2025 23rd European Control Conference (ECC)
June 24-27, 2025. Thessaloniki, Greece

Sensitivity of Online Feedback Optimization to time-varying parameters

Marta Zagorowska[1,2], Lars Imsland[1]

***Abstract*— Online Feedback Optimization uses optimization algorithms as dynamic systems to find optimal control inputs. The results obtained from Online Feedback Optimization depend on the setup of the chosen optimization algorithm. In this work we analyse the sensitivity of Online Feedback Optimization to the parameters of projected gradient descent as the algorithm of choice. We derive closed-form expressions for sensitivities of the objective function with respect to the parameters of the projected gradient and to model mismatch. The formulas are then used for analysis of model mismatch in a gas lift optimization problem. The results of the case study indicate that the sensitivity of Online Feedback Optimization to the model mismatch depends on how long the controller has been running, with decreasing sensitivity to mismatch in individual timesteps for long operation times.**

I. Introduction

Online Feedback Optimization (OFO) relies on using optimization algorithms as dynamic systems to find the optimal control inputs. However, the parameters of the optimization algorithm are typically separate from the controlled system, making analysis of their impact challenging. In this work, we provide an analysis of the sensitivity of OFO to parameters of projected gradient descent used as the algorithm of choice, as well as to time-varying model mismatch.

Sensitivity of optimization-based controllers, such as Model Predictive Control (MPC), allows shaping the performance of the controller. Sensitivity to MPC parameters has been provided in [1] and the results were used to find optimal controller parameters. The sensitivities of MPC with respect to uncertain model parameters were used to find a robust controller in [14]. Extending the robustness and tuning capabilities to Online Feedback Optimization by using sensitivity analysis is the motivation for this work.

A review of OFO controllers was done in [7]. The sensitivity of OFO from the perspective of robustness to model mismatch was demonstrated experimentally [6], [11]. A theoretical analysis of robustness for OFO with relaxed output constraints was provided in [5]. A more general approach to sensitivity of optimization-based controllers was adopted in [9] which indicated conditions for OFO to be continuously differentiable, without providing explicit formulas for the sensitivity. In this paper, we use the fact that OFO with projected gradient descent relies on solving a series of convex quadratic optimization problems to derive closed form expressions for sensitivity.

*Research supported by Marie Curie Horizon Postdoctoral Fellowship project RELIC (grant no 101063948)

[1] Department of Engineering Cybernetics, Norwegian University of Science and Technology, email: lars.imsland@ntnu.no

[2] Currently with Delft Center for Systems and Control, TU Delft, email: m.a.zagorowska@tudelft.nl

Online Feedback Optimization is introduced in Section II. Section III presents the closed-formed expressions for sensitivity of OFO, which are then numerically validated in Section IV in a gas lift optimization problem. Section V concludes the paper and indicates future directions.

1) Notation: This paper follows the notational convention for vector and matrix derivatives from [8]. In particular, we have $\frac{\partial \mathbf{y}}{\partial x} \in \mathbb{R}^{n_y}$ if $\mathbf{y} \in \mathbb{R}^{n_y}$ and $\frac{\partial y}{\partial \mathbf{x}} \in \mathbb{R}^{1\times n_x}$ if $\mathbf{x} \in \mathbb{R}^{n_x}$.

II. Online Feedback Optimization

A. Problem statement

Online Feedback Optimization solves:

$$\min_{u,y} \quad \Phi(u,y) \tag{1a}$$
$$\text{subject to } y = h(u) \tag{1b}$$
$$Au \leq b \tag{1c}$$
$$Cy \leq d \tag{1d}$$

where $\Phi : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}$ is a continuously differentiable cost function, $h : \mathbb{R}^{n_u} \to \mathbb{R}^{n_y}$ is a continuously differentiable nonlinear input-output mapping where $A \in \mathbb{R}^{n_{c_1}\times n_u}$, $b \in \mathbb{R}^{n_{c_1}}$, $C \in \mathbb{R}^{n_{c_2}\times n_y}$, $d \in \mathbb{R}^{n_{c_2}}$ are constant matrices [6], and c_1, c_2 denote the number of input and output constraints, respectively. The controller in this work uses projected gradient descent as the optimization algorithm, with a constant step size $\alpha > 0$ [6]:

$$u^{k+1} = u^k + \alpha \widehat{\sigma}_\alpha(u^k, y^k) \tag{2}$$

where $y^k = h(u^k)$ is the measured system output at time k, and $\widehat{\sigma}_\alpha(u^k, y^k)$ solves (superscript ^{k} dropped for space reasons):

$$\widehat{\sigma}_\alpha(u,y) = \underset{w\in\mathbb{R}^{n_u}}{\arg\min} \left|\left|w + G^{-1}H^\top(u)\nabla\Phi^\top(u,y)\right|\right|_G^2 \tag{3a}$$
$$\text{subject to} \quad A\left(u+\alpha w\right) \leq b \tag{3b}$$
$$C\left(y + \alpha\nabla h(u) w\right) \leq d \tag{3c}$$

where $H(u)^\top = \begin{bmatrix}\mathbb{I}_{n_u} & \nabla h(u)^\top\end{bmatrix}$, $w \in \mathbb{R}^{n_u}$, and $G \in \mathbb{R}^{n_u\times n_u}$ is a positive-definite scaling matrix. The matrix $\mathbb{I}_{n_u}$ is an identity matrix of size $n_u \times n_u$. The formulation (3) requires only the gradient $\nabla h(u)$ instead of the full mapping [6].

B. OFO as a function of parameters

For simplicity, we introduce $\mathbf{p} = [p_s]_{s=0,\dots,k}$, $p_s \in \mathbb{R}^{n_p}$, as the parameter whose changes affect the objective at time k and thus $\Phi^k := \Phi(u^k(\mathbf{p}), y^k(\mathbf{p}), \mathbf{p})$ The parameters $\mathbf{p}$ may include the elements of matrices A, b, C, d, as well as parameters of the derivatives ∇h and $\nabla\Phi$. Usually, G in (3) is chosen as a constant matrix and will be treated as a parameter together with α.

At every iteration k, the constraints (3b) and (3c) are linear with respect to w: $\bar{\bar{A}}w \leq \bar{\bar{b}}$ with $\bar{\bar{A}} = \begin{bmatrix} A\alpha \\ C\alpha\nabla h(u^k) \end{bmatrix} \in \mathbb{R}^{\bar{\bar{n}}\times n_u}$, $\bar{\bar{b}} = \begin{bmatrix} b - Au^k \\ d - Cy^k \end{bmatrix} \in \mathbb{R}^{\bar{\bar{n}}}$ where $\bar{\bar{n}} = n_{c_1} + n_{c_2}$ denotes the total number of constraints in the optimization problem. Then we rewrite the objective function (3a) to get a quadratic formulation $\left|\left|w + G^{-1}H^\top\nabla\Phi^\top\right|\right|_G^2 = \frac{1}{2}w^\top\bar{\bar{G}}w + w^\top\bar{\bar{c}} + \bar{\bar{M}}$ where $\bar{\bar{G}} := 2G$ preserves the positive definite and symmetric properties of G, $\bar{\bar{c}} := 2H^\top\nabla\Phi^\top \in \mathbb{R}^{n_u}$. We note that $\bar{\bar{M}} := (G^{-1}H^\top\nabla\Phi^\top)^\top H^\top\nabla\Phi^\top$ is constant and independent of w so it is omitted and (3) becomes

$$\min_{w\in\mathbb{R}^{n_u}} \frac{1}{2}w^\top\bar{\bar{G}}w + w^\top\bar{\bar{c}} \tag{4a}$$

$$\text{subject to} \quad \bar{\bar{A}}w \leq \bar{\bar{b}} \tag{4b}$$

where the arguments u, y are omitted for space reasons. The Lagrangian of (4) is $\mathcal{L}(w) = \frac{1}{2}w^\top\bar{\bar{G}}w + w^\top\bar{\bar{c}} + \sum_{j=1}^{\bar{\bar{n}}} \lambda_j(\bar{\bar{a}}_j w - \bar{\bar{b}}_j)$ where $\bar{\bar{a}}_j$ is the j-th row of the matrix $\bar{\bar{A}}$, $\bar{\bar{b}}_j$ is the j-th element of $\bar{\bar{b}}$, and $\lambda_j \geq 0$, $j = 1, \ldots, \bar{\bar{n}}$ are dual variables. The matrix $\bar{\bar{G}}$ is positive definite, thus the problem (4) is convex and the KKT conditions give optimality:

$$\nabla_w\mathcal{L}(w) = \bar{\bar{G}}w + \bar{\bar{c}} + \sum_{j=1}^{\bar{\bar{n}}} \lambda_j\bar{\bar{a}}_j^\top = 0 \tag{5a}$$

$$\bar{\bar{a}}_i w - \bar{\bar{b}}_i \leq 0, \forall i = 1, \ldots, \bar{\bar{n}} \tag{5b}$$

$$\lambda_i(\bar{\bar{a}}_i w - \bar{\bar{b}}_i) = 0, \forall i = 1, \ldots, \bar{\bar{n}} \tag{5c}$$

$$\lambda_i \geq 0, \forall i = 1, \ldots, \bar{\bar{n}} \tag{5d}$$

For the remainder of the paper we will assume that the QP in (4) is non-degenerate in the sense introduced by [4] and thus (5) has a solution. Following [2], we will use (5) to find the sensitivity of OFO.

III. Sensitivity of Online Feedback Optimization

A. Sensitivity to problem parameters

The matrices $\bar{\bar{A}}$, $\bar{\bar{b}}$, $\bar{\bar{c}}$, and $\bar{\bar{G}}$ change with iterations s because they depend on current measurements u^s, y^s, or on the parameters p_s. At time k we want to find:

$$\frac{\partial\Phi^k}{\partial p_s} = \frac{\partial\Phi^k}{\partial p_s} + \left(\frac{\partial\Phi^k}{\partial u^k} + \frac{\partial\Phi^k}{\partial y^k}\cdot\frac{\partial y^k}{\partial u^k}\right)\cdot\frac{\partial u^k}{\partial p_s} \in \mathbb{R}^{1\times n_p} \tag{6}$$

for $s = 0, \ldots, k$. From (2), we see that u^k and y^k depend only on p_s for $s = 0, \ldots, k-1$, and hence $\left(\frac{\partial\Phi^k}{\partial u^k} + \frac{\partial\Phi^k}{\partial y^k}\cdot\frac{\partial y^k}{\partial u^k}\right)\cdot\frac{\partial u^k}{\partial p_k} = 0$ in (6). As a result, $\frac{\partial\Phi^k}{\partial p_k}$ corresponds to computing instantaneous sensitivity of Φ^k with respect to parameter p^k, which is unaffected by OFO and will be omitted from further analysis.

We also see that $\frac{\partial\Phi^k}{\partial u^k} + \frac{\partial\Phi^k}{\partial y^k}\cdot\frac{\partial y^k}{\partial u^k}$ in (6) is independent of s and thus we can write in matrix form:

$$\begin{bmatrix} \frac{\partial\Phi^k}{\partial p_{k-1}} \\ \frac{\partial\Phi^k}{\partial p_{k-2}} \\ \vdots \\ \frac{\partial\Phi^k}{\partial p_0} \end{bmatrix} = \mathbb{I}_k \otimes \left(\frac{\partial\Phi^k}{\partial u^k} + \frac{\partial\Phi^k}{\partial y^k}\cdot\frac{\partial y^k}{\partial u^k}\right)\cdot\begin{bmatrix} \frac{\partial u^k}{\partial p_{k-1}} \\ \frac{\partial u^k}{\partial p_{k-2}} \\ \vdots \\ \frac{\partial u^k}{\partial p_0} \end{bmatrix} \tag{7}$$

where $\otimes$ denotes the Kronecker product.

From (7) we obtain the total derivative of Φ at time k:

$$\mathrm{d}\Phi^k = \sum_{i=0}^{k-1}\frac{\partial\Phi^k}{\partial p_i}\cdot\mathrm{d}p_i \tag{8}$$

where $\mathrm{d}p_i$ are individual increments at time i.

B. Derivative of u

From (7), we need $\frac{\partial u^k}{\partial p_s} \in \mathbb{R}^{n_u\times n_p}$, $s = 0, \ldots, k-1$. As α and u_0 are independent of $\mathbf{p}$, we get:

$$\frac{\partial u^k}{\partial p_s} = \alpha\sum_{i=s}^{k-1}\frac{\partial w^i}{\partial p_s} \tag{9}$$

for $s = 0, \ldots, k-1$. In matrix form, we obtain:

$$\begin{bmatrix} \frac{\partial u^k}{\partial p_{k-1}} \\ \frac{\partial u^k}{\partial p_{k-2}} \\ \vdots \\ \frac{\partial u^k}{\partial p_0} \end{bmatrix} = \alpha\begin{bmatrix} \frac{\partial w^{k-1}}{\partial p_{k-1}} & 0 & \\ \frac{\partial w^{k-1}}{\partial p_{k-2}} & \frac{\partial w^{k-2}}{\partial p_{k-2}} & 0 \\ \vdots & & \vdots \\ \frac{\partial w^{k-1}}{\partial p_0} & \frac{\partial w^{k-2}}{\partial p_0} & \frac{\partial w^0}{\partial p_0} \end{bmatrix}\cdot\begin{bmatrix} \mathbb{I}_{n_p} \\ \vdots \\ \mathbb{I}_{n_p} \end{bmatrix} \tag{10}$$

Thus, we obtain a total derivative $\mathrm{d}u^k$ as:

$$\mathrm{d}u^k = \alpha\sum_{s=0}^{k-1}\sum_{i=s}^{k-1}\frac{\partial w^i}{\partial p_s}\mathrm{d}p_s \tag{11}$$

The result from (11) corresponds to the formulas provided by [15] for backpropagation in time.

C. Derivative of w

To obtain (11), we need to calculate $\frac{\partial w^i}{\partial p_s} \in \mathbb{R}^{n_u\times n_p}$. Taking into account that at time k the matrices $\bar{\bar{A}}$, $\bar{\bar{b}}$, $\bar{\bar{c}}$, $\bar{\bar{G}}$ depend on the inputs and the outputs up to $s = k-1$, and the parameters up to $s = k$, we compute $\frac{\partial w^k}{\partial p_s}$ using the chain rule:

$$\frac{\partial w^k}{\partial p_s} = \underbrace{K^s}_{=0 \text{ for } s=0,\ldots,k-1} + \underbrace{\mathrm{d}w_{\bar{\bar{b}}}\mathrm{d}_{p_s}\bar{\bar{b}} + \mathrm{d}w_{\bar{\bar{A}}}\mathrm{d}_{p_s}\bar{\bar{A}} + \mathrm{d}w_{\bar{\bar{c}}}\mathrm{d}_{p_s}\bar{\bar{c}} + \mathrm{d}w_{\bar{\bar{G}}}\mathrm{d}_{p_s}\bar{\bar{G}}}_{=0 \text{ for } s=k} \tag{12}$$

where $K^k = \mathrm{d}w_{\bar{\bar{b}}} + \mathrm{d}w_{\bar{\bar{A}}} + \mathrm{d}w_{\bar{\bar{c}}} + \mathrm{d}w_{\bar{\bar{G}}}$. From [2], the terms $\mathrm{d}w_{\bar{\bar{b}}}$, $\mathrm{d}w_{\bar{\bar{A}}}$, $\mathrm{d}w_{\bar{\bar{c}}}$, and $w_{\bar{\bar{G}}}$ are obtained by differentiating (5):

$$\begin{bmatrix} \bar{\bar{G}} & \bar{\bar{A}}^\top \\ D(\lambda^*)\bar{\bar{A}} & D(\bar{\bar{A}}w^* - \bar{\bar{b}}) \end{bmatrix}\cdot\begin{bmatrix} \mathrm{d}w_X \\ \mathrm{d}\lambda_X \end{bmatrix} = -\begin{bmatrix} \mathrm{d}\bar{\bar{G}}w^* + \mathrm{d}\bar{\bar{c}} + \mathrm{d}\bar{\bar{A}}^\top\lambda^* \\ D(\lambda^*)\mathrm{d}\bar{\bar{A}}w^* - D(\lambda^*)\mathrm{d}\bar{\bar{b}} \end{bmatrix} \tag{13}$$

where $D(x)$ creates a diagonal matrix with entries x, $X \in \{\bar{\bar{b}}, \bar{\bar{A}}, \bar{\bar{c}}, \bar{\bar{G}}\}$. We now need to compute derivatives of $\bar{\bar{A}}$, $\bar{\bar{b}}$, $\bar{\bar{c}}$, $\bar{\bar{G}}$ with respect to the parameters p_s. For instance, we have:

$$\mathrm{d}_{p_s}\bar{\bar{b}}(u^k, y^k) = \underbrace{\frac{\partial \bar{\bar{b}}(u^k, y^k)}{\partial p_s}}_{=0 \text{ for } s=0,\dots,k-1} + \underbrace{\left(\frac{\partial \bar{\bar{b}}(u^k, y^k)}{\partial u_k} + \frac{\partial \bar{\bar{b}}(u^k, y^k)}{\partial y_k} \cdot \frac{\partial y^k}{\partial u_k}\right) \cdot \frac{\partial u^k}{\partial p_s}}_{=0 \text{ for } s=k} \tag{14}$$

The formulas for $\mathrm{d}_{p_s}\bar{\bar{A}} \in \mathbb{R}^{\bar{\bar{n}} \times n_u n_p}$, $\mathrm{d}_{p_s}\bar{\bar{c}} \in \mathbb{R}^{n_u \times n_p}$, $\mathrm{d}_{p_s}\bar{\bar{G}} \in \mathbb{R}^{n_u \times n_u n_p}$ are obtained analogously.

We can now write:

$$\begin{bmatrix} \frac{\partial w^k}{\partial p_k} \\ \frac{\partial w^k}{\partial p_{k-1}} \\ \vdots \\ \frac{\partial w^k}{\partial p_0} \end{bmatrix} = \begin{bmatrix} K^k & 0 \\ 0 & \mathbb{I}_k \otimes K \end{bmatrix} \cdot \begin{bmatrix} \mathbb{I}_{n_u} \\ \frac{\partial u^k}{\partial p_{k-1}} \\ \frac{\partial u^k}{\partial p_{k-2}} \\ \vdots \\ \frac{\partial u^k}{\partial p_0} \end{bmatrix} \tag{15}$$

where $K = K_{\bar{\bar{b}}} + K_{\bar{\bar{A}}} + K_{\bar{\bar{c}}} + K_{\bar{\bar{G}}}$ and $K_X = \mathrm{d}w_X \left(\frac{\partial X}{\partial u_k} + \frac{\partial X}{\partial y_k} \cdot \frac{\partial y^k}{\partial u_k}\right)$, $X \in \{\bar{\bar{b}}, \bar{\bar{A}}, \bar{\bar{c}}, \bar{\bar{G}}\}$.

Inserting (15) into (10) and subsequently into (7), we obtain closed-form expressions for sensitivity of the objective function Φ at time k to the parameter $\mathbf{p}$.

D. Sensitivity to OFO parameters

We will now present sensitivities with respect to parameters of OFO: G, α, and u_0.

1) Sensitivity to G: The sensitivity of the control input and the objective function to the matrix G can be obtained from in Section III-A by taking $p_s = \mathrm{vec}(G)$.

2) Sensitivity to α: When computing the sensitivity with respect to α, the formulas (7) and (15) remain as in Section III-A. To obtain the equivalent to formula (10), we notice from (2) that:

$$u^k = u^0 + \sum_{s=0}^{k-1} \alpha_s w^s \tag{16}$$

where for the ease of notation we took $\alpha = \alpha_s$ as a time-varying parameter. Assuming that u^0 is independent of α_s and taking the derivatives of (16) at time k with respect to α_s, we get $\frac{\partial u^k}{\partial \alpha_s} = w^s + \sum_{i=s}^{k-1} \alpha_s \frac{\partial w^i}{\partial \alpha_s}$, which can be used to obtain the equivalent to (10).

3) Sensitivity to u_0: In this subsection we assume that the starting point for OFO at time $k = 0$ is $u^0 = u_0$ and analyse the sensitivity to its value at time $k \geq 1$. From (7), we get that $\frac{\partial \Phi^k}{\partial u_s} = 0$ for $s = 1, \dots, k-1$ and therefore $\frac{\partial \Phi^k}{\partial u_0} = \left(\frac{\partial \Phi^k}{\partial u_k} + \frac{\partial \Phi^k}{\partial y_k} \cdot \frac{\partial y^k}{\partial u_k}\right) \cdot \frac{\partial u^k}{\partial u_0}$. To obtain $\frac{\partial u^k}{\partial u_0}$ we use (16) and assume that α is independent of u_0 to obtain:

$$\frac{\partial u^k}{\partial u_0} = \mathbf{1}_{n_u} + \alpha \sum_{i=0}^{k-1} \frac{\partial w^i}{\partial u_0} \tag{17}$$

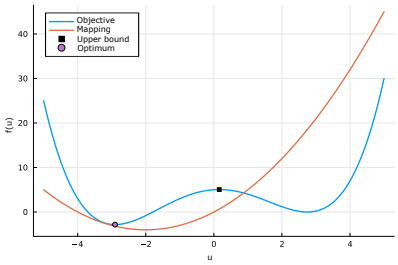


Fig. 1: The objective (18a) and the mapping (18b) with the optimum (circle) and the upper bound of u_0 (square)

where $\mathbf{1}_{n_u} = [1] \in \mathbb{R}^{n_u}$. Finally, to obtain $\frac{\partial w^i}{\partial u_0}$ in (17), we notice that the matrices from (4) are independent of u_0 in a direct way, so the solution w^k at time k will depend on u_0 through u^{k-1} and y^{k-1}. Thus, from (15) we get $\frac{\partial w^k}{\partial u_0} = K \cdot \frac{\partial u^k}{\partial u_0}$.

IV. Numerical results

A. Validation with finite differences

To validate the formulas obtained in Section III, we compared the derived sensitivities with the computations based on finite differences for a one-dimensional optimization problem adapted from [13] (Fig. 1):

$$\min_{u,y} \quad 0.1(u^2 y - 4uy + 5u) + 5 \tag{18a}$$

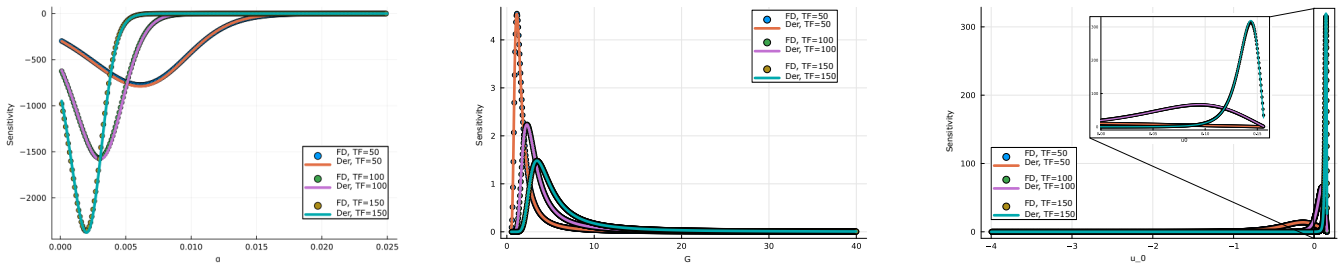
$$\text{subject to } y = u^2 + 4u \tag{18b}$$

$$u \in \mathcal{U}_i, y \in \mathcal{Y}, i = 1, 2 \tag{18c}$$

where $\mathcal{U}_1 = \mathcal{Y} = [-5, 5]$, $\mathcal{U}_2 = [-2, 2]$. The validation was done for $\alpha \in [0.0001, 0.025]$ with $\Delta\alpha = 1e-4$ for finite differences ($G = 1$, $u_0 = -0.63$), $G \in [0.5, 40]$ with $\Delta G = 5e-2$ ($\alpha = 0.01$, $u_0 = -0.63$), and $u_0 \in [-4, 0.156]$ with $\Delta u_0 = 5e-3$, ($G = 1$, $\alpha = 0.01$), for three simulation times $T_F \in \{50, 100, 150\}$.

1) Analysis for unconstrained optimum: We first validate the formulas from Section III for inactive constraints. Figure 2 shows a comparison of the derivatives from Section III-D (solid line) and finite differences (circles). For every time T_F, the results obtained from the formulas in Section III reflect the results from finite differences. This indicates the potential of using the sensitivities to locally approximate the objective as a function of the parameters without multiple runs required for finite differences.

We note that the sensitivities to the parameters change depending on the time horizon T_F. Figure 2a indicates that the sensitivities for $\alpha \geq 0.175$ are close to zero. This is because such α allows reaching the optimum of (1a), driving the derivatives with respect to u and y to zero in (6). However, the smaller the value of α, the more time the controller needs to reach the optimum [16]. As a result, the values $\alpha \leq 0.1$ lead to high sensitivity because the controller does not reach the optimum and the derivatives of the objective at T_F remain non-zero. The behaviour is further confirmed by the negative sign of the derivative in



(a) Sensitivity to α, $\alpha \in [0.0001, 0.025]$ (b) Sensitivity to G, $G \in [0.5, 40]$ (c) Sensitivity to u_0, $u_0 \in [-4, 0.156]$

(d) Sensitivity to α, $\alpha \in [0.0001, 0.025]$ (e) Sensitivity to G, $G \in [0.5, 40]$ (f) Sensitivity to u_0, $u_0 \in [-4, 0.156]$

Fig. 2: Comparison of the formulas from Section III-D (Der) to finite differences (FD) for different T_F

Fig. 2a, indicating that the objective (1a) is a decreasing function of α.

The relationship between G and α has been analysed in [16]. The sensitivity of the objective to the scaling parameter G in Fig. 2b indicates that the value of the objective increases with G. Conversely to the sensitivity to α, $G \leq 2$ allows reaching the optimum, driving the sensitivities to zero.

Figure 2c indicates increased sensitivity for the initial condition close to the boundary of the region of attraction of the sought minimum. An explanation for this increased sensitivity is provided by Fig. 1 and analysis of formula (3) from the perspective of $\nabla\Phi$. The objective function in (18a) (blue in Fig. 1) has a local maximum at 0.156 and thus $\nabla\Phi \approx 0$ in (3) for u_0 in the neighbourhood of the maximum. As a results, the solution $w \approx 0$ yielding slow increments in u from (2) leading to small changes in y from (18b). For shorter time periods, $T_F = 50$ and $T_F = 100$, the slow changes in u and y have little impact on the objective, thus keeping $\frac{\mathrm{d}\Phi}{\mathrm{d}u_0}$ smaller. Longer time periods, $T_F = 150$, allow OFO to escape the neighbourhood of the local maximum without reaching the optimum, leading to increased sensitivity of $\Phi(u_{T_F}, y_{T_F})$ to u_0. The optimum is reached if the final time is increased, $T_F = 300$, making the objective less sensitive. These results emphasise the importance of the choice of the time as well as the step size α.

2) Analysis for constrained optimum: The formulas from Section III remain valid if the constraints are active, as long as the KKT conditions (5) are nondegenerate. In particular, if the objective function (1a) becomes constant when either the input or the output constraints become active, the sensitivity becomes zero (Fig. 2d, 2e, 2f).

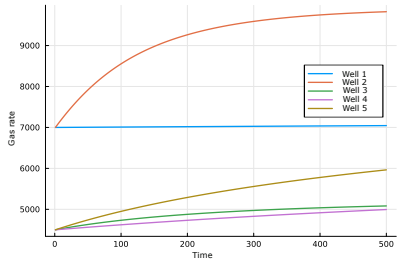
B. Usage

1) Gas lift optimization: The objective it to maximise the cumulative output of two floating oil platforms $\max_{\mathbf{u},\mathbf{y}} y_1(\mathbf{u}) + y_2(\mathbf{u})$, $\mathbf{u} = [u_i]_{i=1,\dots,5}$,$\mathbf{y} = [y_i]_{i=1,2}$ connected to two $y_1(\mathbf{u}) = f_1(\mathbf{u}) + f_2(\mathbf{u})$ and three wells $y_2(\mathbf{u}) = f_3(\mathbf{u}) + f_4(\mathbf{u}) + f_5(\mathbf{u})$ where $f_i(\mathbf{u}) = \sum_{j=0}^{4} s_{4-j} u_i^{4-j}$ describes the characteristics of the i-th well as a function of the amount of natural gas, u_i, injected into the well to facilitate oil extraction. The characteristics used in this work are shown in Fig. 4 (obtained from [3] using [12]). Both the inputs and the outputs are bounded $u_i \in [\underline{u}_i, \overline{u}_i]_{i=1,\dots,5}$, $y_i \in [\underline{y}_i, \overline{y}_i]_{i=1,2}$. The case study is put in the OFO framework (1) with $\Phi(\mathbf{u}, \mathbf{y}) = -y_1(\mathbf{u}) - y_2(\mathbf{u})$, $h(\mathbf{u}) = \begin{bmatrix} f_1(\mathbf{u}) + f_2(\mathbf{u}) & f_3(\mathbf{u}) + f_4(\mathbf{u}) + f_5(\mathbf{u}) \end{bmatrix}^\mathsf{T}$, and input constraint matrices:

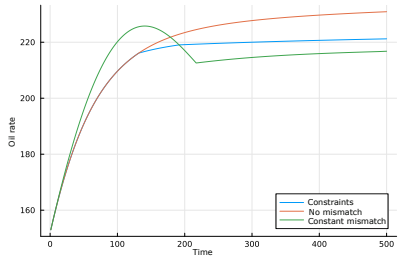
$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}, b = \begin{bmatrix} 9576 \\ -1157 \\ 11745 \\ -6819 \\ 5972 \\ -2714 \\ 7377 \\ -2399 \\ 9043 \\ -4125 \end{bmatrix}$$

The outputs are also bounded, $0 \leq y_i \leq 150$, $i = 1, 2$. The initial condition was set $u_0 = [2500.0, 7000.0, 4500.0, 4500.0, 4500.0]$ $\mathrm{Sm^3 day^{-1}}$.

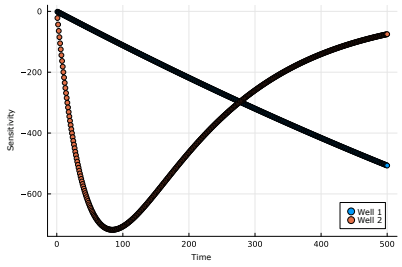
2) Sensitivity to gradient mismatch: We use the formulas from Section III to analyse the impact of the mismatch in the gradients ∇h: $\nabla h^{\text{used}}(\mathbf{u}^k, \mathbf{y}^k) = \nabla h(\mathbf{u}^k, \mathbf{y}^k) + \beta_k$ where $\beta_k = \begin{bmatrix} \beta_k^1 & \beta_k^2 & 0 & 0 & 0 \\ 0 & 0 & \beta_k^3 & \beta_k^4 & \beta_k^5 \end{bmatrix}$. The additive mismatch was chosen to emulate estimation error for every gradient separately. The code is available on github https://github.com/mzagorowska/Publications-/tree/ECC2025.



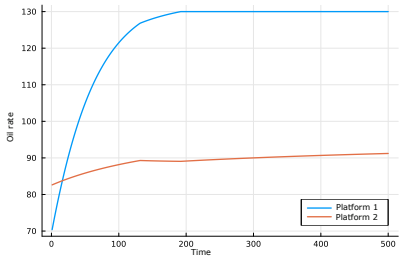
(a) Inputs u_i for the five wells, in $\text{Sm}^3\text{day}^{-1}$



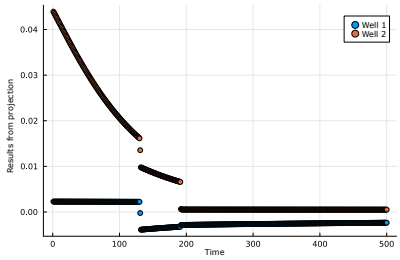
(b) The objective $-\Phi$ for the three cases, in $\text{Sm}^3\text{day}^{-1}$



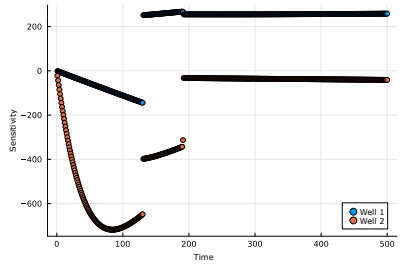
(c) Total sensitivity of Φ to the mismatch β^1 and β^2 (unconstrained)



(d) Outputs y_i of the two platforms (constrained), in $\text{Sm}^3\text{day}^{-1}$



(e) The solution to the projection over time (constrained)



(f) Total sensitivity of Φ to the mismatch β^1 and β^2 (Wells 1 and 2), (constrained)

Fig. 3: Online Feedback Optimization for gas lift optimization over time (in iterations)

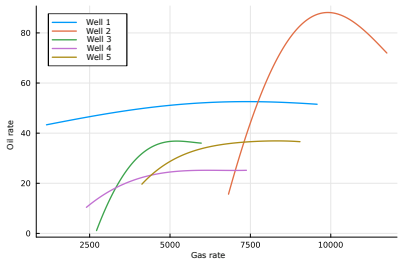


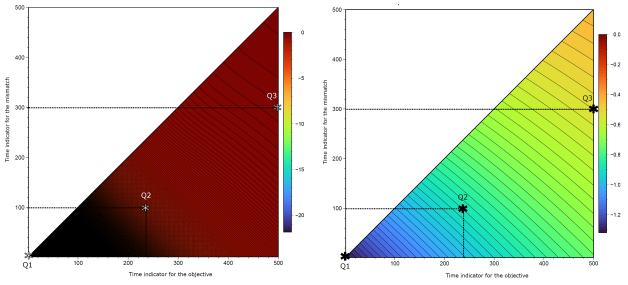
Fig. 4: Characteristics of the wells adapted from [3] (oil and gas rates in $\text{Sm}^3\text{day}^{-1}$)

a) Total sensitivity: The total sensitivities of the objective function (8) for the first platform at time k are shown in Fig. 3c. The sensitivities to the mismatch are primarily influenced by how far from their respective optima the inputs are (Fig. 3a and Fig. 4). The value of the objective is sensitive to β_k^2 in the initial period ($k < 200$). This sensitivity is due to the "steep" characteristics of the second well (orange in Fig. 4) around the initial guess ($u_2^0 = 7000\ \text{Sm}^3\text{h}^{-1}$). This is because the platform with output y_1 reached the maximum of the well number two, bringing the derivative $\frac{\partial y_1}{\partial u_2}$ close to zero in (6), and thus reducing the impact of the mismatch on the objective. Conversely, Well 1, 4 and 5 are operating away from their optima in Fig. 4, so the sensitivities are large.

b) Instantaneous sensitivity: Figure 5 show the instantaneous sensitivities of the objective with respect to β^2 and β^4, which were chosen as the most representative. In both cases, the largest absolute values are shown in dark blue (bottom left) and indicate that at the beginning of OFO, for small k, the objective changes under the influence of the mismatch. Conversely, for large k (right hand-side of the figure), the impact of the mismatch at a single time step j is small (yellow going into red). Such relationship between the objective function and the mismatch suggests that for large enough k OFO is insensitive to mismatch in a single time step. The transition from dark blue to yellow and red for large k also suggests that for large enough k OFO becomes less sensitive to the initial mismatch. This in turn implies that iterative learning algorithms may be used for gradient estimation as long as their accuracy improves with iterations.

The results in Fig. 5 allow us also to see the impact of the mismatch for individual wells. For instance, the mismatch affecting well 2 (Fig. 5a) has a significantly larger impact at the beginning (k close to zero, bottom left) than at the end of the chosen time horizon (k close to 500, right). This result indicates that the characteristics of well 2 should be accurate to capture the impact of well 2 on the objective. Conversely, the instantaneous sensitivity (7) of well 4 changes less from $k = 0$ to $k = 500$ (from left to right in Fig. 5b). At the same time, we see that the sensitivity of the objective for $k = 500$ depends on the timesteps, with a larger impact of mismatch in the initial time steps (light green, bottom right), getting close to zero with iterations (light orange, top right). Combining the transition from left to right and bottom to top suggests that while sensitivity to the mismatch in individual timesteps decreases, the final value is affected by mismatch in all past timesteps, in particular those far from the optimum.

c) Impact of coupling constraints: The matrices in Section IV-B.1 describe a case where the amount of gas available for injection to the platforms is unlimited, and the platforms have oversized capacity, which means that there is no coupling between any two wells. We now analyse the sensitivity to the mismatch if the amount of gas is limited



(a) Instantaneous sensitivity of Φ to the mismatch β in Well 2

(b) Instantaneous sensitivity of Φ to the mismatch β in Well 4

Fig. 5: Instantaneous sensitivity (7) with respect to the mismatch in Well 2 and 4, with three examples, Q1: $\partial\Phi^1/\partial\beta^i_1$, Q2: $\partial\Phi^{240}/\partial\beta^i_{100}$, Q3: $\partial\Phi^{500}/\partial\beta^i_{300}$, $i = 2, 4$, without coupling constraints

by a coupling constraint $\sum_{i=0}^{5} u_i \leq 26000$ Sm^{3}day^{-1}, and the first platform has a decreased capacity, $y_1 \leq 130$.

The results of OFO for the case with coupling constraints are shown in the bottom row of Fig. 3 and the corresponding sensitivities are in Fig. 3f. As long as no constraints are active, $k \leq 120$, the sensitivities in Fig. 3f are identical to the unconstrained case. When the coupling constraint on the amount of gas becomes active, we note discontinuity in time is the sensitivity. This discontinuity is caused by the impact of active constraints on the solutions of the projection QP (4), shown in Fig. 3e for the first platform. The QP is solved at discrete iterations, so there is no assumption about continuity in time. Therefore, as the sensitivity of the objective depends on the sensitivity of all solutions in individual timesteps, we see the discontinuities in Fig. 3f. A similar discontinuity in the sensitivities is observed at $k = 200$ when the output constraint in the first platform becomes active, which is also caused by a discontinuity in the solutions from 3e.

d) Impact of constant gradients: The sensitivities (8) can be used to estimate how the objective will change if the mismatch changes. We treat Φ as a function of β:

$$\Phi(\beta) - \Phi(0) \approx \sum_{i=1}^{5} \frac{\partial \Phi}{\partial \beta^i} \Delta\beta^i \tag{19}$$

For the gas lift case study, we chose $\overline{h} = \nabla h(u_0, y(u_0))$ because the approximation at the initial condition was shown to work in practice [10], and in consequence we got $\Delta\beta^i = [0, -0.04, -0.005, -0.001, -0.007]$, corresponding to the largest mismatch in absolute values. The maximal absolute value of the sensitivities is $[507, 718, 153, 287, 556]$. Plugging the values into (19) yields $\Phi(\beta) - \Phi(0) \geq -32.6$ which means that we can expect a decrease of up to 14% if the constant $\overline{h}$ is used. Because of taking the maximum values for the sensitivity and the mismatch, this is a conservative estimate, as also indicated in Fig. 3b where the actual decrease is 6.1%.

V. Discussion and Conclusions

The objective of the paper is to facilitate the analysis of Online Feedback Optimization controllers with respect to their parameters. The importance of analysing the impact of the parameters of an optimization-based controller has been shown and used for traditional controllers, such as Model Predictive Control, but the analysis for Online Feedback Optimization remains under-explored. This paper addresses this gap by providing closed-form expressions for the sensitivity of Online Feedback Optimization to its parameters.

The formulas in this paper were derived by setting a constant final time and treating online Feedback Optimization as a parametrised system of equations. This assumption is equivalent to discretizing a continuous gradient flow algorithm with a constant time step [6]. A natural extension to the sensitivity analysis will now be to consider the impact of time, both as time step and the final time. In the future, the sensitivities can be used for tuning of the parameters of the controller, as well as checking robustness to model mismatch.

References

[1] B. Amos, I. Jimenez, J. Sacks, B. Boots, and J. Z. Kolter. Differentiable MPC for end-to-end planning and control. *Advances in neural information processing systems*, 31, 2018.

[2] B. Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 136–145. PMLR, 06–11 Aug 2017.

[3] J. R. Andersen, L. Imsland, and A. Pavlov. Data-driven derivative-free trust-region model-based method for resource allocation problems. *Computers & Chemical Engineering*, 176:108282, August 2023.

[4] J. C. G. Boot. On sensitivity analysis in convex quadratic programming problems. *Operations Research*, 11(5):771–786, October 1963.

[5] M. Colombino, J. W. Simpson-Porco, and A. Bernstein. Towards robustness guarantees for feedback-based optimization. In *2019 IEEE 58th Conference on Decision and Control (CDC) Palais des Congrès et des Expositions Nice Acropolis Nice, France, December 11-13, 2019*. IEEE, 2019.

[6] V. Häberle, A. Hauswirth, L. Ortmann, S. Bolognani, and F. Dörfler. Non-convex feedback optimization with input and output constraints. *IEEE Control Systems Letters*, 5(1):343–348, 2020.

[7] A. Hauswirth, S. Bolognani, G. Hug, and F. Dörfler. Optimization algorithms as robust feedback controllers. *Annual Reviews in Control*, 57:100941, 2024.

[8] J. R. Magnus. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Wiley, February 2019.

[9] P. Mestres, A. Allibhoy, and J. Cortés. Regularity properties of optimization-based controllers. November 2023.

[10] L. Ortmann, F. Böhm, F. Klein-Helmkamp, A. Ulbig, S. Bolognani, and F. Dörfler. Tuning and testing an online feedback optimization controller to provide curative distribution grid flexibility. *Electric Power Systems Research*, 234:110660, 2024.

[11] L. Ortmann, A. Hauswirth, I. Caduff, F. Dörfler, and S. Bolognani. Experimental validation of feedback optimization in power distribution grids. *Electric Power Systems Research*, 189:106782, 2020.

[12] A. Rohatgi. Webplotdigitizer, v. 4.1, 2018. automeris.io/WebPlotDigitizer, accessed: 8 Nov 2024.

[13] S. Surjanovic and D. Bingham. Virtual library of simulation experiments: Test functions and datasets. Retrieved November 8, 2024, from https://www.sfu.ca/~ssurjano/stybtang.html.

[14] M. Thombre, Z. J. Yu, J. Jäschke, and L. T. Biegler. Sensitivity-assisted multistage nonlinear model predictive control: Robustness, stability and computational efficiency. *Computers & Chemical Engineering*, 148:107269, 2021.

[15] P.J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[16] M. Zagorowska, L. Ortmann, A. Rupenyan, M. Mercangöz, and L. Imsland. Tuning of online feedback optimization for setpoint tracking in centrifugal compressors. *IFAC-PapersOnLine*, 58(14):881–886, 2024.