European Journal of Control (2003)9:66–76

European Journal of Control

An Application of the Virtual Reference Feedback Tuning Method to a Benchmark Problem*

M.C. Campi[1], A. Lecchini[2,†] and S.M. Savaresi[3,‡]

[1]Dip. di Elettronica per l'Automazione – Università di Brescia, Via Branze 38, Italy; [2]CESAME – Université Catholique de Louvain Bâtiment Euler, 4 Av. Georges Lemaître, B-1348 Louvain-la-Neuve, Belgium and [3]Dip. di Elettronica ed Informazione – Politecnico di Milano Piazza Leonardo da Vinci 32, 20133 Milano, Italy

Virtual Reference Feedback Tuning (VRFT) is a general methodology for the design of a controller when the plant transfer function is unknown, proposed by the same authors in previous contributions. It is a direct method that aims at minimizing a control cost of the 2-norm type by using a batch of data collected from the plant. The minimisation is conducted in one-shot (the method is not iterative) and this makes VRFT particularly handy in many practical applications. This paper presents an application of VRFT to a benchmark active suspension system. As a by-product, this paper also delivers a new extension of VRFT that permits to cope with constraints on the input-sensitivity.

Keywords: Direct controller tuning; Identification; Restricted complexity

1. Introduction

Virtual Reference Feedback Tuning (VRFT) is a method for the direct design of a controller based on a set of input–output measurements with no need for an intermediate step where a model of the plant is identified. It was originally introduced for 1-degree of freedom controllers in [6,7,20] and then extended to a 2-degree of freedom context in [21,22].

In this paper, VRFT is applied to a benchmark control problem where one has to design a controller for an active suspension system (see Section 3 for a precise description of the problem).

One aspect that is particularly emphasized in the benchmark is that the controller should be of restricted complexity. As a matter of fact, the controller complexity is a very important issue in real applications, either because of constraints in the computation time and/or in the hardware or just because simple controllers are easier to understand. Many controller reduction techniques have been proposed in the literature mainly belonging to two different approaches: (i) *a priori* reduction of the model complexity [2,4]; and (ii) *a posteriori* reduction of the controller complexity [3]. Along the first approach, one first reduces the order of the model in such a way that by applying standard controller design techniques one then obtains a simple controller. The main inconvenience

*This paper is supported by MIUR under the project *New Methods for Identification and Adaptive Control for Industrial Systems*, by the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture and by the European Research Network on System Identification (ERNSI) funded by the European Union. The scientific responsibility rests with its authors.
Correspondence and offprint requests to: M.C. Campi, Dip. di Elettronica per l'Automazione – Università di Brescia, Via Branze 38, 25123 Brescia, Italy. E-mail: campi@ing.unibs.it
†E-mail: lecchini@auto.ucl.ac.be
‡E-mail: savaresi@elet.polimi.it

Received 27 June 2002; Accepted in revised form 13 November 2002. Recommended by H. Hjalmarsson and I.D. Landau.

of this approach is that it is difficult to keep control on how the approximation introduced in the model reduction step spreads in the subsequent controller design step. The second approach does not suffer from this drawback but calls for a close attention to preserve the properties of the original controller.

In connection with this restricted controller complexity issue, it is worth mentioning that with VRFT the problem disappears because this method allows the designer to select a controller class of desired complexity at the start. The reason why this is possible is that VRFT is a direct method that does not call for a plant model identification.

Other direct methods where the controller class is selected at the start have recently appeared in the literature, see e.g. [14–18,28]. In contrast, the literature on indirect methods is truly vast and includes, to cite but a few, the contributions [1,5,8–10,13,23,24, 27,29,30].

The paper is organized as follows. The VRFT method is presented in Section 2 and its application to the benchmark problem is discussed in Section 3. The paper is closed by a short section of conclusions.

2. A Review of the VRFT Method

In this section, a review of the VRFT method is presented (the reader is referred to [6,7,20–22] – see also [11,12,25,26] – for a more comprehensive presentation of the method). VRFT was initially conceived for the shaping of the output-sensitivity of a control system. Here, motivated by the benchmark application, we also introduce an extension of the method in order to cope with the input-sensitivity as well. Such an extension is presented in this paper for the first time.

Consider the control system depicted in Fig. 1. It is a 1-degree-of-freedom SISO control architecture, where $P(z)$ is the plant and $C(z,\theta)$ is the controller. θ is a vector of parameters that has to be selected by the controller designer. $P(z)$ is unknown, that is no mathematical description of the plant is available.

Our presentation is specialized to the situation where the control goal is to regulate the system output to zero (that is no tracking of a reference signal is required) since this is the actual situation in the benchmark problem. Thus, $y_0=0$ in Fig. 1. Throughout, the transfer function form d to y is named the *output-sensitivity* and that form d to u the *input-sensitivity*.

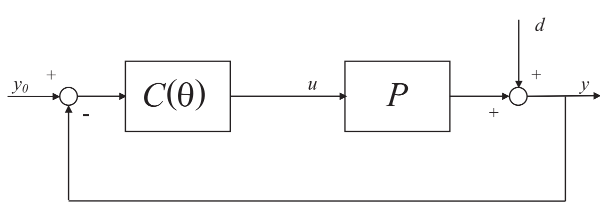

Fig. 1. The control system.

2.1 Standpoint in the VRFT Method

VRFT is a method for the direct design of a controller when no mathematical description of the plant is known. This means that knowledge of $P(z)$ in Fig. 1 is not required. The only information the method uses is a set of I/O data $\{u(t), y(t)\}_{t=1,\dots,N}$ collected from the plant. Thus, in the benchmark application (see Section 3), our design will be based only on the files of input–output data provided, while no use of the plant model will be made.

2.2. Output-Sensitivity Shaping

In VRFT, the problem of shaping the output-sensitivity is stated as a model reference control problem in the 2-norm. Precisely, given a reference-model for the output-sensitivity $S(z)$ and a class of controllers $C(z,\theta)$, the objective is to select the parameter vector θ which minimizes the cost

$$J_S(\theta) = \left\| \left[S(z) - \frac{1}{1+P(z)C(z,\theta)} \right] W_S(z) \right\|_2^2, \tag{1}$$

where $W_S(z)$ is a weighting function chosen by the designer (recall that $\|H(z)\|_2^2 := (1/2\pi) \int_{-\pi}^{\pi} |H(e^{j\omega})|^2 \, d\omega$).

VRFT provides a solution to this problem when $P(z)$ is not known and a set of input–output measurements collected from the plant is available.

The main features of VRFT are as follows:

- requires just a single set of I/O data (in fact, when one also considers the input-sensitivity, a second experiment is required);
- does not require the identification of a mathematical model of the plant (i.e. it is a direct method);
- determines the controller parameters in one-shot, with no need for iterations;
- the controller complexity can be fixed in the beginning when the controller class is selected. So, no controller complexity reduction is required;
- the controller provided by VRFT is only an approximate minimizer of (1). However, by virtue of the way the method has been designed, such a minimizer is close to the optimum in standard situations.

We now describe how the VRFT method works.

The selection of θ through VRFT turns out to be particularly simple for linear-in-the-parameters controllers, namely:

$$C(z,\theta) = \beta^{\mathrm{T}}(z)\theta, \tag{2}$$

where $\beta(z) = [\beta_1(z)\ \ \beta_2(z)\ \ \cdots\ \ \beta_n(z)]^{\mathrm{T}}$ is a vector of linear discrete-time transfer functions and $\theta = [\vartheta_1\ \ \vartheta_2\ \ \cdots\ \ \vartheta_n]^{\mathrm{T}} \in \mathbb{R}^n$ is an n-dimensional vector of real parameters. Since the adopted controller in the benchmark has this structure, here we restrict attention to this situation.

The main idea behind VRFT is to introduce a new control cost – named $J_{\mathrm{VRFT}}(\theta)$ – which exhibits two properties, the first of which is crucial for the applicability of the method:

– it is computable with no knowledge of the plant model $P(z)$;
– it is quadratic in θ, so that the minimisation is easy to carry out.

To describe the idea, a little notation is in order. Let

$$F(z,\theta) := \left[S(z) - \frac{1}{1+P(z)C(z,\theta)}\right] W_S(z);$$
$$G(z,\theta) := 1 + P(z)C(z,\theta).$$

Then, the cost function $J_S(\theta)$ can be written as

$$J_S(\theta) = \|F(z,\theta)\|_2^2. \tag{3}$$

Further, we let

$$J_{\mathrm{VRFT}}(\theta) = \|L(z)G(z,\theta)F(z,\theta)\|_2^2, \tag{4}$$

where $L(z)$ is a filter, whose choice is discussed below.

The name VRFT for cost (4) stems from an interpretation of it given in terms of a virtual reference constructed from data as explained in [6,20,21].

Cost (4) is the asymptotic counterpart (as the number of data points tends to infinity) of a cost that can be computed from a set of I/O data $\{u(t), y(t)\}_{t=1,\ldots,N}$ with no use of $P(z)$. To see this, write

$$J_{\mathrm{VRFT}}(\theta) = \|L(z)S(z)P(z)W_S(z)C(z,\theta) - L(z)[1-S(z)]W_S(z)\|_2^2.$$

If u is filtered through the transfer function under the sign of norm, we obtain

$$\begin{aligned}&[L(z)S(z)P(z)W_S(z)C(z,\theta) - L(z)[1-S(z)]W_S(z)]u(t)\\ &\quad = L(z)S(z)P(z)W_S(z)C(z,\theta)u(t) - L(z)[1-S(z)]W_S(z)u(t)\\ &\quad = L(z)S(z)W_S(z)C(z,\theta)y(t) - L(z)[1-S(z)]W_S(z)u(t),\end{aligned}$$

where in the last step we have used the fact that $P(z)u(t) = y(t)$. Here, one should note that this equality holds true in the context of the present application since the $u - y$ data provided in the benchmark have been collected with no noise acting on the system. Should the experiment be noisy, one could apply the method as well with a bit of additional complication by resorting to instrumental variable techniques, as explained in [6,7,20–22].

From this we see that, under standard stationarity and ergodicity conditions, the data-based cost

$$J_{\mathrm{VRFT}}^N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(L(z)S(z)W_S(z)Q(z)C(z,\theta)y(t) - L(z)[1-S(z)]W_S(z)Q(z)u(t))^2, \tag{5}$$

where $Q(z)$ is the inverse of a canonical spectral factor of u (i.e. $|Q|^2 = 1/\Phi_u$ where Φ_u is the power spectral density of u and Q is stable) converges asymptotically to $J_{\mathrm{VRFT}}(\theta)$. In addition, (5) is quadratic in θ if $C(z,\theta)$ is linearly parametrized as is in (2).

Now, the question arises naturally as to how we should select $L(z)$ so that minimizing (4) leads to an approximate minimizer of the original cost (3). The answer is what we could expect: select $L(z) = S(z)$. As a matter of fact, this choice corresponds to multiplying $F(z,\theta)$ in (4) by $S(z)G(z,\theta)$ where the first term is the desired sensitivity that somehow cancels $G(z,\theta)$, the inverse of the true sensitivity. It is important to note that this choice has a precise mathematical motivation that goes beyond what appears at this superficial level of description. In order to keep the main body of the paper as thin as possible, we have been well advised to put the presentation of this mathematical motivation in an appendix to which we refer the interested reader.

2.3 Shaping of the Input-Sensitivity

In this subsection we extend the method to the shaping of the input-sensitivity.

Given a reference-model $U(z)$ for the input-sensitivity, the goal is to minimize the following cost:

$$J_U(\theta) = \left\|\left[U(z) - \frac{C(z,\theta)}{1+P(z)C(z,\theta)}\right] W_U(z)\right\|_2^2,$$

where $W_U(z)$ is a weighting function chosen by the designer.

Again, the idea is to build a data-based cost function, which approximates $J_U(\theta)$. Similarly to $J_{\mathrm{VRFT}}(\theta)$, such a cost is obtained by multiplying the transfer function under the sign of norm in $J_U(\theta)$ by $1 + P(z)C(z,\theta)$ and further allowing for a filter action $L(z)$ (this $L(z)$ is different from the one used in $J_{\mathrm{VRFT}}(\theta)$). This leads to

$$J_{\mathrm{VRFT}-U}(\theta) = \|L(z)U(z)W_U(z) - L(z)[1 - U(z)P(z)]W_U(z)C(z,\theta)\|_2^2.$$

The data-based counterpart of $J_{\mathrm{VRFT}-U}(\theta)$ is given by:

$$J^N_{\mathrm{VRFT}-U}(\theta) = \frac{1}{N}\sum_{t=1}^{N}(L(z)U(z)W_U(z)Q(z)u(t) - L(z)W_U(z)Q(z)C(z,\theta)[u(t) - U(z)y(t)])^2. \quad (6)$$

In (6), the filter $L(z)$ that provides the best matching of $J_U(\theta)$ and $J_{\mathrm{VRFT}-U}(\theta)$ (in the sense that it equalizes the Hessians of an extended version of these two costs – see the Appendix for a discussion on this matter in the case of $J_S(\theta)$) is

$$L(z) = 1 - U(z)P(z), \quad (7)$$

(the reader can verify this by resorting to arguments similar to those developed for $J_S(\theta)$ in the Appendix).

Notice that, differently from the output-sensitivity, $L(z)$ in (7) contains $P(z)$ explicitly, so that it cannot be directly implemented. Yet, we can spot a way to circumvent this difficulty.

Suppose we make a second experiment on the plant by feeding the plant with $y(t)$. Then, the new output is

$$y'(t) = P(z)y(t).$$

Now, cost $J^N_{\mathrm{VRFT}-U}(\theta)$ with $L(z) = 1 - U(z)P(z)$ can be rewritten as

$$\begin{aligned} J^N_{\mathrm{VRFT}-U}(\theta) &= \frac{1}{N}\sum_{t=1}^{N}([1 - U(z)P(z)]U(z)W_U(z)Q(z)u(t) \\ &\quad - [1 - U(z)P(z)]W_U(z)Q(z)C(z,\theta)[u(t) - U(z)y(t)])^2 \\ &= \frac{1}{N}\sum_{t=1}^{N}(U(z)W_U(z)Q(z)[u(t) - U(z)y(t)] \\ &\quad - W_U(z)Q(z)C(z,\theta)[u(t) - 2U(z)y(t) + U^2(z)y'(t)])^2. \end{aligned}$$

The $J^N_{\mathrm{VRFT}-U}(\theta)$ above is the data-based cost for the input-sensitivity shaping problem.

It is perhaps worth emphasizing that, similarly to the $J_S(\theta)$ case, the plant is assumed to be noise-free here (in fact the relation $y'(t) = P(z)y(t)$ contains no noise term). Should the plant be noisy, the method could still be applied by resorting to an instrumental variable variant as we sketch in the following. Note that $J^N_{\mathrm{VRFT}-U}(\theta)$ is a quadratic cost, whose minimizer is obtained by standard normal equations. In instrumental variable techniques, the regression vector is substituted in the normal equations by a surrogate observation vector, correlated with the original observation vector but uncorrelated with the noise part contained in it. In our context, the surrogate observation vector can be constructed by feeding the plant a second time with the same input $u(t)$, and subsequently feeding back the plant with the measured plant output. If the surrogate observation vector is constructed from the signals collected in these extra experiments, the noise part will exhibit the desired uncorrelation property with the noise part in the original observation vector. The cost paid, of course, is that one needs to experiment four times on the plant, instead of two.

The idea of re-injecting the system output in the control system has been previously used in [15,17,18], though to a different scope.

2.4. The Design Algorithm

When the design of the controller has to be performed with joint specifications on the input- and output-sensitivity, a cost combining the two partial costs $J^N_{\mathrm{VRFT}}(\theta)$ and $J^N_{\mathrm{VRFT}-U}(\theta)$ can be used. The corresponding VRFT method can then be summarized in a ready-to-use algorithmic form as follows.

The design parameters which must be *a priori* selected by the designer are the following:

- $C(z,\theta)$: the parametric controller class;
- $S(z)$: the output-sensitivity reference model;
- $U(z)$: the input-sensitivity reference model;
- $W_S(z)$: the weighting factor for the output-sensitivity;
- $W_U(z)$: the weighting factor for the input-sensitivity.

The algorithm is then as follows (here $y(t)$ is assumed to be noise-free):

Design Algorithm

1. Perform an open-loop experiment on the plant and collect a set of I/O data $\{u(t), y(t)\}_{t=1,\ldots N}$.
2. Perform a second experiment by feeding the plant with $\{y(t)\}_{t=1,\ldots,N}$ and collect the output $\{y'(t)\}_{t=1,\ldots,N}$.
3. Identify a (high-order) model $R(z)$ of $\{u(t)\}_{t=1,\ldots,N}$. Let $Q(z) = R(z)^{-1}$.

4. Select the parameter vector $\hat{\theta}_N$ which minimizes the cost:

$$J^N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(S^2(z)W_S(z)Q(z)C(z,\theta)y(t) - S(z)[1-S(z)]W_S(z)Q(z)u(t))^2 + \frac{1}{N}\sum_{t=1}^{N}(U(z)W_U(z)Q(z)[u(t)-U(z)y(t)] - W_U(z)Q(z)C(z,\theta)[u(t)-2U(z)y(t) + U^2(z)y'(t)])^2 \quad (8)$$

A few comments on the algorithm are now in order.

(i) The $J^N(\theta)$ cost is obtained as the sum of $J^N_{\text{VRFT}}(\theta)$ and $J^N_{\text{VRFT}-U}(\theta)$. Clearly, weights different from 1 can be given to either terms in order to place more emphasis on the output- or on the input-sensitivity shaping.

(ii) As it has been noted in previous contributions, the output-sensitivity shaping can be easily performed in a closed-loop experimental set-up. Instead, shaping the input-sensitivity with data collected in closed-loop is not possible along the indicated approach since the plant cannot be directly fed by $y(t)$ so as to generate $y'(t)$.

(iii) Though the asymptotic VRFT cost is not affected by the frequency content of the input (provided that Φ_u is invertible), the finite sample cost does depend on the input content. As a consequence – and as it is expected – VRFT exhibits better performance when the input signal is rich corresponding to frequencies of interest for control, such as the crossover frequency.

(iv) If $P(z)$ has a large gain, $y(t)$ can be large as compared to $u(t)$. At step 2 of the algorithm, $y(t)$ is re-injected in the plant to collect $y'(t)$, and, should the plant be nonlinear (i.e. $P(z)$ only represents a small signal linearization), this might move the plant out of the linear region. In this case, a suitable scaling of $y(t)$ can be performed: first $y(t)$ is multiplied by a factor $\rho < 1$ and $\rho y(t)$ is injected in the plant. The collected output is then multiplied by $1/\rho$ so as to obtain $y'(t)$.

(v) In the algorithm, $C(z,\theta)$ is kept fixed as the controller class is chosen before running the algorithm. On the other hand, in a real application the quality of the selected controller can be tested only *a posteriori* by observing the control system behavior with the controller placed in the loop. If the performance is unsatisfactory, one may have to change the controller order and run the algorithm again. Thus, the final choice of the controller class may come as a result of compromising performance versus complexity by trial and error.

3. Design of the Controller for the Active Suspension

This section is devoted to the presentation of the results obtained by applying the VRFT approach (as described in the previous section) to the benchmark active suspension system. Specifically, in Section 3.1 the system and control specifications are briefly summarized, whereas in Section 3.2 the VRFT design procedure and the experimental results are illustrated.

3.1. Benchmark Problem: System Description and Control Specifications

The plant considered in the benchmark control problem is an active hydro-suspension system (see e.g. [19] and the website http://iawww.epfl.ch/News/EJC_Benchmark for details). This benchmark laboratory set-up is located at the Laboratoire d'Automatique de Grenoble.

The main parts of the suspension system are (Fig. 2):

- an elastomere cone that encloses the main chamber filled with silicon oil (1);
- an inertia chamber enclosed with a flexible membrane (2);
- a piston (3) that is fixed on a DC motor; when the position of the piston is fixed, the suspension system is passive;
- an orifice (4) that allows oil flow between the two chambers.

The principal idea of the active suspension is to change the elasticity of the system in order to absorb the vibrations generated by the machine that we want to isolate. For the experimental purposes the machine is replaced by a shaker which is driven by a computer generated control signal. The output of the system is the measured voltage corresponding to the residual force. The control input drives the position of the piston via an actuator.

The transfer function $P(z)$ between the control input and the residual force is called the *secondary path*. The magnitude Bode diagram (experimentally estimated) of the secondary path is depicted in Fig. 3. Notice that the plant has zero DC-gain, and that it is characterized by many resonance peaks; the first two (main) resonance peaks are at about 30 and 170 Hz, respectively. The controller is implemented in a Matlab/Simulink PC-based digital environment. The sampling frequency is 800 Hz.

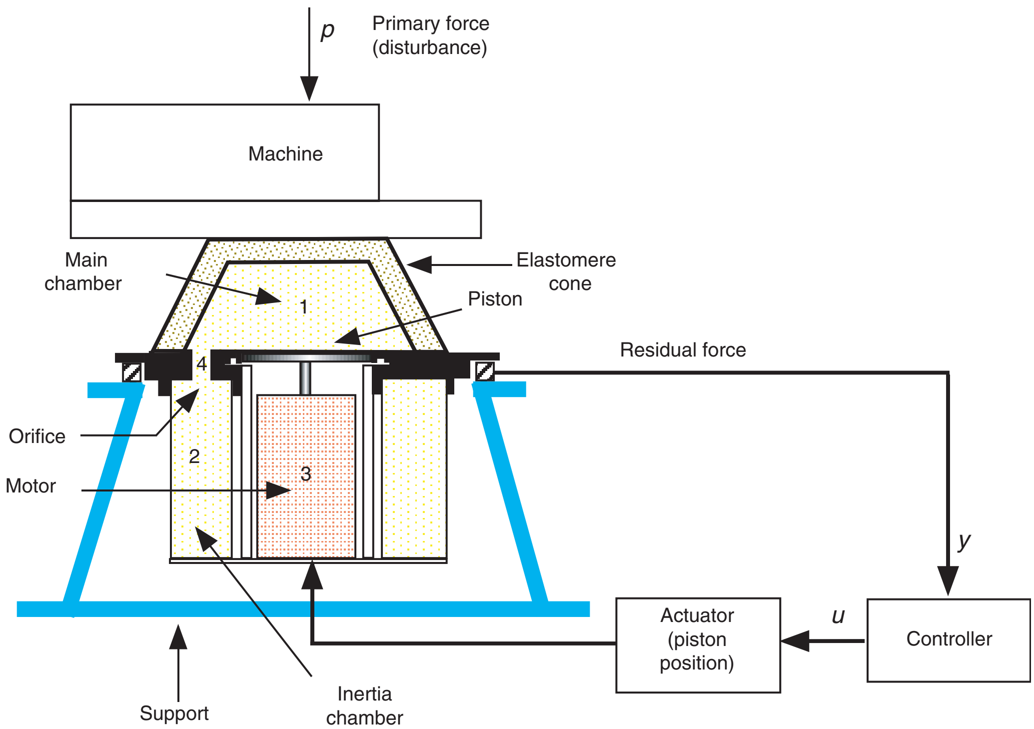

Fig. 2. Representation of the benchmark active suspension system.

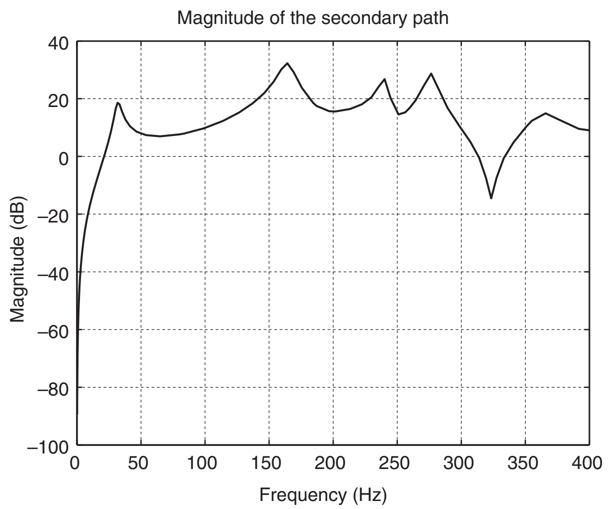

Fig. 3. Bode diagram (experimentally estimated) of the secondary path.

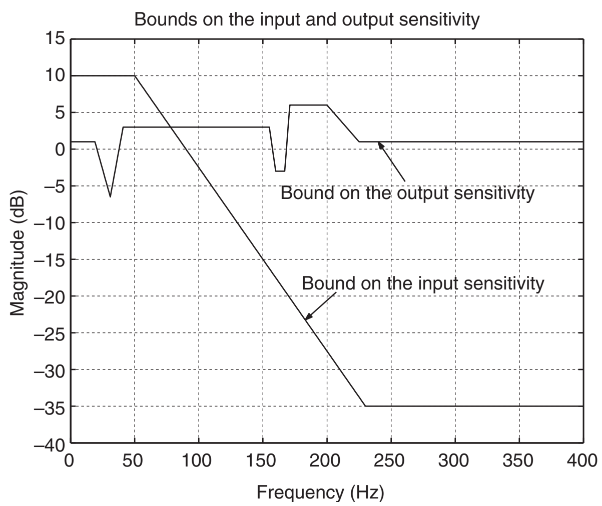

Fig. 4. Bounds on output- and input-sensitivity (specs of the benchmark).

The main goal of the control system is to minimize the residual force. More in detail, the specifications given in the benchmark are:

- the closed-loop output- and input-sensitivity functions must be bounded by the frequency-domain functions depicted in Fig. 4;
- the control architecture must have a standard 1-degree-of-freedom structure;
- the controller gain should be equal to zero at the Nyquist frequency (hence, the term $1+z^{-1}$ should be incorporated in the controller);
- the order that the controller must be as low as possible (in the benchmark the issue of tuning "reduced-complexity" controllers is emphasized).

3.2. Controller Design Using VRFT and Experimental Results

With reference to the VRFT design algorithm summarized in Section 2.4, the following design choices have been made:

- *Controller structure*. The selected parametric controller has a very simple all-zeros transfer function, including the $(1 + z^{-1})$ factor, namely:

$$C(z;\theta) = (\vartheta_0 + \vartheta_1 z^{-1} + \vartheta_2 z^{-2} + \vartheta_3 z^{-3} + \vartheta_4 z^{-4} + \cdots + \vartheta_K z^{-K})(1 + z^{-1}),$$

  where K is the number of free parameters in the controller.
- *Output-sensitivity reference model*. The design of the output-sensitivity reference model $S(z)$ has been done by fitting in the frequency domain the bound defined by the specs with a high-order model (a 30-poles–30-zeros IIR filter has been used). The design of the filter has been done using standard frequency-shaping signal processing techniques. Note that the so-obtained filter has been multiplied by a suitable gain in order to guarantee a monic numerator and a monic denominator (it is easy to see that the sensitivity function must have this property). The magnitude Bode diagram of the output-sensitivity reference model used for the controller design is depicted in Fig. 5.
- *Input-sensitivity reference model*. Similarly to the output-sensitivity, the design of the input-sensitivity reference model $U(z)$ has been done by fitting in the frequency domain the bound defined by the specs with a high-order model. Even in this case the so-obtained filter has been multiplied by an attenuating gain, in order to enforce the fitting to a value lower than the required bound. The magnitude Bode diagram of the input-sensitivity reference model used for the controller design is depicted in Fig. 6.
- *Output-sensitivity weighting function*. The output-sensitivity weighting function $W_S(z)$ has been designed to emphasize the fitting around the two "notches" at about 30 and 170 Hz. As a matter of fact, the shaping around the notches is the most important and critical. The magnitude Bode diagram of the output-sensitivity weighting function used for the controller design is depicted in Fig. 5.
- *Input-sensitivity weighting function*. The input-sensitivity weighting function $W_U(z)$ has been designed to emphasize the fitting at high frequency. As a matter of fact, the fulfillment of the bound at high frequency is the most critical and challenging. The magnitude Bode diagram of the input-sensitivity weighting function used for the controller design is depicted in Fig. 6.

The above reference models have been obtained by simply scaling the bounds for the input- and output-sensitivity given in the benchmark specs. The weighting functions have instead been selected so as to emphasize the frequency regions where the reference models are low in magnitude. The reason is that a large percentage error where the reference is small has little impact on the control cost. Moreover, these regions cover narrow bandwidths so that they have marginal importance in the integral 2-norm cost. With these selection criteria in place, the various choices have been readily made.

Using the design choices described above, the controller has been straightforwardly designed using the I/O data measured on the plant and made available

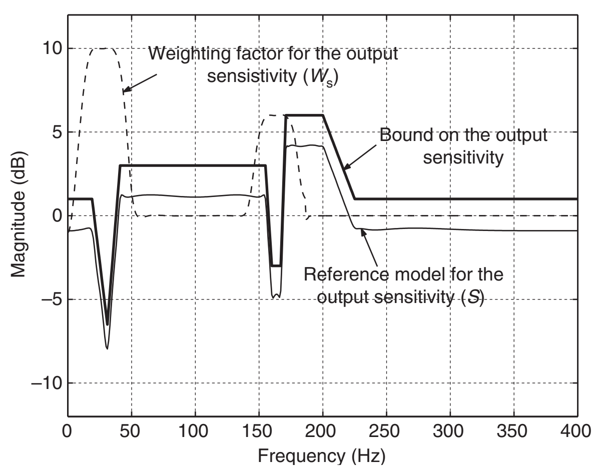

Fig. 5. Reference model and weights for the output-sensitivity.

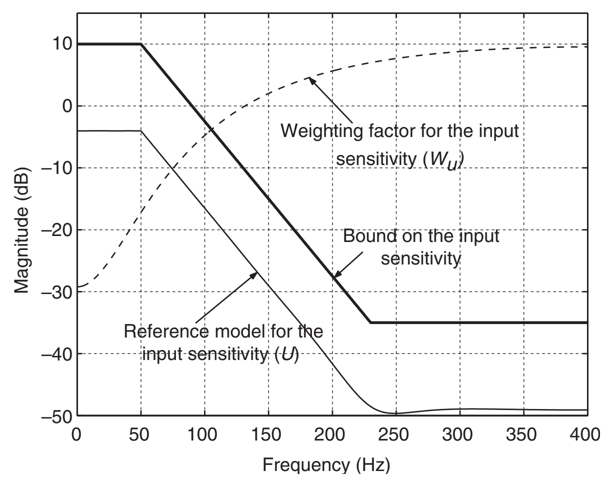

Fig. 6. Reference model and weights for the input-sensitivity.

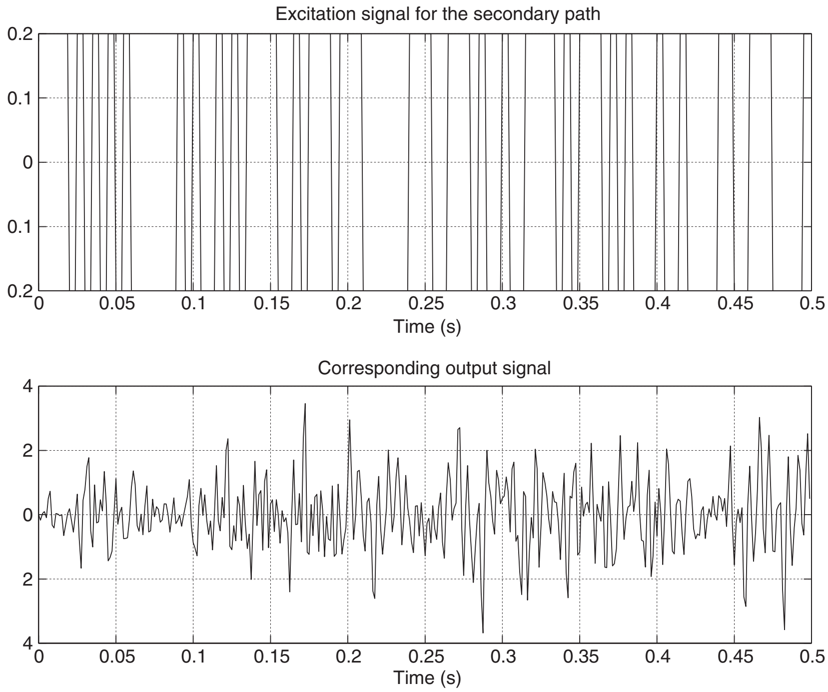

Fig. 7. Sample of 0.5 s of the I/O signal measured on the plant.

for the benchmark. Specifically, the input signal used for open-loop excitation is a PRBS signal sampled at 800 Hz. The length of the data vector is 4096 (corresponding to 5.12 s of data acquisition). A sample of 0.5 s of the I/O signals measured on the plant is displayed in Fig. 7. Accordingly to the characteristics of the measured input signal, the filter $Q(z)$ used by VRFT has been set to 1. Finally, for the sake of simplicity, signal $y'(t) = P(z)y(t)$ has been obtained by filtering $y(t)$ through a high-order model of the secondary path provided in the benchmark. In this connection, it is important to point out that this choice has been made in this specific context since such a high-order model was made available, but in real applications where a model is not available one can directly feed the *real* plant with $y(t)$ to obtain $y'(t)$.

The design of the controller parameters has been performed by minimizing the performance index (8). As already pointed out, being (8) quadratic in θ, the minimisation is straightforward. The tuning procedure has been repeated three times, in correspondence of three different controller orders: $K = 22$, 14, and 6. These three controllers have been sent to Grenoble for testing on the real plant. The results of these experiments are displayed in the frequency-domain in Fig. 8 (controller with 22 parameters), in Fig. 9 (controller with 14 parameters), and in Fig. 10 (controller with six parameters).

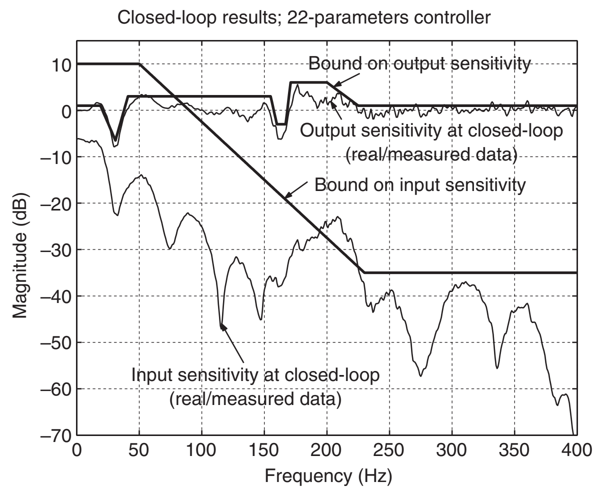

Fig. 8. Closed-loop results obtained on the benchmark system (controller with 22 parameters).

From the results, the following conclusions and remarks can be drawn:

- the performance obtained by the 22-parameters controller is very good: the bound on the output-sensitivity is almost perfectly fulfilled, and the bound on the input sensitivity is slightly violated only at about 200 Hz;
- the performance obtained by the 14-parameters controller is still quite good: notice that the bound

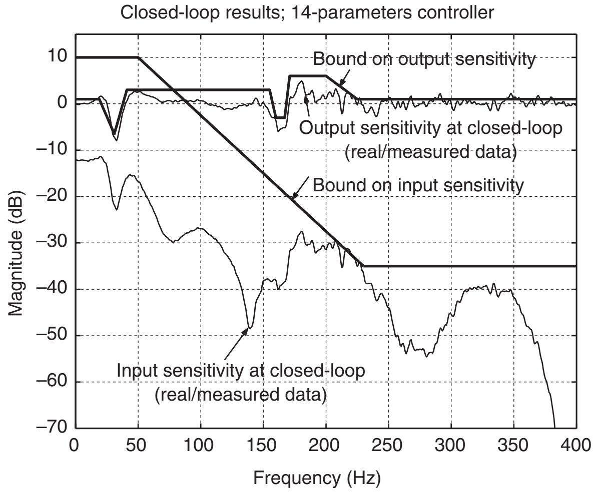

Fig. 9. Closed-loop results obtained on the benchmark system (controller with 14 parameters).

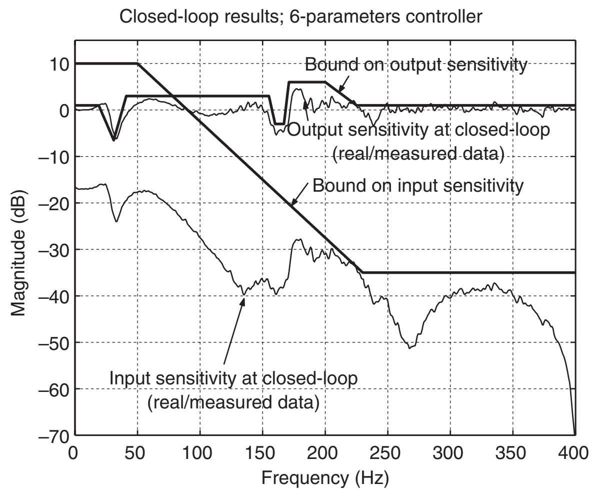

Fig. 10. Closed-loop results obtained on the benchmark system (controller with six parameters).

on the input-sensitivity is completely fulfilled, whereas the bound on the 30 Hz notch in the output-sensitivity is only slightly violated;

- the performance obtained by the 6-parameters controller is not as good as for the 22- and 14-parameters controllers, but they are still acceptable. We believe that this can be considered a very good result, given the very low complexity of the controller.

4. Conclusions

In this paper, an application of the VRFT method to an active suspension system has been presented. The corresponding results show that the method has been able to provide satisfactory controllers in this context.

Among other features, VRFT allows to come up with a controller by the simple minimisation of a quadratic cost with no need for iterations (so that no repetitive experiments on the real plant are required). Moreover, the controller complexity can be freely chosen by the designer at the start so that one need not go through a controller complexity reduction so as to meet specific constraints on the controller structure.

References

1. Anderson BDO, Kosut RL. Adaptive robust control: on-line learning. In: Proceedings of the 30th IEEE conference on decision and control. 1991, pp 297–298
2. Anderson BDO, Liu Y. Controller reduction: concepts and approaches. IEEE Trans Autom Control 1989; 34: 802–812
3. Bendotti P, Cordons B, Falinover C, Gevers M. Control oriented low order modeling of a complex PWR plant: a comparison between open-loop and closed-loop methods. In: Proceedings of the 37th IEEE conference on decision and control. 1998, pp 3390–3395
4. Bendotti P, Cordons B, Falinover C, Gevers M. A comparison between model reduction and controller reduction: application to a pwr nuclear plant. In: Proceedings of the 38th IEEE conference on decision and control. 1999, pp 4625–4630
5. Bitmead RB. Iterative control design approaches. In: Proceedings of 12th IFAC World Congress, vol 9, 1993
6. Campi MC, Lecchini A, Pardo M, Savaresi, SM. Iterative feedback tuning: a direct approach. In: Mathematical theory of networks and systems, Padova, Italy. 1998, pp 719–722
7. Campi MC, Lecchini A, Savaresi SM. Virtual reference feedback tuning: a direct method for the design of feedback controllers. Automatica 2002; 38(8): 1337–1346
8. De Callafon RA, Van den Hof PMJ. Suboptimal feedback control by a scheme of iterative identification and control design. Math Model Syst 1997; 3(1): 77–101
9. Gevers M. Towards a joint design of identification and control? In: Trentekman HL and Willems C (eds) Essays on control: perspectives in the theory and its applications. 1993, pp 1735–1740
10. Gevers M. Identification for control. In: Proceedings of the 5th IFAC symposium on adaptive control and signal processing. 1995, pp 1–12
11. Guardabassi GO, Savaresi SM. Virtual reference direct design method: an off-line approach to data-based control system design. IEEE Trans Autom Control 2000; 45(5): 954–959
12. Guardabassi GO, Savaresi SM. Approximate linearization via feedback: an overview. Survey Paper Autom 2001; 37(1): 1–15

13. Hakvoort RG, Schrama R, Van den Hof PMJ. Approximate identification with closed-loop performance criterion and application to LQG feedback design. Automatica 1994; 30(4): 679–690
14. Hildebrand R, Lecchini A, Solari G, Gevers M. Convergence analysis and optimal prefiltering in iterative feedback tuning. 2002, submitted
15. Hjalmarsson H. Efficient tuning of linear multivariable controllers using iterative feedback tuning Int J Adaptive Control Signal Process 1999; 13: 553–572
16. Hjalmarsson H. Iterative feedback tuning – an overview. Int J Adaptive Control Signal Process, Special Issue: New ideas in recursive estimation and stochastic adaptive control 2002; 16(5): 373–395
17. Hjalmarsson H, Gevers M, Gunnarson S, Lequin O. Iterative feedback tuning: theory and applications. IEEE Control Syst Magazine 1998; 26–41
18. Hjalmarsson H, Gunnarson S, Gevers M. A convergent iterative restricted complexity control design scheme. In: Proceedings of the 33rd IEEE conference on decision and control. 1994, pp 1735–1740
19. Landau ID, Karimi A, Costantinescu A. Direct controller order reduction by identification in closed loop. Automatica 2001; 37: 1689–1702
20. Lecchini A, Campi MC, Savaresi SM. Sensitivity shaping via virtual reference feedback tuning. In: Proceedings of the 40th IEEE conference on decision and control. 2001, pp 750–755
21. Lecchini A, Campi MC, Savaresi SM. Virtual reference feedback tuning for two degree of freedom controllers. In: Proceedings of the european control conference – ECC2001, 2001
22. Lecchini A, Campi MC, Savaresi SM. Virtual reference feedback tuning for two degree of freedom controllers. Int J Adapt Control Signal Process, Special issue on controller design 2002; (16): 355–371
23. Lee WS, Anderson BDO, Kosut RL, Mareels IMY. A new approach to adaptive robust control. Int J Adaptive Control Signal Process 1993; 7: 183–211
24. Lee WS, Anderson BDO, Kosut RL, Mareels IMY. On robust performance improvement through the windsurfer approach to adaptive robust control. In: Proceedings of the 32th IEEE conference on decision and control. 1993, pp 2821–2827
25. Savaresi SM, Guardabassi GO. Approximate feedback linearization of discrete time non-linear systems using virtual input direct design. System & Control Lett 1997; 32: 63–67
26. Savaresi SM, Guardabassi GO. Approximate I/O feedback linearization of discrete-time non-linear systems via virtual input direct design. Automatica 1998; 34(6): 715–722
27. Schrama RJ. Accurate models for control design: the necessity of an iterative scheme. IEEE Trans Autom Control 1992; 37(7): 991–994
28. Spall JC, Criston JA. Model-free control of nonlinear stochastic systems with discrete-time measurements. IEEE Trans Autom Control 1998; 43(9): 1198–1210
29. Van den Hof PMJ, and Schrama R. Identification and control – closed loop issues. Automatica 1995; 32(12): 1751–1770
30. Zang Z, Bitmead RR, Gevers M. Iterative weighted least-squares identification and weighted LQG control. Automatica 1995; 31(11): 1577–1594

Appendix

We discuss why and to what extent $J_{\mathrm{VRFT}}(\theta)$ approximate $J_S(\theta)$.

We need the following definitions:

- *The ideal controller* $C_0(z)$. The ideal controller $C_0(z)$ is the controller that realizes perfect matching between the output-sensitivity and reference-model $S(z)$, namely:

$$C_0(z) = \frac{1}{P(z)} \frac{1 - S(z)}{S(z)}.$$

The controller $C_0(z)$ is only used as an analysis tool. In general, it is a high-order controller and does not belong to the selected controller class. Even more so, it is allowed to be an improper rational function, so that, strictly speaking, it is not a transfer function.

- *The extended class of controller* $C^+(\theta^+)$. Let

$$\Delta C(z) = C_0(z) - \beta^{\mathrm{T}}(z)\bar{\theta},$$

where $\bar{\theta}$ is the parameter vector which minimizes $J_S(\theta)$ and introduce the following extended vectors:

$$\beta^+(z) = [\beta_1(z) \quad \beta_2(z) \quad \cdots \quad \beta_n(z) \quad \Delta C(z)]^{\mathrm{T}},$$
$$\theta^+ = [\vartheta_1 \quad \vartheta_2 \quad \cdots \quad \vartheta_n \quad \vartheta_{n+1}]^{\mathrm{T}}.$$

Then, define an extended family of controllers as follows:

$$C^+(z,\theta^+) = \beta^{+^{\mathrm{T}}}(z)\theta^+.$$

Notice that $C_0(z) = C^+(z,\theta_0^+)$ with $\theta_0^+ = [\bar{\theta}^{\mathrm{T}} \quad 1]^{\mathrm{T}}$.

- *The extended costs*. The extended cost $J_S^+(\theta^+)$ is defined as

$$J_S^+(\theta^+) = \left\| \left[S(z) - \frac{1}{1 + P(z)C^+(z,\theta^+)} \right] W_S(z) \right\|_2^2.$$

Note that the difference between $J_S(\theta)$ and $J_S^+(\theta^+)$ is that $J_S^+(\theta^+)$ is parametrized by the extended family of controllers $\{C^+(z,\theta^+)\}$. Clearly, the parameter vector θ_0^+ is the global minimizer of $J_S^+(\theta^+)$.

In a similar way, we define $J_{\mathrm{VRFT}}^+(\theta^+)$ as

$$J_{\mathrm{VRFT}}^+(\theta) = \|L(z)F^+(z,\theta^+)G^+(z,\theta^+)\|_2^2,$$

where

$$F^+(z,\theta^+) := \left[S(z) - \frac{1}{1 + P(z)C^+(z,\theta^+)} \right] W_S(z),$$
$$G^+(z,\theta^+) := 1 + P(z)C^+(z,\theta^+).$$

Notice that, the two extended performance indices $J_S^+(\theta^+)$ and $J_{\mathrm{VRFT}}^+(\theta^+)$ share the same global minimizer (i.e. θ_0^+).

Using the notation introduced above, one could think of the problem of minimizing $J_S(\theta)$ as the problem of minimizing $J_S^+(\theta^+)$ subject to the constraint given by the controller structure (i.e. $\theta^+ = [\theta^{\mathrm{T}} \quad 0]^{\mathrm{T}}$) and the same thing holds for the minimisation of $J_{\mathrm{VRFT}}(\theta)$. Thus, the VRFT method corresponds to a constrained minimisation problem where the cost shares the same unconstrained minimizer with the original control cost.

In the following proposition, we show that if $L(z)$ is chosen to be equal to $S(z)$, the Hessians of $J_S^+(\theta^+)$ and $J_{\mathrm{VRFT}}^+(\theta^+)$ computed in θ_0^+ coincide. In this way, the two costs not only share the minimizer, but they also have the same shape around their common minimizer so that the constrained minimisation of $J_{\mathrm{VRFT}}^+(\theta^+)$ leads to an approximated constrained minimisation of $J_S^+(\theta^+)$.

Proposition A.1: If

$$L(z) = S(z),$$

then

$$\mathbf{H}[J_S^+(\theta^+)]_{|\theta^+=\theta_0^+} = \mathbf{H}[J_{\mathrm{VRFT}}^+(\theta^+)]_{|\theta^+=\theta_0^+},$$

where $\mathbf{H}$ denotes the Hessian.

Proof. In the following ∇ denotes gradient and $^-$ denotes complex conjugate. Moreover, we drop the argument $\mathrm{e}^{\mathrm{j}\omega}$ in order to simplify the notation.

We have

$$\begin{aligned}\mathbf{H}[J_S^+(\theta^+)] &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathbf{H}\Big[|F^+(\theta^+)|^2\Big]\,\mathrm{d}\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} 2\,\mathrm{Re}\Big\{\mathbf{H}[F^+(\theta^+)]\bar{F}^+(\theta^+) \\ &\quad +\nabla[F^+(\theta^+)]\nabla[\bar{F}^+(\theta^+)]^{\mathrm{T}}\Big\}\,\mathrm{d}\omega,\end{aligned}$$

so that, being $F^+(\theta_0^+) = 0$, we obtain

$$\mathbf{H}[J_S^+(\theta^+)]_{|\theta^+=\theta_0^+} = \frac{1}{2\pi}\int_{-\pi}^{\pi} 2\,\mathrm{Re}\Big\{\nabla[F^+(\theta^+)]_{|\theta^+=\theta_0^+}\nabla[\bar{F}^+(\theta^+)]^{\mathrm{T}}_{|\theta^+=\theta_0^+}\Big\}\,\mathrm{d}\omega. \tag{9}$$

As for $J_{\mathrm{VRFT}}^+(\theta^+)$, we have:

$$\begin{aligned}\mathbf{H}[J_{\mathrm{VRFT}}^+(\theta^+)] &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |L|^2\mathbf{H}\Big[|F^+(\theta^+)G^+(\theta^+)|^2\Big]\,\mathrm{d}\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |L|^2 2\,\mathrm{Re}\big\{\mathbf{H}[F^+(\theta^+)G^+(\theta^+)]\bar{F}^+(\theta^+)\bar{G}^+(\theta^+) \\ &\quad +\nabla[F^+(\theta^+)G^+(\theta^+)]\nabla[\bar{F}^+(\theta^+)\bar{G}^+(\theta^+)]^{\mathrm{T}}\Big\}\,\mathrm{d}\omega,\end{aligned}$$

from which, using again the fact that $F^+(\theta_0^+) = 0$, we obtain

$$\begin{aligned}\mathbf{H}[J_{\mathrm{VRFT}}^+(\theta^+)]_{|_{\theta^+=\theta_0^+}} &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |L||G^+(\theta_0^+)|^2 2\,\mathrm{Re}\Big\{\nabla[F^+(\theta^+)]_{|\theta^+=\theta_0^+} \\ &\quad \times \nabla[\bar{F}^+(\theta^+)]^{\mathrm{T}}_{|_{\theta^+=\theta_0^+}}\Big\}\,\mathrm{d}\omega\end{aligned} \tag{10}$$

Comparing (9) and (10), we conclude that, if $L(z) = G^+(\theta_0^+)^{-1}$, then the two Hessians evaluated in θ_0^+ are equal. On the other hand, by using the definition of $C_0(z)$ we also have

$$G^+(z, \theta_0^+) = 1 + P(z)C_0(z) = S(z)^{-1},$$

and this concludes the proof. □

A point is perhaps worth noticing in the previous proof. In principle, the filter that equalizes the two Hessians, viz. $L(z) = G^+(\theta_0^+)^{-1}$, depends on the unknown θ_0^+. However, such a dependence shows up in fsuch a fashion that $L(z)$ actually coincides with the known $S(z)$, a crucial fact for the filter implementability.