ROBUST CAPON BEAMFORMING BY THE ADAPTIVE PROJECTED SUBGRADIENT METHOD

Konstantinos Slavakis, Masahiro Yukawa, and Isao Yamada

Dept. of Communications and Integrated Systems (S3-60),
Tokyo Institute of Technology, Tokyo 152-8552, JAPAN.
Emails: {slavakis,masahiro,isao}@comm.ss.titech.ac.jp .

ABSTRACT

It is well-known that the Capon beamformer is sensitive to array steering vector errors and may result into a worse performance than classical data-independent beamformers. This paper follows a different path from the well-established Diagonal Loading techniques and designs a robust Capon beamformer by a recent extension of the Adaptive Projected Subgradient Method. The proposed method marks a computational complexity of $O(N^2)$, where N is the number of array elements. The simulation results show that the proposed beamformer achieves excellent performance especially in cases where the Diagonal Loading techniques face difficulties, i.e. in cases where the Interference to Noise Ratio (INR) is moderately larger than SNR.

1. CAPON BEAMFORMING

We will denote the set of all integers, nonnegative and positive integers, real, and complex numbers by $\mathbb{Z}$, $\mathbb{Z}_{\geq 0}$ and $\mathbb{Z}_{>0}$, $\mathbb{R}$, and $\mathbb{C}$ respectively. For any complex number $z \in \mathbb{C}$, we let $\overline{z}$ denote its conjugate. Let also $i := \sqrt{-1}$.

We will consider the *Uniform Linear Array (ULA)* of Fig. 1. The received signal $\boldsymbol{y}(k) := [y_1(k), \ldots, y_N(k)]^t \in \mathbb{C}^N$, $\forall k \in \mathbb{Z}_{\geq 0}$, is a discrete-time complex vector random process indexed on $\mathbb{Z}_{\geq 0}$: $\boldsymbol{y}(k) := q_0(k)\boldsymbol{s}_0 + \sum_{j=1}^{J} q_j(k)\boldsymbol{s}_j + \boldsymbol{n}(k)$, $\forall k \in \mathbb{Z}_{\geq 0}$, where the complex scalar random processes $(q_0(k))_{k\in\mathbb{Z}_{\geq 0}}$ and $(q_j(k))_{k\in\mathbb{Z}_{\geq 0}}$, $j = 1, \ldots, J$, contain information of the *Signal Of Interest (SOI)* and of the J jammers respectively. The *correlation matrix* $\boldsymbol{R}_{\boldsymbol{y}}(k) := \mathsf{E}\{\boldsymbol{y}(k)\boldsymbol{y}^*(k)\}$, $k \in \mathbb{Z}_{\geq 0}$, where $\mathsf{E}\{\cdot\}$ denotes expectation, and the superscript $*$ stands for complex conjugate transposition. The noise process $(\boldsymbol{n}(k))_{k\in\mathbb{Z}_{\geq 0}} \subset \mathbb{C}^N$ is a complex vector i.i.d. Gaussian process with $\mathsf{E}\{\boldsymbol{n}(k)\} = \boldsymbol{0}$ and $\boldsymbol{R}_{\boldsymbol{n}} := \mathsf{E}\{\boldsymbol{n}(k)\boldsymbol{n}^*(k)\}$, $\forall k \in \mathbb{Z}_{\geq 0}$. The *steering vector* associated with the planar wave of wavelength λ arriving from the far-zone field of the ULA with an angle $\theta \in [0, \pi]$, called *Direction Of Arrival (DOA)*, is defined as $\boldsymbol{s} := [1, e^{2\pi i \frac{d}{\lambda}\cos\theta}, \ldots, e^{2\pi i (N-1)\frac{d}{\lambda}\cos\theta}]^t \in \mathbb{C}^N$, where d is the interelement distance of the ULA, and the superscript t denotes transposition. The steering vector corresponding to SOI will be denoted by $\boldsymbol{s}_0$ while $\boldsymbol{s}_j$ is mapped to the j-th jammer, $j = 1, \ldots, J$.

The *Capon Beamformer (CB)* [1, 2] is defined as the solution to the following linearly constrained minimization problem:

$$\text{find } \boldsymbol{w} \in \arg\min_{\boldsymbol{z}\in\mathbb{C}^N,\ \boldsymbol{s}_0^*\boldsymbol{z}=1} \boldsymbol{z}^*\boldsymbol{R}_{\boldsymbol{y}}(k)\boldsymbol{z},\ k \in \mathbb{Z}_{\geq 0}. \quad (1)$$

Then, $\boldsymbol{w}_{\text{CB}}(k) = \frac{\boldsymbol{R}_{\boldsymbol{y}}(k)^{-1}\boldsymbol{s}_0}{\boldsymbol{s}_0^*\boldsymbol{R}_{\boldsymbol{y}}(k)^{-1}\boldsymbol{s}_0}$, $\forall k \in \mathbb{Z}_{\geq 0}$, and

$$\mathsf{E}\{|\boldsymbol{y}^*(k)\boldsymbol{w}_{\text{CB}}(k)|^2\} = \frac{1}{\boldsymbol{s}_0^*\boldsymbol{R}_{\boldsymbol{y}}(k)^{-1}\boldsymbol{s}_0},\ \forall k \in \mathbb{Z}_{\geq 0}. \quad (2)$$

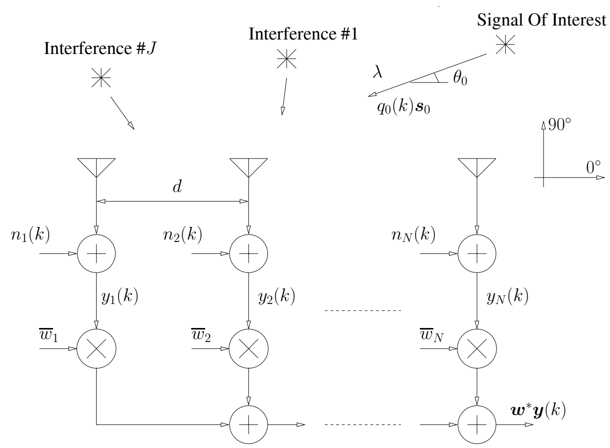

Fig. 1. The narrowband beamformer for a Uniform Linear Array (ULA) of $N \in \mathbb{Z}_{>0}$ elements. The array weighting vector $\boldsymbol{w} := [w_1, \ldots, w_N]^t \in \mathbb{C}^N$ will be adaptively selected.

Array processing methods are susceptible to a wide range of errors like DOA mismatches, poor array calibration, unknown sensor mutual coupling, near-far wavefront mismodeling, signal wavefront distortions, source spreading, and coherent/incoherent local scattering [3]. These model mismatches cause a perturbation $\widetilde{\boldsymbol{s}}_j$ of the actual steering vector $\boldsymbol{s}_j$, $j = 0, \ldots, J$, and thus affect the performance of the beamformer [2]. It has been observed that CB becomes sensitive to such sources of errors and may result into a worse performance than that of a standard data-independent beamformer [3–5].

The most widely used approach to robust adaptive beamforming is the *Diagonal Loading (DL)* technique [2]. To remedy model mismatches, a regularized version of the original Capon beamforming problem is formed by adding an additional quadratic constraint on the array weighting vector. This is equivalent to diagonally loading the correlation matrix [2, §6.6.4]: given $\epsilon_{\text{DL}} > 0$, calculate

$$\boldsymbol{w}_{\text{CB-DL}}(k) = \frac{(\boldsymbol{R}_{\boldsymbol{y}}(k) + \epsilon_{\text{DL}}\boldsymbol{I}_N)^{-1}\widetilde{\boldsymbol{s}}_0}{\widetilde{\boldsymbol{s}}_0^*(\boldsymbol{R}_{\boldsymbol{y}}(k) + \epsilon_{\text{DL}}\boldsymbol{I}_N)^{-1}\widetilde{\boldsymbol{s}}_0},\ \forall k \in \mathbb{Z}_{\geq 0}, \quad (3)$$

where $\boldsymbol{I}_N$ stands for the identity matrix in $\mathbb{C}^{N\times N}$.

Recently, an extensive amount of excellent research has been done on refining the DL approach for the robust adaptive CB problem [4, 5] by devising iterative methods for calculating an optimal (in some sense) DL parameter ϵ_{DL}. Unlike the empirical choice of ϵ_{DL} in the classical (3), these methods make explicit use of an uncertainty set of the array steering vector in order to compute an optimal DL parameter. The study in [5] uses the very successful *Second-Order Cone Programming (SOCP)* approach while the study in [4] exploits the *Lagrange multiplier methodology* after an eigendecomposition of $\boldsymbol{R_y}(k)$, $k \in \mathbb{Z}_{\geq 0}$, is performed.

Due to lack of space, the proofs of various propositions appearing in this paper are omitted. The full discussion will be presented in [6]. A small part of this study can be found in [7].

2. MATHEMATICAL PRELIMINARIES

In this paper we deal with the finite dimensional spaces $\mathbb{R}^m$ and $\mathbb{C}^m$, for $\exists m \in \mathbb{Z}_{>0}$, which are special cases of a Hilbert space $\mathcal{H}$. In $\mathbb{R}^m$, the inner product is defined as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^t \boldsymbol{y}, \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^m$. In $\mathbb{C}^m$, $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^* \boldsymbol{y}, \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{C}^m$. Henceforth, we shall denote by $\|\cdot\|$ the norm for both $\mathbb{R}^m$ and $\mathbb{C}^m$. For any $\boldsymbol{z} \in \mathbb{C}^m$, we let $\Re\{\boldsymbol{z}\}$ and $\Im\{\boldsymbol{z}\}$ stand for its real and imaginary part respectively.

Given an $\boldsymbol{x} \in \mathcal{H}$ and an $\epsilon > 0$, we define the *open ball* $B(\boldsymbol{x}, \epsilon) := \{\boldsymbol{y} \in \mathcal{H} : \|\boldsymbol{x} - \boldsymbol{y}\| < \epsilon\}$. Also, let the *closed ball* $B[\boldsymbol{x}, \epsilon] := \{\boldsymbol{y} \in \mathcal{H} : \|\boldsymbol{x} - \boldsymbol{y}\| \leq \epsilon\}$. Given a nonempty subset S of $\mathcal{H}$, an $\boldsymbol{x} \in S$, and a sufficiently small $\epsilon > 0$, assume that $B_S(\boldsymbol{x}, \epsilon) := B(\boldsymbol{x}, \epsilon) \cap S \neq \emptyset$. The *relative interior* of a nonempty $A \subset \mathcal{H}$ with respect to (w.r.t.) S is defined as $\mathrm{ri}_S(A) := \{\boldsymbol{x} \in A : \exists \epsilon > 0$ such that (s.t.) $B_S(\boldsymbol{x}, \epsilon) \subset A\}$. We let $\mathrm{int}(A) := \mathrm{ri}_{\mathcal{H}}(A)$ denote the *interior* of A.

A set $C \subset \mathcal{H}$ is called *convex* if $\forall \boldsymbol{x}, \boldsymbol{y} \in C$ and $\forall \mu \in [0, 1]$, $\mu \boldsymbol{x} + (1 - \mu) \boldsymbol{y} \in C$. A function $\Theta : C \to \mathbb{R} \cup \{\infty\}$ is called *convex* if $\forall \boldsymbol{x}, \boldsymbol{y} \in C$ and $\forall \mu \in [0, 1]$, $\Theta(\mu \boldsymbol{x} + (1 - \mu) \boldsymbol{y}) \leq \mu \Theta(\boldsymbol{x}) + (1 - \mu) \Theta(\boldsymbol{y})$. Given a nonempty $S \subset \mathcal{H}$ and an $\boldsymbol{x} \in \mathcal{H}$, define the function $d(\boldsymbol{x}, S) := \inf\{\|\boldsymbol{x} - \boldsymbol{y}\| : \boldsymbol{y} \in S\}$. For any nonempty closed convex set $C \subset \mathcal{H}$, the *metric projection onto* C is the mapping $P_C : \mathcal{H} \to C$ which maps $\boldsymbol{x} \in \mathcal{H}$ to the uniquely existing $P_C(\boldsymbol{x}) \in C$ s.t. $\|\boldsymbol{x} - P_C(\boldsymbol{x})\| = d(\boldsymbol{x}, C)$. Next are a few examples of closed convex sets together with their metric projection mappings.

Given an $\boldsymbol{x}_0 \in \mathcal{H}$ and an $\epsilon > 0$, the metric projection mapping onto the closed ball $B[\boldsymbol{x}_0, \epsilon]$ is given simply by $P_{B[\boldsymbol{x}_0, \epsilon]}(\boldsymbol{x}) = \boldsymbol{x}$, if $\boldsymbol{x} \in B[\boldsymbol{x}_0, \epsilon]$, and $P_{B[\boldsymbol{x}_0, \epsilon]}(\boldsymbol{x}) = \boldsymbol{x}_0 + \frac{\epsilon}{\|\boldsymbol{x} - \boldsymbol{x}_0\|}(\boldsymbol{x} - \boldsymbol{x}_0)$, if $\boldsymbol{x} \notin B[\boldsymbol{x}_0, \epsilon]$. Given $\boldsymbol{a} \neq \boldsymbol{0}$ in a real Hilbert space and $\beta \in \mathbb{R}$, the closed convex sets $\Pi := \{\boldsymbol{y} \in \mathcal{H} : \langle \boldsymbol{a}, \boldsymbol{y} \rangle = \beta\}$ and $\Pi^- := \{\boldsymbol{y} \in \mathcal{H} : \langle \boldsymbol{a}, \boldsymbol{y} \rangle \leq \beta\}$ are called *hyperplane* and *halfspace* respectively. The metric projection mappings onto Π and Π^- are given by simple closed forms [8]; $\forall \boldsymbol{x} \in \mathcal{H}$, $P_\Pi(\boldsymbol{x}) = \boldsymbol{x} - \frac{\langle \boldsymbol{a}, \boldsymbol{x} \rangle - \beta}{\|\boldsymbol{a}\|^2} \boldsymbol{a}$, and $P_{\Pi^-}(\boldsymbol{x}) = \boldsymbol{x} - \frac{(\langle \boldsymbol{a}, \boldsymbol{x} \rangle - \beta)^+}{\|\boldsymbol{a}\|^2} \boldsymbol{a}$, where $\alpha^+ := \max\{\alpha, 0\}$, $\forall \alpha \in \mathbb{R}$. An *icecream cone* is the closed convex set defined as $K := \{(\boldsymbol{x}, r) \in \mathcal{H} \times \mathbb{R} : \|\boldsymbol{x}\| \leq r\}$. Its metric projection mapping is given as [8]: $\forall (\boldsymbol{x}, r) \in \mathcal{H} \times \mathbb{R}$,

$$P_K(\boldsymbol{x}, r) = \begin{cases} (\boldsymbol{x}, r), & \|\boldsymbol{x}\| \leq r, \\ (\boldsymbol{0}, 0), & \|\boldsymbol{x}\| \leq -r, \\ \frac{\|\boldsymbol{x}\| + r}{2}(\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}, 1), & \text{otherwise.} \end{cases}$$

3. PROPOSED ALGORITHM

We assume the knowledge of the erroneous steering vector $\widetilde{\boldsymbol{s}}_0$, and the radius $\delta_0 > 0$ of an uncertainty set $B[\widetilde{\boldsymbol{s}}_0, \delta_0]$ to which the actual $\boldsymbol{s}_0$ most likely belongs to.

Instead of the actual $\boldsymbol{R_y}(k)$, which for the sake of simplicity we consider here to be $\boldsymbol{R_y} := \mathsf{E}\{\boldsymbol{y}(k)\boldsymbol{y}^*(k)\}$, $\forall k \in \mathbb{Z}_{\geq 0}$, we have used the following estimate for the calculations:

$$\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k) := \frac{1}{k+1} \sum_{l=0}^{k} \boldsymbol{y}(l)\boldsymbol{y}^*(l) + \frac{\epsilon_{\text{APSM}}}{k+1} \boldsymbol{I}_N, \ \forall k \in \mathbb{Z}_{\geq 0}, \quad (4)$$

where $\epsilon_{\text{APSM}} := 0.1$. The above estimate is asymptotically unbiased, i.e. $\lim_{k\to\infty} \mathsf{E}\{\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)\} = \boldsymbol{R_y}$. The *matrix inversion lemma* [2] implies that $\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(0)^{-1} = \frac{1}{\epsilon_{\text{APSM}}}(\boldsymbol{I}_N - \frac{\boldsymbol{y}(0)\boldsymbol{y}^*(0)}{\epsilon_{\text{APSM}} + \|\boldsymbol{y}(0)\|^2})$ and $\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1} = \frac{k+1}{k} \widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k-1)^{-1}(\boldsymbol{I}_N - \frac{\boldsymbol{y}(k)\boldsymbol{y}^*(k)\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k-1)^{-1}}{k + \boldsymbol{y}^*(k)\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k-1)^{-1}\boldsymbol{y}(k)})$, $k \in \mathbb{Z}_{>0}$.

The following algorithm generates a sequence $(\boldsymbol{u}_k)_{k \in \mathbb{Z}_{\geq 0}} \subset \mathbb{R}^{2N+3}$ by using a recent extension [9, 10] of the *Adaptive Projected Subgradient Method (APSM)* [11] over the fixed point set of strongly attracting nonexpansive mappings in infinite dimensional real Hilbert spaces (see Step 4). The APSM [9–11] addresses the convexly constrained asymptotic minimization problem of certain nonnegative convex continuous functions in an infinite dimensional real Hilbert space. APSM includes many existing projection based adaptive filtering methods like the classical *NLMS* or the *Affine Projection Algorithm (APA)* and it has been showing superior results for real world applications [9–11].

Step 0. Let $k = 0$, and arbitrarily choose a $\boldsymbol{u}_0 \in \mathbb{R}^{2N+3}$.

Step 1. *(Estimates of the SOI power)* The motivation for this subprocess comes from (2). An upper bound of the value in (2) is sought based on the uncertainty set $B[\widetilde{\boldsymbol{s}}_0, \delta_0]$. Define first the bijection $\phi : \mathbb{C}^N \to \mathbb{R}^{2N} : \boldsymbol{s} \mapsto \begin{bmatrix} \Re(\boldsymbol{s}) \\ \Im(\boldsymbol{s}) \end{bmatrix}$. Let $\widetilde{\boldsymbol{v}}_0 := \phi(\widetilde{\boldsymbol{s}}_0)$. Define also $\widetilde{\mathfrak{R}}_{\boldsymbol{y}}(k)^{-1} := \begin{bmatrix} \Re(\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1}) & -\Im(\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1}) \\ \Im(\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1}) & \Re(\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1}) \end{bmatrix} \in \mathbb{R}^{2N \times 2N}$. Choose the number of iterations $N_1(k) \in \mathbb{Z}_{>0}$, an arbitrary $\boldsymbol{v}_0^{(k)}$ and compute for $n = 0, \ldots, N_1(k) - 1$,

$$\boldsymbol{v}_{n+1}^{(k)} := T_{\widetilde{\boldsymbol{v}}_0}(\boldsymbol{v}_n^{(k)}) - 2\kappa_{n+1}^{(k)} \widetilde{\mathfrak{R}}_{\boldsymbol{y}}(k)^{-1} T_{\widetilde{\boldsymbol{v}}_0}(\boldsymbol{v}_n^{(k)}), \quad (5)$$

where $T_{\widetilde{\boldsymbol{v}}_0} := P_{B[\boldsymbol{0}, \sqrt{N}]} P_{\Pi^-_{\widetilde{\boldsymbol{v}}_0}}$, $P_{B[\boldsymbol{0}, \sqrt{N}]}$ is the metric projection mapping onto $B[\boldsymbol{0}, \sqrt{N}] \subset \mathbb{R}^{2N}$ (recall that $\|\boldsymbol{s}_0\| = \sqrt{N}$), and $P_{\Pi^-_{\widetilde{\boldsymbol{v}}_0}}$ is the metric projection mapping onto the halfspace $\Pi^-_{\widetilde{\boldsymbol{v}}_0} := \{\boldsymbol{v} \in \mathbb{R}^{2N} : \langle -\widetilde{\boldsymbol{v}}_0, \boldsymbol{v} \rangle \leq -(N - \frac{\delta_0^2}{2})\}$. If $\kappa_n^{(k)} := 1/n, \forall n \in \mathbb{Z}_{>0}$, then the iterative procedure in (5) is a special case of the *Hybrid Steepest Descent Method (HSDM)* [12] and produces a sequence $(\boldsymbol{v}_n^{(k)})_{n \in \mathbb{Z}_{\geq 0}}$ that (strongly) converges to the (unique) minimizer of the function $\boldsymbol{v}^t \widetilde{\mathfrak{R}}_{\boldsymbol{y}}(k)^{-1} \boldsymbol{v}$, $\boldsymbol{v} \in \mathbb{R}^{2N}$, over the set $B[\boldsymbol{0}, \sqrt{N}] \cap P_{\Pi^-_{\widetilde{\boldsymbol{v}}_0}}$. The HSDM addresses more general convexly constrained minimization problems and allows a wider variety of cost functions, weights, and mappings in infinite dimensional real Hilbert spaces [12]. To visualize the constraint set $B[\boldsymbol{0}, \sqrt{N}] \cap P_{\Pi^-_{\widetilde{\boldsymbol{v}}_0}}$, note that it is the closure of the *convex hull* of the nonconvex set $S := \{\boldsymbol{v} \in \mathbb{R}^{2N} : \|\boldsymbol{v} - \widetilde{\boldsymbol{v}}_0\| \leq \delta_0, \|\boldsymbol{v}\| = \sqrt{N}\}$, i.e. the closure of the smallest convex set containing S.

Now, compute $\widetilde{\boldsymbol{s}}_0^{(k)} := \sqrt{N} \frac{\phi^{-1}(\boldsymbol{v}_{N_1(k)}^{(k)})}{\|\boldsymbol{v}_{N_1(k)}^{(k)}\|}$, and obtain as an estimate of the SOI power $\widetilde{\sigma}^2_{\text{SOI}}(k) := (\widetilde{\boldsymbol{s}}_0^{(k)*} \widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)^{-1} \widetilde{\boldsymbol{s}}_0^{(k)})^{-1}$.

Since $\widetilde{\boldsymbol{R}}_{\boldsymbol{y}}(k)$ is asymptotically unbiased, we let here $N_1(k) := 1$, $\boldsymbol{v}_0^{(k)} := \boldsymbol{v}_{N_1(k-1)}^{(k-1)}$ ($\boldsymbol{v}_{N_1(-1)}^{(-1)} := \widetilde{\boldsymbol{v}}_0$), and $\kappa_{n+1} := 1/k$.

Step 2. *(Data-independent robust beamformer)* Given the estimate $\widetilde{\boldsymbol{s}}_0^{(k)}$ and some δ_k s.t. $0 < \delta_k < \delta_0$, we seek now for an array weighting vector $\boldsymbol{w} \in \mathbb{C}^N$ s.t. $\boldsymbol{s}^* \boldsymbol{w} \in [1 - \epsilon, 1 + \epsilon] + i[-\epsilon, \epsilon]$, $\forall \boldsymbol{s} \in B[\widetilde{\boldsymbol{s}}_0^{(k)}, \delta_k]$, where $\epsilon \geq 0$.

Define the bijection $\psi : \mathbb{C}^N \to \mathbb{R}^{2N\times 2} : \boldsymbol{s} \mapsto \begin{bmatrix} \Re(\boldsymbol{s}) & -\Im(\boldsymbol{s}) \\ \Im(\boldsymbol{s}) & \Re(\boldsymbol{s}) \end{bmatrix}$. Notice now that $\begin{bmatrix} \Re(\boldsymbol{s}^*\boldsymbol{w}) \\ \Im(\boldsymbol{s}^*\boldsymbol{w}) \end{bmatrix} = \psi(\boldsymbol{s})^t \begin{bmatrix} \Re(\boldsymbol{w}) \\ \Im(\boldsymbol{w}) \end{bmatrix}$. If we now let $\widetilde{\boldsymbol{A}}_k := \psi(\widetilde{\boldsymbol{s}}_0^{(k)}) =: [\widetilde{\boldsymbol{a}}_1^{(k)}, \widetilde{\boldsymbol{a}}_2^{(k)}]$, where $\widetilde{\boldsymbol{a}}_1^{(k)}, \widetilde{\boldsymbol{a}}_2^{(k)} \in \mathbb{R}^{2N}$, then we consider the following problem: find $\boldsymbol{x} \in \mathbb{R}^{2N}$ s.t.

$$-\boldsymbol{a}^t\boldsymbol{x} \le \epsilon - 1, \quad \forall \boldsymbol{a} \in B[\widetilde{\boldsymbol{a}}_1^{(k)}, \delta_k], \tag{6a}$$

$$\boldsymbol{a}^t\boldsymbol{x} \le \epsilon, \quad \forall \boldsymbol{a} \in B[\widetilde{\boldsymbol{a}}_2^{(k)}, \delta_k], \tag{6b}$$

$$-\boldsymbol{a}^t\boldsymbol{x} \le \epsilon, \quad \forall \boldsymbol{a} \in B[\widetilde{\boldsymbol{a}}_2^{(k)}, \delta_k]. \tag{6c}$$

The constraints $\boldsymbol{a}^t\boldsymbol{x} \le 1 + \epsilon, \forall \boldsymbol{a} \in B[\widetilde{\boldsymbol{a}}_1^{(k)}, \delta_k]$ were omitted in order to obtain better convergence results for a small number of antenna elements (see Section 4). For a compact notation of (6), we sort the above inequalities in the order of appearance. Then, the problem becomes: find $\boldsymbol{x} \in \mathbb{R}^{2N}$ s.t. $\boldsymbol{h}^t\boldsymbol{x} - \beta_\nu \le 0$, $\forall \boldsymbol{h} \in B[\widetilde{\boldsymbol{h}}_\nu, \gamma_\nu]$, $\nu = 0, 1, 2$, where $\widetilde{\boldsymbol{h}}_0 := -\widetilde{\boldsymbol{a}}_1^{(k)}$, $\widetilde{\boldsymbol{h}}_1 := \widetilde{\boldsymbol{a}}_2^{(k)}$, $\widetilde{\boldsymbol{h}}_2 := -\widetilde{\boldsymbol{a}}_2^{(k)}$, $\beta_0 := \epsilon - 1$, $\beta_1 := \beta_2 := \epsilon$, and $\gamma_\nu := \delta_k$, $\nu = 0, 1, 2$.

Lemma 1 *Given $m \in \mathbb{Z}_{>0}$, $\widetilde{\boldsymbol{h}} \in \mathbb{R}^m$, $\beta \in \mathbb{R}$, and $\gamma > 0$, then $\boldsymbol{x} \in \left\{\boldsymbol{y} \in \mathbb{R}^m : \boldsymbol{h}^t\boldsymbol{y} - \beta \le 0,\ \forall \boldsymbol{h} \in B[\widetilde{\boldsymbol{h}}, \gamma]\right\}$ iff $\left[\begin{smallmatrix} \boldsymbol{x} \\ r \end{smallmatrix}\right] \in K \cap \Pi \subset \mathbb{R}^{m+1}$, where $r \in \mathbb{R}$. The set $K := \left\{\left[\begin{smallmatrix} \boldsymbol{y} \\ \tau \end{smallmatrix}\right] \in \mathbb{R}^{m+1} : \|\boldsymbol{y}\| \le \tau\right\}$ is an icecream cone, and $\Pi := \left\{\left[\begin{smallmatrix} \boldsymbol{y} \\ \tau \end{smallmatrix}\right] \in \mathbb{R}^{m+1} : [\widetilde{\boldsymbol{h}}^t, \gamma]\left[\begin{smallmatrix} \boldsymbol{y} \\ \tau \end{smallmatrix}\right] = \beta\right\}$ is a hyperplane.*

Define, now, $\boldsymbol{\tau} := [\tau_0, \tau_1, \tau_2]^t \in \mathbb{R}^3$. Define for $\nu = 0, 1, 2$, $\Xi_\nu := \{\boldsymbol{y} \in \mathbb{R}^{2N} : \boldsymbol{h}^t\boldsymbol{y} - \beta_\nu \le 0,\ \forall \boldsymbol{h} \in B[\widetilde{\boldsymbol{h}}_\nu, \gamma_\nu]\}$, the icecream cone $K_\nu := \left\{\left[\begin{smallmatrix} \boldsymbol{y} \\ \boldsymbol{\tau} \end{smallmatrix}\right] \in \mathbb{R}^{2N+3} : \|\boldsymbol{y}\| \le \tau_\nu\right\}$, and the hyperplane $\Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)}) := \{\left[\begin{smallmatrix} \boldsymbol{y} \\ \boldsymbol{\tau} \end{smallmatrix}\right] : [\widetilde{\boldsymbol{h}}_\nu^t, 0, \ldots, 0, \gamma_\nu, 0, \ldots, 0]\left[\begin{smallmatrix} \boldsymbol{y} \\ \boldsymbol{\tau} \end{smallmatrix}\right] = \beta_\nu\}$, where γ_ν is placed at the $(2N + \nu + 1)$-th position. Then, Lemma 1 suggests that for any $\nu = 0, 1, 2$, $\boldsymbol{x} \in \Xi_\nu$ iff $\breve{\boldsymbol{x}} := \left[\begin{smallmatrix} \boldsymbol{x} \\ \boldsymbol{r} \end{smallmatrix}\right] \in K_\nu \cap \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)}) \subset \mathbb{R}^{2N+3}$, where $\boldsymbol{r} := [r_0, r_1, r_2]^t \in \mathbb{R}^3$. Therefore, the following problem will be considered: find $\breve{\boldsymbol{x}} := \left[\begin{smallmatrix} \boldsymbol{x} \\ \boldsymbol{r} \end{smallmatrix}\right] \in \bigcap_{\nu=0}^2 \left(K_\nu \cap \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})\right) \subset \mathbb{R}^{2N+3}$.

Step 3. *(Data-dependent closed convex sets: Stochastic Property Sets)* Choose $\epsilon_k \ge 0$ and let $\rho_k := \widetilde{\sigma}_{\mathrm{SOI}}^2(k) + \epsilon_k$. Define $\boldsymbol{Y}(k) := \psi(\boldsymbol{y}(k)) \in \mathbb{R}^{2N\times 2}$. Define also the convex function

$$F_k(\breve{\boldsymbol{x}}) := \left\|\left[\begin{smallmatrix} \boldsymbol{Y}(k) \\ \boldsymbol{0} \end{smallmatrix}\right]^t \breve{\boldsymbol{x}}\right\|^2 - \rho_k,\ \forall \breve{\boldsymbol{x}} \in \mathbb{R}^{2N+3}.$$

A *stochastic property set* is defined as the closed convex set $C_k := \{\breve{\boldsymbol{x}} : F_k(\breve{\boldsymbol{x}}) \le 0\}$. The function F_k is differentiable with differential $F_k'(\breve{\boldsymbol{x}}) = 2\left[\begin{smallmatrix} \boldsymbol{Y}(k) \\ \boldsymbol{0} \end{smallmatrix}\right]\left[\begin{smallmatrix} \boldsymbol{Y}(k) \\ \boldsymbol{0} \end{smallmatrix}\right]^t \breve{\boldsymbol{x}}, \forall \breve{\boldsymbol{x}} \in \mathbb{R}^{2N+3}$. Given an $\breve{\boldsymbol{x}}$, an efficient approximation of the difficult to compute $P_{C_k}(\breve{\boldsymbol{x}})$ is given by means of the metric projection mapping onto the halfspace defined by $\Pi^-_{F_k'(\breve{\boldsymbol{x}})} := \left\{\breve{\boldsymbol{y}} \in \mathbb{R}^{2N+3} : \langle \breve{\boldsymbol{y}} - \breve{\boldsymbol{x}}, F_k'(\breve{\boldsymbol{x}})\rangle + F_k(\breve{\boldsymbol{x}}) \le 0\right\}$. Notice that $C_k \subset \Pi^-_{F_k'(\breve{\boldsymbol{x}})}$ and $\breve{\boldsymbol{x}} \notin C_k$ iff $\breve{\boldsymbol{x}} \notin \Pi^-_{F_k'(\breve{\boldsymbol{x}})}, \forall \breve{\boldsymbol{x}} \in \mathbb{R}^{2N+3}$ [6]. Then, $P_{\Pi^-_{F_k'(\breve{\boldsymbol{x}})}}(\breve{\boldsymbol{x}}) = \breve{\boldsymbol{x}} - \frac{F_k(\breve{\boldsymbol{x}})^+}{\|F_k'(\breve{\boldsymbol{x}})\|^2} F_k'(\breve{\boldsymbol{x}})$, if $F_k'(\breve{\boldsymbol{x}}) \ne \boldsymbol{0}$, and $P_{\Pi^-_{F_k'(\breve{\boldsymbol{x}})}}(\breve{\boldsymbol{x}}) = \breve{\boldsymbol{x}}$, if $F_k'(\breve{\boldsymbol{x}}) = \boldsymbol{0}$.

Step 4. *(Robust Capon beamformer)* Define the index set $\mathcal{L}_k \subset \mathbb{Z}_{\ge 0}$ s.t. $k \in \mathcal{L}_k$ and $\mathrm{card}(\mathcal{L}_k) < \infty$, where $\mathrm{card}(\cdot)$ denotes the cardinality of a set. To $\iota \in \mathcal{L}_k$ we associate a stochastic property set C_ι. We define a set of nonnegative weights $\{\omega_\iota'(k)\}_{\iota\in\mathcal{L}_k}$, $\{\omega_\nu''(k)\}_{\nu=0}^2$ s.t. $\sum_{\iota\in\mathcal{L}_k} \omega_\iota'(k) + \sum_{\nu=0}^2 \omega_\nu''(k) = 1$. Then, by collecting all the projections calculated in the previous steps, and for $\mu_k \in [0, 2\mathcal{M}_k]$,

$$\boldsymbol{u}_{k+1} := P_{K_0}P_{K_1}P_{K_2}\left(\boldsymbol{u}_k + \mu_k\left(\sum_{\iota\in\mathcal{L}_k} \omega_\iota'(k) P_{\Pi^-_{F_\iota'(\boldsymbol{u}_k)}}(\boldsymbol{u}_k) + \sum_{\nu=0}^2 \omega_\nu''(k) P_{\Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})}(\boldsymbol{u}_k) - \boldsymbol{u}_k\right)\right),$$

$$\mathcal{M}_k := \begin{cases} \begin{aligned} &\sum_{\iota\in\mathcal{L}_k} \frac{\omega_\iota'(k)}{\alpha_k}\left\|P_{\Pi^-_{F_\iota'(\boldsymbol{u}_k)}}(\boldsymbol{u}_k) - \boldsymbol{u}_k\right\|^2 \\ &\qquad + \sum_{\nu=0}^2 \frac{\omega_\nu''(k)}{\alpha_k}\left\|P_{\Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})} - \boldsymbol{u}_k\right\|^2, \\ &\quad \boldsymbol{u}_k \notin (\textstyle\bigcap_{\iota\in\mathcal{L}_k} C_\iota) \cap (\bigcap_{\nu=0}^2 \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})), \end{aligned} \\ 1, \quad \text{otherwise}, \end{cases}$$

where $\alpha_k := \|\sum_{\nu=0}^2 \omega_\nu''(k) P_{\Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})}(\boldsymbol{u}_k)$ $+ \sum_{\iota\in\mathcal{L}_k} \omega_\iota'(k) P_{\Pi^-_{F_\iota'(\boldsymbol{u}_k)}}(\boldsymbol{u}_k) - \boldsymbol{u}_k\|^2$. Notice that $\mathcal{M}_k \ge 1$.

Step 5. Set $k \leftarrow k + 1$, and go to Step 1.

Since the most expensive operation above is the multiplication of a matrix with a vector, the computational complexity of the proposed algorithm is $O(N^2)$.

Let us focus now on Step 4 of the above algorithm. Fix $k \in \mathbb{Z}_{\ge 0}$, and define $\mathcal{I}_k' := \{\iota \in \mathcal{L}_k : \boldsymbol{u}_k \notin \Pi^-_{F_\iota'(\boldsymbol{u}_k)}\}$, $\mathcal{I}_k'' := \{\nu \in \{0, 1, 2\} : \boldsymbol{u}_k \notin \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})\}$. If we assume $\Omega_k := (\bigcap_{\nu=0}^2 K_\nu) \cap (\bigcap_{\iota\in\mathcal{I}_k'} \Pi^-_{F_\iota'(\boldsymbol{u}_k)}) \cap (\bigcap_{\nu\in\mathcal{I}_k''} \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})) \ne \emptyset$, then a robust Capon beamformer would be given by the first $2N$ components of a $\boldsymbol{u}_k \in \Omega_k$. Suppose, now, that $\exists k_0 \in \mathbb{Z}_{>0}$ s.t. $\Omega_k \ne \emptyset, \forall k \ge k_0$. Assume also $\Omega := \bigcap_{k\ge k_0} \Omega_k \ne \emptyset$. A solution, thus, to the robust Capon beamforming problem will be given by a point sequence $(\boldsymbol{u}_k)_{k\in\mathbb{Z}_{\ge 0}}$ that converges in some sense to the vicinity of Ω. A mathematical description of these issues is given below by using APSM arguments.

Proposition 2 *Assume that there exists $k_0 \in \mathbb{Z}_{\ge 0}$ s.t. $\Omega \ne \emptyset$. Assume $(\boldsymbol{u}_k)_{k\in\mathbb{Z}_{\ge 0}}$ generated by the proposed algorithm with $\mu_k \in [\mathcal{M}_k\epsilon_1, \mathcal{M}_k(2 - \epsilon_2)]$, $\forall k \ge k_0$, for some small $\epsilon_1, \epsilon_2 > 0$.*

1. *Assume that $\omega_0' := \inf_{\iota\in\mathcal{L}_k, k\ge 0} \omega_\iota'(k) > 0$, and $\omega_0'' := \inf_{\nu\in\{0,1,2\}, k\ge 0} \omega_\nu''(k) > 0$. Then, $\lim_{k\to\infty} d(\boldsymbol{u}_k, \Pi^-_{F_k'(\boldsymbol{u}_k)}) = 0$ and $\lim_{k\to\infty} d(\boldsymbol{u}_k, \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})) = 0$, $\nu = 0, 1, 2$.*
2. *Assume ω_0', ω_0'', and that $(F_k'(\boldsymbol{u}_k))_{k\in\mathbb{Z}_{\ge 0}}$ is bounded. Then, $\lim_{k\to\infty} F_k(\boldsymbol{u}_k)^+ = 0$, where $\alpha^+ := \max\{\alpha, 0\}, \forall \alpha \in \mathbb{R}$.*

Assume that there exists a hyperplane $\Pi \subset \mathcal{H}$ s.t. $\mathrm{ri}_\Pi(\Omega) \ne \emptyset$ in the algorithm above. Then the followings hold.

3. *There exists $\widehat{\boldsymbol{u}} \in \bigcap_{\nu=0}^2 K_\nu \ne \emptyset$ s.t. $\lim_{k\to\infty} \boldsymbol{u}_k = \widehat{\boldsymbol{u}}$.*
4. *Assume also ω_0', ω_0''. Then, $\lim_{k\to\infty} d(\widehat{\boldsymbol{u}}, \Pi^-_{F_k'(\boldsymbol{u}_k)}) = 0$ and $\lim_{k\to\infty} d(\widehat{\boldsymbol{u}}, \Pi_\nu(\widetilde{\boldsymbol{s}}_0^{(k)})) = 0$, $\nu = 0, 1, 2$.*
5. *Assume ω_0', ω_0'' and that the sequences $(F_k'(\boldsymbol{u}_k))_{k\in\mathbb{Z}_{\ge 0}}$ and $(F_k'(\widehat{\boldsymbol{u}}))_{k\in\mathbb{Z}_{\ge 0}}$ are bounded. Then, $\lim_{k\to\infty} F_k(\widehat{\boldsymbol{u}})^+ = 0$.*
6. *(Characterization of the limit point $\widehat{\boldsymbol{u}}$) Assume $\mathrm{int}(\Omega) \ne \emptyset$ and the existence of ω_0', ω_0''. Then, $\widehat{\boldsymbol{u}} \in \overline{\liminf_{k\to\infty} \Omega_k}$, i.e. the limit point belongs to the closure of $\liminf_{k\to\infty} \Omega_k := \bigcup_{k=0}^\infty \bigcap_{m\ge k} \Omega_m$, which is the set of all those points that lie in all but finite Ω_ks.*

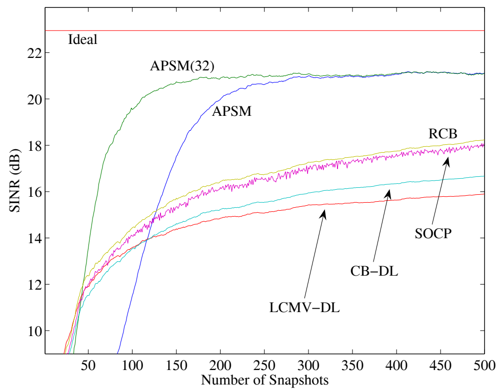

Fig. 2. SNR $= 10$dB, while $\sigma_j^2 = 100$, $j = 1, 2, 3$, and $\sigma_4^2 = \sigma_5^2 = 0.1$; thus INR $= 34.78$dB.

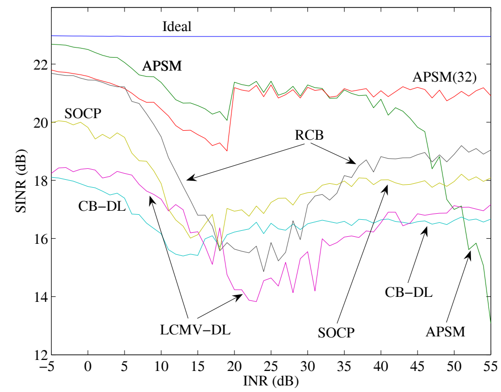

Fig. 3. SNR $= 10$dB. We uniformly increase σ_j^2, $j = 1, 2, 3$, while $\sigma_4^2 = \sigma_5^2 = 0.1$ for INR > 3dB.

4. NUMERICAL RESULTS

We assume $N = 20$ and that $d/\lambda = 0.5$. Five interferences arrive at the ULA with DOAs of $30°$, $50°$, $90°$, $120°$, and $150°$. The DOA of the SOI wavefront is $70°$. The 16-PSK scheme is used to modulate six mutually independent, uniformly distributed sequences of symbols providing us with the random processes $(q_j(k))_{k\in\mathbb{Z}_{\geq 0}} \subset \mathbb{C}$, $j = 0, \ldots, 5$. Define $\sigma_j^2 := \mathsf{E}\{|q_j(k)|^2\}$, $\forall k \in \mathbb{Z}_{\geq 0}$, $j = 0, \ldots, 5$. We fix $\sigma_0^2 = 1$. For the noise process, we let $\boldsymbol{R_n} := \sigma_{\boldsymbol{n}}^2 \boldsymbol{I}_N$. To form the erroneous $\widetilde{s}_0$ we introduce a mismatch of $\pm 1.5°$ for the SOI DOA.

Let, now, SNR $:= \sigma_0^2/\sigma_{\boldsymbol{n}}^2$, and INR $:= \sum_{j=1}^5 \sigma_j^2/\sigma_{\boldsymbol{n}}^2$. The evaluation of various methods will be done by the *Signal to Interference and Noise Ratio (SINR)* function which is defined as $\mathrm{SINR}(k) = \frac{\sigma_0^2 |\boldsymbol{s}_0^* \boldsymbol{w}(k)|^2}{\sum_{j=1}^5 \sigma_j^2 |\boldsymbol{s}_j^* \boldsymbol{w}(k)|^2 + \sigma_{\boldsymbol{n}}^2 \|\boldsymbol{w}(k)\|^2}$, $\forall k \in \mathbb{Z}_{\geq 0}$, where $(\boldsymbol{w}(k))_{k\in\mathbb{Z}_{\geq 0}}$ is the sequence of weighting vectors obtained by the implementation of various adaptive beamforming schemes.

In the above figures, APSM denotes the proposed robust adaptive beamformer with $\mathrm{card}(\mathcal{L}_k) = 1$, while APSM(32) refers to the proposed design with $\mathrm{card}(\mathcal{L}_k) = 32$, $\forall k \in \mathbb{Z}_{\geq 0}$, in Step 4. Whenever INR $>$ SNR $+ 10$, we assign bigger values to the weights $\{\omega_\nu''(k)\}_{\nu=0}^2$ than $\{\omega_\iota'(k)\}_{\iota\in\mathcal{L}_k}$ in order to cancel the strong interference. For INR $\leq$ SNR$+10$, the opposite scenario is followed and we put more weight on the data (stochastic property sets in Step 3) than the sets introduced in Step 2. This explains the 'jumps' observed in the neighborhood of INR=20dB in Fig. 3. CB-DL will stand for the beamformer obtained by (3) with $\epsilon_{\mathrm{DL}} := 10$, while LCMV-DL denotes the diagonally loaded version of the *Linearly Constrained Minimum Variance* beamformer [2, §6.7] where the steering vectors of all jammers are also assumed to be known with some error. SOCP denotes the approach in [5], and RCB corresponds to the *Robust Capon Beamformer* with a spherical constraint in [4]. The term 'Ideal' will refer to the solution of (1). Each point in Figs. 2 and 3 is the uniform average of 100 realizations.

5. REFERENCES

[1] J. Capon, "High resolution frequency wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, August 1969.

[2] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*, John Wiley & Sons, New York, 2002.

[3] S. Shahbazpanahi, A. B. Gershman, Z.-Q. Luo, and K. M. Wong, "Robust adaptive beamforming for general-rank signal models," *IEEE Transactions on Signal Processing*, vol. 51, no. 9, pp. 2257–2269, Sept. 2003.

[4] J. Li, P. Stoica, and Z. Wang, "On robust Capon beamforming and diagonal loading," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1702–1715, July 2003.

[5] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem," *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 313–324, Feb. 2003.

[6] K. Slavakis, M. Yukawa, and I. Yamada, "Efficient robust Capon beamforming by the Adaptive Projected Subgradient Method," Submitted for publication.

[7] K. Slavakis, M. Yukawa, and I. Yamada, "Efficient robust adaptive beamforming by the Adaptive Projected Subgradient Method: A set theoretic time-varying approach over multiple a-priori constraints," Tech. Rep. SIP2005-52, WBS2005-10 (2005-07), IEICE, July 2005.

[8] H. H. Bauschke, *Projection Algorithms and Monotone Operators*, Ph.D. thesis, Simon Fraser University, 1996.

[9] K. Slavakis, I. Yamada, N. Ogura, and M. Yukawa, "Adaptive Projected Subgradient Method and set theoretic adaptive filtering with multiple convex constraints," in *Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers*, Nov. 2004, pp. 960–964.

[10] K. Slavakis, I. Yamada, and N. Ogura, "The Adaptive Projected Subgradient Method over the fixed point set of strongly attracting nonexpansive mappings," Submitted for publication.

[11] I. Yamada and N. Ogura, "Adaptive Projected Subgradient Method for asymptotic minimization of sequence of nonnegative convex functions," *Numerical Functional Analysis and Optimization*, vol. 25, no. 7&8, pp. 593–617, 2004.

[12] I. Yamada, "The Hybrid Steepest Descent Method for the variational inequality problem over the intersection of fixed point sets of nonexpansive mappings," in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, D. Butnariu, Y. Censor, and S. Reich, Eds., pp. 473–504. Elsevier, Amsterdam, 2001.