ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

Alex B. Gershman*, Ulrich Nickel†, & Johann F. Böhme*

*Signal Theory Department, Ruhr University, Bochum, Germany
†Electronics Department, FGAN-FFM, Wachtberg, Germany

ABSTRACT

The performance of adaptive beamforming algorithms is known to degrade in a moving jammer environment. This degradation occurs due to the jammer motion that brings the interfering sources out of the sharp notches of the adapted pattern. In this paper, we consider a unified framework that allows to make a broad class of adaptive array algorithms robust against jammer motion. The robustness is achieved by means of artificially broadening the directional pattern null width in the jammer directions. For this purpose, we use a special type of data-dependent sidelobe derivative constraints that do not require any *a priori* information about the jammers.

I. INTRODUCTION

The performance of adaptive beamforming has been discussed for stationary (non-moving) jammer scenarios, [1], [2]. However, in the future fast moving platforms become more likely in various applications. The performance of adaptive arrays severely degrades if the weights are not able to adapt sufficiently fast to the changing jamming situation. Fast adaptation has therefore be the aim of research in these cases. Moving jammers represent a serious problem, because for large antennas the directional pattern nulls are extremely sharp and jammers may soon move out of the nulls, i.e. high gain antennas are very sensitive to this type of non-stationarity.

Recently, a large number of robust adaptive beamforming methods has been studied [3], [4]. However, only the robustness against desired signal positioning errors was addressed. In this paper, another problem is considered. We assume that the desired signal direction is exactly known, whereas the relative angular motion of the jammers may be fast. In other words, we address another type of robust adaptive array – the robustness against possible jammer motion. We develop a framework allowing to incorporate this robustness property into various adaptive beamforming algorithms. The main idea is to broaden the width of the pattern nulls in the jammer directions. For this purpose, derivative constraints are used, which do not require any *a priori* information about the jammers. Similar constraints have been used in [5]-[7] for several particular problems.

In this paper, we demonstrate the strength and generality of the developed framework and formulate robust modifications of several algorithms, such as the Sample Matrix Inversion (SMI) [1], the diagonally loaded SMI (LSMI) [4], [9], the Hung-Turner (HT) [8], and the Eigenvector Projection (EP) [10] methods. Although we consider only these algorithms, the developed approach allows to incorporate the robustness property into any other type of adaptive beamformer.

II. DATA-DEPENDENT DERIVATIVE CONSTRAINTS

We formulate the problem for a linear array of n sensors taking into account that the approach considered is valid for planar or volume arrays, too. Let q ($q < n$) narrowband jammers and a single narrowband signal impinge on the array from unknown directions $\{\theta_1, \theta_2, \ldots, \theta_q\}$ and from a known direction θ_S, respectively. Let the jammers be uncorrelated with each other and with the signal, too. Unless otherwise specified, the jammer sources are assumed to be stationary (e.g. non-moving) and the array output vectors are assumed to be statistically independent. Then the

ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

output vector at the i-th time instant (or snapshot) of the array can be expressed as:

$$y(i) = As(i) + \beta s_S(i) a_S + n(i), \quad A = [a_1, a_2, \ldots, a_q] \tag{1}$$

where the jammer direction vectors $a_1, a_2, \ldots, a_q$, and desired signal vector a_S can be modelled as

$$a(\theta) = (\exp\{jx_1\xi\}, \exp\{jx_2\xi\}, \ldots, \exp\{jx_n\xi\})^T \tag{2}$$

Here $a(\theta)$ is the $n \times 1$ direction vector corresponding to the angle θ, $a_S = a(\theta_S)$, $\xi = (2\pi/\lambda)\sin\theta$, λ is the wavelength, x_l is the coordinate of lth sensor, $s(i) = (s_1(i), s_2(i), \ldots, s_q(i))^T$ is the $q \times 1$ vector of random jammer waveforms, and $s_S(i)$ is the signal waveform. The $n \times 1$ vector $n(i)$ contains random sensor noise, while parameter β (0 or 1) indicates the presence of the signal. The $n \times n$ interference-plus-noise covariance matrix is

$$R = \sigma^2 I + ASA^H \tag{3}$$

where σ^2 is the noise variance, $S = \mathrm{E}\{s(i)s^H(i)\}$ is the $q \times q$ covariance matrix of the jammer waveforms, I is the identity matrix, and $(\cdot)^H$ denotes Hermitian transpose.

The complex adaptive beamformer output with the adaptive weight vector w at the ith time instant is $z(i) = w(i)^H y(i)$. In the stationary case and if R is known, the solution for the optimum weight vector of the adaptive array maximizing the signal-to-interference-plus-noise ratio (SINR) can be expressed in the well known form [1], [2]:

$$w_{\mathrm{opt}} = \alpha R^{-1} a_S \tag{4}$$

where α is an arbitrary complex constant, which does not affect the output SINR and can be omitted for notational convenience. Typically, jammers are much more powerful than the sensor noises and the signal. We use this assumption below.

Define an orthogonal projection on the column space of a $m_1 \times m_2$ full rank matrix C and an orthogonal projection onto its orthogonal complement as

$$P_C = C\left(C^H C\right)^{-1} C^H, \quad P_C^\perp = I - P_C \tag{5}$$

respectively. Here $m_1 > m_2$, and so P_C and $P_C^\perp$ are $m_1 \times m_1$ matrices.

Applying the matrix inversion lemma [2] to the matrix $I + \alpha CHC^H$, where H is any $m_2 \times m_2$ nonsingular matrix and $\alpha > 0$, gives the property $P_C^\perp = \lim_{\alpha\to\infty}\left(I + \alpha CHC^H\right)^{-1}$. Applying it to (3), we get $\lim_{\sigma^2\to 0} \sigma^2 R^{-1} = P_A^\perp = I - P_A$ where P_A is the orthogonal projection onto the jammer subspace. This implies that for strong jammers the adaptive weight vector (4) tends to be orthogonal to the jammer subspace, i.e.: $\lim_{\sigma^2\to 0} w_{\mathrm{opt}} \perp \{a_1, a_2, \ldots, a_q\}$. This means that the antenna pattern $w^H a(\theta)$ has nulls in the jammer directions. To broaden the null width we may require a higher order of the null, i.e. by requiring p-th order derivative constraints:

$$\partial^m (w^H a(\theta))/\partial \xi^m |_{\theta=\theta_k} = 0, \quad k = 1, 2, \ldots, q, \quad m = 1, 2, \ldots, p \tag{6}$$

Using (2), we can rewrite (6) as

$$w \perp \{B^m a_i\}_{m=1,2,\ldots,p;\ i=1,2,\ldots,q}, \quad B = \mathrm{diag}\{x_1, x_2, \ldots, x_n\} \tag{7}$$

ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

To fulfill the constraints, we assume $n > (p+1)q$.

In practice, neither $\boldsymbol{R}$, nor the jammer directions are known. The only available is the sequence of array snapshots $\boldsymbol{y}(i)$. However, for high interference-to-noise ratios (INR) and small signal a projection orthogonal to the array data is the same as a projection orthogonal to the jammer subspace. I.e., we obtain from (1) the asymptotical relation between a stationary ergodic snapshot sequence and the jammer subspace: $\lim_{\sigma^2\to 0, p_S\to 0} \boldsymbol{P_Y} = \boldsymbol{P_A}$, where p_S denotes the signal power and $\boldsymbol{Y}$ is the $n \times L$ data matrix consisting of $L \geq q$ statistically independent snapshot vectors $\boldsymbol{y}(i)$ in the columns [1] (as in eq. (9) below). With this property we can reformulate the constraints $\boldsymbol{w} \perp \{\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_q\}$ and $\boldsymbol{w} \perp \{\boldsymbol{B}^m \boldsymbol{a}_i\}$, $m = 1, 2, \dots, p$, $i = 1, 2, \dots, q$ in an asymptotically ($\sigma^2 \to 0$, $p_S \to 0$) equivalent form:

$$\boldsymbol{w} \perp \boldsymbol{y}(i), \quad \boldsymbol{w} \perp \{\boldsymbol{B}^m \boldsymbol{y}(i)\}_{m=1,2,\dots,p}, \quad \forall i \tag{8}$$

Equation (8) describes data-dependent derivative constraints that do not require any *a priori* knowledge of the jammer directions. For incorporation these constraints into any adaptive beamforming algorithm, one should use new "derivative" snapshots $\boldsymbol{B}^m \boldsymbol{y}(i)$, $m = 1, 2, \dots, p$ in addition to the conventional data snapshots $\boldsymbol{y}(i)$ for the calculation of the weight vector.

III. ROBUST ALGORITHMS

A. SMI Algorithm

The SMI algorithm [1] computes the weight vector using the sample covariance matrix $\hat{\boldsymbol{R}}$:

$$\boldsymbol{w}(i) = \hat{\boldsymbol{R}}^{-1}(k)\boldsymbol{a}_S, \quad \hat{\boldsymbol{R}}(k) = \frac{1}{L}\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H, \quad \boldsymbol{Y}(k) = [\boldsymbol{y}(k-L), \boldsymbol{y}(k-L+1), \dots, \boldsymbol{y}(k-1)] \tag{9}$$

The matrix $\hat{\boldsymbol{R}}(k)$ is invertible only if $L \geq n$ and $\sigma^2 > 0$. The relationship between the adaptation window and the beamforming snapshot, i.e. the choice of the parameter $k-i$, depends on the specific application of the algorithm and has of course a significant influence in non-stationary situations.

Let us consider asymptotic property of the weight vector (9) for strong jammers. In the signal absent case and $L \geq q$ the sample covariance matrix can be written as $\hat{\boldsymbol{R}} = \boldsymbol{A}\boldsymbol{F}\boldsymbol{A}^H + \sigma^2 \boldsymbol{N}$, where $\boldsymbol{F}$ is the full rank $q \times q$ jammer sample covariance matrix and $\sigma^2 \boldsymbol{N}$ is the $n \times n$ matrix containing the noise sample covariance matrix and jammer-noise sample covariance cross-terms. Applying the matrix inversion lemma, we have $\lim_{\sigma^2\to 0, p_S\to 0} \sigma^2 \hat{\boldsymbol{R}}^{-1} = \boldsymbol{N}^{-1} - \boldsymbol{N}^{-1}\boldsymbol{A}(\boldsymbol{A}^H \boldsymbol{N}^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^H \boldsymbol{N}^{-1}$. This is the oblique projection with respect to the Euclidean scalar product. For large number of snapshots all jammer-noise cross-terms vanish and $\lim_{L\to\infty} \boldsymbol{N} = \boldsymbol{I}$. This means that $\lim_{L\to\infty, \sigma^2\to 0, p_S\to 0} \sigma^2 \hat{\boldsymbol{R}}^{-1} = \boldsymbol{P}_{\boldsymbol{A}}^{\perp}$. Hence, the weight vector of the SMI method converges to the orthogonal projection of the desired signal vector onto the orthogonal complement of the jammer subspace. Therefore, the incorporation of new "derivative" snapshots into the sample covariance matrix means that the weight vector converges to the projection of the desired signal vector onto the orthogonal complement of the subspace spanned by vectors $\{\boldsymbol{B}^m \boldsymbol{a}_i\}$, $m = 0, 1, \dots, p$, $i = 1, 2, \dots, q$. Although any order of constraints is possible, we consider only first-order constraints (i.e. $p = 1$). The robust version of SMI algorithm can then be expressed as:

$$\boldsymbol{w}(i) = \tilde{\boldsymbol{R}}^{-1}(k)\boldsymbol{a}_S, \quad \tilde{\boldsymbol{R}}(k) = \hat{\boldsymbol{R}}(k) + \zeta\, \boldsymbol{B}\hat{\boldsymbol{R}}(k)\boldsymbol{B} = \frac{1}{L}\left(\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H + \zeta\, \boldsymbol{B}\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H \boldsymbol{B}\right) \tag{10}$$

[1] In the case $q < L$ and $\sigma^2 = 0$, $p_S = 0$ the matrix $\boldsymbol{Y}$ becomes rank defficient. In this case, the projection $\boldsymbol{P_Y}$ should not be calculated by (5), but correspondingly with q linear independent columns of $\boldsymbol{Y}$.

ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

The real positive weight ζ controls the relative contribution of the "derivative" data.

B. LSMI Algorithm

In the LSMI algorithm a small real positive number γ is added to the diagonal of the sample covariance matrix, [4], [9]. This diagonal load warrant the sample covariance matrix invertibility in the case $L < n$, and reduces the variance of the adaptive weight vector. This allows to consider cases $q \leq L < n$, without too serious performance loss and makes the LSMI method well suited to non-stationary jammer situations, where the weights should follow the non-stationary jammers. The diagonally loaded sample covariance matrix is given by

$$\hat{\boldsymbol{R}}_{\mathrm{DL}}(k) = \gamma \boldsymbol{I} + \frac{1}{L}\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H \tag{11}$$

It can be shown that for $L \geq q$, $\lim_{\gamma \to 0, \sigma^2 \to 0, p_S \to 0} \gamma(\gamma \boldsymbol{I} + \hat{\boldsymbol{R}})^{-1} = \boldsymbol{P}^{\perp}_{\boldsymbol{A}}$. Hence, the incorporation of new "derivative" snapshots into the matrix (11) corresponds asymptotically to constraining the derivatives of the adapted pattern. As in the modified SMI method, the weight vector will converge to the projection of the desired signal vector onto the orthogonal complement of the subspace spanned by vectors $\{\boldsymbol{B}^m \boldsymbol{a}_i\}$, $m = 0, 1, \ldots, p$, $i = 1, 2, \ldots, q$. However, the large number of snapshots condition is no longer necessary for the convergence to the orthogonal projection onto the orthogonal compliment to the jammer subspace. By this fact one can expect a better performance of the LSMI method for small/moderate number of snapshots than of the SMI method.

Following these considerations, we formulate the robust modification of the LSMI algorithm:

$$\boldsymbol{w}(i) = \tilde{\boldsymbol{R}}_{\mathrm{DL}}^{-1}(k)\boldsymbol{a}_S, \quad \tilde{\boldsymbol{R}}_{\mathrm{DL}}(k) = \hat{\boldsymbol{R}}_{\mathrm{DL}}(k) + \zeta\, \boldsymbol{B}\hat{\boldsymbol{R}}(k)\boldsymbol{B} = \gamma \boldsymbol{I} + \frac{1}{L}\left(\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H + \zeta\, \boldsymbol{B}\boldsymbol{Y}(k)\boldsymbol{Y}(k)^H \boldsymbol{B}\right) \tag{12}$$

C. Hung-Turner Algorithm

In this method, the property given by the 1st equation of (8) is used directly, so that

$$\boldsymbol{w}(i) = \boldsymbol{P}^{\perp}_{\boldsymbol{Y}(k)}\boldsymbol{a}_S \tag{13}$$

The number of snapshots L is here also the dimension of the subspace on the complement of which is projected and should be chosen as $q \leq L < n$. For finite INR it is useful to choose $L > q$. If L is chosen too large, the jammers are well suppressed, but the gain of the antenna in the desired signal direction may be small. It was found in [11] that for large arrays $L \simeq 2q \ldots 3q$ gives a reasonable performance. The robust version [7] of the HT algorithm using "derivative" snapshots up to the pth order can be written as:

$$\boldsymbol{w}(i) = \boldsymbol{P}^{\perp}_{\boldsymbol{Q}(k)}\boldsymbol{a}_S, \quad \boldsymbol{Q}(k) = [\boldsymbol{Y}(k), \boldsymbol{B}\boldsymbol{Y}(k), \ldots, \boldsymbol{B}^p\boldsymbol{Y}(k)] \tag{14}$$

For a non-trivial projection $(p+1)L < n$. The robust algorithm (14) requires additional degrees of freedom as compared with (13). The choice of the optimum number of snapshots L is therefore linked to the order of the derivative constraints. For (14) no weight ζ is necessary.

D. Eigenvector Projection Algorithm

The eigendecomposition of $\hat{\boldsymbol{R}}(k)$ can be written as $\sum_{i=1}^{n} \lambda_i(k)\hat{\boldsymbol{u}}_i(k)\hat{\boldsymbol{u}}_i(k)^H$, where $\lambda_i(k)$, $i = 1, 2, \ldots, n$ $(\lambda_1(k) > \lambda_2(k) > \cdots > \lambda_n(k))$ are the ordered sample eigenvalues (distinct w. p. 1), $\hat{\boldsymbol{u}}_i(k)$ is the sample eigenvector that corresponds to $\lambda_i(k)$.

ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

The weight vector in EP algorithm is calculated as the orthogonal projection of the desired signal vector onto the orthogonal complement to the subspace spanned by M dominant eigenvectors:

$$\boldsymbol{w}(i) = \boldsymbol{P}^{\perp}_{\boldsymbol{U}(k)}\boldsymbol{a}_S, \quad \boldsymbol{U}(k) = [\hat{\boldsymbol{u}}_1(k), \hat{\boldsymbol{u}}_2(k), \ldots, \hat{\boldsymbol{u}}_M(k)] \tag{15}$$

In moving jammer case it is useful to choose $M > q$. From the analogy to the HT algorithm, the robust modification of the EP algorithm (15) can be formulated as:

$$\boldsymbol{w}(i) = \boldsymbol{P}^{\perp}_{\boldsymbol{V}(k)}\boldsymbol{a}_S, \quad \boldsymbol{V}(k) = [\boldsymbol{U}(k), \boldsymbol{B}\boldsymbol{U}(k), \ldots, \boldsymbol{B}^p\boldsymbol{U}(k)] \tag{16}$$

As the robust HT algorithm, this algorithm requires the additional degrees of freedom and the choice of the optimum M is linked to the order p of the derivative constraints.

IV. CHOOSING OF THE DERIVATIVE CONSTRAINT WEIGHT

Let us discuss the choice of the parameter ζ in robust SMI and LSMI methods. In terms of the jammer subspace, the derivative constraints have the effect of additional jammers. If the contribution of the "derivative" data vectors is too strong as compared with the contribution of the original data, the depth of the nulls is not sufficient and, as a result, the jamming power is not sufficiently suppressed. Conversely, when the contribution of the original data is much stronger than that of "derivative" data, the desired null width may be not sufficient. The choice of the parameter ζ in (10), (12) is therefore very important for the optimization of the adaptive array performance. Let us find a value of ζ from the compromise between null depth and width of the adapted pattern. With respect to moving jammer scenarios the robustness should improve with increasing ζ. However, one cannot expect a better result than in a stationary jammer situation. Therefore, the maximal value of the parameter ζ should be limited from above by a maximally admissible loss in SINR in the stationary case, i.e. ζ should be chosen as a highest value satisfying $\mathrm{SINR}_{\mathrm{opt}}/\mathrm{SINR}_{\mathrm{rob}} \leq \Pi$. The loss $\Pi \geq 1$ is the price we accept to pay for the improved robustness. $\mathrm{SINR}_{\mathrm{opt}}$ is the SINR with the weight (4), while $\mathrm{SINR}_{\mathrm{rob}}$ corresponds to $\boldsymbol{w}_{\mathrm{rob}} = (\boldsymbol{R} + \zeta\,\boldsymbol{B}\boldsymbol{R}\boldsymbol{B})^{-1}\,\boldsymbol{a}_S$. The SINR for an arbitrary weight vector $\boldsymbol{w}$ is defined as $\mathrm{SINR} = p_S|\boldsymbol{w}^H\boldsymbol{a}_S|^2/\boldsymbol{w}^H\boldsymbol{R}^{-1}\boldsymbol{w}$. From the matrix inversion lemma [2]

$$\boldsymbol{w}_{\mathrm{rob}} = (\boldsymbol{R}^{-1} - \boldsymbol{R}^{-1}\boldsymbol{B}(\zeta^{-1}\boldsymbol{R}^{-1} + \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B})^{-1}\boldsymbol{B}\boldsymbol{R}^{-1})\boldsymbol{a}_S \tag{17}$$

From (17) we see that the optimum choice of ζ depends on unknown jammer parameters. However, with the following assumptions we can estimate a value of ζ independent of the jammer parameters:

1. $\boldsymbol{R}^{-1} \simeq \boldsymbol{P}^{\perp}_{\boldsymbol{A}}$. This means that we have strong jammers and that we assume (without loss of generality) that the sensor noise power is normalized to one ($\sigma^2 = 1$).

2. $\boldsymbol{P}^{\perp}_{\boldsymbol{A}}\boldsymbol{a}_S \simeq \boldsymbol{a}_S$. This means that the jammers impinge on the array from sidelobe directions and that we have a low sidelobe level, i.e. $|\boldsymbol{a}_S^H\boldsymbol{a}_i|^2 \ll \boldsymbol{a}_S^H\boldsymbol{a}_S\,\boldsymbol{a}_i^H\boldsymbol{a}_i = n^2$, for $i = 1, 2, \ldots, q$. This is given, if $n \gg q$ and then $||\boldsymbol{P}_{\boldsymbol{A}}\boldsymbol{a}_S|| \ll ||\boldsymbol{a}_S||$, hence $\boldsymbol{P}^{\perp}_{\boldsymbol{A}}\boldsymbol{a}_S \simeq \boldsymbol{a}_S$.

3. $\boldsymbol{P}^{\perp}_{\boldsymbol{A}}\boldsymbol{B} \simeq \boldsymbol{B}$. This condition can be approximately fulfilled for large, centered arrays ($n \gg q$, $\sum x_l = 0$). Then the vectors $\boldsymbol{a}_i$ and $\boldsymbol{B}\boldsymbol{a}_i$ are exactly orthogonal for all i. Further, if the jammers are well separated, such that $|\boldsymbol{a}_i^H\boldsymbol{B}\boldsymbol{a}_k|^2 \ll \boldsymbol{a}_i^H\boldsymbol{a}_i\,\boldsymbol{a}_k^H\boldsymbol{B}^2\boldsymbol{a}_k = n\sum x_l^2$, for $i, k = 1, \ldots, q$, then this means that $\boldsymbol{A}^H\boldsymbol{B}\boldsymbol{A} \simeq \boldsymbol{O}$, from which $\boldsymbol{P}^{\perp}_{\boldsymbol{A}}\boldsymbol{B} \simeq \boldsymbol{B}$ follows.

Under these assumptions we can rewrite $\mathrm{SINR}_{\mathrm{opt}}/\mathrm{SINR}_{\mathrm{rob}}$ using (17) as:

$$\frac{\boldsymbol{a}_S^H\boldsymbol{a}_S\left[\boldsymbol{a}_S^H\boldsymbol{a}_S - 2\boldsymbol{a}_S^H\boldsymbol{B}\left(\frac{1}{\zeta}\boldsymbol{I} + \boldsymbol{B}^2\right)^{-1}\boldsymbol{B}\boldsymbol{a}_S + \boldsymbol{a}_S^H\boldsymbol{B}\left(\frac{1}{\zeta}\boldsymbol{I} + \boldsymbol{B}^2\right)^{-1}\boldsymbol{B}^2\left(\frac{1}{\zeta}\boldsymbol{I} + \boldsymbol{B}^2\right)^{-1}\boldsymbol{B}\boldsymbol{a}_S\right]}{\left(\boldsymbol{a}_S^H\boldsymbol{a}_S - \boldsymbol{a}_S^H\boldsymbol{B}\left(\frac{1}{\zeta}\boldsymbol{I} + \boldsymbol{B}^2\right)^{-1}\boldsymbol{B}\boldsymbol{a}_S\right)^2} \leq \Pi \tag{18}$$

ADAPTIVE BEAMFORMING ROBUST AGAINST MOVING JAMMERS

For a given array antenna, a suitable value of ζ can now be found from (18) which depends on the array geometry only. Assume centered array and normalize $\boldsymbol{B}$, such that $||\boldsymbol{Ba}|| = ||\boldsymbol{a}||$ for any direction vector $\boldsymbol{a}$. This means that instead of x_l we have to use x_l/ρ with $\rho = \sqrt{\sum_{l=1}^{n} x_l^2/n}$. In the case of an equispaced centered array with even number of sensors

$$\boldsymbol{B} = \frac{1}{\rho}\,\mathrm{diag}\{-(n-1)/2,\ldots,-1/2,\,1/2,\ldots,(n-1)/2\},\quad \rho = \left(\frac{2}{n}\sum_{l=1}^{n/2}(l-1/2)^2\right)^{1/2} = \sqrt{(n^2-1)/12} \tag{19}$$

Then, the left-hand side of (18) can be simplified to

$$n\left(n - 4\zeta\sum_{l=1}^{n/2}\frac{(l-1/2)^2}{\rho^2+\zeta(l-1/2)^2} + 2\zeta^2\sum_{l=1}^{n/2}\frac{(l-1/2)^4}{(\rho^2+\zeta(l-1/2)^2)^2}\right)\Bigg/\left(n - 2\zeta\sum_{l=1}^{n/2}\frac{(l-1/2)^2}{\rho^2+\zeta(l-1/2)^2}\right)^2$$

This expression is plotted as a function of ζ in Fig. 1 for different numbers of array sensors $n = 8, 16, 32, 64, 128$. One observes that all curves are monotone and merge into one line and are practically independent of n.

V. SIMULATION RESULTS

The effectiveness of the robust algorithms depends of course on the jammer trajectories and on the sampling scheme, in particular on the time delay between the adaptation window (for calculating $\boldsymbol{w}(i)$) and the data snapshots for beamforming. In simulations we assumed three moving narrowband jammers and a uniform linear array of 32 omnidirectional sensors at $\lambda/2$ spacing. The simulated trajectories of angular jammer motion are shown in Fig. 2. The array was centered and the "derivative" data were normalized as in (19). The additive Gaussian sensor noise was assumed to be uncorrelated and to have $\sigma^2 = 1$ in each sensor. The jammers were assumed to be mutually uncorrelated and uncorrelated with the signal, too. The jammer-to-noise ratio was set to 30 dB for each jammer and the signal-to-noise ratio was set to -7.5 dB, in each sensor, respectively. Two sampling schemes were considered. In the first mode we assumed an adaptation period without the desired signal present, i.e. a learning period with $\beta = 0$ in (1), followed by the beamforming snapshot (i.e. $k - i = 0$) with the desired signal present. This scheme is representative for an active system. In the second mode, we assumed that the signal is always present in the data snapshots, i.e., $\beta = 1$ in (1). The snapshots for adaptation are taken from a sliding window of given length. To minimize time delays between adaptation and beamforming, we assumed that beamforming is done with the snapshot in the middle of adaptation window, i.e. $k - i = L/2$ is taken. This scheme is representative for passive systems. In all examples we assumed $\zeta = 1.5$ ($\simeq 1$ dB performance loss in the stationary case, see Fig. 1). The parameter γ taken in the LSMI algorithm was $\gamma = 2\sigma^2$. The order of the derivative constraints was always $p = 1$.

Figs. 3a and 3b show the output SINR of the SMI algorithms for 32 snapshots for the 1st and 2nd mode, respectively. Figs. 4a and 4b show the corresponding curves for $L = 64$. With 64 snapshots the difference between the two SMI versions becomes less pronounced. Actually, in 64 snapshots case sometimes the full periodical movement of the jammers is within the adaptation window (see Fig. 2) and L satisfies the condition $L \geq 2n$ necessary for well jammer suppression in stationary case [1]. Adding up data from different jammer positions in the conventional SMI method also results in a broader null. However, the robust method performs significantly better. Figs. 5a and 5b demonstrate the output SINR versus time for the LSMI algorithms with $L = 10$ for the first and

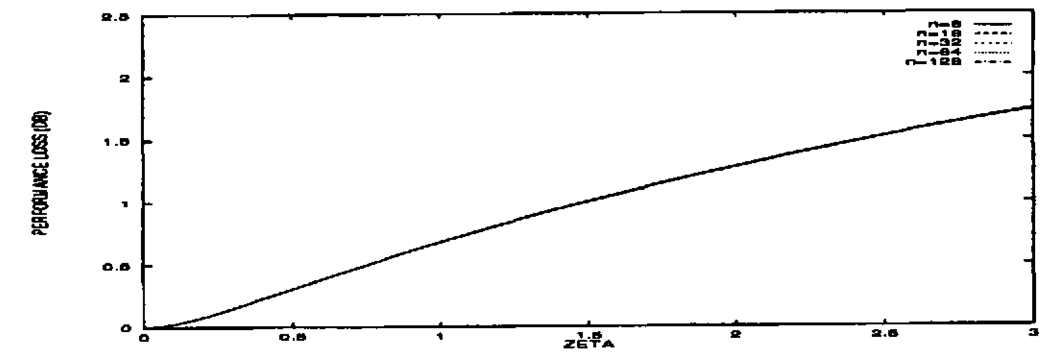

Figure 1: Performance loss for derivative constraint weight ζ.

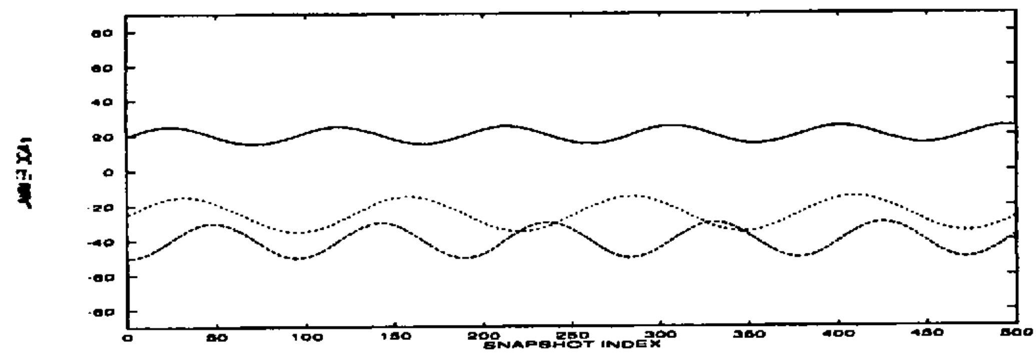

Figure 2: Jammer motions.

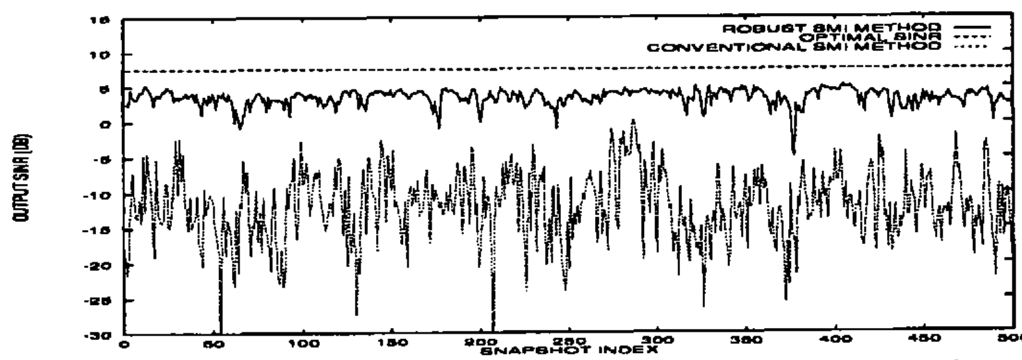

Figure 3a: Performance of SMI methods with 32 snapshots. Mode 1.

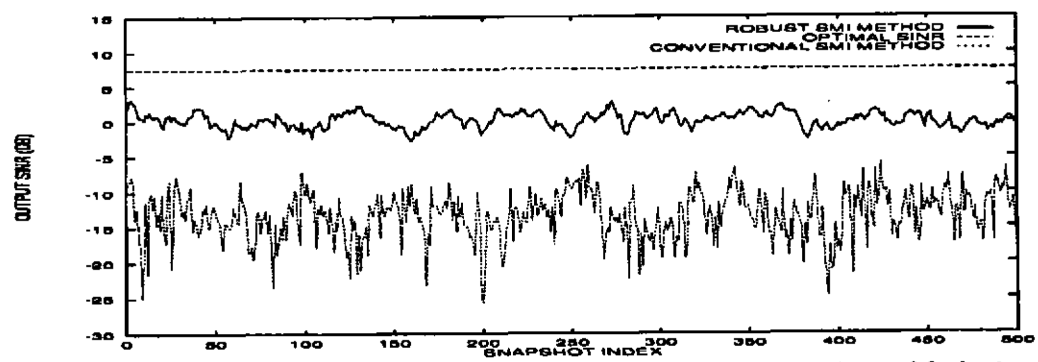

Figure 3b: Performance of SMI methods with 32 snapshots. Mode 2.

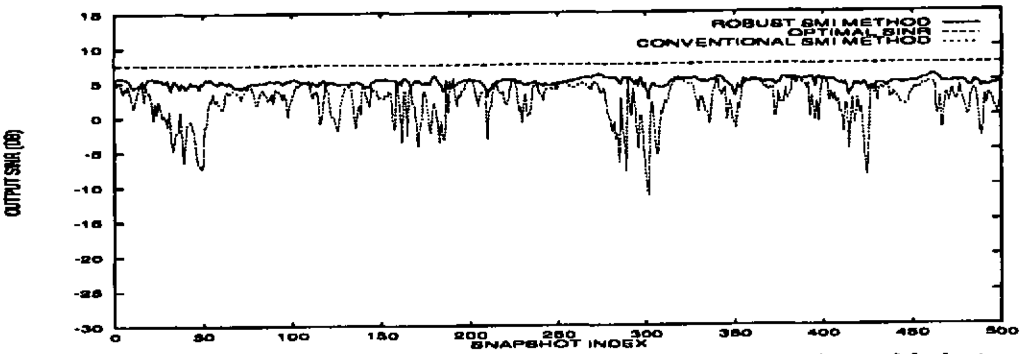

Figure 4a: Performance of SMI methods with 64 snapshots. Mode 1.

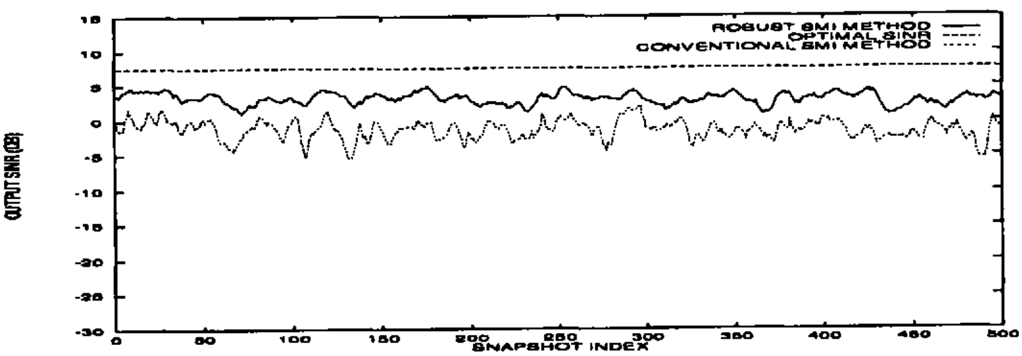

Figure 4b: Performance of SMI methods with 64 snapshots. Mode 2.

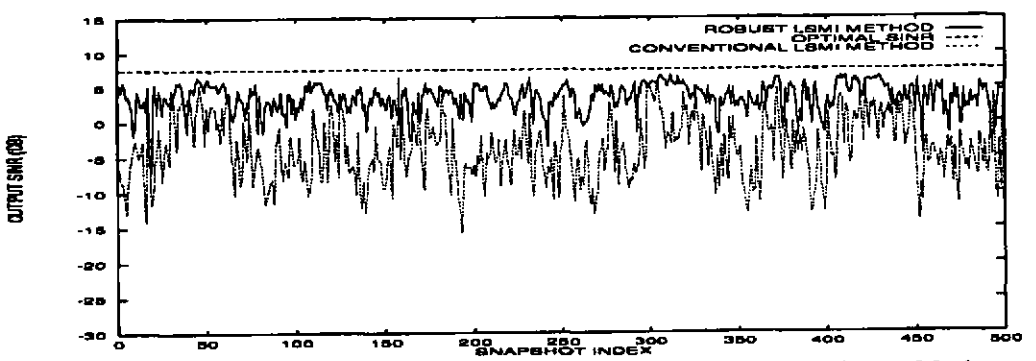

Figure 5a: Performance of LSMI methods with 10 snapshots. Mode 1.

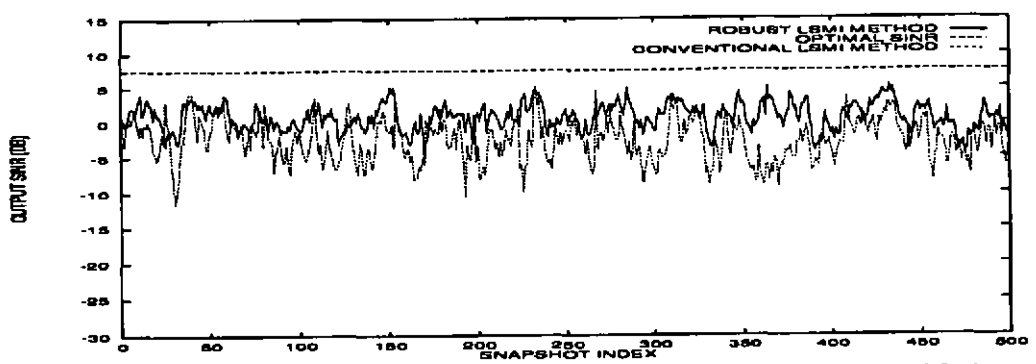

Figure 5b: Performance of LSMI methods with 10 snapshots. Mode 2.

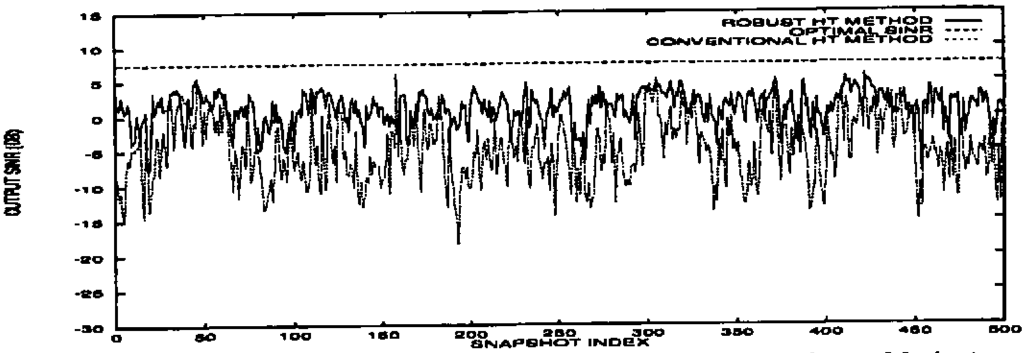

Figure 6a: Performance of HT methods with 10 snapshots. Mode 1.

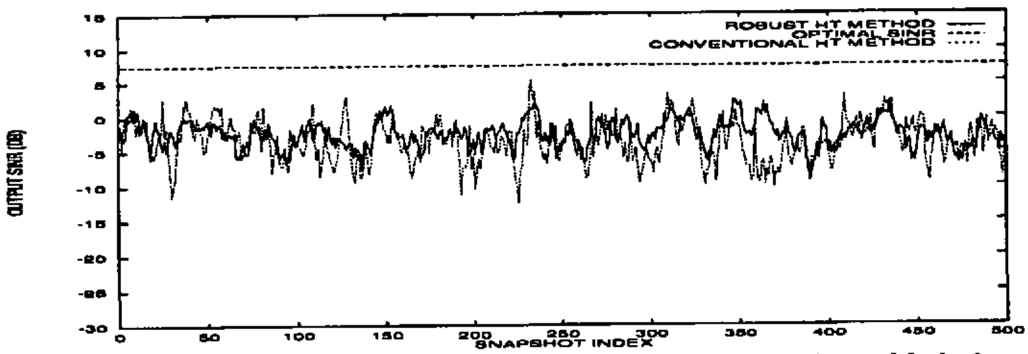

Figure 6b: Performance of HT methods with 10 snapshots. Mode 2.

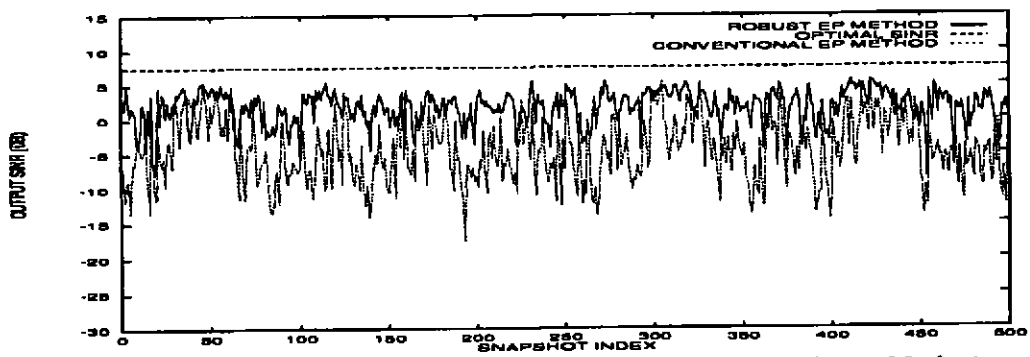

Figure 7a: Performance of EP methods with 10 snapshots. Mode 1.

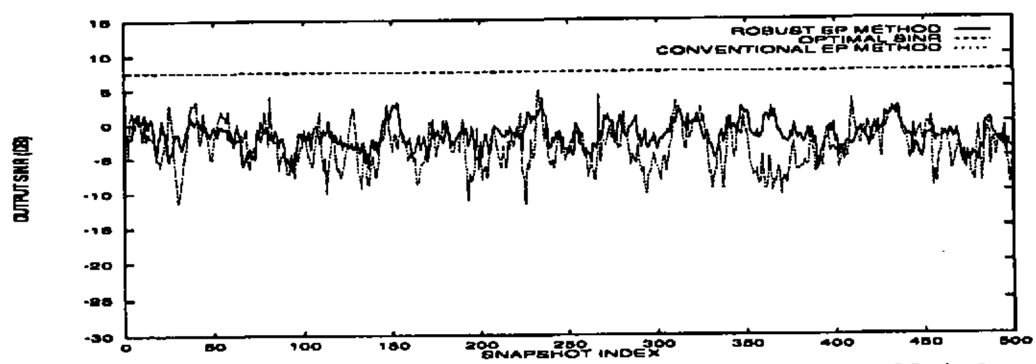

Figure 7b: Performance of EP methods with 10 snapshots. Mode 2.

second mode, respectively. Due to the short data window, the LSMI algorithm can better follow the non-stationary jammer scenario. Still, the robust LSMI version gives a significant improvement. The performance of the HT methods is shown in Figs. 6a and 6b for $L = 10$ for the first and second mode, respectively. The robust version of the HT method performs slightly worse than the robust LSMI, because the dimension of the complement space that is used for matching to the signal direction is only 12. Finally, the performance of the EP methods is demonstrated in Figs. 7a and 7b for $L = 10$, $M = 3q = 9$ for the first and second mode, respectively. We see that the robust EP method provides a significant improvement in both modes.

These simulation results verify that the proposed modified algorithms perform significantly better than the standard techniques. For the first mode the performance improvement is more pronounced because of the longer time delay between adaptation and beamforming windows. The presence of the signal in the second mode typically causes some SINR degradation for all algorithms [1], [2]. This is a known effect and techniques to counter it are available.

VI. CONCLUSION

In this paper, we considered a general approach allowing to incorporate robustness against angular jammer motion into any adaptive beamforming algorithm. This robustness is achieved by means of artificially broadening the adaptive array nulls using special data-dependent sidelobe derivative constraints which do not require any *a priori* information about the jammers. We have shown how to determine the constraint weight, such that the loss in stationary situations is tolerable. As an example, the approach was applied to the SMI method, the LSMI algorithm, the HT algorithm, and the EP technique. The results of computer simulations verified that the proposed robust adaptive algorithms perform significantly better than the standard algorithms in moving jammer scenarios.

REFERENCES

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

[2] J.E. Hudson, *Adaptive Array Principles*, New York: Peter Peregrinus, Stevenage, UK, 1981.

[3] N.K. Jablon, "Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections," *IEEE Trans. Antennas and Propagat.*, Aug. 1986.

[4] H. Cox, R.M. Zeskind, and M.H. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 1365-1376, Oct. 1987.

[5] K. Takao and K. Komiyama, "An adaptive antenna for rejection of wideband interference," *IEEE Trans. Aerospace Electron. Syst.*, vol. AES-16, pp. 452-459, July 1980.

[6] A.B. Gershman, and V.T. Ermolaev, "Synthesis of the weight distribution of an adaptive array with wide dips in the directional pattern," *Radiophys. & Quant. Electr.*, vol.34, pp.720-724,1991.

[7] A.B. Gershman, G.V. Serebryakov, and J.F. Böhme, "Adaptive beamforming projection-type algorithm robust to wideband and moving jammers in narrowband array," to appear in *Proc. Int. Conf. Neural Networks & Signal Processing*, China, 1995.

[8] E.K.L. Hung, and R.M. Turner, "A fast beamforming algorithm for large arrays," *IEEE Trans. Aerospace, Electron. Syst.*, vol. AES-19, pp. 598-607, July 1983.

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

[10] H. Subbaram, K. Abend, "Interference Suppression via Orthogonal Projections: A Performance Analysis", *IEEE Trans. Antennas and Prop.*, Vol. 41, No.9, pp. 1187-1194, Sept. 1993.

[11] U. Nickel, "Some properties of fast projection methods of the Hung–Turner type," in *Signal Processing III: Theories and Applications*, I.T. Young et al, eds, pp.1165-1168, 1986.