Sparse Filter Design Under a Quadratic Constraint: Low-Complexity Algorithms

Dennis Wei, Charles K. Sestok, and Alan V. Oppenheim

***Abstract*—This paper considers three problems in sparse filter design, the first involving a weighted least-squares constraint on the frequency response, the second a constraint on mean squared error in estimation, and the third a constraint on signal-to-noise ratio in detection. The three problems are unified under a single framework based on sparsity maximization under a quadratic performance constraint. Efficient and exact solutions are developed for specific cases in which the matrix in the quadratic constraint is diagonal, block-diagonal, banded, or has low condition number. For the more difficult general case, a low-complexity algorithm based on backward greedy selection is described with emphasis on its efficient implementation. Examples in wireless channel equalization and minimum-variance distortionless-response beamforming show that the backward selection algorithm yields optimally sparse designs in many instances while also highlighting the benefits of sparse design.**

I. Introduction

The efficient implementation of discrete-time filters continues to be of interest given their widespread use in signal processing systems. In many applications, the cost of implementation is dominated by arithmetic operations and can therefore be reduced by designing filters with fewer non-zero coefficients, i.e., sparse filters. Sparse designs are beneficial not only in terms of computation but also other cost metrics such as hardware and energy consumption, depending on the form of implementation. For instance, in an integrated circuit, multipliers and adders may be deactivated or even eliminated to save power and area, or the supply voltage may be lowered to take advantage of a slower computation rate [1]. Sparsity is also of considerable interest for linear sensor arrays [2], a close mathematical parallel to discrete-time FIR filters, since the number of potentially costly array elements can be reduced.

Previous work on sparse filter design has occurred on several fronts. For the classical problem of approximating an ideal frequency response, the techniques can be broadly categorized into two approaches. In the first approach, which is applicable mostly to frequency-selective filters, the locations of zero-valued coefficients are pre-determined in accordance with the desired frequency response. Interpolated FIR filters [3], [4] and frequency-response masking [5], [6] can be viewed in this way since they incorporate a sparse filter with a regular sparsity pattern cascaded with one or more equalizing filters. Cascade structures however are not applicable to arrays. Sparse direct-form designs for approximately nth-band filters were developed in [7] utilizing the property of nth-band filters of having every nth impulse response coefficient equal to zero except for the central coefficient. The second approach is more general and attempts to optimize the locations of zero-valued coefficients so as to maximize their number subject to frequency response constraints. The resulting combinatorial optimization problem can be solved exactly using integer programming [8], [9]. The complexity of optimal design has also motivated the use of low-complexity heuristics, based for example on forcing small coefficients to zero [10], orthogonal matching pursuit [11], ℓ_1 relaxation or iterative thinning [12]. A non-convex approximate measure of sparsity based on p-norms has also been proposed with p fixed [2] as well as gradually decreasing toward zero [13].

All of the references above focus on approximation according to a Chebyshev error criterion. In comparison, weighted least-squares criteria have received less attention. As discussed in [14], a weighted least-squares metric is employed as an alternative to a Chebyshev metric because of greater tractability and an association with signal energy or power. However, this tractability is of limited use in designing sparse filters since the problem is still combinatorial when the weighting is non-uniform. For weighted least-squares sparse filter design, approaches based on zeroing small coefficients [15] and subset selection [16] have been developed.

Discrete-time filters are also used to estimate the values of a signal from those of another. In the context of sparse design, a particularly important example is the equalization of communication channels, which involves the estimation of transmitted values from received values corrupted by noise and inter-symbol interference. Several researchers have observed that the sparse power-delay profiles of many communication channels can be exploited to design sparse equalizers. Exact algorithms for minimizing the mean squared estimation error given a fixed number of equalizer taps are developed in [17] and [18], the former based on branch-and-bound for discrete-time equalizers and the latter on nonlinear optimization for continuous-time tapped-delay-line equalizers. A less complex heuristic method is to choose the locations of non-zero equalizer coefficients to coincide with the locations of large channel

Manuscript received April 14, 2012; accepted October 09, 2012. This work was supported in part by the Texas Instruments Leadership University Program.

D. Wei is with the Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109 USA; e-mail: dlwei@eecs.umich.edu.

C. K. Sestok is with the Systems and Applications R&D Center, Texas Instruments, 12500 TI Boulevard, MS 8649, Dallas, TX 75243 USA; e-mail: sestok@ti.com.

A. V. Oppenheim is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Room 36-615, 77 Massachusetts Avenue, Cambridge, MA 02139 USA; e-mail: avo@mit.edu.

coefficients [19]. This approach is refined in [20] and [21], which predict the locations of large coefficients in conventional equalizers and allocate taps in sparse equalizers accordingly. A modified decision-feedback equalizer (DFE) structure is proposed in [22] to better exploit the sparsity of the channel response. An alternative class of heuristic methods allocates taps according to simplified mean squared error (MSE) or output signal-to-noise ratio (SNR) metrics. The allocation can be done in a single pass [23], two alternating passes [24], or one tap at a time using forward greedy selection [25], [26]. The channel sparsity is used in [26] to further reduce the tap allocation search space.

Signal prediction is a variant of the estimation problem in which past values of a signal are used to predict future values. Sparse linear prediction for speech coding is proposed in [27] using iteratively reweighted ℓ_1 minimization to promote sparsity in the residuals and improve coding performance.

A third context in which filters are used is in the detection of signals in noisy environments, where the objective of filtering is to increase the probability of detection. A widely used performance measure in detection is the SNR of the filter output, which is well-known to be monotonically related to the probability of detection in Gaussian noise [28]. The design of linear detectors that use only a subset of the available measurements was considered in [29], [30] as a way of reducing communication costs in distributed systems.

In this paper, we draw from the applications above and consider three problems in sparse filter design, the first involving a weighted least-squares constraint on the frequency response, the second a constraint on MSE in estimation, and the third a constraint on SNR in detection. It is shown that all three problems can be placed under a common framework corresponding to the following minimization problem:

$$\min_{\mathbf{b}} \quad \|\mathbf{b}\|_0 \qquad \text{s.t.} \qquad (\mathbf{b}-\mathbf{c})^T\mathbf{Q}(\mathbf{b}-\mathbf{c}) \leq \gamma, \quad (1)$$

where $\mathbf{b}$ is a vector of N coefficients, $\mathbf{Q}$ is an $N \times N$ symmetric positive definite matrix, $\mathbf{c}$ is a vector of length N, and $\gamma > 0$. We use for convenience the zero-norm notation $\|\mathbf{b}\|_0$ to refer to the number of non-zero components in $\mathbf{b}$. Our formulation allows for a unified approach in solving not only the three stated problems but also other problems with quadratic performance criteria.

It is important to note that the sparse filter design problem as stated in (1) differs in two key respects from the sparse linear inverse problem, i.e., the problem of obtaining sparse approximate solutions to linear equations, and more specifically its manifestations in compressive sensing with noisy measurements [31]–[34], atomic decomposition in overcomplete dictionaries [35], sparsity-regularized image restoration [36]–[40], and sparse channel estimation [41]–[43]. The sparse linear inverse problem can be formulated in general as

$$\min_{\mathbf{x}} \quad \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{\Phi x}-\mathbf{y}\|_2^2 = (\mathbf{\Phi x}-\mathbf{y})^T(\mathbf{\Phi x}-\mathbf{y}) \leq \varepsilon, \quad (2)$$

where ε is a limit on the residual $\mathbf{\Phi x}-\mathbf{y}$. The first distinction between (1) and (2) is in the nature of the sets of feasible solutions. In many applications of (2), the dimension of $\mathbf{y}$ is significantly lower than that of $\mathbf{x}$ and the system of equations is underdetermined. This is deliberately the case in compressive sensing, overcomplete decomposition, and adaptive channel estimation. As a consequence, the matrix $\mathbf{\Phi}^T\mathbf{\Phi}$, which corresponds to $\mathbf{Q}$ in (1), is rank-deficient and the feasible set is not bounded as in (1) but instead has infinite extent in certain directions. The second difference between sparse filter design and sparse linear inverse problems is one of perspective. In compressive sensing, image restoration, and channel estimation, sparsity or near-sparsity is assumed to enable reconstruction from fewer measurements, leading to a formulation such as (2). However, the actual sparsity level of a solution to (2) is of secondary importance as long as the primary goal of accurate reconstruction is achieved. In contrast, in sparse filter design, maximizing sparsity is the main objective, while no assumption is made regarding the expected level of sparsity. An algorithm that produces near-sparse designs having many small but non-zero coefficients is not sufficient by itself.

Given the above differences, sparse filter design as considered in this paper requires a somewhat different set of approaches than for the sparse linear inverse problem. This paper focuses on design algorithms that are low in complexity, which are important when computation is limited, for example when a filter is redesigned adaptively. In some cases, low-complexity algorithms are sufficient to ensure optimal solutions to (1). We describe several such cases in which the matrix $\mathbf{Q}$ is diagonal, block-diagonal, banded, or has low condition number. More generally however, solving (1) is computationally difficult. For the general case, we discuss an efficient algorithm based on backward greedy selection, similar to one of the algorithms in [12] but adapted to the quadratic performance criterion in (1), and in contrast to the forward greedy approach of [25], [26]. In backward selection, coefficients are removed one at a time in such a way that the performance degradation from one iteration to the next is minimized. A similar idea has also been proposed for subset selection in regression [44]. In design examples, an extensive comparison with an exact branch-and-bound algorithm shows that the backward selection algorithm often yields optimal or near-optimal solutions, even for moderately-sized instances. Compared to other heuristics such as forward selection, backward selection performs favorably and with greater consistency. The examples also illustrate the benefits of sparse design in wireless channel equalization and minimum-variance distortionless-response (MVDR) beamforming.

In a companion paper [45], we present an exact algorithm for the general case in which $\mathbf{Q}$ does not have special structure. Some preliminary results related to the present paper and [45] were reported in [46]. The present paper builds upon [46] by significantly expanding the treatment of special cases in Section III beyond the diagonal case, presenting a backward selection algorithm for the general case, and demonstrating the near-optimality of the algorithm and favorable comparisons to other heuristics in filter design examples. The present paper differs fundamentally from the companion paper [45] in its approach: [45] focuses on a higher-complexity exact algorithm based on branch-and-bound and lower bounding techniques, in contrast to the lower-complexity algorithms in the present

paper.

This paper is organized as follows: In Section II, the three filter design problems considered in this work are formulated and reduced to (1). In Section III, we present efficient methods for solving (1) when the matrix $\mathbf{Q}$ is diagonal, block diagonal, banded, or has low condition number. We also indicate briefly why an extension to the general case of unstructured $\mathbf{Q}$ does not appear to be straightforward. In Section IV, we describe a low-complexity backward selection algorithm for the general case. In Section V, the near-optimality of the backward selection algorithm and its superiority over other heuristics are validated through design examples.

II. Problem formulations and reductions

In this section, we formulate the problems of sparse filter design for weighted least-squares approximation of frequency responses, for estimation or prediction under an MSE constraint, and for signal detection under an SNR constraint. All three problems can be reduced to (1), making it sufficient to focus on (1) alone. More specifically, it is shown that the performance constraint in each problem can be reduced to the inequality

$$\mathbf{b}^T\mathbf{Q}\mathbf{b} - 2\mathbf{f}^T\mathbf{b} \leq \beta, \tag{3}$$

which is equivalent to the constraint in (1) with $\mathbf{f} = \mathbf{Q}\mathbf{c}$ and $\beta = \gamma - \mathbf{c}^T\mathbf{Q}\mathbf{c}$.

This paper focuses on FIR filters. In the FIR case, the total number of coefficients, N, is usually determined by the maximum allowable number of delay elements or array length. Thus we refer to N as the length of the filter, with the understanding that the final design may require fewer delays if coefficients at the ends of the vector $\mathbf{b}$ are zero.

A. Weighted least-squares filter design

The first problem is to design a causal FIR filter with coefficients $b_0, \ldots, b_{N-1}$ and frequency response

$$H(e^{j\omega}) = \sum_{n=0}^{N-1} b_n e^{-j\omega n} \tag{4}$$

chosen to approximate a desired frequency response $D(e^{j\omega})$ (assumed to be conjugate symmetric). Specifically, the weighted integral of the squared error is constrained to not exceed a tolerance δ, i.e.,

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)\left|H(e^{j\omega}) - D(e^{j\omega})\right|^2 d\omega \leq \delta, \tag{5}$$

where $W(\omega)$ is a non-negative and even-symmetric weighting function. The number of non-zero coefficients is to be minimized. Substituting (4) into (5), expanding, and comparing the result with (3), we can identify

$$Q_{mn} = \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)\cos\big((m-n)\omega\big)d\omega, \quad m = 0, \ldots, N-1, \quad n = 0, \ldots, N-1, \tag{6a}$$

$$f_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)D(e^{j\omega})e^{j\omega n}\,d\omega, \quad n = 0, \ldots, N-1, \tag{6b}$$

$$\beta = \delta - \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)\left|D(e^{j\omega})\right|^2 d\omega. \tag{6c}$$

The matrix $\mathbf{Q}$ defined by (6a) is symmetric, Toeplitz, and positive definite, the last property holding as long as $W(\omega)$ is non-zero over some interval. The fact that $\mathbf{Q}$ is Toeplitz is relatively unimportant as we will often work with submatrices extracted from $\mathbf{Q}$, which in general are no longer Toeplitz.

In the present case, the parameter γ is given by

$$\gamma = \delta - \left(\frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega)\left|D(e^{j\omega})\right|^2 d\omega - \mathbf{c}^T\mathbf{Q}\mathbf{c}\right).$$

It can be seen from (3) and (5) that $\mathbf{c} = \mathbf{Q}^{-1}\mathbf{f}$ corresponds to the minimum-error design and the quantity in parentheses above is the minimum error. Hence γ is the amount of additional error permitted relative to the optimal non-sparse design.

B. Estimation, prediction, and equalization

A second problem that can be reduced to the formulation in (1) is the estimation of a random process $x[n]$ from observations of a random process $y[n]$ under the assumption that $x[n]$ and $y[n]$ are jointly WSS. The estimate $\hat{x}[n]$ is produced by processing $y[n]$ with a causal FIR filter of length N,

$$\hat{x}[n] = \sum_{m=0}^{N-1} b_m y[n-m]. \tag{7}$$

The goal is to minimize the number of non-zero coefficients b_m while keeping the mean-squared estimation error below a threshold δ, i.e.,

$$E\left\{(\hat{x}[n] - x[n])^2\right\} \leq \delta. \tag{8}$$

Substituting (7) into (8), expanding, and comparing with (3), we find

$$Q_{mn} = \phi_{yy}\left[|m-n|\right], \tag{9a}$$

$$f_n = \phi_{xy}[n], \tag{9b}$$

$$\beta = \delta - \phi_{xx}[0], \tag{9c}$$

where the cross-correlation is defined as $\phi_{xy}[m] = E\{x[n+m]y[n]\}$. The matrix $\mathbf{Q}$ is again symmetric, Toeplitz, and positive definite since it corresponds to an auto-correlation function. In the estimation context, the vector $\mathbf{c} = \mathbf{Q}^{-1}\mathbf{f}$ corresponds to the causal Wiener filter of length N, $\phi_{xx}[0] - \mathbf{c}^T\mathbf{Q}\mathbf{c}$ is the corresponding error, and γ is again equal to the allowable excess error.

The problem of p-step linear prediction is a special case of the estimation problem with $x[n] = y[n+p]$ and p a positive integer. Equation (9a) remains unchanged while $\phi_{xy}[n]$ is

replaced with $\phi_{yy}[n+p]$ in (9b) and $\phi_{xx}[0]$ with $\phi_{yy}[0]$ in (9c).

An important application of the basic estimation problem above is to the equalization of communication channels. In channel equalization, $x[n]$ represents a transmitted sequence, and in the case of linear equalization, $y[n]$ represents the received sequence and can be modelled according to

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k] + \eta[n],$$

where $h[k]$ represents the overall impulse response due to the combination of the transmit pulse, channel, and receive filter, and $\eta[n]$ is additive noise, assumed to be zero-mean, stationary with autocorrelation $\phi_{\eta\eta}[m]$, and uncorrelated with $x[n]$. Under this channel model, the auto-correlation and cross-correlation in (9) can be expressed as

$$\phi_{yy}[m] = \sum_{k=-\infty}^{\infty} \phi_{hh}[k]\phi_{xx}[m-k] + \phi_{\eta\eta}[m], \tag{10a}$$

$$\phi_{xy}[m] = \sum_{k=-\infty}^{\infty} h[k]\phi_{xx}[m+k], \tag{10b}$$

where $\phi_{hh}[k]$ is the deterministic autocorrelation of $h[n]$. The formulation in this subsection can be extended straightforwardly to more elaborate equalization techniques such as decision-feedback equalization, channel shortening, and MIMO; see [47] for more details on these extensions.

Under the complex-baseband equivalent channel model for quadrature-amplitude modulation (QAM), all of the quantities above become complex-valued, including the equalizer coefficients b_n, and $\mathbf{Q}$ becomes Hermitian positive definite. We can accommodate complex coefficients within our real-valued framework by interleaving the real and imaginary parts of $\mathbf{b}$ to create a $2N$-dimensional real-valued vector $\widetilde{\mathbf{b}} = \begin{bmatrix}\mathrm{Re}(b_1) & \mathrm{Im}(b_1) & \mathrm{Re}(b_2) & \mathrm{Im}(b_2) & \dots\end{bmatrix}^T$. The vector $\mathbf{c}$, which is still equal to $\mathbf{Q}^{-1}\mathbf{f}$, is transformed similarly, and $\mathbf{Q}$ is transformed by replacing each complex-valued entry $\mathbf{Q}_{mn}$ with the 2×2 submatrix

$$\begin{bmatrix}\mathrm{Re}(Q_{mn}) & -\mathrm{Im}(Q_{mn}) \\ \mathrm{Im}(Q_{mn}) & \mathrm{Re}(Q_{mn})\end{bmatrix}.$$

The zero-norm $\left\|\widetilde{\mathbf{b}}\right\|_0$ now measures the number of non-zero real and imaginary components of $\mathbf{b}$ counted separately as opposed to the number of non-zero components of $\mathbf{b}$ as a complex vector. Counting real and imaginary components separately is a reasonable metric because the cost of implementation is usually determined by the number of operations on real numbers, even for complex-valued filters.

C. Signal detection

The design of sparse filters for signal detection can also be formulated as in (1). We assume that a signal $s[n]$ is to be detected in stationary zero-mean additive noise $\eta[n]$ with autocorrelation $\phi_{\eta\eta}[m]$. The received sequence $r[n]$ equals $s[n]+\eta[n]$ when the signal is present and $\eta[n]$ alone when the signal is absent. The sequence $r[n]$ is processed with an FIR filter of length N and sampled at $n = N-1$ to yield

$$y[N-1] = \sum_{n=0}^{N-1} b_n r[N-1-n].$$

The filter coefficients b_n are chosen to ensure that the SNR exceeds a pre-specified threshold ρ, where the SNR is defined as the ratio of the mean of $y[N-1]$ given that the signal is present to the standard deviation of $y[N-1]$, the latter contributed by noise alone. Defining $\mathbf{s} \in \mathbb{R}^N$ and $\mathbf{R} \in \mathbb{R}^{N\times N}$ according to $s_n = s[N-1-n]$ and $R_{mn} = \phi_{\eta\eta}[|m-n|]$, the problem of sparse design can be expressed as

$$\min_{\mathbf{b}} \quad \|\mathbf{b}\|_0 \qquad \text{s.t.} \qquad \frac{\mathbf{s}^T\mathbf{b}}{\sqrt{\mathbf{b}^T\mathbf{R}\mathbf{b}}} \geq \rho. \tag{11}$$

While the SNR constraint in (11) cannot be rewritten directly in the form of (3), problems (11) and (1) can be made equivalent in the sense of having the same optimal solutions. To establish the equivalence, we determine conditions under which feasible solutions to (1) and (11) exist when a subset of coefficients, represented by the index set $\mathcal{Z}$, is constrained to have zero value. Given $b_n = 0$ for $n \in \mathcal{Z}$ and with $\mathcal{Y}$ denoting the complement of $\mathcal{Z}$, (3) becomes

$$\mathbf{b}_\mathcal{Y}^T\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}\mathbf{b}_\mathcal{Y} - 2\mathbf{f}_\mathcal{Y}^T\mathbf{b}_\mathcal{Y} \leq \beta, \tag{12}$$

where $\mathbf{b}_\mathcal{Y}$ is the $|\mathcal{Y}|$-dimensional vector formed from the entries of $\mathbf{b}$ indexed by $\mathcal{Y}$ (similarly for other vectors), and $\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ is the $|\mathcal{Y}|\times|\mathcal{Y}|$ matrix formed from the rows and columns of $\mathbf{Q}$ indexed by $\mathcal{Y}$ (similarly for other matrices). We consider minimizing the left-hand side of (12) with respect to $\mathbf{b}_\mathcal{Y}$. If the minimum value is greater than β, then (12) cannot be satisfied for any value of $\mathbf{b}_\mathcal{Y}$ and a feasible solution with $b_n = 0$, $n \in \mathcal{Z}$ cannot exist. It is straightforward to show by differentiation that the minimum occurs at $\mathbf{b}_\mathcal{Y} = \left(\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}\right)^{-1}\mathbf{f}_\mathcal{Y}$, and consequently the condition for feasibility is

$$-\mathbf{f}_\mathcal{Y}^T\left(\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}\right)^{-1}\mathbf{f}_\mathcal{Y} \leq \beta. \tag{13}$$

We refer to an index set $\mathcal{Y}$ (equivalently its complement $\mathcal{Z}$) as being feasible if (13) is satisfied.

Similarly for problem (11), a subset $\mathcal{Y}$ is feasible if and only if the modified constraint

$$\frac{\mathbf{s}_\mathcal{Y}^T\mathbf{b}_\mathcal{Y}}{\sqrt{\mathbf{b}_\mathcal{Y}^T\mathbf{R}_{\mathcal{Y}\mathcal{Y}}\mathbf{b}_\mathcal{Y}}} \geq \rho$$

is satisfied when the left-hand side is maximized. The maximizing values of $\mathbf{b}_\mathcal{Y}$ are proportional to $\left(\mathbf{R}_{\mathcal{Y}\mathcal{Y}}\right)^{-1}\mathbf{s}_\mathcal{Y}$ and correspond to the whitened matched filter for the partial signal $\mathbf{s}_\mathcal{Y}$ (a.k.a. the restricted-length matched filter in [30]). The resulting feasibility condition is

$$\mathbf{s}_\mathcal{Y}^T\left(\mathbf{R}_{\mathcal{Y}\mathcal{Y}}\right)^{-1}\mathbf{s}_\mathcal{Y} \geq \rho^2 \tag{14}$$

after squaring. Condition (14) is identical to (13) for all $\mathcal{Y}$ with the identifications $\mathbf{Q} = \mathbf{R}$, $\mathbf{f} = \mathbf{s}$, and $\beta = -\rho^2$. It follows that an index set $\mathcal{Y}$ is feasible for problem (11) exactly when it is feasible for problem (1), and therefore the optimal index sets for (1) and (11) coincide.

One application of the basic detection problem above is in minimum-variance distortionless-response (MVDR) beamforming in array processing [48]. In this context, the target signal $\mathbf{s}$ is defined by a direction of interest, $\mathbf{R}$ is the correlation matrix of the array output, and the mean-squared value of the array output is minimized subject to a unit-gain constraint on signals propagating in the chosen direction. To fit the present formulation, the mean-squared output is bounded instead of minimized, which is equivalent to bounding the SNR as in (11), and the number of non-zero array weights is minimized.

In the problems discussed in this section, the assumption of stationarity is not necessary for equivalence with problem (1). In the absence of stationarity, the values of $\mathbf{Q}$, $\mathbf{f}$, and β may vary with time, resulting in a succession of instances of (1).

We have shown in this section that several sparse filter design problems can be reduced to the form of (1). Accordingly, in the remainder of the paper we focus on the solution of (1). To apply the methods to a specific design problem, it suffices to determine the values of the parameters $\mathbf{Q}$, $\mathbf{f}$, β or $\mathbf{Q}$, $\mathbf{c}$, γ using the expressions provided in this section.

III. Exact algorithms for special cases

In general, problem (1) is a difficult combinatorial optimization problem for which no polynomial-time algorithm is known. Efficient and exact solutions exist however when the matrix $\mathbf{Q}$ has special structure. In this section, we discuss several specific examples in which $\mathbf{Q}$ is diagonal, block diagonal, banded, or has low condition number.

The methods in this section solve (1) by determining for each $K = 1, 2, \ldots$ whether a feasible solution with K zero-valued coefficients exists. A condition for the feasibility of such solutions can be derived from (13), which specifies whether a solution exists when a specific subset $\mathcal{Z}$ of coefficients is constrained to have zero value. Condition (13) can be generalized to encompass all subsets of a given size using an argument similar to that made in deriving (13). Specifically, if the minimum value of the left-hand side of (13) taken over all subsets $\mathcal{Y}$ of size $N-K$ is greater than β, then no such subset $\mathcal{Y}$ is feasible and there can be no solution with K zero-valued entries. After a sign change, this gives the condition

$$\max_{|\mathcal{Y}|=N-K} \left\{ \mathbf{f}_{\mathcal{Y}}^T (\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1} \mathbf{f}_{\mathcal{Y}} \right\} \geq -\beta \tag{15}$$

for the feasibility of solutions with K zero-valued components. The number of subsets $\mathcal{Y}$ of size $N-K$ is $\binom{N}{K}$, which can be very large, and in the general case a tractable way of maximizing over all choices of $\mathcal{Y}$ is not apparent. However, for the special cases considered in this section, (15) can be evaluated efficiently.

We will find it convenient to express conditions (13) and (15) in terms of the set $\mathcal{Z}$ rather than $\mathcal{Y}$, especially when $\mathcal{Z}$ is smaller than $\mathcal{Y}$. With $b_n = 0$ for $n \in \mathcal{Z}$, the constraint in (1) becomes

$$\begin{aligned}(\mathbf{b}_{\mathcal{Y}} - \mathbf{c}_{\mathcal{Y}})^T \mathbf{Q}_{\mathcal{Y}\mathcal{Y}} (\mathbf{b}_{\mathcal{Y}} - \mathbf{c}_{\mathcal{Y}}) \\ - 2\mathbf{c}_{\mathcal{Z}}^T \mathbf{Q}_{\mathcal{Z}\mathcal{Y}} (\mathbf{b}_{\mathcal{Y}} - \mathbf{c}_{\mathcal{Y}}) + \mathbf{c}_{\mathcal{Z}}^T \mathbf{Q}_{\mathcal{Z}\mathcal{Z}} \mathbf{c}_{\mathcal{Z}} \leq \gamma,\end{aligned} \tag{16}$$

where $\mathbf{Q}_{\mathcal{Z}\mathcal{Y}}$ denotes the submatrix of $\mathbf{Q}$ with rows indexed by $\mathcal{Z}$ and columns indexed by $\mathcal{Y}$. As in the derivation of (13), we minimize the left-hand side of (16) with respect to $\mathbf{b}_{\mathcal{Y}}$ to obtain a condition for feasibility. The minimum is achieved with $\mathbf{b}_{\mathcal{Y}} - \mathbf{c}_{\mathcal{Y}} = (\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1} \mathbf{Q}_{\mathcal{Y}\mathcal{Z}} \mathbf{c}_{\mathcal{Z}}$, resulting in

$$\mathbf{c}_{\mathcal{Z}}^T (\mathbf{Q}/\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}) \mathbf{c}_{\mathcal{Z}} \leq \gamma, \tag{17}$$

where $\mathbf{Q}/\mathbf{Q}_{\mathcal{Y}\mathcal{Y}} = \mathbf{Q}_{\mathcal{Z}\mathcal{Z}} - \mathbf{Q}_{\mathcal{Z}\mathcal{Y}} (\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1} \mathbf{Q}_{\mathcal{Y}\mathcal{Z}} = \left(\left(\mathbf{Q}^{-1} \right)_{\mathcal{Z}\mathcal{Z}} \right)^{-1}$ is the Schur complement of $\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ [49]. Condition (17) is equivalent to (13). Similarly, the counterpart to (15) is

$$\min_{|\mathcal{Z}|=K} \left\{ \mathbf{c}_{\mathcal{Z}}^T (\mathbf{Q}/\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}) \mathbf{c}_{\mathcal{Z}} \right\} \leq \gamma. \tag{18}$$

A. Diagonal $\mathbf{Q}$

The first example we consider is the case of diagonal $\mathbf{Q}$. This special case arises in least-squares filter design when the weighting is uniform, i.e., $W(\omega) = 1$ in (5), implying that $\mathbf{Q} = \mathbf{I}$ from (6a). In the estimation problem, if the observations $y[n]$ are white, then $\mathbf{Q}$ in (9a) is proportional to $\mathbf{I}$. Similarly, $\mathbf{R}$ is proportional to $\mathbf{I}$ in the detection problem when the noise is white.

When $\mathbf{Q}$ is diagonal, $\mathbf{Q}/\mathbf{Q}_{\mathcal{Y}\mathcal{Y}} = \mathbf{Q}_{\mathcal{Z}\mathcal{Z}}$ and (18) simplifies to

$$\min_{|\mathcal{Z}|=K} \left\{ \sum_{n \in \mathcal{Z}} Q_{nn} c_n^2 \right\} \leq \gamma. \tag{19}$$

The solution to the minimization is to choose $\mathcal{Z}$ to correspond to the K smallest values of $Q_{nn} c_n^2$. Denoting this subset by $\mathcal{Z}_1(K)$, (19) becomes

$$\sum_{n \in \mathcal{Z}_1(K)} Q_{nn} c_n^2 \leq \gamma. \tag{20}$$

Problem (1) can now be solved by checking condition (20) for different values of K. The minimum zero-norm is given by $N - K^*$, where K^* is the largest value of K for which (20) holds. One particular optimal solution results from setting $b_n = c_n$ for n corresponding to the $N-K^*$ largest $Q_{nn} c_n^2$, and $b_n = 0$ otherwise. This solution has an intuitive interpretation in the context of the problems discussed in Section II. In least-squares filter design with $W(\omega) = 1$, we have $f_n = d[n]$ from (6b) and $c_n = f_n$. Thus the solution matches the $N - K^*$ largest values of the desired impulse response $d[n]$ and has zeros in the remaining positions. In the estimation problem with white observations, $c_n \propto f_n = \phi_{xy}[n]$, and hence the cross-correlation plays the role of the desired impulse response. Similarly, in the detection problem with white noise, the largest values of the signal $s[n]$ are matched. If $y[n]$ or $\eta[n]$ is white but non-stationary, the matrices $\mathbf{Q}$ and $\mathbf{R}$ remain diagonal and the solution takes into account any weighting due to a time-varying variance.

B. Low condition number

In Section III-A, it was seen that when $\mathbf{Q}$ is diagonal, the solution to the minimization (18) is the subset $\mathcal{Z}_1(K)$ corresponding to the K smallest values of $Q_{nn} c_n^2$. This section presents sufficient conditions related to the conditioning of $\mathbf{Q}$

for $\mathcal{Z}_1(K)$ to remain the solution to (18) when $\mathbf{Q}$ is non-diagonal.

To derive the conditions, we express $\mathbf{Q}$ as the product $\mathbf{Q} = \mathbf{DTD}$, where $\mathbf{D}$ is a diagonal matrix with non-zero diagonal entries $D_{nn} = \sqrt{Q_{nn}}$ and $\mathbf{T}$ is a positive definite matrix with unit diagonal entries. The non-singularity of $\mathbf{D}$ and positive definiteness of $\mathbf{T}$ follow from the positive definiteness of $\mathbf{Q}$. Straightforward algebra shows that the Schur complement $\mathbf{Q}/\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ is transformed into $\mathbf{D}_{\mathcal{Z}\mathcal{Z}}(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{D}_{\mathcal{Z}\mathcal{Z}}$. Thus the quadratic form in (18) can be rewritten as $\mathbf{g}_{\mathcal{Z}}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{g}_{\mathcal{Z}}$, where $\mathbf{g} = \mathbf{Dc}$ and the components of $\mathbf{g}$ satisfy $g_n^2 = Q_{nn}c_n^2$.

In terms of the rescaled parameters $\mathbf{T}$ and $\mathbf{g}$, the subset $\mathcal{Z}_1(K)$ can be interpreted as the subset that minimizes the norm $\|\mathbf{g}_{\mathcal{Z}}\|_2$ over all subsets $\mathcal{Z}$ of size K. When $\mathbf{Q}$ is diagonal, $\mathbf{T} = \mathbf{I}$ and $\mathcal{Z}_1(K)$ also minimizes the quadratic form $\mathbf{g}_{\mathcal{Z}}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{g}_{\mathcal{Z}}$. By definition, $\mathcal{Z}_1(K)$ continues to minimize $\mathbf{g}_{\mathcal{Z}}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{g}_{\mathcal{Z}}$ in the non-diagonal case if

$$\mathbf{g}_{\mathcal{Z}_1(K)}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}_1(K)\mathcal{Y}_1(K)})\mathbf{g}_{\mathcal{Z}_1(K)} \leq \mathbf{g}_{\mathcal{Z}}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{g}_{\mathcal{Z}} \quad \forall\, \mathcal{Z} : |\mathcal{Z}| = K,\ \mathcal{Z} \neq \mathcal{Z}_1(K), \quad (21)$$

where $\mathcal{Y}_1(K)$ denotes the complement of $\mathcal{Z}_1(K)$. Inequality (21) is in general difficult to verify. A stricter but more easily checked inequality can be obtained through a lower bound on the right-hand side of (21). Let $\lambda_{\min}(\mathbf{T})$ denote the smallest eigenvalue of $\mathbf{T}$. Given that $\lambda_{\min}(\mathbf{T})$ is a lower bound on the smallest eigenvalue of any Schur complement $\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}}$ [49], it follows that $\mathbf{g}_{\mathcal{Z}}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}\mathcal{Y}})\mathbf{g}_{\mathcal{Z}} \geq \lambda_{\min}(\mathbf{T})\,\|\mathbf{g}_{\mathcal{Z}}\|_2^2$ for any $\mathcal{Z}$ [29], [30]. Hence a sufficient condition for the minimality of $\mathcal{Z}_1(K)$ is

$$\mathbf{g}_{\mathcal{Z}_1(K)}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}_1(K)\mathcal{Y}_1(K)})\mathbf{g}_{\mathcal{Z}_1(K)} \leq \lambda_{\min}(\mathbf{T})\,\|\mathbf{g}_{\mathcal{Z}}\|_2^2 \quad \forall\, \mathcal{Z} : |\mathcal{Z}| = K,\ \mathcal{Z} \neq \mathcal{Z}_1(K). \quad (22)$$

Inequality (22) depends on $\mathcal{Z}$ only through the norm $\|\mathbf{g}_{\mathcal{Z}}\|_2^2$ and can therefore be reduced to

$$\mathbf{g}_{\mathcal{Z}_1(K)}^T(\mathbf{T}/\mathbf{T}_{\mathcal{Y}_1(K)\mathcal{Y}_1(K)})\mathbf{g}_{\mathcal{Z}_1(K)} \leq \lambda_{\min}(\mathbf{T})\,\left\|\mathbf{g}_{\mathcal{Z}_2(K)}\right\|_2^2, \quad (23)$$

where $\mathcal{Z}_2(K)$ is the subset corresponding to the second-smallest value of $\|\mathbf{g}_{\mathcal{Z}}\|_2$. Evaluating (23) requires only that the components of $\mathbf{g}$ be sorted by magnitude instead of a full combinatorial search.

The sufficient condition (23) can be related to the condition number of $\mathbf{T}$ by bounding the left-hand side of (23) from above in terms of the largest eigenvalue $\lambda_{\max}(\mathbf{T})$ (and thus further strengthening the inequality). Using the definition of the condition number $\kappa(\mathbf{T}) = \lambda_{\max}(\mathbf{T})/\lambda_{\min}(\mathbf{T})$, we obtain

$$\kappa(\mathbf{T}) \leq \frac{\left\|\mathbf{g}_{\mathcal{Z}_2(K)}\right\|_2^2}{\left\|\mathbf{g}_{\mathcal{Z}_1(K)}\right\|_2^2},$$

which suggests that (23) is more likely to be satisfied when $\kappa(\mathbf{T})$ is low and the ratio of the norms is large, i.e., when $\mathbf{g}$ has K components that are much smaller than the rest. On the other hand, if all components of $\mathbf{g}$ are of the same magnitude, the condition cannot be satisfied unless $\kappa(\mathbf{T}) = 1$ implying $\mathbf{T} \propto \mathbf{I}$.

In summary, it is optimal to set $b_n = 0$ for indices n corresponding to the smallest $Q_{nn}c_n^2$, just as in the diagonal case, provided that inequality (23) is satisfied. The arguments in this section remain valid for any choice of non-singular diagonal matrix $\mathbf{D}$, with corresponding changes in the definitions of the minimizing subsets $\mathcal{Z}_1(K)$ and $\mathcal{Z}_2(K)$ and the matrix $\mathbf{T}$.

C. Block-diagonal $\mathbf{Q}$

A generalization of the diagonal structure in Section III-A is the case of block-diagonal matrices. It was seen in Section II that $\mathbf{Q}$ often corresponds to a covariance matrix and is therefore block-diagonal if the underlying random process can be partitioned into subsets of variables such that variables from different subsets are uncorrelated. This may occur for example in a sensor array in which the sensors are arranged in clusters separated by large distances.

We assume that $\mathbf{Q}$ is block-diagonal with B diagonal blocks:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 & & \\ & \ddots & \\ & & \mathbf{Q}_B \end{bmatrix},$$

where the bth block $\mathbf{Q}_b$ is of dimension $N_b \times N_b$ and indices have been permuted if necessary to convert $\mathbf{Q}$ to block-diagonal form. For every index set $\mathcal{Y}$, let $\mathcal{Y}_b$ be the intersection of $\mathcal{Y}$ with the indices corresponding to the bth block. Then

$$\mathbf{Q}_{\mathcal{Y}\mathcal{Y}} = \begin{bmatrix} \mathbf{Q}_{\mathcal{Y}_1\mathcal{Y}_1} & & \\ & \ddots & \\ & & \mathbf{Q}_{\mathcal{Y}_B\mathcal{Y}_B} \end{bmatrix}$$

is also block diagonal. Hence the maximization in (15) can be rewritten as

$$\max \quad \sum_{b=1}^{B} \mathbf{f}_{\mathcal{Y}_b}^T(\mathbf{Q}_{\mathcal{Y}_b\mathcal{Y}_b})^{-1}\mathbf{f}_{\mathcal{Y}_b} \qquad \text{s.t.} \qquad \sum_{b=1}^{B} |\mathcal{Y}_b| = N - K. \quad (24)$$

A similar and equivalent decomposition could be obtained from (18) since $\mathbf{Q}^{-1}$ is also block diagonal.

The maximization in (24) can be solved via dynamic programming. To derive the dynamic programming recursion, define $V_g(M)$ to be the maximum value over all subsets $\mathcal{Y}$ of size M that are confined to the first g blocks, i.e.,

$$V_g(M) = \max \quad \sum_{b=1}^{g} \mathbf{f}_{\mathcal{Y}_b}^T(\mathbf{Q}_{\mathcal{Y}_b\mathcal{Y}_b})^{-1}\mathbf{f}_{\mathcal{Y}_b} \quad \text{s.t.} \quad \sum_{b=1}^{g} |\mathcal{Y}_b| = M, \qquad g = 1, \ldots, B.$$

The maximum value in (24) is thus $V_B(N-K)$. Also define $v_b(M_b)$ to be the maximum value over subsets of size M_b restricted to the bth block,

$$v_b(M_b) = \max_{|\mathcal{Y}_b| = M_b} \mathbf{f}_{\mathcal{Y}_b}^T(\mathbf{Q}_{\mathcal{Y}_b\mathcal{Y}_b})^{-1}\mathbf{f}_{\mathcal{Y}_b}, \quad b = 1, \ldots, B, \quad M_b = 0, 1, \ldots, N_b. \quad (25)$$

It follows that $V_1(M) = v_1(M)$. For $g = 2, \ldots, B$, $V_g(M)$ may be computed through the following recursion:

$$V_g(M) = \max_{M_g = 0, 1, \ldots, \min(M, N_g)} \left\{ v_g(M_g) + V_{g-1}(M - M_g) \right\}, \quad (26)$$

which corresponds to optimally allocating M_g indices to the gth block, optimally allocating the remaining $M - M_g$ indices to the first $g-1$ blocks, and then maximizing over all choices of M_g between 0 and the lesser of M and N_g, the dimension of the gth block.

Dynamic programming decomposes the maximization in (15) into B smaller problems (25) of dimension N_b, together with a recursion (26) to compute the overall maximum. This results in a significant decrease in computation since the complexity of exhaustively evaluating (15) for a number of K values proportional to N is at least exponential in N, whereas the complexity of (25) is only exponential in N_b. The decomposition is particularly efficient when the blocks are small in an absolute sense. The computational complexity added by the recursion is comparatively modest. Each evaluation of (26) requires at most $N_g + 1$ additions and comparisons. Assuming in the worst case that $V_g(M)$ is computed for all $g = 2, \ldots, B$ and $M = 0, \ldots, N$, the total number of operations in the recursion is

$$\sum_{g=2}^{B} \sum_{M=0}^{N} (N_g + 1) = (N+1)(N - N_1 + B - 1) \sim \mathcal{O}(N^2).$$

D. Banded $\mathbf{Q}$

Another generalization of the diagonal case is to consider banded matrices, i.e., matrices in which the non-zero entries in row n are confined to columns $n - w$ to $n + w$. Banded structure implies that the submatrices $\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ are block-diagonal for certain subsets $\mathcal{Y}$. As in Section III-C, an exhaustive search for the best subset can be simplified with dynamic programming. In this paper, we focus on tridiagonal matrices ($w = 1$). Detailed analysis of higher-bandwidth matrices is presented in [29], [50].

For the tridiagonal case, consider expressing a subset $\mathcal{Y}$ as a union of subsets $\mathcal{Y}_1, \ldots, \mathcal{Y}_C$ such that all indices in each subset $\mathcal{Y}_c$ are consecutive and each subset is separated from all others by at least one index. In this case, $\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ is block-diagonal and the quadratic form in (15) can be decomposed as

$$\mathbf{f}_{\mathcal{Y}}^T (\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1} \mathbf{f}_{\mathcal{Y}} = \sum_{c=1}^{C} \mathbf{f}_{\mathcal{Y}_c}^T (\mathbf{Q}_{\mathcal{Y}_c\mathcal{Y}_c})^{-1} \mathbf{f}_{\mathcal{Y}_c}. \tag{27}$$

The dynamic programming recursion based on (27) is slightly different than the recursion for block-diagonal matrices in Section III-C. The elementary quantities are quadratic forms for sets of consecutive indices, expressed as

$$w_i(p) = \mathbf{f}_{\mathcal{Y}_c}^T (\mathbf{Q}_{\mathcal{Y}_c\mathcal{Y}_c})^{-1} \mathbf{f}_{\mathcal{Y}_c} \quad \text{where} \quad \mathcal{Y}_c = \{i - p + 1, \ldots, i\}.$$

In addition, the quadratic form for an empty index set is defined as $w_i(0) = 0$. The state variables for the dynamic program are the best subsets of given size and upper bound on the indices in the subset, and the associated quadratic forms. These are defined as

$$\begin{aligned} W_i(M) = \max \quad & \mathbf{f}_{\mathcal{Y}}^T (\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1} \mathbf{f}_{\mathcal{Y}} \\ \text{s.t.} \quad & |\mathcal{Y}| = M \qquad \text{and} \qquad j \le i \quad \forall\, j \in \mathcal{Y}. \end{aligned}$$

These definitions imply that $W_i(i) = w_i(i)$ since the only set of i indices with maximum index i is $\mathcal{Y} = \{1, \ldots, i\}$.

The dynamic program proceeds in stages defined by the maximum index i with i increasing from 1 to N. At the end of the computation, the left-hand side in the feasibility test (15) is given by $W_N(N - K)$. The states $W_i(i)$ are already given by $w_i(i)$. The states $W_i(1)$ to $W_i(i-1)$ are computed from $W_j(M)$ for $j < i$ using the following recursion:

$$W_i(M) = \max_{p=0,\ldots,M} \{w_i(p) + W_{i-p-1}(M - p)\}. \tag{28}$$

The first term on the right-hand side corresponds to fixing a final subset $\{i - p + 1, \ldots, i\}$ of p consecutive indices. Given this final run, the remaining $M - p$ indices are restricted to the range $1, \ldots, i - p - 1$, and the optimal choice of these $M - p$ indices yields the second term $W_{i-p-1}(M - p)$. Since the index $i - p$ is not included, the two terms simply add as in (27). The sum is then maximized over all choices of p. For $p = 0$, the right-hand side of (28) reduces to $W_{i-1}(M)$, i.e., the last index is not used.

The computational complexity of the algorithm is controlled by the number of elementary subsets. In the tridiagonal case, these subsets are composed of consecutive indices. There are $O(N^2)$ subsets of this type. For each such subset, the dynamic programming algorithm computes the associated quadratic form $w_i(p)$, requiring $O(N^3)$ operations in the worst case. The cost of computing all of the $w_i(p)$ values exceeds the cost of updating the $W_i(M)$ variables [29], and hence the total computational complexity of the dynamic program is $O(N^5)$ for tridiagonal matrices. As noted in Section III-C, the complexity of an exhaustive search algorithm is at least exponential in N.

For general banded matrices, the elementary subsets are composed of indices separated by fewer than w places. For each subset, the associated quadratic form is computed and the state variables for the dynamic program are updated. As shown in [50, App. A.1], the number of elementary subsets is proportional to 2^{M_0}, where M_0 is the largest value of $N - K$. If M_0 is proportional to N, the dynamic programming algorithm considers an exponential number of subsets, just as an exhaustive search does.

A variation on the banded case is that of $\mathbf{Q}^{-1}$ being banded. Unlike in the diagonal or block-diagonal cases, $\mathbf{Q}$ having a low bandwidth does not imply that $\mathbf{Q}^{-1}$ has the same bandwidth, and vice versa. If $\mathbf{Q}^{-1}$ is tridiagonal, we may work with condition (18) instead of (15). The above algorithm then applies with $\mathbf{Q}$ replaced by $\mathbf{Q}^{-1}$, $\mathbf{f}$ by $\mathbf{c}$, and maximization by minimization. The number of zero coefficients K is incremented instead of the number of non-zero coefficients M.

E. Challenges in generalizing to unstructured $\mathbf{Q}$

In Sections III-A–III-D, we discussed several special cases of problem (1) in which the structure of the matrix $\mathbf{Q}$ allows for an efficient solution. It is natural to ask whether these special cases can be generalized. In particular, the fact that any symmetric matrix can be diagonalized by a unitary transformation suggests the possibility of exploiting such transformations to reduce the general problem to the diagonal case of Section III-A. Unfortunately, this approach to generalization does not appear to be straightforward.

One way of reducing an instance to the diagonal case is to apply whitening. In the estimation problem, the whitening is done on the observations $y[n]$, while in the detection problem, it is the noise $\eta[n]$ that is whitened. The process of whitening however requires additional processing of the input, for example with a prefilter. The task then shifts to designing a whitening prefilter that does not significantly increase the total implementation cost. Moreover, since the whitening is likely to be imperfect, further measures may be needed. There are also applications in which cascade structures are not applicable, e.g. arrays.

A different approach is to solve problem (1) by first transforming the feasible set into one that is easier to optimize over, for example a set corresponding to a diagonal matrix, and then inverting the transformation to map the solution found in the transformed space to one in the original space. It is necessary for the transformation to preserve the ordering of vectors by their zero-norm to ensure optimality in the original space. As shown in [50, App. A.2], the only invertible linear transformations that preserve ordering by zero-norm in a certain global sense are composed of invertible diagonal scalings and permutations. These operations cannot transform a dense $\mathbf{Q}$ matrix into a diagonal, block-diagonal, or banded matrix. It appears therefore that (1) in its general form is not reducible to one of the special cases and hence remains a difficult problem. Nevertheless, the special case solutions in Sections III-A–III-D can provide the basis for approximations, for example in [45] to derive bounds on the optimal cost.

IV. Low-complexity algorithm for the general case

In this section we consider the more general case in which the matrix $\mathbf{Q}$ does not have any of the properties identified in Section III. In keeping with the focus in this paper on low-complexity algorithms, we discuss a heuristic algorithm for solving (1) based on backward greedy selection. We give an overview of the algorithm before describing an efficient implementation in detail. Optimal algorithms for the general case are treated in a companion paper [45].

The backward selection algorithm iteratively thins a predesigned non-sparse filter by constraining more and more coefficients to zero while re-optimizing the remaining non-zero coefficients to compensate. Each new zero-valued coefficient is chosen to minimize the increase in the quadratic error (the left-hand side of the constraint in (1)), and zero-value constraints once added are never removed. The algorithm can be viewed as a simplification of the exact method described at the beginning of Section III, which involves evaluating (18) for $K = 1, 2, \ldots$. For $K = 1$, the algorithm carries out the minimization in (18) exactly, yielding a minimizing subset (in this case a single index) that we denote as $\mathcal{Z}^{(1)}$. For $K > 1$, the subsets $\mathcal{Z}$ considered in the minimization are constrained to contain $\mathcal{Z}^{(K-1)}$, the minimizer for the previous value of K, thus limiting the search to adding a single index to $\mathcal{Z}^{(K-1)}$. The algorithm terminates when the minimum value corresponding to $\mathcal{Z}^{(K+1)}$ exceeds γ for some K, at which point the last feasible subset $\mathcal{Z}^{(K)}$ is taken to be the final subset of zero-valued coefficients. Since the number of subsets explored in the Kth iteration is at most $N - K + 1$ (corresponding to the $N - (K - 1)$ choices for the index to be added to $\mathcal{Z}^{(K-1)}$), and the number of iterations is at most N, the total number of subsets grows only quadratically with N. In comparison, an exhaustive evaluation of (18) for $K = 1, 2, \ldots$ would involve an exponential number of subsets in total.

Greedy selection is guaranteed to result in a maximally sparse solution when the matrix $\mathbf{Q}$ is diagonal. From Section III-A, the solution to the minimization in (18) in the diagonal case is to choose $\mathcal{Z}$ to correspond to the K smallest $Q_{nn}c_n^2$. Since the subset of the K smallest $Q_{nn}c_n^2$ is contained in the subset of the $K + 1$ smallest, the nesting property assumed by the algorithm is satisfied and the algorithm finds the true minimizing subsets. In other cases however, backward greedy selection does not appear to guarantee an optimal solution. Nevertheless, the examples in Section V demonstrate that the algorithm often yields optimal or near-optimal designs.

To describe the algorithm in more detail, we use $\mathcal{Z}^{(K)}$ as above to represent the subset of coefficients that are constrained to zero in iteration K. The complement of $\mathcal{Z}^{(K)}$, previously denoted $\mathcal{Y}^{(K)}$, is now partitioned into two subsets $\mathcal{U}^{(K)}$ and $\mathcal{F}^{(K)}$. The subset $\mathcal{U}^{(K)}$ consists of those coefficients for which a zero value is no longer feasible because of zero-value constraints imposed on coefficients in $\mathcal{Z}^{(K)}$, which shrink the feasible set. It will be seen shortly that the coefficients in $\mathcal{U}^{(K)}$ can be eliminated to reduce the dimension of the problem. The subset $\mathcal{F}^{(K)}$ consists of the remaining coefficients.

Each iteration of the algorithm is characterized by the partitioning of the variables into the subsets $\mathcal{Z}^{(K)}$, $\mathcal{U}^{(K)}$, and $\mathcal{F}^{(K)}$. We assume for simplicity that no coefficients are constrained to zero in the beginning, i.e., $\mathcal{Z}^{(0)} = \mathcal{U}^{(0)} = \emptyset$ and $\mathcal{F}^{(0)} = \{1, \ldots, N\}$; other initializations are possible. In subsequent iterations, both $\mathcal{Z}^{(K)}$ and $\mathcal{U}^{(K)}$ grow while $\mathcal{F}^{(K)}$ shrinks, giving rise to increasingly constrained versions of the original problem that we refer to as subproblems. Each subproblem can be reduced to a lower-dimensional instance of the original problem (1), a fact that simplifies the algorithm. Specifically, it is shown in the Appendix that a subproblem defined by $(\mathcal{Z}, \mathcal{U}, \mathcal{F})$ can be reformulated as a minimization over $\mathbf{b}_\mathcal{F}$ only:

$$\begin{aligned} \min_{\mathbf{b}_\mathcal{F}} \quad & |\mathcal{U}| + \|\mathbf{b}_\mathcal{F}\|_0 \\ \text{s.t.} \quad & \left(\mathbf{b}_\mathcal{F} - \mathbf{c}_{\text{eff}}\right)^T \mathbf{Q}_{\text{eff}} \left(\mathbf{b}_\mathcal{F} - \mathbf{c}_{\text{eff}}\right) \le \gamma_{\text{eff}}, \end{aligned} \tag{29}$$

where

$$\mathbf{Q}_{\text{eff}} = \mathbf{Q}_{\mathcal{F}\mathcal{F}} - \mathbf{Q}_{\mathcal{F}\mathcal{U}} \left(\mathbf{Q}_{\mathcal{U}\mathcal{U}}\right)^{-1} \mathbf{Q}_{\mathcal{U}\mathcal{F}}, \tag{30a}$$

$$\mathbf{c}_{\text{eff}} = \mathbf{c}_\mathcal{F} + (\mathbf{Q}_{\text{eff}})^{-1} \big(\mathbf{Q}_{\mathcal{F}\mathcal{Z}} - \mathbf{Q}_{\mathcal{F}\mathcal{U}} (\mathbf{Q}_{\mathcal{U}\mathcal{U}})^{-1} \mathbf{Q}_{\mathcal{U}\mathcal{Z}}\big) \mathbf{c}_\mathcal{Z}, \tag{30b}$$

$$\gamma_{\text{eff}} = \gamma - \mathbf{c}_\mathcal{Z}^T \left(\left(\mathbf{Q}^{-1}\right)_{\mathcal{Z}\mathcal{Z}}\right)^{-1} \mathbf{c}_\mathcal{Z}. \tag{30c}$$

The variables b_n for $n \in \mathcal{Z}$ are absent from (29) because they have been set to zero. The variables b_n, $n \in \mathcal{U}$ have also been eliminated, with the term $|\mathcal{U}|$ accounting for their contribution to the zero-norm. This reduction allows each iteration of the algorithm after the first to be treated as if it were the first iteration applied to a lower-dimensional instance of (1).

In the sequel, we use a superscript K to label the parameters of the subproblem in iteration K, namely $\mathbf{Q}^{(K)}$, $\mathbf{c}^{(K)}$, and $\gamma^{(K)}$. We also define $\mathbf{P}^{(K)} = \left(\mathbf{Q}^{(K)}\right)^{-1}$ and will find it more convenient to specify the computations in terms of $\mathbf{P}$ rather than $\mathbf{Q}$. Assuming that the algorithm starts with no zero-value constraints, $\mathbf{P}^{(0)} = \mathbf{Q}^{-1}$, $\mathbf{c}^{(0)} = \mathbf{c}$, and $\gamma^{(0)} = \gamma$.

The first task in each iteration is to update the subset $\mathcal{U}^{(K)}$ by adding to it any coefficients in $\mathcal{F}^{(K)}$ that no longer yield feasible solutions when constrained to a value of zero. Such coefficients can be identified by specializing condition (17) to subsets consisting of a single index, $\mathcal{Z} = \{n\}$. Condition (17) simplifies to

$$\frac{\left(c_n^{(K)}\right)^2}{P_{nn}^{(K)}} \leq \gamma^{(K)}, \tag{31}$$

where we have substituted the parameters for the Kth (i.e., current) subproblem. The indices n for which (31) is not satisfied correspond to coefficients for which a zero value is infeasible. Hence these indices are removed from $\mathcal{F}^{(K)}$ and added to $\mathcal{U}^{(K)}$, i.e.,

$$\mathcal{U}^{(K+1)} = \mathcal{U}^{(K)} \cup \left\{ n \in \mathcal{F}^{(K)} : \frac{\left(c_n^{(K)}\right)^2}{P_{nn}^{(K)}} > \gamma^{(K)} \right\}. \tag{32}$$

If no indices remain in $\mathcal{F}^{(K)}$ after this removal, the filter cannot be thinned further and the algorithm terminates. Otherwise, an index m is removed from $\mathcal{F}^{(K)}$ and added to $\mathcal{Z}^{(K)}$ to form $\mathcal{Z}^{(K+1)}$. As discussed earlier, m is chosen to minimize the left-hand side of (18) over $\mathcal{Z}^{(K+1)}$ of the form $\mathcal{Z}^{(K+1)} = \mathcal{Z}^{(K)} \cup \{m\}$. This is equivalent to choosing

$$m = \arg \min_{n \in \mathcal{F}^{(K)}} \frac{\left(c_n^{(K)}\right)^2}{P_{nn}^{(K)}}.$$

The indices remaining in $\mathcal{F}^{(K)}$ after removing m form the new subset $\mathcal{F}^{(K+1)}$.

To calculate the values of the new parameters $\mathbf{P}^{(K+1)}$, $\mathbf{c}^{(K+1)}$, and $\gamma^{(K+1)}$ from the current parameters $\mathbf{P}^{(K)}$, $\mathbf{c}^{(K)}$, and $\gamma^{(K)}$, we make use of (30) with the current parameters playing the role of $\mathbf{Q}$, $\mathbf{c}$, and γ, $\mathcal{Z} = \{m\}$ to represent the new zero-value constraint, $\mathcal{U}$ composed of the indices added to $\mathcal{U}^{(K)}$ in (32), and $\mathcal{F} = \mathcal{F}^{(K+1)}$. With these replacements, (30a) gives $\mathbf{Q}^{(K+1)}$ in terms of $\mathbf{Q}^{(K)}$. It can be shown that the equivalent recursion for $\mathbf{P}$ is

$$\mathbf{P}^{(K+1)} = \mathbf{P}^{(K)}_{\mathcal{F}^{(K+1)}\mathcal{F}^{(K+1)}} - \frac{1}{P_{mm}^{(K)}} \mathbf{P}^{(K)}_{\mathcal{F}^{(K+1)},m} \mathbf{P}^{(K)}_{m,\mathcal{F}^{(K+1)}}. \tag{33}$$

Similarly, (30b) can be rewritten in terms of $\mathbf{P}$ instead of $\mathbf{Q}$ to yield

$$\mathbf{c}^{(K+1)} = \mathbf{c}^{(K)}_{\mathcal{F}^{(K+1)}} - \frac{c_m^{(K)}}{P_{mm}^{(K)}} \mathbf{P}^{(K)}_{\mathcal{F}^{(K+1)},m}. \tag{34}$$

Neither (33) nor (34) require matrix inversion. Lastly, (30c) gives the following recursion for γ:

$$\gamma^{(K+1)} = \gamma^{(K)} - \frac{\left(c_m^{(K)}\right)^2}{P_{mm}^{(K)}}. \tag{35}$$

This completes the operations in iteration K.

Once the algorithm has terminated with a final subset $\mathcal{Z}^{(f)}$ of zero-valued coefficients, it remains to determine the values of the non-zero coefficients $\mathbf{b}_{\mathcal{Y}^{(f)}}$. We choose $\mathbf{b}_{\mathcal{Y}^{(f)}}$ specifically to maximize the margin in the quadratic constraint subject to $b_n = 0$ for $n \in \mathcal{Z}^{(f)}$, i.e., to minimize the left-hand side of (16). The solution as given in Section III is

$$\mathbf{b}_{\mathcal{Y}^{(f)}} = \mathbf{c}_{\mathcal{Y}^{(f)}} + \left(\mathbf{Q}_{\mathcal{Y}^{(f)}\mathcal{Y}^{(f)}}\right)^{-1} \mathbf{Q}_{\mathcal{Y}^{(f)}\mathcal{Z}^{(f)}} \mathbf{c}_{\mathcal{Z}^{(f)}}.$$

The most computationally intensive step in the iterative part of the algorithm is the update of $\mathbf{P}$ in (33), an $O(N^2)$ operation. The total complexity is $O(N^3)$ since the number of iterations is linear in N and the initialization of $\mathbf{P}^{(0)}$ and computation of the final solution are both $O(N^3)$.

V. Design examples

In this section, two filter design examples are presented, the first demonstrating the design of sparse equalizers for a wireless communication channel, and the second the design of non-uniformly spaced beamformers for detection. The backward selection algorithm of Section IV is compared to three algorithms: the forward selection algorithm used (with some variations) in [25], [26], a heuristic based on ordering the coefficients of the optimal non-sparse solution $\mathbf{c}$ (similar in spirit to [20], [21]), and an exact branch-and-bound algorithm [45]. The comparisons reveal that backward selection consistently outperforms the other two heuristics and produces optimally sparse or near-optimal designs in many instances. The examples also highlight the potential gains of sparse design in these applications.

A. Wireless channel equalization

In the first example, sparse equalizers are designed for a test channel used to evaluate terrestrial broadcast systems for high-definition television. This example was also considered in [21], [22]. To facilitate the comparison with the branch-and-bound algorithm, the channel is simplified by reducing the multipath delays by half and converting complex amplitudes to real amplitudes with the same magnitude. The modified multipath parameters are shown in Table I. We emphasize that this simplification would be unnecessary for comparing the heuristic algorithms alone. The effective discrete-time channel response is given by

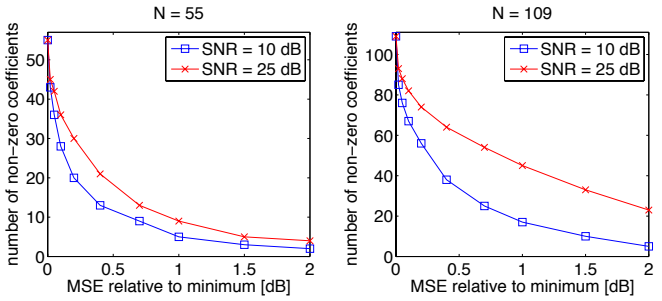
$$h[n] = \sum_{i=0}^{5} a_i p(n - \tau_i),$$

where the sampling period is normalized to unity and the pulse $p(t)$ is the convolution of the transmit and receive filter responses, each chosen to correspond to a square-root raised-cosine filter with excess bandwidth parameter $\beta = 0.115$ following [21], [22]. The transmitted sequence $x[n]$ and noise $\eta[n]$ are assumed to be white with $\phi_{xx}[m] = \sigma_x^2 \delta[m]$ and $\phi_{\eta\eta}[m] = \delta[m]$ so that the input SNR is σ_x^2. The translation of channel parameters into problem parameters $\mathbf{Q}$, $\mathbf{c}$, and γ is as described in Section II-B, specifically in (10) and (9) together with the relations $\mathbf{c} = \mathbf{Q}^{-1}\mathbf{f}$ and $\gamma = \beta + \mathbf{c}^T\mathbf{Q}\mathbf{c}$.

In our simulations, the equalizer length N is varied between $L + 1$ and $2L + 1$, where $L = 54$ is the largest delay in the

TABLE I
NOMINAL MULTIPATH PARAMETERS FOR THE EQUALIZATION EXAMPLE.

i	0	1	2	3	4	5
τ_i	0	4.84	5.25	9.68	20.18	53.26
a_i	0.5012	-1	0.1	0.1259	-0.1995	-0.3162



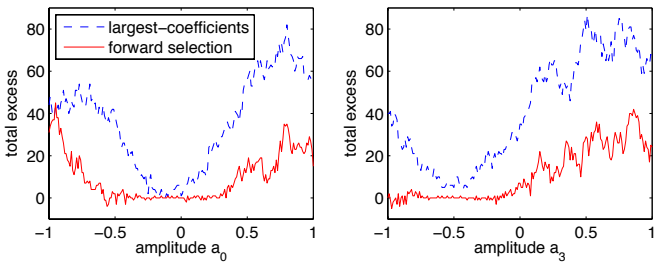
(a) (b)

Fig. 1. Number of non-zero equalizer coefficients resulting from the backward selection algorithm as a function of the MSE ratio $\delta/\delta_{\min}$ for equalizer lengths (a) $N = 55$ and (b) $N = 109$.

channel (rounded up). The allowable MSE δ is varied between the minimum MSE $\delta_{\min} = \sigma_x^2 - \mathbf{c}^T\mathbf{Q}\mathbf{c}$ and 2 dB above $\delta_{\min}$. We also introduce a delay Δ into the estimate, i.e., $x[n]$ in (8) is changed to $x[n-\Delta]$, to accommodate the causality of the channel and the equalizer. Equations (9b) and (10b) are modified accordingly to yield $f_n = \phi_{xy}[n-\Delta] = \sigma_x^2 h[\Delta - n]$. The MSE performance depends weakly on Δ; for these simulations a value of $\Delta = 0.8L + 0.2N$ is a reasonably good choice.

In Fig. 1, we plot the number of non-zero equalizer coefficients given by the backward selection algorithm as a function of the MSE ratio $\delta/\delta_{\min}$ for equalizer lengths $N = 55$ and 109 and input SNR $\sigma_x^2 = 10, 25$ dB. The SNR required for digital television reception can reach 25 dB for severe multipath channels [51]. The MMSE equalizers achieve MSE values (normalized by σ_x^2) of -5.74 and -7.30 dB for $N = 55$ and $\sigma_x^2 = 10, 25$ dB, and -6.80 and -9.76 dB for $N = 109$ and the same SNR values. The number of non-zero coefficients decreases steeply as the MSE increases from its minimum, e.g. for $N = 55$ and $\sigma_x^2 = 10$ dB the number is nearly halved with only a 0.1 dB increase in MSE. Implementation losses of a few tenths of a dB are usually acceptable for wireless receivers [52], [53]. Less sparsity is seen at the higher SNR value. This behavior is consistent with previous findings (e.g. in [26]).

Table II compares the backward selection algorithm against the other algorithms, both for the nominal channel in Table I with 10 dB SNR and a modified channel with a_0 changed to -0.95. Backward selection consistently outperforms the largest-coefficients algorithm, which chooses the support to correspond to the M largest coefficients of the optimal non-sparse equalizer with M decreasing until the MSE constraint can no longer be satisfied. The forward selection algorithm starts with the all-zero equalizer and iteratively adds in a greedy fashion the coefficient resulting in the greatest MSE reduction until a feasible solution is obtained. Backward selection performs at least as well or better than forward selection in all but two instances, and the differences can be significant for the modified channel at longer lengths. Backward selection also matches the cost achieved by branch-and-bound in a large majority of instances, with the difference never exceeding 2. Five of the $N = 109$ instances are very difficult to solve to optimality and the branch-and-bound algorithm did not converge within the allotted solution time (10^5 sec), yielding instead an interval containing the true optimal cost. The upper end of the interval represents the sparsest solution found by branch-and-bound, which in 4 out of the 5 instances does not improve upon the backward greedy solution. Our results suggest therefore that backward selection can often produce optimal designs with much lower complexity than branch-and-bound.



(a) (b)

Fig. 2. Total excess with respect to the backward selection algorithm as a function of channel amplitudes a_0 (a) and a_3 (b).

Fig. 2 shows a further comparison of the heuristic algorithms in which one of the channel amplitudes is varied while all other parameters are fixed at their nominal values in Table I. The total excess is a summary statistic and refers to the sum of the differences in non-zero coefficients between either the largest-coefficients or forward selection algorithms and the backward selection algorithm, where the sum is taken over all N and $\delta/\delta_{\min}$ in Table II. Fig. 2 shows that backward selection consistently performs at least as well or better than the other heuristics. Plots for variations in other amplitudes are similar. This suggests that backward selection is a more robust choice when there is uncertainty or variation in the channel, as is often the case in wireless communication.

Table III shows average execution times and numbers of iterations for MATLAB implementations of the heuristic algorithms running on a 2.4 GHz Linux computer with 4 GB of memory. The averages are taken over all MSE ratios and channel amplitudes. The largest-coefficients and forward selection algorithms are implemented using rank-1 updates as in (33)–(35), and hence all of the heuristics are very efficient. In this particular equalization example, backward selection requires more iterations and correspondingly longer times because of the relatively high sparsity levels.

B. MVDR beamforming

The second example concerns the design of non-uniformly spaced MVDR beamformers, an application of the detection problem in Section II-C. The beamformers are non-uniform in the sense that the element positions are constrained to an

TABLE II
NUMBERS OF NON-ZERO COEFFICIENTS RETURNED BY THE LARGEST-COEFFICIENT (LC), FORWARD SELECTION (FS), BACKWARD SELECTION (BS), AND BRANCH-AND-BOUND (BB) ALGORITHMS IN THE EQUALIZATION EXAMPLE.

		nominal channel (Table I)				modified ($a_0 = -0.95$)			
N	$\delta/\delta_{\min}$ [dB]	LC	FS	BS	BB	LC	FS	BS	BB
55	0.02	44	45	43	43	40	38	38	38
	0.05	38	37	36	36	34	34	34	34
	0.1	30	28	28	28	30	29	30	29
	0.2	22	20	20	20	26	22	22	22
	0.4	15	14	13	13	16	18	16	16
	0.7	10	8	9	8	14	11	11	11
	1.0	5	5	5	5	8	7	7	7
	1.5	3	3	3	3	3	3	3	3
	2.0	2	2	2	2	2	2	2	2
82	0.02	65	64	63	63	64	64	62	62
	0.05	57	56	55	55	59	64	57	57
	0.1	48	48	47	47	54	58	51	51
	0.2	36	34	34	34	46	47	44	42
	0.4	25	22	22	22	33	31	30	29
	0.7	17	14	14	14	21	22	20	20
	1.0	14	11	10	10	18	16	16	15
	1.5	7	5	5	5	9	9	9	9
	2.0	4	3	3	3	6	6	6	6
109	0.02	87	86	85	85	85	85	83	83
	0.05	78	78	76	76	76	77	75	74
	0.1	69	70	67	[64, 67]	69	75	68	68
	0.2	58	56	56	[50, 56]	61	67	59	[53, 58]
	0.4	46	38	38	[35, 38]	49	45	43	[37, 43]
	0.7	29	26	25	25	29	28	27	27
	1.0	20	18	17	17	22	20	20	20
	1.5	14	10	10	10	15	13	13	13
	2.0	6	5	5	5	8	7	7	7

TABLE III
AVERAGE EXECUTION TIMES AND NUMBERS OF ITERATIONS FOR THE LARGEST-COEFFICIENT (LC), FORWARD SELECTION (FS), AND BACKWARD SELECTION (BS) ALGORITHMS IN THE EQUALIZATION EXAMPLE.

N	time [ms]			iterations		
	LC	FS	BS	LC	FS	BS
55	1.1	1.4	3.5	15.7	15.1	38.9
82	2.5	5.1	8.7	26.3	25.3	55.9
109	9.4	8.6	14.5	39.1	37.5	70.8

underlying uniform grid but only a subset of the positions are used.

As in Section V-A, the backward selection algorithm is compared to other heuristics and an exact branch-and-bound algorithm. To apply the branch-and-bound algorithm in particular, we focus on a real-valued formulation of the beamforming problem as opposed to the more conventional complex-valued formulation. Although the reduction in Section II-C of the sparse detection problem to (1) can be generalized to the complex case with minor modifications, some parts of the branch-and-bound algorithm assume real values and their complex-valued generalization is a subject for future study. We assume that a signal at an angle θ_0 from the array axis is to be detected in the presence of discrete interferers at θ_1 and θ_2 and isotropic (white) noise $\boldsymbol{\eta}$. The received signal at the nth array element is

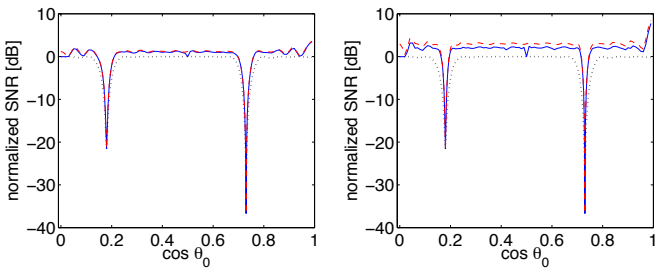
$$y_n = \cos(n\pi\cos\theta_0) + \sum_{i=1}^{2} A_i \cos(n\pi\cos\theta_i) + \eta_n,$$

$$n = \pm\frac{1}{2}, \pm\frac{3}{2}, \ldots, \pm\frac{N-1}{2},$$

assuming a half-wavelength spacing between elements. The interferer amplitudes A_1 and A_2 are modelled as zero-mean random variables with variances σ_1^2 and σ_2^2. We use the nominal values $\cos\theta_1 = 0.18$, $\cos\theta_2 = 0.73$, $\sigma_1^2 = 10$ dB, $\sigma_2^2 = 25$ dB, the last two values being relative to the white noise power σ_η^2. The target angle is swept from $\cos\theta_0 = 0$ to $\cos\theta_0 = 1$ and the target amplitude is normalized to unity. With $\mathbf{s}_i$ denoting the array manifold vector with components $\cos(n\pi\cos\theta_i)$, the covariance of the array output is given by $\mathbf{R} = \sigma_\eta^2\mathbf{I} + \sigma_1^2\mathbf{s}_1\mathbf{s}_1^T + \sigma_2^2\mathbf{s}_2\mathbf{s}_2^T$.

We fix the number of active elements M at 30 and consider array lengths $N = 30, 40, 50, 60$. For each N and target angle θ_0, the output SNR, defined as the ratio of the mean of the array output to the standard deviation, is maximized. For $N = 30$, the SNR is maximized by the non-sparse MVDR solution, i.e., $\mathbf{b} \propto \mathbf{R}^{-1}\mathbf{s}_0$. For $N > 30$, we use the sparse design algorithms to perform a search over SNR values, i.e., values of ρ in (11), starting at the maximum SNR for the next lowest value of N, which is always achievable, and increasing in 0.05 dB increments. For each ρ, the algorithm is run in an attempt to obtain a feasible solution to (11) with M non-zero coefficients. The algorithm can be terminated as soon as such a solution is found. The highest SNR achieved by each algorithm is recorded.

In Fig. 3, we compare the SNR as a function of θ_0 for non-sparse MVDR beamformers of lengths 30, 40, and 60, and sparse beamformers of lengths 40 and 60 designed using the backward selection algorithm. The SNR values are normalized



(a) (b)

Fig. 3. Panel (a): Normalized SNR as a function of target angle θ_0 for MVDR beamformers of length 30 (dotted black), sparse beamformers of length 40 designed by the backward selection algorithm (solid blue), and MVDR beamformers of length 40 (dashed red). Panel (b): Same as (a) except that the two upper curves represent beamformers of length 60.

so that 0 dB corresponds to the maximum SNR for a length 30 MVDR beamformer subject to white noise alone, i.e., $\sigma_1^2 = \sigma_2^2 = 0$. With the addition of interference, the SNR for the length 30 MVDR beamformer falls below 0 dB at all angles, with deep notches at $\cos\theta_0 = \cos\theta_1 = 0.18$ and $\cos\theta_0 = \cos\theta_2 = 0.73$ where the target coincides with an interferer. As the array length increases, so too does the angular resolution and consequently the notch widths decrease. Moreover, the sparse beamformers achieve nearly the same interference rejection as the MVDR beamformers of the same lengths despite having only three-quarters or one-half as many active elements. Increasing the array length also improves the SNR away from the interferers. The length 40 sparse beamformer nearly matches the SNR improvement of the length 40 MVDR beamformer due to judicious placement of the 30 active elements, while the same is true to a lesser extent for the length 60 beamformers. Significant gaps exist only around $\cos\theta_0 = 0$ and $\cos\theta_0 = 1/2$. The array manifold vectors at these angles have components of equal or nearly equal magnitude, and hence a beamformer with more active elements can collect appreciably more energy from the target direction.

Table IV summarizes the relative performance of the algorithms. On the left, the interferer parameters are set to their nominal values and the percentage of instances (corresponding to different θ_0 values) in which the heuristic algorithms agree with the branch-and-bound algorithm is recorded. On the right, additional instances are generated by varying $\cos\theta_1$ between 0 and 1 while using nominal values for the other parameters, and also varying σ_1^2 between 1 and 40 dB in the same manner, for a total of over 19000 instances. The largest-coefficients and forward selection algorithms are then compared to the backward selection algorithm in terms of SNR achieved. As in Section V-A, backward selection is optimal in almost all instances and consistently performs as well or better than the other heuristics, with the differences becoming more apparent as N increases. In the instances in which the other heuristics are better, the SNR difference is never more than 0.05 dB (a single increment). In the other direction, the differences are larger but rarely exceed a few tenths of a dB in this beamforming example.

In Table V, we report average execution times per instance of (11) for the heuristic algorithms. As in Table III, all of the heuristics are comparable and very efficient.

TABLE IV
PERFORMANCE OF THE LARGEST-COEFFICIENT (LC), FORWARD SELECTION (FS), BACKWARD SELECTION (BS), AND BRANCH-AND-BOUND (BB) ALGORITHMS IN THE BEAMFORMING EXAMPLE.

% in agreement with BB
($\cos\theta_1 = 0.18$, $\sigma_1^2 = 10$ dB)

N	LC	FS	BS
40	89.3	97.9	100
50	70.0	87.1	98.6
60	48.6	67.1	97.9

% relative to backward selection (all θ_1 and σ_1^2)

	largest-coefficients			forward selection		
N	better	same	worse	better	same	worse
40	$\ll 0.1$	91.1	8.9	0.1	98.8	1.2
50	$\ll 0.1$	71.8	28.2	0.3	81.0	18.7
60	< 0.1	50.7	49.3	0.8	65.3	33.9

TABLE V
AVERAGE EXECUTION TIMES IN MILLISECONDS FOR THE HEURISTIC ALGORITHMS IN THE BEAMFORMING EXAMPLE.

N	largest-coefficients	forward selection	backward selection
40	1.0	1.6	1.9
50	1.3	2.3	2.6
60	1.6	3.5	4.3

We note that there is partial theoretical support for the near-optimality of the backward selection algorithm observed in this section. In [54], Couvreur and Bresler prove that for full-rank sparse linear inverse problems, the solution produced by backward selection is optimal if the associated residual (the value of the quadratic form $(\mathbf{b}-\mathbf{c})^T\mathbf{Q}(\mathbf{b}-\mathbf{c})$ in the present context) is smaller than a threshold. Unfortunately, computing the threshold requires combinatorial optimization in its own right, so the result in [54] does not yield a practical test for optimality.

VI. Conclusions and Future Work

We have shown that the problems of sparse filter design for least-squares frequency response approximation, for signal estimation, and for signal detection can be unified under the single framework of quadratically-constrained sparsity maximization. This framework is quite general and has potential applications beyond filter design, for example in subset selection for linear regression [44] and cardinality-constrained portfolio optimization [55]. Several special cases were identified, namely those with diagonal, block-diagonal, banded, or well-conditioned $\mathbf{Q}$ matrices, in which the main optimization problem (1) admits efficient and exact solutions. These special case solutions could be extended to yield approximations in more general cases, and the deviation from optimality could perhaps be quantified if the required conditions for exactness (diagonality, etc.) are approximately satisfied. In [45], we explore one such approximation based on the diagonal case

for the specific purpose of obtaining bounds for use in branch-and-bound.

For the general case, we focused in this paper on a low-complexity backward selection algorithm with attention paid to its efficient implementation. We consider exact algorithms in a companion paper [45]. Design experiments demonstrated that backward selection consistently outperforms the largest-coefficients and forward greedy heuristics and often results in optimal or near-optimal designs. Our results therefore lend confidence to the use of backward selection in settings where computation is limited. It would be instructive in future work to identify instances in which backward selection is far from optimal to indicate where further improvements can be made.

APPENDIX

In this appendix, we show that an arbitrary subproblem defined by subsets $(\mathcal{Z},\mathcal{U},\mathcal{F})$ can be reduced to the problem in (29) with parameters given by (30) in terms of the original parameters $\mathbf{Q}$, $\mathbf{c}$, and γ. The subsets $\mathcal{Z}$, $\mathcal{U}$, and $\mathcal{F}$ are as defined in Section IV.

The reduction can be carried out in the two steps $(\emptyset,\emptyset,\{1,\ldots,N\}) \longrightarrow (\mathcal{Z},\emptyset,\mathcal{Y}=\mathcal{U}\cup\mathcal{F}) \longrightarrow (\mathcal{Z},\mathcal{U},\mathcal{F})$. In the first step, the constraints $b_n = 0$ for $n \in \mathcal{Z}$ reduce the zero-norm $\|\mathbf{b}\|_0$ to $\|\mathbf{b}_\mathcal{Y}\|_0$ and the quadratic constraint in (1) to (16). By completing the square, (16) can be rewritten as

$$\begin{bmatrix}\mathbf{b}_\mathcal{U}-\mathbf{c}'_\mathcal{U}\\ \mathbf{b}_\mathcal{F}-\mathbf{c}'_\mathcal{F}\end{bmatrix}^T \begin{bmatrix}\mathbf{Q}_{\mathcal{U}\mathcal{U}} & \mathbf{Q}_{\mathcal{U}\mathcal{F}}\\ \mathbf{Q}_{\mathcal{F}\mathcal{U}} & \mathbf{Q}_{\mathcal{F}\mathcal{F}}\end{bmatrix} \begin{bmatrix}\mathbf{b}_\mathcal{U}-\mathbf{c}'_\mathcal{U}\\ \mathbf{b}_\mathcal{F}-\mathbf{c}'_\mathcal{F}\end{bmatrix} \le \gamma_{\text{eff}}, \qquad (36)$$

where $\mathcal{Y}$ has been partitioned into $\mathcal{U}$ and $\mathcal{F}$, $\mathbf{c}'_\mathcal{U} = \mathbf{c}_\mathcal{U} + \left((\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1}\mathbf{Q}_{\mathcal{Y}\mathcal{Z}}\mathbf{c}_\mathcal{Z}\right)_\mathcal{U}$, $\mathbf{c}'_\mathcal{F} = \mathbf{c}_\mathcal{F} + \left((\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1}\mathbf{Q}_{\mathcal{Y}\mathcal{Z}}\mathbf{c}_\mathcal{Z}\right)_\mathcal{F}$, and γ_{eff} is as defined in (30c).

In the second step $(\mathcal{Z},\emptyset,\mathcal{U}\cup\mathcal{F}) \longrightarrow (\mathcal{Z},\mathcal{U},\mathcal{F})$, the infeasibility of zero values for b_n, $n \in \mathcal{U}$ allows the cost $\|\mathbf{b}_\mathcal{Y}\|_0$ to be rewritten as $|\mathcal{U}| + \|\mathbf{b}_\mathcal{F}\|_0$. Since $\mathbf{b}_\mathcal{U}$ no longer has any effect on the cost, its value can be freely chosen, and in the interest of minimizing $\|\mathbf{b}_\mathcal{F}\|_0$, $\mathbf{b}_\mathcal{U}$ should be chosen as a function of $\mathbf{b}_\mathcal{F}$ to maximize the margin in constraint (36), thereby making the set of feasible $\mathbf{b}_\mathcal{F}$ as large as possible. This is equivalent to minimizing the left-hand side of (36) with respect to $\mathbf{b}_\mathcal{U}$ while holding $\mathbf{b}_\mathcal{F}$ constant. Similar to the minimization of (16) with respect to $\mathbf{b}_\mathcal{Y}$, we obtain

$$\mathbf{b}^*_\mathcal{U} = \mathbf{c}'_\mathcal{U} - (\mathbf{Q}_{\mathcal{U}\mathcal{U}})^{-1}\mathbf{Q}_{\mathcal{U}\mathcal{F}}(\mathbf{b}_\mathcal{F} - \mathbf{c}'_\mathcal{F}) \qquad (37)$$

as the minimizer of (36). Substituting (37) into (36) results in the constraint in (29) except with $\mathbf{c}'_\mathcal{F}$ in place of $\mathbf{c}_{\text{eff}}$. By expressing $(\mathbf{Q}_{\mathcal{Y}\mathcal{Y}})^{-1}$ in terms of the block decomposition of $\mathbf{Q}_{\mathcal{Y}\mathcal{Y}}$ in (36), it can be shown that $\mathbf{c}'_\mathcal{F} = \mathbf{c}_{\text{eff}}$, thus completing the reduction.

REFERENCES

[1] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, Apr. 1992.

[2] R. M. Leahy and B. D. Jeffs, "On the design of maximally sparse beamforming arrays," *IEEE Trans. Antennas Propag.*, vol. 39, pp. 1178–1187, Aug. 1991.

[3] Y. Neuvo, C.-Y. Dong, and S. Mitra, "Interpolated finite impulse response filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, pp. 563–570, Jun. 1984.

[4] T. Saramaki, T. Neuvo, and S. K. Mitra, "Design of computationally efficient interpolated FIR filters," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 70–88, Jan. 1988.

[5] Y. C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Trans. Circuits Syst.*, vol. 33, pp. 357–364, Apr. 1986.

[6] Y. C. Lim and Y. Lian, "Frequency-response masking approach for digital filter design: complexity reduction via masking filter factorization," *IEEE Trans. Circuits Syst. II*, vol. 41, pp. 518–525, Aug. 1994.

[7] J. L. H. Webb and D. C. Munson, "Chebyshev optimization of sparse FIR filters using linear programming with an application to beamforming," *IEEE Trans. Signal Process.*, vol. 44, pp. 1912–1922, Aug. 1996.

[8] J. T. Kim, W. J. Oh, and Y. H. Lee, "Design of nonuniformly spaced linear-phase FIR filters using mixed integer linear programming," *IEEE Trans. Signal Process.*, vol. 44, pp. 123–126, Jan. 1996.

[9] Y.-S. Song and Y. H. Lee, "Design of sparse FIR filters based on branch-and-bound algorithm," in *Proc. Midwest Symp. Circuits. Syst.*, vol. 2, Aug. 1997, pp. 1445–1448.

[10] J.-K. Liang, R. de Figueiredo, and F. Lu, "Design of optimal Nyquist, partial response, Nth band, and nonuniform tap spacing FIR digital filters using linear programming techniques," *IEEE Trans. Circuits Syst.*, vol. 32, pp. 386–392, Apr. 1985.

[11] D. Mattera, F. Palmieri, and S. Haykin, "Efficient sparse FIR filter design," in *Proc. ICASSP*, vol. 2, May 2002, pp. 1537–1540.

[12] T. Baran, D. Wei, and A. V. Oppenheim, "Linear programming algorithms for sparse filter design," *IEEE Trans. Signal Process.*, vol. 58, pp. 1605–1617, Mar. 2010.

[13] D. Wei, "Non-convex optimization for the design of sparse FIR filters," in *IEEE 15th Workshop on Statistical Signal Processing*, Sep. 2009, pp. 117–120.

[14] J. W. Adams, "FIR digital filters with least-squares stopbands subject to peak-gain constraints," *IEEE Trans. Circuits Syst.*, vol. 39, pp. 376–388, Apr. 1991.

[15] M. Smith and D. Farden, "Thinning the impulse response of FIR digital filters," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 6, 1981, pp. 240–242.

[16] R. J. Hartnett and G. F. Boudreaux-Bartels, "On the use of cyclotomic polynomial prefilters for efficient FIR filter design," *IEEE Trans. Signal Process.*, vol. 41, pp. 1766–1779, May 1993.

[17] S. A. Raghavan, J. K. Wolf, L. B. Milstein, and L. C. Barbosa, "Non-uniformly spaced tapped-delay-line equalizers," *IEEE Trans. Commun.*, vol. 41, no. 9, pp. 1290–1295, Sep. 1993.

[18] I. Lee, "Optimization of tap spacings for the tapped delay line decision feedback equalizer," *IEEE Commun. Lett.*, vol. 5, no. 10, pp. 429–431, Oct. 2001.

[19] M. Kocic, D. Brady, and M. Stojanovic, "Sparse equalization for real-time digital underwater acoustic communications," in *IEEE OCEANS*, vol. 3, Oct. 1995, pp. 1417–1422.

[20] A. A. Rontogiannis and K. Berberidis, "Efficient decision feedback equalization for sparse wireless channels," *IEEE Trans. Wireless Commun.*, vol. 2, no. 3, pp. 570–581, May 2003.

[21] F. K. H. Lee and P. J. McLane, "Design of nonuniformly spaced tapped-delay-line equalizers for sparse multipath channels," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 530–535, Apr. 2004.

[22] I. J. Fevrier, S. B. Gelfand, and M. P. Fitz, "Reduced complexity decision feedback equalization for multipath channels with large delay spreads," *IEEE Trans. Commun.*, vol. 47, no. 6, pp. 927–937, Jun. 1999.

[23] S. Ariyavisitakul, N. R. Sollenberger, and L. J. Greenstein, "Tap-selectable decision-feedback equalization," *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1497–1500, Dec. 1997.

[24] M. J. Lopez and A. C. Singer, "A DFE coefficient placement algorithm for sparse reverberant channels," *IEEE Trans. Commun.*, vol. 49, no. 8, pp. 1334–1338, Aug. 2001.

[25] H. Sui, E. Masry, and B. D. Rao, "Chip-level DS-CDMA downlink interference suppression with optimized finger placement," *IEEE Trans. Signal Process.*, vol. 54, no. 10, pp. 3908–3921, Oct. 2006.

[26] G. Kutz and D. Raphaeli, "Determination of tap positions for sparse equalizers," *IEEE Trans. Commun.*, vol. 55, no. 9, pp. 1712–1724, Sep. 2007.

[27] D. Giacobello, M. G. Christensen, M. N. Murthi, S. H. Jensen, and M. Moonen, "Sparse linear prediction and its applications to speech processing," *IEEE Audio, Speech, Language Process.*, vol. 20, no. 5, pp. 1644–1657, Jul. 2012.

[28] H. L. V. Trees, *Detection, Estimation, and Modulation Theory*. New York: John Wiley & Sons, 2004, vol. 1.

[29] C. K. Sestok, "Data selection in binary hypothesis testing," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, Dec. 2003.

[30] ——, "Data selection for detection of known signals: The restricted-length matched filter," in *Proc. ICASSP*, vol. 2, May 2004, pp. 1085–1088.

[31] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, pp. 489–509, Feb. 2006.

[32] ——, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, pp. 1207–1223, Aug. 2006.

[33] J. J. Fuchs, "Recovery of exact sparse representations in the presence of bounded noise," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3601–3608, Oct. 2005.

[34] J. A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1030–1051, Mar. 2006.

[35] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, pp. 33–61, Aug. 1998.

[36] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, Nov. 2004.

[37] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. Simul.*, vol. 4, no. 4, pp. 1168–1200, Nov. 2005.

[38] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, "A variational formulation for frame-based inverse problems," *Inverse Problems*, vol. 23, no. 4, pp. 1495–1518, Jun. 2007.

[39] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2980–2991, Dec. 2007.

[40] P. L. Combettes and J.-C. Pesquet, "A proximal decomposition method for solving convex variational inverse problems," *Inverse Problems*, vol. 24, no. 6, pp. 65 014–65 040, Dec. 2008.

[41] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 374–377, Mar. 2002.

[42] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1708–1721, Mar. 2010.

[43] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. D. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proc. IEEE*, vol. 98, no. 6, pp. 1058–1076, Jun. 2010.

[44] A. J. Miller, *Subset selection in regression*, 2nd ed. Boca Raton, FL: Chapman & Hall/CRC, 2002.

[45] D. Wei and A. V. Oppenheim, "A branch-and-bound algorithm for quadratically-constrained sparse filter design," *IEEE Trans. Signal Process.*, to appear.

[46] ——, "Sparsity maximization under a quadratic constraint with applications in filter design," in *Proc. ICASSP*, Mar. 2010, pp. 117–120.

[47] A. Gomaa and N. Al-Dhahir, "A new design framework for sparse FIR MIMO equalizers," *IEEE Trans. Commun.*, 2011, to appear.

[48] D. H. Johnson and D. E. Dudgeon, *Array signal processing*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1993.

[49] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*. Cambridge, UK: Cambridge University Press, 1994.

[50] D. Wei, "Design of discrete-time filters for efficient implementation," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, May 2011.

[51] Y. Wu, X. Wang, R. Citta, B. Ledoux, S. Lafleche, and B. Caron, "An ATSC DTV receiver with improved robustness to multipath and distributed transmission environments," *IEEE Trans. Broadcast.*, vol. 50, pp. 32–41, Mar. 2004.

[52] A. Chini, Y. Wu, M. El-Tanany, and S. Mahmoud, "Filtered decision feedback channel estimation for OFDM-based DTV terrestrial broadcasting system," *IEEE Trans. Broadcast.*, vol. 44, no. 1, pp. 2–11, Mar. 1998.

[53] K. Manolakis, A. Ibing, and V. Jungnickel, "Performance evaluation of a 3GPP LTE terminal receiver," in *Proc. 14th Eur. Wireless Conf.*, Jun. 2008, pp. 1–5.

[54] C. Couvreur and Y. Bresler, "On the optimality of the backward greedy algorithm for the subset selection problem," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 3, pp. 797–808, 2000.

[55] D. Bertsimas and R. Shioda, "Algorithm for cardinality-constrained quadratic optimization," *Comput. Optim. Appl.*, vol. 43, pp. 1–22, May 2009.

Dennis Wei (S'09–M'11) received S.B. degrees in electrical engineering and in physics in 2006, the M.Eng. degree in electrical engineering in 2007, and the Ph.D. degree in electrical engineering in 2011, all from the Massachusetts Institute of Technology. He is currently a post-doctoral research fellow in the Department of Electrical Engineering and Computer Science at the University of Michigan. His research interests lie broadly in signal processing, optimization, and statistical inference and learning. Areas of focus include adaptive sensing and processing, filter design, and non-uniform sampling. Dr. Wei is a member of Phi Beta Kappa, Sigma Xi, Eta Kappa Nu, and Sigma Pi Sigma. He has been a recipient of the William Asbjornsen Albert Memorial Fellowship at MIT and a Siebel Scholar.

Charles K. Sestok (M'99) received the S.B. degree in physics, the S.M. degree in electrical engineering, and the Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology (MIT) in 1997, 1999, and 2003 respectively. He is currently a member of the Texas Instruments Systems and Applications Research and Development Center, where his research interests include signal processing, communication theory, and digitally enhanced analog systems. Dr. Sestok has been issued eight patents in these areas. He received a National Science Foundation Graduate Research Fellowship in 1997 and is a Member of Sigma Pi Sigma and Phi Beta Kappa.

Alan V. Oppenheim (M'65–SM'71–F'77–LF'03) was born in New York, New York on November 11, 1937. He received S.B. and S.M. degrees in 1961 and an Sc.D. degree in 1964, all in Electrical Engineering, from the Massachusetts Institute of Technology. He is also the recipient of an honorary doctorate from Tel Aviv University.

In 1964, Dr. Oppenheim joined the faculty at MIT, where he is currently Ford Professor of Engineering. Since 1967 he has been affiliated with MIT Lincoln Laboratory and since 1977 with the Woods Hole Oceanographic Institution. His research interests are in the general area of signal processing and its applications. He is coauthor of the widely used textbooks Discrete-Time Signal Processing (now in its third edition), Signals and Systems and Digital Signal Processing. He is also editor of several advanced books on signal processing and coauthor of the text Signals, Systems, and Inference, published online through MITs OpenCourseWare.

Dr. Oppenheim is a member of the National Academy of Engineering, a fellow of the IEEE, and a member of Sigma Xi and Eta Kappa Nu. He has been a Guggenheim Fellow and a Sackler Fellow. He has received a number of awards for outstanding research and teaching, including the IEEE Education Medal, the IEEE Jack S. Kilby Signal Processing Medal, the IEEE Centennial Award and the IEEE Third Millennium Medal. From the IEEE Signal Processing Society he has been honored with the Education Award, the Society Award, the Technical Achievement Award and the Senior Award. He has also received a number of awards at MIT for excellence in teaching, including the Bose Award, the Everett Moore Baker Award, and several awards for outstanding advising and mentoring.