Jointly Optimized Error-Feedback and Realization for Roundoff Noise Minimization in State-Space Digital Filters

Wu-Sheng Lu
Dept. of Elec. and Comp. Eng.
University of Victoria, Canada
E-mail: wslu@ece.uvic.ca

Takao Hinamoto
Graduate School of Engineering
Hiroshima University, Japan
E-mail: hinamoto@ecl.sys.hiroshima-u.ac.jp

Abstract

Roundoff noise (RN) is known to exist in digital filters and systems under finite-precision operations and can become a critical factor for severe performance degradation in IIR filters and systems. Two classes of methods are available for RN reduction or minimization — one uses state-space coordinate transformation, the other uses error feedback of state variables. In this paper, we propose a method for the joint optimization of error feedback and state-space realization. It is shown that the problem at hand can be solved in an unconstrained optimization setting. With a closed-form formula for gradient evaluation and an efficient quasi-Newton solver, the unconstrained minimization problem can be solved efficiently.

1. Introduction

Since the work of [1][2], it has been well understood that the roundoff noise (RN) of infinite-impulse-response (IIR) digital filters under fixed-point arithmetic operations can be substantially reduced by using adequately chosen state-space realizations [3]–[5]. It has also been known that RN reduction can be accomplished by feeding the quantization error of state variables back to the filter's input through a memoryless (constant) *error-feedback* matrix without affecting the filter's input-output characteristics [6]–[8]. The success of these techniques leads to the consideration of a *joint optimization* of error feedback and state-space realization so as to achieve greater reduction in RN. It turns out that obtaining such a jointly optimized error feedback and realization requires the solution of a sophisticated constrained nonlinear minimization problem. In [8], an iterative algorithm for the above optimization problem was proposed for IIR filters with *scalar* error-feedback matrices, but it appears to be inherently difficult to extend the algorithm to the cases where the error-feedback matrices are diagonal or general.

In this paper, the problem of joint optimization of error-feedback and state-space realization for RN minimization is investigated in a general nonlinear optimization framework where the error-feedback matrix can be a scalar, diagonal, or general matrix. Using linear-algebraic techniques, we convert the constrained optimization problem at hand into an unconstrained problem which can be solved using powerful quasi-Newton algorithms [10]. A nice feature of employing a general optimization setting for our problem is that both the realization optimization [1][2] and the error-feedback-matrix optimization [8] become special cases of the proposed formulation that explains why digital filters with jointly optimal error feedback and realization always outperform previously reported systems.

2. Preliminaries

Let $(\boldsymbol{A},\ \boldsymbol{b},\ \boldsymbol{c},\ d)_n$ be a minimal state-space realization of a stable IIR digital filter of order n. This realization can be expressed as

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{b}u(k) \tag{1a}$$
$$y(k) = \boldsymbol{c}\boldsymbol{x}(k) + du(k) \tag{1b}$$

where $\boldsymbol{A} \in \mathcal{R}^{n\times n}$, $\boldsymbol{b} \in \mathcal{R}^{n\times 1}$, $\boldsymbol{c} \in \mathcal{R}^{1\times n}$, and $d \in \mathcal{R}$. Now assume that the filter is implemented subject to finite-word-length (FWL) constraint and quantization takes place before matrix-vector multiplications, and an error-feedback for state variables is used for the sake of RN reduction, then the filter's model becomes [8]

$$\tilde{\boldsymbol{x}}(k+1) = \boldsymbol{A}Q[\tilde{\boldsymbol{x}}(k)] + \boldsymbol{b}u(k) + \boldsymbol{D}\boldsymbol{e}(k) \tag{2a}$$
$$\tilde{y}(k) = \boldsymbol{c}Q[\tilde{\boldsymbol{x}}(k)] + du(k) \tag{2b}$$

where $Q[\cdot]$ denotes the quantizer that rounds the fraction of each input component to a b-bit number, $\boldsymbol{e}(k)$ is the quantization error defined by

$$\boldsymbol{e}(k) = \tilde{\boldsymbol{x}}(k) - Q[\tilde{\boldsymbol{x}}(k)]$$

and $\boldsymbol{D}$ is referred to an error-feedback matrix. Fig. 1 shows a block diagram of the state-space filter described by (2).

From (1) and (2), the roundoff noise for the filter can be modeled as

$$\Delta\boldsymbol{x}(k+1) = \boldsymbol{A}\Delta\boldsymbol{x}(k) + (\boldsymbol{A}-\boldsymbol{D})\boldsymbol{e}(k) \tag{3a}$$
$$\Delta y(k) = \boldsymbol{c}\Delta\boldsymbol{x}(k) + \boldsymbol{c}\boldsymbol{e}(k) \tag{3b}$$

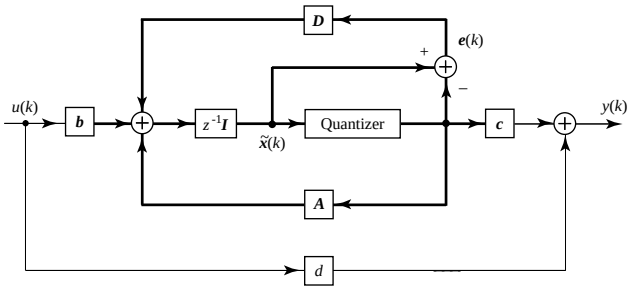

Figure 1. Error feedback in a state-space digital filter.

where $\Delta \boldsymbol{x}(k) = \boldsymbol{x}(k) - \tilde{\boldsymbol{x}}(k)$ and $\Delta y(k) = y(k) - \tilde{y}(k)$. In the frequency domain, the noise process is modeled by

$$\Delta Y(z) = \boldsymbol{G}_D(z)\boldsymbol{E}(z) \tag{4a}$$

$$\boldsymbol{G}_D(z) = \boldsymbol{c}(z\boldsymbol{I} - \boldsymbol{A})^{-1}(\boldsymbol{A} - \boldsymbol{D}) + \boldsymbol{c} \tag{4b}$$

where $\Delta Y(z)$ and $\boldsymbol{E}(z)$ are the z-transforms of $\Delta y(k)$ and $\boldsymbol{e}(k)$, respectively, and $\boldsymbol{G}_D(z)$ denotes the transfer function from the quantization error to output roundoff noise. Therefore, a *noise gain* due to quantization error can be defined as

$$I(\boldsymbol{D}) = \text{tr}(\boldsymbol{W}_D) \tag{5a}$$

where

$$\boldsymbol{W}_D = \frac{1}{2\pi j} \oint_{|z|=1} \boldsymbol{G}_D(z)\boldsymbol{G}_D^*(z)\frac{dz}{z} \tag{5b}$$

It is known that the matrix $\boldsymbol{W}_D$ in (5b) can be expressed as [8]

$$\boldsymbol{W}_D = (\boldsymbol{A} - \boldsymbol{D})^T\boldsymbol{W}_o(\boldsymbol{A} - \boldsymbol{D}) + \boldsymbol{c}^T\boldsymbol{c} \tag{6}$$

where $\boldsymbol{W}_o$ is the observability Gramian of the filter and can be computed by solving the Lyapunov equation [11]

$$\boldsymbol{W}_o - \boldsymbol{A}^T\boldsymbol{W}_o\boldsymbol{A} = \boldsymbol{c}^T\boldsymbol{c} \tag{7}$$

3. Joint Optimization of Error-Feedback and Realization

3.1. An Optimization Formulation

As is well-known [11], the state-space realizations that are equivalent to a particular realization of a given digital filter, say $(\boldsymbol{A},\ \boldsymbol{b},\ \boldsymbol{c},\ d)_n$, are characterized by $(\overline{\boldsymbol{A}},\ \overline{\boldsymbol{b}},\ \overline{\boldsymbol{c}},\ \overline{d})_n = (\boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T},\ \boldsymbol{T}^{-1}\ \boldsymbol{b},\ \boldsymbol{c}\boldsymbol{T},\ d)_n$ where $\boldsymbol{T} \in \mathcal{R}^{n\times n}$ is a non-singular coordinate transformation matrix. Under a transformation $\boldsymbol{T}$, the noise gain defined in (5) becomes

$$I(\boldsymbol{D}) = \text{tr}(\overline{\boldsymbol{W}}_D) \tag{8a}$$

where

$$\overline{\boldsymbol{W}}_D = (\overline{\boldsymbol{A}} - \boldsymbol{D})^T\overline{\boldsymbol{W}}_o(\overline{\boldsymbol{A}} - \boldsymbol{D}) + \overline{\boldsymbol{c}}^T\overline{\boldsymbol{c}} \tag{8b}$$

$$\overline{\boldsymbol{W}}_o = \boldsymbol{T}^T\boldsymbol{W}_0\boldsymbol{T} \tag{8c}$$

A basic constraint imposed on RN minimization is the l_2-norm dynamic range of the state variables [1][2]. Under a coordinate transformation, this constraint can be expressed as

$$(\overline{\boldsymbol{K}}_c)_{ii} = 1 \qquad \text{for } 1 \le i \le n \tag{9a}$$

where

$$\overline{\boldsymbol{K}}_c = \boldsymbol{T}^{-1}\boldsymbol{K}_c\boldsymbol{T}^{-T} \tag{9b}$$

and $\boldsymbol{K}_c$ is the controllability Gramian of the original realization and can be computed by solving the Lyapunov equation [11]

$$\boldsymbol{K}_c - \boldsymbol{A}\boldsymbol{K}_c\boldsymbol{A}^T = \boldsymbol{b}\boldsymbol{b}^T \tag{10}$$

Primarily we are concerned with the minimization of the roundoff noise gain $I(\boldsymbol{D})$ in (8) subject to the constraints in (9). For the sake of reducing solution sensitivity (as will be detailed shortly), however, the "magnitude" of the observability Gramian, $\overline{\boldsymbol{W}}_o$, needs to be controlled during the optimization process. To this end the objective function needs to be modified and the constrained minimization problem can be described as

$$\underset{\boldsymbol{D},\ \boldsymbol{T}}{\text{minimize}} \quad J(\boldsymbol{D},\boldsymbol{T}) = \text{tr}[(1-\mu)\overline{\boldsymbol{W}}_D + \mu\overline{\boldsymbol{W}}_o] \tag{11a}$$

$$\text{subject to:} \quad (\boldsymbol{T}^{-1}\boldsymbol{K}_c\boldsymbol{T}^{-T})_{ii} = 1 \quad \text{for } 1 \le i \le n \tag{11b}$$

where $\overline{\boldsymbol{W}}_D$ and $\overline{\boldsymbol{W}}_o$ are given in (8), and $0 \le \mu \le 1$ is a scalar that weighs the importance of reducing tr($\overline{\boldsymbol{W}}_o$) relative to reducing tr($\overline{\boldsymbol{W}}_D$).

The problem formulation in (11) is rather general. As a matter of fact, it includes the following two special cases: (i) if error feedback is not used, then $\boldsymbol{D}$ is set to zero, which in conjunction with (8b) and (7) implies that

$$J(\boldsymbol{0},\boldsymbol{T}) = \text{tr}(\overline{\boldsymbol{W}}_o) \tag{12}$$

Several methods of minimizing $J(\boldsymbol{0},\boldsymbol{T})$ in (12) subject to (11b) were investigated in [1][2]. (ii) to minimize the objective function $J(\boldsymbol{D},\boldsymbol{T})$ for a fixed $\boldsymbol{T}$ (i.e., for a given state-space realization). This problem has been addressed in [8]. Consequently, the solution of the general optimization problem in (11) that finds the jointly optimized error-feedback matrix $\boldsymbol{D}$ and coordinate transformation matrix $\boldsymbol{T}$ is expect to be superior to the solutions obtained from these two special cases.

3.2. An Equivalent Unconstrained Problem

Since the IIR filter at hand is assumed to be stable, controllable and observable, the controllability matrix $\boldsymbol{K}_c$ is positive definite [11]. Let $\boldsymbol{K}_c^{1/2}$ denote the symmetric square root of $\boldsymbol{K}_c$, i.e., $\boldsymbol{K}_c^{1/2}$ is a symmetric matrix satisfying $\boldsymbol{K}_c^{1/2}\boldsymbol{K}_c^{1/2} = \boldsymbol{K}_c$, then $\boldsymbol{K}_c^{1/2}$ is also positive define and we can define

$$\hat{\boldsymbol{T}} = \boldsymbol{T}^T\boldsymbol{K}_c^{-1/2} \tag{13}$$

which implies that $\boldsymbol{T}^{-1} = \hat{\boldsymbol{T}}^{-T}\boldsymbol{K}_c^{-1/2}$ and the constraints in (11b) become

$$(\hat{\boldsymbol{T}}^{-T}\hat{\boldsymbol{T}}^{-1})_{ii} = 1 \qquad \text{for } 1 \le i \le n \tag{14}$$

The constraints in (14) simply mean that each cloumn in $\hat{\boldsymbol{T}}^{-1}$ must be a unity vector. This can be satisfied if $\hat{\boldsymbol{T}}^{-1}$ assumes the form

$$\hat{\boldsymbol{T}}^{-1} = \left[\frac{\boldsymbol{t}_1}{\|\boldsymbol{t}_1\|} \quad \frac{\boldsymbol{t}_2}{\|\boldsymbol{t}_2\|} \quad \cdots \quad \frac{\boldsymbol{t}_n}{\|\boldsymbol{t}_n\|} \right] \tag{15}$$

with $\boldsymbol{t}_i \in \mathcal{R}^{n\times 1}$. To complete our problem conversion, we need to re-write the objective function in (11a) in terms of $\boldsymbol{D}$ and $\hat{\boldsymbol{T}}$, and this can be done as follows.

From (8) and (13), we can write

$$\begin{aligned} &\overline{\boldsymbol{W}}_D = \\ &(\boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T} - \boldsymbol{D})^T \boldsymbol{T}^T \boldsymbol{W}_o \boldsymbol{T} (\boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T} - \boldsymbol{D}) + \boldsymbol{T}^T \boldsymbol{c}^T \boldsymbol{c} \boldsymbol{T} \\ =&\hat{\boldsymbol{T}}[(\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T})^T \hat{\boldsymbol{W}}_o (\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T}) + \hat{\boldsymbol{C}}] \hat{\boldsymbol{T}}^T \end{aligned} \tag{16a}$$

where

$$\hat{\boldsymbol{A}} = \boldsymbol{K}_c^{-1/2} \boldsymbol{A} \boldsymbol{K}_c^{1/2} \tag{16b}$$

$$\hat{\boldsymbol{C}} = \boldsymbol{K}_c^{1/2} \boldsymbol{c}^T \boldsymbol{c} \boldsymbol{K}_c^{1/2} \tag{16c}$$

$$\hat{\boldsymbol{W}}_o = \boldsymbol{K}_c^{1/2} \boldsymbol{W}_o \boldsymbol{K}_c^{1/2} \tag{16d}$$

and the objective function in (11a) becomes

$$\begin{aligned} J(\boldsymbol{D}, \hat{\boldsymbol{T}}) = \text{tr}\{&\hat{\boldsymbol{T}}[(1-\mu)(\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T})^T \hat{\boldsymbol{W}}_o \\ &(\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T}) + (1-\mu)\hat{\boldsymbol{C}} + \mu \hat{\boldsymbol{W}}_o] \hat{\boldsymbol{T}}^T\} \end{aligned} \tag{17}$$

which can also be expressed as

$$J(\boldsymbol{D}, \hat{\boldsymbol{T}}) = (1-\mu)(J_1 + J_2) + \mu J_3 \tag{18a}$$

with

$$J_1 = \text{tr}[\hat{\boldsymbol{T}}(\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T})^T \hat{\boldsymbol{W}}_o (\hat{\boldsymbol{A}} - \hat{\boldsymbol{T}}^T \boldsymbol{D} \hat{\boldsymbol{T}}^{-T}) \hat{\boldsymbol{T}}^T] \tag{18b}$$

$$J_2 = \text{tr}(\hat{\boldsymbol{T}} \hat{\boldsymbol{C}} \hat{\boldsymbol{T}}^T) \tag{18c}$$

$$J_3 = \text{tr}(\hat{\boldsymbol{T}} \hat{\boldsymbol{W}}_o \hat{\boldsymbol{T}}^T) \tag{18d}$$

Because $\hat{\boldsymbol{W}}_o$ is positive definite and $\hat{\boldsymbol{C}}$ is positive semidefinite, we have $J_i \geq 0$ for $i = 1,\ 2,\ 3$. It follows immediately that if the error-feedback matrix $\boldsymbol{D}$ is allowed to be a general matrix, then the optimal choice of $\boldsymbol{D}$ is

$$\boldsymbol{D} = \hat{\boldsymbol{T}}^{-T} \hat{\boldsymbol{A}} \hat{\boldsymbol{T}}^T \tag{19}$$

as it leads to $J_1 = 0$. In other words, in the case of $\boldsymbol{D}$ being a general matrix, the objective function is simplified to

$$J(\hat{\boldsymbol{T}}^{-T} \hat{\boldsymbol{A}} \hat{\boldsymbol{T}}^T, \hat{\boldsymbol{T}}) = \text{tr}\{\hat{\boldsymbol{T}}[(1-\mu)\hat{\boldsymbol{C}} + \mu \hat{\boldsymbol{W}}_o] \hat{\boldsymbol{T}}^T\} \tag{20}$$

Another case that entails a simplified $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ is when $\boldsymbol{D}$ is a scalar matrix, i.e., $\boldsymbol{D} = \alpha \boldsymbol{I}$, which leads $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ to

$$\begin{aligned} J(\alpha \boldsymbol{I}, \hat{\boldsymbol{T}}) = \text{tr}\{&\hat{\boldsymbol{T}}[(1-\mu)(\hat{\boldsymbol{A}} - \alpha \boldsymbol{I})^T \hat{\boldsymbol{W}}_o (\hat{\boldsymbol{A}} - \alpha \boldsymbol{I}) \\ &+ (1-\mu)\hat{\boldsymbol{C}} + \mu \hat{\boldsymbol{W}}_o] \hat{\boldsymbol{T}}^T\} \end{aligned} \tag{21}$$

In summary, the joint optimization problem in (11) is now re-formulated as

$$\underset{\boldsymbol{D},\, \hat{\boldsymbol{T}}}{\text{minimize}}\ J(\boldsymbol{D}, \hat{\boldsymbol{T}}) \tag{22}$$

where $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ assumes the form in (20) if $\boldsymbol{D}$ is a general matrix, the form in (21) if $\boldsymbol{D}$ is a scalar matrix, and otherwise the form in (17); and $\hat{\boldsymbol{T}}$ assumes the form in (15). From above discussion, it is quite clear that the variables in problem (22) consist of the vectors $\boldsymbol{t}_1,\ \boldsymbol{t}_2,\ \cdots,\ \boldsymbol{t}_n$ (see (15)) plus certain entries of $\boldsymbol{D}$: for example it would include α if $\boldsymbol{D} = \alpha \boldsymbol{I}$, the n diagonal elements of $\boldsymbol{D}$ if $\boldsymbol{D}$ is a diagonal matrix, but no entries of $\boldsymbol{D}$ need to be included if $\boldsymbol{D}$ is a general matrix. It should be emphasized that although the vectors $\{\boldsymbol{t}_i,\ 1 \leq i \leq n\}$ have to be such that $\hat{\boldsymbol{T}}$ is nonsingular, this type of "constraint" needs not to be imposed explicitly because a near singular $\hat{\boldsymbol{T}}$ would make the value of $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ very large, hence the process of minimizing $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ automatically avoids considering ill-conditioned $\hat{\boldsymbol{T}}$. Consequently, the problem in (22) is practically an *unconstrained* minimization problem.

Finally, a remark on the term μJ_3 in (18): from (8) it is clear that if $\boldsymbol{D} \neq \overline{\boldsymbol{A}}$ then the contribution from term $(\overline{\boldsymbol{A}} - \boldsymbol{D})^T \overline{\boldsymbol{W}}_o (\overline{\boldsymbol{A}} - \boldsymbol{D})$ to the noise gain is critically depending on the "magnitude" of $\overline{\boldsymbol{W}}_o$. Since in a practical implementation $\boldsymbol{D}$ is of finite precision, $\overline{\boldsymbol{A}} - \boldsymbol{D}$ is always a nonzero matrix and therefore the magnitude of $\overline{\boldsymbol{W}}_o$ (in terms of its norm, for example) should be controlled. In particular, when $\boldsymbol{D}$ is allowed to be a general matrix, the objective function then becomes the form in (20) and μJ_3 is the only term there to control the value of J_3 (which turns out to be the Frobenius norm of $\overline{\boldsymbol{W}}_o^{1/2}$). This necessitates the use of a nonzero μ for the objective function in (20). In other cases such as those in (17) and (21), the term equivalent to $(\overline{\boldsymbol{A}} - \boldsymbol{D})^T \overline{\boldsymbol{W}}_o (\overline{\boldsymbol{A}} - \boldsymbol{D})$ always presents, and the use of $\mu = 0$ would not in general lead to ill-conditioned results.

3.3. A Quasi-Newton Algorithm for Problem (22)

Let $\boldsymbol{x}$ be the column vector that collects the variables in $\boldsymbol{D}$ and $\hat{\boldsymbol{T}}$, thus $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ is a function of $\boldsymbol{x}$, denoted by $J(\boldsymbol{x})$. The algorithm starts with a trivial initial point $\boldsymbol{x}_0$ obtained by letting $\boldsymbol{D} = \boldsymbol{I}$ and $\hat{\boldsymbol{T}} = \boldsymbol{I}$. Now suppose we are in the kth iteration to update the most recent point $\boldsymbol{x}_k$. A quasi-Newton algorithm updates $\boldsymbol{x}_k$ to $\boldsymbol{x}_{k+1}$ as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \lambda_k \boldsymbol{d}_k \tag{23}$$

in two steps: (i) Determine a search direction $\boldsymbol{d}_k = -\boldsymbol{S}_k \boldsymbol{g}_k$ where $\boldsymbol{g}_k = \nabla J(\boldsymbol{x})$ is the gradient of the objective function and $\boldsymbol{S}_k$ is a positive-definite approximation of the inverse Hessian matrix of $J(\boldsymbol{x})$. A popular quasi-Newton algorithm is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [10] which updates $\boldsymbol{S}_k$ through the recursive relation

$$\boldsymbol{S}_{k+1} = \boldsymbol{S}_k + \left(1 + \frac{\boldsymbol{\gamma}_k^T \boldsymbol{S}_k \boldsymbol{\gamma}_k}{\boldsymbol{\gamma}_k^T \boldsymbol{\delta}_k}\right) \frac{\boldsymbol{\delta}_k \boldsymbol{\delta}_k^T}{\boldsymbol{\gamma}_k^T \boldsymbol{\delta}_k} - \frac{(\boldsymbol{\delta}_k \boldsymbol{\gamma}_k^T \boldsymbol{S}_k + \boldsymbol{S}_k \boldsymbol{\gamma}_k \boldsymbol{\delta}_k^T)}{\boldsymbol{\gamma}_k^T \boldsymbol{\delta}_k} \tag{24}$$

where $\boldsymbol{S}_0 = \boldsymbol{I}$, $\boldsymbol{\delta}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$, and $\boldsymbol{\gamma}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k$, (ii) Once the search direction $\boldsymbol{d}_k$ is computed, the one-dimensional optimization (often called line search)

$$\lambda_k = \arg \underset{\lambda}{\text{minimize}}\, J(\boldsymbol{x}_k + \lambda \boldsymbol{d}_k) \tag{25}$$

is carried out to determine the value of λ_k. If the iteration progress measured by $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|$, is greater than a prescribed tolerance ε, then set $k := k + 1$ and repeat from Step (i), otherwise the iteration is terminated and $\boldsymbol{x}_{k+1}$ is claimed to be a solution point.

The implementation of (24) requires the gradient of $J(\boldsymbol{x})$. Closed form expressions for $J(\boldsymbol{x})$ with scalar, diagonal, and general error-feedback matrix $\boldsymbol{D}$ are given in Appendix A.

3.4. Examples

We present two examples to illustrate the proposed optimization method. The first example concerns a 3rd-order lowpass IIR digital filter which was also used in [8]. The second example is about a 9th-order lowpass IIR filter which is used to demonstrate the ability of the proposed algorithm to deal with relatively large number of variables.

Example 1 Consider a 3rd-order stable IIR lowpass digital filter whose controllable canonical realization is denoted by $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}, d)_3$ with

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.339377 & -1.152652 & 1.52016 \end{bmatrix} \quad (26a)$$

$$\boldsymbol{b} = [0 \quad 0 \quad 0.4437881]^T \quad (26b)$$

$$\boldsymbol{c} = [0.212964 \quad 0.293733 \quad 0.718718] \quad (26c)$$

$$d = 6.59592 \times 10^{-2} \quad (26d)$$

The controllability Gramian $\boldsymbol{K}_c$ of the above filter has been normalized to satisfy the constraints $(\boldsymbol{K}_c)_{ii} = 1$ for $i = 1, 2, 3$. Without error feedback (i.e., $\boldsymbol{D} = \boldsymbol{0}$), the noise gain of filter (26) was found to be [8]

$$\text{tr}(\boldsymbol{W}_o) = 11.1332$$

If one applies the method of [1][2] to filter (26) to obtain a realization $(\overline{\boldsymbol{A}}, \overline{\boldsymbol{b}}, \overline{\boldsymbol{c}}, d)_3$ for roundoff noise minimization (without error feedback), then the noise gain was reduced to [8]

$$\text{tr}(\overline{\boldsymbol{W}}_o) = 2.3554$$

Next, we compute jointly optimal error-feedback and state-space realization, with $\boldsymbol{D}$ being scalar, diagonal, and general matrices, by solving the respective minimization problem (22) using BFGS updates. The total number of variables involved in the optimization are 10 (for a scalar $\boldsymbol{D}$), 12 (for a diagonal $\boldsymbol{D}$), and 9 (for a general $\boldsymbol{D}$). For the cases where $\boldsymbol{D}$ is either a scalar or diagonal matrix, $\mu = 0$ was used. For the case of $\boldsymbol{D}$ being a general matrix, $\mu = 0.01$ was assumed. In all three cases the initial point used corresponds to $\boldsymbol{D} = \boldsymbol{I}$ and $\hat{\boldsymbol{T}} = \boldsymbol{I}$, and with $\varepsilon = 10^{-8}$ the algorithm converges with less than 20 iterations. The minimized noise gains obtained are given in Table 1. For comparison purposes, Table 1 also includes the noise gain values obtained in [8] where the error-feedback matrix is optimized for a fixed state-space realization that is optimal (without error feedback) for roundoff noise minimization. These values are listed in Table 1 in the lines where "Separate" is indicated for the column "Joint/Separate". Form the simulation results, it is evident that the proposed joint optimization offers improved performance for RN reduction for all three types of $\boldsymbol{D}$.

Table 1. Performance Comparison for Example 1

Realization	Error Feedback	Joint/Separate Optimization	Noise Gain
Canonical	0	—	11.1332
Optimal	0	—	2.3554
Optimal	Scalar	Separate	1.5350
Optimal	Scalar	Joint	1.4500
Optimal	Diagonal	Separate	1.4338
Optimal	Diagonal	Joint	1.3090
Optimal	General	Separate	0.7798
Optimal	General	Joint	0.4208

Example 2 Now we consider a 9th-order stable IIR lowpass filter whose transfer function is denoted by

$$H(z) = \frac{b_1 z^9 + b_2 z^8 + \cdots + b_9 z + b_{10}}{a_1 z^9 + a_2 z^8 + \cdots + a_9 z + a_{10}}$$

where the coefficients are given in Table 2.

Table 2. Coefficients of $H(z)$

i	a_i	b_i
1	1	0.002198
2	−3.640015	−0.007993
3	7.148374	0.010478
4	−9.521133	−0.007251
5	9.297299	0.011297
6	−6.830931	−0.010582
7	3.754299	−0.017728
8	−1.485109	0.023996
9	0.384233	0.006712
10	−0.049851	0.046116

Next we obtain the controllable canonical realization of the filter and normalize its controllability matrix by scaling so as to satisfy the constraints $(\boldsymbol{K}_c)_{ii} = 1$ for $i = 1, 2, \ldots, 9$. The state-space realization obtained is denoted by $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}, d)_9$ and its noise gain (without error feedback) was found to be

$$\text{tr}(\boldsymbol{W}_o) = 3.1354 \times 10^3$$

Without using error feedback, the method of [1][2] was applied to obtain a realization $(\overline{\boldsymbol{A}}, \overline{\boldsymbol{b}}, \overline{\boldsymbol{c}}, \overline{d})_9$ for RN minimization, whose noise gain was reduced to

$$\text{tr}(\overline{\boldsymbol{W}}_o) = 2.5315$$

The proposed joint optimization method was then applied to the controllable canonical realization with error-feedback matrix $\boldsymbol{D}$ being scalar, diagonal, and general matrices. For the cases of scalar and diagonal $\boldsymbol{D}$, the weighting factor μ was set to zero; while for the case of a general $\boldsymbol{D}$, $\mu = 0.03$ was used. As in Example 1, for all three cases the same initial point, which corresponds to the choice of $\boldsymbol{D} = \boldsymbol{I}$, $\hat{\boldsymbol{T}} = \boldsymbol{I}$, was used. With $\varepsilon = 10^{-4}$, it took the algorithm 35 (for a scalar $\boldsymbol{D}$), 196 (for a diagonal $\boldsymbol{D}$), and 54 (for a general $\boldsymbol{D}$) iterations to converge. The minimized noise gains are given in Table 3. Again, for comparison purposes, Table 3 also lists the noise gains obtained using separate realization and error-feedback matrix optimization proposed in [8]. From the table, it is observed that the performance improvement provided by using the joint optimization appears to be more pronounced. Based on this and a large number of simulations conducted so far, we conclude that the proposed joint optimization can offer improved performance gain for IIR state-space digital filters of relatively high order.

Table 3. Performance Comparison for Example 2

Realization	Error Feedback	Joint/Separate Optimization	Noise Gain
Canonical	0	—	3.1354×10^3
Optimal	0	—	2.5315
Optimal	Scalar	Separate	1.3622
Optimal	Scalar	Joint	1.1628
Optimal	Diagonal	Separate	1.2995
Optimal	Diagonal	Joint	0.9866
Optimal	General	Separate	0.2776
Optimal	General	Joint	0.0868

Appendix A Evaluation of $\nabla J(\boldsymbol{x})$

Depending on the type of matrix $\boldsymbol{D}$, the objective function $J(\boldsymbol{x})$ may assume one of the three expressions in (17), (20), and (21). In what follows, the derivation of $\nabla J(\boldsymbol{x})$ is carried out for two separate cases.

A.1 If D is a general or a scalar matrix

The objective function in both (20) and (21) has the form $J(\boldsymbol{x}) = \text{tr}(\hat{\boldsymbol{T}}\boldsymbol{M}\hat{\boldsymbol{T}}^T)$ which, in the light of (15), can be expressed as

$$J(\boldsymbol{x}) = \text{tr}\left\{ \left[\frac{\boldsymbol{t}_1}{\|\boldsymbol{t}_1\|} \;\; \cdots \;\; \frac{\boldsymbol{t}_n}{\|\boldsymbol{t}_n\|} \right]^{-1} \boldsymbol{M} \left[\frac{\boldsymbol{t}_1}{\|\boldsymbol{t}_1\|} \;\; \cdots \;\; \frac{\boldsymbol{t}_n}{\|\boldsymbol{t}_n\|} \right]^{-T} \right\} \tag{A1}$$

- In the case of $\boldsymbol{D}$ being a general matrix, $\boldsymbol{M}$ is a constant matrix (see (20)) and $\boldsymbol{x}$ contains a total of n^2 variables, i.e., $\boldsymbol{t}_1,\ \boldsymbol{t}_2,\ \ldots,\ \boldsymbol{t}_n$. To compute $\partial J(\boldsymbol{x})/\partial t_{ij}$, we perturb the ith component of vector $\boldsymbol{t}_j$ by a small amount, say δ, and keep the rest of $\hat{\boldsymbol{T}}$ unchanged. If we denote the perturbed jth column of $\hat{\boldsymbol{T}}^{-1}$ by $\tilde{\boldsymbol{t}}_j/\|\tilde{\boldsymbol{t}}_j\|$, then we can write a linear approximation of $\tilde{\boldsymbol{t}}_j/\|\tilde{\boldsymbol{t}}_j\|$ as

$$\frac{\tilde{\boldsymbol{t}}_j}{\|\tilde{\boldsymbol{t}}_j\|} \approx \frac{\boldsymbol{t}_j}{\|\boldsymbol{t}_j\|} - \delta \boldsymbol{g}_{ij}$$

where $\boldsymbol{g}_{ij}$ is a vector given by

$$\boldsymbol{g}_{ij} = \frac{1}{\|\boldsymbol{t}_j\|^3}(t_{ij}\boldsymbol{t}_j - \|\boldsymbol{t}_j\|^2 \boldsymbol{e}_i) \tag{A2}$$

and $\boldsymbol{e}_i$ is the ith coordinate vector. Now let $\hat{\boldsymbol{T}}_{ij}$ be the matrix obtained from $\hat{\boldsymbol{T}}$ with a perturbed (i,j)th component, then up to the first order the matrix inversion formula [11, p. 655] gives

$$\hat{\boldsymbol{T}}_{ij} = \hat{\boldsymbol{T}} + \frac{\delta(\hat{\boldsymbol{T}}\boldsymbol{g}_{ij})(\boldsymbol{e}_j^T\hat{\boldsymbol{T}})}{1 - \delta \boldsymbol{e}_j^T\hat{\boldsymbol{T}}\boldsymbol{g}_{ij}}$$

Consequently, we have

$$\begin{aligned} \frac{\partial J(\boldsymbol{x})}{\partial t_{ij}} &= \lim_{\delta \to 0}[\text{tr}(\hat{\boldsymbol{T}}_{ij}\boldsymbol{M}\hat{\boldsymbol{T}}_{ij}^T) - \text{tr}(\hat{\boldsymbol{T}}\boldsymbol{M}\hat{\boldsymbol{T}}^T)]/\delta \\ &= 2\text{tr}[(\hat{\boldsymbol{T}}\boldsymbol{g}_{ij})(\boldsymbol{e}_j^T\hat{\boldsymbol{T}})\boldsymbol{M}\hat{\boldsymbol{T}}^T] \\ &= 2\boldsymbol{e}_j^T(\hat{\boldsymbol{T}}\boldsymbol{M}\hat{\boldsymbol{T}}^T\hat{\boldsymbol{T}})\boldsymbol{g}_{ij} \;\; \text{for } 1 \le i,j \le n \end{aligned} \tag{A3}$$

- If $\boldsymbol{D}$ is a scalar matrix, $\boldsymbol{D} = \alpha\boldsymbol{I}$, then vector $\boldsymbol{x}$ contains a total of $n^2 + 1$ variables, i.e., $\alpha,\ \boldsymbol{t}_1,\ \boldsymbol{t}_2,\ \ldots,\ \boldsymbol{t}_n$. In this case $\partial J(\boldsymbol{x})/\partial t_{ij}$ can also be computed using (A3), and it follows from (21) that

$$\frac{\partial J(\boldsymbol{x})}{\partial \alpha} = (1-\mu)\text{tr}[\hat{\boldsymbol{T}}(2\alpha\hat{\boldsymbol{W}}_o - \hat{\boldsymbol{A}}^T\hat{\boldsymbol{W}}_o - \hat{\boldsymbol{W}}_o\hat{\boldsymbol{A}})\hat{\boldsymbol{T}}^T] \tag{A4}$$

A.2 If matrix D contains certain number of zero components in fixed places

The type of $\boldsymbol{D}$ matrices we deal with here obviously includes the case of $\boldsymbol{D}$ being a diagonal matrix. To evaluate the gradient of $J(\boldsymbol{D}, \hat{\boldsymbol{T}})$ in (17), we write it as

$$\begin{aligned} J(\boldsymbol{x}) = \; & \text{tr}(\hat{\boldsymbol{T}}\boldsymbol{M}\hat{\boldsymbol{T}}^T) + (1-\mu)[\text{tr}(\boldsymbol{D}^T\hat{\boldsymbol{T}}\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{D}) \\ & -2\text{tr}(\hat{\boldsymbol{T}}\hat{\boldsymbol{A}}^T\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{D})] \end{aligned} \tag{A5}$$

where

$$\boldsymbol{M} = (1-\mu)(\hat{\boldsymbol{A}}^T\hat{\boldsymbol{W}}_o\hat{\boldsymbol{A}} + \hat{\boldsymbol{C}}) + \mu\hat{\boldsymbol{W}}_o$$

and $\boldsymbol{x}$ in this case contains the nonzero entries of $\boldsymbol{D}$ plus vectors $\boldsymbol{t}_1,\ \boldsymbol{t}_2,\ \ldots,\ \boldsymbol{t}_n$. To compute $\partial J(\boldsymbol{x})/\partial t_{ij}$, we treat all the quantities other than t_{ij} in (A5) including $\boldsymbol{D}$ as constant terms. It then follows from Sec. A.1 that

$$\frac{\partial J(\boldsymbol{x})}{\partial t_{ij}} = 2\beta_1 + 2(1-\mu)(\beta_2 - \beta_3) \qquad \text{for } 1 \le i,j \le n \tag{A6}$$

with

$$\beta_1 = \boldsymbol{e}_j^T(\hat{\boldsymbol{T}}\boldsymbol{M}\hat{\boldsymbol{T}}^T\hat{\boldsymbol{T}})\boldsymbol{g}_{ij} \tag{A7}$$

$$\beta_2 = \boldsymbol{e}_j^T(\hat{\boldsymbol{T}}\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{D}\boldsymbol{D}^T\hat{\boldsymbol{T}})\boldsymbol{g}_{ij} \tag{A8}$$

$$\beta_3 = \boldsymbol{e}_j^T\hat{\boldsymbol{T}}(\hat{\boldsymbol{A}}^T\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{D} + \hat{\boldsymbol{W}}_o\hat{\boldsymbol{A}}\hat{\boldsymbol{T}}^T\boldsymbol{D}^T)\hat{\boldsymbol{T}}\boldsymbol{g}_{ij} \tag{A9}$$

Finally, using (A5) we compute for $\boldsymbol{D} = \{d_{ij}\}$ the derivative

$$\frac{\partial J(\boldsymbol{x})}{\partial d_{ij}} = 2(1-\mu)\boldsymbol{e}_j^T(\boldsymbol{D}^T\hat{\boldsymbol{T}} - \hat{\boldsymbol{T}}\hat{\boldsymbol{A}}^T)\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{g}_{ij} \tag{A10}$$

In particular, if $\boldsymbol{D}$ is a diagonal matrix, i.e., $\boldsymbol{D} = \text{diag}\{d_1, d_2, \ldots, d_n\}$, then

$$\frac{\partial J(\boldsymbol{x})}{\partial d_i} = 2(1-\mu)\boldsymbol{e}_i^T(\boldsymbol{D}^T\hat{\boldsymbol{T}} - \hat{\boldsymbol{T}}\hat{\boldsymbol{A}}^T)\hat{\boldsymbol{W}}_o\hat{\boldsymbol{T}}^T\boldsymbol{g}_{ii} \tag{A11}$$

References

[1] S. Y. Hwang, "Roundoff noise in state-space digital filtering: A general analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, pp. 254-262, June 1976.

[2] C. T. Mullis and R. A. Roberts, "Synthesis of minimum roundoff noise fixed-point digital filters," *IEEE Trans. Circuits syst.*, vol. 23, pp. 551-562, Sept. 1976.

[3] S. Y. Hwang, "Minimum uncorrelated unit noise in state-space digital filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 25, pp. 273-281, Aug. 1977.

[4] L. B. Jackson, A. G. Lindgren and Y. Kim, "Optimal synthesis of second-order state-space structures for digital filters," *IEEE Trans. Circuits Syst.*, vol. 26, pp. 149-153, Mar. 1979.

[5] B. W. Bomar, "State-space structure for the realization of low roundoff noise digital filters," *Dissertation*, University of Tennessee, 1983.

[6] D. Williamson, "Roundoff noise minimization and pole-zero sensitivity in fixed-point digital filter using residue feedback," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, pp. 1210-1220, Oct. 1986.

[7] G. Li and M. Gevers, "Roundoff noise minimization using delta-operator realizations," *IEEE Trans. Signal Processing*, vol. 41, pp. 629-637, Feb. 1993.

[8] T. Hinamoto, H. Ohnishi, and W.-S. Lu, "Roundoff noise minimization of state-space digital using separate and joint error feedback/coordinate transformation optimization," *IEEE Trans. Circuits Syst., I*, 2003. (to appear)

[9] T. Hinamoto, S. Karino, N. Kuroda and T. Kuma, "Error spectrum shaping in two-dimensional recursive digital filters," *IEEE Trans. Circuits Syst.*, vol. 46, pp. 1203-1215, Oct. 1999.

[10] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., Wiley, New York, 1987.

[11] T. Kailath, *Linear Systems*, Prentice Hall, 1980.