MINIMUM-NORM REALIZATION OF 2-D RECURSIVE FILTERS: A QUASICONVEX PROGRAMMING APPROACH

W.-S. Lu

Department of Electrical and Computer Engineering
University of Victoria
Victoria, BC, Canada V8W 3P6

ABSTRACT

The minimum-norm realization (MNR) of a 2-D recursive digital filter is considered and it is shown that an MNR problem can be formulated as a quasiconvex optimization problem in which the largest generalized eigenvalue of certain matrix pencil is minimized. It is demonstrated that efficient interior-point convex programming techniques such as Nesterov and Nemirovskii's projective method can be used to perform the optimization with considerably reduced computational complexity compared to the existing minimization techniques.

1. INTRODUCTION

Minimum-norm realizations (MNR) of digital filters have been known to possess several desirable properties [1][2]. These include the freedom of overflow limit cycles, low parameter sensitivity, and low roundoff output noise power. For 1-D recursive digital filters, the (spectral) norm of the system matrix of a filter's MNR is equal to the largest magnitude of the filter's poles and obtaining a MNR of a stable recursive filter is a rather simple linear algebraic problem [1]. In the 2-D case, all feasible state-space models, which have been employed for finite-wordlength-effect (FWE) analysis to date, are local models. And the effort to establish an explicit relation of the minimum norm of the system matrix under local state-variable transformations to the system's "poles" is considerably complicated by the fact that the "poles" of a recursive 2-D filter are no longer isolated points but composed of continuous and often unbounded algebraic curves [2][3].

In [4], the problem of obtaining a MNR of a given 2-D recursive filter was formulated as an unconstrained optimization problem, and was tackled using quasi-Newton algorithms [5][6]. A problem with the approach in [4] is its exceedingly high computational complexity even for 2-D filters of moderate sizes. In this paper, we describe an alternative optimization-based method to obtain MNRs for 2-D recursive filters. The problem at hand is formulated as a *quasiconvex programming* problem, which has recently been intensively studied in the optimization and control-system communities in the context of convex and semidefinite programming [7]-[18], As will be seen in Section 3, the proposed algorithm requires only a small fraction of the computations that would be needed by the method of [4].

2. PROBLEM FORMULATION

2.1. Formulate an Eigenvalue Minimization Problem as a Semidefinite Programming Problem

Semidefinite programming (SDP) is a relatively new optimization methodology, which is primarily concerned with the problem

$$\text{minimize} \quad c^T x \tag{1a}$$

$$\text{subject to:} \quad F(x) \succeq 0 \tag{1b}$$

$$F(x) = F_0 + \sum_{i=1}^{n} x_i F_i \tag{1c}$$

where $F_i \in R^{n\times n}$ $(i = 0, \ldots, n)$ are given constant matrices with F_i symmetric, and $F(x) \succeq 0$ denotes that $F(x)$ is positive semidefinite at x. Note that the constraint matrix $F(x)$ in (1) is *affine* with respect to x. SDP includes both linear and quadratic programming as special cases, and it represents a broad and important class of convex programming problems. More important, many interior-point methods have recently been generalized to SDP [7]–[13].

Now let us consider the problem of minimizing the largest eigenvalue of a symmetric matrix $S(x)$, which depends on parameter vector x affinely. By writing

$$S(x) = V\Sigma V^T$$

where V is orthogonal and $\Sigma = \text{diag}\{\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n\}$ with $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, we have

$$\lambda I - S(x) = V\text{diag}\{\lambda - \lambda_1,\ \lambda - \lambda_2,\ \ldots,\ \lambda - \lambda_n\}V^T$$

0-7803-5471-0/99/$10.00©1999 IEEE

Evidently, $\lambda I - S(x)$ remain positive definite as long as $\lambda > \lambda_1$ = the largest eigenvalue of $S(x)$. Therefore, minimizing the largest eigenvalue of $S(x)$ can be formulated as

$$\text{minimize} \quad \hat{c}^T\hat{x} \tag{2a}$$
$$\text{subject to:} \quad F(\hat{x}) \succeq 0 \tag{2b}$$

where $\hat{x} = [\lambda \;\; x^T]$, $\hat{c} = [1 \;\; 0 \;\; \dots \;\; 0]^T$, and $F(\hat{x}) = \lambda I - S(x)$. Note that $F(\hat{x})$ is affine w.r.t. $\hat{x}$ as $S(x)$ is affine w.r.t. x. Obviously, the problem in (2) is a SDP problem.

2.2. Formulate the 2-D MNR Problem as a Generalized Eigenvalue Minimization Problem

For the 2-D case, let $\Sigma = \{A, b, c, d\}$ be a Roesser's state-space characterization of a given 2-D filter with $A \in \mathcal{R}^{(n_1+n_2)\times(n_1+n_2)}$, $b \in \mathcal{R}^{(n_1+n_2)\times 1}$, $c \in \mathcal{R}^{1\times(n_1+n_2)}$ and $d \in \mathcal{R}$, then $\hat{\Sigma} = \{\hat{A}, \hat{b}, \hat{c}, \hat{d}\} = \{T^{-1}AT, T^{-1}b, cT, d\}$ with nonsingular $T = T_1 \oplus T_2$, $T_1 \in \mathcal{R}^{n_1\times n_1}$, $T_2 \in \mathcal{R}^{n_2\times n_2}$ describes the same 2-D filter. A state-space model $\tilde{\Sigma} = \{\tilde{A}, \tilde{b}, \tilde{c}, \tilde{d}\} = \{\tilde{T}^{-1}A\tilde{T}, \tilde{T}^{-1}b, c\tilde{T}, d\}$ is said to be a MNR of the filter if

$$\underset{T=T_1\oplus T_2}{\text{minimize}} \; ||T^{-1}AT|| = ||\tilde{A}||$$

where $||\cdot||$ denotes the spectral norm.

By definition, $||T^{-1}AT||^2$ is equal to the largest eigenvalue of $T^{-1}ATT^TA^TT^{-T}$. Since the nonzero eigenvalues of YZ and ZY are identical, $||T^{-1}AT||^2$ is the largest eigenvalue of $P^{-1}APA^T$ with $P = TT^T = T_1T_1^T \oplus T_2T_2^T$. If λ is an eigenvalue of $P^{-1}APA^T$, then there exists column vector $x \neq 0$ such that

$$P^{-1}APA^Tx = \lambda x$$

i.e.,

$$APA^Tx = \lambda Px$$

Thus λ is also a generalized eigenvalue of matrix pencil (APA^T, P) [14]. Therefore, one concludes that $||T^{-1}AT||^2$ is the largest generalized eigenvalue of (APA^T, P) with $P = TT^T$, and that the problem of finding a MNR of the 2-D filter amounts to solving the minimization problem

$$\underset{P=TT^T}{\text{minimize}} \; \lambda_{max}(P, APA^T) \tag{3}$$

where $\lambda_{max}(Y, Z)$ denotes the largest generalized eigenvalue of the pencil $\lambda Y - Z$. Note that matrix P in (3) is positive definite, and the two matrices involved there are *affine* functions of P. Thus the problem in (3) can be reformulated as

$$\text{minimize} \quad \lambda_{max}(P, APA^T) \tag{4a}$$
$$\text{subject to} \quad P = P_1 \oplus P_2,\; P_1 > 0,\; P_2 > 0 \tag{4b}$$

Obviously, the constraints in (4b) are convex with respect to parameter matrices P_1 and P_2. In addition, for $P = P_1 \oplus P_2 > 0$, $\hat{P} = \hat{P}_1 \oplus \hat{P}_2 > 0$, and $0 \le \theta \le 1$, it can be shown that [10]

$$\begin{aligned} &\lambda_{max}(\theta P + (1-\theta)\hat{P},\; A[\theta P + (1-\theta)\hat{P}]A^T) \\ &\le \max(\lambda_{max}(P,\; APA^T),\; \lambda_{max}(\hat{P},\; A\hat{P}A^T)) \end{aligned} \tag{5}$$

which means that the objective function in (4a) is *quasiconvex* and the minimization problem (4) is a quasiconvex programming problem [9][10].

3. ALGORITHM IMPLEMENTATION AND SIMULATION

Presently, there is only a limited number of software packages available that can be used to solve the quasiconvex minimization problem as formulated in (4). One of them is the LMI control Toolbox from MathWorks Inc. [15]. The toolbox works with MATLAB and is aimed primarily at solving design problems arising from control engineering by using linear matrix inequality (LMI) techniques [10]. The LMI toolbox includes a command `gevp` which implements the projective method proposed in [9] as applied to the generalized eigenvalue minimization problem.

We now present two numerical examples. The first example is a stable recursive 2-D state-space filter of order (2, 2), characterized by

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix},\; b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},\; c = [c_1 \;\; c_2]$$

where

$$A_1 = \begin{bmatrix} 1.8890 & -0.9122 \\ 1 & 0 \end{bmatrix} \; A_2 = \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0.0277 & -0.0258 \\ -0.0258 & 0.0243 \end{bmatrix},\; A_4 = \begin{bmatrix} 1.8890 & 1 \\ -0.9122 & 0 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0.2191 \\ 0 \end{bmatrix},\; b_2 = \begin{bmatrix} -0.0289 \\ 0.0912 \end{bmatrix}$$

and

$$c_1 = [0.2889 \;\; -0.0912],\; c_2 = [-0.2191 \;\; 0].$$

By applying Nesterov-Nemirouski's projective method to problem (4), we obtain a MNR of the above system as

$$\tilde{A} = \begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 \\ \tilde{A}_3 & \tilde{A}_4 \end{bmatrix},\; \tilde{b} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix},\; \tilde{c} = [\tilde{c}_1 \;\; \tilde{c}_2]$$

where

$$\tilde{A}_1 = \begin{bmatrix} 0.9412 & -0.1601 \\ 0.1257 & 0.9478 \end{bmatrix},\; \tilde{A}_2 = \begin{bmatrix} -0.0995 & 0.0813 \\ 0.0813 & -0.0664 \end{bmatrix}$$

$$\tilde{A}_3 = \begin{bmatrix} 0.1344 & -0.0366 \\ -0.0366 & 0.1263 \end{bmatrix}, \tilde{A}_4 = \begin{bmatrix} 0.9413 & 0.1258 \\ -0.1600 & 0.9477 \end{bmatrix}$$

$$\tilde{b}_1 = \begin{bmatrix} 0.1157 \\ -0.0946 \end{bmatrix}, \tilde{b}_2 = \begin{bmatrix} 1.1369 \\ 1.5793 \end{bmatrix}$$

and

$$\tilde{c}_1 = [1.5329 \;\; 1.2068], \; \tilde{c}_2 = [-0.0413 \;\; 0.0337].$$

The norm of the system matrix is reduced from $||A|| = 2.7584$, to $||\tilde{A}|| = 0.9867$. The number of floating-point operations (flops) used is 2353, with 0.66 seconds of CPU time on a Pentium Pro 200. For the comparison purposes, the BFGS-based minimization algorithm proposed in [4] was applied to the same problem, which leads to a MNR with the same system matrix norm as $||\tilde{A}||$. The algorithm of [4] however requires 2.05×10^6 flops and 3.68 seconds of CPU time on the same computer. The unweighted mixed L_1/L_2 sensitivity, and L_2 sensitivity, which are defined and studied in [16], and the output roundoff-noise power of the original system are 2.3179×10^5, 1.9831×10^7, and 3.2243×10^3 as compared to 2.6391×10^3, 3.3118×10^3, and 129.4412 for the MNR system obtained above.

The second example is a stable 2-D system of order (3, 3), characterized by [16]

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}, \; b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \text{ and } c = [c_1 \;\; c_2]$$

where

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.38315 & -1.38605 & 1.90670 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} -0.06280 & 0.06190 & 0.00654 \\ -0.02810 & 0.03956 & -0.02248 \\ 1.24452 & -0.57092 & 2.05865 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} -0.00003 & 0.00038 & -0.00053 \\ -0.00001 & 0.00018 & -0.00026 \\ -0.00008 & 0.00023 & -0.00017 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.38238 & -1.38178 & 1.90253 \end{bmatrix}$$

$$\begin{aligned} b_1 &= b_2 = [0 \;\; 0 \;\; 1]^T \\ c_1 &= [0.01141 \;\; -0.00540 \;\; 0.01956] \\ c_2 &= [0.01164 \;\; -0.00545 \;\; 0.01960] \end{aligned}$$

It took the projective algorithm 4364 flops and 1.27 seconds of CPU time to generate an MNR of the system as

$$\tilde{A} = \begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 \\ \tilde{A}_3 & \tilde{A}_4 \end{bmatrix}, \; \tilde{b} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix} \text{ and } \tilde{c} = [\tilde{c}_1 \;\; \tilde{c}_2]$$

where

$$\tilde{A}_1 = \begin{bmatrix} 0.6683 & 0.3025 & 0.0251 \\ -0.2674 & 0.5781 & 0.3265 \\ 0.0850 & -0.3000 & 0.6603 \end{bmatrix}$$

$$\tilde{A}_2 = \begin{bmatrix} 0.0088 & 0.0075 & 0.0065 \\ -0.0323 & -0.0269 & -0.0230 \\ 0.0344 & 0.0286 & 0.0243 \end{bmatrix}$$

$$\tilde{A}_3 = \begin{bmatrix} 0.0072 & 0.0101 & -0.0190 \\ 0.0077 & -0.0311 & -0.0086 \\ -0.0065 & 0.0316 & 0.0069 \end{bmatrix}$$

$$\tilde{A}_4 = \begin{bmatrix} 0.6419 & 0.2831 & 0.1457 \\ -0.3424 & 0.5865 & 0.2601 \\ -0.3337 & -0.2898 & 0.6742 \end{bmatrix}$$

$$\begin{aligned} \tilde{b}_1 &= [0.0051 \;\; -0.0181 \;\; 0.0193]^T \\ \tilde{b}_2 &= [1.5886 \;\; -4.4301 \;\; 4.6081]^T \\ \tilde{c}_1 &= [4.5415 \;\; 3.5599 \;\; 3.1759] \\ \tilde{c}_2 &= [0.0167 \;\; 0.0139 \;\; 0.0119]. \end{aligned}$$

The system norm was reduced from $||A|| = 3.9560$ to $||\tilde{A}|| = 0.7490$. The BGFS based minimization method also works but requires 16.75×10^6 flops and 13.51 seconds of CPU time. As well, the unweighted mixed L_1/L_2 sensitivity, L_2 sensitivity and the output roundoff-noise power are reduced from 1.7344×10^4, 2.8990×10^5, and 72.8485 for the original system to 8.7248×10^3, 8.7450×10^3, and 10.8377 for the MNR system.

4. CONCLUSION

It has been demonstrated that the recently developed interior-point convex programming algorithms have a positive impact on finding minimum-norm realization of 2-D recursive digital filters.

Acknowledgement

The author is grateful to the Natural Sciences and Engineering Research Council of Canada for supporting this work.

5. REFERENCES

[1] C. W. Barnes and A. T. Fam, "Minimum norm recursive digital filters that are free of overflow limit cycles," *IEEE Trans. Circuits Syst.*, vol. 24, pp. 569-574, Oct. 1977.

[2] W.-S. Lu and A. Antoniou, *Two-Dimensional Digital Filters*, New York: Marcel Dekker, 1992.

[3] N. K. Bose, *Applied Multidimensional Systems Theory*, New York: Van Nostrand Reinhold, 1982.

[4] Q.-H. Meng, A. Hasan, W.-S. Lu, and V. K. Bhargava, "On the minimization of system matrix for two-dimensional digital filters," in *Proc. IEEE Pacific Rim Conf.*, pp. 396-399, Victoria, Canada, May 1991.

[5] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed., Reading, MA.: Addison Wesley, 1984.

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

[7] F. Alizadeh, *Combinatorial Optimization with Interior Point Methods and Semidefinite Matrices*, Ph.D. thesis, Computer Science Department, University of Minnesota, Minneapolis, MN, 1991.

[8] F. Alizadeh, "Interior point methods in semidefinite programming with applications to combinatorial optimization," *SIAM J. Optimization*, vol. 5, pp. 13-51, 1995.

[9] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, Philadelphia: SIAM, 1994.

[10] S. Boyd, L. El-Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Philadelphia: SIAM, 1994.

[11] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, "Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability, and numerical results," *SIAM J. Optimization*, vol. 8, pp. 746-768, 1998.

[12] A. Nemirovskii and P. Gahinet, "The projective method for solving linear matrix inequalities," *Proc. Amer. Contr. Conf.*, pp. 840-844, Baltimore, MD, 1994.

[13] L. Vandenberghe and S. Boyd, "Semidefinite Programming," *SIAM Review*, vol. 38, pp. 49-95, March 1996.

[14] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins University Press, 1989.

[15] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, *Manual of LMI Control Toolbox*, The MathWorks Inc., May 1995.

[16] T. Hinamoto, Y. Zempo, Y.Nishino, and W.-S. Lu, "An analytical approach for the synthesis of 2-D state-space filter structures with minimum weighted sensitivity," to appear in *IEEE Trans. CAS I*, 1999.