded, while (53) has a solution. The black lines report the level curves of the cost of Problem (67); the grey lines report the level curves of the cost of Problem (68); and the light grey lines report the level curves of the cost of Problem (69) with  $Q$  given by (63). In this example, ignoring the contribution of  $\nabla^2\phi_0$  in the Directional-Modifier Adaptation (DMA) algorithm leads it to privilege directions (light grey dashed line) that are significantly different from the ones privileged by (67) (grey dashed line). The latter point to the solution of the original Problem (53).



**Figure 6.** Example of problem where the quadratic approximation (58) is unbounded, while (53) has a solution. The black lines report the level curves of the cost of Problem (67); the grey lines report the level curves of the cost of Problem (68); and the light grey lines report the level curves of the cost of Problem (69) where  $Q = \|\Sigma_{\hat{p}_0}\|^{-2} \Sigma_{\hat{p}_0}^2$  and  $c = \Sigma_{\nabla\phi}^{-1} \|\Sigma_{\hat{p}_0} \nabla^2\phi_0^{-1} \Sigma_{\hat{p}_0}\|$ . Adopting a matrix  $Q$  delivering a full-rank approximation of  $\Sigma_{\hat{p}_0}^2$  does not help the DMA algorithm adopting directions (see the light-grey dashed line) that point to the direction of the solution to (67); hence, ignoring  $\nabla^2\phi_0$  is problematic here.

The DMA-based problem (60) reads as:

$$u_0 = \arg \min_{u_0} \frac{1}{2} u_0^\top (R - cQ) u_0 + \left( \hat{p}_0 + f - \frac{c}{2} u_{-1}^\top Q \right)^\top u_0. \tag{69}$$

The behaviors of the DMA problem (69) and its proposed counterpart (68) are reported in Figures 5–8. In Figures 5 and 6, the two problems are compared for the setup:

$$Q = \begin{bmatrix} 0.5060 & 0 \\ 0 & 1.2358 \end{bmatrix}, \quad f = 0, \quad u_- = 0, \tag{70}$$

$$\hat{p}_0 = \begin{bmatrix} 5 \cdot 10^{-3} \\ 0 \end{bmatrix}, \quad \Sigma_{\hat{p}_0} = \begin{bmatrix} 0.0990 & 0 \\ 0 & 0.1638 \end{bmatrix}, \quad \Sigma_{\nabla\phi} = 0.0202,$$

resulting in an unbounded problem for both problems. In this case, the DMA approach (69) with a reduced choice of  $c$  would ensure a bounded problem, while a regularization or trust-region technique for Problem (68) would deliver a solution. We observe in Figures 5 and 6 that ignoring the term  $\nabla^2\phi_0$  in the DMA problem can lead the algorithm to favor a solution that departs significantly from the ones proposed by (53).

In Figure 7, the two problems are compared for the setup:

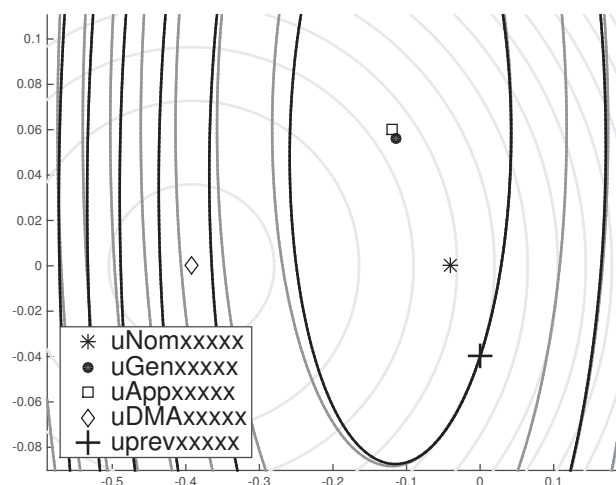
$$Q = I, \quad f = 0, \quad u_- = \begin{bmatrix} 0 \\ -0.0444 \end{bmatrix}, \quad \hat{p}_0 = \begin{bmatrix} 0.04 \\ 0 \end{bmatrix}, \quad \Sigma_{\hat{p}_0} = \begin{bmatrix} 1 & 0 \\ 0 & 0.95 \end{bmatrix}, \quad \Sigma_{\nabla\phi} = 1.5. \tag{71}$$

In this case, ignoring the term  $\nabla^2\phi_0 = Q = I$  does not yield any difficulty. However, because all parameters  $\hat{p}_0$  have a very similar covariance, the rank-one approximation of  $\Sigma_{\hat{p}_0}$  misleads the DMA algorithm into selecting a solution that departs significantly from the one of (53). Finally, in Figure 8, the two problems are compared for the setup:

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0.8 \end{bmatrix}, \quad f = 0, \quad u_- = \begin{bmatrix} 0 \\ -0.05 \end{bmatrix}, \tag{72}$$

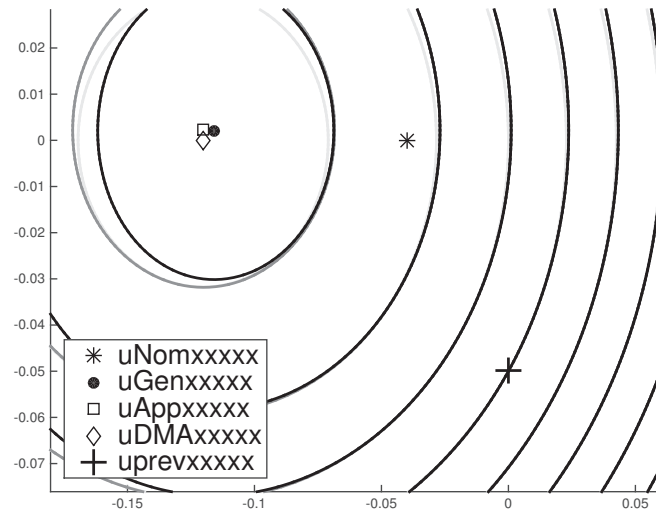
$$\hat{p}_0 = \begin{bmatrix} 0.04 \\ 0 \end{bmatrix}, \quad \Sigma_{\hat{p}_0} = \begin{bmatrix} 1 & 0 \\ 0 & 0.2 \end{bmatrix}, \quad \Sigma_{\nabla\phi} = 1.5.$$

In this last case, both the DMA problem (69) and (68) deliver solutions that are very close to the one of Problem (53), i.e., in this scenario, ignoring the term  $\nabla^2\phi_0$  and forming a rank-one approximation do not affect the solution significantly.



**Figure 7.** Illustration for Section 3.3, setup (71). The black lines report the level curves of the cost of Problem (67); the grey lines report the level curves of the cost of Problem (68); and the light grey lines report the level curves of the cost of Problem (69) with  $Q$  given by (63). In this example, the rank-one approximation for  $Q$  leads the DMA algorithm to propose a solution that is far from the one of (67).





**Figure 8.** Illustration for Section 3.3, setup (72). The black lines report the level curves of the cost of Problem (67); the grey lines report the level curves of the cost of Problem (68); and the light grey lines report the level curves of the cost of Problem (69) with  $Q$  given by (63). In this example, all problems deliver very similar solutions.

#### 4. Conclusions

In this paper, we have proposed a novel view of real-time optimization and of the modifier approach from an experimental design perspective. While some methods are available to handle the trade-off between process optimality and the gathering of information for the performance of future runs, this paper proposes a formal framework to construe this trade-off as an optimization problem and develops a tractable approximation of this problem. The paper then shows that the recent directional-modifier adaptation algorithm is a special formulation of this approximation. This observation allows one to further justify the directional-modifier adaptation algorithm from a theoretical standpoint and to consider a refined tuning of the algorithm. The theory presented in the paper is illustrated via simple examples.

**Acknowledgments:** This research was partially supported by Freiburg Institute for Advanced Studies (FRIAS), University of Freiburg, Germany. Helpful remarks have been provided by B. Houska.

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

#### Appendix

We provide here the lemma used in Proposition 2.

**Lemma 3.** If  $x \in \mathbb{R}^m$  is a normally distributed, centered variable of covariance  $\Sigma \in \mathbb{R}^{m \times m}$  and  $f(x) : \mathbb{R}^m \rightarrow \mathbb{R}$  a smooth function polynomially bounded as:

$$|f(x)| \leq P_{m,n}(x) \tag{A1}$$

for some  $n$ -th-order polynomial of the form  $P_{m,n}(x) = \sum_{k=m}^n \alpha_k \|x\|^k$ , then the following inequality holds locally:

$$|\mathbb{E}_x [f(x)]| \leq c \|\Sigma\|^{\text{ceil}(\frac{m}{2})}. \tag{A2}$$

**Proof.** We first observe that:

$$\mathbb{E}_x [P_{m,n}(x)] \leq \sum_{k=m}^n \beta_k \|\Sigma\|_{\infty}^{\frac{k}{2}} \tag{A3}$$

holds for some suite  $\beta_k \geq 0$ , with  $\beta_k = 0$  for  $k$  odd. This is a direct consequence of the generalized Isserlis theorem [27,28], which states that the expected value of any even-order moment of a multivariate normal centered distribution is a sum of products between  $k/2$  entries of the covariance matrix  $\Sigma$ , while odd-order moments are null. It then also holds that:

$$|\mathbb{E}_x [f(x)]| \leq \sum_{k=m}^n \beta_k \|\Sigma\|_{\infty}^{\frac{k}{2}}, \quad (\text{A4})$$

and the inequality:

$$|\mathbb{E}_x [f(x)]| \leq c \|\Sigma\|^{\text{ceil}(\frac{m}{2})} \quad (\text{A5})$$

holds locally.  $\square$

We observe that this Lemma appears to be a simple special case of the Theorem proposed by [26] on the delta method, restricted to the normal distribution.

## References

1. Chen, C.; Joseph, B. On-line optimization using a two-phase approach: An application study. *Ind. Eng. Chem. Res.* **1987**, *26*, 1924–1930.
2. Jang, S.; Joseph, B. On-line optimization of constrained multivariable chemical processes. *Ind. Eng. Chem. Res.* **1987**, *33*, 26–35.
3. Forbes, J.; Marlin, T.; MacGregor, J. Model adequacy requirements for optimizing plant operations. *Comput. Chem. Eng.* **1994**, *18*, 497–510.
4. 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.
5. Agarwal, M. Feasibility of on-line reoptimization in batch processes. *Chem. Eng. Commun.* **1997**, *158*, 19–29.
6. Agarwal, M. Iterative set-point optimization of batch chromatography. *Comput. Chem. Eng.* **2005**, *29*, 1401–1409.
7. Marchetti, A. Modifier-Adaptation Methodology for Real-Time Optimization. Ph.D. Thesis, EPFL, Lausanne, Switzerland, 2009.
8. Roberts, P. An algorithm for steady-state system optimization and parameter estimation. *Int. J. Syst. Sci.* **1979**, *10*, 719–734.
9. Gao, W.; Engell, S. Comparison of iterative set-point optimization strategies under structural plant-model mismatch. *IFAC Proc. Vol.* **2005**, *16*, 401.
10. Marchetti, A.; Chachuat, B.; Bonvin, D. Modifier-adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.* **2009**, *48*, 6022–6033.
11. Roberts, P. Coping with model-reality differences in industrial process optimisation—A review of integrated system optimization and parameter estimation (ISOPE). *Comput. Ind.* **1995**, *26*, 281–290.
12. Tatjewski, P. Iterative optimizing set-point control—the basic principle redesigned. *IFAC Proc. Vol.* **2002**, *35*, 49–54.
13. François, G.; Bonvin, D. Use of convex model approximations for real-time optimization via modifier adaptation. *Ind. Eng. Chem. Res.* **2014**, *52*, 11614–11625.
14. Bunin, G.; Wuillemin, Z.; François, G.; Nakajo, A.; Tsikonis, L.; Bonvin, D. Experimental real-time optimization of a solid oxide fuel cell stack via constraint adaptation. *Energy* **2012**, *39*, 54–62.
15. Serralunga, F.; Mussati, M.; Aguirre, P. Model adaptation for real-time optimization in energy systems. *Ind. Eng. Chem. Res.* **2013**, *52*, 16795–16810.
16. Navia, D.; Marti, R.; Sarabia, R.; Gutierrez, G.; Prada, C. Handling infeasibilities in dual modifier-adaptation methodology for real-time optimization. *IFAC Proc. Vol.* **2012**, *45*, 537–542.
17. Darby, M.; Nikolaou, M.; Jones, J.; Nicholson, D. RTO: An overview and assessment of current practice. *J. Process Control* **2011**, *21*, 874–884.

18. Costello, S.; François, G.; Bonvin, D.; Marchetti, A. Modifier adaptation for constrained closed-loop systems. *IFAC Proc. Vol.* **2014**, *47*, 11080–11086.
19. Faulwasser, T.; Bonvin, D. On the Use of Second-Order Modifiers for Real-Time Optimization. In Proceedings of the 19th IFAC World Congress, Cape Town, South Africa, 24–29 August 2014.
20. Bunin, G.; 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.
21. Serralunga, F.; Aguirre, P.; Mussati, M. Including disjunctions in real-time optimization. *Ind. Eng. Chem. Res.* **2014**, *53*, 17200–17213.
22. Houska, B.; Telenb, D.; Logist, F.; Diehl, M.; Van Impe, J.F.M. An economic objective for the optimal experiment design of nonlinear dynamic processes. *Automatica* **2015**, *51*, 98–103.
23. Costello, S.; François, G.; Bonvin, D. Directional Real-Time Optimization Applied to a Kite-Control Simulation Benchmark. In Proceedings of the European Control Conference 2015, Linz, Austria, 15–17 July 2015.
24. Costello, S.; François, G.; Bonvin, D. A directional modifier-adaptation algorithm for real-time optimization. *J. Process Control* **2016**, *39*, 64–76.
25. Marchetti, A.; Chachuat, B.; Bonvin, D. A dual modifier-adaptation approach for real-time optimization. *J. Process Control* **2010**, *20*, 1027–1037.
26. Oehlert, G. A note on the delta method. *Am. Stat.* **1992**, *46*, 27–29.
27. Withers, C. The moments of the multivariate normal. *Bull. Aust. Math. Soc.* **1985**, *32*, doi:10.1017/S000497270000976X.
28. Vignat, C. A generalized Isserlis theorem for location mixtures of Gaussian random vectors. *Stat. Probab. Lett.* **2012**, *82*, 67–71.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).