Set-Membership Affine Projection Channel Estimation for Wireless Sensor Networks

Pouya Ghofrani, Tong Wang* and Anke Schmeink
Institute for Theoretical Information Technology
RWTH Aachen University, D-52074 Aachen, Germany
ghofrani@ti.rwth-aachen.de
{tong.wang, schmeink}@umic.rwth-aachen.de

***Abstract*—A set-membership affine projection algorithm is applied to estimate the communication channel between the wireless sensor nodes in a general form, where the channel is modeled as a complex matrix in the presence of additive white gaussian noise. An efficient hybrid model for the affine projection algorithm is briefly introduced and the problem of matrix invertibility in some cases of the affine projection algorithm is resolved by a new method which does not use any matrix inversion. Simulations show good performance of our proposed algorithm in terms of convergence speed and demonstrate reduced complexity.**

I. Introduction

Wireless sensor networks (WSN) have been under special attention by researchers and engineers due to their wide range of applications in health, military, home, etc. Although new technologies in the WSN electronics are developing each day, there might be some features for sensor networks that do not change. Some examples can be their low-cost, low-power, small-size and rapid deployment properties. Since there are many constraints on sensor networks, the best solutions are the ones which can make a suitable trade-off between the restrictions. Unlike a multiple-input multiple-output (MIMO) system in which we can apply various techniques for estimation and detection, a sensor network may fail to apply these methods since there might not be enough computational capacity and memory to perform those tasks, or the high power consumption drastically reduces the life-time of the network. This is also true for the layering structure of the network and the protocols. For instance, many of the proposed protocols for the traditional wireless ad hoc networks are not well suited to the requirements and unique features of the sensor networks [1]. Consequently, the regular channel estimation methods are not suitable here due to their high computational loads. For this purpose, a set of algorithms called set membership (SM) algorithms have been proposed which can reduce the computation load by lowering the update rates based on the required resolution. Algorithms such as SM normalized least mean squares (SM-NLMS) and SM recursive least squares (SM-RLS or BEACON) have been proposed and investigated in [2] which can estimate a matrix-based complex channel. The two-dimensional SM affine projection (SM-AP) channel estimation has been investigated in [3] which is for OFDM

*The work of T. Wang was supported by Alexander von Humboldt Foundation.

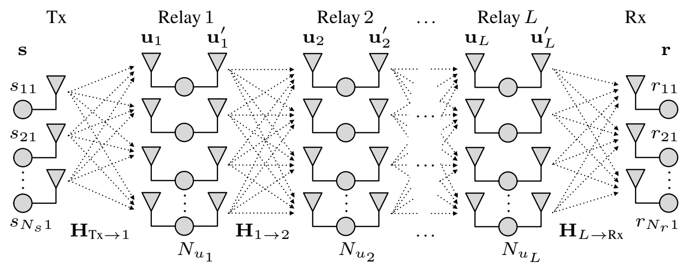

Fig. 1. A WSN consisting of transmitter Tx, L relays and receiver Rx. The number of sensor nodes in the Tx, i^{th} relay and the Rx are denoted by N_s , N_{u_i} and N_r respectively.

systems. A MIMO channel equalization using a real-valued SM-AP is discussed in [4]. In this article, we focus on the channel estimation problem between the sensor nodes in a general form using the SM-AP algorithm. The data reuse feature in AP and SM-AP results in fast convergence. Although the idea behind the SM-AP estimation already exists [5], it is mainly for the estimation of real-valued vector-based channel matrices. Moreover, the algorithms generally include a sort of matrix inversion which is not suitable here for the purpose of WSNs when the size of matrix is large. Proposing the matrix-based and complex-valued AP and SM-AP algorithms are the major novelties of this article. Another novelty of this work is proposing the finite iterative algorithm [6], which can solve complex matrix equations with arbitrary accuracy and fast convergence, to convert the matrix inversion process to an iterative algorithm. Also we briefly introduce the hybrid AP (HAP) algorithm which has a low complexity, low steady-state error (SSE) and a fast convergence. The rest of this paper is organized as follows: in Section II a WSN system model is described. In Section III the proposed channel estimation algorithm is discussed in detail and the solutions for AP and SM-AP are derived for different cases of the problem, and finally, the simulation results are presented in Section IV.

II. WSN System Model

Figure 1 illustrates a wireless sensor network with L relay groups between N_s transmitter nodes (Tx) and N_r receiver nodes (Rx), where the communication channels from the Tx to relay 1, from relay i to relay $i+1, 1 \leq i \leq L-1$, and from relay L to the Rx are denoted by $\mathbf{H}_{\text{Tx}\to 1}, \mathbf{H}_{i\to i+1}, \mathbf{H}_{L\to \text{Rx}}$, respectively. For each $i, 1 \leq i \leq L$, the i^{th} relay group consists of N_{u_i} sensor nodes. The received and transmitted signals by/from the i^{th} relay are denoted by the complex vectors $\mathbf{u}_i$, $\mathbf{u}'_i \in \mathbb{C}^{N_{u_i}}$, respectively. For the sake of clarity, each sensor node in any relay group is shown by two antennas for transmission and reception separately. As Figure 1 depicts, the complex signal vector $\mathbf{s} = [s_{11}\, s_{21} \ldots s_{N_s 1}]^{\mathrm{T}}$ is transmitted from the Tx to the first relay through the channel $\mathbf{H}_{\text{Tx}\to 1}$, then the signal is received by relay 1 as vector $\mathbf{u}_1$ and a signal $\mathbf{u}'_1$ which contains the desired information acquired from $\mathbf{u}_1$ is retransmitted to the next relay group through the channel $\mathbf{H}_{1\to 2}$ and the same procedure happens for other relays. Finally, the signal is received at the destination as a complex vector $\mathbf{r} = [r_{11}\; r_{21}\; \ldots\; r_{N_r 1}]^{\mathrm{T}}$ through $\mathbf{H}_{L\to\text{Rx}}$. Assuming an additive white Gaussian noise (AWGN) system model and defining the integer set $\mathbb{F}_i^j := \{i, i{+}1, \ldots, j\}\,, \forall i, j \in \mathbb{Z}, i \leq j$, the whole communication process described above can be formulated as follows

$$\begin{cases} \mathbf{u}_1 &= \mathbf{H}_{\text{Tx}\to 1}\ \mathbf{s} \quad + \mathbf{n}_{\text{Tx}\to 1} \\ \mathbf{u}_{i+1} &= \mathbf{H}_{i\to i+1}\, \mathbf{u}'_i \ + \mathbf{n}_{i\to i+1}\,, \quad \forall i \in \mathbb{F}_1^{L-1} \\ \mathbf{r} &= \mathbf{H}_{L\to\text{Rx}}\ \mathbf{u}'_L + \mathbf{n}_{L\to\text{Rx}} \end{cases}$$

where $\mathbf{u}_i, \mathbf{u}'_i \in \mathbb{C}^{N_{u_i}}$,$\forall i \in \mathbb{F}_1^L$, $\mathbf{H}_{\text{Tx}\to 1} \in \mathbb{C}^{N_{u_1}\times N_s}$, $\mathbf{H}_{L\to\text{Rx}} \in \mathbb{C}^{N_r \times N_{u_L}}$, $\mathbf{H}_{i\to i+1} \in \mathbb{C}^{N_{u_{i+1}}\times N_{u_i}}$, $\forall i \in \mathbb{F}_1^{L-1}$ and $\mathbf{n}_{\text{Tx}\to 1} \in \mathbb{C}^{N_{u_1}}$, $\mathbf{n}_{L\to\text{Rx}} \in \mathbb{C}^{N_r}$, $\mathbf{n}_{i\to i+1} \in \mathbb{C}^{N_{u_{i+1}}}$, $\forall i \in \mathbb{F}_1^{L-1}$ are the corresponding noise vectors in each step. Depending on the type of the WSN and its configuration, vector $\mathbf{u}'_i$ can be a modified or amplified version of $\mathbf{u}_i$ such as amplify-and-forward or decode-and-forward strategies in cooperative WSNs [7].

III. Proposed Channel Estimation Algorithm

For our channel estimation problem, and independent of the WSN's strategy, we consider the system model for each hop with N_s sensor nodes transmitting, and N_r sensor nodes receiving, described by $\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n}$, where $\mathbf{r}$ is the $N_r \times 1$ received signal vector, $\mathbf{H}$ is the $N_r \times N_s$ complex channel matrix, $\mathbf{s}$ is the $N_s \times 1$ transmitted signal vector and $\mathbf{n}$ is the $N_r \times 1$ AWGN vector. Moreover, we assume that the channel matrix $\mathbf{H}$ is constant during the transmission and reception of K pilot vectors $\mathbf{s}_i, \forall i \in \mathbb{F}_1^K$, which are known to the receiver. So at any time instant $i \in \mathbb{F}_1^K$, $\mathbf{r}_i = \mathbf{H}\mathbf{s}_i + \mathbf{n}_i$, where $\mathbf{r}_i$ and $\mathbf{n}_i$ are the corresponding received and noise vectors at time instant i and $\mathbf{H}$ is the matrix to be estimated.

A. Optimization

Denoting the estimated matrix at k^{th} time-step of estimation ($k \in \mathbb{F}_P^K$) by $\mathbf{H}_k$, the objective of the AP algorithm is to

$$\text{minimize } \|\mathbf{H}_{k+1} - \mathbf{H}_k\|_{\mathrm{F}}^2$$
$$\text{subject to : } \mathbf{r}_{k-i} - \mathbf{H}_{k+1}\mathbf{s}_{k-i} = \mathbf{0}\,, \forall i \in \mathbb{F}_0^{P-1}$$

where the subscript F denotes the Frobenius norm and P is the number of data reuse. Note that the initial $\mathbf{H}_k$ matrix at $k = P$ can be set as $\mathbf{H}_P = \mathbf{0}_{N_r \times N_s}$. For simulations, we shift the curves by $k = P$ to compensate for this starting point offset. The objective above can be interpreted as keeping the new update as close as possible to the current value while forcing the *a posteriori* error to be zero. Since forcing this error to be zero in comparison with the case that some error can be tolerated is a more stringent constraint that has to be fulfilled and needs more power consumption which is not desired in a sensor network, we weaken this condition by keeping the a posteriori error in an acceptable region rather than just zero, based on the demanded resolution. This can be achieved by applying vector parameters $\mathbf{g}_{k-i} \in \mathbb{C}^{N_r \times 1}\,, \forall i \in \mathbb{F}_0^{P-1}$ to our calculations. These vectors are bounded so that the error is limited. So the constraint for our problem can be rewritten as

$$\mathbf{r}_{k-i} - \mathbf{H}_{k+1}\mathbf{s}_{k-i} - \mathbf{g}_{k-i} = \mathbf{0}, \quad \forall i \in \mathbb{F}_0^{P-1}.$$

Different choices for vectors $\mathbf{g}_{k-i}$, $i \in \mathbb{F}_0^{P-1}$ are discussed in Subsection III-B, but for the time being, we assume that any $\mathbf{g}_{k-i}$, $i \in \mathbb{F}_0^{P-1}$ is valid as long as $\|\mathbf{g}_{k-i}\|$ is bounded by a positive constant like γ. As we will later see, the SM algorithm makes decision about updating based on comparing a scalar function of an error vector with the threshold γ. To solve the optimization problem we use the method of Lagrange multipliers. Since the objective function includes a complex-valued constraint in general, a suggestion is to decompose the whole complex variables and parameters as a sum of their real and imaginary parts. Therefore we make an equivalent version of the objective function in which the whole parameters and variables are real. Using the superscripts R and I for real and imaginary parts respectively, the channel matrix can be expressed as

$$\mathbf{H}_k = \mathbf{H}_k^R + j\mathbf{H}_k^I\,, \quad \mathbf{H}_k^R\,, \mathbf{H}_k^I \in \mathbb{R}^{N_r \times N_s}\,, \quad \forall k \in \mathbb{F}_P^{K+1}.$$

By applying the same method to $\|\mathbf{H}_{k+1} - \mathbf{H}_k\|_{\mathrm{F}}^2$ and the constraints, we deduce the following real-valued relations

$$\|\mathbf{H}_{k+1} - \mathbf{H}_k\|_{\mathrm{F}}^2 = \|\mathbf{H}_{k+1}^R - \mathbf{H}_k^R\|_{\mathrm{F}}^2 + \|\mathbf{H}_{k+1}^I - \mathbf{H}_k^I\|_{\mathrm{F}}^2,$$

$$\mathbf{r}_{k-i}^R - \mathbf{H}_{k+1}^R\mathbf{s}_{k-i}^R + \mathbf{H}_{k+1}^I\mathbf{s}_{k-i}^I - \mathbf{g}_{k-i}^R = \mathbf{0} \quad, \forall i \in \mathbb{F}_0^{P-1}, \tag{1}$$

$$\mathbf{r}_{k-i}^I - \mathbf{H}_{k+1}^I\mathbf{s}_{k-i}^R - \mathbf{H}_{k+1}^R\mathbf{s}_{k-i}^I - \mathbf{g}_{k-i}^I = \mathbf{0} \quad, \forall i \in \mathbb{F}_0^{P-1}. \tag{2}$$

Now, the Lagrange function $\mathcal{L}$ can be defined as

$$\begin{aligned} \mathcal{L}(\mathbf{H}_{k+1}^R, \mathbf{H}_{k+1}^I, \boldsymbol{\lambda}_i, \boldsymbol{\mu}_i) &:= tr\{(\mathbf{H}_{k+1}^R - \mathbf{H}_k^R)(\mathbf{H}_{k+1}^R - \mathbf{H}_k^R)^{\mathrm{T}}\} \\ &+ tr\{(\mathbf{H}_{k+1}^I - \mathbf{H}_k^I)(\mathbf{H}_{k+1}^I - \mathbf{H}_k^I)^{\mathrm{T}}\} \\ &+ \sum_{i=0}^{P-1} \boldsymbol{\lambda}_i^{\mathrm{T}}(\mathbf{r}_{k-i}^R - \mathbf{H}_{k+1}^R\mathbf{s}_{k-i}^R + \mathbf{H}_{k+1}^I\mathbf{s}_{k-i}^I - \mathbf{g}_{k-i}^R) \\ &+ \sum_{i=0}^{P-1} \boldsymbol{\mu}_i^{\mathrm{T}}(\mathbf{r}_{k-i}^I - \mathbf{H}_{k+1}^I\mathbf{s}_{k-i}^R - \mathbf{H}_{k+1}^R\mathbf{s}_{k-i}^I - \mathbf{g}_{k-i}^I) \end{aligned}$$

where tr stands for the trace function and $\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i \in \mathbb{R}^{N_r}\,, \forall i \in \mathbb{F}_0^{P-1}$ are the vectors of Lagrange multipliers. Since at the optimum point $\nabla\mathcal{L} = \mathbf{0}$, the four partial derivatives should be zero as follows

$$\nabla\mathcal{L} = \left(\frac{\partial\mathcal{L}}{\partial\mathbf{H}_{k+1}^R}\,,\ \frac{\partial\mathcal{L}}{\partial\mathbf{H}_{k+1}^I}\,,\ \frac{\partial\mathcal{L}}{\partial\boldsymbol{\lambda}_i}\,,\ \frac{\partial\mathcal{L}}{\partial\boldsymbol{\mu}_i}\right)^{\mathrm{T}} = \mathbf{0}\,, \forall i \in \mathbb{F}_0^{P-1}. \tag{3}$$

From the equations $\frac{\partial \mathcal{L}}{\partial \mathbf{H}_{k+1}^R} = \mathbf{0}$, $\frac{\partial \mathcal{L}}{\partial \mathbf{H}_{k+1}^I} = \mathbf{0}$ in (3) we deduce

$$\mathbf{H}_{k+1}^R = \mathbf{H}_k^R + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}}, \quad (4)$$

$$\mathbf{H}_{k+1}^I = \mathbf{H}_k^I - \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}} + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} \quad (5)$$

respectively, and the last two equations $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}_i} = \mathbf{0}$, $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_i} = \mathbf{0}$ in (3) return the constraints (1) and (2). By incorporating (4) , (5) into (1) , (2) and using the index *m* instead of i in (1) , (2) for the sake of clarity, for every $m \in \mathbb{F}_0^{P-1}$ we obtain

$$\begin{aligned} &\mathbf{r}_{k-m}^R - \mathbf{H}_k^R \mathbf{s}_{k-m}^R + \mathbf{H}_k^I \mathbf{s}_{k-m}^I - \mathbf{g}_{k-m}^R \\ &= (\frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}}) \mathbf{s}_{k-m}^R \\ &-(\frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} - \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}}) \mathbf{s}_{k-m}^I \end{aligned}$$

and

$$\begin{aligned} &\mathbf{r}_{k-m}^I - \mathbf{H}_k^I \mathbf{s}_{k-m}^R - \mathbf{H}_k^R \mathbf{s}_{k-m}^I - \mathbf{g}_{k-m}^I \\ &= (\frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} - \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}}) \mathbf{s}_{k-m}^R \\ &+(\frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}}) \mathbf{s}_{k-m}^I. \end{aligned}$$

Defining three new matrices $\mathbf{X}_k, \mathbf{Y}_k$ and $\mathbf{Z}_k$ as

$$\begin{aligned} \mathbf{X}_k &:= \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} + \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}} \\ \mathbf{Y}_k &:= \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\mu}_i \mathbf{s}_{k-i}^{R\,\mathrm{T}} - \frac{1}{2}\sum_{i=0}^{P-1} \boldsymbol{\lambda}_i \mathbf{s}_{k-i}^{I\,\mathrm{T}} \\ \mathbf{Z}_k &:= \mathbf{X}_k + j\mathbf{Y}_k, \end{aligned}$$

one can prove that $\mathbf{H}_{k+1} = \mathbf{H}_k + \mathbf{Z}_k$ and $\mathbf{Z}_k \mathbf{s}_{k-m} = \mathbf{r}_{k-m} - \mathbf{H}_k \mathbf{s}_{k-m} - \mathbf{g}_{k-m}$, $\forall m \in \mathbb{F}_0^{P-1}$. Now the whole P equations in the latter relation can be cast into one equation as follows

$$\mathbf{Z}_k \mathbf{S}_k = \mathbf{R}_k - \mathbf{H}_k \mathbf{S}_k - \mathbf{G}_k = \mathbf{E}_k - \mathbf{G}_k \quad (6)$$

where

$$\begin{aligned} \mathbf{S}_k &:= [\, \mathbf{s}_{k-P+1} \; \mathbf{s}_{k-P+2} \; \dots \; \mathbf{s}_k \,] \in \mathbb{C}^{N_s \times P} \\ \mathbf{R}_k &:= [\, \mathbf{r}_{k-P+1} \; \mathbf{r}_{k-P+2} \; \dots \; \mathbf{r}_k \,] \in \mathbb{C}^{N_r \times P} \\ \mathbf{G}_k &:= [\, \mathbf{g}_{k-P+1} \; \mathbf{g}_{k-P+2} \; \dots \; \mathbf{g}_k \,] \in \mathbb{C}^{N_r \times P} \\ \mathbf{E}_k &:= [\, \boldsymbol{\epsilon}_{k-P+1} \; \boldsymbol{\epsilon}_{k-P+2} \; \dots \; \mathbf{e}_k \,] \in \mathbb{C}^{N_r \times P} \end{aligned}$$

and the vectors $\boldsymbol{\epsilon}_{k-i} := \mathbf{r}_{k-i} - \mathbf{H}_k \mathbf{s}_{k-i}$, $\forall i \in \mathbb{F}_1^{P-1}$ are the a posteriori error vectors while $\mathbf{e}_k := \mathbf{r}_k - \mathbf{H}_k \mathbf{s}_k$ is the current error vector. Note that according to the objective function defined for the AP, we have to find the matrix $\mathbf{Z}_k$ which has the minimum Frobenius norm among the whole set of solutions for $\mathbf{Z}_k \mathbf{S}_k = \mathbf{E}_k - \mathbf{G}_k$ in cases which the solution is not unique. Therefore we consider several distinct cases based on the characteristics of the matrix $\mathbf{S}_k$.

1) $\mathbf{S}_k$ square and full rank: In this case since *rank*$\{\mathbf{S}_k\} = N_s = P$, $\mathbf{Z}_k$ can be found uniquely by

$$\mathbf{Z}_k = (\mathbf{E}_k - \mathbf{G}_k)\mathbf{S}_k^{-1}.$$

2) $\mathbf{S}_k$ broad and full rank: In this case, the matrix $\mathbf{S}_k$ is a broad matrix, i.e. $N_s < P$ and has full rank of $\min\{N_s, P\} = N_s$. Thus, since *rank*$\{\mathbf{S}_k \mathbf{S}_k^H\}$ = *rank*$\{\mathbf{S}_k\} = N_s$, matrix $\mathbf{S}_k \mathbf{S}_k^H$ is invertible and necessarily $\mathbf{Z}_k$ is yielded as follows

$$\mathbf{Z}_k = (\mathbf{E}_k - \mathbf{G}_k)\mathbf{S}_k^{\mathrm{H}}(\mathbf{S}_k \mathbf{S}_k^{\mathrm{H}})^{-1},$$

but using this solution for $\mathbf{Z}_k$ results in the following equality

$$(\mathbf{E}_k - \mathbf{G}_k)\mathbf{S}_k^{\mathrm{H}}(\mathbf{S}_k \mathbf{S}_k^{\mathrm{H}})^{-1}\mathbf{S}_k = \mathbf{E}_k - \mathbf{G}_k$$

which is not satisfied in general case. So in this case there is only one solution as derived above or there is not any solution.

3) $\mathbf{S}_k$ tall and full rank: In this case since $\mathbf{S}_k$ is a tall matrix, i.e. $P < N_s$ and has full rank of $\min\{N_s, P\} = P$, the solution for $\mathbf{Z}_k$ is not unique. Inevitably we need to find the solution which has the minimum Frobenius norm. Defining $\mathbf{A}_k := \mathbf{E}_k - \mathbf{G}_k$ and separating the real and imaginary parts of $\mathbf{Z}_k \mathbf{S}_k = \mathbf{A}_k$, we obtain two real constraints or their vector equivalents as follows

$$\mathbf{Z}_k^R \mathbf{S}_k^R - \mathbf{Z}_k^I \mathbf{S}_k^I = \mathbf{A}_k^R \iff \mathbf{vec}(\mathbf{Z}_k^R \mathbf{S}_k^R - \mathbf{Z}_k^I \mathbf{S}_k^I) = \mathbf{vec}(\mathbf{A}_k^R),$$

$$\mathbf{Z}_k^I \mathbf{S}_k^R + \mathbf{Z}_k^R \mathbf{S}_k^I = \mathbf{A}_k^I \iff \mathbf{vec}(\mathbf{Z}_k^I \mathbf{S}_k^R + \mathbf{Z}_k^R \mathbf{S}_k^I) = \mathbf{vec}(\mathbf{A}_k^I),$$

where the equivalencies above are due to the same size of matrices on the left and right-hand sides of the equalities. Ignoring the subscript k for the sake of simplicity, the equivalent statements for the constraints above in the vector form are

$$(\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^R) - (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^I) - \mathbf{vec}(\mathbf{A}^R) = \mathbf{0}, \quad (7)$$

$$(\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^I) + (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^R) - \mathbf{vec}(\mathbf{A}^I) = \mathbf{0}, \quad (8)$$

respectively, where $\otimes$ stands for Kronecker product. The new Lagrange function $\mathcal{W}$ is defined as follows

$$\begin{aligned} &\mathcal{W}(\mathbf{vec}(\mathbf{Z}^R), \mathbf{vec}(\mathbf{Z}^I), \boldsymbol{\zeta}, \boldsymbol{\eta}) := \|\mathbf{vec}(\mathbf{Z}^R)\|^2 + \|\mathbf{vec}(\mathbf{Z}^I)\|^2 \\ &+\boldsymbol{\zeta}^{\mathrm{T}}((\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^R) - (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^I) - \mathbf{vec}(\mathbf{A}^R)) \\ &+\boldsymbol{\eta}^{\mathrm{T}}((\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^I) + (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{Z}^R) - \mathbf{vec}(\mathbf{A}^I)) \end{aligned}$$

where $\boldsymbol{\zeta}, \boldsymbol{\eta} \in \mathbb{R}^{PN_r}$ are Lagrange multiplier vectors. Setting $\nabla \mathcal{W} = \mathbf{0}$ results in

$$\nabla \mathcal{W} = \left(\frac{\partial \mathcal{W}}{\partial \mathbf{vec}(\mathbf{Z}^R)} \, , \, \frac{\partial \mathcal{W}}{\partial \mathbf{vec}(\mathbf{Z}^I)} \, , \, \frac{\partial \mathcal{W}}{\partial \boldsymbol{\zeta}} \, , \, \frac{\partial \mathcal{W}}{\partial \boldsymbol{\eta}} \right)^{\mathrm{T}} = \mathbf{0}.$$

From the first two derivatives above we conclude

$$2\mathbf{vec}(\mathbf{Z}^R) + (\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})^{\mathrm{T}} \boldsymbol{\zeta} + (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})^{\mathrm{T}} \boldsymbol{\eta} = \mathbf{0}, \quad (9)$$

$$2\mathbf{vec}(\mathbf{Z}^I) - (\mathbf{S}^{I^{\mathrm{T}}} \otimes \mathbf{I})^{\mathrm{T}} \boldsymbol{\zeta} + (\mathbf{S}^{R^{\mathrm{T}}} \otimes \mathbf{I})^{\mathrm{T}} \boldsymbol{\eta} = \mathbf{0}. \quad (10)$$

After incorporating (9) , (10) into (7) , (8) and denoting the Hermitian transpose by superscript H, one can prove the following relationships

$$\mathbf{vec}(\mathbf{Z}) = -\frac{1}{2}(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}(\boldsymbol{\zeta} + j\boldsymbol{\eta}),$$

$$(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}(\boldsymbol{\zeta} + j\boldsymbol{\eta}) = -2\mathbf{vec}(\mathbf{A}).$$

Now since $\mathit{rank}\{(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}\} = \mathit{rank}\{\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I}\} = \mathit{rank}\{\mathbf{S}\} \times \mathit{rank}\{\mathbf{I}\} = P \times N_r$, the square matrix $(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}$ has full rank and consequently the unique solution for $\mathbf{vec}(\mathbf{Z})$ is found by

$$\mathbf{vec}(\mathbf{Z}) = (\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}\big[(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}\big]^{-1}\mathbf{vec}(\mathbf{A}). \quad (11)$$

Using the properties of Kronecker product a compact form for (11) can be obtained as follows

$$\begin{aligned}
\mathbf{vec}(\mathbf{Z}) =& (\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}\big[(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})(\mathbf{S}^{\mathrm{T}} \otimes \mathbf{I})^{\mathrm{H}}\big]^{-1}\mathbf{vec}(\mathbf{A}) \\
=& (\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I})^{\mathrm{T}}\big[(\mathbf{S} \otimes \mathbf{I})^{\mathrm{T}}(\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I})^{\mathrm{T}}\big]^{-1}\mathbf{vec}(\mathbf{A}) \\
=& \Big[\big[(\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I})(\mathbf{S} \otimes \mathbf{I})\big]^{-1}(\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I})\Big]^{\mathrm{T}}\mathbf{vec}(\mathbf{A}) \\
=& \big[(\mathbf{S}^{\mathrm{H}}\mathbf{S} \otimes \mathbf{I})^{-1}(\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I})\big]^{\mathrm{T}}\mathbf{vec}(\mathbf{A}) \\
=& \Big[\big[(\mathbf{S}^{\mathrm{H}}\mathbf{S})^{-1} \otimes \mathbf{I}\big]\big[\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I}\big]\Big]^{\mathrm{T}}\mathbf{vec}(\mathbf{A}) \\
=& \big[(\mathbf{S}^{\mathrm{H}}\mathbf{S})^{-1}\mathbf{S}^{\mathrm{H}} \otimes \mathbf{I}\big]^{\mathrm{T}}\mathbf{vec}(\mathbf{A}) \\
=& (\big[(\mathbf{S}^{\mathrm{H}}\mathbf{S})^{-1}\mathbf{S}^{\mathrm{H}}\big]^{\mathrm{T}} \otimes \mathbf{I})\mathbf{vec}(\mathbf{A}) \\
=& \mathbf{vec}(\mathbf{A}(\mathbf{S}^{\mathrm{H}}\mathbf{S})^{-\mathbf{1}}\mathbf{S}^{\mathrm{H}})
\end{aligned}$$

and since $\mathit{size}\{\mathbf{Z}\} = \mathit{size}\{\mathbf{A}(\mathbf{S}^{\mathrm{H}}\mathbf{S})^{-\mathbf{1}}\mathbf{S}^{\mathrm{H}}\} = N_r \times N_s$,

$$\mathbf{Z}_k = (\mathbf{E}_k - \mathbf{G}_k)(\mathbf{S}_k^{\mathrm{H}}\mathbf{S}_k)^{-1}\mathbf{S}_k^{\mathrm{H}}. \quad (12)$$

4) $\mathbf{S}_k$ not full rank: In this case depending on values of the matrices $\mathbf{S}_k$, $\mathbf{E}_k$ and $\mathbf{G}_k$, there may be no solution, or an infinite number of solutions to the equation $\mathbf{Z}_k\mathbf{S}_k = \mathbf{E}_k - \mathbf{G}_k$. For the channel estimation, we use linearly independent signal vectors such that $\mathbf{S}_k$ has full rank and we set $P \leq N_s$ so that the case 2 in Subsection III-A does not not happen.

B. Applying Set Membership

Analogous to [5], in this section two choices for the matrix $\mathbf{G}_k$ are introduced. First defining the *feasibility set* $\boldsymbol{\Theta}$ as

$$\boldsymbol{\Theta} := \bigcap_{(\mathbf{r},\mathbf{s})\in\mathbb{S}} \{\mathbf{H} \in \mathbb{C}^{N_r \times N_s} : \|\mathbf{r} - \mathbf{H}\mathbf{s}\|_{\max} \leq \frac{\gamma}{\sqrt{N_r}}\}$$

where $\mathbb{S}$ denotes the set of all possible data pairs $(\mathbf{r}, \mathbf{s})$ and for any matrix $\mathbf{Q}$, $\|\mathbf{Q}\|_{\max} := \max\{|q_{ij}|\}\ \forall i, j$, the *constraint set* at any time instant k, $\mathbb{H}_k$, as

$$\mathbb{H}_k := \{\mathbf{H} \in \mathbb{C}^{N_r \times N_s} : \|\mathbf{r}_k - \mathbf{H}\mathbf{s}_k\|_{\max} \leq \frac{\gamma}{\sqrt{N_r}}\}$$

and the *membership set* at time instant k, $\boldsymbol{\psi}_k$ as $\boldsymbol{\psi}_k := \bigcap_{i=1}^{k} \mathbb{H}_i$, it is clear that we can rewrite $\boldsymbol{\psi}_k$ as follows

$$\boldsymbol{\psi}_k = \bigcap_{i=1}^{k-P} \mathbb{H}_i \bigcap_{j=k-P+1}^{k} \mathbb{H}_j = \boldsymbol{\psi}_k^{(k-P)} \bigcap \boldsymbol{\psi}_k^{(P)}$$

where $\boldsymbol{\psi}_k^{(k-P)}, \boldsymbol{\psi}_k^{(P)}$ are the intersections of first $k - P$ and last P constraint sets, respectively. Now according to the objective of the SM-AP, $\mathbf{H}_{k+1}$ should belong to $\boldsymbol{\psi}_k^{(P)}$ where the mathematical solutions for $\mathbf{H}_{k+1} \in \boldsymbol{\psi}_k^{(P)}$ have been obtained in previous sections in terms of the complex matrix $\mathbf{G}_k$ and here in this section, an appropriate bounded matrix $\mathbf{G}_k$ are proposed. Obviously, a trivial choice for this matrix is $\mathbf{G}_k = \mathbf{0}_{N_r \times P}$. Therefore the objective function for the SM-AP in this case would be the same as that of the AP algorithm but the update rate for the SM-AP is reduced by comparing a criterion for $\mathbf{E}_k$ with a threshold as follows

$$\mathbf{H}_{k+1} = \begin{cases} \mathbf{H}_k + \Gamma\mathbf{Z}_k & \text{if } \|\mathbf{e}_k\|_{\max} > \frac{\gamma}{\sqrt{N_r}}; \\ \mathbf{H}_k & \text{else,} \end{cases}$$

where $\gamma \in \mathbb{R}^+$ is a positive constant and $0 < \Gamma \leq 1$ is the step size which is used to make a trade-off between the convergence speed and the steady state error. Hence by this condition, a new update occurs only when the current error vector $\mathbf{e}_k$ has elements which are greater than $\frac{\gamma}{\sqrt{N_r}}$ in magnitude. Note that $\sqrt{N_r}$ works such that if $\|\mathbf{e}_k\|_{\max} \leq \frac{\gamma}{\sqrt{N_r}}$ then $\|\mathbf{e}_k\| \leq \gamma$. However this is a more stringent constraint on the error than $\|\mathbf{e}_k\| \leq \gamma$ as it imposes a single element adjustment. Another choice is to set $\mathbf{G}_k = \mathbf{G}'_k$ where

$$\mathbf{G}'_k := [\boldsymbol{\epsilon}_{k-P+1}\ \boldsymbol{\epsilon}_{k-P+2}\ \ldots\ \boldsymbol{\epsilon}_{k-1}\ \mathbf{g}'_k] \quad (13)$$

$$\mathbf{g}'_k := [g_{11}'^{(k)}\ g_{21}'^{(k)}\ \ldots\ g_{N_r 1}'^{(k)}]^{\mathrm{T}}, \ \mathbf{e}_k := [e_{11}^{(k)}\ e_{21}^{(k)}\ \ldots\ e_{N_r 1}^{(k)}]^{\mathrm{T}},$$

$$g_{i1}'^{(k)} := \min\Big\{|e_{i1}^{(k)}|, \frac{\gamma}{\sqrt{N_r}}\Big\}\frac{e_{i1}^{(k)}}{|e_{i1}^{(k)}|} \quad ,\forall i \in \mathbb{F}_1^{N_r}.$$

This is according to the fact that since we already know $\mathbf{H}_k \in \mathbb{H}_{k-i+1}$, setting $\mathbf{g}_{k-i+1} = \boldsymbol{\epsilon}_{k-i+1}, \forall i \in \mathbb{F}_2^P$ results in

$$\|\mathbf{g}_{k-i+1}\|_{\max} = \|\mathbf{r}_{k-i+1} - \mathbf{H}_k\mathbf{s}_{k-i+1}\|_{\max} \leq \frac{\gamma}{\sqrt{N_r}}, \forall i \in \mathbb{F}_2^P$$

and eventually $\mathbf{G}_k$ is bounded so that $\|\mathbf{g}_{k-i}\| \leq \gamma \quad ,\forall i \in \mathbb{F}_0^{P-1}$. This adapted strategy (see [5]) for setting $\mathbf{G}'_k$ causes the matrix $\mathbf{E}_k - \mathbf{G}'_k$ required for updating $\mathbf{Z}_k$ to be always like $\mathbf{E}_k - \mathbf{G}'_k = [\mathbf{0}_{N_r \times 1}\ \mathbf{0}_{N_r \times 1}\ \ldots\ \mathbf{0}_{N_r \times 1}\ \mathbf{e}_k - \mathbf{g}'_k]$, where the zero vectors and the zero entries of $\mathbf{e}_k - \mathbf{g}'_k$ simplify the subsequent calculations. Despite (13) can be interpreted as a generalized case of the corresponding choices in [5], one may use different criteria. For example, by defining $\boldsymbol{\Theta}$, $\mathbb{H}_k$ as

$$\boldsymbol{\Theta} := \bigcap_{(\mathbf{r},\mathbf{s})\in\mathbb{S}} \{\mathbf{H} \in \mathbb{C}^{N_r \times N_s} : \|\mathbf{r} - \mathbf{H}\mathbf{s}\| \leq \gamma\},$$

$$\mathbb{H}_k := \{\mathbf{H} \in \mathbb{C}^{N_r \times N_s} : \|\mathbf{r}_k - \mathbf{H}\mathbf{s}_k\| \leq \gamma\}, \quad (14)$$

a more flexible updating condition is obtained by

$$\mathbf{H}_{k+1} = \begin{cases} \mathbf{H}_k + \Gamma\mathbf{Z}_k & \text{if } \|\mathbf{e}_k\| > \gamma; \\ \mathbf{H}_k & \text{else.} \end{cases}$$

Keeping $\mathbf{g}_{k-j} = \boldsymbol{\epsilon}_{k-j}, \forall j \in \mathbb{F}_1^{P-1}$ as before, in this case the nearest boundary of $\mathbb{H}_k$ for the solution to lie on is obtained by minimizing the distance function $\|\mathbf{g}_k - \mathbf{e}_k\|^2$ subject to the constraint $\|\mathbf{g}_k\|^2 = \gamma^2$. Again using the method of Lagrange multipliers, since at the optimum point $\nabla\mathcal{B}=\mathbf{0}$, where

$$\mathcal{B}(\mathbf{g}_k,\delta) := \|\mathbf{g}_k-\mathbf{e}_k\|^2+\delta(\|\mathbf{g}_k\|^2-\gamma^2)\,,\gamma\in\mathbb{R},$$

one can simply show that the closest boundary to set $\mathbf{g}_k$, denoted by $\mathbf{g}_k^*$, is achieved when $\mathbf{g}_k^*=\gamma\frac{\mathbf{e}_k}{\|\mathbf{e_k}\|}$. Figure 2 illustrates the geometrical explanation where the red point indicates the closest point of the hyper-sphere created by

$$\|\mathbf{g}_k\|^2=\sum_{i=1}^{N_r}|g_{i1}^{(k)}|^2=\sum_{i=1}^{N_r}\mathrm{Re}\{g_{i1}^{(k)}\}^2+\mathrm{Im}\{g_{i1}^{(k)}\}^2=\gamma^2$$

to the tip of the vector $\mathbf{e}_k$. So for the recent case we should set $\mathbf{G}_k=\mathbf{G}_k^*$ where $\mathbf{G}_k^*:=[\boldsymbol{\epsilon}_{k-P+1}\,\boldsymbol{\epsilon}_{k-P+2}\,\ldots\,\boldsymbol{\epsilon}_{k-1}\,\gamma\frac{\mathbf{e}_k}{\|\mathbf{e_k}\|}]$. By this choice, $\mathbf{E}_k-\mathbf{G}_k^*$ has the structure below

$$\mathbf{E}_k-\mathbf{G}_k^*:=[\mathbf{0}_{N_r\times 1}\,\mathbf{0}_{N_r\times 1}\,\ldots\,\mathbf{0}_{N_r\times 1}\,(1-\frac{\gamma}{\|\mathbf{e_k}\|})\mathbf{e}_k]$$

where the sparsity significantly reduces the computation load for subsequent calculations. In spite of this, since we update only when $\|\mathbf{e}_k\|>\gamma$, the entries of last column vector are zero if and only if the corresponding entries of $\mathbf{e}_k$ are zero. Consequently, $\mathbf{E}_k-\mathbf{G}_k^*$ here is not as sparse as $\mathbf{E}_k-\mathbf{G}_k'$ in the previous section. Therefore, if simplicity prevails over the SSE value, we can set $\mathbf{G}_k=\mathbf{G}_k''$, where

$$\mathbf{G}_k'':=[\boldsymbol{\epsilon}_{k-P+1}\,\boldsymbol{\epsilon}_{k-P+2}\,\ldots\,\boldsymbol{\epsilon}_{k-1}\,\mathbf{0}_{N_r\times 1}]$$

so that the current error vector $\mathbf{e}_k$ is completely considered for updating and also the computation of $\gamma\frac{\mathbf{e}_k}{\|\mathbf{e_k}\|}$ is not required.

C. Hybrid Affine Projection (HAP)

The idea behind HAP and its set-membership version (SM-HAP) is to combine different AP algorithms to achieve a fast convergence while the final misadjustment is comparable to that of the NLMS/SM-NLMS algorithm. Thus, we can use the AP algorithm with $P>1$ for all time steps $k\in\mathbb{F}_1^{\Delta}$ where $\Delta\in\mathbb{F}_1^{K-1}$ and thereafter, the AP or SM-AP algorithm with $P=1$ for $k\in\mathbb{F}_{\Delta+1}^{K}$. Examples of HAP and SM-HAP are shown in the simulations of Section IV.

D. Resolving Matrix Inversion Issues

Since finding $\mathbf{Z}_k$ from (6) is a special case of the equation

$$\mathbf{A}_1\mathbf{V}\mathbf{B}_1+\mathbf{C}_1\mathbf{W}\mathbf{D}_1+\mathbf{A}_2\overline{\mathbf{V}}\mathbf{B}_2+\mathbf{C}_2\overline{\mathbf{W}}\mathbf{D}_2$$
$$+\mathbf{A}_3\mathbf{V}^{\mathrm{H}}\mathbf{B}_3+\mathbf{C}_3\mathbf{W}^{\mathrm{H}}\mathbf{D}_3+\mathbf{A}_4\mathbf{V}^{\mathrm{T}}\mathbf{B}_4+\mathbf{C}_4\mathbf{W}^{\mathrm{T}}\mathbf{D}_4=\mathbf{E}$$

where $\mathbf{A}_1,\mathbf{A}_2,\mathbf{C}_1,\mathbf{C}_2\in\mathbb{C}^{m\times r}$, $\mathbf{B}_1,\mathbf{B}_2,\mathbf{D}_1,\mathbf{D}_2\in\mathbb{C}^{s\times n}$, $\mathbf{A}_3,\mathbf{A}_4,\mathbf{C}_3,\mathbf{C}_4\in\mathbb{C}^{m\times s}$, $\mathbf{B}_3,\mathbf{B}_4,\mathbf{D}_3,\mathbf{D}_4\in\mathbb{C}^{r\times n}$, $\mathbf{E}\in\mathbb{C}^{m\times r}$ are given matrices and $\mathbf{V},\mathbf{W}\in\mathbb{C}^{r\times s}$ are the matrices to be determined, we can use the finite iterative algorithm proposed by [6] to find a solution for $\mathbf{Z}_k$ with *arbitrary* accuracy. The algorithm for our problem is summarized in Algorithm 1. Note that Algorithm 1 finds only one solution for equation $\mathbf{Z}_k\mathbf{S}_k=\mathbf{E}_k-\mathbf{G}_k$ if there exists any; therefore we should use it only for the case A where $\mathbf{S}_k$ is square and has full rank. For the case 3 the algorithm can be used to find the unique solution for the equation $\mathbf{Z}'\mathbf{S}_k^{\mathrm{H}}\mathbf{S}_k=\mathbf{I}_{P\times P}$ where $\mathbf{I}_{P\times P}$ is the identity matrix and $\mathbf{Z}'$ is the matrix to be determined. Now $\mathbf{Z}_k$ for case 3 is found by $\mathbf{Z}_k=(\mathbf{E}_k-\mathbf{G}_k)\mathbf{Z}'\mathbf{S}_k^{\mathrm{H}}$.

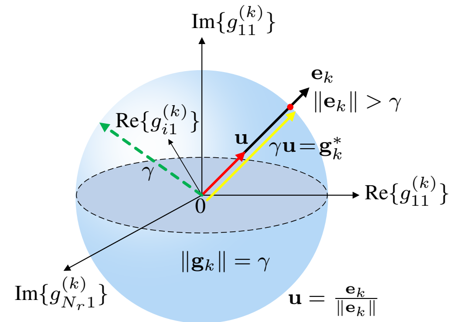

Fig. 2. The supposed hyper-sphere created by $\|\mathbf{g}_k\|=\gamma$ and the current error vector $\mathbf{e}_k$ with $\|\mathbf{e}_k\|>\gamma$. The yellow vector $\gamma\mathbf{u}$ is the solution to set $\mathbf{g}_k$ in case the nearest boundary is interested for (14).

Algorithm 1. Solving $\mathbf{Z}_k\mathbf{S}_k=\mathbf{E}_k-\mathbf{G}_k$

(1) Choose arbitrary matrix $\widehat{\mathbf{Z}}_0$

(2) Set $\mathbf{L}_0=\mathbf{E}_k-\mathbf{G}_k-\widehat{\mathbf{Z}}_0\mathbf{S}_k$ and $\mathbf{J}_0=\mathbf{L}_0{\mathbf{S}_k}^{\mathrm{H}}$

(3) If $\|\mathbf{L}_0\|_{\mathrm{F}}^2\le\xi$, set $\mathbf{Z}_k=\widehat{\mathbf{Z}}_0$ then stop; else go to (4)

(4) Set

$$\widehat{\mathbf{Z}}_{n+1}=\widehat{\mathbf{Z}}_n+\frac{\|\mathbf{L}_n\|_{\mathrm{F}}^2}{\|\mathbf{J}_n\|_{\mathrm{F}}^2}\mathbf{J}_n$$
$$\mathbf{L}_{n+1}=\mathbf{E}_k-\mathbf{G}_k-\widehat{\mathbf{Z}}_{n+1}\mathbf{S}_k$$
$$\mathbf{J}_{n+1}=\mathbf{L}_{n+1}{\mathbf{S}_k}^{\mathrm{H}}+\frac{\|\mathbf{L}_{n+1}\|_{\mathrm{F}}^2}{\|\mathbf{L}_n\|_{\mathrm{F}}^2}\mathbf{J}_n$$

(5) If $\|\mathbf{L}_{n+1}\|_{\mathrm{F}}^2\le\xi$, set $\mathbf{Z}_k=\widehat{\mathbf{Z}}_{n+1}$ then stop; else let $n=n+1$ and go to step (4).

IV. Simulations

In this section, we consider a 2-hop WSN ($L=1$) with $N_s=5\,,N_{u_1}=6$ and $N_r=5$. The channel matrices $\mathbf{H}_{\mathrm{Tx}\to 1}$ and $\mathbf{H}_{1\to\mathrm{Rx}}$ are of size 6×5 and 5×6, respectively. The SM-AP algorithm based on the update criterion $\|\mathbf{e}_k\|>\gamma$ and $\mathbf{G}_k=\mathbf{G}_k''$, along with the AP algorithm were used to estimate the complex channel matrix $\mathbf{H}_{1\to\mathrm{Rx}}$ in the second hop of the WSN and by the aid of (12). The SNR was set to 30dB and the original matrices $\mathbf{H}_{1\to\mathrm{Rx}}$ for producing the received vectors $\mathbf{r}_i$ were chosen randomly with uniform distribution of phase and amplitude in intervals $(-\pi,\pi)$ and (0,1), respectively. Also the pilots $\mathbf{s}_i$ were generated from a normal distribution of real and imaginary parts, each with mean zero and variance 1. To conform to the notation of the context, we denote the N_{u_1} by N_s for this hop. Figure 3 illustrates the effect of increasing the error bound γ on lowering the average update rate per step k ($\overline{\beta}$) for SM-AP with $P=2$ and $\Gamma=1$, where MSE is the mean squared error. Figure 4 shows how the tradeoff between the convergence speed and SSE can be handled by the step-size value Γ for AP$\,,P=2$. Figure 5 shows the MSE performance of different algorithms, all with the same step-size $\Gamma=1$, where for each SM-AP algorithm, γ^2 is set to 0.02. As in the real-valued and vector-based cases [8], the convergence speed and the misadjustment increase in P, so that the MSE of corresponding algorithms with larger P decline faster than that of the algorithms with lower P, at cost of a higher SSE. Note that the AP and SM-AP algorithms with $P=1$ are the same as the NLMS and SM-NLMS algorithms, respectively. Therefore,

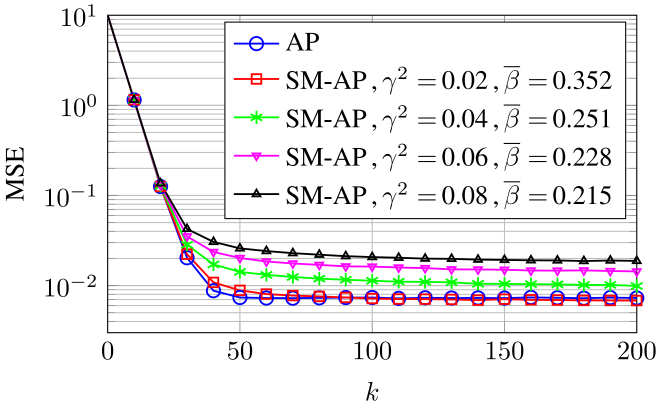

Fig. 3. The effect of increasing γ on $\overline{\beta}$ and MSE of SM-AP , $P = 2$, $\Gamma = 1$.

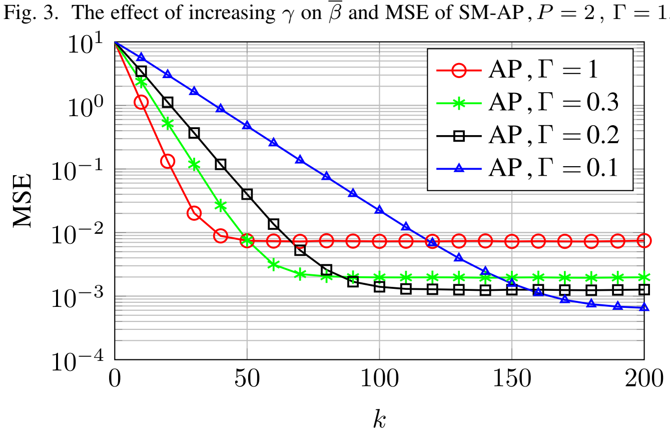

Fig. 4. The trade-off between the SSE and convergence speed for AP , $P = 2$.

as expected, our proposed algorithm converges faster than the NLMS/SM-NLMS algorithms. The most important conclusion is that we can achieve a performance from the SM-AP which is very similar to the corresponding AP algorithm, but with much less computational complexity due to the systematic matrix sparsity $(\mathbf{E}_k - \mathbf{G}''_k)$ and the selective updates $(\|\mathbf{e}_k\| > \gamma)$. In Figure 6, using $\Gamma = 1$ for all algorithms, the $AP , P = 4$ algorithm with and without using Algorithm 1 were simulated where $\overline{\alpha}$ is the average number of loops in Algorithm 1 per step k in our proposed algorithm and $\xi = 10^{-10}$. The similarity of these two curves shows that the matrix inversion process was successfully replaced by Algorithm 1 with a reasonable $\overline{\alpha}$ value. Note that the HAP (combination of AP , $P = 4$ and AP , $P = 1$) and SM-HAP (combination of AP , $P = 4$ and SM-AP , $P = 1$) algorithms with $\Delta = 20$ in Figure 6 have both of the advantages of AP and NLMS/SM-NLMS algorithms, i.e., fast convergence and low SSE. Moreover, compared to the pure AP algorithms (here AP , $P = 4$), they are simpler. In practice, finding the suitable Δ value is outside the scope of this article and can be a topic of future work.

V. Conclusions

An SM channel estimation algorithm based on AP was proposed to estimate the complex channels in a WSN. The proposed algorithm is able to estimate any channel modeled as a complex matrix in the presence of AWGN, as opposed to the conventional real-valued and vector-based investigations ([5],[8]) using SM-AP. The matrix-sparsity feature and the selective update property in SM-AP can significantly reduce the complexity. Moreover, we showed that the matrix inversion process can be efficiently converted to an iterative algorithm [6] when the inversion complexity is high for the WSN. Similar to the real-valued cases, the convergence speed and the misadjustment increase in the data reuse number P.

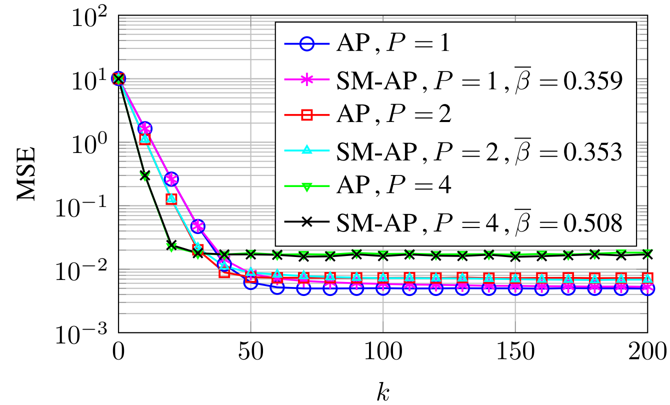

Fig. 5. MSE performance of AP and SM-AP with $\gamma^2 = 0.02$, $\Gamma = 1$.

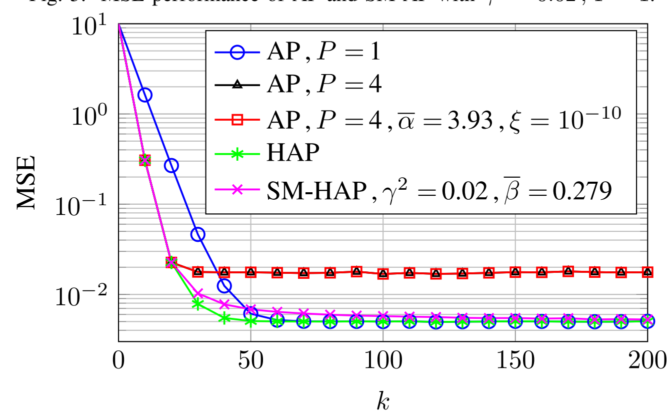

Fig. 6. The effectiveness of using Algorithm 1 as a substitution for the matrix inversion process $(\mathbf{S}_k^{\mathrm{H}}\mathbf{S}_k)^{-1}$ and the MSE performance of HAP and SM-HAP both with $\Delta = 20$ compared with the regular AP channel estimation algorithms. For all algorithms in this figure, we set $\Gamma = 1$.

References

[1] I.F. Akyildiz *et al.*, "A survey on sensor networks," *IEEE Commun. Mag.*, vol 40, no. 8, pp.102-114, Aug. 2002.

[2] T. Wang *et al.*, "Low-complexity set-membership channel estimation for cooperative wireless sensor networks," in *IEEE Trans. Veh. Technol.*, vol. 60, no. 6, pp. 2594-2607, Jul. 2011.

[3] A. Soni *et al.*, "Two-dimensional set membership affine projection adaptive cahnnel estimatin for OFDM systems," in *Proc. ICT National Conf.: Theory, Applications and Practices (ICTTAP)*, 2010.

[4] R.C. Mishra *et al.*, "MIMO channel equalization using set-membership affine projection algorithm," in *TENCON IEEE Region 10 Conference*, Cebu, 2012, pp.1 - 5.

[5] S. Werner, P. S.R. Diniz, "Set-membership affine projection algorithm," in *IEEE Signal Process. Lett.*, vol. 8, no. 8, pp. 231 - 235, Aug. 2001.

[6] M. A. Ramadan *et al.*, "Finite Iterative Algorithm for Solving a Complex of Conjugate and Transpose Matrix Equation," in *Journal of Discrete Mathematics*, vol. 2013, Article ID 170263, 13 pages, 2013. doi:10.1155/2013/170263

[7] Yao-Win Hong *et al.*, "Cooperative Communications in Resource-Constrained Wireless Networks," in *IEEE Signal Processing Mag.*, vol. 24, no. 3, pp. 47 - 57, May. 2007.

[8] Paulo S.R. Diniz, "Data-Selective Adaptive Filtering," in *Adaptive Filtering Algorithms and Practical Implementation*, 3rd ed. New York, Springer, 2008, ch. 6, pp. 231-287.