Direct Nonlinear Control Design: The Virtual Reference Feedback Tuning (VRFT) Approach

Marco C. Campi, *Member, IEEE,* and Sergio M. Savaresi, *Member, IEEE*

***Abstract*—This paper introduces the virtual reference feedback tuning (VRFT) approach for controller tuning in a nonlinear setup. VRFT is a data-based method that permits to directly select the controller based on data, with no need for a model of the plant. It is based on a global model reference optimization procedure and, therefore, does not require to access the plant for experiments many times so as to estimate the control cost gradient. For this reason, it represents a very appealing controller design methodology for many control applications.**

***Index Terms*—Controller tuning, data-based tuning, direct control, model reference control, nonlinear systems.**

I. Introduction

IN THIS paper, we consider the problem of designing a controller for a nonlinear plant on the basis of input/output measurements (data-based design) with no need for a mathematical description of the plant.

Designing controllers based on measurements is of great importance in connection with industrial applications since it is common experience in industrial control design that a mathematical description of the plant is not available and that undertaking a modeling study is too costly and time-consuming. Moreover, even when a mathematical description of a nonlinear plant is available, such a description is often too complex to be used for design purposes. Rather, it can be used as a plant simulator, and this is synergic to data-based design methods: Data are generated by the simulator without upsetting the real plant operation and then they are used in the data-based method to design the controller.

The method developed in this paper is called virtual reference feedback tuning (VRFT) and generalizes a previously introduced method—still known under the same name of VRFT—for linear design; see [1]–[5].

VRFT is a "one-shot" direct data-based controller design method (this terminology is explained in detail in Sections I-A and B). In the linear context, VRFT does have features that make it particularly appealing, but it is just an alternative to other existing methods in this same category. In contrast, no other "one-shot" direct data-based controller design methods seem to exist for nonlinear controller tuning, which makes VRFT a unique methodology.

A. Direct Data-Based Controller Design

Data-based controller design methods are adaptive methods where adaptation is performed offline. This means that a controller class is first selected, and then a specific controller is chosen in the class based on the collected data. Differently from standard online adaptive control where the controller choice (adaptation) is performed online during the normal operation of the closed-loop control system, in data-based controller design the selection is performed offline, based on a batch of data collected from the plant, and the designed controller is then placed in the loop without allowing for any further adaptation. The advantage gained over online adaptive control is that the designed controller can be tested for diverse requirements (e.g., stability) before it is placed in the loop, so overcoming the traditional difficulty of standard online adaptive control that the closed-loop behavior is difficult to predict. This is a substantial advantage in control applications.

More specifically, VRFT is a *direct* data-based method. Being direct means that the controller is directly selected without preliminarily using the data to identify a model of the plant: the controller selection is obtained through an optimization procedure where the optimization variables are directly the controller parameters. Direct methods present two fundamental advantages over indirect ones.

i) In indirect methods, the identification phase has to return a model which meets a suitable compromise between simplicity and reliability. However, deciding which aspects of the plant dynamics have to be kept in the model since they significantly impact the final control objective is in general a difficult task. In direct methods, the degrees-of-freedom in the controller selection are directly spent toward the achievement of the final control goal and the aforementioned modeling problem disappears. So to say, the relevant aspects in the plant dynamics take care of themselves through the impact they have on the optimization control cost.

ii) In direct methods, a controller class of specified structure can be *a-priori* selected. If, e.g., one is intentioned to use, say, a PID controller, the PID controller class can be considered. This is different from indirect methods where relatively complex model structures have often to be used in identification, which then result in complex controller structures (the structure of the controller is related to the structure of the model for many control design methods). Thus, if one desires

Manuscript received February 7, 2005; revised August 4, 2005. Recommended by Associate Editor M.-Q. Xiao. This work was supported by the Ministero dell'Istruzione, dell'Universita' e della Ricerca (MIUR) under the project New Methods for Identification and Adaptive Control for Industrial Systems.

M. C. Campi is with the Dipartimento di Elettronica per l'Automazione, Università di Brescia, 25123 Brescia, Italy (e-mail: campi@ing.unibs.it).

S. M. Savaresi is with the Dipartimento di Elettronica ed Informazione, Politecnico di Milano, 20133 Milano, Italy.

Digital Object Identifier 10.1109/TAC.2005.861689

a simple controller, he/she has often to a-posteriori undertake a difficult controller reduction phase. For the sake of completeness, however, it should also be mentioned that the *a-priori* selection of a suitable controller class is not always a simple task in direct methods.

B. VRFT: A One-Shot Direct Data-Based Method

The VRFT method introduced in this paper is a *one-shot* direct data-based method: One collects one batch of data from the plant (two batches in case of noisy data) and the procedure returns a controller, without requiring iterations and/or further accesses to the plant for experiments. The reason why this is possible is that the "design engine" inside VRFT is intrinsically global, and no gradient-descent techniques are involved.

For the sake of completeness, we must add that the quality of the selected controller obviously depends on the experimental conditions in which data have been collected. For badly exciting signals, the selected controller can be poorly performing, and this may require undertaking additional rounds of data collection. This is intrinsically so: If data do not "explore" enough, not enough information is available for design. The point is that VRFT directly searches for the global optimum, relative to the information content present in the given batch of data.

The fact that VRFT is one-shot makes it very appealing because

a) it is low-demanding, i.e., one has not to access the plant for experiments many times, halting the normal operation of the plant;
b) VRFT does not suffer from local minima and initialization problems.

The VRFT method was originally proposed in a *linear framework* by the same authors of this paper, see [1]–[3], and then used by others; see. e.g., [6]–[10]. For linear plants, VRFT provided an alternative to most traditional methodologies, such as Ziegler and Nichols tuning method and alike, [11], [12]. In the *nonlinear context* of this paper, no direct one-shot data-based methods exist for controller tuning and VRFT offers a viable, simple, and convenient way to address this problem. In the earlier paper [4], only the bare idea behind designing a controller based on a virtual reference was introduced. Here, VRFT is developed in all its aspects to a ready-to-use procedure.

A method alternative to VRFT—but based on an iterative gradient-descent approach—is iterative feedback tuning (IFT). In a sense, IFT is complementary to VRFT in that it calls for many accesses to the plant for data collection and suffers from local minima problems, but—if suitably initialized—it provides the optimal controller minimizing the control cost. In contrast, VRFT is one-shot and therefore very convenient when accessing the plant many times is a problem, but it provides the optimal controller only under ideal conditions, while a nearly optimal controller is achieved in general situations (see Section III). IFT was initially introduced in a *linear context* in [13]–[15], grounded on a bright idea for the calculation of the gradient using global (i.e., nonincremental) signals. Turning to a nonlinear context, [16]–[18], the computation of the gradient does require the use of incremental signals, so leading to a delicate compromise: The increments must be not too large (since gradient is inherently a local concept), but they need not be too small as well (since otherwise they get buried in noise). VRFT avoids this problem since it performs a global search in one-shot and uses global signals.

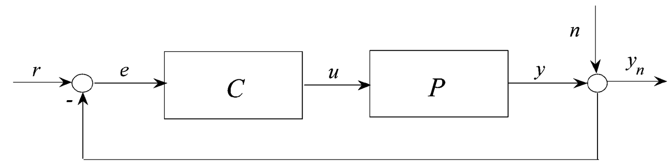

Fig. 1. Control system.

C. Structure of the Paper

Section II contains the control setting, while the VRFT method is presented in Sections III–V. Finally, Section VI delivers a multipass procedure for refining the experimental conditions in which data are collected.

II. Control Setting

A. Control System

The control system we make reference to is as represented in Fig. 1. It is a classical one-degree-of-freedom control system where the controller C processes the error signal e so as to generate the control input u to the plant P. y_n is the plant output corrupted by noise n, and r is the reference signal. P and C are in general *nonlinear* systems.

B. Control Objective (General)

The control objective is to design a controller C so that the control system behavior adheres as much as possible to that of a given model reference M when the reference trajectory is a given signal $\tilde{r}$. When $\tilde{r}$ is sufficiently exciting and the controller C does achieve a good adherence, this requirement also entails that the feedback control system resembles M for a large class of reference signals. In general, however, the control objective is a requirement on the reference trajectory $\tilde{r}$ only.

A precise mathematical specification of the control objective is given in Section II-I.

C. Plant

The nonlinear plant P is a discrete-time single-input–single-output nonlinear dynamical system described as

$$y(t) = p(y(t-1), \ldots, y(t-n_{Py}) \qquad u(t-1), \ldots, u(t-n_{Pu})) \quad (1)$$

where p is a nonlinear function.

Remark 1: The u to y delay in P is 1. Generalizing the results in this paper to a multiple delay setting presents no difficulties.

Later on, we will consider reference models with delay 1. If the plant delay is $d > 1$, a same delay should be introduced in the model. If not, minimizing a model reference cost may produce severe deteriorations in the control system performance, a fact that is true in general, not just for VRFT. $*$

We want to see the plant as operating as follows:

— it is initialized with the initial conditions: $i.c. = y(0), \ldots, y(1-n_{Py}), u(-1), \ldots, u(1-n_{Pu})$;
— it is fed by an input signal applied in a given interval, say $[0, N-1]$: $u(0{:}N-1) := [u(0) \cdots u(N-1)]^T$.

Then, P generates an output $y(1{:}N)$ according to (1). This output is written as: $y(1{:}N) = P[u(0{:}N-1), i.c.]$, so emphasizing the fact that—for a given $i.c.$—P operates as a nonlinear map from $\mathbb{R}^N$ to $\mathbb{R}^N$.

Example 1: To illustrate ideas in an easy-to-follow manner, the theoretical developments of this paper will be accompanied step by step by the following simple example. Consider the plant

$$y(t) = y(t-1) + u(t-1)^3.$$

Take $N = 2$, and $i.c. = 0$; operator P is then described by

$$\begin{bmatrix} y(1) \\ y(2) \end{bmatrix} = \begin{bmatrix} u(0)^3 \\ u(0)^3 + u(1)^3 \end{bmatrix}.$$

This result illustrates a general fact: P is a (nonlinear) lower triangular operator (i.e., the ith element of the output depends on the first jth, $j \leq i$, input elements only). $*$

We make the following assumptions.

Assumptions

— **A.1** p is smooth;
— **A.2** for any given i.c., if $u_1(0{:}N-1) \neq u_2(0{:}N-1)$, then $P[u_1(0{:}N-1), i.c.] \neq P[u_2(0{:}N-1), i.c.]$. $*$

Remark 2: Assumption A.2 is an invertibility condition on map P. In [19], it is proven that the invertibility of map P for inputs defined over the time horizon $[0{:}N-1]$ implies the invertibility of the same map over any interval $[0{:}T]$, with $T \leq N-1$. $*$

Remark 3: In Assumption A.2, the invertibility of the plant is required to hold for any possible input sequence. While this condition is satisfied for linear plants (and it holds true for some notable classes of nonlinear plants as well), yet it can be restrictive for general nonlinear systems. However, as it can be easily verified, the whole theory of this paper goes through unaltered if invertibility is required in suitable U and Y neighborhoods of given input and output trajectories only, and the u and y signals stay in these neighborhoods. We have preferred to assume invertibility in the large to ease the notations. $*$

D. Controller

The controller is a nonlinear system (that we select in a prespecified class according to the VRFT procedure that will be illustrated later on) described as

$$u(t) = c(u(t-1), \ldots, u(t-n_{Cu}), e(t), \ldots, e(t-n_{Ce})).$$

Similarly to P, we want to see the controller operating as follows:

— it is initialized with the initial conditions: $i.c. = u(-1), \ldots, u(-n_{Cu}), e(-1), \ldots, e(-n_{Ce})$;
— it is fed by the error signal $e(0{:}N-1)$.

Then, C generates signal $u(0{:}N-1)$, which we write as $u(0{:}N-1) = C[e(0{:}N-1), i.c.]$.

E. Feedback Operation

In order to describe the feedback control system in Fig. 1, the plant and controller equations in Sections II-C and D have to be complemented with the relations describing the control system interconnections. This leads to equation

$$\begin{aligned} y(1{:}N) &= P[u(0{:}N-1), i.c.] \\ &= P[C[e(0{:}N-1), i.c.], i.c.] \end{aligned} \tag{2}$$

with

$$e(0{:}N-1) := r(0{:}N-1) - y(0{:}N-1).$$

Given $r(0{:}N-1)$ and the $i.c.$'s for the plant and the controller, (2) defines one and only one $y(1{:}N)$, as is clear by solving (2) recursively.

F. Initial Conditions and Shorthand Notations

In the sequel, we let: $i.c.$ of $P = 0$, $i.c.$ of $C = 0$.

More generally, one could assume nonzero initial conditions with some extra notational complications. Moreover, there is no need for the initial conditions to be consistent one with another; for example, $u(t-2)$ is an argument of the plant equation which gives the initial condition $u(-1)$, and in controller we also have $u(t-1)$ which again gives the initial condition $u(-1)$. The fact that in both cases we have the same symbol $u(t-1)$ stems from the fact that—during the normal operation—the two variables are the same. However, one could as well use different symbols for the two initial conditions $u(-1)$ in the plant and in the controller and let them assume different values.

The assumption on the initial conditions to be all zero is made to ease notations and generalizations are easy to obtain. We also remark that—under stability conditions—if N is large the initial conditions play a marginal role.

Since the initial conditions are set to zero, from now on we omit indicating them explicitly. Moreover, we gain in readability by also dropping the time argument, and we shall write: u for $u(0{:}N-1)$, r for $r(0{:}N-1)$, e for $e(0{:}N-1)$, and y for $y(1{:}N)$. Also, $y(0{:}N-1)$ is written as Dy, where D is the delay matrix defined as

$$D := \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \tag{3}$$

With all these notational conventions in place, the closed-loop system writes $y = P[C[r - Dy]]$.

G. Parameterized Controller Class

In VRFT, the goal is to select a suitable controller in a given parameterized controller class, viz.

$$u(t) = c(u(t-1), \ldots, u(t-n_{Cu}), e(t), \ldots, e(t-n_{Ce}); \theta).$$

Given a $\theta \in \mathbb{R}^{n_\theta}$, the corresponding controller is written C_θ and the closed-loop system is $y_\theta = P[C_\theta[r - Dy_\theta]]$, where the index θ in y_θ emphasizes the controller used.

The following assumption is in place.

Assumption

— **A.3** $c : \mathbb{R}^{n_{Cu}+n_{Ce}+1+n_\theta} \to \mathbb{R}$ is smooth. $*$

H. Desired r to y Mapping

The control objective is expressed by saying that—for a given reference $\tilde{r}$ of interest—the control system behaves as closely as possible as an assigned reference model M. While we postpone details on the control objective to Section II-I, we here introduce in a formal way the reference model M

$$M \text{ is a linear map } r \to y.$$

Remark 4: While nonlinear maps M could as well be considered, it is usually more convenient for the user to express his/her control objective through a linear system M where concepts like frequency response, bandwidth etc. make sense to give an intuitive handle on the choice made. $*$

Assumption

— **A.4** M is lower triangular and invertible. $*$

The fact that M is lower triangular simply means that operator M is causal with a delay at least of 1, i.e., at least the plant delay (remember that $r = r(0{:}N-1)$ and $y = y(1{:}N)$ are defined with a 1-time delay shift one with respect to the other). Invertibility of M entails that the delay is actually equal to 1.

Example 2: The simplest example of M is $M = I$, which corresponds to requiring that $y(t) = r(t-1)$. $*$

As a typical choice, M is assigned through a reference model filter

$$M(z^{-1}) = \frac{b_1 z^{-1} + \cdots + b_{n_{Mr}} z^{-n_{Mr}}}{1 + a_1 z^{-1} + \cdots + a_{n_{My}} z^{-n_{My}}} \qquad (4)$$

which in the time domain corresponds to

$$y(t) = -a_1 y(t-1) - \cdots - a_{n_{My}} y(t-n_{My}) + b_1 r(t-1) + \cdots + b_{n_{Mr}} r(t-n_{Mr}).$$

Supposing $b_1 \neq 0$ and that, for simplicity, the filter initial conditions are zero, then (4) defines M such that Assumption A.4 holds.

I. Control Objective (Mathematical Specification)

The control objective is expressed as

$$\text{minimize } J(\theta) := \|y_\theta - M[\tilde{r}]\|^2 \quad y_\theta = P[C_\theta[\tilde{r} - Dy_\theta]] \quad (5)$$

where $\tilde{r}$ is a given reference signal, and $\|\cdot\|$ is Euclidean norm.

Remark 5: If $\{C_\theta, \theta \in \mathbb{R}^{n_\theta}\}$ is sufficiently rich and $\tilde{r}$ is sufficiently exciting, solving (5) returns a controller such that the feedback control system is close to M. In a general nonlinear context, (5) is a requirement on the reference trajectory $\tilde{r}$ only. $*$

As it has been repeated more than once, our standpoint is that a description of the plant is not available (P in (5) is unknown), so that the optimization problem (5) is not a standard optimal control problem. Along the VRFT approach, the lack of knowledge of P is compensated for through input/output data collected from the plant and elaborated in an offline direct data-based controller design scheme. In Sections III–V, we present the heart of the method, where one batch of input/output data is used to come up with a controller that allows the feedback control system to suitably track a reference signal $\tilde{r}$.

III. VRFT Approach

A. Data

We assume that a batch of input/output data coming from the plant is available. How this batch has been generated is immaterial for the description of the VRFT algorithm and, therefore, we omit any comment on this point in this section and direct the reader to Section VI for further discussion. Moreover, for the sake of presentation clarity, we assume for the time being that the data have been generated noise-free. The noisy case is treated in Section V-C.

The batch of data is

$$\tilde{u}(0{:}N-1),\ \tilde{y}(1{:}N) \quad \text{with } \tilde{y}(1{:}N) = P[\tilde{u}(0{:}N-1)].$$

We shall write $\tilde{u}$ for $\tilde{u}(0{:}N-1)$, and $\tilde{y}$ for $\tilde{y}(1{:}N)$.

B. Meeting the Control Objective (5) When $\tilde{r}$ is the "Virtual Reference"

Introduce the reference signal $\tilde{r} := M^{-1}[\tilde{y}]$, where $\tilde{y}$ is the actual output signal collected from the plant. Our goal here is to design C_θ so as to meet (5) for this reference signal.

Remark 6: The reference signal $\tilde{r} := M^{-1}[\tilde{y}]$ is "artificially constructed" from data and it is well possible that it does not coincide with the reference trajectory one is interested in. On the other hand, as already pointed out in Remark 5, if $\{C_\theta, \theta \in \mathbb{R}^{n_\theta}\}$ is sufficiently rich and $\tilde{r}$ is sufficiently exciting, solving (5) delivers a controller such that the closed-loop resembles M for a large class of reference signals, and it is our experience that the design performed on $\tilde{r} := M^{-1}[\tilde{y}]$ is practically suitable for other references of interest. The reader is also referred to [1] for a theoretical study of this issue in a linear context.

When, however, the minimization of (5) in correspondence of a specific reference signal $\tilde{r}$ is required, a multi-step procedure can be undertaken, where a sequence of $\tilde{r} := M^{-1}[\tilde{y}]$ signals are generated converging to the desired $\tilde{r}$ of interest. This is explained in Section VI. $*$

Remark 7: $\tilde{r}$ admits a simple interpretation: It is the reference signal such that—when it is injected at the control system input—we are happy to see $\tilde{y}$ as the corresponding output since $\tilde{y} = M[\tilde{r}]$. $*$

Remark 8: $\tilde{r}$ is called the *"virtual reference"* where "virtual" indicates that it does not exist in reality and, in particular, it was not in place when data $\tilde{u}$ and $\tilde{y}$ were collected. It only exists in our computer where we construct it by relation $\tilde{r} := M^{-1}[\tilde{y}]$ and it is generated for future use in the VRFT algorithm. The virtual reference $\tilde{r}$ gives the name to the whole method: VRFT = Virtual Reference Feedback Tuning. $*$

The basic idea behind VRFT is now explained. The control cost in (5) depends on P, i.e., on the unknown plant, so that we cannot minimize it directly. However, we can set out to minimizing the following alternative P-free cost:

$$\text{minimize } J_{\mathrm{VRFT}}(\theta) := \|F[C_\theta[\tilde{e}]] - F[\tilde{u}]\|^2 \quad \tilde{e} = \tilde{r} - D\tilde{y} \tag{6}$$

where $F : \mathbb{R}^N \to \mathbb{R}^N$ is a filter to be chosen. The important fact is that $J_{\mathrm{VRFT}}(\theta)$ in (6) is a purely data-dependent cost [differently from (5), P does not show up explicitely in (6)] and it can, therefore, be minimized.

The intuitive logic behind (6) is as follows (for the time being, let us forget about F whose role is unessential to the following explanation): A good controller is one that produces $\tilde{u}$ when fed by $\tilde{e} = \tilde{r} - D\tilde{y}$ because—through P—this generates $\tilde{y}$, the desired output when reference is $\tilde{r}$. This reasoning represents the core of the VRFT method.

The rest of this section is devoted to showing that (6) can actually be used for controller selection in place of (5). Specifically, we show that minimizing (6) returns a minimizer of (5) in case perfect matching ($y_\theta = M[\tilde{r}]$, for some θ) is possible. This result holds regardless of the choice of F. The significance of F comes into play to match up (6) to (5) when perfect matching cannot be achieved and this is discussed in Section IV.

Theorem 1 substantiates the fact that minimizing (6) returns a minimizer of (5) when perfect matching is possible. Before the theorem, we give a simple example to illustrate ideas.

Example 3 (Example 1 Continued): For the plant in Example 1, consider the controller class $u(t) = \theta e(t)^{1/3}$ and the reference model $y(t) = r(t-1)$.

The system is operated for $N = 2$ instants with the input $\tilde{u}(0) = 1$, $\tilde{u}(1) = 1$, so generating the output $\tilde{y}(1) = 1$, $\tilde{y}(2) = 2$, from which we compute $\tilde{r}(0) = 1$, $\tilde{r}(1) = 2$.

Suppose now for a moment that the plant is known. If this were the case, the control objective in (5) could be explicitly calculated as follows:

$$\begin{aligned} y_\theta(1) &= y_\theta(0) + \theta^3(\tilde{r}(0) - y_\theta(0)) \\ &= 0 + \theta^3(1-0) = \theta^3; \\ y_\theta(2) &= y_\theta(1) + \theta^3(\tilde{r}(1) - y_\theta(1)) \\ &= \theta^3 + \theta^3(2-\theta^3) = 3\theta^3 - \theta^6 \end{aligned}$$

so that

$$\begin{aligned} J(\theta) &= (\theta^3 - 1)^2 + ((3\theta^3 - \theta^6) - 2)^2 \\ &= 5 - 14\theta^3 + 14\theta^6 - 6\theta^9 + \theta^{12}. \end{aligned}$$

Note that $\theta_0 = 1$ gives perfect tracking. Function $J(\theta)$ is depicted in Fig. 2.

Next, we write the J_{VRFT} cost, with $F = I$

$$\begin{aligned} \tilde{e}(0) &= \tilde{r}(0) - 0 = 1 - 0 = 1 \\ \tilde{e}(1) &= \tilde{r}(1) - \tilde{y}(1) = 2 - 1 = 1 \end{aligned}$$

so that

$$\begin{aligned} J_{\mathrm{VRFT}}(\theta) &= (\theta \cdot 1^{1/3} - 1)^2 + (\theta \cdot 1^{1/3} - 1)^2 \\ &= 2 - 4\theta + 2\theta^2. \end{aligned}$$

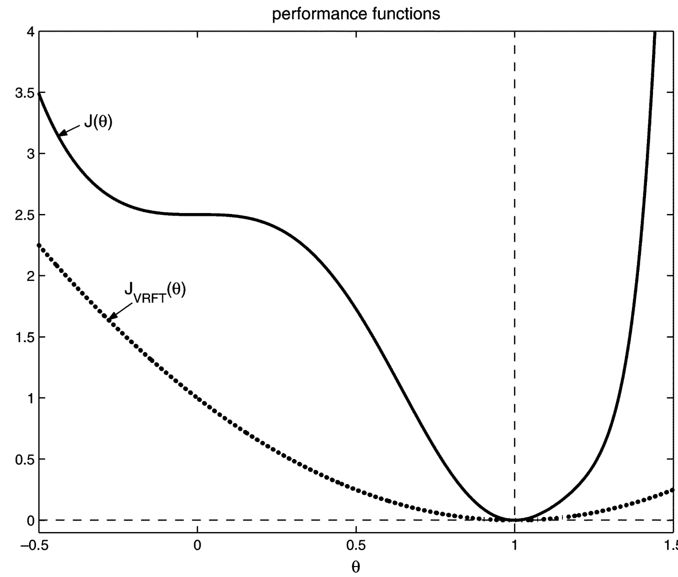

Fig. 2. $J(\theta)$ vs. $J_{\mathrm{VRFT}}(\theta)$.

The important fact to note here is that $J_{\mathrm{VRFT}}(\theta_0) = 0$, i.e., θ_0 is also obtained by minimizing $J_{\mathrm{VRFT}}(\theta)$, despite that $J_{\mathrm{VRFT}}(\theta) \neq J(\theta)$ (see Fig. 2 for a graphical comparison of $J(\theta)$ and $J_{\mathrm{VRFT}}(\theta)$).

Two comments are in order.

— In contrast with J, the J_{VRFT} cost can be constructed out of data without knowledge of P.

— Since the controller class is in this example linearly parameterized, J_{VRFT} is quadratic in θ (and, therefore, easy to minimize). $*$

The fact seen in the previous example that the ideal θ_0 giving perfect matching can be found by minimizing $J(\theta)$ is a general fact and it is proven in the next theorem.

Theorem 1: If θ_0 gives perfect tracking: $\|y_{\theta_0} - M[\tilde{r}]\|^2 = 0$, then θ_0 is a minimizer of $\|F[C_\theta[\tilde{e}]] - F[\tilde{u}]\|^2$. $*$

Thus, if $\|F[C_\theta[\tilde{e}]] - F[\tilde{u}]\|^2$ has a unique minimizer, such a minimizer also minimizes $\|y_{\theta_0} - M[\tilde{r}]\|^2$ and gives perfect matching.

Proof: $\|y_{\theta_0} - M[\tilde{r}]\|^2 = 0$ entails

$$y_{\theta_0} = \tilde{y}. \tag{7}$$

However

$$y_{\theta_0} = P[C_{\theta_0}[\tilde{r} - Dy_{\theta_0}]] \text{ and } \tilde{y} = P[\tilde{u}]$$

so that in view of Assumption A.2 we get

$$C_{\theta_0}[\tilde{r} - Dy_{\theta_0}] = \tilde{u}. \tag{8}$$

Also, from (7)

$$\tilde{r} - Dy_{\theta_0} = \tilde{r} - D\tilde{y} = \tilde{e}$$

which, used in (8), gives

$$C_{\theta_0}[\tilde{e}] = \tilde{u}$$

from which the thesis follows. $*$

The theorem result points to a conceptually interesting property of J_{VRFT} that has a great importance for applications. This property can be rephrased as follows: we set out in the first place to minimize a (generally highly nonconvex) control cost J which, however, cannot be computed since it depends on P, an unknown element in the control problem. On the other hand, another cost J_{VRFT} can be constructed from data without knowledge of P. This cost is different from J (and it is in fact quadratic in case of linearly parameterized controllers as in Example 3), but it shares with J the same minimizer in case the controller class is large enough to allow for perfect matching and therefore it can be used to minimize J.

It remains to understand what happens when perfect matching is not possible. In Section IV, we shall see that—by suitably selecting the prefiltering action F—minimizing J_{VRFT} leads to a θ that 'nearly minimizes' J even when no perfect matching can be achieved. Since perfect matching is not expected in practical applications, the use of filter F is important and the (somewhat complicated) analysis of Section IV is justified.

IV. Filter Design

A. Filter Objective (General)

When perfect matching is not possible, we use F so that minimizing J_{VRFT} generates a "nearly minimizer" of J.

The logic behind the selection of F is as follows. We first introduce a so-called "*ideal controller*," i.e., a controller that, if put in the loop, generates a closed-loop system that coincides with M, the reference model. We prove that such an ideal controller exists. However, the ideal controller is usually a complex nonlinear system and it does not belong to our controller class: It is introduced for analysis purposes only, and moreover its expression is not used in the filter F, so that the actual computation of the ideal controller is not required when implementing the filter. Next—again for analysis purposes—we consider an "*ideal control design problem*" where J is minimized over an expanded controller class that contains the ideal controller and show that a suitable selection of F permits to make the J_{VRFT} cost for this ideal problem to be the second order expansion of J for the same problem. Then, returning to the original J and J_{VRFT} costs, we can see that these two costs are the same as the ideal J and J_{VRFT} except that the minimization is conducted in a constrained sense over the selected controller class. However, then, minimizing J_{VRFT} with the selected filter returns a nearly minimizer of J since we are minimizing in a constrained sense the second-order expansion of J.

We first start with the theoretical developments. Simulation examples are provided in Section IV-E.

B. The "Ideal Controller"

To ease the notations, we here indicate with M the (lower triangular) matrix that represents operator M defined in Section II-H: Given r, the matrix product Mr is the output of the linear operator M applied to r.

Using the delay matrix D in (3), we have that matrix $I-MD$ takes on the form

$$I-MD=\begin{bmatrix}1 & 0 & \cdots & 0 & 0\\ * & 1 & \cdots & 0 & 0\\ * & * & \cdots & 0 & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ * & * & \cdots & * & 1\end{bmatrix}$$

($*$ denotes a generic element) and is therefore invertible. Since in view of Assumption A.2 map P is invertible too, the following definition makes sense:

$$C^0 := P^{-1}(I-MD)^{-1}M.$$

C^0 is the "ideal controller."

If such a C^0 is put in the loop, the closed-loop r to y map is given by M, as it can be easily verified

$$\begin{aligned}y &= P[C^0[r-Dy]] = (I-MD)^{-1}M(r-Dy)\\ &\Rightarrow (I-MD)y = Mr - MDy\\ &\Rightarrow y - MDy = Mr - MDy\\ &\Rightarrow y = Mr.\end{aligned}$$

Example 4 (Example 3 Continued): For the situation in Example 3 we have $M=I$, so that

$$(I-MD)^{-1}M=(I-D)^{-1}=\begin{bmatrix}1 & 0\\ -1 & 1\end{bmatrix}^{-1}=\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}.$$

Moreover

$$P:\begin{cases}y(1)=u(0)^3\\ y(2)=u(0)^3+u(1)^3\end{cases}$$
$$P^{-1}:\begin{cases}u(0)=y(1)^{1/3}\\ u(1)=(y(2)-y(1))^{1/3}\end{cases}$$

leading to

$$\begin{aligned}C^0\begin{bmatrix}e(0)\\ e(1)\end{bmatrix} &= P^{-1}(I-MD)^{-1}M\begin{bmatrix}e(0)\\ e(1)\end{bmatrix}\\ &= P^{-1}\begin{bmatrix}e(0)\\ e(0)+e(1)\end{bmatrix}=\begin{bmatrix}e(0)^{1/3}\\ e(1)^{1/3}\end{bmatrix}\end{aligned}$$

i.e., the ideal controller is $u(t)=e(t)^{1/3}$. $*$

C. The Ideal Control Design Problem

Let

$$\theta^+ := [\theta^T\,\tilde{\theta}]^T \quad \tilde{\theta}\in\mathbb{R};$$
$$C_{\theta^+} := C_\theta + \tilde{\theta}(C^0 - C_0)$$

where C_0 is C_θ computed for $\theta=0$. Note that

— C^0 is obtained for $\theta_0^+ := [0^T\ 1]^T$;

— $\{C_\theta\}$ is obtained by imposing the constraint $\tilde{\theta}=0$.

The ideal control objective is given by

$$\begin{aligned}\text{minimize } & J(\theta^+) := \|y_{\theta^+} - M[\tilde{r}]\|^2\\ \text{with } & y_{\theta^+} = P[C_{\theta^+}[\tilde{r}-Dy_{\theta^+}]].\end{aligned}\tag{9}$$

Remark 9: Given $\tilde{r}$, $y_{\theta^+} = P[C_{\theta^+}[\tilde{r}-Dy_{\theta^+}]]$ in (9) defines one and only one y_{θ^+} so that y_{θ^+} is well defined. This follows from the fact (showed next) that $C_{\theta+} = C_\theta + \tilde{\theta}(P^{-1}(I - MD)^{-1}M - C_0)$ is lower triangular (i.e., when such an operator is applied to a vector in $\mathbb{R}^N$, the ith element of output only depends on the first i elements of input), so that existence and uniqueness of $y_{\theta+}$ follows by recursively solving (9).

The fact that $C_{\theta+}$ is lower triangular follows from the following reasoning: $I - MD$ is a lower triangular matrix, so that $(I - MD)^{-1}$ is a lower triangular matrix too and operator $(I - MD)^{-1}$ is lower triangular. M and C_θ are lower triangular operators. The fact that P^{-1} is lower triangular follows from Assumption A.2 (see Remark 2). Since composition and sum of lower triangular operators is lower triangular, $C_{\theta+}$ is lower triangular. $*$

D. Filter Objective

Let

$$J_{\mathrm{VRFT}}(\theta^+) := \|F[C_{\theta+}[\tilde{e}]] - F[\tilde{u}]\|^2.$$

We want to select F so that

$$\left.\frac{\partial^2 J_{\mathrm{VRFT}}(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+} = \left.\frac{\partial^2 J(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+}. \tag{10}$$

Remark 10: If $C_{\theta+}[\tilde{e}]$ is linear in θ^+, under (10)$J_{\mathrm{VRFT}}(\theta^+)$ is the second-order expansion of $J(\theta^+)$. In general cases, matching up the second order derivatives produces well-tuned controllers whenever a controller exists in the selected class that is not "too far" from C^0. In loose words, this means that the controller class should not be too under-parameterized with respect to the control goal at hand. $*$

The following theorem specifies how F must be selected so that (10) is satisfied.

Theorem 2: If

$$F = (I - MD)\left(\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}\right) \tag{11}$$

then (10) holds.

Proof: See the Appendix. $*$

A few remarks are in order.

Remark 11 (About the Structure of the Filter): The filter in (11) is formed by two parts: i) $(\partial P[u]/\partial u)|_{\tilde{u}}$; and ii) $(I - MD)$. $(\partial P[u]/\partial u)|_{\tilde{u}}$ keeps into account the effect of input (used in the VRFT cost) on output (used in the control objective cost). Term ii) calls for a bit of more-in-depth analysis. To make things more concrete, suppose that M is given by a transfer function $M(z^{-1})$ (see Sections II-H) and let $L(z^{-1}) = M(z^{-1})/(1 - M(z^{-1}))$ be the corresponding open-loop transfer function. It is then easy to see that $(I - MD)$ is a linear operator whose magnitude frequency response is small at frequencies where $M(z^{-1}) \sim 1$, i.e., where $L(z^{-1})$ is large. Thus, $(I - MD)$ in (11) de-emphasizes those frequencies where $L(z^{-1})$ is large and this can be interpreted as follows: The closed-loop J cost is little sensitive to errors where $L(z^{-1})$ is large; on the other hand, J_{VRFT} is an open-loop cost that does not automatically incorporate this effect. Then, the term $(I - MD)$ is forced in artificially through the filter in order to level out such a difference between J and J_{VRFT}.

Typically, the magnitude of $M(z^{-1})$ drops off over the high frequency range so that the main feature of $(I - MD)$ is that of giving little emphasis to the low frequency range where $M(z^{-1}) \sim 1$. $*$

Remark 12 (Direct Versus Indirect): The term $(\partial P[u]/\partial u)|_{\tilde{u}}$ appearing in the filter expression has to be estimated from data. One thing that should be noted is that this is the incremental linear operator around the actual input $\tilde{u}$ (so that in order to equalize the second order derivatives around the ideal controller one has in fact to linearize around the actual trajectory at hand, a perhaps surprising result). System $(\partial P[u]/\partial u)|_{\tilde{u}}$ is linear and time-varying and it can be estimated, e.g., via forgetting factor identification techniques [20]–[22].

Since $(\partial P[u]/\partial u)|_{\tilde{u}}$ has to be estimated, strictly the VRFT method is not direct. On the other hand, one should note that the estimated expression of $(\partial P[u]/\partial u)|_{\tilde{u}}$ is not used for design, it is only used as a filter to process the data. As a consequence, a precise determination of $(\partial P[u]/\partial u)|_{\tilde{u}}$ is not required: imprecision in $(\partial P[u]/\partial u)|_{\tilde{u}}$ only reflects in that the second derivative of J_{VRFT} will not precisely match that of J around θ_0^+.

We finally remark that $(\partial P[u]/\partial u)|_{\tilde{u}}$ can be a rapidly varying system if the plant is highly nonlinear and signals change value at a fast rate. In these cases, resorting to forgetting factor identification methods can be inadequate. One can then resort to an approximate model, obtained by any means, e.g., first principles. Moreover, it is our practical experience that in many applications neglecting the $(\partial P[u]/\partial u)|_{\tilde{u}}$ part of the filter (11) and only implementing the $(I - MD)$ part still produces acceptable results. $*$

Remark 13 (Comparing (11) With Filter for Linear VRFT): It is interesting to compare filter (11) with the filter in (10) of [1] valid for the linear case. In [1], the filter goal was that of matching up a model reference complementary sensitivity cost (with no concern for specific reference signals) with the VRFT cost. The filter turned out to be (in the linear and stationary notations of [1]): $F = (1 - M)M(1/\Phi_{\tilde{u}}^{1/2})$, where $\Phi_{\tilde{u}}^{1/2}$ is a spectral factorization of the spectral density $\Phi_{\tilde{u}}$ of $\tilde{u}$, i.e., $\left|\Phi_{\tilde{u}}^{1/2}\right|^2 = \Phi_{\tilde{u}}$. In the nonlinear set-up of this paper, we consider a model reference cost involving the reference sinal $\tilde{r} = M^{-1}[\tilde{y}] = M^{-1}P[\tilde{u}]$. Again in the notations of [1], this corresponds to adding $\Phi_{\tilde{r}}^{1/2} = M^{-1}P\Phi_{\tilde{u}}^{1/2}$ to the model reference cost, so that the filter matching up the two costs now becomes: $(1 - M)M\left(1/\Phi_{\tilde{u}}^{1/2}\right)M^{-1}P\Phi_{\tilde{u}}^{1/2} = (1 - M)P$. This is indeed the linear counterpart of our filter (11). $*$

Example 5 (Example 4 Continued): We have

$$\begin{aligned} C_\theta &\text{ given by :} u(t) = \theta e(t)^{1/3};\\ C^0 &\text{ given by :} u(t) = e(t)^{1/3};\\ C_{\theta+} &\text{ given by :} u(t) = \theta e(t)^{1/3} + \tilde{\theta} e(t)^{1/3}. \end{aligned}$$

So, letting $\theta_{eq} := \theta + \tilde{\theta}$, $C_{\theta+}$ is given by $u(t) = \theta_{eq} e(t)^{1/3}$ and $J(\theta_{eq}) = 5 - 14\theta_{eq}^3 + 14\theta_{eq}^6 - 6\theta_{eq}^9 + \theta_{eq}^{12}$, $J_{\mathrm{VRFT}}(\theta_{eq}) = 2 - 4\theta_{eq} + 2\theta_{eq}^2$ (compare with Example 3). Computing second derivatives yields

$$\left.\frac{\partial^2 J(\theta_{eq})}{\partial \theta_{eq}^2}\right|_1 = -84\theta_{eq} + 420\theta_{eq}^4 - 432\theta_{eq}^7 + 132\theta_{eq}^{10}\Big|_1 = 36$$

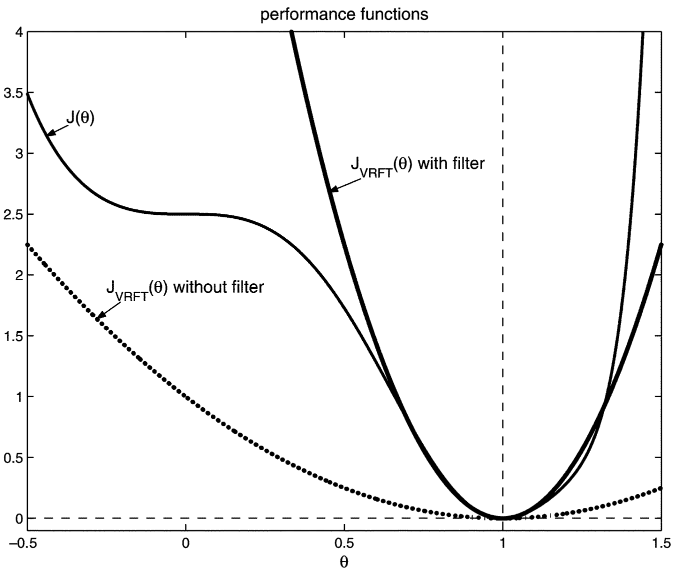

Fig. 3. $J_{\mathrm{VRFT}}(\theta_{eq})$ with and without F for $\theta_{eq} = \theta$.

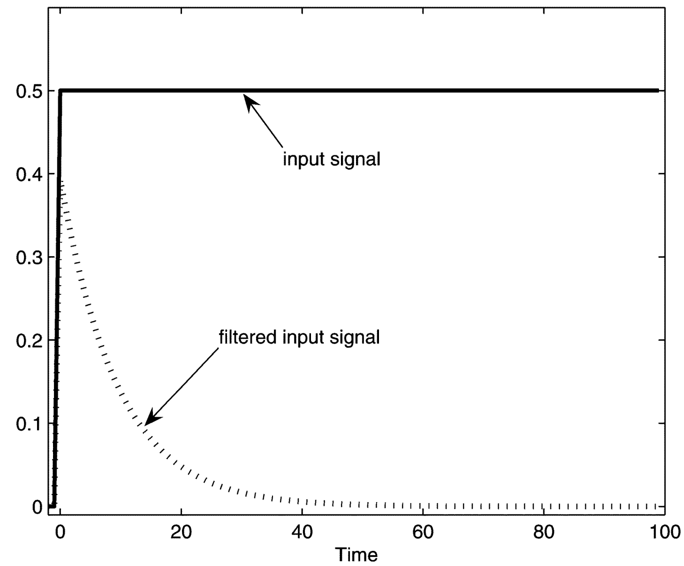

Fig. 4. Plant input $\tilde{u}$ and filtered input $F[\tilde{u}]$.

while, without filter F

$$\left.\frac{\partial^2 J_{\mathrm{VRFT}}(\theta_{eq})}{\partial \theta_{eq}^2}\right|_1 = 4|_1 = 4.$$

Instead, including filter $F = (I - MD)\left(\partial P[u]/\partial u|_{\tilde{u}}\right) = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 3\tilde{u}(0)^2 & 0 \\ 3\tilde{u}(0)^2 & 3\tilde{u}(1)^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 3 & 3 \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$ gives

$$\begin{aligned} J_{\mathrm{VRFT}}(\theta_{eq}) &= (3\theta_{eq} \cdot 1^{1/3} - 3)^2 + (3\theta_{eq} \cdot 1^{1/3} - 3)^2 \\ &= 18 - 36\theta_{eq} + 18\theta_{eq}^2 \end{aligned}$$

with the second-order derivative

$$\left.\frac{\partial^2 J_{\mathrm{VRFT}}(\theta_{eq})}{\partial \theta_{eq}^2}\right|_1 = 36|_1 = 36 = \left.\frac{\partial^2 J(\theta_{eq})}{\partial \theta_{eq}^2}\right|_1.$$

For a visual comparison, $J_{\mathrm{VRFT}}(\theta_{eq})$ with and without filter are displayed in Fig. 3. $*$

E. An Example With an Under-Parameterized Controller Class

We here present an example where the role of prefiltering is discussed in connection with an under-parameterized controller class (the most realistic setup in practice).

Example 6: Consider the nonlinear plant

$$y(t) = 0.9y(t-1) + \tanh(u(t-1)) \tag{12}$$

where $\tanh(x) = (e^x - e^{-x})/(e^x + e^{-x})$ is the hyperbolic tangent, and suppose that the reference model is $y(t) = r(t-1)$. As is easily seen, the ideal controller is

$$u(t) = \tanh^{-1}(v(t)) \tag{13}$$

where

$$v(t) = v(t-1) + e(t) - 0.9e(t-1) \tag{14}$$

with initial conditions $v(-1) = 0$ and $e(-1) = 0$ [just solve recursively (12)–(14) to see that $y(t) = r(t-1)$]. Consider now a PI (proportional-integral) controller class C_θ

$$u(t) = \left(\frac{0.2}{1 - z^{-1}} + \theta\right) e(t)$$

that is $u(t) = u(t-1) + (0.2 + \theta)e(t) - \theta e(t-1)$, where the integral coefficient is set to 0.2 and the proportional coefficient θ has to be selected (we have fixed the integral coefficient to facilitate the visualization of the results).

In order to design the controller, plant P has been fed with a step input $\tilde{u}$ of amplitude 0.5 (see Fig. 4). Using the corresponding output $\tilde{y}$, the performance indices $J(\theta^+)$ and $J_{\mathrm{VRFT}}(\theta^+)$ (with and without filtering) have been computed and their contour plots are displayed in Fig. 5(a)–(c). By inspecting these three figures, the following observations can be drawn.

- All three two-dimensional indexes share the same minimum point ($\theta = 0$, and $\tilde{\theta} = 1$), corresponding to the ideal controller. This confirms that, if the controller class is not under-parameterized, the VRFT approach provides the exact solution, regardless of data-filtering.
- The shape of the contour plot of $J(\theta^+)$ reveals that this performance index is not a quadratic function of θ^+. Instead, $J_{\mathrm{VRFT}}(\theta^+)$ is quadratic, since $C_{\theta+}$ linearly depends on θ^+.
- Even if $J(\theta^+)$ and the nonfiltered $J_{\mathrm{VRFT}}(\theta^+)$ [figures (A) and (B)] share the same minimum, it is apparent that their shape is completely different. Instead, notice that the effect of using the filter F is to rotating and warping $J_{\mathrm{VRFT}}(\theta^+)$ [figure (C)] in order to make it equal to the second-order expansion of $J(\theta^+)$ around its minimum.
- The PI performance indices (which depend on the parameter θ only) can be obtained by cutting the original

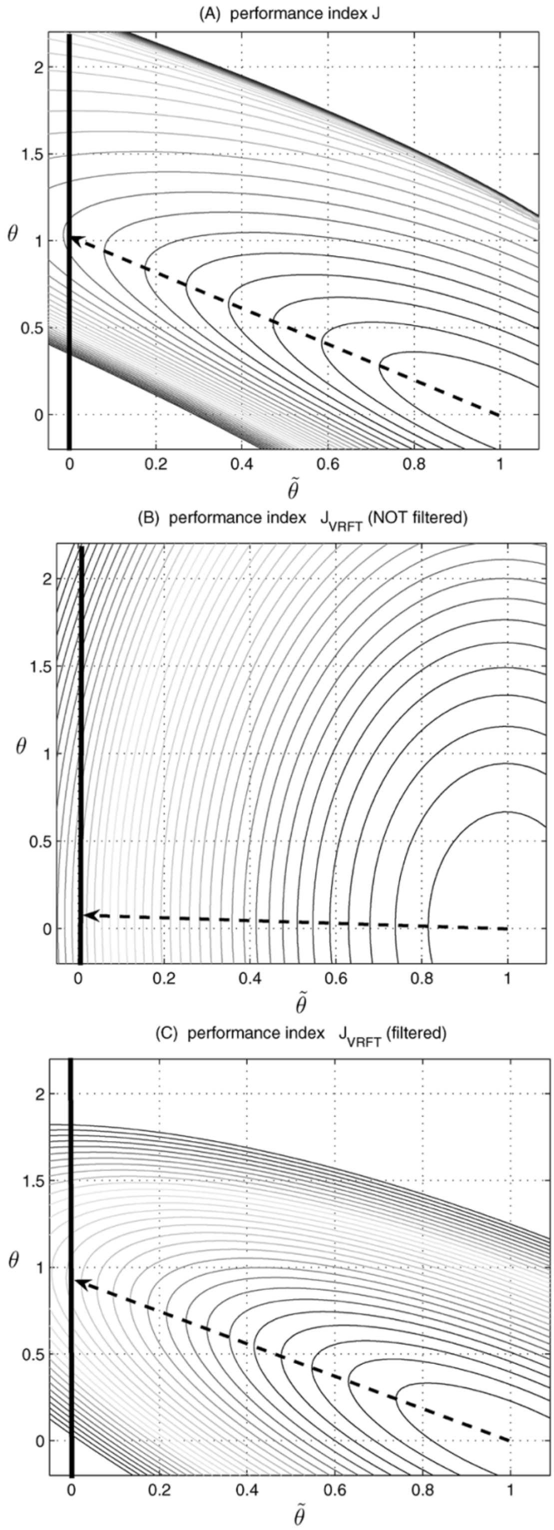

Fig. 5. Contour plots of performance indexes. (A) $J(\theta^{+})$. (B) $J_{\mathrm{VRFT}}(\theta^{+})$ (not filtered). (C) $J_{\mathrm{VRFT}}(\theta^{+})$ (filtered).

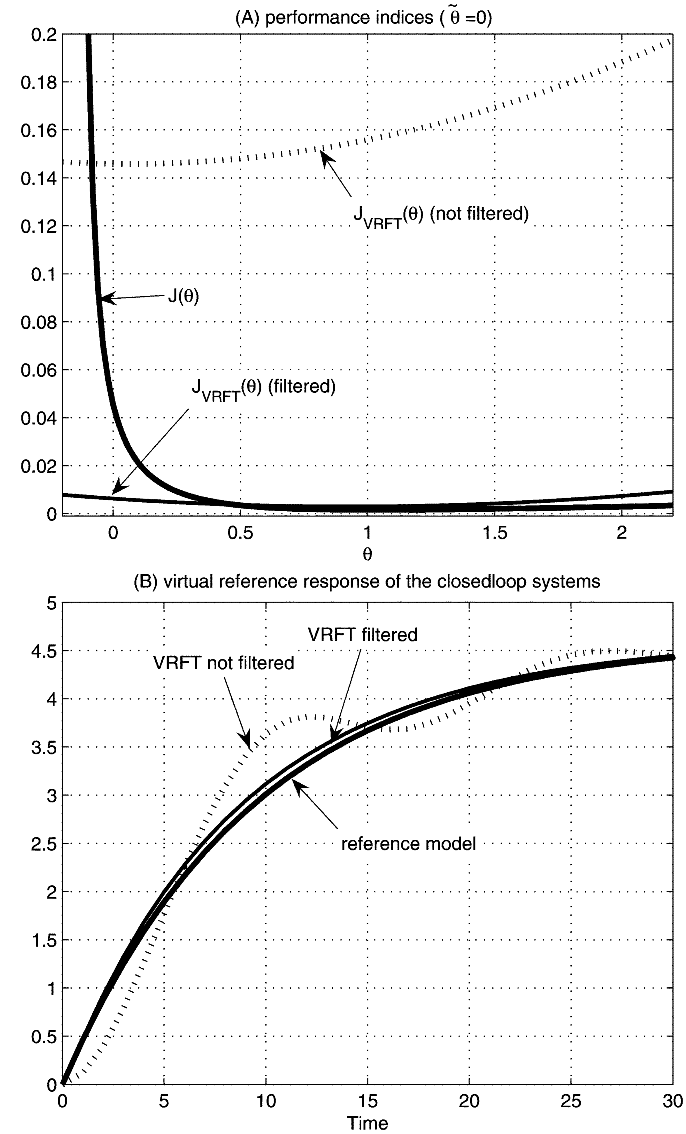

Fig. 6. (A) Reduced-order performance indices. (B) Closed-loop response to the virtual reference input $\tilde{r}$—plant (12).

two-dimensional performance indices with the hyperplane $\tilde{\theta} = 0$. In the figures, the arrows show how the global minimizer $\theta_0^{+} = [0 \quad 1]^T$ moves to the minimizer of the reduced-order performance indexes. Apparently, thanks to the fact that the filtered $J_{\mathrm{VRFT}}(\theta^{+})$ is the second-order expansion of $J(\theta^{+})$, the minimizer of $J(\theta)$ and the minimizer of the filtered $J_{\mathrm{VRFT}}(\theta)$ are very close; on the contrary, the minimizer of the nonfiltered $J_{\mathrm{VRFT}}(\theta)$ drifts away from the minimizer of $J(\theta)$. This fact can be even better appreciated from Fig. 6(a), where $J(\theta)$ and $J_{\mathrm{VRFT}}(\theta)$ (filtered and nonfiltered) are displayed. It is apparent that the minimizer of $J(\theta)$ (in $\theta = 1.02$) is close to the minimizer of the filtered $J_{\mathrm{VRFT}}(\theta)$ (in $\theta = 0.92$), whereas the minimizer of the nonfiltered $J_{\mathrm{VRFT}}(\theta)$ (in $\theta = 0.06$) is far from the correct solution.

- Fig. 6(b) shows the closed-loop output when the reference signal is $\tilde{r}$ and the controller has been designed with the filtered and nonfiltered J_{VRFT}.
- As an additional remark, the reader may be interested in noting the effect of the filter F on $\tilde{u}$ (see Fig. 4): The effect of F looks pretty much like a high-pass filtering where the transient phase is emphasized and the steady-state part of the signal is almost annihilated. This is in line with Remark 11.

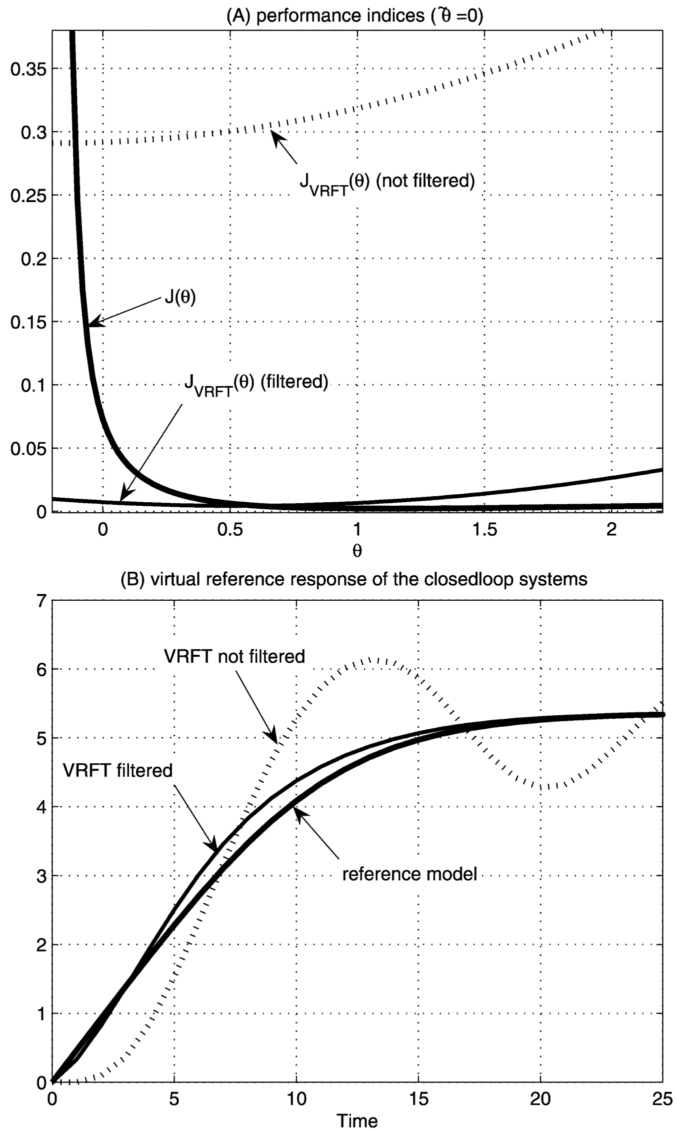

Fig. 7. (A) Reduced-order performance indexes. (B) Closed-loop response to the virtual reference input $\tilde{r}$—plant (15).

Plant (12) exhibits a linear dependence on $y(t-1)$. This choice allowed us to present the simulation results in a more explicit way, since we were able to represent the ideal controller in the recursive form (13) and (14).

We next consider a variation of plant (12) with a nonlinear dependence on $y(t-1)$

$$y(t) = 10\tanh(0.1y(t-1)) + \tanh(u(t-1)). \qquad (15)$$

A repeat of the experiment as for plant (12) gave the results of Fig. 7(a) and (b). The interpretation is pretty much akin to that for (12). What the reader should note is that the ideal controller is here a truly complicated system, not amenable of a simple recursive representation as in (13) and (14). Nevertheless, computation of $J_{\mathrm{VRFT}}(\theta)$ is possible without knowledge of the ideal controller. $*$

V. Implementation Issues

A. Filter

When M is given by (4), the $(I-MD): v(1{:}N) \to z(1{:}N)$ part of the filter in (11) can be implemented recursively as

$$\begin{aligned} z(t) = &-a_1 z(t-1) - \cdots - a_{n_{My}} z(t-n_{My}) \\ &- (b_1 v(t-1) + \cdots + b_{n_{Mr}} v(t-n_{Mr})) \\ &+ (v(t) + a_1 v(t-1) + \cdots + a_{n_{My}} v(t-n_{My})) \end{aligned}$$

with initial conditions $= 0$. This is a linear invariant system with transfer function $1 - M(z^{-1})$.

Instead, $(\partial P[u]/\partial u)|_{\tilde{u}} : u(0{:}N-1) \to v(1{:}N)$ can be implemented recursively as follows:

$$\begin{aligned} v(t) = &\left(\left.\frac{\partial p}{\partial y(t-1)}\right|_{\tilde{u},\tilde{y}} \right) v(t-1) + \cdots \\ &+ \left(\left.\frac{\partial p}{\partial y(t-n_{Py})}\right|_{\tilde{u},\tilde{y}} \right) v(t-n_{Py}) \\ &+ \left(\left.\frac{\partial p}{\partial u(t-1)}\right|_{\tilde{u},\tilde{y}} \right) u(t-1) + \cdots \\ &+ \left(\left.\frac{\partial p}{\partial u(t-n_{Pu})}\right|_{\tilde{u},\tilde{y}} \right) u(t-n_{Pu}) \end{aligned}$$

again with initial conditions $= 0$. This is a linear time-variant system whose implementation, as noted in the previous section, requires the identification of the plant in a neighborhood of the trajectory obtained for $u = \tilde{u}$.

Implementation of F is obtained by applying the previous equations in cascade.

B. Quadratic Optimization

If $C_\theta[\tilde{e}]$ is linear in θ, then $J_{\mathrm{VRFT}}(\theta)$ is quadratic, and therefore easy to minimize. This happens, e.g., for the controller class

$$\begin{aligned} u(t) = \theta_1 s_1(e(t), \ldots, e(t-n_{Ce})) + \cdots \\ + \theta_{n_\theta} s_{n_\theta}(e(t), \ldots, e(t-n_{Ce})) \end{aligned}$$

where $s_1, \ldots, s_{n_\theta}$ are given (possibly nonlinear) functions. In this case

$$F[C_\theta[\tilde{e}]] = \theta_1 F[C_1[\tilde{e}]] + \cdots + \theta_{n_\theta} F[C_{n_\theta}[\tilde{e}]]$$

and the quantities $C_i[\tilde{e}]$ can be precomputed as $s_i(\tilde{e}(t), \ldots, \tilde{e}(t-n_{Ce}))$ and then filtered through F, so that the quadratic cost can be constructed and minimized without any iterative implementation of a minimization procedure.

More specifically, letting

$$F[C[\tilde{e}]] = [\,F[C_1[\tilde{e}]] \quad F[C_2[\tilde{e}]] \quad \cdots \quad F[C_{n_\theta}[\tilde{e}]]\,]$$

($F[C[\tilde{e}]]$ is a matrix of dimension $N \times n_\theta$) and

$$\theta = [\,\theta_1 \quad \theta_2 \quad \cdots \quad \theta_{n_\theta}\,]^T$$

$F[C_\theta[\tilde{e}]]$ can be rewritten as

$$F[C_\theta[\tilde{e}]] = F[C[\tilde{e}]]\theta$$

so that

$$J_{\mathrm{VRFT}}(\theta) = \|F[C[\tilde{e}]]\theta - F[\tilde{u}]\|^2. \tag{16}$$

The minimizer of (16) is obtained by solving the linear equation

$$F[C[\tilde{e}]]^T(F[C[\tilde{e}]]\theta - F[\tilde{u}]) = 0. \tag{17}$$

Equation (17) has a simple (and well-known) interpretation: The solution $\hat{\theta}$ is the parameter vector which makes the regressor matrix $F[C[\tilde{e}]]$ (which represents the input data vector of the model) uncorrelated with the model residual error $(F[C[\tilde{e}]]\hat{\theta} - F[\tilde{u}])$. This interpretation of $\hat{\theta}$ will be further elaborated on in Section VI, where the issue of the treatment of noise will be considered.

Finally, note that a particular instance of the previously described setting is obtained when $s_i(e(t), \dots, e(t - n_{Ce}))$ are achieved by expanding given transfer functions $\beta_i(z^{-1})$ and by truncating them after N terms. In this case, the $s_i(e(t), \dots, e(t - n_{Ce}))$ can be recursively computed.

C. Treatment of Noise

So far, we have assumed that the input/output measured data $\tilde{u}$ and $\tilde{y}$ (and, correspondingly, the virtual reference $\tilde{r}$ and the virtual error $\tilde{e}$) are noise-free. This assumption has helped to avoid a notational burden and to keep the presentation of the method tidy and clean.

In practice, the noise-free assumption is realistic only when the input/output data set is generated by a software simulator of the plant. However, when we have on-field measurements collected from the real plant P, we must assume that the measured system output—say $\tilde{y}_n$—is affected by noise, namely

$$\tilde{y}_n = \tilde{y} + \tilde{n}$$

where $\tilde{y}$ is the noise-free output, and $\tilde{n}$ is the noise. Correspondingly (with obvious meaning of the symbols) we can compute a "noisy virtual reference" $\tilde{r}_n$, and a "noisy virtual error" $\tilde{e}_n$.

Consider now the case of a linearly parameterized controller, as outlined at the end of the previous section. In this case, the "noisy" regressor matrix $F[C[\tilde{e}_n]]$ can be rewritten as

$$F[C[\tilde{e}_n]] = F[C[\tilde{e}]] + \Psi \tag{18}$$

where $F[C[\tilde{e}]]$ is the noise-free regressor matrix, and $\Psi := F[C[\tilde{e}_n]] - F[C[\tilde{e}]]$ is the noise matrix (obviously, $\Psi = 0$ if $\tilde{n} = 0$).

If the noisy output signal $\tilde{y}_n$ is used, the VRFT method reduces to the following equation:

$$F[C[\tilde{e}_n]]^T(F[C[\tilde{e}_n]]\theta - F[\tilde{u}]) = 0 \tag{19}$$

and the solution, say $\hat{\theta}_n$, is in general different from $\hat{\theta}$ if $\Psi \neq 0$.

In order to overcome this problem, a simple and effective approach is to resort to instrumental variable (IV) techniques, [22]. This can be done in many different manners and we here only outline a possible way. Let us assume that it is possible to make a second (noisy) experiment on the plant with the same input signal $\tilde{u}$; the corresponding output—say $\tilde{y}_n^*$—will differ from $\tilde{y}_n$ for the noise part only, namely

$$\tilde{y}_n^* = \tilde{y} + \tilde{n}^*.$$

Since $\tilde{n}^*$ and $\tilde{n}$ correspond to two different experiments, we can assume that they are independent. Using $\tilde{y}_n^*$, the IV regressor matrix $F[C[\tilde{e}_n^*]]$ can be constructed and, similarly to (18), we write

$$F[C[\tilde{e}_n^*]] = F[C[\tilde{e}]] + \Psi^*.$$

Then, (19) is replaced by its IV counterpart

$$F[C[\tilde{e}_n^*]]^T(F[C[\tilde{e}_n]]\theta^* - F[\tilde{u}]) = 0.$$

The analysis of this equation to show that the solution $\hat{\theta}^*$ is such that $\hat{\theta}^* \sim \hat{\theta}$ (and that $\hat{\theta}^* = \hat{\theta}$ when $N \to \infty$) follows standard routes in IV analysis.

In case of nonlinearly parameterized controllers, treatment of noise is more complex and it is a problem worthy of further consideration.

VI. Multipass Procedure for Refining the Experimental Conditions in Which Data are Collected

In Sections III–V, the VRFT algorithm has been described for the tuning of a controller when the reference signal is $\tilde{r} := M^{-1}[\tilde{y}]$, where $\tilde{y}$ is the actually measured plant output. As we have already noticed, this one-shot tuning of the controller oftentimes suffices since it makes the closed-loop system behave similarly to M for a reasonable large class of reference signals, including that of interest. In this section, we further discuss this issue: When VRFT does not directly generate a suitable controller for the *a-priori* specified reference $\tilde{r}$ (which is, in general, different from $M^{-1}[\tilde{y}]$), one can then resort to a multipass procedure, as described here.

Suppose that at pass k a controller C_k has been designed. To move one pass ahead and design C_{k+1} we proceed as follows: First, reference $\tilde{r}$ is injected in the control system with C_k in the loop and the corresponding output y_k is measured; y_k is then used to determine the next virtual reference signal, $\tilde{r}_{k+1} = M^{-1}[y_k]$, and VRFT is used with $\tilde{r}_{k+1}$ to design C_{k+1}.

It has been observed in simulation examples that this procedure produces virtual references $\tilde{r}_k$ that rapidly approach $\tilde{r}$, so that, after few passes, VRFT generates a controller well-suited for the reference signal $\tilde{r}$ of interest. A complete analysis of this observed behavior goes beyond the scopes of this paper; however, a simplified analysis is presented with the objective of providing insight in why such a behavior can be expected.

Let H_k be the closed-loop operator from r to y at step k, and assume no noise is present so that $y_k = H_k[\tilde{r}]$. We assume that VRFT delivers a controller C_{k+1} which is perfectly performing in correspondence of the used virtual reference $M^{-1}H_k[\tilde{r}]$ (notation $M^{-1}H_k[\tilde{r}]$ means that operator $M^{-1}H_k$—obtained by applying in cascade H_k and then M^{-1}—is applied to $\tilde{r}$. Similar notations are in use elsewhere), namely

$$H_{k+1}M^{-1}H_k[\tilde{r}] = H_k[\tilde{r}]. \tag{20}$$

Also, let $\Delta_k := H_kM^{-1} - I$ and $\delta_k := H_k[\tilde{r}] - M[\tilde{r}]$.

To clarify ideas, let us first focus on a linear (single-input–single-output) setup and assume that H_k is invertible. In this case

$$\begin{aligned}\delta_{k+1} &= H_{k+1}\tilde{r} - M\tilde{r} = MH_k^{-1}H_kM^{-1}(H_{k+1}\tilde{r} - M\tilde{r})\\ &= MH_k^{-1}(H_kM^{-1}H_{k+1}\tilde{r} - H_k\tilde{r})\\ &= MH_k^{-1}(H_{k+1}M^{-1}H_k\tilde{r} - H_k\tilde{r})\\ &= \text{[using (20)]} = MH_k^{-1}(H_k\tilde{r} - H_k\tilde{r})\\ &= 0\end{aligned} \tag{21}$$

so that $H_{k+1}\tilde{r} = M\tilde{r}$ and the new virtual reference $M^{-1}H_{k+1}\tilde{r}$ is downright $\tilde{r}$ and convergence is obtained in one step. Moreover, (21) also gives

$$\begin{aligned}&\Delta_{k+1}M\tilde{r} - \Delta_{k+1}(M\tilde{r} + \delta_k)\\ &\quad= -\Delta_{k+1}\delta_k = -(H_{k+1}M^{-1} - I)(H_k\tilde{r} - M\tilde{r})\\ &\quad= -H_{k+1}M^{-1}H_k\tilde{r} + H_{k+1}\tilde{r} + H_k\tilde{r} - M\tilde{r}\\ &\quad= \text{[using (20)]} = H_{k+1}\tilde{r} - M\tilde{r}\\ &\quad= \text{[using (21)]} = 0.\end{aligned} \tag{22}$$

We interpret this result by saying that $\Delta_{k+1} := H_{k+1}M^{-1} - I$ has learned how to compensate for the mismatch δ_k.

In a nonlinear setting, (20) tells us that H_{k+1} behaves the same as the linear M for signal $M^{-1}H_k[\tilde{r}]$. Thus, if $M^{-1}H_k[\tilde{r}]$ is not too different from $\tilde{r}$ (this, in turn, requires that the first controller in the multipass procedure is not too poor) we can intuitively expect that H_{k+1} behaves not too nonlinearly in a neighborhood of $\tilde{r}$ and a condition similar to (22) will still hold.

Assumption

— **A.4** $\|\Delta_{k+1}[M\tilde{r}] - \Delta_{k+1}[M\tilde{r} + \delta_k]\| \le \rho\|\delta_k\|$, for some $\rho < 1$. *

Note that since ρ is only required to be < 1, this assumption is rather mild.

Under Assumption A.4, we have

$$\begin{aligned}&\|\delta_{k+1}\|\\ &\quad= \|\Delta_{k+1}[M\tilde{r}]\| = \text{[using (20)]}\\ &\quad= \|\Delta_{k+1}[M\tilde{r}] - H_{k+1}M^{-1}H_k[\tilde{r}] + H_k[\tilde{r}]\|\\ &\quad= \|\Delta_{k+1}[M\tilde{r}] - (H_{k+1}M^{-1} - I)[M\tilde{r} + H_k[\tilde{r}] - M\tilde{r}]\|\\ &\quad= \|\Delta_{k+1}[M\tilde{r}] - \Delta_{k+1}[M\tilde{r} + \delta_k]\| \le \rho\|\delta_k\|\end{aligned}$$

which entails that $\delta_k \to 0$ and therefore that the virtual reference $M^{-1}H_k[\tilde{r}]$ tends to $\tilde{r}$.

The convergence analysis of this section has similarities with repetitive processes; see, e.g., [23], [24], and the recent interesting paper [25]. Yet, the adaptive setup of this section looks outside the so-far developed repetitive processes framework and it would be interesting to see whether concepts from repetitive processes can be extended to VRFT.

Appendix
Proof of Theorem 2

We start by noting some elementary facts. First

$$\tilde{u} = C_{\theta_0^+}[\tilde{e}]. \tag{23}$$

In fact, $\tilde{u} = P^{-1}[\tilde{y}] = P^{-1}[(I - MD)^{-1}(I - MD)\tilde{y}] = P^{-1}[(I - MD)^{-1}(M\tilde{r} - MD\tilde{y})] = P^{-1}[(I - MD)^{-1}M\tilde{e}] = C^0[\tilde{e}] = C_{\theta_0^+}[\tilde{e}]$. Next

$$\tilde{y} = y_{\theta_0^+}. \tag{24}$$

In fact, $\tilde{y} = P[\tilde{u}] = \text{[using (23)]} = P[C_{\theta_0^+}[\tilde{e}]] = P[C_{\theta_0^+}[\tilde{r} - D\tilde{y}]]$ and $y_{\theta_0^+} = P[C_{\theta_0^+}[\tilde{r} - Dy_{\theta_0^+}]]$. Thus, both $\tilde{y}$ and $y_{\theta_0^+}$ correspond to $\tilde{r}$ in the r to y map given by $y = P[C_{\theta_0^+}[r - Dy]]$. Since in such a map to a r it corresponds only one y, we conclude that $\tilde{y} = y_{\theta_0^+}$, that is (24). From (24), we also immediately have

$$\tilde{r} - Dy_{\theta_0^+} = \tilde{e}. \tag{25}$$

Finally

$$\tilde{u} = C_{\theta_0^+}[\tilde{r} - Dy_{\theta_0^+}]. \tag{26}$$

In fact, $\tilde{u} = \text{[using (23)]} = C_{\theta_0^+}[\tilde{e}] = \text{[using (25)]} = C_{\theta_0^+}[\tilde{r} - Dy_{\theta_0^+}]$.

Now, letting $x_{\theta^+} := F[C_{\theta^+}[\tilde{e}]] - F[\tilde{u}]$ and $w_{\theta^+} := y_{\theta^+} - \tilde{y}$, we can rewrite the costs as

$$J_{\mathrm{VRFT}}(\theta^+) = \|x_{\theta^+}\|^2 \quad J(\theta^+) = \|w_{\theta^+}\|^2.$$

— Notation: By $(\partial x_{\theta^+}/\partial\theta^+)$ it is meant the matrix whose (i,j)th element (i = row; j = column) is $(\partial x_{\theta^+}(i)/\partial\theta_j^+)$ (so, different columns correspond to derivatives with respect to different parameters, and down the columns we have time evolution). This applies similarly to all other signals.

The derivatives of J_{VRFT} and J can be expressed as

$$\begin{aligned}\frac{\partial J_{\mathrm{VRFT}}(\theta^+)}{\partial\theta^+} &= \frac{\partial x_{\theta^+}^T x_{\theta^+}}{\partial\theta^+}\\ &= 2x_{\theta^+}^T\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right) \quad \text{(this is a row)}\\ \frac{\partial J(\theta^+)}{\partial\theta^+} &= \frac{\partial x_{\theta^+}^T w_{\theta^+}}{\partial\theta^+}\\ &= 2w_{\theta^+}^T\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right) \quad \text{(this is a row)}\\ \frac{\partial^2 J_{\mathrm{VRFT}}(\theta^+)}{\partial\theta^{+2}} &= 2x_{\theta^+}^T\left(\frac{\partial^2 x_{\theta^+}}{\partial\theta^{+2}}\right)\\ &\quad + 2\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right)^T\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right)\\ \frac{\partial^2 J(\theta^+)}{\partial\theta^{+2}} &= 2x_{\theta^+}^T\left(\frac{\partial^2 w_{\theta^+}}{\partial\theta^{+2}}\right)\\ &\quad + 2\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right)^T\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right).\end{aligned}$$

In the last two expressions, $(\partial^2 x_{\theta^+}/\partial\theta^{+2})$ and $(\partial^2 w_{\theta^+}/\partial\theta^{+2})$ should be defined with some precision. However, we omit to do so since we will only compute the second-order derivatives for $\theta^+ = \theta_0^+$, where the multiplying terms $x_{\theta^+}^T$ and $w_{\theta^+}^T$ are zero [see (23) and (24)], so that the product is zero.

Using (23) and (24) in the previous expressions yields

$$J_{\mathrm{VRFT}}(\theta^+)\big|_{\theta_0^+} = J(\theta^+)\big|_{\theta_0^+} = 0;$$
$$\left.\frac{\partial J_{\mathrm{VRFT}}(\theta^+)}{\partial \theta^+}\right|_{\theta_0^+} = \left.\frac{\partial J(\theta^+)}{\partial \theta^+}\right|_{\theta_0^+} = 0;$$
$$\left.\frac{\partial^2 J_{\mathrm{VRFT}}(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+} = 2\left(\left.\frac{\partial x_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+}\right)^T\left(\left.\frac{\partial x_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+}\right);$$
$$\left.\frac{\partial^2 J(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+} = 2\left(\left.\frac{\partial w_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+}\right)^T\left(\left.\frac{\partial w_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+}\right).$$

Thus, if

$$\left.\frac{\partial x_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} = \left.\frac{\partial w_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \tag{27}$$

then the filter objective (10) is met. F is now selected so that (27) holds.

— Notation: By $(\partial P[u]/\partial u)$ it is meant the matrix whose (i,j)th element (i = row; j = column) is $\partial P[u](i)/\partial u(j-1)$ (so, different columns correspond to derivatives with respect to input at different time instants, and down the columns we have time evolution). This applies similarly to all other systems.

We compute explicit expressions for the two sides of (27) and then equalize them. As for the left-hand side, we have (using linearity of F)

$$\left.\frac{\partial x_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} = \left.\frac{\partial F[C_{\theta^+}[\tilde{e}]]}{\partial \theta^+}\right|_{\theta_0^+} = F\left(\left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+}\right). \tag{28}$$

To compute the right-hand side, note first that [since $C_{\theta_0^+}[\tilde{e}] = \tilde{u}$, see (23)]

$$\begin{aligned}\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{e}} &= \left.\frac{\partial P[C_{\theta_0^+}[e]]}{\partial e}\right|_{\tilde{e}} \\ &= \left.\frac{\partial (I-MD)^{-1}M[e]}{\partial e}\right|_{\tilde{e}} \\ &= (I-MD)^{-1}M. \end{aligned}\tag{29}$$

We then have

$$\begin{aligned}\left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} &= \left.\frac{\partial P[C_{\theta^+}[\tilde{r} - Dy_{\theta^+}]]}{\partial \theta^+}\right|_{\theta_0^+} \\ &= \left.\frac{\partial P[u]}{\partial u}\right|_{C_{\theta_0^+}[\tilde{r} - Dy_{\theta_0^+}]} \\ &\quad \times \left\{ \left.\frac{\partial C_{\theta^+}[\tilde{r} - Dy_{\theta_0^+}]}{\partial \theta^+}\right|_{\theta_0^+} \right. \\ &\quad \left. - \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{r} - Dy_{\theta_0^+}} \left.\frac{\partial Dy_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \right\}.\end{aligned}$$

Now, using (25) and (26)

$$\begin{aligned}\left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} = \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left\{ \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+} \right. \\ \left. - \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{e}} \left.\frac{\partial Dy_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \right\}\end{aligned}$$

and, hence

$$\begin{aligned}\left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} + \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{e}} \left.\frac{\partial Dy_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \\ = \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+}.\end{aligned}$$

Substituting (29), we obtain

$$\begin{aligned}\left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} + (I-MD)^{-1}M \left.\frac{\partial Dy_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \\ = \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+}\end{aligned}$$

from which, by multiplying by $(I - MD)$

$$\begin{aligned}\left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} - MD \left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} + M \left.\frac{\partial Dy_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \\ = (I-MD) \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+}.\end{aligned}$$

Thus

$$\begin{aligned}\left.\frac{\partial w_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} &= \left.\frac{\partial y_{\theta^+}}{\partial \theta^+}\right|_{\theta_0^+} \\ &= (I-MD) \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\right|_{\theta_0^+}. \end{aligned}\tag{30}$$

Comparing (28) with (30), we finally obtain

$$F = (I - MD)\left(\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}\right).$$

References

[1] M. C. Campi, A. Lecchini, and S. Savaresi, "Virtual reference feedback tuning: A direct method for the design of feedback controllers," *Automatica*, vol. 38, pp. 1337–1346, 2002.

[2] A. Lecchini, M. C. Campi, and S. Savaresi, "Virtual reference feedback tuning for two degree of freedom controllers," *Int. J. Adapt. Control Signal Process.*, vol. 16, pp. 355–371, 2002.

[3] M. C. Campi, A. Lecchini, and S. Savaresi, "An application of the virtual reference feedback tuning method to a benchmark problem," *Eur. J. Control*, vol. 9, pp. 66–76, 2003.

[4] S. Savaresi and G. Guardabassi, "Approximate I/O feedback linearization of discrete-time nonlinear systems via virtual input direct design," *Automatica*, vol. 34, pp. 715–722, 1998.

[5] G. Guardabassi and S. Savaresi, "Approximate linearization via feedback: An overview," *Automatica*, vol. 37, pp. 1–15, 2001.

[6] M. Nakamoto, "An application of the virtual reference feedback tuning, for an mimo process," in *Proc. SICE Annu. Conf.*, 2004, pp. 587–592.

[7] D. Kostic, "Data-driven robot motion control design," Ph.D. dissertation, Tech. Univ. Eindhoven, Eindhoven, The Netherlands, 2004.

[8] F. Previdi, T. Schauer, S. Savaresi, and K. Hunt, "Data-driven control design for neuroprotheses: A virtual reference feedback tuning (vrft) approach," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 1, pp. 176–182, Jan. 2004.

[9] F. Previdi, M. Ferrarin, S. Savaresi, and S. Bittanti, “Closed-loop control of fes supported standing up and sitting down using virtual reference feedback tuning,” *Control Eng. Prac.*, vol. 13, pp. 1173–1182, 2005.

[10] A. Sala and A. Esparza, “Virtual reference feedback tuning in restricted complexity controller design of nonminimum phase systems,” in *Proc. 16th IFAC World Congr.*, vol. Mo-M03-TP/20, 2005, pp. 1–6.

[11] J. Ziegler and N. Nichols, “Optimum settings for automatic controllers,” *ASME Trans.*, vol. 64, pp. 759–768, 1942.

[12] K. Astrom and T. Hagglund, *PID Controllers: Theory, Design, and Tuning*. Research Triangle Park, NC: ISA, 1995.

[13] H. Hjalmarsson, M. Gevers, S. Gunnarson, and O. Lequin, “Iterative feedback tuning: Theory and applications,” *IEEE Control Syst. Mag.*, vol. 18, no. 4, pp. 26–41, Apr. 1998.

[14] H. Hjalmarsson, “Efficient tuning of linear multivariable controllers using iterative feedback tuning,” *Int. J. Adapt. Control Signal Process.*, vol. 13, pp. 553–572, 1999.

[15] ——, “Iterative feedback tuning—An overview,” *Int. J. Adapt. Control Signal Process.*, vol. 17, pp. 373–395, 2002.

[16] ——, “Control of nonlinear systems using iterative feedback tuning,” in *Proc. Amer. Control Conf.*, 1998, pp. 2083–2087.

[17] L. Gerencser, Z. Vago, and H. Hjalmarsson, “Randomized iterative feedback tuning,” in *Proc. 15th IFAC World Congr.*, vol. T-Th-A03, Barcelona, Spain, 2002, pp. 1–6.

[18] H. Jansson and H. Hjalmarsson, “Gradient approximations in iterative feedback tuning of multivariable processes,” *Int. J. Adapt. Control Signal Process.*, vol. 18, pp. 665–681, 2004.

[19] M. C. Campi and S. Savaresi, “Invertibility of nonlinear maps,” Univ. Brescia, Brescia, Italy, Int. Rep., 2004.

[20] T. Soderstrom and P. Stoica, *System Identification*. Upper Saddle River, NJ: Prentice-Hall, 1988.

[21] M. C. Campi, “On the convergence of minimum-variance directional-forgetting adaptive control schemes,” *Automatica*, vol. 28, pp. 221–225, 1991.

[22] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, 1999.

[23] E. Rogers and D. Owens, *Stability Analysis for Linear Repetitive Processes*, ser. Lecture notes in control and information sciences. New York: Springer-Verlag, 1992, vol. 175.

[24] D. Owens, N. Amann, E. Rogers, and M. French, “Analysis of linear iterative control schemes—A 2D systems/repetitive processes approach,” *Multidimen. Syst. Signal Process.*, vol. 11, pp. 125–127, 2000.

[25] B. Sulikowski, K. Galkowski, E. Rogers, and D. Owens, “Output feedback control of discrete linear repetitive processes,” *Automatica*, vol. 40, pp. 2167–2173, 2004.

Marco C. Campi was born in Tradate, Italy, on December 7, 1963. He received the Doctor degree in electronic engineering from the Politecnico di Milano, Milano, Italy, in 1988.

He is a Professor of Automatic Control at the University of Brescia, Brescia, Italy. From 1988 to 1989, he was a Research Assistant at the Department of Electrical Engineering of the Politecnico di Milano. From 1989 to 1992, he worked as a Researcher at the Centro di Teoria dei Sistemi of the National Research Council (CNR), Milano. Since 1992, he has been with the University of Brescia, Brescia, Italy. He has held visiting and teaching positions at many universities and institutions including the Australian National University, Canberra, Australia, the University of Illinois at Urbana-Champaign, the Center for Artificial Intelligence and Robotics, Bangalore, India, and the University of Melbourne, Australia. His research interests include: Adaptive and data-based control, system identification, robust convex optimization, robust control and estimation, and learning theory. His research activity has been conducted through years under many Italian and European projects. Currently, he is leader of the Brescia unit of the European IST project “Distributed control and stochastic analysis of hybrid systems supporting safety critical real-time systems design.”

Dr. Campi is an Associate Editor of *Automatica* and *Systems and Control Letters,* and a past Associate Editor of the *European Journal of Control*. He serves as Chair of the Technical Committee IFAC on Stochastic Systems (SS) and is a member of the Technical Committee IFAC on Modeling, Identification and Signal Processing (MISP). Moreover, he is a Distinguished Lecturer under the IEEE Control System Society (CSS) Program. His doctoral thesis was awarded the “Giorgio Quazza” prize as the best original thesis for the year 1988.

Sergio M. Savaresi was born in Manerbio, Italy, on September 21, 1968. He received the M.Sc. degree in electrical engineering from the Politecnico di Milano, Milano, Italy, in 1992, the Ph.D. degree in systems and control engineering in 1997, and the M.Sc. in applied mathematics from the Universita’ Cattolica, Brescia, Italy, in 2002.

In 1998, he was with McKinsey & Co., Milan, Italy. Since 1999, he has been with the Politecnico di Milano, where he is currently Associate Professor in Automatic Control. He has been a Visiting Researcher at Lund University, Lund, Sweden, at University of Twente, Enschede, The Netherlands, at Canberra National University, Canberra, Australia, and at the University of Minnesota, Minneapolis. His main interests are in the areas of vehicles control, automotive systems, data analysis and system identification, nonlinear control theory, and control applications.

Dr. Savaresi is an Associate Editor of the *European Journal of Control*.