FULLY AUTOMATIC COMPUTATION OF DIAGONAL LOADING LEVELS FOR ROBUST ADAPTIVE BEAMFORMING

Jian Li, Lin Du[*]

University of Florida
Dept. of Electrical and Computer Engineering
Gainesville, FL 32611-6130, USA.

Petre Stoica

Uppsala University
Dept. of Information Technology
SE-75105, Uppsala, Sweden.

ABSTRACT

One of the most well-known robust adaptive beamforming approaches is diagonal loading. However, there are usually no clear guidelines on how to choose the diagonal loading level reliably. In this paper, we present algorithms that can compute the diagonal loading level fully automatically from the given data without the need of specifying any user parameters. The proposed diagonal loading algorithms use shrinkage-based covariance matrix estimates, instead of the conventional sample covariance matrix, in the standard Capon beamforming formulation. The performance of the resulting beamformers is illustrated via numerical examples and compared with other adaptive beamforming techniques.

Index Terms— Diagonal loading, Adaptive beamforming

1. INTRODUCTION

The Standard Capon Beamformer (SCB) is an optimal spatial filter that maximizes the array output signal to interference plus noise ratio (SINR), provided that the true covariance matrix and the signal steering vector are accurately known. However, the covariance matrix can be inaccurately estimated due to limited data samples and the knowledge of the steering vector can be imprecise due to look direction errors or imperfect array calibration. Whenever these factors exist, there is a clear performance degradation for SCB. Therefore, adaptive beamforming approaches robust to small sample size problems and steering vector errors are needed.

One of the most well-known robust adaptive beamforming approaches is diagonal loading [1]. The main drawback of this method is that there is no clear way to choose the diagonal loading level reliably. Several recent robust adaptive beamformers have been proposed [2], which can be regarded as diagonal loading approaches, with the diagonal loading level calculated based on the uncertainty set of the array steering vector. However, we still need to specify the parameter related to the size of the uncertainty set. Indeed, fully parameter-free robust adaptive beamformers are scarce. One example is the HKB-based SCB [3], which is also a diagonal loading algorithm. However, it may have an inherent problem in choosing an appropriate diagonal loading level.

We provide alternative approaches for the fully automatic computation of the diagonal loading level. We replace the conventional sample covariance matrix used in SCB by an enhanced estimate based on a shrinkage method [4]. Numerical examples are presented to compare the performance of the proposed beamformers with that of HKB and SCB in terms of output SINR and signal-of-interest (SOI) power estimation.

Notation: The superscript $(\cdot)^*$ denotes the conjugate transpose, $(\cdot)^T$ denotes the transpose, $E(\cdot)$ is the expectation operator, $\mathrm{tr}(\cdot)$ is the trace operator, and $\|\cdot\|$ denotes the Frobenius norm for a matrix or the Euclidean norm for a vector.

2. PROBLEM FORMULATION

Consider an array comprising M sensors and let $\mathbf{R}$ denote the theoretical covariance matrix of the array output vector. We assume that $\mathbf{R} > 0$ (positive definite) has the following form:

$$\mathbf{R} = \sigma_0^2 \mathbf{a}_0 \mathbf{a}_0^* + \mathbf{Q}, \tag{1}$$

where σ_0^2 denotes the power of the SOI, $\mathbf{a}_0$ is the array steering vector of the SOI with $\|\mathbf{a}_0\|^2 = M$, and $\mathbf{Q}$ is the interference-plus-noise covariance matrix.

Under ideal conditions, i.e., $\mathbf{a}_0$ and $\mathbf{R}$ are accurately known, the SCB maximizes the output SINR and the optimal value is $\mathrm{SINR}_{\mathrm{opt}} = \sigma_0^2 \mathbf{a}_0^* \mathbf{Q}^{-1} \mathbf{a}_0$. In practice, the exact covariance matrix $\mathbf{R}$ is unavailable. Therefore, $\mathbf{R}$ is replaced by the sample covariance matrix $\hat{\mathbf{R}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{y}(n)\mathbf{y}^*(n)$, with N denoting the number of snapshots and $\mathbf{y}(n)$ representing the nth snapshot. As N increases, $\hat{\mathbf{R}}$ converges to $\mathbf{R}$, and the value of the corresponding SINR will approach $\mathrm{SINR}_{\mathrm{opt}}$ eventually. However, when $\hat{\mathbf{R}}$ contains samples from SOI (e.g., in mobile communications applications), the convergence rate of SCB can be very slow ($N \gg M$ is required). Consequently, the performance of SCB degrades substantially in the presence of small sample size problems, even when $\mathbf{a}_0$ is exactly known.

[*]The work was supported in part by the Office of Naval Research under Grant No. N00014-07-1-0193, by the National Science Foundation under Grant No. CCF-0634786, ECCS-0729727 and by the Swedish Research Council (VR).

Moreover, the mismatch between the true and assumed steering vectors ($\mathbf{a}_0$ and $\mathbf{a}$) can also significantly deteriorate the performance of SCB.

To improve the performance of SCB, we replace $\hat{\mathbf{R}}$ by an enhanced covariance matrix estimate based on a shrinkage-based method [4]. The enhanced estimate is obtained by linearly combining $\hat{\mathbf{R}}$ and a shrinkage target (a given matrix with some structure) in an optimal mean-squared error (MSE) sense, which can be done via both analytical and convex optimization approaches as shown in the next section.

3. SHRINKAGE-BASED COVARIANCE MATRIX ESTIMATION

A linear shrinkage estimate, which we refer to as the Convex Combination (CC), has the form:

$$\tilde{\mathbf{R}} = \alpha\mathbf{I} + (1-\alpha)\hat{\mathbf{R}}, \tag{2}$$

where α is the shrinkage intensity, $\tilde{\mathbf{R}}$ is an enhanced estimate of $\mathbf{R}$ and we use the most commonly employed shrinkage target - the identity matrix $\mathbf{I}$. We also consider a more general linear combination (GLC):

$$\tilde{\mathbf{R}} = \alpha\mathbf{I} + \beta\hat{\mathbf{R}}. \tag{3}$$

The shrinkage parameters for both CC and GLC can be chosen by minimizing (an estimate of) the MSE of the estimator $\tilde{\mathbf{R}}$ [4], where MSE$(\tilde{\mathbf{R}}) = E\{\|\tilde{\mathbf{R}} - \mathbf{R}\|^2\}$.

Note that the constraints $\alpha \in [0,1]$ for CC and $\alpha \geq 0$, $\beta \geq 0$ for GLC can be imposed to guarantee that $\tilde{\mathbf{R}} \geq 0$. Alternatively, we can impose $\tilde{\mathbf{R}} \geq 0$ directly for both CC and GLC. In the rest of the section, we first review the approaches in [5], where the constraints in the former case are used, and then we extend the approaches further by formulating the MSE minimization problems as convex optimization problems, where all the aforementioned constraints can be imposed.

3.1. Review of the Approaches in [5]

We consider the MSE minimization problem for GLC first.

$$\begin{aligned}\mathrm{MSE}(\tilde{\mathbf{R}}) \quad &= \|\alpha\mathbf{I} - (1-\beta)\mathbf{R}\|^2 + \beta^2 E\{\|\hat{\mathbf{R}} - \mathbf{R}\|^2\} \\ &= \alpha^2 M - 2\alpha(1-\beta)\,\mathrm{tr}(\mathbf{R}) \\ &\quad + (1-\beta)^2\|\mathbf{R}\|^2 + \beta^2 E\{\|\hat{\mathbf{R}} - \mathbf{R}\|^2\}. \end{aligned} \tag{4}$$

The optimal values for β and α can be readily obtained:

$$\beta_0 = \frac{\gamma}{\rho + \gamma}, \tag{5}$$

$$\alpha_0 = \nu(1-\beta_0) = \nu\frac{\rho}{\gamma + \rho}, \tag{6}$$

where $\rho = E\{\|\hat{\mathbf{R}} - \mathbf{R}\|^2\}$, $\nu = \frac{\mathrm{tr}(\mathbf{R})}{M}$, and $\gamma = \|\nu\mathbf{I} - \mathbf{R}\|^2$. We note that $\beta_0 \in [0,1]$ and $\alpha_0 \geq 0$.

To estimate α_0 and β_0 from the given data, we need an estimate of ρ, which can be calculated as (see [5] for details):

$$\hat{\rho} = \frac{1}{N^2}\sum_{n=1}^{N}\|\mathbf{y}(n)\|^4 - \frac{1}{N}\|\hat{\mathbf{R}}\|^2. \tag{7}$$

Consequently, we can get estimates for α_0 and β_0:

$$\hat{\beta}_0^{(1)} = \frac{\hat{\gamma}}{\hat{\gamma} + \hat{\rho}}, \tag{8}$$

and

$$\hat{\alpha}_0^{(1)} = \hat{\nu}(1 - \hat{\beta}_0^{(1)}), \tag{9}$$

where $\hat{\nu} = \frac{\mathrm{tr}(\hat{\mathbf{R}})}{M}$, and $\hat{\gamma} = \|\hat{\nu}\mathbf{I} - \hat{\mathbf{R}}\|^2$. Note that $\hat{\alpha}_0^{(1)}$ and $\hat{\beta}_0^{(1)}$ satisfy the constraints $\alpha \geq 0$ and $\beta \geq 0$. In addition, note that $\gamma + \rho = E\{\|\hat{\mathbf{R}} - \nu\mathbf{I}\|^2\}$, an estimate of which is given by $\|\hat{\mathbf{R}} - \hat{\nu}\mathbf{I}\|^2$. Then we can get alternative estimates of α_0 and β_0 (we need to guarantee that they are nonnegative):

$$\hat{\alpha}_0^{(2)} = \min\left[\hat{\nu}\frac{\hat{\rho}}{\|\hat{\mathbf{R}} - \hat{\nu}\mathbf{I}\|^2}, \hat{\nu}\right], \tag{10}$$

$$\hat{\beta}_0^{(2)} = 1 - \frac{\hat{\alpha}_0^{(2)}}{\hat{\nu}}. \tag{11}$$

We will refer to the GLC using (8)-(9) as GLC_1, and to the GLC using (10)-(11) as GLC_2.

Note that α_0 and β_0 can be rewritten as $\alpha_0 = \nu\tau_0$ and $\beta_0 = 1 - \tau_0$ ($\tau_0 = \frac{\rho}{\rho+\gamma}$), which implies that GLC reduces to CC when $\nu = 1$. Therefore, setting $\hat{\nu} = 1$, we can obtain $\hat{\alpha}_0^{(1)}$ from (9) and $\hat{\alpha}_0^{(2)}$ from (10) for CC, which we refer to as CC_1 and CC_2, respectively.

3.2. Extensions

Below the MSE minimization problems for GLC and CC are formulated as convex optimization problems.

Consider first the convex formulation of GLC. From (4),

$$\begin{aligned}\mathrm{MSE}(\tilde{\mathbf{R}}) \quad &= \alpha^2 M + 2\alpha\beta\,\mathrm{tr}(\mathbf{R}) + \beta^2\|\mathbf{R}\|^2 + \beta^2\rho \\ &\quad -2\alpha\,\mathrm{tr}(\mathbf{R}) - 2\beta\|\mathbf{R}\|^2 + \mathrm{const} \\ &= \boldsymbol{\theta}^T\mathbf{A}\boldsymbol{\theta} - 2\mathbf{b}^T\boldsymbol{\theta} + \mathrm{const}. \end{aligned} \tag{12}$$

where $\boldsymbol{\theta} = [\alpha \quad \beta]^T$, $\mathbf{b} = \left[\mathrm{tr}(\mathbf{R}) \quad \|\mathbf{R}\|^2\right]^T$, and

$$\mathbf{A} = \begin{bmatrix} M & \mathrm{tr}(\mathbf{R}) \\ \mathrm{tr}(\mathbf{R}) & \|\mathbf{R}\|^2 + \rho \end{bmatrix}. \tag{13}$$

Note that $\mathbf{A} > 0$ and hence, (12) has a unique (unconstrained) minimum solution given by:

$$\boldsymbol{\theta}_0 = [\alpha_0 \quad \beta_0]^T = \mathbf{A}^{-1}\mathbf{b}, \tag{14}$$

which is equivalent to the optimal solution in (5) and (6). Next, we rewrite (12) as:

$$\left[\boldsymbol{\theta} - \mathbf{A}^{-1}\mathbf{b}\right]^T\mathbf{A}\left[\boldsymbol{\theta} - \mathbf{A}^{-1}\mathbf{b}\right] + \mathrm{const}. \tag{15}$$

Then, the MSE minimization problem for GLC under the constraint $\tilde{\mathbf{R}} \geq 0$ can be formulated as the following Semidefinte Program (SDP):

$$\begin{aligned} \min_{\delta,\boldsymbol{\theta}} \quad & \delta \\ \text{subject to} \quad & \begin{bmatrix} \delta & \left[\boldsymbol{\theta} - \hat{\mathbf{A}}^{-1}\hat{\mathbf{b}}\right]^T \\ \left[\boldsymbol{\theta} - \hat{\mathbf{A}}^{-1}\hat{\mathbf{b}}\right] & \hat{\mathbf{A}}^{-1} \end{bmatrix} \geq 0 \\ & \tilde{\mathbf{R}}(\boldsymbol{\theta}) \geq 0. \end{aligned} \tag{16}$$

which can be solved in polynomial time using public domain software [6]. In addition, it is easy to obtain a convex optimization formulation for CC by adding the constraint:

$$\mathbf{u}^T\boldsymbol{\theta} = 1, \quad \mathbf{u} = [1 \quad 1]^T \tag{17}$$

to (16). The so-obtained problem is still a SDP.

Note that $\mathbf{A}$ and $\mathbf{b}$ are replaced by their estimates $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$ in (16). One way to obtain $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$ is to use $\hat{\rho}$ (7) and $\hat{\mathbf{R}}$, respectively, in lieu of ρ and $\mathbf{R}$ in $\mathbf{A}$ and $\mathbf{b}$. Then, we can obtain estimates $\hat{\alpha}_0^{(1')}$ and $\hat{\beta}_0^{(1')}$ of α_0 and β_0 by solving (16). We refer to this method as $\text{GLC}_{1'}$. Similarly, we can also obtain $\hat{\alpha}_0^{(1')}$ by adding (17) to (16) for CC, which we refer to as $\text{CC}_{1'}$. Note that $\text{GLC}_{1'}$ and $\text{CC}_{1'}$ can be readily shown to be equivalent to GLC_1 and CC_1, respectively. Indeed, the constraint $\tilde{\mathbf{R}} \geq 0$ is inactive due to the $\text{GLC}_{1'}$ solution satisfying $\hat{\alpha}_0^{(1')} \geq 0$ and $\hat{\beta}_0^{(1')} \geq 0$, and to the $\text{CC}_{1'}$ solution satisfying $\hat{\alpha}_0^{(1')} \in [0, 1]$, which guarantees that $\tilde{\mathbf{R}} \geq 0$.

Exactly as in CC_2 and GLC_2, we can also use alternative estimates of the unknown quantities in $\mathbf{A}$ and $\mathbf{b}$. Noting that $\rho + \|\mathbf{R}\|^2 = E\{\|\hat{\mathbf{R}}\|^2\}$, so we can estimate $\rho + \|\mathbf{R}\|^2$ in $\mathbf{A}$ by $\|\hat{\mathbf{R}}\|^2$, and estimate $\|\mathbf{R}\|^2$ in $\mathbf{b}$ by $\|\hat{\mathbf{R}}\|^2 - \hat{\rho}$. We also replace $\mathbf{R}$ by $\hat{\mathbf{R}}$ in $\text{tr}(\mathbf{R})$. Consequently, we can obtain estimates $\hat{\alpha}_0^{(3)}$ and $\hat{\beta}_0^{(3)}$ from (16) for GLC, which we refer to as GLC_3, and an estimate $\hat{\alpha}_0^{(3)}$ from (17) and (16) for CC, which we refer to as CC_3. GLC_3 and CC_3 are in general different from GLC_2 and CC_2, respectively, due to GLC_3 and CC_3 enforcing $\tilde{\mathbf{R}} \geq 0$ directly while minimizing (15) (with $\mathbf{A}$ and $\mathbf{b}$ replaced by $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$). GLC_2 and CC_2, on the other hand, minimize (15) (with $\mathbf{A}$ and $\mathbf{b}$ replaced by the same $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$) without imposing any constraints, and then clip the solutions to satisfy $\hat{\alpha}_0^{(2)} \geq 0$ and $\hat{\beta}_0^{(2)} \geq 0$ for GLC and $\hat{\alpha}_0^{(2)} \in [0, 1]$ for CC. Therefore, GLC_2 and CC_2 are suboptimal. The optimal version of GLC_2, which we refer to as GLC_4, can be obtained by using the constraints $\alpha \geq 0$ and $\beta \geq 0$ instead of $\tilde{\mathbf{R}}(\boldsymbol{\theta}) \geq 0$ in (16) and calculating $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$ in the same way as in GLC_3. We can similarly get CC_4, which is the optimal version of CC_2.

4. SHRINKAGE-BASED ROBUST CAPON BEAMFORMERS

We have 8 methods to obtain the enhanced estimates of the covariance matrix, i.e.,

$$\tilde{\mathbf{R}}_{\text{GLC}_i} = \hat{\alpha}_0^{(i)}\mathbf{I} + \hat{\beta}_0^{(i)}\hat{\mathbf{R}}, \quad i = 1, \cdots, 4, \tag{18}$$

and

$$\tilde{\mathbf{R}}_{\text{CC}_i} = \hat{\alpha}_0^{(i)}\mathbf{I} + (1 - \hat{\alpha}_0^{(i)})\hat{\mathbf{R}}, \quad i = 1, \cdots, 4. \tag{19}$$

Using one of the above enhanced estimates $\tilde{\mathbf{R}}$ in lieu of $\hat{\mathbf{R}}$ in the SCB formulation yields the shrinkage-based robust adaptive beamformer: $\tilde{\mathbf{w}} = \frac{\tilde{\mathbf{R}}^{-1}\mathbf{a}}{\mathbf{a}^*\tilde{\mathbf{R}}^{-1}\mathbf{a}}$. The resulting beamformer output SINR is given by $\text{SINR} = \frac{\sigma_0^2|\tilde{\mathbf{w}}^*\mathbf{a}_0|^2}{\tilde{\mathbf{w}}^*\mathbf{Q}\tilde{\mathbf{w}}}$, and the SOI power estimate is $\hat{\sigma}_0^2 = \tilde{\mathbf{w}}^*\tilde{\mathbf{R}}\tilde{\mathbf{w}}$.

From (18)-(19), we note that the shrinkage-based robust adaptive beamformers are diagonal loading approaches with the diagonal loading levels ($\hat{\alpha}_0/\hat{\beta}_0$ for GLC and $\hat{\alpha}_0/(1 - \hat{\alpha}_0)$ for CC) determined automatically from the observed data snapshots $\{\mathbf{y}(n)\}_{n=1}^N$.

5. NUMERICAL EXAMPLES

We present below several numerical examples comparing the performance of the shrinkage-based robust adaptive beamformers with that of HKB and SCB. Interestingly, in all of these examples, the GLC_2 solutions did not need clipping, and hence GLC_2, GLC_3, and GLC_4 were equivalent, in addition, GLC_1 performed similarly to GLC_2. The same was true for CC. Therefore, only the results obtained by GLC_2 and CC_2 will be presented, and we will refer to GLC_2 as GLC and CC_2 as CC for short. In all examples, we assume a uniform linear array with $M = 10$ sensors and half-wavelength interelement spacing. The noise is assumed to be white complex Gaussian random process with zero-mean and covariance matrix $\mathbf{I}$. A SOI with a 10 dB power is assumed to impinge on the array from $0°$, and two interferences, each with a 20 dB power, are assumed to be present at $10°$ and $60°$. For each scenario, 1000 Monte-Carlo trials are performed.

First, we examine the output SINR convergence performance of the beamformers. Fig. 1 shows the mean of the output SINRs versus the number of snapshots N when $\mathbf{a}_0$ is known. As shown in the figure, SCB converges to SINR_{opt} very slowly. Both GLC and CC outperform SCB. Whereas, unlike CC, GLC provides a significant improvement over SCB for all values of N considered. Figs. 2(a) and (b) show the mean values of the diagonal loading levels of GLC and CC, respectively, as a function of N. We observe that the diagonal loading level of CC is much lower than that of GLC. Another observation from Fig. 1 is that the output SINR of HKB decreases when N is beyond a certain number. Unlike GLC and CC, as shown in Fig. 2(c), the mean of the diagonal loading level for HKB starts from a very small value and monotonically and quickly increases with N. This behavior limits HKB's performance improvement over SCB when N is small and deteriorates its performance when N is large.

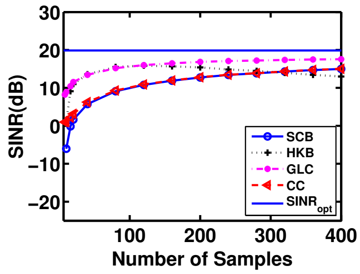
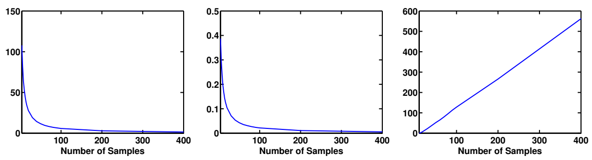
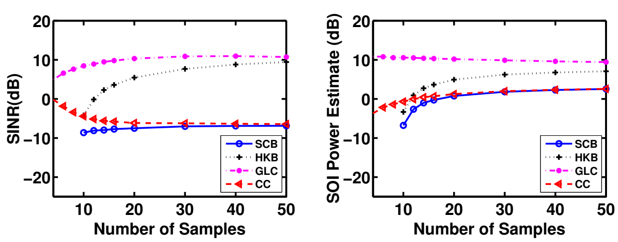


Fig. 1. Beamformer output SINR versus N.



(a) (b) (c)

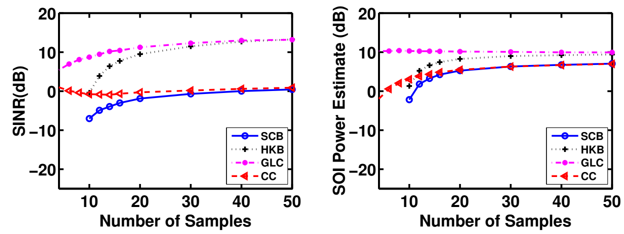
Fig. 2. Average diagonal loading levels versus N: (a) GLC (b) CC (c) HKB.



(a) (b)

Fig. 3. Performance comparison in the presence of a 2° steering angle error: (a) SINR, (b) SOI power estimates versus N.

Next, we examine the robustness of the beamformers to small sample size problems and to steering vector errors. Figs. 3 and 4, respectively, show the performance in the presence of look direction errors (2° SOI steering angle mismatch) and array calibration errors. The array calibration errors are simulated by perturbing each element of the array steering vector using independent zero-mean complex Gaussian random variables with variance 0.01. GLC shows the best performance, especially when N is small. Moreover, HKB and SCB are not applicable when $\hat{\mathbf{R}}$ is rank deficient ($N < M$), whereas, GLC and CC can be used. Note that when $N \gg M$, $\hat{\mathbf{R}}$ becomes very close to $\mathbf{R}$, and hence the shrinkage-based approaches will choose small diagonal loading levels. Then their ability to combat steering vector errors will diminish. Yet the shrinkage based methods are very useful for the case of small sample sizes. This case is often encountered in practice and is most critically in need of performance improvement.



(a) (b)

Fig. 4. Performance comparison in the presence of array calibration errors: (a) SINR, (b) SOI power estimates versus N.

6. CONCLUSIONS

We have presented several approaches to the fully automatic computation of diagonal loading levels. In our diagonal loading algorithms, the conventional sample covariance matrix used in the SCB formulation is replaced by an enhanced covariance matrix estimate based on shrinkage. We have shown how to efficiently obtain the shrinkage covariance matrix estimates from the available data. Several numerical examples have been used to compare the performance of the proposed beamformers with that of SCB and HKB. The shrinkage-based approaches improve the robustness of SCB against small sample size problems and steering vector errors, with GLC having the best performance among the methods tested. More importantly, we have demonstrated that GLC is very useful in the case of small sample sizes - the case in which the users of adaptive arrays are most interested.

7. REFERENCES

[1] H. Cox, R. Zeskind, and M. Owen, "Robust adaptive beamforming," *Proceedings of IEEE*, vol. ASSP-35, pp. 1365–1375, October 1987.

[2] J. Li and P. Stoica, eds., *Robust Adaptive Beamforming*. New York, NY: John Wiley & Sons, 2005.

[3] Y. Selen, R. Abrahamsson, and P. Stoica, "Automatic robust adaptive beamforming via ridge regression," to appear in *Signal Processing*, 2008.

[4] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *Journal of Multivariate Analysis*, vol. 88, pp. 365–411, 2004.

[5] P. Stoica, J. Li, X. Zhu, and J. R. Guerci, "On using a priori knowledge in space-time adaptive processing," to appear in *IEEE Transactions on Signal Processing*.

[6] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," Available: http://www2.unimaas.nl/ sturm/software/sedumi.html.