Semidefinite Programming: A Versatile Tool for Analysis and Design of Digital Filters

W.-S. Lu
Dept. of Electrical and Computer Engineering
University of Victoria, P.O. Box 3055
Victoria, BC, Canada V8W 3P6

Abstract

Semidefinite programming (SDP) is a relatively new methodology for constrained optimization of a linear matrix-variable function subject to linear equality and inequality constraints as well as linear positive-semidefinite constraints. The primary purpose of this paper is to demonstrate that many digital-filter analysis and design problems can be formulated as SDP problems and, therefore, they can be solved effectively using powerful SDP solvers.

1 Introduction

Semidefinite programming (SDP) is a relatively new methodology for constrained optimization of linear matrix-variable functions subject to linear equality and inequality constraints as well as linear positive-semidefinite constraints. SDP includes the important linear programming (LP) and convex quadratic programming (QP) problems as its special cases. More importantly, many interior-point optimization algorithms that have proven efficient for LP and QP problems have recently been extended to SDP [1]-[4].

The primary purpose of this paper is to demonstrate that many digital-filter analysis and design problems can be formulated as SDP problems and, therefore, they can be solved effectively using powerful SDP solvers [2][5]. The analysis and design problems to be discussed in this paper include minimum-norm realization of two-dimensional (2-D) digital filter, minimax design of one-dimensional (1-D) and 2-D digital filters, and equiripple-passbands and least-squares-stopbands design of 1-D digital filters.

There are several ways a SDP problem can be formulated, and the one which turns out to be convenient for filter design purposes is given as

\[
\text{minimize} \quad c^T x \\
\text{subject to} \quad F(x) \succeq 0
\]

(1a)

(1b)

\[ F(x) = F_0 + \sum_{i=1}^{n} x_i F_i \quad (1c) \]

In (1), \( x \in R^{n \times 1} \) is the variable, \( c \in R^{n \times 1} \), \( F_i \in R^{n \times n} \) \( (i = 0, 1, \ldots, n) \) are given constant matrices with \( F_i \) symmetric, and \( F(x) \succeq 0 \) denotes that \( F(x) \) is positive semidefinite. Note that the constraint matrix \( F(x) \) in (1) is affine with respect to \( x \). SDP includes both linear and quadratic programming (QP) as special cases, and it represents a broad and important class of convex programming problems. More important, many interior-point methods which have proven efficient for linear programming, have recently been generalized to SDP [2][3].

2 Design of Nonlinear Phase FIR Filters

We consider the problem of designing a nonlinear-phase, FIR digital filter that approximates a desired frequency response (both magnitude and phase responses) in the passbands in the Chebyshev sense, and approximates desired (zero) magnitude response in the stopbands in the least-squares sense. Consideration of such designs has been justified by many, see for example Adams [20]. Design of nonlinear-phase equiripple FIR filters is also considered in the paper. The term “nonlinear-phase design” is referred to a filter design in which the phase response in stopbands and transition bands are not required to be linear. As expected, the phase-response relaxation in the stopbands and transition bands from strict linearity has been found useful in enhancing the performance of the filter designed [21][22].

Consider the transfer function of an \( N \)-tap FIR filter

\[ H(z) = \sum_{k=0}^{N-1} h_k z^{-k} \]

(2)
and denote its frequency response by

$$H(\omega) = \sum_{k=0}^{N-1} h_k e^{-j k \omega} = h^T [c(\omega) - j s(\omega)]$$  \hspace{1cm} (3)$$

where $h = [h_0, \ldots, h_{N-1}]^T$, $c(\omega) = [1 \cos \omega \cdots \cos(N-1)\omega]^T$, and $s(\omega) = [0 \sin \omega \cdots \sin(N-1)\omega]^T$. Here we do not assume any symmetry in $h$. Let $H_d(\omega)$ be the desired frequency response, which is usually complex-valued. In an equiripple design, one seeks to find coefficient vector $h$ that solves the optimization problem

$$\text{minimize} \ h \ \text{maximize} \ W(\omega)|H(\omega) - H_d(\omega)|$$  \hspace{1cm} (4)$$

where $\Omega$ is a compact region on $[-\pi, \pi]$. The minimax problem in (4) can be reformulated as

$$\text{minimize} \ \delta$$  \hspace{1cm} (5a)$$

subject to $W^2(\omega)|H(\omega) - H_d(\omega)|^2 \leq \delta$ for $\omega \in \Omega$  \hspace{1cm} (5b)$$

Now let

$$H_d(\omega) = H_r(\omega) - j H_i(\omega)$$

with $H_r(\omega)$ and $H_i(\omega)$ real, and use (3) to write the left-hand side of the constraint in (5b) as

$$W^2(\omega)|H(\omega) - H_d(\omega)|^2 = \alpha_1^2(\omega) + \alpha_2^2(\omega)$$  \hspace{1cm} (6)$$

where

$$\alpha_1(\omega) = h^T c_r(\omega) - H_r(\omega)$$

$$\alpha_2(\omega) = h^T s_r(\omega) - H_i(\omega)$$

$$c_r(\omega) = W(\omega)c(\omega)$$

$$s_r(\omega) = W(\omega)s(\omega)$$

$$H_r(\omega) = W(\omega)H_r(\omega)$$

$$H_i(\omega) = W(\omega)H_i(\omega)$$

Constraint (5b) then becomes

$$\delta - \alpha_1^2(\omega) - \alpha_2^2(\omega) \geq 0 \ \omega \in \Omega$$  \hspace{1cm} (7)$$

It can be shown that (7) is equivalent to

$$\Delta(\omega) = \begin{bmatrix} \delta & \alpha_1(\omega) & \alpha_2(\omega) \\ \alpha_1(\omega) & 1 & 0 \\ \alpha_2(\omega) & 0 & 1 \end{bmatrix} \succeq 0 \ \omega \in \Omega$$  \hspace{1cm} (8)$$

If we denote $x = [\delta \ h^T]^T$, then the linear dependence of $\alpha_1(\omega)$ and $\alpha_2(\omega)$ on $h$ implies that matrix $\Delta(\omega)$ in (8) is affine w.r.t. $x$. Therefore, if $\{\omega_i, i = 1, \ldots, M\} \subseteq \Omega$ is a set of grid points that are sufficiently dense in $\Omega$, then a discretized version of (5) can be described as

$$\text{minimize} \ c^T x$$  \hspace{1cm} (9a)$$

subject to $F(x) \succeq 0$  \hspace{1cm} (9b)$$

where $c = [1 \ 0 \ \cdots \ 0]^T$, and $F(x) = \text{diag}\{\Delta(\omega_1), \Delta(\omega_2), \ldots, \Delta(\omega_M)\}$. Obviously, $F(x)$ in (9b) is affine w.r.t. $x$, hence (9) is a SDP problem. Note that $F(x)$ is a tridiagonal matrix of size $3M \times 3M$, which becomes increasingly sparse with $M$.

In an EPPLCSS type design, one seeks to find $h$ which minimizes the weighted $L_2$ error function

$$e(h) = \int_{\Omega} W(\omega)|H(\omega) - H_d(\omega)|^2 d\omega$$  \hspace{1cm} (10a)$$

subject to constraints

$$|H(\omega) - H_d(\omega)|^2 \leq \delta_\rho \ \omega \in \Omega_\rho$$  \hspace{1cm} (10b)$$

$$|H(\omega)|^2 \leq \delta_\alpha \ \omega \in \Omega_\alpha$$  \hspace{1cm} (10c)$$

where $\Omega_\rho$ and $\Omega_\alpha$ denote the unions of passbands and stopbands, respectively. Simple manipulations of the integral in (10a) yields

$$e(h) = h^T Ph - 2h^T q + c_0$$  \hspace{1cm} (11)$$

where

$$P = \int_{\Omega} W(\omega)[c(\omega) \ s(\omega)] [c(\omega) \ s(\omega)]^T d\omega$$

is positive definite for a compact $\Omega$, and

$$q = \int_{\Omega} W(\omega)[H_r(\omega)c(\omega) + H_i(\omega)s(\omega)] d\omega$$

and

$$c_0 = \int_{\Omega} |H_d(\omega)|^2 d\omega$$

Let $P^{1/2}$ be the symmetric square root of $P$, i.e., $P^{T/2} = P^{1/2}$ and $P^{1/2}P^{1/2} = P$. Then (11) can be written as

$$e(h) = \|P^{1/2}h - P^{-1/2}q\|^2 - (\|P^{-1/2}q\|^2 - c_0)$$

Hence

$$e(h) \leq \delta$$

is equivalent to

$$\delta + c_1 - \|P^{1/2}h - P^{-1/2}q\|^2 \geq 0$$  \hspace{1cm} (12)$$
where \( c_0 = \|P^{-1/2}q\|^2 - c_0 \). It can be shown that (12) holds if and only if

\[
\Gamma_0 = \begin{bmatrix}
\delta + c_1 & h^T P^{1/2} - q^T P^{-1/2} \\
\frac{I_N}{P^{1/2} h - P^{-1/2} q}
\end{bmatrix} \succeq 0
\]

(13)

where \( I_N \) is the \( N \times N \) identity matrix. Note that matrix \( \Gamma_0 \) in (13) is affine w.r.t. \( \delta \) and \( h \), and does not depend on \( \omega \). Similar to the way we treat constraint (5b), it can be shown that constraint in (10b) is equivalent to

\[
\Gamma(\omega) = \begin{bmatrix}
\beta_1(\omega) & \beta_2(\omega) \\
\beta_2(\omega) & 0 & 1
\end{bmatrix} \succeq 0 \quad \omega \in \Omega_p
\]

(14)

where \( \beta_1(\omega) = h^T c(\omega) - H_p(\omega) \) and \( \beta_2 = h^T s(\omega) - H_s(\omega) \), and that constraint in (10c) is equivalent to

\[
\Phi(\omega) = \begin{bmatrix}
\gamma_1(\omega) & \gamma_2(\omega) \\
\gamma_2(\omega) & 0 & 1
\end{bmatrix} \succeq 0 \quad \omega \in \Omega_a
\]

(15)

where \( \gamma_1(\omega) = h^T c(\omega) \) and \( \gamma_2(\omega) = h^T s(\omega) \). Now if \( \{\omega_i^{(p)}, i = 1, \ldots, M_p\} \subseteq \Omega_p \) and \( \{\omega_i^{(a)}, i = 1, \ldots, M_a\} \subseteq \Omega_a \) are the sets of grid points in the passbands and stopbands, respectively, on which the constraints (14) and (15) are imposed, then a discretized version of minimizing (10a) subject to (10b) and (10c) can be formulated as

\[
\text{minimize} \quad c^T x
\]

subject to \( F(x) \succeq 0 \)

(16a)

(16b)

where \( x = [\delta, \delta_p, \delta_a, h^T]^T \), \( c = [1, w_p, w_a, 0 \ldots 0]^T \) with scalar weights \( w_p \) and \( w_a \), and \( F(x) = \text{diag}\{\Gamma_0, \Gamma(\omega^{(p)}) \ldots, \Gamma(\omega^{(a)}), \Phi(\omega^{(a)}) \ldots, \Phi(\omega^{(a)})\} \).

Clearly, \( F(x) \) in (16b) is affine w.r.t. \( x \), therefore (16) is a SDP problem.

Presently, there is only a limited number of software packages available that can be used to solve the SDP problem as formulated in (9). One of them is the LMI Control Toolbox from MathWorks Inc. [5]. The toolbox works with MATLAB and is aimed primarily at solving design problems arising from control engineering by using linear matrix inequality (LMI) techniques [6]. The LMI toolbox includes a command named mincx which implements the projective method proposed in [2][7] for solving SDP problems. For the sake of completeness, the next section offers a brief review of several key elements of the projective method.

Figure 1 shows the amplitude response of a 91-tap equiripple FIR filters designed by the proposed method to approximate a lowpass frequency response with normalized \( \omega_p = 0.2375 \), \( \omega_a = 0.2625 \), and group delay = 40. The LMI Control Toolbox was used to perform the design with 36 iterations and 190 Kflops.

3 Minimum-Norm Realization of 2-D Recursive Filters

Minimum-norm realizations (MNR) of digital filters have been known to possess several desirable properties [8][9]. 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 [8]. 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 [9][10].

In [11], 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 [12][13]. A problem with the approach in [11] is its exceedingly high computational complexity even for 2-D filters of moderate sizes. In what follows, we describe an alternative optimization-based method to obtain MNRs for 2-D recursive filters.
The problem at hand is formulated as a \textit{quasiconvex programming} problem, which has recently been intensively studied in the context of convex and semidefinite programming \cite{2,6}.

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 \( \Sigma \in \mathbb{R}^{(n_1+n_2) \times (n_1+n_2)} \), \( b \in \mathbb{R}^{(n_1+n_2) \times 1} \), \( c \in \mathbb{R}^{n_1 \times (n_1+n_2)} \), and \( d \in \mathbb{R} \), then \( \tilde{\Sigma} = \{\tilde{A}, \tilde{b}, \tilde{c}, \tilde{d}\} = \{T^{-1}AT, T^{-1}b, cT, d\} \) with nonsingular \( T = T_1 \oplus T_2 \), \( T_1 \in \mathbb{R}^{n_1 \times n_1} \), \( T_2 \in \mathbb{R}^{n_2 \times n_2} \) describes the same 2-D filter. A state-space model \( \tilde{\Sigma} = \{\tilde{A}, \tilde{b}, \tilde{c}, \tilde{d}\} = \{T^{-1}AT, T^{-1}b, cT, d\} \) is said to be a MNR of the filter if

\[
\min T_{\in T_1 \oplus T_2} ||T^{-1}AT|| = ||\tilde{A}||
\]

where \( ||\cdot|| \) denotes the spectral norm.

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

\[
P^{-1}AP^Tx = \lambda x
\]

i.e.,

\[
AP^Tx = \lambda Px
\]

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

\[
\min \lambda_{\text{max}}(P, AP^T)
\]

where \( \lambda_{\text{max}}(Y, Z) \) denotes the largest generalized eigenvalue of the pencil \( \lambda Y - Z \). Note that matrix \( P \) in (17) is positive definite, and the two matrices involved there are affine functions of \( P \). Thus the problem in (17) can be reformulated as

\[
\min \lambda_{\text{max}}(P, AP^T) \quad \text{subject to} \quad P = P_1 \oplus P_2, \quad P_1 > 0, \quad P_2 > 0 \quad \text{(18a)}
\]

Obviously, the constraints in (18b) are convex with respect to parameter matrices \( P_1 \) and \( P_2 \). In addition, for \( P = P_1 \oplus P_2 > 0 \), \( P = P_1 \oplus P_2 > 0 \), and \( 0 \leq \theta \leq 1 \), it can be shown that \( \lambda \)

\[
\lambda_{\text{max}}(\theta P + (1 - \theta) \tilde{P}, A(\theta P + (1 - \theta) \tilde{P})\tilde{P}) 
\]

\[
\leq \max(\lambda_{\text{max}}(P, AP^T), \lambda_{\text{max}}(\tilde{P}, \tilde{A}P^T))
\]

which means that the objective function in (18a) is \textit{quasiconvex} and the minimization problem (18) is a quasiconvex programming problem \cite{2,6}.

We now present a numerical example. Consider a stable 2-D system of order \( (3, 3) \), characterized by \cite{16}

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

where

\[
A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

\[
0.38315 & -1.38605 & 1.90670 \\
0.38238 & -1.38178 & 1.90253
\]

\[
A_2 = \begin{bmatrix} -0.06280 & 0.06190 & 0.00654 \\ -0.02810 & 0.03956 & -0.02248 \end{bmatrix}
\]

\[
1.24452 & -0.57092 & 2.05865 \\
0.00003 & 0.00038 & -0.00053
\]

\[
A_3 = \begin{bmatrix} -0.00001 & 0.00018 & -0.00026 \\ -0.00008 & 0.00023 & -0.00017 \end{bmatrix}
\]

\[
0.01141 & -0.00540 & 0.01956 \\
0.01164 & -0.00545 & 0.01960
\]

\[
A_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

b_1 = b_2 = [0 0 1]^T

c_1 = [0.0141 -0.0054 0.01956]

c_2 = [0.0164 -0.0054 0.01960]

\[b_1 = b_2 = [0 0 1]^T, \quad c_1 = [0.01141 -0.00540 0.01956], \quad c_2 = [0.01164 -0.00545 0.01960].\]

It took the projective algorithm \cite{2} 4364 flops and 1.27 seconds of CPU time on a PentiumPro 200 to generate an MNR of the system as

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

where

\[
\tilde{A}_1 = \begin{bmatrix} 0.6683 & 0.3025 & 0.0251 \\ 0.0850 & -0.3000 & 0.6603 \end{bmatrix}
\]

\[
-0.2674 & 0.5781 & 0.3265 \\
0.0088 & 0.0075 & 0.0065
\]

\[
-0.0323 & -0.0296 & -0.0230 \\
0.0344 & 0.0286 & 0.0243
\]

\[
0.0072 & 0.0101 & -0.0190 \\
0.0077 & -0.0311 & -0.0086
\]

\[
-0.0065 & 0.0316 & 0.0069 \\
-0.3424 & 0.2831 & 0.1457
\]

\[
\tilde{A}_4 = \begin{bmatrix} 0.6419 & 0.2831 & 0.1457 \\ -0.3424 & 0.2831 & 0.1457 \\
-0.3337 & -0.2898 & 0.6742 \end{bmatrix}
\]

\[
\tilde{b}_1 = [0.0051 -0.0181 0.0193]^T
\]
\[ b_2 = [1.5886 \quad -4.4301 \quad 4.6081]^T \]
\[ c_1 = [4.5415 \quad 3.5599 \quad 3.1759]^T \]
\[ \hat{c}_2 = [0.0167 \quad 0.0139 \quad 0.0119]^T \]

The system norm was reduced from \(|A| = 3.9560 \times 10^9\) to \(|\hat{A}| = 0.7490\). The BGFS based minimization method proposed in [11] also works but requires 16.75 \times 10^6 \text{flops} and 13.51 \text{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 \times 10^4, 2.8990 \times 10^3, and 72.8485 for the original system to 8.7248 \times 10^3, 8.7450 \times 10^3, and
10.8377 for the MNR system.

4 A Minimax Design of Nonrecursive 2-D Digital Filters

It is well-known that minimax designs of 2-D FIR filters can be accomplished with various methods as
shown, for example, in [16]-[19]. However, computer simulations have demonstrated that SDP can yield the required design with significantly improved computational efficiency.

Let the transfer function of a 2-D FIR digital filter be

\[
H(z_1, z_2) = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} h_{ij} z_1^{-i} z_2^{-j} = z_1^T \hat{H} z_2 \tag{20}
\]

where \(N_1\) and \(N_2\) are odd integers, \(z_1 = [1 \quad z_1^{-1} \quad \ldots \quad z_1^{-(N_1-1)}]^T\), \(z_2 = [1 \quad z_2^{-1} \quad \ldots \quad z_2^{-(N_2-1)}]^T\), and \(\hat{H} \in \mathbb{R}^{N_1 \times N_2}\). To derive a compact expression for the transfer function of a linear phase 2-D FIR filter, we partition \(\hat{H}\) as

\[
\hat{H} = \begin{bmatrix}
H_{11} & h_{12} & H_{13} \\
H_{21} & h_{22} & H_{23} \\
H_{31} & h_{32} & H_{33}
\end{bmatrix}
\]

where \(H_{11}, H_{13}, H_{31}, H_{33} \in \mathbb{R}^{n_1 \times n_2}\), \(h_{12}, h_{32} \in \mathbb{R}^{n_1 \times 1}\), \(h_{21}, h_{23} \in \mathbb{R}^{1 \times n_2}\), \(h_{22} \in \mathbb{R}\), \(n_1 = (N_1 - 1)/2\), and \(n_2 = (N_2 - 1)/2\). Now if

\[
H_{13} = \text{flipud} (H_{33}), \quad H_{31} = \text{fiplr} (H_{33}),
\]
\[
H_{11} = \text{flipud} (\text{fliplr} (H_{33})),
\]
\[
h_{12} = \text{flipud} (h_{32}), \quad h_{21} = \text{fiplr} (h_{23})
\]

where flipud and fiplr represent the operations of flipping a matrix upside down and from left to
right, respectively, then the filter characterized by \(H(z_1, z_2)\) has linear phase response with group delay
\((n_1 T_1, n_2 T_2)\), and the frequency response of the filter is given by

\[
H(e^{j\omega_1}, e^{j\omega_2}) = e^{-j(n_1 \omega_1 + n_2 \omega_2)} c_1^T (\omega_1) H c_2(\omega_2)
\]

where \(c_1(\omega) = [1 \quad \cos \omega_1 \quad \ldots \quad \cos n_1 \omega_1]^T\) for \(i = 1, 2\), and

\[
H = \begin{bmatrix}
h_{22} & 2h_{23}^T \\
2h_{32} & 4h_{33}
\end{bmatrix}
\]

Consequently, the minimax design of a linear-phase 2-D FIR filter is the solution of the optimization problem

\[
\min_{\hat{H}} \{ \max_{(\omega_1, \omega_2) \in \Omega} e(\omega_1, \omega_2, H) \} \tag{21}
\]

where

\[
e(\omega_1, \omega_2, H) = W(\omega_1, \omega_2) |c_1^T (\omega_1) H c_2(\omega_2) - D(\omega_1, \omega_2)| \]

\(\Omega\) denotes the frequency region of interest, \(W(\omega_1, \omega_2) \geq 0\) is a weighting function over region \(\Omega\), and \(D(\omega_1, \omega_2)\) is the desired amplitude response.

Evidently, the optimization problem in (21) is equivalent to

\[
\minimize \delta \tag{22a}
\]

subject to \(e(\omega_1, \omega_2, H) \leq \delta\) for \((\omega_1, \omega_2) \in \Omega\) \tag{22b}

Furthermore, the constraint in (22b) can be expressed as

\[
\delta - e(\omega_1, \omega_2, H) \geq 0 \tag{23}
\]

for \((\omega_1, \omega_2) \in \Omega\). The inequality in (23) implies that

\[
\delta \geq 0 \quad \text{for} \quad (\omega_1, \omega_2) \in \Omega \tag{24}
\]

It is known from linear algebra that the inequalities in (23) and (24) hold if and only if

\[
\Delta(\omega_1, \omega_2) \succeq 0 \quad \text{for} \quad (\omega_1, \omega_2) \in \Omega \tag{25a}
\]

where

\[
\Delta(\omega_1, \omega_2) = \begin{bmatrix}
\delta & e(\omega_1, \omega_2, H) \\
e(\omega_1, \omega_2, H) & 1
\end{bmatrix} \tag{25b}
\]

It is important to note that matrix \(\Delta(\omega_1, \omega_2)\) in (25b) is affine w.r.t. design matrix \(H\) and the scalar auxiliary variable \(\delta\). A discretized version of the positive semidefinite condition in (25) is given by

\[
D(x) = \text{diag} \{ \Delta(\omega_{i1}, \omega_{i2}), \ldots, \Delta(\omega_{i1}, \omega_{i2}) \} \succeq 0 \tag{26}
\]

where the set \(\{ (\omega_{i1}, \omega_{i2}) \}, 1 \leq i \leq M\) forms a sufficiently dense grid in region \(\Omega\), and \(x = [x_1, x_2 \ldots]^T\) is vector of dimension \((n_1 + 1)(n_2 + 1)\) where \(x_1 = \delta\).
and $x_1, x_3, \ldots$ are the entries of $H$. Taking the above analysis into account, a discretized version of the optimization problem in (22) can be formulated as

$$\text{minimize} \quad c^T x \quad \text{(27a)}$$

subject to $D(x) \succeq 0 \quad \text{(27b)}$

where $c = [1 \quad 0 \quad \ldots \quad 0]^T$ and $D(x)$ is defined by (26) and (25b). Since matrix $D(x)$ in (27b) is affine w.r.t. $x$, (27) is an SDP problem.

Acknowledgement

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

References


