Adaptive Beamforming With Sidelobe Control: A Second-Order Cone Programming Approach

Jing Liu, Alex B. Gershman, *Senior Member, IEEE*, Zhi-Quan Luo, *Member, IEEE*, and Kon Max Wong, *Fellow, IEEE*

***Abstract*—A new approach to adaptive beamforming with sidelobe control is developed. The proposed beamformer represents a modification of the popular minimum variance distortionless response (MVDR) beamformer. It minimizes the array output power while maintaining the distortionless response in the direction of the desired signal and a sidelobe level that is strictly guaranteed to be lower than some given (prescribed) threshold value. The resulting modified MVDR problem is shown to be convex, and its second-order cone (SOC) formulation is obtained that facilitates a computationally efficient way to implement our beamformer using the interior point method.**

***Index Terms*—Adaptive beamforming, second-order cone (SOC) programming, sidelobe control.**

I. Introduction

ADAPTIVE beamforming has found numerous applications to radar, sonar, wireless communications, seismology, and microphone arrays. One of the most popular approaches to adaptive beamforming is the so-called minimum variance distortionless response (MVDR) processor, which minimizes the array output power while maintaining a distortionless mainlobe response toward the desired signal [1]. Unfortunately, the MVDR beamformer can have unacceptably high sidelobes in the case of low sample support [2]. In adaptive array systems, this may lead to a substantial performance degradation in the presence of unexpected (i.e., suddenly appearing) interferers. Indeed, after the appearance of any new interferer, the MVDR beamformer requires a certain transition time to suppress it, and therefore, during this time interval its performance may break down.

To overcome this shortcoming of the MVDR beamformer, some sidelobe pattern control is required. Several approaches to sidelobe control have been proposed (see [2]–[6] and references therein). For example, an iterative approach to adaptive beamforming with sidelobe control has been developed in [2] and [3] using quadratic beampattern constraints limiting deviations from the desired beampattern. Other well-known approaches are the diagonal loading (DL) method [5] and the penalty function technique [6]. A common shortcoming of these techniques is that they cannot guarantee that the sidelobes are exactly below the required level, even when the corresponding optimization problem is feasible. If the problem is infeasible, these techniques are not able to detect this fact and correct the corresponding optimization parameters.

In this letter, we propose a new alternative approach to adaptive beamforming with sidelobe control that mitigates the aforementioned drawbacks of the earlier approaches. We reformulate the MVDR beamforming problem using multiple additional quadratic inequality constraints outside the mainlobe beampattern area. These constraints guarantee that the beampattern sidelobe level remains lower than a certain prescribed value. Then, we show that this optimization problem can be rewritten in an equivalent form of the second-order cone (SOC) program, which can be solved efficiently using the well-established interior point method [7]. Our approach also allows us to determine whether the problem is feasible and, in the case of an infeasible problem, to correct the optimization parameters in an adaptive way to ensure feasibility.

II. Background

Assume that an array of N sensors receives a narrowband signal $s(t)$. The $N \times 1$ vector of the array snapshot data can be modeled as

$$\mathbf{x}(t) = \beta s(t)\mathbf{a}_s + \mathbf{x}_i(t) + \mathbf{x}_n(t)$$

where β is a binary parameter indicating whether the desired signal is present in the training snapshots; $\mathbf{a}_s$ is the $N \times 1$ steering vector of the desired signal; $\mathbf{x}_i(t)$ is the $N \times 1$ vector of interference; and $\mathbf{x}_n$ is the $N \times 1$ vector of sensor noise.

The beamformer output is given by

$$y(t) = \mathbf{w}^H \mathbf{x}(t)$$

where $\mathbf{w}$ is the $N \times 1$ weight vector, and $(\cdot)^H$ denotes the Hermitian transpose.

The MVDR beamformer minimizes the array output power while keeping the unit gain in the direction of the desired signal

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \mathbf{a}_s = 1 \tag{1}$$

where

$$\mathbf{R} = E\{\mathbf{x}(t)\mathbf{x}^H(t)\} \tag{2}$$

is the $N \times N$ array covariance matrix.

A closed-form solution to (1) is given by [1]

$$\mathbf{w}_{\mathrm{MVDR}} = \frac{1}{\mathbf{a}_s^H \mathbf{R}^{-1} \mathbf{a}_s} \mathbf{R}^{-1} \mathbf{a}_s. \tag{3}$$

Manuscript received July 9, 2002; revised November 22, 2002. This work was supported in parts by the Natural Sciences and Engineering Research Council (NSERC) of Canada; Communications and Information Technology Ontario (CITO); the Premier Research Excellence Award Program of the Ministry of Energy, Science, and Technology (MEST) of Ontario; the Canada Research Chair Program; and the Wolfgang Paul Award Program of the Alexander von Humboldt Foundation.

The authors are with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON L8S 4K1 Canada.

Digital Object Identifier 10.1109/LSP.2003.817852

In practice, the sample covariance matrix

$$\hat{\mathbf{R}} = \frac{1}{M}\sum_{t=1}^{M}\mathbf{x}(t)\mathbf{x}(t)^H \tag{4}$$

is used in (3) instead of the true array covariance matrix (2). Here, M is the number of snapshots.

III. Adaptive Beamforming With Sidelobe Control

Let $\theta_j \in \Theta$ $(j = 1, \ldots, J)$ be a chosen (uniform or nonuniform) grid that approximates the sidelobe beampattern areas Θ using a finite number of angles. To control the sidelobe level, we use the following multiple quadratic inequality constraints outside the mainlobe beampattern area:

$$|\mathbf{w}^H\mathbf{a}(\theta_j)|^2 \le \varepsilon, \qquad j = 1, \ldots, J \tag{5}$$

where ε is the prescribed sidelobe level.

Adding the constraints (5) to the MVDR beamforming problem (1) and using the sample covariance matrix (4) instead of the true covariance matrix (2), we obtain the following modified MVDR problem:

$$\begin{aligned} &\min_{\mathbf{w}} \quad \mathbf{w}^H\hat{\mathbf{R}}\mathbf{w} \\ &\text{subject to} \quad \mathbf{w}^H\mathbf{a}_s = 1, \\ &|\mathbf{w}^H\mathbf{a}(\theta_j)|^2 \le \varepsilon, \quad j = 1, \ldots, J. \end{aligned} \tag{6}$$

This problem has quadratic objective function (note that $\hat{\mathbf{R}}$ is positive semidefinite). There is a single linear equality constraint and multiple quadratic inequality constraints. Therefore, the optimization problem (6) is convex. In Section IV, we will convert this problem to the standard SOC program.

IV. Second-Order Cone Formulation

In [7], the dual standard form of convex conic optimization problem is defined as

$$\max_{\mathbf{y}} \mathbf{b}^T\mathbf{y} \quad \text{subject to} \quad \mathbf{c} - \mathbf{A}^T\mathbf{y} \in \mathcal{K} \tag{7}$$

where $\mathbf{y}$ is a vector containing the designed variables; $\mathbf{A}$ is an arbitrary matrix; $\mathbf{b}$ and $\mathbf{c}$ are arbitrary vectors; $\mathcal{K}$ is a symmetric cone consisting of Cartesian products of elementary cones (each corresponding to a constraint); and $(\cdot)^T$ denotes the transpose. Note that $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}$ can be complex-valued and must have matching dimensions.

For our problem, the elementary cones are either SOCs (for inequality constraints) or zero-cones $\{0\}$ (for equality constraints). The q-dimensional SOC is defined as

$$\text{SOC}^{q+1} \triangleq \{(x_1, \mathbf{x}_2) \in \mathcal{R} \times \mathcal{C}^q \mid x_1 \ge \|\mathbf{x}_2\|\}$$

where x_1 is a real scalar; $\mathbf{x}_2$ is a complex q-dimensional vector; and $\|\cdot\|$ denotes the Euclidean norm. The zero cone is defined as

$$\{0\} \triangleq \{x \in \mathcal{C} \mid x = 0\}.$$

For example, with $\mathcal{K}$ as $\{0\}^f \times \text{SOC}^{q+1}$, the condition $\mathbf{c} - \mathbf{A}^T\mathbf{y} \in \mathcal{K}$ indicates that the first f elements of vector $\mathbf{c} - \mathbf{A}^T\mathbf{y}$ are constrained to be equal to zero, while the remaining $(q+1)$ elements must lie in a SOC.

In what follows we formulate the convex optimization problem (6) in the dual standard form (7) of SOC program, which can be solved efficiently using the well-established interior point method [7].

First of all, we convert the quadratic objective function of (6) to a linear one. Notice that

$$\mathbf{w}^H\hat{\mathbf{R}}\mathbf{w} = \|\mathbf{L}\mathbf{w}\|^2$$

where $\mathbf{L}$ is the Cholesky factor of $\hat{\mathbf{R}}$, i.e., $\mathbf{L}^H\mathbf{L} = \hat{\mathbf{R}}$. Clearly, minimizing $\mathbf{w}^H\hat{\mathbf{R}}\mathbf{w}$ is equivalent to minimizing $\|\mathbf{L}\mathbf{w}\|$. Introducing a new scalar nonnegative variable τ and a new constraint $\|\mathbf{L}\mathbf{w}\| \le \tau$, we can convert (6) into the following form:

$$\begin{aligned} &\min_{\tau,\mathbf{w}} \tau \quad \text{subject to} \quad \|\mathbf{L}\mathbf{w}\| \le \tau, \quad \mathbf{a}_s^H\mathbf{w} = 1, \\ &|\mathbf{a}^H(\theta_j)\mathbf{w}|^2 \le \varepsilon, \qquad j = 1, \ldots, J. \end{aligned}$$

Since $|\mathbf{a}^H(\theta_j)\mathbf{w}|^2 \le \varepsilon, \quad j = 1, \ldots, J$, the quadratic inequality constraints can be written as SOC constraints $|\mathbf{a}^H(\theta_j)\mathbf{w}| \le \sqrt{\varepsilon}, \quad j = 1, \ldots, J$. Introducing new variables $y_1 = \tau$, $y_2 = \sqrt{\varepsilon}$, $\mathbf{y}_3 = \mathbf{w}$, and $\mathbf{y} = [y_1, y_2, \mathbf{y}_3^T]^T$, we obtain the following problem:

$$\begin{aligned} &\min_{\mathbf{y}} y_1 \quad \text{subject to} \quad y_2 = \sqrt{\varepsilon}, \qquad \mathbf{a}_s^H\mathbf{y}_3 = 1, \\ &\|\mathbf{L}\mathbf{y}_3\| \le y_1, \quad |\mathbf{a}^H(\theta_j)\mathbf{y}_3| \le y_2, \qquad j = 1, \ldots, J. \end{aligned} \tag{8}$$

Define $\mathbf{b} \triangleq [-1, 0, \ldots, 0]^T$ so that $-y_1 = \mathbf{b}^T\mathbf{y}$. Among the constraints of (8), the first two equalities $y_2 = \sqrt{\varepsilon}$ and $\mathbf{a}_s^H\mathbf{y}_3 = 1$ can be represented as two zero-cone constraints

$$\begin{aligned} \begin{pmatrix} \sqrt{\varepsilon} - y_2 \\ 1 - \mathbf{a}_s^H\mathbf{y}_3 \end{pmatrix} &= \begin{pmatrix} \sqrt{\varepsilon} \\ 1 \end{pmatrix} - \begin{pmatrix} 0 & 1 & \mathbf{0}^T \\ 0 & 0 & \mathbf{a}_s^H \end{pmatrix}\mathbf{y} \\ &\triangleq \mathbf{c}_1 - \mathbf{A}_1^T\mathbf{y} \in \{0\}^2. \end{aligned}$$

The SOC constraints

$$\|\mathbf{L}\mathbf{y}_3\| \le y_1, \qquad |\mathbf{a}^H(\theta_j)\mathbf{y}_3| \le y_2, \quad j = 1, \ldots, J$$

take the form

$$\begin{aligned} \begin{pmatrix} y_1 \\ \mathbf{L}\mathbf{y}_3 \end{pmatrix} &= \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} -1 & 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{0} & -\mathbf{L} \end{pmatrix}\mathbf{y} \\ &\triangleq \mathbf{c}_2 - \mathbf{A}_2^T\mathbf{y} \in \text{SOC}^{N+1}, \\ \begin{pmatrix} y_2 \\ \mathbf{a}^H(\theta_j)\mathbf{y}_3 \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 & -1 & \mathbf{0}^T \\ 0 & 0 & -\mathbf{a}^H(\theta_j) \end{pmatrix}\mathbf{y} \\ &\triangleq \mathbf{c}_{j+2} - \mathbf{A}_{j+2}^T\mathbf{y} \in \text{SOC}^2, \qquad j = 1, \ldots, J. \end{aligned}$$

Let

$$\mathbf{c} \triangleq [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{J+2}] \quad \mathbf{A}^T \triangleq [\mathbf{A}_1^T, \mathbf{A}_2^T, \ldots, \mathbf{A}_{J+2}^T]$$

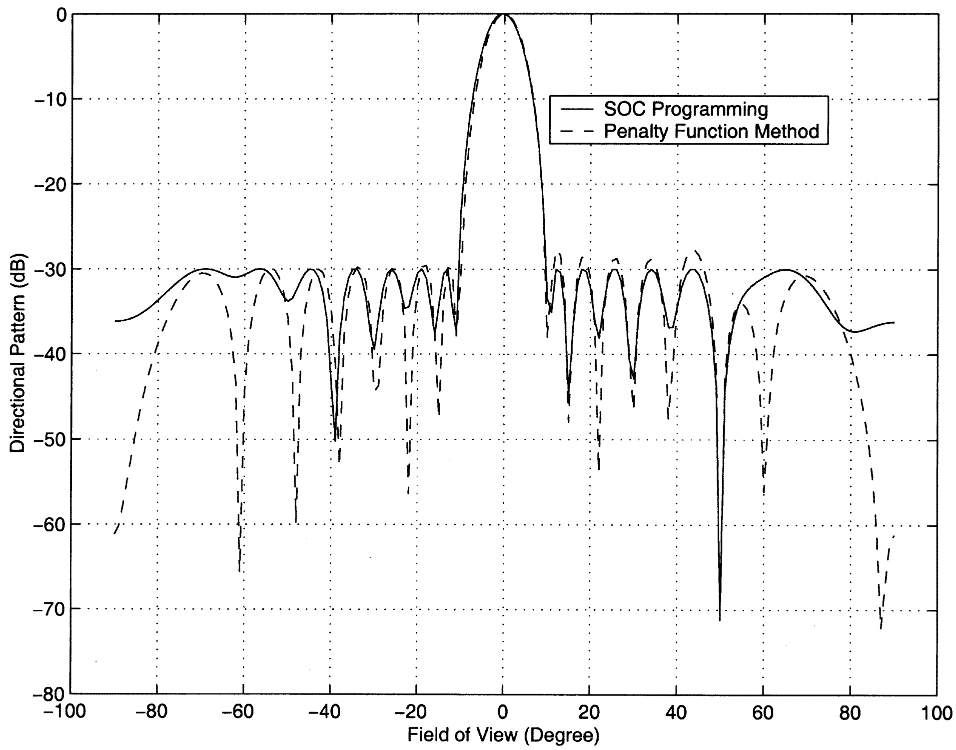

Fig. 1. Directional patterns of the SOC programming and penalty-function-based adaptive beamforming methods.

where $\mathbf{c}_j$ and $\mathbf{A}_j$ $(j = 1, \ldots, J+2)$ are defined according to the equations above. Then, the constraints of (8) become

$$\mathbf{c} - \mathbf{A}^T\mathbf{y} \in \mathcal{K}$$

where $\mathcal{K}$ is the symmetric cone corresponding to the constraints

$$\mathcal{K} = \{0\}^2 \times \text{SOC}^{N+1} \times \underbrace{\text{SOC}^2 \times \cdots \times \text{SOC}^2}_{J}.$$

Thus, with the above definitions of $\mathbf{y}, \mathbf{L}, \mathbf{A}, \mathbf{b}, \mathbf{c}$, and $\mathcal{K}$, we have rewritten the convex optimization problem (6) in the standard form of dual SOC program (7). Notice that we have complex data and variables in the formulation, but this can be accommodated by the existing interior point codes [7].

If the above convex optimization problem is feasible, the proposed adaptive beamformer can be obtained from the optimal solution for $\mathbf{y}$. However, the problem may be infeasible, depending on the chosen sidelobe control parameter ε and the mainlobe width. The advantage of the SOC-programming-based approach is that the problem infeasibility can be easily detected by the convex optimization software [7] (in which case we can adjust the sidelobe control parameters accordingly to ensure feasibility).

The computational complexities of the SOC-programming-based technique and the penalty function method are $O(N^{3.5} + JN^{2.5})$ and $O(N^3 + JN^2)$, respectively. Therefore, the proposed beamformer has a slightly higher computational cost as compared to the beamformer of [6].

V. Simulation Results

We consider a uniform linear array of $N = 16$ sensors spaced half-wavelength apart. The direction of the desired signal is $\theta_s = 0°$. The sidelobe beampattern areas $\Theta = [-90°, -12°] \cup [12°, 90°]$ are chosen and a uniform grid is used to obtain the angles θ_j $(j = 1, \ldots, J)$. The SNR is equal to 0 dB. A single interferer is assumed to have the direction of 50° and a power of 45 dB relative to the signal in each sensor. Sensor noises are modeled as spatially and temporally white Gaussian processes. It is assumed that $\varepsilon = 10^{-3}$, i.e., we require the beampattern sidelobe level to be below −30 dB (under these conditions, the optimization problem is feasible).

The SeDuMi MATLAB toolbox [7] is used to solve the SOC problem. We compare our method with the penalty function method [6], where Chebychev quiescent beampattern is chosen with −30-dB sidelobe level, and the scalar weighting factor $k = 15$ (see [6] for details), while the unconstrained region is chosen to be $[-11°, 11°]$.

In the first example, we use $M = 32$ snapshots to estimate the sample covariance matrix and compute the directional patterns of the penalty-function-based beamformer and the SOC-programming-based beamformer. These directional patterns are plotted in Fig. 1. This figure shows that, although the beampatterns of the methods tested are quite similar (and, in particular, both of them have their mainlobe in the signal direction and deep null in the direction of interferer), the penalty function approach does not guarantee that the sidelobe constraints are exactly satisfied. At the same time, as can be seen from Fig. 1, the proposed method guarantees that these constraints are satisfied exactly.

In the second example, the number of snapshots is varied. Fig. 2 displays the average output SINR of the penalty function

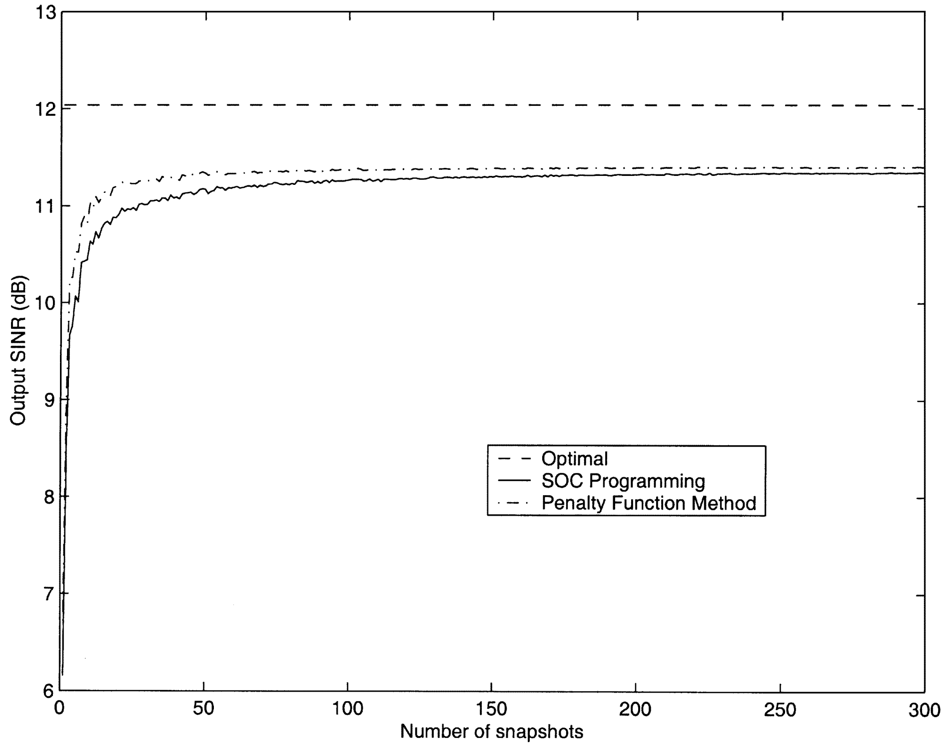

Fig. 2. Output SINR of the SOC-programming and penalty-function-based adaptive beamforming methods versus the number of snapshots.

method and our SOC programming method versus the number of snapshots. It can be seen that at low sample sizes, the penalty function algorithm has a very slight improvement over the SOC programming method (which can be viewed as a price we accept to pay for the strict sidelobe control), while at larger values of M, the performances of both methods merge and achieve the optimal SINR.

Note that we have performed more simulations (whose results are not shown in the interest of brevity) with multiple interferers. These additional simulations have shown that in the case when the number of interferers is increased, the performance of both the methods tested is not seriously affected (although in the latter case the sidelobe constraints in the penalty function method become even more violated than in the single interferer case).

VI. Conclusion

A new approach to adaptive beamforming with sidelobe control has been proposed. Our method minimizes the power of interferences and noise, while maintaining distortionless response in the direction of signal and providing that the beampattern sidelobes are lower than some given threshold value. Convex SOC-programming-based formulation of our beamformer is derived. The advantage of the proposed technique as compared to earlier approaches is that it can guarantee that the sidelobes are strictly below the threshold and detect infeasibility of the problem.

References

[1] R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*. New York: Wiley, 1980.

[2] K. L. Bell, H. L. Van Trees, and L. J. Griffiths, "Adaptive beampattern control using quadratic constraints for circular array STAP," in *Proc. 8th Annual Workshop on Adaptive Sensor Array Processing*, Lexington, MA, Mar. 2000, pp. 43–48.

[3] K. L. Bell and H. L. Van Trees, "Adaptive and nonadaptive beampattern control using quadratic beampattern constraints," in *Proc. 33rd Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 1999, pp. 486–490.

[4] B. D. Van Veen, "Minimum variance beamforming with soft response constraints," *IEEE Trans. Signal Processing*, vol. 39, pp. 1964–1972, Sept. 1991.

[5] B. D. Carlson, "Covariance matrix estimation errors and diagonal loading in adaptive arrays," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 24, pp. 397–401, July 1988.

[6] D. T. Hughes and J. G. McWhirter, "Sidelobe control in adaptive beamforming using a penalty function," in *Proc. ISSPA*, Gold Coast, Australia, 1996.

[7] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optim. Meth. Softw.*, vol. 11–12, pp. 625–653, Aug. 1999.