Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005



# Virtual Reference Feedback Tuning for non-linear systems

Marco C. Campi and Sergio M. Savaresi

*Abstract*— **This paper introduces the VRFT - Virtual Reference Feedback Tuning - approach for controller tuning in a nonlinear set-up. 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, and represents a very appealing controller design methodology for many industrial control applications.**

## 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.

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 [2], [3] and the earlier reference [7]).

VRFT is a *data-based, direct, one-shot* controller design method ([2]). 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.

The VRFT method was originally proposed in a *linear framework* by the same authors of this paper, see [2], [3]. For linear plants, VRFT provided an alternative to most traditional methodologies, such as Ziegler and Nichols tuning method and alike, [12], [6]. 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.

A method alternative to VRFT - but based on an iterative gradient-descent approach, and more suited for a linear-setting - is IFT (Iterative Feedback Tuning, see e.g.[9], [8]).

The outline of the paper is as follows: Section II contains the control setting, while the VRFT method is presented in Section III. In Section IV the problem of designing an "optimal" filter is discussed.

## II. CONTROL SETTING

The control system we make reference to 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.

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, meeting 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.

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

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

where $p$ is a nonlinear function. Note that the $u$ to $y$ delay in $P$ is 1. Generalizing the results in this paper to a multiple delay setting presents no difficulties.

We assume that the plant is initialized with the initial conditions: $i.c. = y(0), \ldots, y(1-n_{Py}), u(-1), \ldots, u(1-n_{Pu})$; moreover, we assume that 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 equation (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:* 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 work is supported by MIUR under the project *New methods for Identification and Adaptive Control for Industrial Systems*

Marco Campi is with Dipartimento di Elettronica per l'Automazione - Università di Brescia, Via Branze 38, 25123 Brescia, Italy campi@ing.unibs.it

Sergio M. Savaresi is with Dipartimento di Elettronica ed Informazione - Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy savaresi@elet.polimi.it

This result illustrates a general fact: $P$ is a (nonlinear) lower triangular operator (i.e. the $i$-th element of the output depends on the first $j$-th, $j \leq i$, 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 $P[u_1(0{:}N-1), i.c.] \neq P[u_2(0{:}N-1), i.c.]$. $*$

Note that Assumption A.2 is an invertibility condition on map $P$. In [4], 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 $T \leq N-1$.

The controller is a nonlinear system described as

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

Similarly to $P$, we want to see the controller operating as follows: it is initialized with the initial conditions: $i.c. = u(-1), \dots, u(-n_{Cu}), e(-1), \dots, e(-n_{Ce})$, and it is fed by the error signal $e(0{:}N-1)$. Then, $C$ generates signal $u(0{:}N-1)$, which we write as $u(0{:}N-1) = C[u(0{:}N-1), i.c.]$.

In order to describe the feedback control system, the plant and controller equations 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.

In the sequel, we let: $i.c.$ of $P = 0$, $i.c.$ of $C = 0$ (more generally, one could assume non-zero initial conditions with some extra notational complications). 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: $y = P[C[r - Dy]]$.

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

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

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

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 $M$. Formally, the reference model $M$ is:

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

When a map is linear, we identify the map with the matrix that represents it. So, $M$ is also a $N \times N$ matrix and we write $y = Mr$ in place of $y = M[r]$.

**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.

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

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

which in the time domain corresponds to

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

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.

The control objective (to be minimized) is expressed as:

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

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

Finally, notice that, 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.

## III. THE VRFT APPROACH

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. 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 [5].

The batch of data is

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

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

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 $C_\theta$ so as to meet (5) for this reference signal.

Note that 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, 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.

$\tilde{r}$ 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 $\tilde{y} = M\tilde{r}$. $\tilde{r}$ 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.

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

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

where $F : \mathbb{R}^N \to \mathbb{R}^N$ is a filter to be chosen. The important fact is that $J_{VRFT}(\theta)$ 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 $\theta$) 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 the next Section IV.

*Example 2 (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 case, the control objective in (5) could be explicitly calculated as follows:

$$\begin{aligned} y_\theta(1) &= \theta^3 \\ y_\theta(2) &= 3\theta^3 - \theta^6 \end{aligned}$$

so that

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

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

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

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

so that

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

The important fact to note here is that $J_{VRFT}(\theta_0) = 0$, i.e. $\theta_0$ is also obtained by minimizing $J_{VRFT}(\theta)$, despite that $J_{VRFT}(\theta) \neq J(\theta)$ (see Figure 1 for a graphical comparison of $J(\theta)$ and $J_{VRFT}(\theta)$).

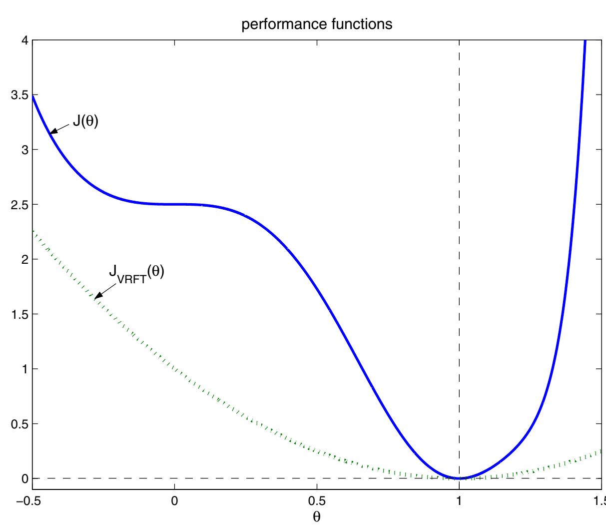


Fig. 1. $J(\theta)$ vs. $J_{VRFT}(\theta)$

Notice that, in contrast with $J$ , the $J_{VRFT}$ cost can be constructed out of data without knowledge of $P$; moreover, $J_{VRFT}$ is quadratic in $\theta$ (and therefore easy to minimize).$*$

The fact seen in the previous example that the ideal $\theta_0$ 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 $\theta_0$ gives perfect tracking: $\|y_{\theta_0} - M[\tilde{r}]\|^2 = 0$, then $\theta_0$ 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:** see [5].

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 2), 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 $J$.

## IV. FILTER DESIGN

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. But 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 $J$.

Using the delay matrix $D$ in (3) and $M$, 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:

$$C^0 := P^{-1}(I - MD)^{-1}M. \tag{7}$$

$C^0$ 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:

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

*Example 3 (Example 2 continued):* For the situation in Example 2 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,

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

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

leading to

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

$$= P^{-1} \begin{bmatrix} e(0) \\ e(0) + e(1) \end{bmatrix} = \begin{bmatrix} e(0)^{1/3} \\ e(1)^{1/3} \end{bmatrix},$$

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

Let

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

where $C_0$ is $C_\theta$ 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

$$J(\theta^+) := \|y_{\theta^+} - M[\tilde{r}]\|^2, \quad y_{\theta^+} = P[C_{\theta^+}[\tilde{r} - Dy_{\theta^+}]]. \tag{8}$$

Let

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

We want to select $F$ so that

$$\left.\frac{\partial^2 J_{VRFT}(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+} = \left.\frac{\partial^2 J(\theta^+)}{\partial \theta^{+2}}\right|_{\theta_0^+}. \qquad (9)$$

Note that, if $C_{\theta+}[\tilde{e}]$ is linear in $\theta^+$, under (9) $J_{VRFT}(\theta^+)$ is the 2-nd order expansion of $J(\theta^+)$.

The following theorem specifies how $F$ must be selected so that (9) is satisfied.

*Theorem 2:* If

$$F = (I - MD)\left(\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}\right), \qquad (10)$$

then (9) holds. $*$

**proof:** See [5].

A few remarks are in order.

*Remark 1 (about the structure of the filter):* The filter in (10) is formed by two parts: (i) $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ and (ii) $(I - MD)$. $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ keeps into account the effect of input (used in the VRFT cost) on output (used in the control objective cost). $*$

*Remark 2 (Direct vs. indirect):* The term $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ 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 $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ is linear and time-varying and it can be estimated e.g. via forgetting factor identification techniques, [11], [1], [10] .
Since $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ has to be estimated, strictly the VRFT method is not direct. On the other hand, one should note that the estimated expression of $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ is not used for design, it is only used as a filter to process the data. As a consequence, a precise determination of $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ is not required: imprecision in $\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}$ only reflects in that the second derivative of $J_{VRFT}$ will not precisely match that of $J$ around $\theta_0^+$. $*$

*Example 4 (Example 3 continued):* We have:

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

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

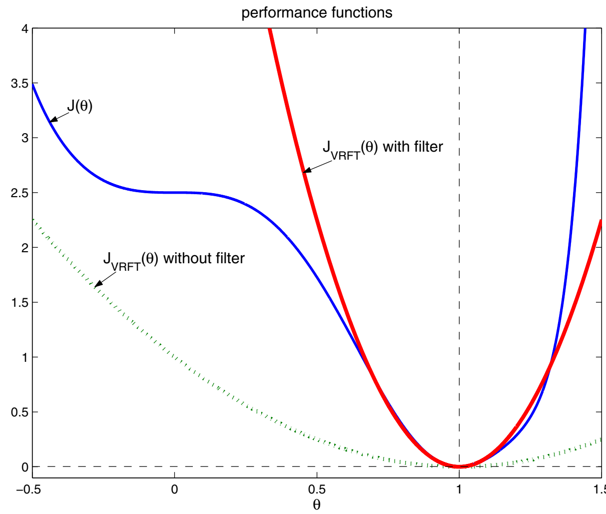


Fig. 2. $J_{VRFT}(\theta_{eq})$ with and without $F$ for $\theta_{eq} = \theta$

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

while, without filter $F$,

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

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

$$J_{VRFT}(\theta_{eq}) = (3\theta_{eq} \cdot 1^{1/3} - 3)^2 + (3\theta_{eq} \cdot 1^{1/3} - 3)^2 = \\ = 18 - 36\theta_{eq} + 18\theta_{eq}^2,$$

with the second order derivative

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

For a visual comparison, $J_{VRFT}(\theta_{eq})$ with and without filter are displayed in Figure 2. $*$

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

*Example 5:*

$$y(t) = 0.9y(t-1) + \tanh(u(t-1)),$$

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

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

where

$$v(t) = v(t-1) + e(t) - 0.9e(t-1).$$

Consider now a PI (proportional-integral) controller class $C_\theta$:

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

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 $\theta$ has to be selected.

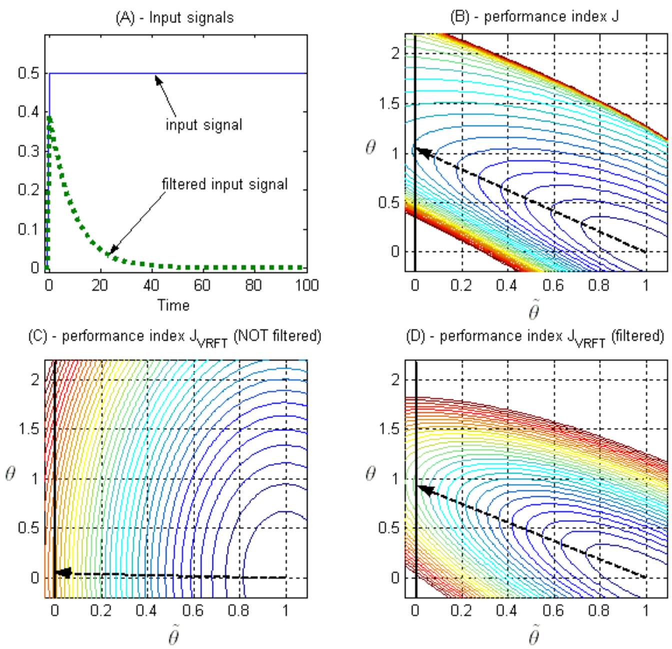


Fig. 3. (A): plant input $\tilde{u}$ and filtered input $F[\tilde{u}]$. Contour plots of performance indices: (B): $J(\theta^+)$; (C): $J_{VRFT}(\theta^+)$ (not filtered); (D): $J_{VRFT}(\theta^+)$ (filtered)

In order to design the controller, plant $P$ has been fed with a step input $\tilde{u}$ of amplitude $0.5$ (see Figure 3-(A)). Using the corresponding output $\tilde{y}$, the performance indices $J(\theta^+)$ and $J_{VRFT}(\theta^+)$ (with and without filtering) have been computed and their contour plots are displayed in Figures 3-(B),(C),(D). By inspecting these three figures, the following observations can be drawn:

- All three 2-dimensional indices 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 $\theta^+$. Instead, $J_{VRFT}(\theta^+)$ is quadratic.
- Even if $J(\theta^+)$ and the non-filtered $J_{VRFT}(\theta^+)$ (figures (B) and (C)) 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 $J_{VRFT}(\theta^+)$ (figure (D)) in order to make it equal to the 2-nd order expansion of $J(\theta^+)$ around its minimum.
- The PI performance indices (which depend on the parameter $\theta$ only) can be obtained by cutting the original 2-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 indices. Apparently, thanks to the fact that the filtered $J_{VRFT}(\theta^+)$ is the 2-nd order expansion of $J(\theta^+)$, the minimizer of $J(\theta)$ and the minimizer of the filtered $J_{VRFT}(\theta)$ are very close; on the contrary, the minimizer of the non-filtered $J_{VRFT}(\theta)$ drifts away from the minimizer of $J(\theta)$.
- Figure 4 shows the closed-loop output when the reference signal is $\tilde{r}$ and the controller has been designed with the filtered and non-filtered $J_{VRFT}$. Note the remarkable effect of filtering.

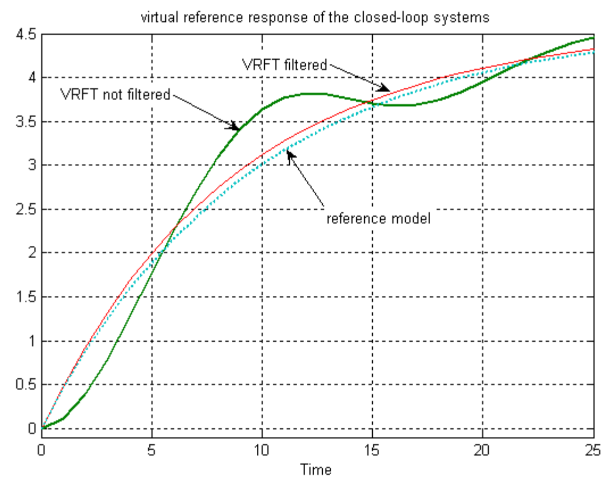


Fig. 4. Closed loop response to the virtual reference input $\tilde{r}$

## REFERENCES


[1] M.C. Campi. On the convergence of minimum-variance directional-forgetting adaptive control schemes. *Automatica*, 28:221-225, 1991.

[2] M.C. Campi, A. Lecchini, and S.M. Savaresi. Virtual Reference Feedback Tuning: a direct method for the design of feedback controllers. *Automatica*, 38:1337-1346, 2002.

[3] M.C. Campi, A. Lecchini, and S.M. Savaresi. An application of the virtual reference feedback tuning method to a benchmark problem. *European Journal of Control*, Special Issue on ”Design and Optimisation of Restricted Complexity Controllers”, 1:66-76, 2003.

[4] M.C. Campi and S.M. Savaresi. Invertibility of nonlinear maps. *Internal Report, University of Brescia*, 2004.

[5] M.C. Campi and S.M. Savaresi. Direct Nonlinear Control Design: the Virtual Reference Feedback Tuning (VRFT) Approach. *Internal Report, University of Brescia*, 2005.

[6] E.B. Dahlin. Designing and tuning digital controllers. *Instruments and Control Systems*, 42:77–83, June 1968.

[7] G.O. Guardabassi, and S. Savaresi. Approximate Linearization via Feedback: an Overview. *Automatica*, 37:1-15, 2001.

[8] H. Hjalmarsson. Control of nonlinear systems using iterative feedback tuning. *American Control Conference*, 4:2083-2087, 1998.

[9] H. Hjalmarsson, M. Gevers, S. Gunnarson, and O. Lequin. Iterative Feedback Tuning: theory and applications. *IEEE Control Systems Magazine*, 18(4):26-41, 1998.

[10] L. Ljung. *System Identification: theory for the user*. Prentice Hall, 1999.

[11] T. Soderstrom, and P. Stoica. *System Identification*. Prentice Hall, 1988.

[12] J.G. Ziegler, and N.B. Nichols. Optimum settings for automatic controllers. *Trans. ASME*, 64:759-768, 1942.