Moving Horizon Design of Discrete Coefficient FIR Filters

Daniel E. Quevedo, *Student Member, IEEE* and
Graham C. Goodwin, *Fellow, IEEE*

***Abstract*—We show how the discrete coefficient filter design problem can be solved with a moving horizon optimization approach. The computation time of this procedure is determined by the optimization horizon and does not grow exponentially with the filter length. $\Sigma\Delta$ design methods are a special case of the proposed procedure.**

***Index Terms*—Digital filter wordlength effects, least mean square methods, moving horizon optimization, optimization methods, quadratic programming, quantized coefficients, sigma-delta modulation.**

I. Introduction

In many hardware critical applications, the problem of approximating an infinite precision *target filter* $T(z)$ with a discrete coefficient one $H(z)$ arises. A straightforward solution to this problem is to synthesize $H(z)$ by simply rounding (quantizing) the corresponding coefficients of $T(z)$; see, e.g., [1]. Alternatively, in [2]–[7], $\Sigma\Delta$-modulation techniques have been utilized. The rationale behind these techniques relies on exploiting the *noise-shaping* capabilities of a closed loop. The adoption of a white quantization noise model allows the designer to *push* the finite wordlength artifacts to noncritical frequency bands by proper tuning of a feedback filter. As a consequence, the resulting filter $H(z)$ is more similar to $T(z)$ in important frequency bands and, thus, corresponds to a better design.

It is also possible to pose the problem directly in an optimization framework; see, e.g., [8]. Here, it is useful to introduce a frequency weighting function $W(e^{j\omega})$ and to minimize some measure of the weighted frequency response error $W(e^{j\omega})(T(e^{j\omega}) - H(e^{j\omega}))$. Most research efforts have dealt with linear-phase FIR filters and has concentrated on minimizing the peak weighted error ($\mathcal{L}_\infty$-norm) or the mean square norm $\mathcal{L}_2$ over a finite frequency grid.

In the case of the $\mathcal{L}_2$-norm, the resulting combinatorial optimization problem can be stated as an integer quadratic program. While, in principle, it can be solved exactly via tree search algorithms (see, e.g., [9]), these lead to prohibitive computation times for long finite impulse response (FIR) structures. Thus, more efficient procedures that yield suboptimal designs have been developed. Examples include relaxation methods [10], local searches [11], simulated evolution [12], and adaptations of recursive least squares [13].

In the present work, we propose a novel approach to the $\mathcal{L}_2$ discrete coefficient FIR filter design problem without being restricted to linear-phase structures. Rather than evaluating the objective function over a finite set of frequencies, we pose the approximation problem as an exact minimization in the time domain. This motivates us to formulate a practical iterative design procedure, where, at each step, a simpler quadratic program with finite set constraints is solved. The size of each of these programs depends on a design parameter: the *horizon*. This allows the designer to trade off the quality of the filter obtained versus the computational effort required. Larger horizons yield, in general, better designs. In the simplest case of a unitary horizon, the methodology proposed reduces to the $\Sigma\Delta$ encoders mentioned earlier. Thus, our approach establishes a link between optimization-based methods and $\Sigma\Delta$ ideas. Our procedure is also related to the approach proposed in [14], where a time-domain description of the quadratic cost is split into a set of partial costs. A distinguishing feature of our method is that it deploys moving horizon optimization, where exactly one coefficient is fixed at every optimization step. Moreover, unlike tree search and simulated annealing-based methods, the computation time does not increase exponentially in the filter length, making it especially suitable for long FIR filters.

Manuscript received July 8, 2003; revised March 9, 2004 and June 21, 2004. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Trac D. Tran.

The authors are with the School of Electrical Engineering and Computer Science, The University of Newcastle, Callaghan, NSW 2308, Australia (e-mail: dquevedo@ieee.org; eegcg@ee.newcastle.edu.au).

Digital Object Identifier 10.1109/TSP.2005.847856

II. Formulation of the Problem

In the sequel, we will use upper case boldface letters to denote matrices and lower case boldface letters to denote vectors. In particular, $\mathbf{I_n}$ denotes the identity matrix in $R^{n\times n}$. The superscript T refers to transposition. The symbol z is the argument of the $\mathcal{Z}$-transform, whereas ρ denotes the shift operator that characterizes recursions such as $\rho\mathbf{x}_k = \mathbf{x}_{k+1}$.

As foreshadowed in the introduction, we consider an FIR or stable infinite impulse response (IIR) discrete-time filter

$$T(z) = \sum_{i=0}^{\infty} t_i z^{-i}.$$

Our goal is to approximate this target via an FIR filter

$$H(z) = \sum_{i=0}^{M-1} h_i z^{-i} \tag{1}$$

where M is the filter impulse response length. Each of the coefficients h_i in (1) is restricted to belonging to a finite set of scalars[1] $\mathbb{U}$, i.e.,

$$h_j \in \mathbb{U}, \quad \forall\, j \in \{0, 1 \ldots, M-1\}. \tag{2}$$

Remark 1: Note that we do not need to specify the constraint set $\mathbb{U}$ further. Thus, the framework adopted encompasses various finite-set constraints on the coefficients. Examples include where a finite set of consecutive integers is used or where the coefficients are expressible as a finite sum of powers-of-two terms.

The associated discrete coefficient FIR filter design problem can be stated in an optimization framework by utilizing the following (frequency domain) $\mathcal{L}_2$ performance measure

$$V \triangleq \frac{1}{2\pi}\int_0^{2\pi} |W(e^{j\omega})(T(e^{j\omega}) - H(e^{j\omega}))|^2 d\omega \tag{3}$$

where $T(e^{jw})$ and $H(e^{jw})$ are the frequency responses of $T(z)$ and $H(z)$, respectively. In this cost function, we have included frequency weighting by means of the term $W(e^{j\omega})$. This filter weights the relative importance of the approximation error (ripple) in different frequency bands. Thus, the finite wordlength effect can be concentrated in tolerant bands and reduced in more critical bands. (If the power spectrum of the signal to be applied at the input to the filter is known, then $W(e^{j\omega})$ can be adjusted to include this effect.) We assume that $W(z)$ is of order $n \in \mathbb{N}$, stable, and described via

$$W(z) = d + \mathbf{c}^T(z\mathbf{I_n} - \mathbf{A})^{-1}\mathbf{b} \tag{4}$$

[1]We do not consider the design of an unconstrained filter gain explicitly.

or, equivalently, via

$$W(z) = \sum_{i=0}^{\infty} w_i z^{-i}$$

with[2]

$$w_0 = d, \quad w_i = \mathbf{c}^T \mathbf{A}^{i-1} \mathbf{b}, \quad i \geq 1. \tag{5}$$

In these two expressions, d is a scalar, $\mathbf{A} \in \mathbb{R}^{n \times n}$, and $\mathbf{c}$ and $\mathbf{b}$ are vectors in $\mathbb{R}^n$.

The discrete-coefficient filter design problem can thus be regarded as one of finding coefficients in (1) that yield the minimum *cost* V defined in (3). Minimizing V is a mixed-integer programming problem, which is, in general, an *NP-hard* combinatorial optimization problem. Its complexity is exponential in the filter impulse response length. As a consequence, direct minimization of V becomes impractical if $H(z)$ is allowed to have a large number of coefficients.

The key contribution of the present work is an algorithm that yields near-optimal discrete coefficient filter designs, without the need to solve the entire combinatorial optimization problem. To achieve this goal, we translate V into the *time*-domain by utilizing Parseval's theorem. This leads to

$$V = \sum_{i=0}^{\infty} e_i^2 \tag{6}$$

where the terms $\{e_i\}$ satisfy $\sum_{i=0}^{\infty} e_i z^{-i} = E(z)$, with

$$E(z) \triangleq W(z)(T(z) - H(z)). \tag{7}$$

From (4), the sequence $\{e_i\}$ can also be described as the output of a state-space system:

$$\begin{aligned} \mathbf{x}_{i+1} &= \mathbf{A}\mathbf{x}_i + \mathbf{b}(t_i - h_i) \\ e_i &= \mathbf{c}^T \mathbf{x}_i + d(t_i - h_i) \end{aligned} \tag{8}$$

where $\mathbf{x}_i \in \mathbb{R}^n$ is the system state, and $e_i \in \mathbb{R}$.

Expressions (3) and (6) are equivalent, and in principle, their minimization over all possible finite set constrained coefficients h_j, $j \in \{0, \ldots, M-1\}$ requires a similar computational effort. However, in the sequel, we will further embellish the time-domain description (6), leading to a simpler computational problem.

III. Moving Horizon Optimization

In this section, we develop a practical method for designing discrete-coefficient filters by posing the problem as one of minimizing a set of constrained quadratic programs of moderate size.

The cost function (6) motivates us to develop an iterative procedure to optimize the filter coefficients h_j. It is based on the fact that the effect of h_k on distant values of e_{k+i} can often be neglected. Following this idea, we propose to fix a relatively small *horizon* N, where $1 \leq N < M$, and to consider the following set of finite horizon cost functions:

$$V_k \triangleq \sum_{j=k}^{k+N-1} e_j^2, \quad k \in \{0, 1, \ldots, M-N\}. \tag{9}$$

Thus, we have replaced V by a set of finite horizon costs V_k. Each of these costs takes into account overlapping windows of data e_j^2 (compare with the formulation in [14]) and examines the approximation error

[2]Note that, if $\boldsymbol{W(z)}$ is FIR, then $\boldsymbol{w_i = 0, \forall\, i > n}$.

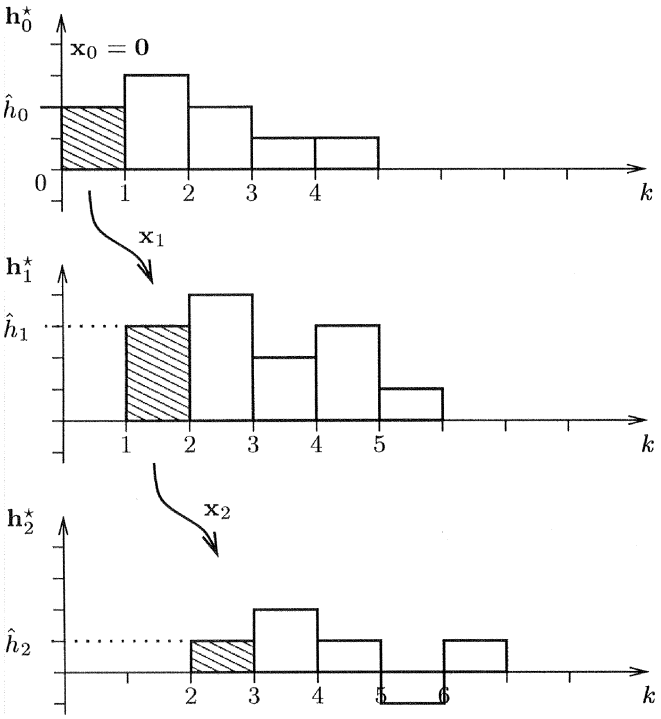

Fig. 1. Moving horizon principle $N = 5$.

that results when deciding on N coefficients of $H(z)$. These decision variables can be gathered into the vectors

$$\begin{aligned} \mathbf{h}_k &\triangleq [\, h_k \quad h_{k+1} \quad \ldots \quad h_{k+N-1} \,]^T \\ & k \in \{0, 1, \ldots, M-N\}. \end{aligned}$$

We will write $V_k(\mathbf{h}_k)$ in order to make this dependence explicit.

In accordance with the constraint (2), given the state $\mathbf{x}_k$ [see (8)], the resulting optimization problem corresponds to finding

$$\mathbf{h}_k^\star \triangleq \arg \min_{\mathbf{h}_k \in \mathbb{U}^N} V_k(\mathbf{h}_k) \tag{10}$$

where the set $\mathbb{U}^N \subset \mathbb{R}^N$ is defined via the Cartesian product

$$\mathbb{U}^N \triangleq \mathbb{U} \times \cdots \times \mathbb{U}.$$

Although the vector $\mathbf{h}_k^\star$ obtained from the finite-set constrained quadratic program (10) contains N coefficients, we only fix its first element, namely

$$\hat{h}_k \triangleq [\, 1 \quad 0 \quad \ldots \quad 0 \,] \mathbf{h}_k^\star. \tag{11}$$

This value is utilized in the design of $H(z)$ by setting

$$h_k \longleftarrow \hat{h}_k \tag{12}$$

[see (1)]. Furthermore, $h_k^\star$ also yields the successor state $\mathbf{x}_{k+1}$ via

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{b}(t_k - \hat{h}_k) \tag{13}$$

which follows from (8).

Having fixed h_k, at the next step, a new optimization is carried out using the updated state $\mathbf{x}_{k+1}$ and cost $V_k(\mathbf{h}_{k+1})$, yielding $\mathbf{h}_{k+1}^\star$, etc. As can be seen in Fig. 1, which depicts the startup of the procedure, the window, which is here of fixed size $N = 5$, slides (or moves) forward at each optimization step.

Thus, the methodology proposed here forms an iterative procedure to optimize the filter coefficients. It mirrors the strategy deployed in some predictive control architectures, where the ℓ_∞-norm of decision variables needs to be bounded;[3] see e.g., [16].

In order to provide the entire filter $H(z)$, the procedure starts with $k = 0$ and $\mathbf{x}_0 = \mathbf{0}$, since all filters are causal; see (6). It finishes at $k = M - N$, after $M - N + 1$ minimization steps. At this last step, N coefficients have been obtained by setting

$$[\, h_{M-N} \quad h_{M-N+1} \quad \dots \quad h_{M-1} \,] \longleftarrow (\mathbf{h}^\star_{M-N})^T .$$

Since the discrete optimization problem (10) involves only N decision variables, in the case of large M, the complexity of the computations required is significantly lower than that of the original problem of minimizing the infinite-horizon cost V defined in (3) via an exhaustive search. More precisely, the computational complexity of the design procedure proposed here is only linear in $(M - N)$ and exponential in the horizon length N. This should be contrasted with direct minimization of V, which requires a number of computations and is exponential in the filter length M! As a consequence, the new procedure is especially useful when long impulse response filters are to be designed. In principle, choosing larger values of N will provide a better approximation to the target filter $T(z)$. However, as illustrated by means of the example included in Section VI, a *good* filter may often be obtained with a relatively small horizon, hence, at the expense of only very modest computational time.

Since the complexity of minimizing V_k is only moderate for small horizons N, our proposal here is to solve for $\mathbf{h}^\star_k$ without any further approximations. To that extent, in the following section, we will provide an expression for the optimizer (10), which allows us to implement the optimization procedure in a simple manner and to establish the relationship with existing design methods.

Remark 2 (Extension to IIR Filters): While the main focus of the present work lies in the design of FIR filters, the proposed method can be extended to the IIR case. Details are included in Appendix A.

IV. Closed-Loop Implementation

The solution of (10) requires one to solve a quadratic program with finite-set constraints. This can be done, in principle, by utilizing Dynamic Programming recursions; see, e.g., [11]. More conveniently, the solution $\mathbf{h}^\star_k$ can be characterized by means of Lemma 1 stated below. It makes use of the following definition of a vector quantizer.

Definition 1 (Nearest Neighbor Quantizer): Given a countable set of vectors $\mathcal{B} = \{\mathbf{b_1}, \mathbf{b_2}, \dots\} \subset \mathbb{R}^{n_\mathcal{B}}$, the nearest neighbor quantizer is defined as a mapping $q_\mathcal{B}: \mathbb{R}^{n_\mathcal{B}} \to \mathcal{B}$ that assigns to each vector $\mathbf{f} \in \mathbb{R}^{n_\mathcal{B}}$ the closest element of $\mathcal{B}$ (as measured by the Euclidean norm), i.e., $q_\mathcal{B}(\mathbf{f}) = \mathbf{b} \in \mathcal{B}$ if and only if $\mathbf{b}$ satisfies

$$(\mathbf{f} - \mathbf{b})^T(\mathbf{f} - \mathbf{b}) \le (\mathbf{f} - \mathbf{b_j})^T(\mathbf{f} - \mathbf{b_j}), \quad \forall\, \mathbf{b}_j \in \mathcal{B}.$$

Note that in the scalar case ($n_\mathcal{B} = 1$), this definition allows one to characterize the procedure of direct rounding of coefficients as analyzed, e.g., in [1] via

$$h_j \longleftarrow q_\cup(t_j), \quad \forall\, j \in \{0, \dots, M-1\}. \tag{14}$$

Lemma 1: Suppose $\mathbb{U}^N = \{\mathbf{v_1}, \mathbf{v_2}, \dots, \mathbf{v_r}\}$. Then, $\mathbf{h}^\star_k$ in (10) is given by

$$\mathbf{h}^\star_k = \mathbf{\Phi}^{-1} q_{\widetilde{\mathbb{U}}^N}(\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k), \quad \text{where:} \tag{15}$$

[3]Recently, we have been working on extending the framework to finite-set constraints; see, e.g., [15].

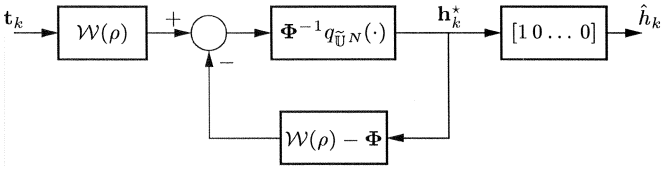

Fig. 2. Closed-loop implementation of the optimization procedure.

$$\mathbf{t}_k \triangleq \begin{bmatrix} t_k \\ t_{k+1} \\ \vdots \\ t_{k+N-1} \end{bmatrix}, \quad \mathbf{\Gamma} \triangleq \begin{bmatrix} \mathbf{c}^T \\ \mathbf{c}^T\mathbf{A} \\ \vdots \\ \mathbf{c}^T\mathbf{A}^{N-1} \end{bmatrix}$$
$$\mathbf{\Phi} \triangleq \begin{bmatrix} d & 0 & \dots & 0 \\ w_1 & d & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ w_{N-1} & \dots & w_1 & d \end{bmatrix} \tag{16}$$

and where w_i are the impulse responses of $W(z)$ included in (5).

The nonlinearity $q_{\widetilde{\mathbb{U}}^N}(\cdot)$ is a nearest neighbor quantizer, as described in Definition 1. The image of this mapping is the set

$$\widetilde{\mathbb{U}}^N \triangleq \{\widetilde{\mathbf{v_1}}, \widetilde{\mathbf{v_2}}, \dots, \widetilde{\mathbf{v_r}}\} \subset \mathbb{R}^N, \quad \text{with: } \widetilde{\mathbf{v}}_\mathbf{i} = \mathbf{\Phi}\mathbf{v_i}, \mathbf{v_i} \in \mathbb{U}^N. \tag{17}$$

Proof: The proof is included in Appendix B. ∎

The characterization given in Lemma 1 allows us to implement the iterative design method presented in Section III (not the filter $H(z)$) as the closed loop depicted in Fig. 2. In this figure, the filter $\mathcal{W}(\rho)$ has N inputs and outputs and is defined as

$$\mathcal{W}(\rho) \triangleq \mathbf{\Phi} + \mathbf{\Gamma}(\rho\mathbf{I_n} - \mathbf{A})^{-1}\mathbf{b}[\,1 \quad 0 \quad \dots \quad 0\,].$$

Note that $\mathcal{W}$ depends only on the impulse responses of W and not on the particular realization chosen in (4). This circuit summarizes the main contribution of this work.

Lemma 1 allows us to establish the relationship that exists to some schemes described in the literature, as detailed in Section V.

V. Relationship to ΣΔ-Modulation Encoders

As a special case, consider a unitary horizon, namely, $N = 1$ and a filter $W(z)$ with unitary feed-through, i.e., with $d = 1$. In this simple case, the vectors and matrices defined in (16) simplify to $\mathbf{t}_k = t_k$, $\mathbf{\Gamma} = \mathbf{c}^T$, and $\mathbf{\Phi} = 1$. Moreover, from (11), it follows that $\mathbf{h}^\star_k = \hat{h}_k$, and the set $\widetilde{\mathbb{U}}^N$ defined in (17) reduces to $\mathbb{U}$. Thus, the result (15) gives

$$\hat{h}_k = q_\mathbb{U}(t_k + \mathbf{c}^T\mathbf{x}_k).$$

On the other hand, (4), (12), and (13) yield that

$$\mathbf{c}^T\mathbf{x}_k = (W(\rho) - 1)(t_k - \hat{h}_k).$$

Therefore, the proposed method satisfies

$$\hat{h}_k = q_\mathbb{U}(W(\rho)t_k - (W(\rho) - 1)\hat{h}_k) \tag{18}$$

and can be implemented as in Fig. 3. As a consequence, in the *horizon-one* case, our scheme is equivalent to the ΣΔ-Modulation scheme utilized in [6] for the design of discrete coefficient FIR filters. It also embraces [7], where $W(\rho)$ is restricted to be of the form $W(\rho) = \left(1 + \sum_{i=1}^{p-1} b_i \rho^{-i}\right)^{-1}$ and (save for a unitary delay element) the single- and double-loop structures of [2]–[5], in which case, $W(\rho) = (1 - \rho^{-1})^{-1}$ and $W(\rho) = (1 - \rho^{-1})^{-2}$, respectively.[4]

[4]Minor differences exist with respect to initialization procedures [4]–[7] and, in some cases [2], [3] the use of oversampling.

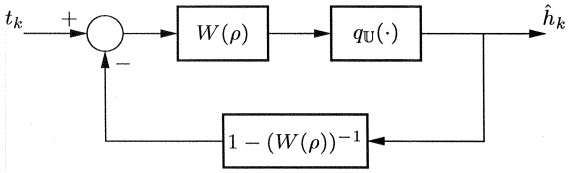

Fig. 3. Closed-loop implementation of the horizon-one case.

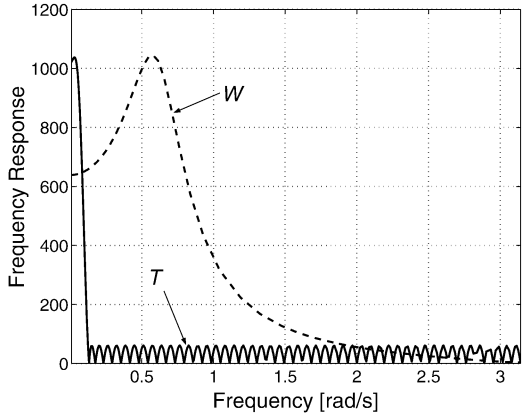

Fig. 4. Frequency response of the target filter (solid) $T(z)$ and of the frequency weighting filter (dashed) $W(z)$.

The relationship established above (which incidentally is related to the work described by us in [17]) allows us to reinterpret the $\Sigma\Delta$-Modulation schemes from an optimization based point of view. We see that they are embedded in the more general design scheme proposed here. The main advantage of using $N > 1$ follows from the fact that, with larger horizons, more information is taken into account in the coefficient allocation process. As a consequence, the proposed scheme with $N > 1$ will typically give rise to better filters than those provided by $\Sigma\Delta$-based methods.

VI. Example

Suppose that the target filter $T(z)$ is an equiripple lowpass FIR filter of length 100 generated by the Parks–McClellan algorithm and scaled, such that its coefficients satisfy $-32 \leq t_i \leq 32, \forall\, i \in \{0, \ldots, 99\}$. The frequency weighting is given by

$$W(z) = \frac{z^2 + 0.91z}{z^2 - 1.335z + 0.664}.$$

This filter has been used to model the human ear's sensitivity to low-level noise power (at 44.1 [kHz] sampling rate); see, e.g., [17]. Fig. 4 shows the frequency response of both filters.

Given the constraint set

$$\mathbb{U} = \{-32, -31, \ldots, 31\}$$

we synthesize FIR filters of length $M = 100$ with moving horizons of $N = 1$, 2 and 3. Fig. 5 displays the achieved costs V; see (3). Here, $N = 0$ denotes direct quantization of the coefficients as in (14). As can be seen in Fig. 5, the design with $N = 1$ (or, equivalently, with $\Sigma\Delta$-encoding) is better than the filter obtained by direct quantization. Both filters are outperformed by the designs generated with our procedure with larger horizons.

Further insight can be gained by inspecting the frequency distribution of the approximation error in Figs. 6 and 7. Fig. 6 depicts the frequency response of the filtered error function $E(z)$ defined in (7) and illustrates that the filter designed with $N = 3$ is indeed *closer* to $T(z)$ than the filter obtained by directly quantizing coefficients. As is apparent from the frequency responses of the unfiltered error $T(z) - H(z)$ included in Fig. 7, this performance increase is accomplished by concentrating the approximation error mostly in the *less-important* higher frequencies, as dictated by the weighting function $W(z)$.

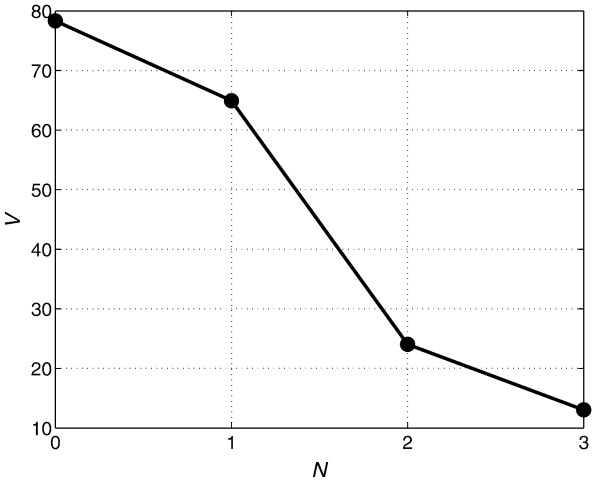

Fig. 5. Achieved cost as a function of the horizon N.

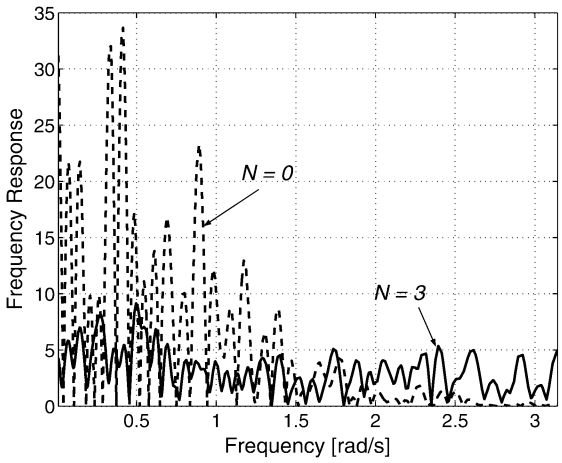

Fig. 6. Frequency response of the filtered error function $E(z)$ for (solid) $N = 3$ and for (dashed) direct quantization.

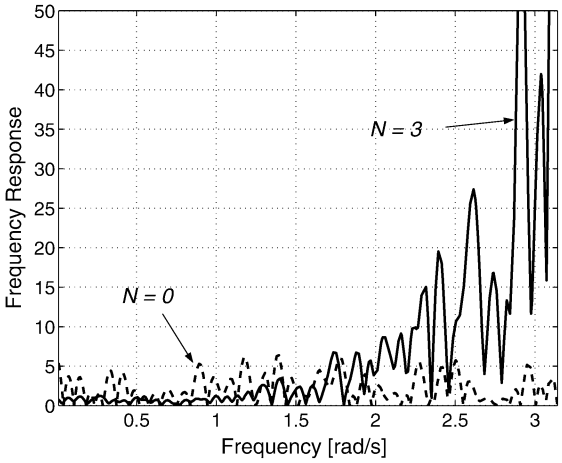

Fig. 7. Frequency response of the unfiltered error $T(z) - H(z)$ for (solid) $N = 3$ and for (dashed) direct quantization.

VII. Conclusions

This correspondence has introduced a novel methodology for the design of discrete coefficient FIR filters. In particular, we have formulated the problem as a moving horizon time domain optimization problem. This leads to a practical procedure that provides a near-optimal solution with low computational complexity. The method is suitable for the design of long FIR filters. The design method can be implemented as a closed loop and includes $\Sigma\Delta$ encoders as a special case. The scheme typically provides better performance than $\Sigma\Delta$-based approaches.

Appendix A
Treatment of IIR Filters

We will briefly outline how to treat the design of rational filters with discrete coefficients. For that purpose, suppose that the (stable) target filter $T(z)$ is described via

$$T(z) = \frac{\sum_{i=0}^{m} t_{bi} z^{m-i}}{\sum_{i=0}^{m} t_{ai} z^{m-i}} = \frac{T_B(z)}{T_A(z)}$$

and the discrete coefficient filter to be designed is parameterized as

$$H(z) = \frac{\sum_{i=0}^{m} h_{bi} z^{m-i}}{\sum_{i=0}^{m} h_{ai} z^{m-i}} = \frac{H_B(z)}{H_A(z)}. \tag{19}$$

The design problem now consists of minimizing the cost function (3) by choosing the $2m$ coefficients in (19). As before, each of these parameters is restricted to belong to the set $\mathbb{U}$.

This problem can be translated into the present framework via some straightforward approximations. To that extent, we define

$$\Delta_B(z) \triangleq H_B(z) - T_B(z)$$
$$\Delta_A(z) \triangleq H_A(z) - T_A(z)$$

so that[5]

$$\begin{aligned} T - H &= \frac{T_B}{T_A} - \frac{T_B + \Delta_B}{T_A + \Delta_A} = \frac{T_B}{T_A} - \frac{T_B + \Delta_B}{T_A(1 + (T_A)^{-1}\Delta_A)} \\ &\approx \frac{T_B}{T_A} - \frac{(T_B + \Delta_B)(1 + (T_A)^{-1}\Delta_A)}{T_A} \\ &\approx -\frac{\Delta_B}{T_A} - \frac{T_B \Delta_A}{(T_A)^2}. \end{aligned}$$

If we define now the two-input one-output filter

$$\overline{W} \triangleq -W \begin{bmatrix} T_A^{-1} & T_B T_A^{-2} \end{bmatrix}$$

and the one-input two-output filters

$$\overline{T} \triangleq \begin{bmatrix} T_B & T_A \end{bmatrix}^T$$
$$\overline{H} \triangleq \begin{bmatrix} H_B & H_A \end{bmatrix}^T$$

then

$$W(T - H) \approx \overline{W}(\overline{H} - \overline{T}).$$

As a consequence, the cost V in (6) can be approximated by J, which is defined as

$$J \triangleq \frac{1}{2\pi} \int_0^{2\pi} |\overline{W}(e^{j\omega})(\overline{T}(e^{j\omega}) - \overline{H}(e^{j\omega}))|^2 d\omega.$$

[5]For ease of notation, we omit the argument $\boldsymbol{z}$ in these expressions.

The methodology described in Section III can now be readily applied to this cost function. The only difference resides in the fact that, in the IIR case, the decision variables are vector valued. This presents no conceptual difficulties. In particular, (8) should simply be replaced by

$$\begin{aligned} \overline{\mathbf{x}}_{i+1} &= \overline{\mathbf{A}}\overline{\mathbf{x}}_i + \overline{\mathbf{B}}(\overline{\mathbf{t}}_i - \overline{\mathbf{h}}_i) \\ \overline{e}_i &= \overline{\mathbf{c}}^T \overline{\mathbf{x}}_i + \overline{\mathbf{d}}^T (\overline{\mathbf{t}}_i - \overline{\mathbf{h}}_i) \end{aligned}$$

where $\overline{\mathbf{x}}_i \in \mathbb{R}^{\overline{n}}$ is the state vector in

$$\overline{W}(z) = \overline{\mathbf{d}}^T + \overline{\mathbf{c}}^T (z\mathbf{I}_{\overline{\mathbf{n}}} - \overline{\mathbf{A}})^{-1} \overline{\mathbf{B}}$$

and

$$\overline{\mathbf{t}}_i \triangleq \begin{bmatrix} t_{bi} & t_{ai} \end{bmatrix}^T$$
$$\overline{\mathbf{h}}_i \triangleq \begin{bmatrix} h_{bi} & h_{ai} \end{bmatrix}^T.$$

Each of the vectors $\overline{\mathbf{h}}_i$ is restricted to belong to $\mathbb{V} \triangleq \mathbb{U}^2$, and the decision variables in the resulting moving horizon approach are now constrained to $\mathbb{V}^N$, rather than to $\mathbb{U}^N$. Note that as in (8), $\{\overline{e}_i\}$ are scalars.

Appendix B
Proof of Lemma 1

The cost function V_k defined in (9) can be written in vector form as $V_k = \mathbf{e}_k^T \mathbf{e}_k$, with

$$\mathbf{e}_k \triangleq \begin{bmatrix} e_k & e_{k+1} & \dots & e_{k+N-1} \end{bmatrix}.$$

Iteration of (8) yields

$$\mathbf{e}_k = \mathbf{\Phi}(\mathbf{t}_k - \mathbf{h}_k) + \mathbf{\Gamma}\mathbf{x}_k$$

so that

$$\begin{aligned} V_k(\mathbf{h}_k) = (\mathbf{\Phi}\mathbf{h}_k)^T \mathbf{\Phi}\mathbf{h}_k - 2(\mathbf{\Phi}\mathbf{h}_k)^T(\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k) \\ + (\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k)^T(\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k). \end{aligned} \tag{20}$$

We introduce the change of variables

$$\mathbf{p}_k \triangleq \mathbf{\Phi}\mathbf{h}_k$$

which transforms the set $\mathbb{U}^N$ into $\widetilde{\mathbb{U}}^N$, which is defined in (17). Expression (20) then allows us to characterize the optimizer $\mathbf{h}_k^\star$ as

$$\mathbf{h}_k^\star = \mathbf{\Phi}^{-1} \arg \min_{\mathbf{p}_k \in \widetilde{\mathbb{U}}^N} \varphi_k(\mathbf{p}_k) \tag{21}$$

where

$$\varphi_k(\mathbf{p}_k) \triangleq \mathbf{p}_k^T \mathbf{p}_k - 2(\mathbf{p}_k)^T(\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k).$$

The level sets of φ_k are spheres in $\mathbb{R}^N$ centered at $\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k$. Therefore, the constrained optimizer is given by

$$\arg \min_{\mathbf{p}_k \in \widetilde{\mathbb{U}}^N} \varphi_k(\mathbf{p}_k) = q_{\widetilde{\mathbb{U}}^N}(\mathbf{\Phi}\mathbf{t}_k + \mathbf{\Gamma}\mathbf{x}_k).$$

Equation (21) then establishes the Lemma.

References

[1] D. S. K. Chan and L. R. Rabiner, "Analysis of quantization errors in the direct form for finite impulse response digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AE-21, no. 4, pp. 354–366, Aug. 1973.

[2] P. W. Wong and R. M. Gray, "FIR filters with sigma-delta modulation encoding," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 6, pp. 979–990, Jun. 1990.

[3] P. W. Wong, "Fully sigma-delta modulation encoded FIR filters," *IEEE Trans. Signal Process.*, vol. 40, no. 6, pp. 1605–1610, Jun. 1992.

[4] S. R. Powell and P. M. Chau, "Multiplierless FIR filters with discrete-value sigma-delta encoded coefficients," in *Conf. Rec. Twenty-Fifth Asilomar Conf. Signals, Syst., Comput.*, vol. 2, 1991, pp. 1010–1014.
[5] ——, "Efficient narrowband FIR and IFIR filters based on powers-of-two sigma-delta coefficient truncation," *IEEE Trans. Circuits Syst. II*, vol. 41, no. 8, pp. 497–505, Aug. 1994.
[6] C. L. Chen and A. N. Willson, "Higher order Σ-Δ modulation encoding for design of multiplierless FIR filters," *Electron. Lett.*, vol. 34, no. 24, pp. 2298–2230, Nov. 1998.
[7] J. J. Nielsen, "Design of linear-phase direct-form FIR digital filters with quantized coefficients using error spectrum shaping," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 1020–1026, Jul. 1989.
[8] Y. C. Lim, S. R. Parker, and A. G. Constantinides, "Finite word length FIR filter design using integer programming over a discrete coefficient space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-30, no. 4, pp. 661–664, Aug. 1982.
[9] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon an LMS criteria," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 723–739, Oct. 1983.
[10] W.-Y. Yan and K. L. Teo, "Optimal finite-precision approximation of FIR filters," *Signal Process.*, vol. 82, pp. 1695–1705, 2002.
[11] N. Benvenuto, L. Franks, and F. Hill Jr, "Dynamic programming methods for designing FIR filters using coefficients -1, 0, and $+1$," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, no. 4, pp. 785–792, Aug. 1986.
[12] S. S. Rao and A. Ramasubrahmanyan, "Design of discrete coefficient FIR filters by simulated evolution," *IEEE Signal Process. Lett.*, vol. 3, no. 5, pp. 137–140, May 1996.
[13] G. Evangelista, "Least mean squared error-design of complex FIR filters with quantized coefficients," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 778–784, Aug. 2001.
[14] K. Nakayama, "A discrete optimization method for high-order FIR filters with finite wordlength coefficients," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 8, pp. 1215–1217, Aug. 1987.
[15] D. E. Quevedo, G. C. Goodwin, and J. A. De Doná, "Finite constraint set receding horizon control," *Int. J. Robust Nonlin. Contr.*, vol. 14, no. 4, pp. 355–377, Mar. 2004.
[16] J. M. Maciejowski, *Predictive Control With Constraints*. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[17] G. C. Goodwin, D. E. Quevedo, and D. McGrath, "Moving-horizon optimal quantizer for audio signals," *J. Audio Eng. Soc.*, vol. 51, no. 3, pp. 138–149, Mar. 2003.

Stability Analysis of 2-D Digital Filters With Saturation Arithmetic: An LMI Approach

Haranath Kar and Vimal Singh

***Abstract*—An improved LMI-based criterion for the nonexistence of overflow oscillations in two–dimensional (2-D) digital filters described by the Roesser model employing saturation arithmetic is presented. The criterion makes use of the structural properties (as prevailing in the system under consideration) of the saturation nonlinearities in a greater detail than the usual sector restriction $[0, 1]$.**

***Index Terms*—Asymptotic stability, digital filter wordlength effects, Lyapunov methods, multidimensional digital filters.**

I. Introduction

Two-dimensional (2-D) systems have found many applications such as image processing, seismographic data processing, thermal processes, gas absorbtion, water stream heating, etc. [1]. Thus, the design of 2-D systems is an interesting and challenging problem. When designing discrete systems using fixed-point arithmetic, one encounters quantization and overflow nonlinearities. The presence of such nonlinearities may result in the instability of the designed system. The quantization and overflow nonlinearities may interact with each other. However, if the number of quantization steps is large or, in other words, the internal wordlength is sufficiently long, then they can be regarded as decoupled or noninteracting and can be investigated separately. Under this decoupling approximation, quantization effects may be neglected when studying the effects of overflow [2]–[4].

This correspondence deals with the problem of global asymptotic stability of zero-input 2-D digital filters described by the Roesser model [5] using saturation overflow arithmetic. It will be assumed that the effects of quantization are negligible. Specifically, consider the state-space quarter-plane model given by (1a)–(1e), shown at the bottom of the next page, where $\boldsymbol{x}^h \in \boldsymbol{R}^m$, $\boldsymbol{x}^v \in \boldsymbol{R}^n$, $\boldsymbol{A}_{11} \in \boldsymbol{R}^{m\times m}$, $\boldsymbol{A}_{12} \in \boldsymbol{R}^{m\times n}$, $\boldsymbol{A}_{21} \in \boldsymbol{R}^{n\times m}$, $\boldsymbol{A}_{22} \in \boldsymbol{R}^{n\times n}$, and T denotes the transpose. The saturation nonlinearities given by

$$f_i^h\left(y_i^h(k,l)\right) = \begin{cases} 1, & y_i^h(k,l) > 1 \\ y_i^h(k,l), & \left|y_i^h(k,l)\right| \le 1 \\ -1, & y_i^h(k,l) < -1 \end{cases} \quad i = 1,\ldots,m \tag{1f}$$

$$f_i^v\left(y_i^v(k,l)\right) = \begin{cases} 1, & y_i^v(k,l) > 1 \\ y_i^v(k,l), & |y_i^v(k,l)| \le 1 \\ -1, & y_i^v(k,l) < -1 \end{cases} \quad i = 1,\ldots,n \tag{1g}$$

are under consideration. It is understood that system (1) has a finite set of initial conditions, i.e., there exist two positive integers K and L such that [4], [6]

$$\boldsymbol{x}^v(k,l) = \boldsymbol{0}, \quad k \ge K; \quad \boldsymbol{x}^h(k,l) = \boldsymbol{0}, \quad l \ge L. \tag{1h}$$

Manuscript received May 7, 2003; revised June 21, 2004. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Trac D. Tran.

H. Kar was with the Department of Electrical-Electronics Engineering, Atilim University, Ankara 06836, Turkey. He is now with the Department of Electronics Engineering, Motilal Nehru National Institute of Technology, Allahabad 211004, India (e-mail: hnkar1@rediffmail.com).

V. Singh is with the Department of Electrical-Electronics Engineering, Atilim University, Ankara 06836, Turkey (e-mail: vsingh11@rediffmail.com).

Digital Object Identifier 10.1109/TSP.2005.847857