FIR Filter Design via Spectral Factorization and Convex Optimization

Shao-Po Wu
Stephen Boyd
Lieven Vandenberghe

ABSTRACT We consider the design of finite impulse response (FIR) filters subject to upper and lower bounds on the frequency response magnitude. The associated optimization problems, with the filter coefficients as the variables and the frequency response bounds as constraints, are in general *nonconvex*. Using a change of variables and spectral factorization, we can pose such problems as linear or nonlinear *convex* optimization problems. As a result we can solve them efficiently (and globally) by recently developed interior-point methods. We describe applications to filter and equalizer design, and the related problem of antenna array weight design.

- To appear as Chapter 1 of *Applied Computational Control, Signal and Communications*, Biswa Datta editor, Birkhauser, 1997.
- The paper is also available at URL `http://www-isl.stanford.edu/people/boyd` and from anonymous FTP to `isl.stanford.edu` in `pub/boyd/reports`.

1 Introduction

A *finite impulse response* (FIR) filter is a linear system described by a convolution input-output relation

$$y(t) = \sum_{i=0}^{n-1} h(i)u(t-i), \tag{1.1}$$

where $u : \mathbf{Z} \to \mathbf{R}$ is the input signal and $y : \mathbf{Z} \to \mathbf{R}$ is the output signal. We say n is the filter *order*, and $h = \big(h(0), h(1), \ldots, h(n-1)\big) \in \mathbf{R}^n$ are the filter *coefficients*. The filter *frequency response* $H : \mathbf{R} \to \mathbf{C}$ is defined as

$$H(\omega) = h(0) + h(1)e^{-j\omega} + \cdots + h(n-1)e^{-j(n-1)\omega} \tag{1.2}$$

where $j = \sqrt{-1}$. Since H is 2π periodic and satisfies $H(-\omega) = \overline{H(\omega)}$, it is sufficient to specify it over the interval $\omega \in [0, \pi]$.

A (frequency response) *magnitude specification* has the form

$$L(\omega) \le |H(\omega)| \le U(\omega) \quad \text{for all} \quad \omega \in [0, \pi]. \tag{1.3}$$

We refer to L and U as the (lower and upper) *frequency response bound functions.*

We can assume that $0 \le L(\omega) \le U(\omega)$ for all ω, since $L(\omega)$ can be replaced by $\max\{L(\omega), 0\}$ without changing the constraint, and if $L(\omega) > U(\omega)$ for some ω then the magnitude specification is evidently infeasible. We allow $U(\omega)$ to take on the value $+\infty$ (which means there is no upper bound on $|H(\omega)|$). Similarly, $L(\omega) = 0$ means there is no lower bound on $|H(\omega)|$. For any ω with $L(\omega) = U(\omega)$, the magnitude specification (1.3) implies the (magnitude) equality constraint $|H(\omega)| = L(\omega)$. The magnitude specification (1.3) is sometimes called a *semi-infinite* constraint since it consists of an infinite number of inequalities, *i.e.*, one for each $\omega \in [0, \pi]$.

In this paper we consider the design of FIR filters (*i.e.*, the choice of $h \in \mathbf{R}^n$) that satisfy magnitude specifications. We will refer to such design problems as *magnitude filter design problems.* A classic example is the design of a lowpass filter that maximizes stopband attenuation subject to a given maximum passband ripple. This design problem can be expressed as the optimization problem

$$\begin{array}{ll} \text{minimize} & \delta \\ \text{subject to} & 1/\alpha \le |H(\omega)| \le \alpha, \quad \omega \in [0, \omega_{\rm p}] \\ & |H(\omega)| \le \delta, \quad \omega \in [\omega_{\rm s}, \pi]. \end{array} \tag{1.4}$$

The optimization variables are the filter coefficients $h \in \mathbf{R}^n$ and the stopband attenuation (bound) $\delta \in \mathbf{R}$. The problem parameters are the filter order n, the passband frequency $\omega_{\rm p}$, the stopband frequency $\omega_{\rm s}$, and the maximum allowed passband ripple α. Note that the passband ripple specification is logarithmic or fractional; the upper and lower bounds are given by a constant factor. Passband ripple is commonly expressed in *decibels*, *i.e.*, as $20\log_{10}\alpha$ dB. Similarly, the stopband attenuation is also often expressed in decibels (as $20\log_{10}\delta$ dB).

The magnitude specifications in (1.4), which are given over separate intervals, can be converted to a single magnitude specification of the form (1.3). In this example, there are no specifications over the transition band (ω_p, ω_s), so $L(\omega) = 0$ and $U(\omega) = \infty$ for $\omega_p < \omega < \omega_s$; similarly we have $L(\omega) = 0$ for $\omega \geq \omega_s$.

Magnitude filter design problems (such as the lowpass filter design problem (1.4)) are not, in general, convex optimization problems, except when certain symmetry (sometimes called linear phase) constraints are imposed (*e.g.*, the condition that the filter coefficients are symmetric around the middle index); see, *e.g.*, [40]. Therefore numerical methods that directly solve magnitude filter design problems are not guaranteed to find the *globally* optimal design; such methods can get "stuck" at a locally optimal, but globally suboptimal, design. For some applications this doesn't matter, as long as a good design is found. For cases where finding the global minimum is valuable, there are several heuristic methods for minimizing the likelihood of not finding the global solution, *e.g.*, running the algorithm repeatedly from different starting points. There are also methods for global optimization, *e.g.*, branch-and-bound (see, *e.g.*, [33, 9]) that are guaranteed to find the globally optimal design. These methods, however, are often orders of magnitude less efficient than the standard (local optimization) methods.

The purpose of this paper is to show that by a change of variables, a wide variety of magnitude filter design problems can be posed as *convex* optimization problems, *i.e.*, problems in which the objective and constraint functions are convex. As a consequence we can (globally) solve such magnitude filter design problems with great efficiency using recently developed interior-point methods for convex optimization (see, *e.g.*, [39, 58, 61, 57]).

The new variables are the autocorrelation coefficients of the filter; the filter coefficients are then recovered by spectral factorization (see §2). The idea of designing FIR filters via spectral factorization was first used by Herrmann and Schüssler in the 1970 paper [25]. Since then many authors have studied variations on and extensions of this idea, including different methods for performing the spectral factorization (see, *e.g.*, [12, 38]), and different methods for solving the transformed problem *e.g.*, by exchange-type algorithms or linear or quadratic programming (see, *e.g.*, [12, 29, 18, 51, 21]).

In this paper we extend the idea and apply *nonlinear* convex optimization techniques to a variety of magnitude filter design problems. Linear and quadratic programming are well-developed fields; extremely efficient software is widely available. It is less well known that in the last decade or so the powerful interior-point methods that have been used for linear and quadratic programming have been extended to handle a huge variety of *nonlinear* convex optimization problems (see, *e.g.*, [39, 56, 36]). Making use of convex optimization methods more general than linear or quadratic programming preserves the solution efficiency, and allows us to handle a wider class of problems, *e.g.*, problems with logarithmic (decibel) objectives.

We close this introduction by comparing magnitude filter design, the topic of this paper, with *complex Chebychev design*, in which the specifications have the general form

$$|H(\omega) - D(\omega)| \leq U(\omega) \quad \text{for all} \;\; \omega \in [0, \pi],$$

where $D : [0, \pi] \rightarrow \mathbf{C}$ is a desired or target frequency response, and $U : [0, \pi] \rightarrow \mathbf{R}$ (with $U(\omega) \geq 0$) is a frequency-dependent weighting function. Complex Chebychev design problems, unlike the magnitude design problems addressed in this paper, are generally convex, directly in the variables h; see, *e.g.*, [17, 45].

2 Spectral factorization

The *autocorrelation coefficients* associated with the filter (1.1) are defined as

$$r(t) = \sum_{i=-n+1}^{n-1} h(i)h(i+t), \quad t \in \mathbf{Z}, \tag{1.5}$$

where we interpret $h(t)$ as zero for $t < 0$ or $t > n-1$. Since $r(t) = r(-t)$ and $r(t) = 0$ for $t \geq n$, it suffices to specify the autocorrelation coefficients for $t = 0, \ldots, n-1$. With some abuse of notation, we will write the autocorrelation coefficients as a vector $r = \big(r(0), \ldots, r(n-1)\big) \in \mathbf{R}^n$.

The Fourier transform of the autocorrelation coefficients is

$$R(\omega) = \sum_{t\in\mathbf{Z}} r(t)e^{-j\omega t} = r(0) + \sum_{t=1}^{n-1} 2r(t)\cos\omega t = |H(\omega)|^2, \tag{1.6}$$

i.e., the squared magnitude of the filter frequency response. We will use the autocorrelation coefficients $r \in \mathbf{R}^n$ as the optimization variable in place of the filter coefficients $h \in \mathbf{R}^n$. This change of variables has to be handled carefully, since the transformation from filter coefficients into autocorrelation coefficients is not one-to-one, and not all vectors $r \in \mathbf{R}^n$ are the autocorrelation coefficients of some filter.

From (1.6) we immediately have a necessary condition for r to be the autocorrelation coefficients of some filter: R must satisfy $R(\omega) \geq 0$ for all ω. It turns out that this condition is also sufficient: The *spectral factorization theorem* (see, *e.g.*, [4, CH9]) states that there exists an $h \in \mathbf{R}^n$ such that $r \in \mathbf{R}^n$ is the autocorrelation coefficients of h if and only if

$$R(\omega) \geq 0 \quad \text{for all} \quad \omega \in [0,\pi]. \tag{1.7}$$

The process of determining filter coefficients h whose autocorrelation coefficients are r, given an $r \in \mathbf{R}^n$ that satisfies the spectral factorization condition (1.7), is called *spectral factorization*. Some widely used spectral factorization methods are summarized in the appendix.

Let us now return to magnitude filter design problems. The magnitude specification (1.3) can be expressed in terms of the autocorrelation coefficients r as

$$L(\omega)^2 \leq R(\omega) \leq U(\omega)^2 \quad \text{for all} \quad \omega \in [0,\pi].$$

Of course, we must only consider r that are the autocorrelation coefficients of some h, so we must append the spectral factorization condition to obtain:

$$L(\omega)^2 \leq R(\omega) \leq U(\omega)^2, \quad R(\omega) \geq 0 \quad \text{for all} \quad \omega \in [0,\pi]. \tag{1.8}$$

These conditions are equivalent to the original magnitude specifications in the following sense: there exists an h that satisfies (1.3) if and only if there exists an r that satisfies (1.8). Note that the spectral factorization constraint $R(\omega) \geq 0$ is redundant: it is implied by $L(\omega)^2 \leq R(\omega)$.

For each ω, the constraints in (1.8) are a pair of *linear inequalities* in the vector r; hence the overall constraint (1.8) is *convex* in r. In many cases, this change of variable converts the original, *nonconvex* optimization problem in the variable h into an equivalent, *convex* optimization problem in the variable r.

As a simple example, consider the classical lowpass filter design problem (1.4) described above. We can pose it in terms of r as

$$\begin{array}{ll} \text{minimize} & \tilde{\delta} \\ \text{subject to} & 1/\alpha^2 \leq R(\omega) \leq \alpha^2, \quad \omega \in [0, \omega_{\mathrm{p}}] \\ & R(\omega) \leq \tilde{\delta}, \quad \omega \in [\omega_{\mathrm{s}}, \pi] \\ & R(\omega) \geq 0, \quad \omega \in [0, \pi]. \end{array} \tag{1.9}$$

($\tilde{\delta}$ here corresponds to δ^2 in the original problem (1.4).)

In contrast to the original lowpass filter design problem (1.4), the problem (1.9) is a *convex* optimization problem in the variables $r \in \mathbf{R}^n$ and $\tilde{\delta} \in \mathbf{R}$. In fact it is a semi-infinite linear program, since the objective is linear and the constraints are semi-infinite, *i.e.*, a pair of linear inequalities for each ω.

3 Convex semi-infinite optimization

In this paper we encounter convex optimization problems of the general form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Ax = b, \\ & f_i(x) \leq 0, \quad i = 1, \ldots, m, \\ & g_i(x, \omega) \leq 0, \quad \omega \in [0, \pi], \quad i = 1, \ldots, p, \end{array} \tag{1.10}$$

where $x \in \mathbf{R}^k$ is the optimization variable, $f_0, \ldots, f_m : \mathbf{R}^k \to \mathbf{R}$ are convex functions, and for each $\omega \in [0, \pi]$, $g_i(x, \omega)$ are convex functions of x. Note the three types of constraints: $Ax = b$ are the equality constraints; $f_i(x) \leq 0$ are the (ordinary) inequality constraints; and $g_i(x, \omega) \leq 0$ for all ω are the semi-infinite inequality constraints. If the objective function f_0 is identically zero, the problem (1.10) reduces to verifying whether the constraints are feasible or not, *i.e.*, to a feasibility problem.

In the remainder of this section we give a brief discussion of various methods for handling the semi-infinite constraints. General semi-infinite convex optimization is a well-developed field; see for example [8, 44, 7, 27, 26, 41, 47]. While the theory and methods for *general* semi-infinite convex optimization can be technical and complex, our feeling is that the very special form of the semi-infinite constraints appearing in filter design problems allows them to be solved with no great theoretical or practical difficulty.

Let us start by pointing out that the semi-infinite inequality constraint

$$g_i(x,\omega) \le 0 \quad \text{for all} \ \ \omega \in [0,\pi]$$

can be handled by expressing it as the ordinary inequality constraint

$$h_i(x) = \sup_{\omega \in [0,\pi]} g_i(x,\omega) \le 0.$$

It is easily verified that h_i is a convex function of x, since for each ω, $g_i(x,\omega)$ is convex in x. (On the other hand, h_i is often nondifferentiable, even if the functions g_i are differentiable.) Thus, the semi-infinite constraints in (1.10) can be handled by several methods for general (nondifferentiable) convex optimization, *e.g.*, bundle methods [28], ellipsoid methods [11], or cutting plane methods [30, 23, 46]. What is required in these methods is an efficient method for evaluating h_i and a subgradient at any x. This involves computing a frequency ν for which $g_i(x,\nu) = h_i(x)$. In some problems this can be done analytically; in any case it can be found approximately by a (one-dimensional) search over frequency. (See, *e.g.*, [15, 13].)

It is also possible to solve some magnitude filter design problems exactly, by transforming the semi-infinite constraints into (finite-dimensional) constraints that involve linear matrix inequalities [14, 55, 63], but we will not pursue this idea here.

The semi-infinite constraints can also be *approximated* in a very straightforward way by *sampling* or *discretizing* frequency. We choose a set of frequencies

$$0 \le \omega_1 \le \omega_2 \le \cdots \le \omega_N \le \pi,$$

often uniformly or logarithmically spaced, and replace the semi-infinite inequality constraint

$$g_i(x,\omega) \le 0 \quad \text{for all} \ \ \omega \in [0,\pi]$$

with the set of N ordinary inequality constraints

$$g_i(x,\omega_k) \le 0 \quad k = 1,\ldots,N.$$

Note that sampling preserves convexity. When N is sufficiently large, discretization yields a good approximation of the SIP (see [19]). A standard rule of thumb is to choose $N \approx 15n$ (assuming linear spacing).

As an example consider the lowpass filter design problem given by (1.9). The discretized approximation has the form

$$\begin{array}{ll} \text{minimize} & \tilde{\delta} \\ \text{subject to} & 1/\alpha^2 \le R(\omega_k) \le \alpha^2, \quad \omega_k \in [0, \omega_{\rm p}] \\ & R(\omega_k) \le \tilde{\delta}, \quad \omega_k \in [\omega_{\rm s}, \pi] \\ & R(\omega_k) \ge 0, \quad \omega_k \in [0, \pi]. \end{array} \tag{1.11}$$

This is in fact a linear program (LP) with $n+1$ variables $(r, \tilde{\delta})$, and $2N$ linear inequality constraints. One immediate advantage is that existing, efficient LP software can be used to solve this (discretized) lowpass filter design problem.

The reader should note the differences between the three variations on the lowpass filter design problem. The original problem (1.4) is a semi-infinite, nonconvex problem. The transformed problem (1.9) is a semi-infinite but convex problem, which is equivalent (by change of variables) to the original problem (1.4). The transformed and discretized problem (1.11) is an *approximation* of the semi-infinite problem (1.9). It is also a linear program, and so can be solved using standard linear programming codes (or, for even greater efficiency, by an interior-point LP solver that exploits the special problem structure [16]).

Evidently for the discretized problem we can no longer guarantee that the semi-infinite constraints are satisfied between frequency samples. Indeed the sampled version of the problem is an *outer approximation* of the original problem; its feasible set includes the feasible set of the original problem. In many cases this causes no particular harm, especially if N is large. But if the spectral factorization condition does not hold, *i.e.*, $R(\nu) < 0$ for some ν between samples, then spectral factorization breaks down; we cannot find a set of filter coefficients h that have r as its autocorrelation. Several methods can be used to avoid this pitfall. The simplest is to add a small safety margin to the sampled version of the spectral factorization condition, *i.e.*, replace it by

$$R(\omega_k) \ge \epsilon, \quad k = 1, \dots, N \tag{1.12}$$

where ϵ is small and positive. This can be done in an ad hoc way, by increasing ϵ (and re-solving the problem) until the spectral factorization of R is successful. If N is large this will occur when ϵ is small.

We can also analyze the approximation error induced by discretization by bounding the variation of the functions $g_i(x, \omega)$ for ω between samples. To give a very simple example, assume we use uniform frequency sampling, *i.e.*, $\omega_k = (k - 1/2)\pi/N$, $k = 1, \dots, N$. We assume we have (or impose) some bound on the size of h, say,

$$\|h\| = \sqrt{h(0)^2 + \cdots + h(n-1)^2} \le M.$$

Thus, $r(0) = \|h\|^2 \leq M^2$, and a standard result shows that $|r(t)| \leq M^2$ for all t.

Now let ω be any frequency in $[0, \pi]$ and let ω_k denote the nearest sampling frequency, so that $|\omega - \omega_k| \leq \pi/(2N)$. We have

$$|R(\omega) - R(\omega_k)| = \left| \sum_{t=1}^{n-1} 2r(t)(\cos \omega t - \cos \omega_k t) \right|$$

$$\leq 4M^2 \sum_{t=1}^{n-1} \left| \sin |\omega - \omega_k| t \right| \leq 4M^2 \sum_{t=1}^{n-1} \left| \omega - \omega_k \right| t \leq M^2 n(n-1)\pi/N$$

Thus, the intersample error cannot exceed $M^2 n(n-1)\pi/N$ (which evidently converges to zero as $N \to \infty$). For example, if we take $\epsilon = M^2 n(n-1)\pi/N$ in (1.12), then it is guaranteed that the (semi-infinite) spectral factorization condition (1.7) will be met. The bound developed here is very simple, and only meant to give the general idea; far more sophisticated bounds can be derived; see, *e.g.*, [19, §3.7].

Let us mention several reasons why simple discretization works very well in practice and is widely used. First of all, its computational cost is not prohibitive. Even when special structure (see below) is not exploited, the complexity of solving the discretized problem grows approximately as a linear function of N, so making N large (to get a good discretized approximation) incurs only a linear increase in computation required.

When uniform frequency sampling is used, the fast Fourier transform can be used to great advantage to speed up many of the computations in interior-point methods. In [16] the authors demonstrate that FIR design problems with 1000s of variables and 10000s of constraints can be solved in tens of seconds on a small workstation, by exploiting the special structure of the discretized problems.

Finally, we should point out that frequency sampling plays a key role in the more sophisticated methods for handling semi-infinite constraints. Indeed, most of the more sophisticated algorithms for SIPs solve a sequence of discretized problems with carefully chosen frequency samples (see, *e.g.*, [26, 47].

In the remainder of this paper, we will pose problems first as convex, semi-infinite problems. We then either give the sampled version, or point out if it has a special form such as LP. The sampled versions can be thought of as approximations (which are probably more than adequate for practical designs) or as subproblems that arise in sophisticated algorithms that handle the semi-infinite constraints exactly. The numerical results given in this paper were obtained from (fine) discretizations, using the nonlinear convex programming solver SDPSOL [62].

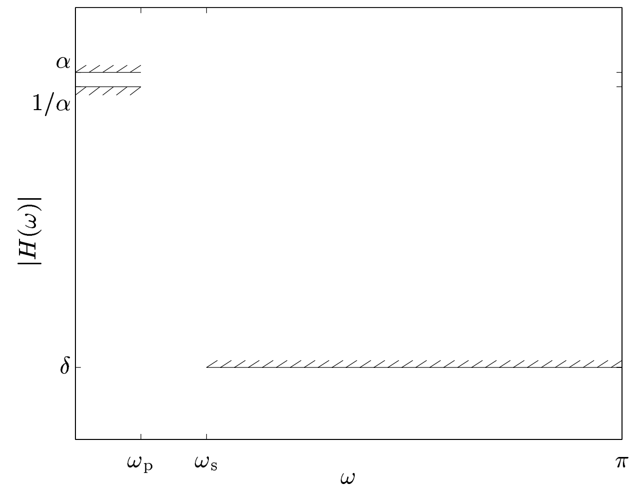

FIGURE 1. Lowpass filter design specifications.

4 Lowpass filter design

In this section we consider the lowpass filter design problem described above as an example. Of course the same techniques are readily applied to other filters such as highpass, bandpass, notch, bandstop, or complex filter types with multiple stop and pass bands.

The constraints consist of a passband ripple specification,

$$1/\alpha \leq |H(\omega)| \leq \alpha \quad \text{for} \;\; \omega \in [0, \omega_{\rm p}],$$

where $\omega_{\rm p}$ is the passband frequency, and $\alpha > 1$ gives the passband ripple. The stopband attenuation specification is given by

$$|H(\omega)| \leq \delta \quad \text{for} \;\; \omega \in [\omega_{\rm s}, \pi],$$

where $\omega_{\rm p} > \omega_{\rm p}$ is the stopband frequency, and δ gives the maximum stopband gain. These specifications are illustrated in figure 1.

We have already seen that the problem of maximizing stopband attenuation (*i.e.*, minimizing δ) can be formulated as the convex optimization problem

$$\begin{array}{ll} \text{minimize} & \tilde{\delta} \\ \text{subject to} & 1/\alpha^2 \leq R(\omega) \leq \alpha^2, \quad \omega \in [0, \omega_{\rm p}] \\ & R(\omega) \leq \tilde{\delta}, \quad \omega \in [\omega_{\rm s}, \pi] \\ & R(\omega) \geq 0, \quad \omega \in [0, \pi], \end{array}$$

which, when discretized in frequency, yields an LP.

An example design is shown in figure 2. The filter order is $n = 30$, the passband ripple is $\alpha = 1.1$ (about 1dB), the passband frequency is $\omega_{\rm p} =$

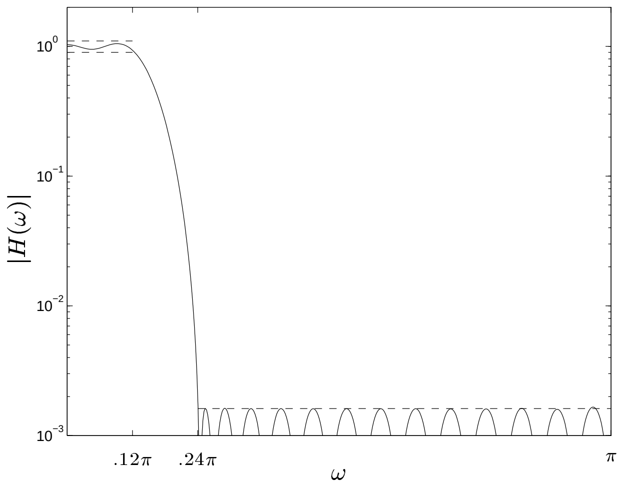

FIGURE 2. Maximum stopband attenuation design. The filter order is $n = 30$, maximum passband ripple is $\alpha = 1.1$ (about 1dB), passband frequency is $\omega_{\mathrm{p}} = 0.12\pi$, and stopband frequency is $\omega_{\mathrm{s}} = 0.24\pi$. The stopband attenuation achieved is -56dB.

0.12π, and the stopband frequency is one octave higher, *i.e.*, $\omega_{\mathrm{s}} = 0.24\pi$. The maximum attenuation achieved is $\delta_{\mathrm{opt}} = 0.0016$, *i.e.*, around -56dB. We remind the reader that this design is the *globally* optimal solution: it is not merely the best design achieved by *our* method, but in fact the best design that can be achieved by *any* method.

We can consider several variations on this problem. Suppose we fix the stopband attenuation and wish to minimize the passband ripple. This can be expressed as the optimization problem

$$\begin{array}{ll} \text{minimize} & \tilde{\alpha} \\ \text{subject to} & 1/\tilde{\alpha} \le R(\omega) \le \tilde{\alpha}, \quad \omega \in [0, \omega_{\mathrm{p}}] \\ & R(\omega) \le \delta^2, \quad \omega \in [\omega_{\mathrm{s}}, \pi] \\ & R(\omega) \ge 0, \quad \omega \in [0, \pi], \end{array}$$

where the optimization variables are r and $\tilde{\alpha}$. The problem parameters are δ, ω_{p}, ω_{s} (and the filter order n). (The optimization variable $\tilde{\alpha}$ corresponds to α^2 in the ripple specification.)

This problem is in fact a *convex* optimization problem. To see this we consider the specifications at a fixed frequency ω. The constraints

$$R(\omega) \le \tilde{\alpha}, \quad R(\omega) \le \delta^2, \quad R(\omega) \ge 0$$

are *linear* inequalities on the variables $(r, \tilde{\alpha})$. The remaining, nonlinear constraint is

$$1/\tilde{\alpha} - R(\omega) \le 0.$$

The function $1/\tilde{\alpha} - R(\omega)$ can be verified to be *convex* in the variables $(r, \tilde{\alpha})$ (since $\tilde{\alpha} > 0$). Indeed, when sampled this problem can be very efficiently solved as a second-order cone program (SOCP); see [36, §3.3] and [35]. Note that passband ripple minimization (in dB) *cannot* be solved (directly) by linear programming; it can, however, be solved by *nonlinear* convex optimization.

We can formulate other variations on the lowpass filter design problem by fixing the passband ripple and stopband attenuation, and optimizing over one of the remaining parameters. For example we can minimize the stopband frequency ω_{s} (with the other parameters fixed). This problem is *quasiconvex*; it is readily and efficiently solved using bisection on ω_{s} and solving the resulting convex feasibility problems (which, when sampled, become LP feasibility problems). Similarly, ω_{p} (which is quasiconcave) can be maximized, or the filter order (which is quasiconvex) can be minimized.

It is also possible to include several types of constraints on the *slope* of the magnitude of the frequency response. We start by considering upper and lower bounds on the absolute slope, *i.e.*,

$$a \leq \frac{d|H(\omega)|}{d\omega} \leq b.$$

This can be expressed as

$$a \leq \frac{dR(\omega)^{1/2}}{d\omega} = \frac{dR/d\omega}{2\sqrt{R(\omega)}} \leq b,$$

which (since $R(\omega)$ is constrained to be positive) we can rewrite as

$$2\sqrt{R(\omega)}a \leq dR/d\omega \leq 2\sqrt{R(\omega)}b.$$

Now we introduce the assumption that $a \leq 0$ and $b \geq 0$. The inequalities can be written

$$2a\sqrt{R(\omega)} - dR/d\omega \leq 0, \quad dR/d\omega - b\sqrt{R(\omega)} \leq 0. \tag{1.13}$$

Since $R(\omega)$ is a linear function of r (and positive), $\sqrt{R(\omega)}$ is a concave function of r. Hence the functions $a\sqrt{R(\omega)}$ and $-b\sqrt{R(\omega)}$ are convex (since $a \leq 0$ and $b \geq 0$). Thus, the inequalities (1.13) are convex (since $dR/d\omega$ is a linear function of r). When discretized, this constraint can be handled via SOCP. As a simple application we can design a lowpass filter subject to the additional constraint that the frequency response magnitude is monotonic decreasing.

Let us now turn to the more classical specification of frequency response slope, which involves the *logarithmic* slope:

$$a \leq \frac{d|H|}{d\omega}\frac{\omega}{|H(\omega)|} \leq b.$$

(The logarithmic slope is commonly given in units of dB/octave or dB/decade.) This can be expressed as

$$a \leq (1/2) \frac{dR}{d\omega} \frac{\omega}{R(\omega)} \leq b,$$

which in turn can be expressed as a pair of linear inequalities on r:

$$2R(\omega)a/\omega \leq \frac{dR}{d\omega} \leq 2R(\omega)b/\omega.$$

Note that we can incorporate arbitrary upper and lower bounds on the logarithmic slope, including equality constraints (when $a = b$). Thus, constraints on logarithmic slope (in dB/octave) are readily handled.

5 Log-Chebychev approximation

Consider the problem of designing an FIR filter so that its frequency response magnitude best approximates a target or desired function, in the sense of minimizing the maximum approximation error in decibels (dB). We can formulate this problem as

$$\text{minimize} \quad \sup_{\omega \in [0,\pi]} \big| \log |H(\omega)| - \log D(\omega) \big| \tag{1.14}$$

where $D : [0, \pi] \to \mathbf{R}$ is the desired frequency response magnitude (with $D(\omega) > 0$ for all ω). We call (1.14) a *logarithmic Chebychev approximation* problem, since it is a minimax (Chebychev) problem on a logarithmic scale.

We can express the log-Chebychev problem (1.14) as

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & 1/\alpha \leq R(\omega)/D(\omega)^2 \leq \alpha, \quad \omega \in [0, \pi], \\ & R(\omega) \geq 0, \omega \in [0, \pi] \end{array}$$

where the variables are $r \in \mathbf{R}^n$ and $\alpha \in \mathbf{R}$. This is a convex optimization problem (as described above), efficiently solved, for example, as an SOCP. Simple variations on this problem include the addition of other constraints, or a frequency-weighted log-Chebychev objective.

As an example we consider the design of a $1/f$ spectrum-shaping filter, which is used to generate $1/f$ noise by filtering white noise through an FIR filter. The goal is to approximate the magnitude $D(\omega) = \omega^{-1/2}$ over a frequency band $[\omega_a, \omega_b]$. If white noise (*i.e.*, a process with spectrum $S_u(\omega) = 1$) is passed through such a filter, the output spectrum S_y will satisfy $S_y(\omega) \approx 1/\omega$ over $[\omega_a, \omega_b]$. This signal can be used to simulate $1/f$ noise (*e.g.*, in a circuit or system simulation) or as a test signal in audio applications (where it is called *pink noise* [54]).

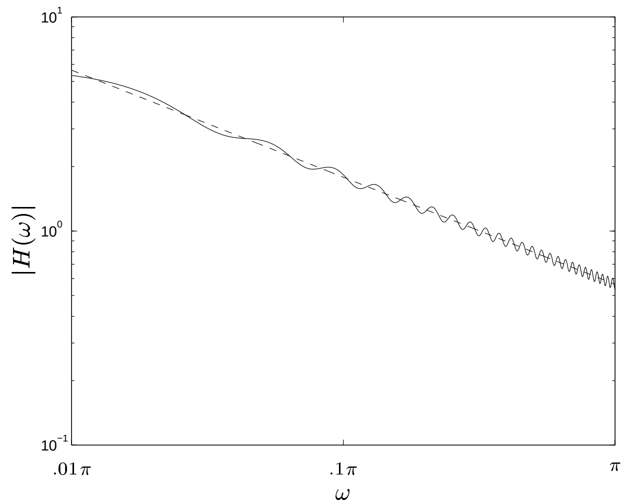

FIGURE 3. Magnitude frequency response of a 50th order $1/f$ spectrum shaping filter for frequency range $[0.01\pi, \pi]$. Dashed line shows ideal $1/\sqrt{f}$ magnitude, which falls at 10dB/octave. The maximum approximation error is ± 0.5dB.

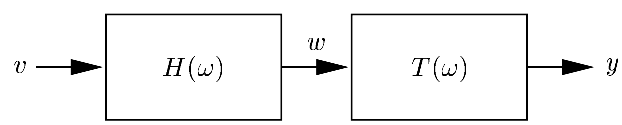

FIGURE 4. Magnitude equalization. The frequency response T of the system is known; the goal is to design the equalizer H so the product TH has approximately constant magnitude.

Using a minimax logarithmic (dB) criterion, and frequency sampling yields the problem

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & 1/\alpha \leq (\omega_i/\alpha)R(\omega_i) \leq \alpha, \quad \omega_i \in [\omega_a, \omega_b] \\ & R(\omega_i) \geq 0, \quad i = 0, \ldots, N, \end{array} \tag{1.15}$$

which is readily solved as an SOCP. Figure 3 shows a design example, with filter order $n = 50$, $\omega_a = 0.01\pi$, $\omega_b = \pi$. The fit obtained is $\alpha_{\mathrm{opt}} = 1.12$, which is around 0.5dB.

6 Magnitude equalizer design

In the simplest magnitude equalizer problem, we are given a function $T : [0, \pi] \to \mathbf{C}$ (not necessarily the frequency response of an FIR filter), and need to design an FIR filter (equalizer) H so that the product TH has approximately constant magnitude, *e.g.*, one:

$$|T(\omega)H(\omega)| \approx 1 \quad \text{for all} \;\; \omega \in [0, \pi].$$

This is illustrated in figure 4, where the equalizer processes the signal before the given function; but the problem is the same if the order of the equalizer and given function is reversed. Note also that we only need to know the magnitude $|T(\omega)|$, and not $T(\omega)$, for $\omega \in [0, \pi]$.

The equalizer problem can be posed as the log-Chebychev approximation problem

$$\text{minimize} \quad \sup_{\omega \in [0,\pi]} \left| \log |T(\omega)H(\omega)| \right|,$$

which is readily formulated as a convex problem using the autocorrelation coefficients r as the design variables:

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & 1/\alpha \le R(\omega)|T(\omega)|^2 \le \alpha, \quad \omega \in [0,\pi] \\ & R(\omega) \ge 0, \quad \omega \in [0,\pi]. \end{array}$$

In many applications we must add *regularization constraints* on the equalizer frequency response, to keep the magnitude or its slope from being too large. These constraints are readily handled. For example, we can impose (frequency-dependent) bounds on $|H|$ and its absolute or logarithmic derivative, as described above. A very simple way to bound the size of h is to impose the constraint

$$r(0) = h(0)^2 + \cdots + h(n-1)^2 \le M^2$$

(which is a single linear inequality on r).

So far we have assumed that the target equalized gain, *i.e.*, the desired value of $|TH|$, is one. We can also allow some freedom in the target value of the equalized gain. This type of problem can be handled using an absolute (or more accurately, squared) criterion

$$\text{minimize} \quad \sup_{\omega \in [0,\pi]} \left| |T(\omega)H(\omega)|^2 - \gamma \right|,$$

where γ is subject to some bounds such as $\gamma_{\rm l} \le \gamma \le \gamma_{\rm h}$, and h is subject to some regularization constraints. Note that the optimization variables here are h and γ.

We can cast this as the convex problem

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & \left| R(\omega)|T(\omega)|^2 - \gamma \right| \le \alpha, \quad \omega \in [0,\pi] \\ & R(\omega) \ge 0, \quad \omega \in [0,\pi] \end{array}$$

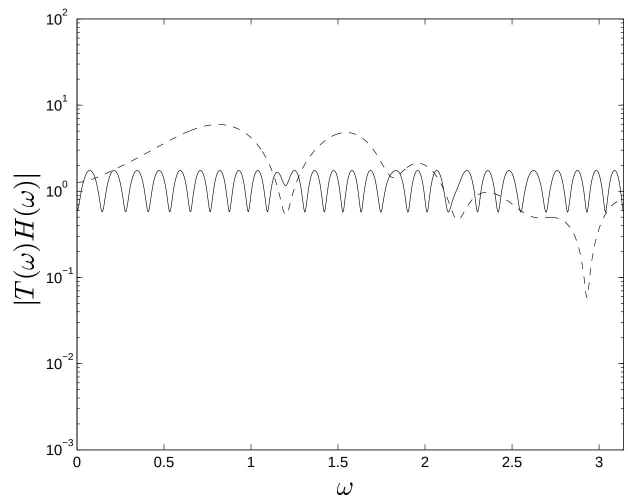

FIGURE 5. Log-Chebychev magnitude equalization. Magnitude of function $T(\omega)$ (dashed) and equalized frequency response $T(\omega)H(\omega)$ (solid). The maximum error is ± 4.8dB.

where in addition we have limits on γ and some constraints on H (*i.e.*, R). If this problem is discretized, it becomes an LP.

An example of log-Chebychev magnitude approximation is illustrated in figure 5. The 50th order equalizer achieves a maximum equalization error of 4.8dB.

We now turn to an interesting and useful extension of the magnitude equalizer problem: simultaneous equalization of several (or many) functions. We are given functions $T_k : [0, \pi] \to \mathbf{C}$, $k = 1, \ldots, K$, and need to design H so that $|T_k H|$ are all approximately constant. In other words, we need to design a single equalizer for multiple functions. This situation arises in several contexts. As an example, suppose that T_k is the frequency response from the (electrical) input to a public address or sound reinforcement system to the (acoustic) response at a location k in the theater. The equalizer H is meant to give approximately constant magnitude response at any of K locations in the theater. The multi-system equalization problem setup is illustrated in figure 6.

We can formulate the multi-system magnitude equalization problem as a minimax log-Chebychev approximation problem:

$$\text{minimize} \quad \max_{k=1,\ldots,K} \sup_{\omega \in [0,\pi]} \left| \log |T_k(\omega) H(\omega)| \right|. \qquad (1.16)$$

In this formulation we have fixed the target value for each $|T_k H|$ as one; it is of course possible to have different target values for different k. We should point out that this minimax formulation already builds in a form of regularization: $H(\omega)$ will become large only if *all* of the $T_k(\omega)$ are small.

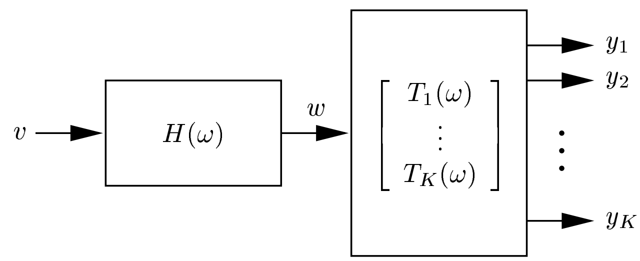

FIGURE 6. Multisystem magnitude equalization. The goal is to choose H so that the magnitude response at $y_1, \ldots, y_K$ are all approximately constant over ω.

The idea of allowing the equalized gain to 'float' is quite useful in the context of multi-system equalization. In the sound reinforcement application mentioned above, the gain at different locations could vary by, say, 10dB, as long as each equalized frequency response is flat within, say, ±4dB. To allow the equalized gains to differ, we can formulate a minimax squared magnitude problem such as

$$\text{minimize} \max_{k=1,\ldots,K} \sup_{\omega\in[0,\pi]} \left| |T_k(\omega)H(\omega)|^2 - \gamma_k \right|,$$

to which we might add constraints on γ_k such as a lower and upper bound. Note that the variables here are h and $\gamma_1, \ldots, \gamma_K$. This problem can be cast as the convex problem

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & \left| |T_k(\omega)|^2 R(\omega) - \gamma_k \right| \le \alpha, \quad k = 1, \ldots, K, \quad \omega \in [0,\pi] \\ & R(\omega) \ge 0, \quad \omega \in [0,\pi]. \end{array}$$

This becomes an LP when discretized. An example with $K = 2$ and filter order $n = 25$ is shown in figure 7.

Our last topic is equalization over frequency bands, which is the most common method used in audio applications. We define a set of K frequency intervals

$$[\Omega_1, \Omega_2], \quad [\Omega_2, \Omega_3], \quad \ldots \quad [\Omega_K, \Omega_{K+1}],$$

where $0 < \Omega_1 < \ldots < \Omega_{K+1} \le \pi$. A common choice of frequencies differ by one-third octave, *i.e.*, $\Omega_k = 2^{(k-1)/3}\Omega_1$ for $k = 1, \ldots, K$. The *average gain* of a function $G : [0,\pi] \to \mathbf{C}$ over the kth band $[\Omega_k, \Omega_{k+1}]$ is defined by

$$\left(\frac{1}{\Omega_{k+1} - \Omega_k} \int_{\Omega_k}^{\Omega_{k+1}} |G(\omega)|^2 \, d\omega \right)^{1/2}.$$

In *frequency band equalization*, we choose the equalizer H so that the average gain of the equalized frequency response TH in each frequency

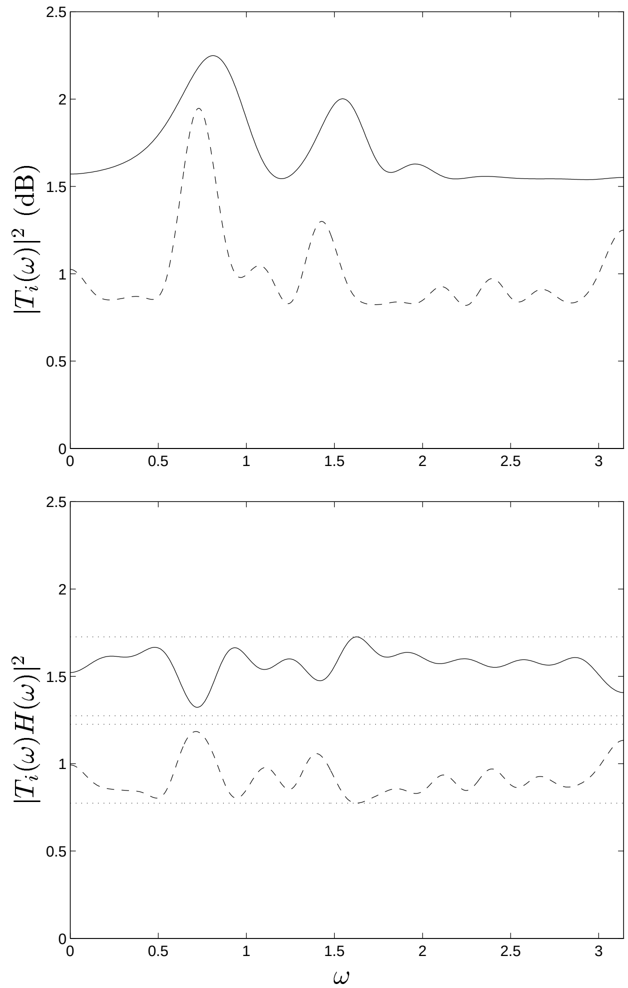

FIGURE 7. Example of multi-system magnitude equalization. The two frequency response squared magnitudes $|T_1|^2$ (solid) and $|T_2|^2$ (dashed) are shown in the upper plot. The equalized frequency response magnitudes $|T_1H|^2$ (solid) and $|T_2H|^2$ (dashed) are shown in the lower plot, along with the ranges $\gamma_{\text{opt}} \pm \alpha_{\text{opt}}$ (dotted).

band is approximately the same. Using a log-Chebychev (minimax dB) criterion for the gains and r as the variable, we can express this equalization problem as

$$\begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & 1/\alpha \leq \frac{1}{\Omega_{k+1}-\Omega_k} \int_{\Omega_k}^{\Omega_{k+1}} R(\omega)|T(\omega)|^2 \, d\omega \leq \alpha, \quad k = 1, \dots, K, \\ & R(\omega) \geq 0, \quad \omega \in [0, \pi]. \end{array}$$

This is a convex problem in r and α. To solve it numerically, we can approximate the integral by frequency sampling. (Indeed, $|T|$ is likely to be given by its values at a fine sampling of frequencies, and not in some analytical form.) We can also, of course, add constraints on H.

An example is shown in figure 8. The equalizer has order $n = 20$, and we consider 15 third-octave bands from $\Omega_1 = 0.031\pi$ to $\Omega_{16} = \pi$. Note that the function $|T|$ has several deep 'dips' and 'notches'. This implies that ordinary Chebychev equalization would require very large values of $|H|$ to achieve good equalization.

7 Linear antenna array weight design

Our last application example treats a problem closely related to FIR filter design: antenna array weight design. Consider a linear array of n isotropic antennas spaced uniformly a distance d apart in a plane, as shown in figure 9. A plane harmonic wave of wavelength λ is incident on the array from angle θ. The antennas sample the incident wave, and the resulting signals are demodulated and then linearly combined with *antenna weights* $w_1, \dots, w_n \in \mathbf{C}$ (which are our design variables) to form the combined output of the antenna array, which is a complex number G. The array output G depends on the incidence angle θ of the incoming wave (and also the weights). As a function of the incidence angle, $G : [0, \pi] \to \mathbf{C}$ is called the *pattern function*, and is given by

$$G(\theta) = \sum_{k=0}^{n-1} w_k e^{-jk\Omega}, \tag{1.17}$$

where Ω depends on the incidence angle as

$$\Omega = -\frac{2\pi d}{\lambda} \cos\theta. \tag{1.18}$$

More details, and a derivation of this formula for G, can be found in any text or survey on antennas or antenna arrays, *e.g.*, [37, 22, 20].

Note that G has the same form as the frequency response of an FIR filter (given in (1.2)), with two differences: the filter coefficients are replaced with

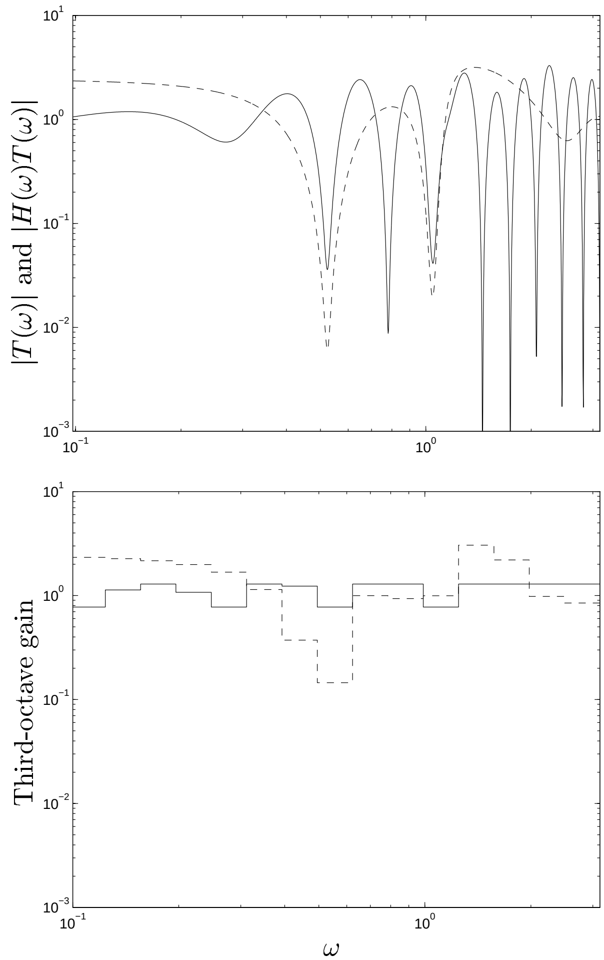

FIGURE 8. Example of third-octave equalization, with constraint that equalizer frequency response magnitude not exceed 10. Upper plot shows $|T|$ (dashed) and the equalized frequency response $|TH|$ (solid). Lower plots shows third-octave gains for T (dashed) and TH (solid). The third-octave gains have been equalized within ± 1.1dB. Since $|T|$ is small at several frequencies, ordinary log-Chebychev equalization to ± 1.1dB would require very large values of $|H|$.

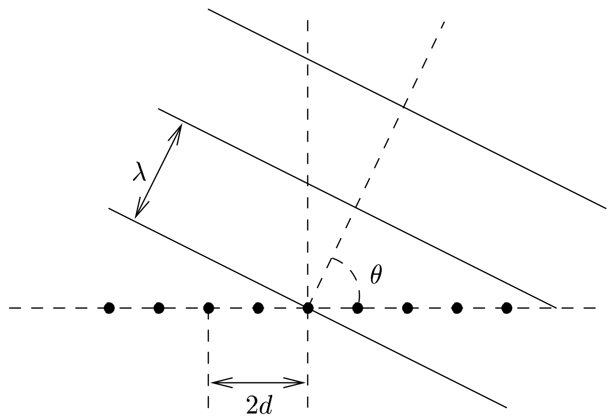

FIGURE 9. Linear antenna array of antennas (shown as dots), with spacing d, in the plane. A plane wave with wavelength λ is incident from angle θ.

the antenna array weights (which can be complex), and the "frequency" variable Ω is related to the incidence angle θ by (1.18).

If we define $H : [-\pi, \pi] \to \mathbf{C}$ as

$$H(\Omega) = \sum_{k=0}^{n-1} w_k e^{-jk\Omega},$$

then we have $G(\theta) = H(\Omega)$. H is then the frequency response of an FIR filter with (complex) coefficients $w_1, \ldots, w_n$. Since H does not satisfy $H(-\Omega) = \overline{H(\Omega)}$ (as the frequency response of an FIR filter with real coefficients does), we must specify H over $\Omega \in [-\pi, \pi]$.

For $\theta \in [0, \pi]$, Ω is monotonically increasing function of θ, which we will denote Ψ, *i.e.*, $\Psi(\theta) = -2\pi d/\lambda \cos\theta$. As the incidence angle θ varies from 0 to π, the variable $\Omega = \Psi(\theta)$ varies over the range $\pm 2\pi d/\lambda$. To simplify the discussion below, we make the (common) assumption that $d < \lambda/2$, *i.e.*, the element spacing is less than one half wavelength. This implies that for $\theta \in [0, \pi]$, $\Psi(\theta) = \Omega$ is restricted to an interval inside $[-\pi, \pi]$. An interval

$$\theta_{\min} \le \theta \le \theta_{\max},$$

where $\theta_{\min}$, $\theta_{\max} \in [0, \pi]$, transforms under (1.18) to the corresponding interval

$$\Omega_{\min} \le \Omega \le \Omega_{\max},$$

where $\Omega_{\min} = \Psi(\theta_{\min})$ and $\Omega_{\max} = \Psi(\theta_{\max})$, which lie in $[-\pi, \pi]$. (The importance of this will soon become clear.)

By analogy with the FIR filter design problem, we can define an antenna pattern *magnitude specification* as

$$L(\theta) \le |G(\theta)| \le U(\theta) \quad \text{for all} \quad \theta \in [0, \pi]. \tag{1.19}$$

An antenna array weight design problem involves such specifications. As a simple example, suppose we want the array to have approximately uniform sensitivity for $\theta \in [0, \theta_{\rm b}]$, and sensitivity as small as possible in the interval $[\theta_{\rm s}, \pi]$. This problem can be posed as

$$\begin{array}{ll} \text{minimize} & \delta \\ \text{subject to} & 1/\alpha \leq |G(\theta)| \leq \alpha, \quad \theta \in [0, \theta_{\rm b}] \\ & |G(\theta)| \leq \delta, \quad \theta \in [\theta_{\rm s}, \pi]. \end{array} \tag{1.20}$$

This problem is the analog of the lowpass filter design problem (1.4). Here, $\theta_{\rm b}$ denotes the (half) *beamwidth*, $\theta_{\rm s}$ denotes the beginning of the *sidelobe*, and δ is called the *sidelobe attenuation level*.

We can recast this problem as

$$\begin{array}{ll} \text{minimize} & \delta \\ \text{subject to} & 1/\alpha \leq |H(\Omega)| \leq \alpha, \quad \Omega \in [\Psi(0), \Psi(\theta_{\rm b})] \\ & |H(\Omega)| \leq \delta, \quad \Omega \in [\Psi(\theta_{\rm s}), \Psi(\pi)]. \end{array} \tag{1.21}$$

(Here we use the interval mapping property mentioned above).

Now (1.21) is a lowpass filter design problem, but with complex coefficients w_i, and specifications over the interval $[-\pi, \pi]$. It can be handled like an FIR filter magnitude design problem, by an extension of spectral factorization to the complex case.

We define the (now complex) autocorrelation coefficients $r(k)$, associated with w, as

$$r(k) = \sum_{i=-n+1}^{n-1} w_i \overline{w_{i+k}}, \quad k = 0, \ldots, n-1. \tag{1.22}$$

The Fourier transform of r is

$$R(\theta) = \sum_{k=-(n-1)}^{n-1} r(k) e^{-j\Omega k} = |G(\theta)|^2,$$

is the squared magnitude of the antenna pattern function (where θ and Ω are related as in (1.17)). We can use $r \in \mathbf{C}^n$ as the design variables, provided we add the spectral factorization condition $R(\theta) \geq 0$ for all $\theta \in [0, \pi]$.

The magnitude constraint can be expressed in terms of R as

$$L(\theta)^2 \leq R(\theta) \leq U(\theta)^2 \quad \text{for all} \;\; \theta \in [0, \pi],$$

i.e., as an (infinite) set of linear inequalities on r.

An example of an antenna array weight design problem is shown in figure 10. The problem is (1.20), with the following parameters: antenna element spacing $d = 0.45\lambda$; $n = 12$ (antenna elements); (half) beamwidth $\theta_{\rm b} = 30°$; allowed ripple $\alpha = 1.58$ (which corresponds to ± 2dB); and sidelobe angle $\theta_{\rm s} = 45°$. The sidelobe attenuation level achieved is around 0.11 (-19dB).

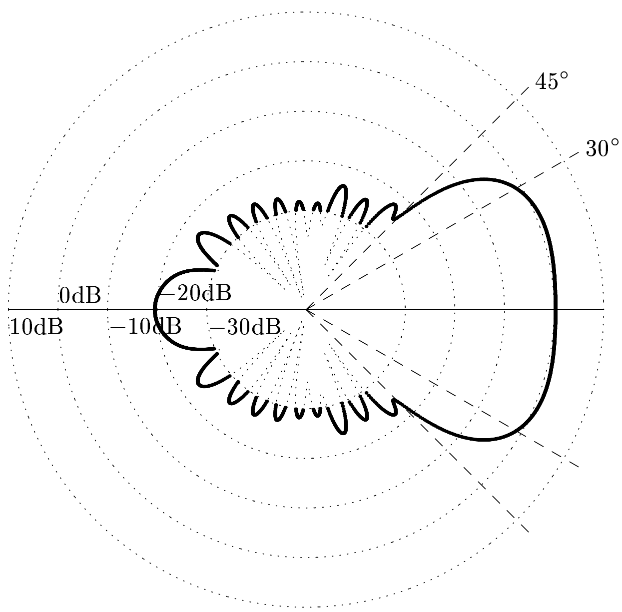

FIGURE 10. Example of antenna array weight design for 12-element array with spacing $d = 0.45\lambda$. The constraints require ±2dB coverage over a beam of ±30° and the objective is to minimize the maximum pattern response outside a ±45° sector. The sidelobe level achieved is about −19dB.

8 Conclusions

We have shown that a variety of magnitude FIR filter design problems can be formulated, in terms of the autocorrelation coefficients, as (possibly nonlinear) convex semi-infinite optimization problems. As a result, the globally optimal solution can be efficiently computed. By considering *nonlinear* convex optimization problems, we can solve a number of problems of practical interest, *e.g.*, minimax decibel problems, with an efficiency not much less than standard methods that rely on, for example, linear or quadratic programming.

Acknowledgments: This research was supported in part by AFOSR (under F49620-95-1-0318), NSF (under ECS-9222391 and EEC-9420565), and MURI (under F49620-95-1-0525). The authors would like to thank Babak Hassibi, Laurent El Ghaoui, and Hervé Lebret for very helpful discussions and comments. They also thank Paul Van Dooren for bringing reference [59] to their attention.

9 References

[1] F. A. Aliyev, B. A. Bordyug, and V. B. Larin. Factorization of polynomial matrices and separation of rational matrices. *Soviet Journal of Computer and Systems Sciences*, 28(6):47–58, 1990.

[2] F. A. Aliyev, B. A. Bordyug, and V. B. Larin. Discrete generalized algebraic Riccati equations and polynomial matrix factorization. *Syst. Control Letters*, 18:49–59, 1992.

[3] B. Anderson. An algebraic solution to the spectral factorization problem. *IEEE Trans. Aut. Control*, AC-12(4):410–414, Aug. 1967.

[4] B. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-Hall, 1979.

[5] B. Anderson and S. Vongpanitlerd. *Network analysis and synthesis: a modern systems theory approach*. Prentice-Hall, 1973.

[6] B. D. O. Anderson, K. L. Hitz, and N. D. Diem. Recursive algorithm for spectral factorization. *IEEE Transactions on Circuits and Systems*, 21:742–750, 1974.

[7] E. J. Anderson and P. Nash. *Linear Programming in Infinite-Dimensional Spaces: Theory and Applications*. John Wiley & Sons, 1987.

[8] E. J. Anderson and A. B. Philpott, editors. *Infinite Programming*. Springer-Verlag Lecture Notes in Economics and Mathematical Systems, Sept. 1984.

[9] V. Balakrishnan and S. Boyd. Global optimization in control system analysis and design. In C. T. Leondes, editor, *Control and Dynamic Systems: Advances in Theory and Applications*, volume 53. Academic Press, New York, New York, 1992.

[10] F. L. Bauer. Ein direktes Iterationsverfahren zur Hurwitz-Zerlegung eines Polynoms. *Arch. Elek. Übertr.*, 9:844–847, 1955.

[11] R. G. Bland, D. Goldfarb, and M. J. Todd. The ellipsoid method: A survey. *Operations Research*, 29(6):1039–1091, 1981.

[12] R. Boite and H. Leich. A new procedure for the design of high order minimum phase FIR digital or CCD filters. *Signal Processing*, 3:101–108, 1981.

[13] S. Boyd and C. Barratt. *Linear Controller Design: Limits of Performance*. Prentice-Hall, 1991.

[14] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.

[15] S. Boyd and L. Vandenberghe. Introduction to convex optimization with engineering applications. Course Notes, 1997. `http://www-leland.stanford.edu/class/ee364/`.

[16] S. Boyd, L. Vandenberghe, and M. Grant. Efficient convex optimization for engineering design. In *Proceedings IFAC Symposium on Robust Control Design*, pages 14–23, Sept. 1994.

[17] D. Burnside and T. W. Parks. Optimal design of fir filters with the complex chebyshev error criteria. *IEEE Transactions on Signal Processing*, 43(3):605–616, March 1995.

[18] X. Chen and T. W. Parks. Design of optimal minimum phase FIR filters by direct factorization. *Signal Processing*, 10:369–383, 1986.

[19] E. W. Cheney. *Introduction to Approximation Theory*. Chelsea Publishing Company, New York, second edition, 1982.

[20] D. K. Cheng. Optimization techniques for antenna arrays. *Proceedings of the IEEE*, 59(12):1664–1674, Dec. 1971.

[21] J. O. Coleman. *The Use of the FF Design Language for the Linear Programming Design of Finite Impulse Response Digital Filters for Digital Communication and Other Applications*. PhD thesis, Unversity of Washington, 1991.

[22] R. S. Elliott. *Antenna Theory and Design*. Prentice-Hall, 1981.

[23] J. Elzinga and T. Moore. A central cutting plane algorithm for the convex programming problem. *Math. Program. Studies*, 8:134–145, 1975.

[24] P. A. Fuhrmann. Elements of factorization theory from a polynomial point of view. In H. Nijmeijer and J. M. Schumacher, editors, *Three Decades of Mathematical System Theory*, volume 135 of *Lecture Notes in Control and Information Sciences*, pages 148–178. Springer Verlag, 1989.

[25] O. Herrmann and H. W. Schüssler. Design of nonrecursive digital filters with minimum-phase. *Electronic Letter*, 6:329–330, 1970.

[26] R. Hettich. A review of numerical methods for semi-infinite programming and applications. In A. V. Fiacco and K. O. Kortanek, editors, *Semi-Infinite Programming and Applications*, pages 158–178. Springer, Berlin, 1983.

[27] R. Hettich and K. O. Kortanek. Semi-infinite programming: theory, methods and applications. *SIAM Review*, 35:380–429, 1993.

[28] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*, volume 306 of *Grundlehren der mathematischen Wissenschaften.* Springer-Verlag, New York, 1993.

[29] Y. Kamp and C. J. Wellekens. Optimal design of minimum-phase FIR filters. *IEEE Trans. Acoust., Speech, Signal Processing*, 31(4):922–926, 1983.

[30] J. E. Kelley. The cutting-plane method for solving convex programs. *J. Soc. Indust. Appl. Math*, 8(4):703–712, Dec. 1960.

[31] P. Lancaster and L. Rodman. Solutions of the continuous and discrete time algebraic Riccati equations: a review. In S. Bittanti, A. J. Laub, and J. C. Willems, editors, *The Riccati equation*, pages 11–51. Springer Verlag, Berlin, Germany, 1991.

[32] V. B. Larin. An algorithm for factoring a matrix polynomial relative to a unit-radius circle. *Journal of Automation and Information Sciences*, 26(1):1–6, 1993.

[33] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14:699–719, 1966.

[34] H. Lebret and S. Boyd. Antenna array pattern synthesis via convex optimization. *IEEE Trans. on Signal Processing*, 45(3):526–532, March 1997.

[35] M. S. Lobo, L. Vandenberghe, and S. Boyd. SOCP*: Software for Second-Order Cone Programming.* Information Systems Laboratory, Stanford University, 1997.

[36] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Second-order cone programming: interior-point methods and engineering applications. *Linear Algebra and Appl.*, 1997. Submitted.

[37] M. T. Ma. *Theory and Application of Antenna Arrays.* John Wiley and Sons, 1974.

[38] G. A. Mian and A. P. Nainer. A fast procedure to design equiripple minimum-phase FIR filters. *IEEE Trans. Circuits Syst.*, 29(5):327–331, 1982.

[39] Y. Nesterov and A. Nemirovsky. *Interior-point polynomial methods in convex programming*, volume 13 of *Studies in Applied Mathematics.* SIAM, Philadelphia, PA, 1994.

[40] A. V. Oppenheim and R. W. Schafer. *Digital Signal Processing.* Prentice-Hall, Englewood Cliffs, N. J., 1970.

[41] E. Panier and A. Tits. A globally convergent algorithm with adaptively refined discretization for semi-infinite optimization problems arising in engineering design. *IEEE Trans. Aut. Control*, AC-34(8):903–908, 1989.

[42] A. Papoulis. *Signal Analysis.* McGraw-Hill, New York, 1977.

[43] T. W. Parks and C. S. Burrus. *Digital Filter Design.* Topics in Digital Signal Processing. John Wiley & Sons, New York, 1987.

[44] E. Polak. Semi-infinite optimization in engineering design. In A. Fiacco and K. Kortanek, editors, *Semi-infinite Programming and Applications.* Springer-Verlag, 1983.

[45] A. W. Potchinkov and R. M. Reemtsen. The design of FIR filters in the complex plane by convex optimization. *Signal Processing*, 46(2):127–146, Oct. 1995.

[46] R. M. Reemtsen. A cutting-plane method for solving minimax problems in the complex plane. *Numerical Algorithms*, 2:409–436, 1992.

[47] R. M. Reemtsen. Some outer approximation methods for semi-infinite optimization problems. *Journal of Computational and Applied Mathematics*, 53:87–108, 1994.

[48] R. M. Reemtsen and A. W. Potchinkov. FIR filter design in regard to frequency response, magnitude, and phase by semi-infinite programming. In J. Guddat, H. T. Jongen, F. Nozicka, G. Still, and F. Twilt, editors, *Parametric Optimization and Related Topics IV*. Verlag Peter Lang, Frankfurt, 1996.

[49] J. Rissanen. Algorithms for triangular decomposition of block Hankel and Toeplitz matrices with application to factoring positive matrix polynomials. *Mathematics of Computation*, 27:147–154, 1973.

[50] J. Rissanen and L. Barbosa. Properties of infinite covariance matrices and stability of optimum predictors. *Information Sciences*, 1:221–236, 1969.

[51] H. Samueli. Linear programming design of digital data transmission filters with arbitrary magnitude specifications. In *Conference Record, International Conference on Communications*, pages 30.6.1–30.6.5. IEEE, June 1988.

[52] L. L. Scharf. *Statistical Signal Processing*. Addison-Wesley, 1991.

[53] K. Steiglitz, T. W. Parks, and J. F. Kaiser. METEOR: A constraint-based fir filter design program. *IEEE Trans. Acoust., Speech, Signal Processing*, 40(8):1901–1909, Aug. 1992.

[54] M. Talbot-Smith, editor. *Audio Engineer's Reference Handbook*. Focal Press, Oxford, 1994.

[55] L. Vandenberghe and S. Boyd. Connections between semi-infinite and semidefinite programming. In R. Reemtsen and J.-J. Rueckmann, editors, *Proceedings of the International Workshop on Semi-Infinite Programming*. 1996. To appear.

[56] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, Mar. 1996.

[57] R. J. Vanderbei. LOQO User's Manual. Technical Report SOL 92–05, Dept. of Civil Engineering and Operations Research, Princeton University, Princeton, NJ 08544, USA, 1992.

[58] R. J. Vanderbei. *Linear Programming: Foundations and Extensions*. Kluwer, Boston, 1996.

[59] Z. Vostrý. New algorithm for polynomial spectral factorization with quadratic convergence. Part I. *Kybernetika*, 11:415–422, 1975.

[60] G. Wilson. Factorization of the covariance generating function of a pure moving average process. *SIAM J. on Numerical Analysis*, 6:1–7, 1969.

[61] S. J. Wright. *Primal-Dual Interior-Point Methods.* SIAM, Philadelphia, 1997.

[62] S.-P. Wu and S. Boyd. SDPSOL: *A Parser/Solver for Semidefinite Programming and Determinant Maximization Problems with Matrix Structure. User's Guide, Version Beta.* Stanford University, June 1996.

[63] S.-P. Wu, S. Boyd, and L. Vandenberghe. FIR filter design via semidefinite programming and spectral factorization. In *Proc. IEEE Conf. on Decision and Control*, pages 271–276, 1996.

[64] D. C. Youla. On the factorization of rational matrices. *IRE Trans. Information Theory*, IT-7(3):172–189, July 1961.

Appendix: Spectral factorization

In this section we give a brief overview of methods for spectral factorization. This problem has been studied extensively; general references include [64, 3, 5, 42, 4]. We consider spectral factorization for real-valued h; the extensions to h complex (which arises in the antenna weight design problem, for example) can be found in the references. We assume (without loss of generality) that $r(n-1) \neq 0$ (since otherwise we can define n as the smallest k such that $r(t) = 0$ for $t \geq k$).

In this appendix we assume that R satisfies the following strengthened version of the spectral factorization condition:

$$R(\omega) > 0 \quad \text{for all} \quad \omega \in \mathbf{R}.$$

All of the methods described below can be extended to handle the case where $R(\omega)$ is nonnegative for all ω, but zero for some ω. The strengthened condition greatly simplifies the discussion; full details in the general case can be found in the references.

We first describe the classical method based on factorizing a polynomial. We define the rational complex function

$$T(z) = r(n-1)z^{n-1} + \cdots + r(1)z + r(0) + r(1)z^{-1} + \cdots + r(n-1)z^{-(n-1)},$$

so that $R(\omega) = T(e^{j\omega})$. We will show how to construct S, a polynomial in z^{-1},

$$S(z) = h(0) + h(1)z^{-1} + \cdots + h(n-1)z^{-(n-1)}, \tag{1.23}$$

so that $T(z) = S(z)S(z^{-1})$. Expanding this product and equating coefficients of z^k shows that r are the autocorrelation coefficients of h. In other words, the coefficients of S give the desired spectral factorization: $H(\omega) = S(e^{j\omega})$.

Let $P(z) = z^{n-1}T(z)$. Then P is a polynomial of degree $2(n-1)$ (since $r(n-1) \neq 0$), with real coefficients. Now suppose $\lambda \in \mathbf{C}$ is a zero of P, *i.e.*, $P(\lambda) = 0$. Since $P(0) = r(n-1) \neq 0$, we have $\lambda \neq 0$. Since the coefficients of P are real, we have that $P(\overline{\lambda}) = 0$, *i.e.*, $\overline{\lambda}$ is also a zero of P. From the fact that $T(z^{-1}) = T(z)$, we also see that

$$P(\lambda^{-1}) = \lambda^{-(n-1)}T(\lambda^{-1}) = 0,$$

i.e., λ^{-1} is also a zero of P. In other words, the zeros of $P(z)$ are symmetric with respect to the unit circle and also the real axis. For every zero of P that is inside the unit circle, there is a corresponding zero outside the unit circle, and vice versa. Moreover, our strengthened spectral factorization condition implies that none of the zeros of P can be on the unit circle.

Now let $\lambda_1, \ldots, \lambda_{n-1}$ be the $n-1$ roots of P that are inside the unit circle. These roots come in pairs if they are complex: if λ is a root inside the unit circle and is complex, then so is $\overline{\lambda}$, hence $\lambda_1, \ldots, \lambda_{n-1}$ has conjugate symmetry. Note that the $2(n-1)$ roots of P are precisely

$$\lambda_1, \ldots, \lambda_{n-1}, \quad 1/\overline{\lambda_1}, \ldots, 1/\overline{\lambda_{n-1}}.$$

It follows that we can factor P in the form

$$P(z) = c\prod_{i=1}^{n-1}(z - \lambda_i)(\overline{\lambda}_i z - 1),$$

where c is a constant. Thus we have

$$T(z) = z^{-(n-1)}P(z) = c\prod_{i=1}^{n-1}(1 - \lambda_i z^{-1})(\overline{\lambda}_i z - 1).$$

By our strengthened assumption, $R(0) > 0$, so

$$R(0) = T(1) = c\prod_{i=1}^{n-1}|1 - \lambda_i|^2 > 0,$$

so that $c > 0$.

Finally we can form a spectral factor. Define

$$S(z) = \sqrt{c}\,\prod_{i=1}^{n-1}(1 - \lambda_i z^{-1}),$$

so that $T(z) = S(z)S(z^{-1})$. Since $\lambda_1, \ldots, \lambda_{n-1}$ have conjugate symmetry, the coefficients of S are real, and provide the required FIR impulse response coefficients from (1.23).

The construction outlined here yields the so-called *minimum-phase spectral factor*. (Other spectral factors can be obtained by different choice of $n-1$ of the $2(n-1)$ roots of P).

Polynomial factorization is the spectral factorization method used by Herrmann and Schüssler in the early paper [25]. Root finding methods that take advantage of the special structure of P improve the method; see [18] and the references therein. Root finding methods are generally used only when n is small, say, several 10s.

Several other spectral factorization methods compute the spectral factor without computing the roots of the polynomial P. One group of methods [10, 50, 49] is based on the Cholesky factorization of the infinite banded Toeplitz matrix

$$\begin{bmatrix} r(0) & r(1) & r(2) & \cdots & r(n-1) & 0 & \cdots \\ r(1) & r(0) & r(1) & \cdots & r(n-2) & r(n-1) & \cdots \\ r(2) & r(1) & r(0) & \cdots & r(n-3) & r(n-2) & \cdots \\ \vdots & \vdots & \vdots & \ddots & & & \\ r(n-1) & r(n-2) & r(n-3) & & & & \\ 0 & r(n-1) & r(n-2) & & & & \\ \vdots & \vdots & \vdots & & & & \end{bmatrix}.$$

This matrix is positive definite (*i.e.*, all its principal minors are positive definite) if $R(\omega) > 0$ for all ω, and it was shown in [50] that the elements of the Cholesky factors converge to the coefficients of the minimum-phase spectral factor of R. As a consequence, fast recursive algorithms for Cholesky factorization of positive definite Toeplitz matrices also yield iterative methods for spectral factorization.

Wilson [60] and Vostrý [59] have developed a method for spectral factorization based on directly applying Newton's method to the set of nonlinear equations (1.5). Their method has quadratic convergence, so very high accuracy can be obtained rapidly once an approximate solution is found. (Indeed, Newton's method can be used to refine an approximate solution obtained by any other method.)

Anderson *et al.* [3, 6] show that the minimum-phase spectral factor can also be obtained from the solution of a discrete-time algebraic Riccati equation. They present an iterative method, based on iterating the corresponding Riccati difference equation, and retrieve from this some of the earlier spectral factorization methods (*e.g.*, [10]) as special cases.

We conclude by outlining a fourth method, which is based on the *fast Fourier transform* (FFT) (see, *e.g.*, [42, §7.2] or [52, §10.1]). The idea behind the method is usually credited to Kolmogorov.

The method is based on the following explicit construction of the minimum-phase spectral factor S_{mp} (as described above) from r. It can be shown that $\log S_{\mathrm{mp}}$ is defined and analytic in the exterior of the unit disk, *i.e.*, it can be expressed as a power series in z^{-1},

$$\log S_{\rm mp}(z) = \sum_{k=0}^{\infty} a_k z^{-k}$$

for $|z| > 1$, where $a_k \in \mathbf{R}$. Now consider the real part of $\log S_{\rm mp}$ on the unit circle:

$$\Re \log S_{\rm mp}(e^{j\omega}) = \log |S_{\rm mp}(e^{j\omega})| = (1/2)\log R(\omega) = \sum_{k=0}^{\infty} a_k \cos k\omega.$$

Therefore we can find the coefficients a_k as the Fourier coefficients of the function $(1/2)\log R(\omega)$ (where we use the strengthened spectral factorization condition), *i.e.*,

$$a_k = \frac{1}{2\pi}\int_0^{2\pi} (1/2)\log R(\omega)\; e^{-jk\omega}\; d\omega, \quad k = 0, 1, \ldots . \tag{1.24}$$

Once the coefficients a_k are known, we can reconstruct $S_{\rm mp}(e^{j\omega})$ as

$$S_{\rm mp}(e^{j\omega}) = \exp \sum_{k=0}^{\infty} a_k e^{-jk\omega}.$$

The Fourier coefficients of $S_{\rm mp}$ give us the required impulse response coefficients:

$$h(t) = \frac{1}{2\pi}\int_0^{2\pi} e^{j\omega t} \exp \sum_{k=0}^{\infty} a_k e^{-jk\omega}\; d\omega, \quad t = 0, \ldots, n-1. \tag{1.25}$$

Taken together, equations (1.24) and (1.25) give an explicit construction of the impulse response coefficients of the minimum-phase spectral factor, starting from R.

In the language of signal processing, this construction would be described as follows. We know the log-magnitude of $S_{\rm mp}$ since it is half the log-magnitude of R, which is given (or found from r via a Fourier transform). We apply a Hilbert transform to find the phase of $S_{\rm mp}$, and then by exponentiating get $S_{\rm mp}$ (in the frequency domain). Its Fourier coefficients are the desired impulse response.

The method is applied in practice as follows. We pick $\tilde{n}$ as a power of two with $\tilde{n} \gg n$ (say, $\tilde{n} \geq 15n$). We use an FFT of order $\tilde{n}$ to obtain (from r) $R(\omega)$ at $\tilde{n}$ points uniformly spaced on the unit circle. Assuming the result is nonnegative, we compute half its log. Another FFT yields the coefficients $a_0, \ldots, a_{\tilde{n}-1}$ (using (1.24)). Another FFT yields the function $\sum_{k=0}^{\infty} a_k e^{-jk\omega}$ for ω at the $\tilde{n}$ points around the unit circle. A final FFT yields $h(0), \ldots, h(n-1)$ using (1.25). (In a practical implementation of this algorithm symmetry and realness can be exploited at several stages, and the last FFT can be of order n.)

FFT-based spectral factorization is efficient and numerically stable, and is readily applied even when n, the order of the filter, is large, say several hundred. This method has been independently rediscovered several times; see, *e.g.*, [12, 38].