Journal of Control, Automation and Electrical Systems (2019) 30:465–478
https://doi.org/10.1007/s40313-019-00468-9

Data-Driven Control Design by Prediction Error Identification for Multivariable Systems

Daniel D. Huff[1] · Luciola Campestrini[1] · Gustavo R. Gonçalves da Silva[1] · Alexandre S. Bazanella[1]

Received: 29 December 2018 / Revised: 14 April 2019 / Accepted: 16 April 2019 / Published online: 30 April 2019

Abstract
This paper deals with data-driven control design in a model reference framework for multivariable systems. Based on a single batch of input–output data collected from the process, a fixed structure controller is estimated without using a process model, by embedding the control design problem in the prediction error identification of an optimal controller. This is an extension of optimal controller identification (OCI) for multivariable systems. Even though the multiple-input multiple-output (MIMO) formulation is extended from its single-input single-output version in a natural way, the solution of the optimization problem is rather complex due to the special structure the inverse of the controller assumes in its MIMO version. Comparisons between the OCI and the virtual reference feedback tuning—a well-known data-driven control method—are provided, showing the efficiency of the OCI controller estimate. We also explore the case where the batch of design data is collected in closed loop. Simulated and experimental results show the efficiency of the proposed methodology.

Keywords Data-driven control · Multivariable systems · OCI · System identification

1 Introduction

Tight constraints on performance due to energy saving and quality standards make impossible to neglect interactions among process variables in several control systems. For example, heat exchangers, distillation columns and chemical reactors will only provide an adequate outcome if the control of their variables takes into account the presence of disturbances in the process due to other variables, which makes the control design a difficult task. There are several methodologies to design a multiple-input multiple-output (MIMO) controller, most of them based on a mathematical model of the process, which can be obtained from an identification procedure.

✉ Daniel D. Huff
daniel.huff@ufrgs.br

Luciola Campestrini
luciola@ufrgs.br

Gustavo R. Gonçalves da Silva
gustavo.rgs@ufrgs.br

Alexandre S. Bazanella
bazanella@ufrgs.br

[1] Department of Automation and Energy, Federal University of Rio Grande do Sul, Porto Alegre 90035-190, Brazil

However, as it happens to MIMO control, MIMO identification is definitely not an easy task, usually involving time and money consumption, and a simple controller is commonly dependent on a simple process model. An alternative to that is MIMO data-driven control (Bazanella et al. 2012), an approach that estimates a controller without using a model for the plant. Thus, a fixed structure controller can be designed directly from data, without deriving a process model either through the identification of such model or by a simplification of a complex and/or nonlinear model. In so doing, the drawbacks of system identification and model reduction are avoided.

There are several data-driven methods developed for single-input single-output (SISO) control problems in the literature. However, since SISO methods are not tailored to be used when interactions between variables are significant, some effort has been put in developing the extensions of these methods for the MIMO case: some are iterative (Jansson and Hjalmarsson 2004; Miskovic et al. 2005) whereas others are one-shot (Formentin et al. 2012; Campestrini et al. 2016), based only on one experiment (or sometimes two when the collected data are corrupted by noise). Although some methods, like virtual reference feedback tuning (VRFT) (Campi et al. 2002; Campestrini et al. 2016), are extended to multivariable systems in a rather natural way, it is often the case

a higher number of experiments on the plant is needed. In Yubai et al. (2009), for instance, the proposed method, even though being considered one-shot, requires n_u batches of data, where n_u is the dimension of the plant input. Another example is found in Hjalmarsson (1999), where $n_y n_u + 1$ experiments per iteration are necessary, with n_y being the number of outputs.

Among the one-shot methods, both correlation-based tuning (CbT) (Karimi et al. 2007; Yubai et al. 2009) and VRFT use instrumental variables (IV) to obtain unbiased controller estimates when signals are corrupted by noise, and it is often the case that these instruments are constructed based on a second experiment on the plant repeating the input signal (Campi et al. 2002), which may be a difficulty under process operation. Besides, it is well known that IV estimates, although being consistent, are significantly less efficient than prediction error ones (Söderström 2007), which may result in even unstable closed loops when the collected data present a low signal-to-noise ratio (SNR) (Boeira and Eckhard 2018; Rallo et al. 2016). The OCI method on the other hand uses only one batch of input–output data to perform an identification of the controller through the prediction error approach, resulting in an unbiased estimate without the need of instrumental variables (Campestrini et al. 2017). The main idea of OCI methodology is to write the plant as a function of the user-defined reference model and the controller to be identified and then perform the identification of the controller and the noise model.

We present in this paper a MIMO version of the OCI method (Campestrini et al. 2017). The extension to the multivariable case results in an optimization problem that cannot be dealt with as in the SISO case, due to the structure of the MIMO controller inverse, which has to be identified. Commercial tools (like `ident` from Matlab) cannot be applied in this case, and a dedicated optimization solution is necessary in order to obtain the optimal controller. The developed algorithm is presented in the paper.

With only one batch of open-loop data collected on the process, an unbiased estimate of the controller is obtained if the chosen controller class is of full order. The case where data are collected in closed loop is also explored in the paper, which is of high interest in industrial applications. In this case, we perform the identification of the controller and of a noise model for the process in order to obtain a consistent estimate of the controller parameters. We also compare our methodology with the instrumental variable method when data are highly corrupted by noise. Simulated and experimental results show the OCI estimate is more efficient than the IV one.

The paper is organized as follows. Section 2 presents definitions and problem formulation. The OCI MIMO formulation is presented in Sect. 3. Section 4 shows some illustrative simulated examples, where the statistical properties of the methodology concerning open-loop and closed-loop experiments are presented and compared to IV estimates. Section 5 presents experimental results concerning a level plant, and Conclusions are presented at the end of paper.

2 Preliminaries

Consider a linear time-invariant discrete-time MIMO process

$$y(t) = G_0(q)u(t) + H_0(q)w(t), \tag{1}$$

where q is the forward-shift operator, $u(t)$ and $y(t)$ are n-vectors representing the process' input and output, respectively, and $w(t)$ is a sequence of independent random n-dimensional vectors with zero mean values, covariance matrix $E[w(t)w^T(t)] = \Lambda$ and bounded fourth moments. The transfer matrix $G_0(q)$ and the noise model $H_0(q)$ are square $n \times n$ matrices whose elements are proper rational transfer functions and $H_0(\infty) = I$.

The design task is to tune the parameter vector $P \in \mathbb{R}^{n_P}$ of a linear time-invariant controller $C(q, P)$ in order to achieve a desired closed-loop response. We assume that this controller belongs to a given user-specified controller class $\mathcal{C}$ such that all elements of the loop transfer matrix $L(q, P) = G_0(q)C(q, P)$ have positive relative degree for all $C(q, P) \in \mathcal{C}$. The control action $u(t)$ can be written as

$$u(t) = C(q, P)e(t) = C(q, P)(r(t) - y(t)), \tag{2}$$

where $r(t)$ is the reference signal, which is assumed to be quasi-stationary and uncorrelated with the noise $w(t)$, that is, $\bar{E}\left[r(t)w^T(t-\tau)\right] = 0\ \forall\tau$, where

$$\bar{E}[f(t)] \triangleq \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} E[f(t)]$$

with $E[\cdot]$ denoting expectation (Ljung 1999). The system (1)–(2) in closed loop becomes

$$y(t, P) = T(q, P)r(t) + [I - T(q, P)]H_0(q)w(t), \tag{3}$$

$$T(q, P) = [I + G_0(q)C(q, P)]^{-1}G_0(q)C(q, P). \tag{4}$$

The controller class $\mathcal{C}$ is defined as

$$\mathcal{C} = \left\{C(q, P) : P \in \mathcal{D}_P \subseteq \mathbb{R}^{n_P}\right\},$$

where $\mathcal{D}_P$ is a set of admissible parameters and $C(q, P)$ is invertible for all $P \in \mathcal{D}_P$. The structure of the controller to be designed is defined as

$$C(q,P)=\begin{bmatrix} C_{11}(q,\rho_{11}) & C_{12}(q,\rho_{12}) & \dots & C_{1n}(q,\rho_{1n}) \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1}(q,\rho_{n1}) & C_{n2}(q,\rho_{n2}) & \dots & C_{nn}(q,\rho_{nn}) \end{bmatrix} \tag{5}$$

where $P=[\rho_{11}^T \;\; \rho_{12}^T \;\; \dots \;\; \rho_{n1}^T \;\; \dots \;\; \rho_{nn}^T]^T$. A particularly relevant class, which will be used in the case studies to be presented in this paper, is that of proportional–integral–derivative (PID) controllers. In such PID controllers, each element of the controller matrix $C(q,P)$ in (5) has the following parametrized structure when the derivative pole is fixed at zero:

$$C_{ij}(q,\rho_{ij})=\frac{a_{ij}q^2+b_{ij}q+c_{ij}}{q(q-1)} \tag{6}$$

where $\rho_{ij}=[a_{ij} \;\; b_{ij} \;\; c_{ij}]^T$.

In the Model Reference approach to the design, the closed-loop performance is specified through the desired closed-loop transfer matrix $T_d(q)$, also known as the reference model. The controller parameters are then tuned as the solution of the problem

$$P^{\mathrm{MR}}=\arg\min_P \; J^{\mathrm{MR}}(P), \tag{7}$$

$$J^{\mathrm{MR}}(P)\triangleq \bar{E}||(T_d(q)-T(q,P))r(t)||_2^2 \tag{8}$$

where $r(t)$ is the reference signal of interest.

The *ideal controller* $C_d(q)$ is the one that allows the closed-loop system behavior to match exactly the one prescribed by $T_d(q)$ and is given by

$$C_d(q)=G_0(q)^{-1}T_d(q)[I-T_d(q)]^{-1}. \tag{9}$$

If (9) was used in the closed loop, then the objective function (7) would evaluate to zero. For our further analysis, we will sometimes consider the situation where $C_d(q)\in\mathcal{C}$, in which case we shall say that the following assumption holds.

Assumption 1 Matching condition of the controller

$$\exists\, P_d\in\mathcal{D}_P \;\text{ such that }\; C(q,P_d)=C_d(q).$$

However, this ideal controller may not correspond to any controller in the controller set $\mathcal{C}$; actually in most practical applications, it will not belong to $\mathcal{C}$. In this case, we would like to estimate a controller that resembles the ideal controller, making the closed-loop response as close as possible to the desired $T_d(q)$.

Notice that (8) depends on the process model $G_0(q)$. Data-driven control methods aim to minimize $J^{\mathrm{MR}}(P)$ without using the model of the plant. As a result, these methods minimize other cost functions that, under some ideal conditions, present the same minimum as the model based $J^{\mathrm{MR}}(P)$. VRFT and CbT are one-shot data-driven methods which solve quadratic functions and therefore are very appealing for computational aspects. However, when the collected data are corrupted with noise, unbiased controller parameters are estimated by both methods only with the use of instrumental variables (IV), which are known to provide estimates with large variance compared to the prediction error approach (Söderström and Stoica 1989).

In the sequel, we present a data-driven method that solves the reference model control problem for MIMO systems using only one batch of input–output data, even when using noisy signals. It differs from other one-shot methods (VRFT and CbT) by how it deals with the noise. Since it solves a prediction error (PE) identification problem, an unbiased estimate of the controller parameters can be obtained without using an IV, and the estimates present lower variance.

3 Optimal Controller Identification

Using the concept of the ideal controller, it is possible to turn the model reference control design problem into an identification problem of the controller, without using a model for the process. This data-driven design method was presented in Campestrini et al. (2017) for SISO systems and introduced in Huff et al. (2018), where it was applied to a benchmark control problem. However, the benchmark system was not corrupted by noise, and in this paper, we explore the properties of the method to deal with noisy signals.

The core idea of the OCI method is to rewrite the input–output system (1) in terms of the ideal controller $C_d(q)$, which is done by inverting the relation (9), i.e.,

$$G_0(q)=T_d(q)\,(I-T_d(q))^{-1}\,C_d^{-1}(q). \tag{10}$$

Then, a model for the plant can be written in terms of the controller parameters as

$$G(q,P)\triangleq T_d(q)\,(I-T_d(q))^{-1}\,C^{-1}(q,P), \tag{11}$$

where

$$G(q,P)=G_0(q)\Leftrightarrow C(q,P)=C_d(q). \tag{12}$$

The task will then be to identify an estimate $C(q,\hat{P})$ of the ideal controller $C_d(q)$ within the parametrized controller class defined by $\mathcal{C}$. In other words, this corresponds to an identification of a plant model $G(q,P)$ with a fixed part, which is a function of the reference model $T_d(q)$, and a parametrized

part, which is a function of the controller inverse. Thus, (1) can be rewritten as

$$y(t,\Theta) = G(q,P)u(t) + H(q,\Theta)w(t) \tag{13}$$

where $\theta \in \mathcal{D}_\theta \subseteq \mathbb{R}^{n_\theta}$ is an additional parameter vector that appears in the noise model and $\Theta = \begin{bmatrix} P^T & \theta^T \end{bmatrix}^T \in \mathcal{D}_\Theta = \mathcal{D}_P \times \mathcal{D}_\theta$.

From N measured input–output data, the parameter vector estimate $\hat{\Theta}_N = \begin{bmatrix} \hat{P}_N^T & \hat{\theta}_N^T \end{bmatrix}^T$ is defined as (Campestrini et al. 2017):

$$\hat{\Theta}_N = \underset{\Theta \in \mathcal{D}_\Theta}{\arg\min}\, V(\Theta) \tag{14}$$

where

$$V(\Theta) \triangleq \frac{1}{N}\sum_{t=1}^{N} \|\epsilon(t,\Theta)\|_2^2, \tag{15}$$

$\epsilon(t,\Theta)$ is the prediction error

$$\epsilon(t,\Theta) \triangleq y(t) - \hat{y}(t|t-1,\Theta) \tag{16}$$

and

$$\begin{aligned}\hat{y}(t|t-1,\Theta) &= H^{-1}(q,\Theta)T_d(q)\left(I - T_d(q)\right)^{-1} C^{-1}(q,P)u(t) \\ &\quad + \left[I - H^{-1}(q,\Theta)\right] y(t)\end{aligned} \tag{17}$$

is the one-step-ahead predictor for model (13), where $G(q,P)$ has been replaced by (11). The predictor is now a function of the noise model $H(q,\Theta)$ and the inverse of the controller $C^{-1}(q,P)$.

Instead of minimizing $J^{\mathrm{MR}}(P)$, which depends on the unknown plant $G_0(q)$, now the design can be made by minimizing the cost function $V(\Theta)$, which is purely data dependent and no model of the plant $G_0(q)$ is used. Since the estimation of the optimal MR controller has been transformed into a PE identification problem, all statistical properties of PE identification theory apply (Ljung 1999; Söderström and Stoica 1989).

Specifically, if the model structure satisfies some mild conditions, the estimate in (14) converges with probability one (w.p. 1) as $N \to \infty$ to the vector $\Theta^* = \begin{bmatrix} P^{*T} & \theta^{*T} \end{bmatrix}^T$ defined as follows:

$$\hat{\Theta}_N \to \Theta^* = \underset{\Theta \in \mathcal{D}_\Theta}{\arg\min}\, \bar{V}(\Theta) \tag{18}$$

where

$$\bar{V}(\Theta) = \bar{E}\,\|\epsilon(t,\Theta)\|_2^2. \tag{19}$$

Taking into account (18)–(19) and (12), two consistency results follow. The first one concerns identification in open loop.

Lemma 1 (Söderström and Stoica 1989) *If Assumption 1 is satisfied, an informative enough data set is collected in open loop and $C(q,P)$ and $H(q,\Theta)$ are parametrized independently (that is, $\frac{\partial H(q,\Theta)}{\partial P} = 0$) then, for $N \to \infty$:*

$$C(q,\hat{P}_N) \to C_d(q) \;\; w.p.\ 1 \tag{20}$$

If data are not necessarily collected in open loop, we have the following result.

Lemma 2 (Söderström and Stoica 1989) *If an informative enough data set is collected in open or closed loop, then (20) holds provided that $\exists \Theta_d = \begin{bmatrix} P_d^T & \theta_d^T \end{bmatrix}^T \in \mathcal{D}_\Theta$ such that $C(q,P_d) = C_d(q)$ and $H(q,\Theta_d) = H_0(q)$. We also have for $N \to \infty$:*

$$H(q,\hat{\Theta}_N) \to H_0(q) \;\; w.p.\ 1. \tag{21}$$

Notice that $\frac{\partial H(q,\Theta)}{\partial P} = 0$ is not necessary in this case. Considering Lemma 2, let us define the following assumption, which is an extension of Assumption 1:

Assumption 2 Matching condition of the controller and the noise model

$$\begin{aligned}&\exists\, \Theta_d = [P_d^T \;\; \theta_d^T]^T \in \mathcal{D}_\Theta \;\text{ such that} \\ &C(q,P_d) = C_d(q) \;\text{ and }\; H(q,\Theta_d) = H_0(q)\end{aligned}$$

When neither one of those sets of conditions is satisfied (related to Lemmas 1 and 2), the controller parameters related to the minima of $\bar{V}(\Theta)$ and $J^{\mathrm{MR}}(P)$ are distinct. So, let us define the bias and the variance errors of the controller estimate as follows:

$$\begin{aligned}C_d(q) - C(q,\hat{P}_N) &= \underbrace{C_d(q) - C(q,P^*)}_{BIAS} \\ &\quad + \underbrace{C(q,P^*) - C(q,\hat{P}_N)}_{VARIANCE}\end{aligned} \tag{22}$$

where the bias error is zero when the ideal controller belongs to the chosen controller class and the other conditions above are satisfied. These metrics are going to be used in the examples in order to study the statistical properties of the proposed methodology compared to the IV solution.

When choosing the controller structure, it is often the case that one imposes some fixed part in the controller, the most common instance of this fact probably being the imposition of a pole at $q = 1$ to guarantee zero steady-state error for constant reference tracking and disturbance rejection. This

fixed part does not need to be identified. So, we call $C_F(q)$ this fixed part and rewrite the controller transfer function as

$$C(q, P) = C_I(q, P)C_F(q). \tag{23}$$

For instance, consider a 2×2 PI controller. Then,

$$C_F(q) = \frac{1}{q-1} I \tag{24}$$

$$C_I(q, P) = \begin{bmatrix} a_{11}q + b_{11} & a_{12}q + b_{12} \\ a_{21}q + b_{21} & a_{22}q + b_{22} \end{bmatrix} \tag{25}$$

where $P = [a_{11}\, b_{11}\, a_{12}\, \ldots\, a_{21}\, \ldots\, a_{22}\, b_{22}]^T$. In this example, $C_F(q)$ is just a scalar times the identity matrix, but it does not actually need to be this way.

However, it is important to mention that (16), which appears in the cost function, is, by nature, nonlinear in Θ even if the controller is linear in P. That is, there is no reason to restrict ourselves to linearly parametrized controllers from an optimization viewpoint (as it happens in methods that solve least squares problems) and we can work with quite general controller structures. In Huff et al. (2018), for instance, the derivative pole of a PID controller structure is let free to be identified, resulting in a closed loop closer to the desired one compared to a classical PID structure.

Using (23) and (11), (13) can be written as

$$\begin{aligned} y(t, \Theta) = & \underbrace{T_d(q)\,(I - T_d(q))^{-1}\, C_F^{-1}(q)}_{F(q)} \underbrace{C_I^{-1}(q, P)}_{\tilde{C}(q,P)} u(t) \\ & + H(q, \Theta)w(t) \end{aligned} \tag{26}$$

where $F(q)$ is a fixed transfer matrix formed by the fixed part of $G(q, P)$. It is assumed that the reference model $T_d(q)$ and the controller structure are chosen in such a way that $F(q)$ and $F(q)\tilde{C}(q, P)$ are causal. Notice that in the SISO case (and also in the MIMO case when both $T_d(q)$ and $C_F(q)$ can be written as scalars times the identity matrix), $F(q)$ commutes with $\tilde{C}(q, P)$ and (26) can be written as

$$\begin{aligned} y(t, \Theta) = & \underbrace{C_I^{-1}(q, P)}_{\tilde{C}(q,P)} \times \underbrace{T_d(q)\,(I - T_d(q))^{-1}\, C_F^{-1}(q) u(t)}_{\tilde{u}(t)} \\ & + H(q, \Theta)w(t) \\ & \triangleq \tilde{C}(q, P)\tilde{u}(t) + H(q, \Theta)w(t). \end{aligned} \tag{27}$$

In the SISO case, the solution for the identification problem can be easily obtained through available toolboxes like Matlab® `ident` (Ljung 1991) because of the special form assumed by (27). We show that in the next simple example.

Example 1 Suppose the user chooses the following reference model and controller class, where the controller is already written as $C_F(q)$ and $C_I(q)$:

$$T_d(q) = \frac{0.16q}{(q-0.6)^2},$$

$$C_F(q) = \frac{1}{q-1}, \quad C_I(q, P) = \frac{q^3 + aq^2 + bq + c}{dq^2 + eq + f}.$$

The transfer function to be identified is the inverse of $C_I(q, P)$, given by

$$\tilde{C}(q, P) = \frac{dq^2 + eq + f}{q^3 + aq^2 + bq + c}$$

using

$$\tilde{u}(t) = \frac{0.16q}{q - 0.36} u(t)$$

as input signal, according to (27). Notice that $H(q, \Theta)$ can also be identified to enhance the controller estimate properties, even though one may not be interested in the model $H(q, \hat{\Theta}_N)$ *per se*. □

However, filters $F(q)$ and $\tilde{C}(q, P)$ in (26) do not usually commute in the MIMO case. So, it is not possible to rewrite (26) as in (27). And even if it were possible, the matrix $\tilde{C}(q, P)$ possesses a rather unusual structure, as shown in the next example.

Example 2 Consider a noise-free system given by

$$G_0(q) = \begin{bmatrix} \frac{0.095q}{(q-0.92)(q-0.8)} & \frac{0.04q}{(q-0.9)(q-0.85)} \\ \frac{-0.03q}{(q-0.92)(q-0.8)} & \frac{0.05q}{(q-0.9)(q-0.85)} \end{bmatrix} \tag{28}$$

where $C(q, P)$ is given by (24)–(25) and the chosen reference model is

$$T_d(q) = \frac{0.2}{q - 0.8} I \tag{29}$$

that is, we are specifying a decoupled closed-loop system with the same performance for both outputs. In this case,

$$\tilde{C}(q, P) = \frac{1}{\det\,(C_I(q, P))} \mathrm{cof}^T\,(C_I(q, P)) \tag{30}$$

where $\det(\cdot)$ is the determinant and $\mathrm{cof}(\cdot)$ is the cofactor matrix. The transpose of the cofactor matrix is

$$\mathrm{cof}^T\,(C_I(q, P)) = \begin{bmatrix} a_{22}q + b_{22} & -(a_{12}q + b_{12}) \\ -(a_{21}q + b_{21}) & a_{11}q + b_{11} \end{bmatrix} \tag{31}$$

and the determinant is given by

$$\det\left(C_I(q,P)\right) = (a_{11}q + b_{11})(a_{22}q + b_{22}) - (a_{12}q + b_{12})(a_{21}q + b_{21}) \tag{32}$$

Notice that even when the controller is linearly parametrized, its inverse is not. We can see the controller inverse $\tilde{C}(q,P)$ as a MIMO gray-box system to be identified.

In this proposed example, $F(q)$ and $\tilde{C}(q,P)$ in (26) do commute. However, it is still not possible to use, for instance, the `ident` toolbox, because of the special form assumed by $\tilde{C}(q,P)$. Let us see, then, what happens if we attempt to re-parametrize $\tilde{C}(q,P)$ as shown below:

$$\begin{aligned}\tilde{C}(q,\bar{P}) &= \begin{bmatrix} q^2+d_1q+d_2 & 0 \\ 0 & q^2+d_3q+d_4 \end{bmatrix}^{-1} \\ &\quad\times \begin{bmatrix} c_1q+c_2 & c_3q+c_4 \\ c_5q+c_6 & c_7q+c_8 \end{bmatrix} \\ &= \begin{bmatrix} \dfrac{c_1q+c_2}{q^2+d_1q+d_2} & \dfrac{c_3q+c_4}{q^2+d_1q+d_2} \\ \dfrac{c_5q+c_6}{q^2+d_3q+d_4} & \dfrac{c_7q+c_8}{q^2+d_3q+d_4} \end{bmatrix}\end{aligned} \tag{33}$$

where $\bar{P} = [c_1 \ \ldots \ c_8 \, d_1 \ \ldots \ d_4]^T$. This model structure is standard (compare with the diagonal form presented in Söderström and Stoica (1989), for instance), being covered by `ident`. However, it is also more flexible than (30) for two reasons. First of all, the coefficients that appear in the denominators of the elements of (33) should actually be related to the numerators' coefficients. Besides that, *all* elements of (33) should have the same denominator (independently of the corresponding row).

If we apply a sequence of steps as inputs to the open-loop system and perform the identification of $\tilde{C}(q,\bar{P})$ through `ident`, we obtain

$$\tilde{C}(q,\hat{\bar{P}}_N) = \begin{bmatrix} \frac{0.461(q+0.008)}{(q-0.810)(q-0.917)} & \frac{0.262(q-0.198)}{(q-0.810)(q-0.917)} \\ \frac{-0.118(q+0.208)}{(q-0.837)(q-0.907)} & \frac{0.262(q-0.031)}{(q-0.837)(q-0.907)} \end{bmatrix} \tag{34}$$

resulting in the following controller:

$$C(q,\hat{\bar{P}}_N) = \begin{bmatrix} \frac{1.727(q-0.917)(q-0.810)(q-0.031)}{(q+0.085)(q-0.101)(q-1)} & \frac{-1.729(q-0.907)(q-0.837)(q-0.198)}{(q+0.085)(q-0.101)(q-1)} \\ \frac{0.778(q-0.917)(q-0.810)(q+0.208)}{(q+0.085)(q-0.101)(q-1)} & \frac{3.045(q-0.907)(q-0.837)(q+0.008)}{(q+0.085)(q-0.101)(q-1)} \end{bmatrix} \tag{35}$$

which is not a PI controller as we would like, because we solved the design problem using more degrees of freedom than we actually have, as explained above.

Now, if we identify the structure (30) by our proposed methodology (using the optimization algorithm presented in the sequel), we get

$$\tilde{C}(q,\hat{P}_N) = \begin{bmatrix} \frac{1.751}{q-0.941} & \frac{0.801}{q-0.939} \\ \frac{-0.543}{q-0.941} & \frac{1.037}{q-0.939} \end{bmatrix} \tag{36}$$

[1] resulting in:

$$C(q,\hat{P}_N) = \begin{bmatrix} \frac{0.461(q-0.942)}{q-1} & \frac{-0.356(q-0.941)}{q-1} \\ \frac{0.241(q-0.938)}{q-1} & \frac{0.778(q-0.939)}{q-1} \end{bmatrix} \tag{37}$$

which is, in fact, a PI controller.

An important remark is that costs (8) and (15) related to (35) are actually smaller than the ones corresponding to (37), at the price of using a more complex controller structure, which is not always possible/suitable in a real-world application. □

So, due to the particular and unfamiliar parametrization of model (26), it is not in general possible to employ any standard identification toolbox in the MIMO case and a dedicated optimization solution must be used in order to minimize (15). In this work, we propose to apply the steepest descent and the Levenberg–Marquardt methods. These methods require an initial parameter vector P_0 of the controller, and for this purpose, we use a MIMO version of VRFT (Campestrini et al. 2016). When the noise model $H(q,\Theta)$ is also identified, we consider $\frac{\partial H(q,\Theta)}{\partial P} = 0$ and use as initial condition $H(q,\theta_0) = I$.

More specifically, we employ first the steepest descent method because of its large region of attraction to a local minimum. In this case, the recursion formula is given by Bazanella et al. (2012)

$$\Theta_{k+1} = \Theta_k - \gamma_k \nabla V(\Theta_k), \tag{38}$$

where for each iteration k, γ_k is a positive scalar and $\nabla V(\Theta_k)$ is the gradient of the cost function $V(\cdot)$ evaluated at the parameter vector Θ_k. Quantity γ_k is increased by 1% if $V(\Theta_{k+1}) < V(\Theta_k)$. Otherwise, it is decreased by 1% and one has $\Theta_{k+1} = \Theta_k$. As stopping criterion, we verify at each iteration if the relative decrease in the gradient is less than 1%.

In the second step, the Levenberg–Marquardt method (Fletcher 1987), whose convergence is faster, is applied. Its search direction is a cross between the Gauss–Newton and the steepest descent directions, and is given by

[1] Notice that the denominators of (36) are all equal *before* we carry out the cancellations of the common factors between the numerators and denominators (considering three significant digits).

$$\Theta_{k+1} = \Theta_k - \left(\nabla^2 V(\Theta_k) + \lambda_k I\right)^{-1} \nabla V(\Theta_k), \tag{39}$$

where $\nabla^2 V(\Theta_k)$ is an *approximation* of the Hessian of $V(\Theta)$ at the point Θ_k. When λ_k is zero, the search direction corresponds to that of Gauss–Newton method. As $\lambda_k \to \infty$, the direction tends toward the steepest descent one, and the magnitude of the step tends to zero. Then, for some sufficiently large λ_k, the relation $V(\Theta_{k+1}) < V(\Theta_k)$ holds true. The term λ_k can therefore be controlled to ensure (a fast) descent. In this work, we set $\lambda_{k+1} = \lambda_k/10$ when the iteration is successful (lower function value found). Otherwise, the algorithm sets $\lambda_{k+1} = 10\lambda_k$ and $\Theta_{k+1} = \Theta_k$. The stopping criterion is equal to the steepest descent one but with tolerance 0.01%.

In general, the minimization of the cost function requires approximately 2000 iterations of the steepest descent method and 50–100 iterations of the Levenberg–Marquardt one. The algorithm takes about 5 s to run on a computer with a processor Intel(R) Core(TM) i7 and 12 GB of RAM. Moreover, in order to calculate the derivatives of $V(\Theta)$, we employ in our code some functions from the `symbolic math` toolbox of Matlab®.

We now present numerical and experimental results to show the efficiency of the OCI methodology when the collected data are corrupted with noise.

4 Numerical Examples

Consider a plant whose model is given by (28) with respective noise model given by

$$H_0(q) = \begin{bmatrix} \frac{(q-0.4)q}{(q-0.8)(q-0.9)} & \frac{0.5(q-0.2)}{(q-0.8)(q-0.9)} \\ \frac{0.5(q-0.5)}{(q-0.8)(q-0.9)} & \frac{(q-0.6)q}{(q-0.8)(q-0.9)} \end{bmatrix}. \tag{40}$$

The noise sequence $w(t)$ in (1) has covariance matrix $\Lambda = 0.002I$, and the sampling time is $T_s = 1$ s.

We choose a reference model given by

$$T_d(q) = \begin{bmatrix} \frac{0.2}{q-0.8} & 0 \\ 0 & \frac{0.4}{q-0.6} \end{bmatrix}, \tag{41}$$

which specifies a decoupled closed-loop system, with zero steady-state error for step references and with loop 2 faster than loop 1.

In this case, the ideal controller (9) is given by

$$C_d(q) = \begin{bmatrix} \frac{1.6807(q-0.8)(q-0.92)}{q(q-1)} & \frac{-2.6891(q-0.8)(q-0.92)}{q(q-1)} \\ \frac{1.0084(q-0.85)(q-0.9)}{q(q-1)} & \frac{6.3866(q-0.85)(q-0.9)}{q(q-1)} \end{bmatrix}, \tag{42}$$

which will be used only for comparison purposes with the obtained controllers through our data-driven approach. Notice that the desired closed-loop behavior can be achieved by means of a centralized PID controller, so we can analyze the cases where the ideal controller belongs to the chosen class and where it does not, i.e., whether Assumption 1 is satisfied or not.

In all examples, results will be evaluated through the estimation of the Model Reference cost (8) using collected noisy data as

$$\hat{J}^{\mathrm{MR}}(P) \triangleq \frac{1}{N}\sum_{t=1}^{N} ||T_d(q)r(t) - y(t, P)||_2^2. \tag{43}$$

Besides, an error measure, denoted by E_c, will be computed for each controller as

$$E_c(P) \triangleq \left\| \frac{q-1}{q}\left[C(q, P) - C_d(q)\right] \right\|_2 \tag{44}$$

where $\|\cdot\|_2$ denotes the H_2 norm of the transfer function. We have removed the unstable modes from the controller in order to obtain a finite value.

We first present a comparison between the estimates of a MIMO controller obtained through the proposed prediction error approach and an instrumental variable one. The second numerical result explores closed-loop data and the estimation of the noise model.

4.1 Prediction Error Versus Instrumental Variable Approaches

We will compare the OCI with the VRFT method when the latter one uses instrumental variables (IVs) in order to deal with noisy data when estimating the controller parameters. As suggested in Campestrini et al. (2016), the IV is obtained from an additional experiment on the plant using the same input signal and corresponds to the resulting output signal, which depends on a different realization of the process noise. The procedure can be described as follows. Perform one experiment on the plant; with data from this single experiment, use the standard VRFT to tune an initial (biased) controller estimate. Using the same batch of data and the initial controller, tune a controller using our OCI methodology. Perform a second experiment on the plant; with data from the first and the second experiment, tune a new controller using the VRFT with IV. Compare the obtained controllers via OCI and VRFT-IV.

The system (28)–(40) is excited in open loop with a square wave of amplitude $A = 0.7$ and period of 200 samples. Figure 1 shows the outputs, which are affected by filtered white noise, and the applied inputs. The signal-to-noise ratio (SNR) is around 15 dB in both channels.

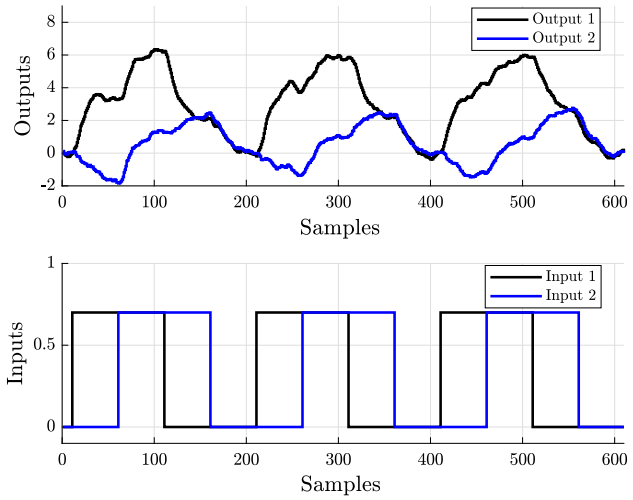

Fig. 1 Open-loop responses of system (28)–(40) to a square wave

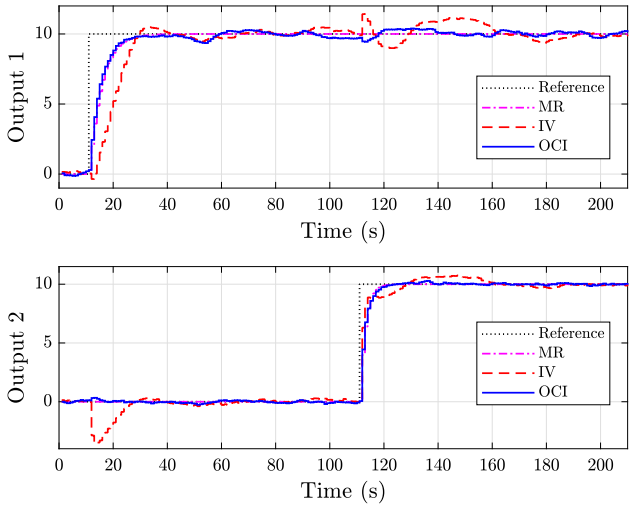

Fig. 2 Closed-loop responses of system (28)–(40) with controllers (46) and (47)

Table 1 Medians of $\hat{J}^{\mathrm{MR}}$ and E_c obtained for an open-loop batch of data when $C_d(q) \in \mathcal{C}$

Controller	$\hat{J}^{\mathrm{MR}}$	E_c
$C(q, \hat{P}_{\mathrm{IV}})$	2.0025	15.9981
$C(q, \hat{P}_{\mathrm{OCI}})$	0.07971	4.1200

Recall that if data are collected in open loop, then there is no need to identify $H(q, \Theta)$ when applying the OCI in order to obtain a consistent estimate of the controller. Thus, in the following, we consider $H(q, \Theta) = I$ and use this batch of data to tune the parameters of different controller structures.

4.1.1 The Ideal Case: $C_d(q) \in \mathcal{C}$

In order to satisfy Assumption 1, a centralized PID controller is designed. To analyze the estimate properties, two hundred Monte Carlo runs considering different realizations of the batch of open-loop data were performed and one hundred different controllers were designed using both the OCI and IV methods.[2] In order to compute Model Reference cost (43), we performed a closed-loop experiment for each obtained controller where the reference is a sequence of steps of amplitude 10, so the closed-loop output has a SNR of approximately 25 dB, which is higher than the open-loop one in order to facilitate the comparison of the results. For each case, we also computed the corresponding error measure (44).

The medians of $\hat{J}^{\mathrm{MR}}$ as well as of E_c obtained from the Monte Carlo runs are shown in Table 1. Notice that the values related to the IV approach are considerably greater than the corresponding OCI ones. In fact, several designed IV controllers (around 35%) do not even stabilize the plant in closed loop. That is, even though both controller estimates do not present bias error in this case [recall (22)], OCI provides a smaller variance error. At least from a qualitative viewpoint, this observation makes sense, since the prediction error estimate $\hat{\Theta}_N$ is (under some ideal conditions) asymptotically statistically efficient (Söderström and Stoica 1989).

Next, we consider a single run (out of 100) of the simulations above. The initial controller used by the OCI method obtained using the standard VRFT method is

$$C(q, \hat{P}_0) = \begin{bmatrix} \frac{0.441(q-0.942)(q-0.210)}{q(q-1)} & \frac{-0.281(q-0.955)(q+1.374)}{q(q-1)} \\ \frac{0.039(q^2-1.123q+0.385)}{q(q-1)} & \frac{0.271(q-0.892)(q+0.201)}{q(q-1)} \end{bmatrix} \tag{45}$$

whereas the resulting OCI controller is given by

$$C(q, \hat{P}_{\mathrm{OCI}}) = \begin{bmatrix} \frac{1.722(q-0.917)(q-0.812)}{q(q-1)} & \frac{-3.252(q-0.917)(q-0.841)}{q(q-1)} \\ \frac{1.348(q-0.923)(q-0.854)}{q(q-1)} & \frac{6.837(q^2-1.798q+0.813)}{q(q-1)} \end{bmatrix}. \tag{46}$$

The controller estimated through VRFT using IV is given by

$$C(q, \hat{P}_{\mathrm{IV}}) = \begin{bmatrix} \frac{1.612(q-0.914)(q-0.796)}{q(q-1)} & \frac{-3.074(q-0.928)(q-0.807)}{q(q-1)} \\ \frac{-4.829(q^2-1.946q+0.954)}{q(q-1)} & \frac{11.087(q^2-1.888q+0.901)}{q(q-1)} \end{bmatrix}. \tag{47}$$

Figure 2 shows the closed-loop responses with controllers (46) and (47), where the reference signal is the same as the

[2] Only half of the batches were used for OCI estimations.

Table 2 Medians of $\hat{J}^{\mathrm{MR}}$ and E_c obtained for an open-loop batch of data when $C_d(q) \notin \mathcal{C}$

Controller	$\hat{J}^{\mathrm{MR}}$	E_c
$C(q, \hat{P}_{\mathrm{IV}})$	0.7381	13.333
$C(q, \hat{P}_{\mathrm{OCI}})$	0.7375	13.391

one used to compute (43). Notice that, in one hand, the IV controller presents tracking and decoupling problems. The OCI one, on the other hand, provides a closed-loop response almost equal to the reference model, disregarding the effect of the noise on the outputs.

4.1.2 The Nonideal Case: $C_d(q) \notin \mathcal{C}$

Consider now the design of a centralized PI controller, that is, $C_d(q) \notin \mathcal{C}$. Again, one hundred different controllers were found considering the collected data used in the previous example.

Table 2 is analogous to Table 1. Notice that in this case the medians related to the two methods are almost equal, but very different from the ones shown in the previous table. In particular, the controllers of both methods are biased now, explaining why the OCI medians have greatly increased. Despite of the bias error, the IV medians have decreased.

For a single Monte Carlo run (out of 100), the initial controller is given in this case by

$$C(q, \hat{P}_0) = \begin{bmatrix} \frac{0.342(q-0.945)}{q-1} & \frac{-0.659(q-0.944)}{q-1} \\ \frac{0.040(q-0.781)}{q-1} & \frac{0.294(q-0.872)}{q-1} \end{bmatrix} \tag{48}$$

whereas the OCI methodology yields

$$C(q, \hat{P}_{\mathrm{OCI}}) = \begin{bmatrix} \frac{0.542(q-0.953)}{q-1} & \frac{-0.743(q-0.960)}{q-1} \\ \frac{0.238(q-0.941)}{q-1} & \frac{1.404(q-0.926)}{q-1} \end{bmatrix}. \tag{49}$$

The IV method results in

$$C(q, \hat{P}_{\mathrm{IV}}) = \begin{bmatrix} \frac{0.474(q-0.957)}{q-1} & \frac{-0.799(q-0.972)}{q-1} \\ \frac{0.263(q-1.012)}{q-1} & \frac{2.708(q-0.983)}{q-1} \end{bmatrix}. \tag{50}$$

Closed-loop responses with controllers (49) and (50) are shown in Fig. 3. In this case, since Assumption 1 is not satisfied, both responses are (significantly) different from the reference model, presenting overshoot problems and also coupling when the second reference changes its value.

At last, Fig. 4 shows the estimated cost $\hat{J}^{\mathrm{MR}}$ through box plots[3] considering the case $C_d(q) \notin \mathcal{C}$ but also $C_d(q) \in \mathcal{C}$ (the IV cost was omitted when $C_d(q) \in \mathcal{C}$ because the resulting box plot is too large). Notice that, when $C_d(q) \notin \mathcal{C}$, the variance of the OCI cost is smaller than the IV one, as it also happened when $C_d(q) \in \mathcal{C}$.

[3] On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually.

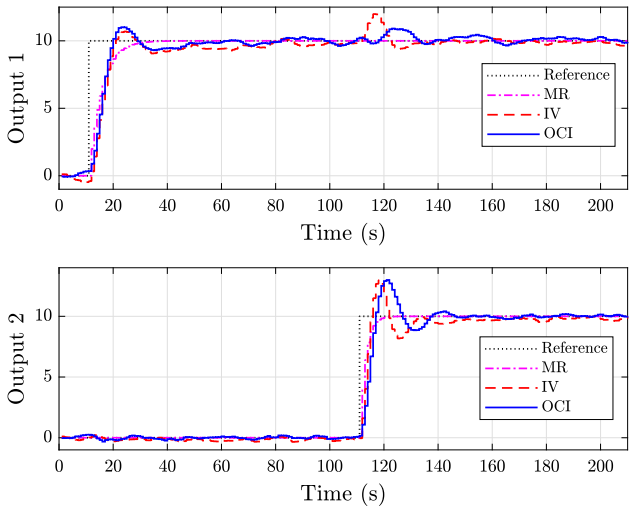

Fig. 3 Closed-loop responses of system (28)–(40) with controllers (49) and (50)

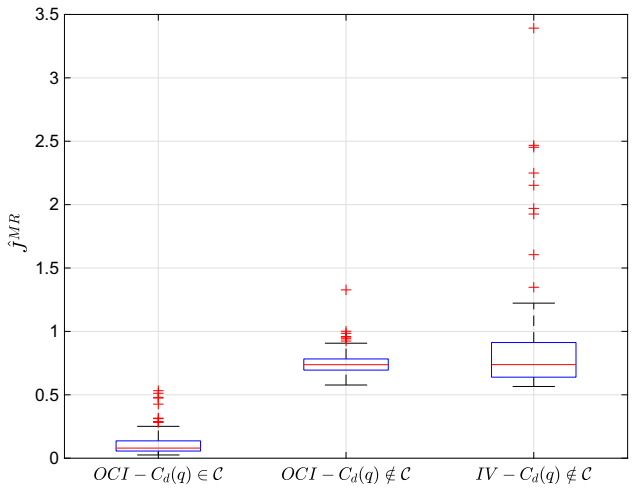

Fig. 4 Distribution of the estimated cost $\hat{J}^{\mathrm{MR}}$ considering 100 Monte Carlo runs for the OCI and IV controllers considering an open-loop batch of data

4.2 Collecting Closed-Loop Data

As pointed out in Sect. 3, when data are collected in closed-loop, it is necessary to identify a (flexible enough) noise model $H(q, \Theta)$ in order to obtain a consistent controller estimate. So, let us assume that the system (28)–(40) operates with the following initial stabilizing PI controller:

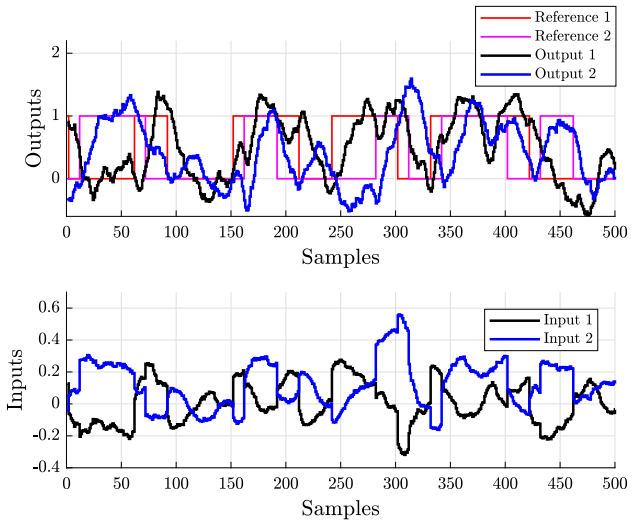

Fig. 5 Closed-loop response of system (28)–(40) with controller (51) to a PRBS reference signal of amplitude 1 (SNR = 5 dB)

$$C_0(q) = \begin{bmatrix} \frac{0.2(q-0.9)}{q-1} & \frac{-0.1(q-0.9)}{q-1} \\ \frac{-0.1(q-0.9)}{q-1} & \frac{0.2(q-0.9)}{q-1} \end{bmatrix}. \tag{51}$$

Data $\{u(t), y(t)\}$ are collected in closed-loop where the reference is set as a PRBS (pseudo random binary signal) with a fundamental clock period of $T = 30$ s and duration of 2000 s. We consider two different amplitudes for the PRBS, 5 and 1, which correspond, respectively, to SNRs of 20 dB and 5 dB at the system's outputs, allowing the analysis of the proposed methodology for a low SNR. Figure 5 shows (part of) the batch of collected data in the second case, where the SNR is low.

Different PID controller designs are made (Assumption 1 is satisfied), using different parametrizations for $H(q, \Theta) = H(q, \theta)$:

$$H_1(q, \theta) = I, \tag{52}$$

$$H_2(q, \theta) = \begin{bmatrix} \frac{q^2 + \theta_1 q + \theta_2}{q^2 + \theta_3 q + \theta_4} & 0 \\ 0 & \frac{q^2 + \theta_5 q + \theta_6}{q^2 + \theta_7 q + \theta_8} \end{bmatrix}, \tag{53}$$

$$H_3(q, \theta) = \begin{bmatrix} \frac{q + \theta_1}{q + \theta_5} & \frac{\theta_2}{q + \theta_5} \\ \frac{\theta_3}{q + \theta_6} & \frac{q + \theta_4}{q + \theta_6} \end{bmatrix}, \tag{54}$$

$$H_4(q, \theta) = \begin{bmatrix} \frac{q^2 + \theta_1 q + \theta_2}{q^2 + \theta_9 q + \theta_{10}} & \frac{\theta_3 q + \theta_4}{q^2 + \theta_9 q + \theta_{10}} \\ \frac{\theta_5 q + \theta_6}{q^2 + \theta_{11} q + \theta_{12}} & \frac{q^2 + \theta_7 q + \theta_8}{q^2 + \theta_{11} q + \theta_{12}} \end{bmatrix}. \tag{55}$$

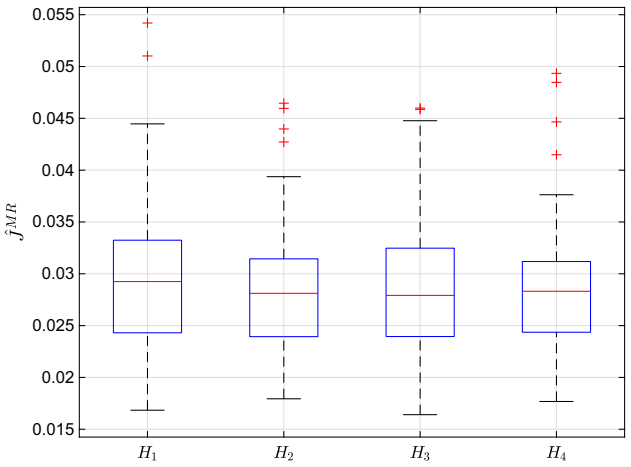

Fig. 6 Distribution of the estimated cost $\hat{J}^{\mathrm{MR}}$ for 100 Monte Carlo runs considering models (52), (53), (54) and (55) for the noise and SNR = 20 dB

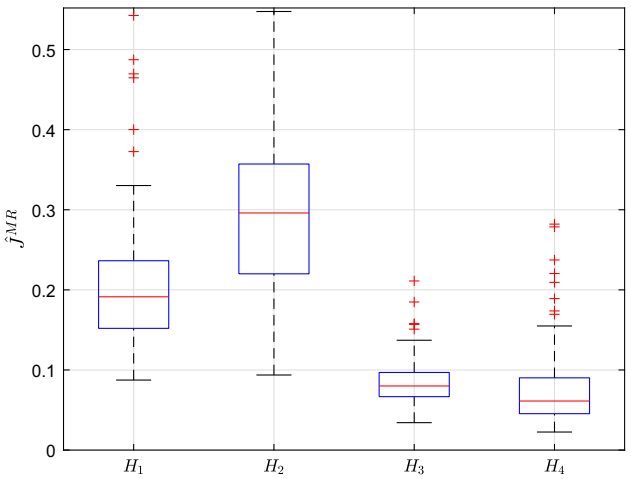

Fig. 7 Distribution of the estimated cost $\hat{J}^{\mathrm{MR}}$ for 100 Monte Carlo runs considering models (52), (53), (54) and (55) for the noise and SNR = 5 dB

Notice that only $H_4(q, \theta)$ is capable of representing the "true" noise model. That is, only in the fourth case, Assumption 2 is satisfied and the controller estimate is consistent.

One hundred Monte Carlo runs were performed. But, when the SNR is low, the controller obtained through least squares (and even the instrumental variable) approach does not even stabilize the plant (in all 100 runs), so we used (51) as initial condition for the OCI algorithm.

The cost $\hat{J}^{\mathrm{MR}}$ was calculated considering step references of amplitude 10, as in Sect. 4.1. The resulting box plots related to the high and low SNRs are shown, respectively, in Figs. 6 and 7.

Notice that when SNR is 20 dB, the choice of the noise model structure does not make much difference in the calculated cost and one could just fix $H(q, \theta) = I$ (compare with Fig. 4). However, when the SNR is low, this is not true.

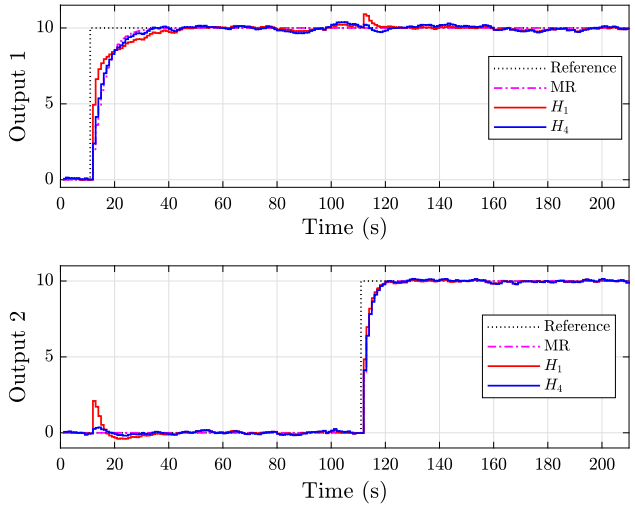

Fig. 8 Closed-loop responses of system (28)–(40) with PID controllers designed considering the models (52) and (55) for the noise when SNR = 5 dB

In particular, in this case, it is not advantageous to use a decoupled noise model, since $H_0(q)$ is not. The closed-loop performance with the designed controller is improved if we choose structures $H_3(q,\theta)$ or $H_4(q,\theta)$ instead of $H_1(q,\theta)$. In fact, results with model structure $H_3(q,\theta)$ is almost as "good" as with $H_4(q,\theta)$ in this example, even though it is underparametrized.

Figure 8 shows the closed-loop responses when choosing structures $H_1(q,\theta)$ and $H_4(q,\theta)$ for a single realization of the design when the SNR is low, where the designed controllers are, respectively:

$$C_1(q,\hat{P}_N) = \begin{bmatrix} \frac{2.757(q^2-1.811q+0.821)}{q(q-1)} & \frac{-2.514(q-0.918)(q-0.785)}{q(q-1)} \\ \frac{5.944(q^2-1.932q+0.936)}{q(q-1)} & \frac{8.200(q-0.921)(q-0.853)}{q(q-1)} \end{bmatrix}, \tag{56}$$

$$C_4(q,\hat{P}_N) = \begin{bmatrix} \frac{1.855(q-0.905)(q-0.845)}{q(q-1)} & \frac{-2.621(q-0.926)(q-0.775)}{q(q-1)} \\ \frac{1.736(q^2-1.846q+0.857)}{q(q-1)} & \frac{6.759(q^2-1.767q+0.781)}{q(q-1)} \end{bmatrix}. \tag{57}$$

Notice that the closed-loop response related to the choice $H_4(q,\theta)$ (and hence to (57)) is indeed closer to $T_d(q)$ than the one provided by (56), specially when the value of the reference of loop 1 changes.

The medians of $\hat{J}^{\mathrm{MR}}$ from Figs. 6 and 7 and also the ones of E_c are shown in Table 3. Notice that the pattern of E_c is not equal to the one of $\hat{J}^{\mathrm{MR}}$, but its values lead to the same conclusion: it is advantageous to identify a noise model as the SNR decreases.

Table 3 Medians of $\hat{J}^{\mathrm{MR}}$ and E_c obtained considering the models (52), (53), (54) and (55) for the noise

SNR	20 dB	20 dB	5 dB	5 dB
Model	$\hat{J}^{\mathrm{MR}}$	E_c	$\hat{J}^{\mathrm{MR}}$	E_c
$H_1(q,\theta)$	0.029250	0.99725	0.19135	9.4594
$H_2(q,\theta)$	0.028116	0.72686	0.29601	7.4962
$H_3(q,\theta)$	0.027914	0.78107	0.080046	3.6975
$H_4(q,\theta)$	0.028312	0.60645	0.061248	2.8712

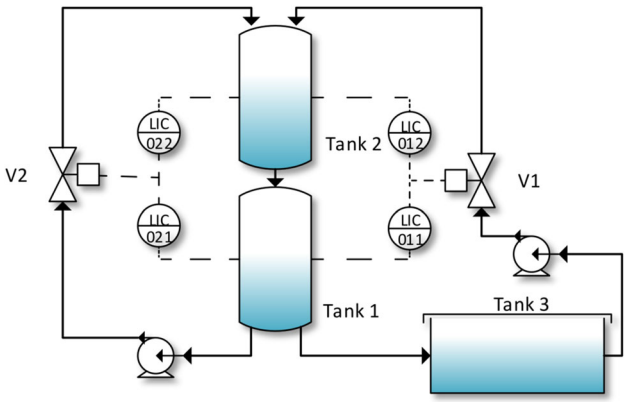

Fig. 9 Schematic diagram of the pilot plant

At last, notice the identified noise model

$$H_4(q,\bar{\theta}_N) = \begin{bmatrix} \frac{(q-0.391)(q-0.009)}{(q-0.793)(q-0.899)} & \frac{0.502(q-0.196)}{(q-0.793)(q-0.899)} \\ \frac{0.502(q-0.494)}{(q-0.792)(q-0.897)} & \frac{(q-0.590)(q-0.003)}{(q-0.792)(q-0.897)} \end{bmatrix},$$

where $\bar{\theta}_N$ is the mean value of $\hat{\theta}_N$ obtained from the Monte Carlo runs (when SNR = 5 dB), is indeed similar to (40).

5 Experimental Results

The OCI methodology presented in this paper was applied to the level control of a pilot plant, which is also the objective of study of Boeira et al. (2018). Its schematic diagram is presented in Fig. 9 and describes the process, which is built with of-the-shelf industrial equipment (pumps, valves, sensors and tanks). Tanks 1 and 2 have 70 liters each, whereas tank 3 is a 250 liters reservoir. The goal is to control the level of the two first tanks.

Communication between devices is made up via a *Foundation Fieldbus H1* network. The pumps are driven by frequency inverters, and the valves are sliding stem pneumatic with embedded PID positioners. Level measurement is carried out by pressure sensors at the bottom of each tank, and plant control and data acquisition are done in a super-

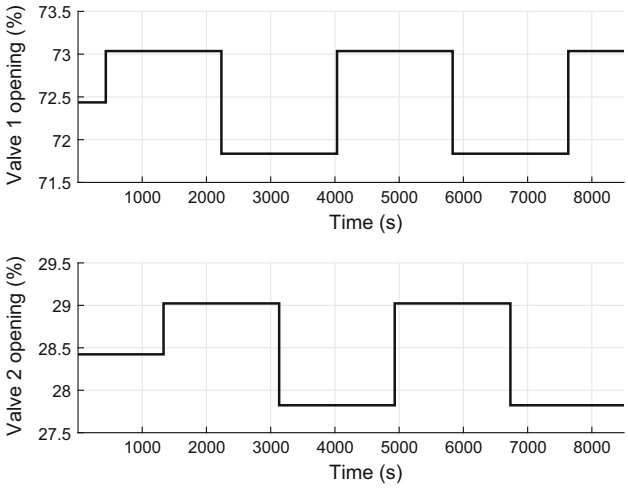

Fig. 10 Input signals of the level plant in the open-loop experiment

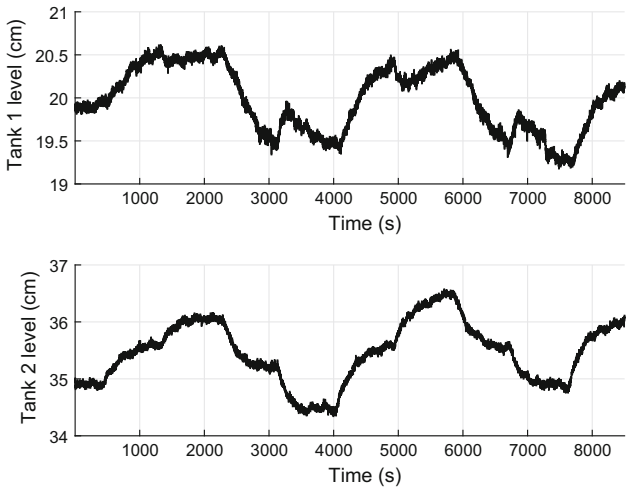

Fig. 11 Output signals of the level plant in the open-loop experiment

visory software *Elipse SCADA* which communicates via an OPC server with a sampling time of 1 s.

During all the experiments, the frequencies of the inverters that drive the pumps are kept at a constant value, while control is performed by valves $V1$ and $V2$. The system's multivariable behavior is represented by

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} G_{11}(q) & G_{12}(q) \\ G_{21}(q) & G_{22}(q) \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}, \tag{58}$$

where $y_1(t)$ and $y_2(t)$ are the tanks' levels, and $u_1(t)$ and $u_2(t)$ are the valves' openings.

5.1 Data Acquisition

Before conducting the experiments, openings of 72.4% and 28.4% were applied to valves 1 and 2, respectively, in order to stabilize the level of tanks 1 and 2 at 20 cm and 35 cm, respectively. Then, square waves of small amplitude were applied to the system in open loop in order to collect data from it, as shown in Fig. 10. Figure 11 in turn shows the corresponding outputs of the system, which are affected by noise. Notice the inputs' variations were chosen so as to have a small SNR, a typical situation in some real-life applications.

5.2 Controller Design

Besides the experimental data, designing the controller requires the definition of its structure as well as the choice of the reference model, as stated earlier.

Though it is not always possible, it is useful and even recommended to use some prior knowledge of the system in the choice of the reference model. This allows the designer to be realistic with respect to the closed-loop performance criteria chosen. In the case treated here, it is known from Campestrini et al. (2016) that the settling times of the tanks 1 and 2 are roughly 900 s and 700 s, respectively. Considering that, the reference model was chosen as

$$T_d(q) = \begin{bmatrix} \frac{0.03}{q-0.97} & 0 \\ 0 & \frac{0.02}{q-0.98} \end{bmatrix} \tag{59}$$

where the desired settling times of tanks 1 and 2 are, respectively, 128 s and 193 s. Besides, it was proposed the decoupling of the system dynamics as it can be seen from the fact that there are nonzero elements only in the main diagonal of the matrix.

The defined controller structure is that of a centralized PI. First of all, let us apply the least-squares method, which gives the following controller:

$$C(q, \hat{P}_{\mathrm{LS}}) = \begin{bmatrix} \frac{0.1247(q-1.024)}{q-1} & \frac{0.2323(q-0.9677)}{q-1} \\ \frac{-0.7519(q-0.9602)}{q-1} & \frac{0.2369(q-0.9205)}{q-1} \end{bmatrix}. \tag{60}$$

Then, we obtain the OCI one:

$$C(q, \hat{P}_{\mathrm{OCI}}) = \begin{bmatrix} \frac{7.2651(q-0.9949)}{q-1} & \frac{5.7126(q-1.001)}{q-1} \\ \frac{-9.0917(q-0.9939)}{q-1} & \frac{-0.2961(q-1.121)}{q-1} \end{bmatrix}. \tag{61}$$

A second open-loop experiment was performed on the plant with exactly the same input in order to obtain an instrumental variable for the VRFT method. The resulting controller is:

$$C(q, \hat{P}_{\mathrm{IV}}) = \begin{bmatrix} \frac{13.629(q-0.9946)}{q-1} & \frac{9.1391(q-1.003)}{q-1} \\ \frac{-7.1033(q-0.9939)}{q-1} & \frac{-0.1432(q-1.214)}{q-1} \end{bmatrix}. \tag{62}$$

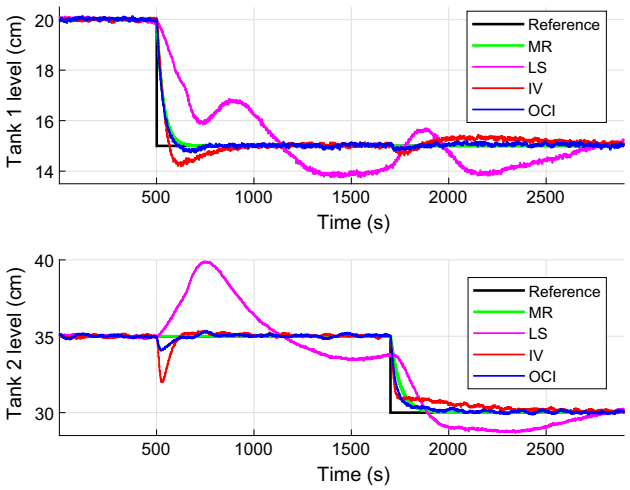

Fig. 12 Output signals of the level plant in the closed-loop experiment

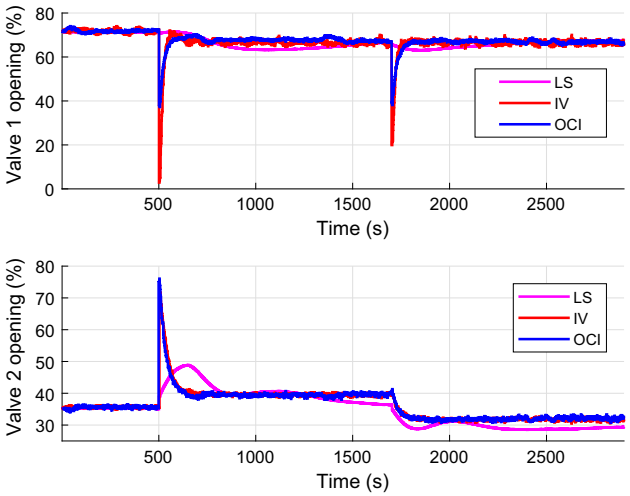

Fig. 13 Input signals of the level plant in the closed-loop experiment

5.3 Closed-Loop Experiment

Controllers (60)–(62) were applied to the closed-loop plant, producing the results shown in Figs. 12 and 13. The MR-calculated costs (43) are $\hat{J}^{\mathrm{MR}}(\hat{P}_{\mathrm{LS}}) = 3.384\ \mathrm{cm}^2$, $\hat{J}^{\mathrm{MR}}(\hat{P}_{\mathrm{IV}}) = 0.26827\ \mathrm{cm}^2$ and $\hat{J}^{\mathrm{MR}}(\hat{P}_{\mathrm{OCI}}) = 0.04401\ \mathrm{cm}^2$, the last one being the smallest one, as it also happened in the simulated example presented in the previous section.

Notice in Fig. 12 that the response of the system using (61) is almost equal to the reference model (MR), indicating that Assumption 1 was not greatly violated. The levels of tanks are almost decoupled. Moreover, there is no overshoot and, apart from the noise, the tracking error is zero in steady state for step references. On the other hand, the performance attained with controller (62) is clearly worse, presenting tracking and decoupling problems. Controller (60), in turn, is not suitable for practical use.

6 Conclusions

A multivariable formulation of the OCI method was presented, where the controller estimate is consistent under ideal conditions. Numerical results have shown that OCI performs well when the collected data are corrupted by noise, even if the signal-to-noise ratio is not high, unlike the instrumental variables approach, which is used in other direct data-driven methods. Simulations also show that it is advantageous to identify the noise model in order to improve the controller estimate, specially if data are collected in closed-loop and the SNR is low.

Moreover, an experiment was conducted in a level plant, where the goal is to control the level of two tanks through the opening of two pneumatic valves. Both OCI and VRFT methods were applied using an open-loop batch of data, and the resulting OCI controller provided a closed-loop response considerably closer to the reference model than the VRFT one, even when the latter uses instrumental variables (and thus an extra batch of data).

Acknowledgements This work was supported by the National Council for Technological and Scientific Development—CNPq/BR.

References

Bazanella, A. S., Campestrini, L., & Eckhard, D. (2012). *Data-driven controller design: The H_2 approach*. Dordrecht: Springer.

Boeira E. C., & Eckhard D. (2018) Multivariable virtual reference feedback tuning with Bayesian regularization. In *XXII Congresso Brasileiro de Automática*, SBA, João Pessoa.

Boeira, E. C., Bordignon, V., Eckhard, D., & Campestrini, L. (2018). Comparing mimo process control methods on a pilot plant. *Journal of Control, Automation and Electrical Systems*, *29*(4), 411–425.

Campestrini, L., Eckhard, D., Chía, A., & Boeira, E. (2016). Unbiased MIMO VRFT with application to process control. *Journal of Process Control*, *39*, 35–49.

Campestrini, L., Eckhard, D., Bazanella, A. S., & Gevers, M. (2017). Data-driven model reference control design by prediction error identification. *Journal of the Franklin Institute*, *354*(6), 2628–2647.

Campi, M., Lecchini, A., & Savaresi, S. (2002). Virtual reference feedback tuning: A direct method for the design of feedback controllers. *Automatica*, *38*, 1337–1346.

Fletcher, R. (1987). *Practical methods of optimization*. New York: Wiley.

Formentin, S., Savaresi, S., & del Re, L. (2012). Non-iterative direct data-driven controller tuning for multivariable systems: Theory and application. *Control Theory Applications, IET*, *6*(9), 1250–1257.

Hjalmarsson, H. (1999). Efficient tuning of linear multivariable controllers using iterative feedback tuning. *International Journal of Adaptive Control and Signal Processing*, *13*(7), 553–572.

Huff D. D., Gonçalves da Silva G. R., & Campestrini L. (2018). Data-driven control design by prediction error identification for a refrigeration system based on vapor compression. In *3rd IFAC conference on advances in proportional-integral-derivative control, IFAC, gent* (Vol. 51, pp. 704–709).

Jansson, H., & Hjalmarsson, H. (2004). Gradient approximations in iterative feedback tuning for multivariable processes. *International Journal of Adaptive Control and Signal Processing*, *18*(8), 665–681.

Karimi, A., van Heusden, K., & Bonvin, D. (2007). Noniterative data-driven controller tuning using the correlation approach. *European Control Conference*, *2007*, 5189–5195.

Ljung, L. (1991). *System identification toolbox: For use with MATLAB—User's guide*. Natick: Math Works.

Ljung, L. (1999). *System identification: Theory for the User* (2nd ed.). Upper Saddle River: Prentice-Hall.

Miskovic, L., Karimi, A., Bonvin, D., & Gevers, M. (2005). Correlation-based tuning of linear multivariable decoupling controllers. In *44th IEEE conference on decision and control, 2005 and 2005 European control conference, CDC-ECC '05* (pp. 7144–7149).

Rallo, G., Formentin, S., Chiuso, A., & Savaresi, S. M. (2016). Virtual reference feedback tuning with Bayesian regularization. In *2016 European control conference (ECC)* (pp. 507–512).

Söderström, T. (2007). Errors-in-variables methods in system identification. *Automatica*, *43*(6), 939–958.

Söderström, T., & Stoica, P. (1989). *System identification*. Upper Saddle River: Prentice-Hall.

Yubai, K., Usami, H., & Hirai, J. (2009). Correlation-based direct tuning of MIMO controllers by least-squares and its application to tension-and-speed control apparatus. In *ICCAS-SICE* (pp. 931–936).