EFFICIENT SENSOR SUBSET SELECTION AND LINK FAILURE RESPONSE FOR LINEAR MMSE SIGNAL ESTIMATION IN WIRELESS SENSOR NETWORKS

Alexander Bertrand and Marc Moonen*

Dep. Electrical Engineering (ESAT/SCD-SISTA), Katholieke Universiteit Leuven
Kasteelpark Arenberg 10, B-3001, Leuven, Belgium
email: alexander.bertrand@esat.kuleuven.be, marc.moonen@esat.kuleuven.be

ABSTRACT

We consider two aspects of linear MMSE signal estimation in wireless sensor networks, i.e. sensor subset selection and link failure response. Both aspects are of great importance in low-delay signal estimation with high sampling frequency, where the estimator must be quickly updated in case of a link failure, and where sensor subset selection allows for a significant energy saving. Both problems are related since they require knowledge of the new optimal estimator when sensors are removed or added. We derive formulas to efficiently compute the optimal fall-back estimator in case of a link failure. Furthermore, we derive formulas to efficiently monitor the utility of each sensor signal that is currently used in the estimation, and the utility of extra sensor signals that are not yet used. Simulation results demonstrate that a significant amount of energy can be saved at the cost of a slight decrease in estimation performance.

1. INTRODUCTION

A wireless sensor network (WSN) consists of a large number of sensor nodes that are (usually randomly) deployed in an environment, and where each node has a wireless link to exchange data with neighbouring nodes [1]. The sensor nodes cooperate to perform a certain task such as signal estimation, detection, localization, etc. For this task, the data of the different sensors can be centralized in a so-called fusion center, or it can be partially or fully distributed over the different nodes in the network.

In this paper, we consider the case where a WSN is used for adaptive linear minimum mean squared error (MMSE) signal estimation, where the goal is to recover an unknown signal from noisy sensor observations. By using a WSN, a large area can be covered, yielding a significant amount of spatial information. This additional spatial information may result in an improved estimation performance compared to beamforming systems with small local arrays. However, WSN's often suffer from link failures, e.g. due to power shortage or interference in the wireless communication. For real-time signal estimation, the network must be able to swiftly adapt to these link failures to maintain sufficient estimation quality. In this paper, we provide an efficient procedure to compute the optimal fall-back estimators in case of a link failure, by exploiting the knowledge of the inverse sensor signal correlation matrix as used before the link failure. Due to the low complexity of the procedure, sensor nodes are able to react very quickly to link failures, even for high data rate applications such as in acoustic WSN's for speech enhancement [2,3].

*Alexander Bertrand is supported by a Ph.D. grant of the I.W.T. (Flemish Institute for the Promotion of Innovation through Science and Technology). This research work was carried out at the ESAT Laboratory of Katholieke Universiteit Leuven, in the frame of K.U.Leuven Research Council CoE EF/05/006 Optimization in Engineering (OPTEC), Concerted Research Action GOA-MaNet, the Belgian Programme on Interuniversity Attraction Poles initiated by the Belgian Federal Science Policy Office IUAP P6/04 (DYSCO, 'Dynamical systems, control and optimization', 2007-2011), Research Project FWO nr. G.0600.08 ('Signal processing and network design for wireless acoustic sensor networks'). The scientific responsibility is assumed by its authors.

As the sensors in a WSN are usually battery-powered, energy efficiency is of great importance. To prolong the life-time of the network, it is therefore important to only use those sensors that yield a significant contribution to the signal estimation process, while putting other sensors to sleep. This is the well known sensor subset selection problem. The sensor subset selection problem is also important in bandwidth constrained WSN's where each node can only transmit a subset of its available sensor signals. This is for instance the case in wireless binaural hearing aids with multiple microphones, where each hearing aid can only transmit a single microphone signal through the wireless link [3–5]. Notice that a quick link failure response is also an important aspect in this application.

Solving the sensor subset selection problem is generally computationally expensive due to its combinatorial nature. If the sensor signal statistics are known in advance, e.g. after an initial training phase, the sensor selection can be solved off line with unlimited power. However, in adaptive untrained WSN's the problem has to be solved during operation of the estimation algorithm. In this case, due to the limited power of a WSN, the sensor subset selection must be performed in an efficient way, generally yielding a suboptimal solution. We provide efficient closed-form formulas to compute the contribution of each sensor signal to the mean squared error (MSE) cost, i.e. the utility of each sensor signal, which can then be used in an adaptive greedy fashion to sequentially add or remove sensors in the estimation procedure. Simulation results demonstrate that a significant amount of energy can be saved in this way, at the cost of a slight decrease in estimation performance.

The paper is organized as follows. In section 2, we briefly review the linear MMSE (LMSSE) signal estimation procedure, and address some of the aspects in adaptive LMMSE estimation. In section 3, we derive a formula to efficiently compute the optimal fall-back estimator in case of a link failure. In section 4, we describe an efficient procedure to monitor the utility of the sensor signals used in the current estimator, and to compute the potential utility of sensor signals not currently used. Simulation results are given in section 5. Conclusions are drawn in section 6.

2. REVIEW OF LINEAR MMSE SIGNAL ESTIMATION

In this section, we briefly review linear MMSE signal estimation, which is often used in signal enhancement [2–9]. We consider an ideal WSN with M sensors. Without loss of generality, we assume that all sensor signals are centralized in a fusion center. However, the results in this paper can be equally applied to the distributed case where each sensor node solves a local LMMSE problem, as in [2–4,8–11]. Sensor k collects observations of a complex[1] valued signal $y_k[t]$, where $t \in \mathbb{N}$ is the discrete time index. For the sake of an easy exposition, we will mostly omit the time index in the sequel. We assume that all sensor signals and the desired signal, are stationary and ergodic. In practice, the stationarity and ergodicity assumption can be relaxed to short-term stationarity and ergodicity, in which case the theory should be applied to finite signal segments

[1]Throughout this paper, all signals are assumed to be complex valued to permit frequency-domain descriptions, e.g. when using a short-time Fourier transform (STFT).

that are assumed to be stationary and ergodic. We define $\mathbf{y}$ as the M-channel signal gathered at the fusion center in which all signals $y_k, \forall k \in \{1,\ldots,M\}$, are stacked.

The goal is to estimate a complex valued desired signal d from the sensor signal observations $\mathbf{y}$. We consider the general case where d is not an observed signal, i.e. it is assumed to be unknown, as it is the case in signal enhancement (e.g. in speech enhancement, d is the speech component in a noisy reference microphone signal). We consider LMMSE signal estimation, i.e. a linear estimator $\overline{d} = \hat{\mathbf{w}}^H \mathbf{y}$ that minimizes the MSE cost function

$$J(\mathbf{w}) = E\{|d - \mathbf{w}^H \mathbf{y}|^2\} \tag{1}$$

i.e.

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} J(\mathbf{w}) \tag{2}$$

where $E\{.\}$ denotes the expected value operator and where the superscript H denotes the conjugate transpose operator[2]. It is noted that the above estimation procedure does not use multi-tap estimation, i.e. it does not explicitly exploit temporal correlation. However, this can be easily included by expanding $\mathbf{y}$ with delayed copies of itself. Expression (1) can also be viewed as a frequency domain description, such that it defines an estimator for a specific frequency bin. When (2) is solved for each individual frequency bin, this is equivalent to multi-tap estimation. In its multi-tap form, the solution of (2) is often referred to as a multi-channel Wiener filter (MWF) [6, 7].

Assuming that the correlation matrix $\mathbf{R}_{yy} = E\{\mathbf{y}\mathbf{y}^H\}$ has full rank[3], the unique solution of (2) is [12]:

$$\boxed{\hat{\mathbf{w}} \;=\; \mathbf{R}_{yy}^{-1}\mathbf{r}_{yd}} \tag{3}$$

with $\mathbf{r}_{yd} = E\{\mathbf{y}d^*\}$, where d^* denotes the complex conjugate of d. The MMSE corresponding to this optimal estimator is

$$J(\hat{\mathbf{w}}) = P_d - \mathbf{r}_{yd}^H \mathbf{R}_{yy}^{-1}\mathbf{r}_{yd} \tag{4}$$

$$= P_d - \mathbf{r}_{yd}^H \hat{\mathbf{w}} \tag{5}$$

with $P_d = |d|^2$. Based on the assumption that the signals are ergodic, $\mathbf{R}_{yy}$ can be adaptively estimated from the sensor signal observations by time averaging. Since d is assumed to be unknown, the estimation of the correlation vector $\mathbf{r}_{yd}$ has to be done indirectly, based on application-specific strategies, e.g. by exploiting the on-off behavior of the target signal (as often done in speech enhancement [2, 3, 6]), by periodic broadcasts of known training sequences, or by incorporating prior knowledge on the signal statistics in case of partially static scenarios [10]. In the sequel, we assume that both $\mathbf{R}_{yy}$ and $\mathbf{r}_{yd}$ are known, or that both can be estimated adaptively.

Notice that the inverse of $\mathbf{R}_{yy}$ is required for the computation of (3), rather than the matrix $\mathbf{R}_{yy}$ itself. When M is large, computing this matrix inverse is however computationally expensive, i.e. $O(M^3)$, and should be avoided in adaptive applications with high data rates. Let $\mathbf{R}_{yy}[t]$ denote the estimate of $\mathbf{R}_{yy}$ at time t. Instead of updating $\mathbf{R}_{yy}[t]$ for each new sample $\mathbf{y}[t]$, and recomputing the full matrix inversion $\mathbf{R}_{yy}^{-1}[t] = \left(\mathbf{R}_{yy}[t]\right)^{-1}$, the previous matrix $\mathbf{R}_{yy}^{-1}[t-1]$ is directly updated. For example, $\mathbf{R}_{yy}$ is often estimated by means of a forgetting factor $0 < \lambda < 1$, i.e.

$$\mathbf{R}_{yy}[t] = \lambda \mathbf{R}_{yy}[t-1] + (1-\lambda)\mathbf{y}[t]\mathbf{y}[t]^H \ . \tag{6}$$

[2]In the sequel, we use the superscript T to denote the normal transpose, i.e. without conjugation.

[3]This assumption is mostly satisfied in practice because of a noise component at every sensor that is independent of other sensor signals, e.g. thermal noise. If not, pseudo-inverses should be used.

In this case, $\mathbf{R}_{yy}^{-1}[t]$ can be recursively updated by means of the matrix inversion lemma, a.k.a. the Woodbury identity [12], yielding

$$\mathbf{R}_{yy}^{-1}[t] = \frac{1}{\lambda}\mathbf{R}_{yy}^{-1}[t-1] - \frac{\mathbf{R}_{yy}^{-1}[t-1]\mathbf{y}[t]\mathbf{y}[t]^H\mathbf{R}_{yy}^{-1}[t-1]}{\frac{\lambda^2}{1-\lambda} + \lambda \mathbf{y}[t]^H \mathbf{R}_{yy}^{-1}[t-1]\mathbf{y}[t]} \tag{7}$$

which has a computational complexity of $O(M^2)$. It is noted that, when (7) is used to update $\mathbf{R}_{yy}^{-1}[t]$, the correlation matrix $\mathbf{R}_{yy}[t]$ itself does not need to be kept in memory.

3. LINK FAILURE RESPONSE

Now assume a link failure with sensor k during operation of the estimation process. This means that the fusion center now only has access to the $(M-1)$-channel signal $\mathbf{y}_{-k}$, which is defined as the vector $\mathbf{y}$ with y_k removed. In this case, the optimal LMMSE solution is

$$\hat{\mathbf{w}}_{-k} = \mathbf{R}_{yy-k}^{-1}\mathbf{r}_{yd-k} \tag{8}$$

where $\mathbf{R}_{yy-k} = E\{\mathbf{y}_{-k}\mathbf{y}_{-k}^H\}$ and $\mathbf{r}_{yd-k} = E\{\mathbf{y}_{-k}d^*\}$. Hence, when the wireless link of sensor k breaks down, estimator $\hat{\mathbf{w}}$ (3) becomes suboptimal, and should be replaced by (8). However, computing (8) requires knowledge of $\mathbf{R}_{yy-k}^{-1}$, which is not directly available. If $\mathbf{R}_{yy}$ were kept in memory, it is possible to invert its submatrix $\mathbf{R}_{yy-k}$ to obtain $\mathbf{R}_{yy-k}^{-1}$. However, this has a large computational cost when M is large, i.e. $O\left(M^3\right)$.

In the sequel, we derive an efficient formula to compute $\hat{\mathbf{w}}_{-k}$ without knowledge of $\mathbf{R}_{yy}$, and without explicitly computing matrix inversions. As explained in section 2, we only assume that the previous estimate of $\mathbf{R}_{yy}^{-1}$ is known. For the sake of an easy exposition, but without loss of generality, we assume that $k = M$, i.e. the last element of $\mathbf{y}$ is removed. We consider a block partitioning of the inverse correlation matrix

$$\mathbf{R}_{yy}^{-1} = \left[\begin{array}{c|c} \mathbf{A}_M & \mathbf{b}_M \\ \hline \mathbf{b}_M^H & Q_M \end{array}\right] \tag{9}$$

where $\mathbf{A}_M$ is an $(M-1)\times(M-1)$ matrix, $\mathbf{b}_M$ is an $(M-1)$-dimensional vector, and Q_M is a real-valued scalar. We define a similar partitioning of the corresponding (and also assumed known) optimal LMMSE estimator $\hat{\mathbf{w}}$ (3) before the link failure with sensor M:

$$\hat{\mathbf{w}} = \left[\begin{array}{c} \mathbf{c}_M \\ \hline W_M \end{array}\right] \tag{10}$$

where $\mathbf{c}_M$ denotes the subvector containing the first $(M-1)$ elements of $\hat{\mathbf{w}}$, and where W_M defines the scaling that is applied to the sensor signal M in the estimation process. Similar to (9), we define the following block partitioning of the correlation matrix

$$\mathbf{R}_{yy} = \left[\begin{array}{c|c} \mathbf{R}_{yy-M} & \mathbf{r}_M \\ \hline \mathbf{r}_M^H & P_M \end{array}\right] \tag{11}$$

where $\mathbf{r}_M$ is an $(M-1)$-dimensional vector, and where P_M is a real-valued scalar, corresponding to the power of the signal y_M. By using the matrix inversion lemma, one can verify that the inverse of this block matrix is:

$$\mathbf{R}_{yy}^{-1} = \left[\begin{array}{c|c} \mathbf{R}_{yy-M}^{-1} + \alpha_M \mathbf{v}_M \mathbf{v}_M^H & -\alpha_M \mathbf{v}_M \\ \hline -\alpha_M \mathbf{v}_M^H & \alpha_M \end{array}\right] \tag{12}$$

with

$$\mathbf{v}_M = \mathbf{R}_{yy-M}^{-1}\mathbf{r}_M \tag{13}$$

$$\alpha_M = \frac{1}{P_M - \mathbf{r}_M^H \mathbf{v}_M} \ . \tag{14}$$

By comparing (9) and (12), we find that

$$\boxed{\mathbf{R}_{yy-M}^{-1} = \mathbf{A}_M - \tfrac{1}{Q_M}\mathbf{b}_M\mathbf{b}_M^H} \tag{15}$$

and therefore the optimal fall-back estimator is

$$\hat{\mathbf{w}}_{-M} = \left(\mathbf{A}_M - \frac{1}{Q_M}\mathbf{b}_M\mathbf{b}_M^H\right)\mathbf{r}_{yd-M} \;. \tag{16}$$

By plugging (9) and (10) into (3) we obtain

$$\mathbf{c}_M = \mathbf{A}_M\mathbf{r}_{yd-M} + R_{y_Md}\mathbf{b}_M \tag{17}$$

$$W_M = \mathbf{b}_M^H\mathbf{r}_{yd-M} + Q_M R_{y_Md} \tag{18}$$

where R_{y_Md} denotes the last element of the correlation vector $\mathbf{r}_{yd}$. When comparing (16) with (17)-(18), we find with some straightforward algebraic manipulation that the optimal fall-back estimator can be readily computed as

$$\boxed{\hat{\mathbf{w}}_{-M} = \mathbf{c}_M - \tfrac{W_M}{Q_M}\mathbf{b}_M \;.} \tag{19}$$

Since all variables in (19) are directly available, this allows a very efficient computation, i.e. $O(M)$.

Remark: The above formulas can also be used in the case where an additional sensor signal becomes available. That is, formulas (12)-(14) can be used to efficiently compute the new inverse correlation matrix $\mathbf{R}_{yy}^{-1}$ when sensor M is added in the estimation process. We will return to this in section 4.2.

4. SENSOR SUBSET SELECTION

Assume that we have an optimal M-channel LMMSE estimator $\hat{\mathbf{w}}$. The goal is now to efficiently monitor the utility of each sensor signal, i.e. we wish to identify how much the MSE cost (1) increases when a specific sensor is removed from the signal estimation procedure (sensor deletion), or how much the MSE cost decreases if a specific additional sensor would be included in the estimator (sensor addition). We will refer to this MSE cost decrease or increase as the 'utility' of the sensor signal. To allow monitoring this utility, we want to be able to compute it in an efficient way, i.e. without explicit matrix inversions and without actually computing the optimal estimator for all possible scenarios. In the case of sensor deletion, we will show that the utility of each sensor can be monitored at a computational cost which is negligible compared to the estimator update based on (7). In the case of sensor addition, the cost of monitoring the potential utility of N extra sensors is more significant, i.e. N times the cost of (7).

4.1 Sensor deletion

For sensor deletion, the goal is to monitor the contribution of each sensor to the current MSE cost. The utility of sensor k is defined as

$$U_k = J(\hat{\mathbf{w}}_{-k}) - J(\hat{\mathbf{w}}) \;. \tag{20}$$

The goal is to efficiently compute U_k, $\forall\, k \in \{1,\ldots,M\}$. From (5), and with the notations[4] introduced in section 3, we find that

$$U_M = \mathbf{r}_{yd}^H\hat{\mathbf{w}} - \mathbf{r}_{yd-M}^H\hat{\mathbf{w}}_{-M} \;. \tag{21}$$

By using (19), and by using the partitioning of $\hat{\mathbf{w}}$ as defined in (10), we can rewrite (21) as

$$U_M = R_{y_Md}^* W_M + \frac{W_M}{Q_M}\mathbf{r}_{yd-M}^H\mathbf{b}_M \;. \tag{22}$$

[4]Again, we assume that $k = M$, without loss of generality.

From (18), we find that

$$\mathbf{r}_{yd-M}^H\mathbf{b}_M = W_M^* - Q_M R_{y_Md}^* \;. \tag{23}$$

By substituting (23) in (22), we find that

$$U_M = \frac{1}{Q_M}|W_M|^2 \;. \tag{24}$$

To monitor the utility of all the sensors simultaneously, i.e. the vector $\mathbf{u} = [U_1 \ldots U_M]^T$, it is thus sufficient to monitor the squared components of the current estimator $\hat{\mathbf{w}}$, normalized with the diagonal elements of the *inverted* correlation matrix $\mathbf{R}_{yy}^{-1}$, i.e.

$$\boxed{\mathbf{u} = \mathbf{\Lambda}^{-1}|\hat{\mathbf{w}}|^2} \tag{25}$$

with

$$\mathbf{\Lambda} = \mathscr{D}\{\mathbf{R}_{yy}^{-1}\} \tag{26}$$

where the operator $\mathscr{D}\{\mathbf{X}\}$ sets all off-diagonal elements of $\mathbf{X}$ to zero, and where the element-wise operator $|\mathbf{x}|^2$ replaces all elements in the vector $\mathbf{x}$ with their squared absolute value. Expression (25) is computationally efficient, i.e. $O(M)$. Therefore, the complexity of monitoring the utility of each sensor is negligible compared to the estimator update based on (7). When the utility of a certain sensor drops below a certain threshold, this sensor can be put to sleep, and the new optimal LMMSE estimator can then be readily computed as in expression (19). The reduced inverse correlation matrix can be readily computed with (15), which is then required for future estimator updates with (7).

4.2 Sensor addition

Assume that we have an optimal MMSE estimator $\hat{\mathbf{w}}$ that linearly combines M sensor signals, and that a set of N additional sensor signals is available. Which one of these sensor signals would bring the greatest benefit to the estimator?

To use the results from section 3, we assume that the current estimator is the $(M-1)$-channel estimator $\hat{\mathbf{w}}_{-M}$. The utility of adding sensor M to the estimation process, i.e. the decrease in MSE cost, is again given by (20). However, expression (25) cannot be used in this case, since W_M is not known. Indeed, this time only $\mathbf{R}_{yy-M}^{-1}$ is kept in memory, instead of $\mathbf{R}_{yy}^{-1}$. This makes the problem of sensor addition substantially different from sensor deletion.

By using (4), we can rewrite (20) as

$$U_M = \mathbf{r}_{yd}^H\mathbf{R}_{yy}^{-1}\mathbf{r}_{yd} - \mathbf{r}_{yd-M}^H\mathbf{R}_{yy-M}^{-1}\mathbf{r}_{yd-M} \;. \tag{27}$$

By using expression (12), we find that

$$\begin{aligned}\mathbf{r}_{yd}^H\mathbf{R}_{yy}^{-1}\mathbf{r}_{yd} = \;& \mathbf{r}_{yd-M}^H\left(\mathbf{R}_{yy-M}^{-1} + \alpha_M\mathbf{v}_M\mathbf{v}_M^H\right)\mathbf{r}_{yd-M} \\ & -2\alpha_M\mathscr{R}\{\mathbf{r}_{yd-M}^H\mathbf{v}_M\} + \alpha_M|R_{y_Md}|^2\end{aligned} \tag{28}$$

where $\mathscr{R}\{X\}$ denotes the real part of X. By substituting (28) in (27), we find that the utility of sensor M can be computed as

$$U_M = \alpha_M|\mathbf{r}_{yd-M}^H\mathbf{v}_M - R_{y_Md}|^2 \;. \tag{29}$$

The computational complexity is $O(M^2)$, which is the same order of magnitude as the computation of the estimator update based on (7). Notice that, as opposed to the sensor deletion case, we now do need the cross correlation between the currently used sensor signals, and the added sensor signal y_M (used in the computation of $\mathbf{v}_M$, as given in (13)). This cannot be circumvented because the current optimal estimator only uses $\mathbf{R}_{yy-M}^{-1}$, which indeed does not incorporate any statistics of y_M.

Let us now consider the general case where N extra sensor signals become available. Define $\mathbf{y}_c$ as the stacked vector of the M

sensor signals that are currently used in the estimation process, and define $\mathbf{y}_e$ as the stacked N-channel signal that contains the N extra sensor signals that can be added to the estimation process. We redefine $\mathbf{R}_{yy}$ as

$$\mathbf{R}_{yy} = \left[\begin{array}{c|c} \mathbf{R}_{y_c y_c} & \mathbf{R}_{y_c y_e} \\ \hline \mathbf{R}_{y_c y_e}^H & \mathbf{R}_{y_e y_e} \end{array} \right] \tag{30}$$

where $\mathbf{R}_{y_c y_c} = E\{\mathbf{y}_c \mathbf{y}_c^H\}$, $\mathbf{R}_{y_c y_e} = E\{\mathbf{y}_c \mathbf{y}_e^H\}$, and $\mathbf{R}_{y_e y_e} = E\{\mathbf{y}_e \mathbf{y}_e^H\}$. We assume that $\mathbf{R}_{y_c y_c}^{-1}$ is kept in memory, since this was used in the computation of the current optimal estimator. We also assume that $\mathbf{R}_{y_c y_e}$ is available, i.e. the cross correlation between the currently used sensor signals and the extra sensor signals, which can be estimated through time averaging. Finally, we assume that the power of each additional sensor signal is known, i.e. the diagonal elements of $\mathbf{R}_{y_e y_e}$.

Similar to (29), we can compute the vector $\mathbf{u} = [U_1 \dots U_N]^T$, which gives the utility of each additional sensor signal:

$$\boxed{\mathbf{u} = \mathbf{\Sigma}^{-1} |\mathbf{V}^T \mathbf{r}_{y_c d}^* - \mathbf{r}_{y_e d}|^2} \tag{31}$$

where

$$\mathbf{V} = \mathbf{R}_{y_c y_c}^{-1} \mathbf{R}_{y_c y_e} \tag{32}$$

$$\mathbf{\Sigma} = \mathscr{D}\{\mathbf{R}_{y_e y_e}\} - \mathscr{D}\{\mathbf{R}_{y_c y_e}^H \mathbf{V}\} \tag{33}$$

and where $\mathbf{r}_{y_c d} = E\{\mathbf{y}_c d^*\}$ and $\mathbf{r}_{y_e d} = E\{\mathbf{y}_e d^*\}$. The computational complexity of (32) is the dominant part, which makes the total computational complexity $O(M^2 N)$.

Let $U_k = \max_{i \in \{1,\dots,N\}} U_i$, which means that sensor k will be selected as providing the most useful additional sensor signal. To incorporate sensor signal y_k in the estimation procedure, the inverse correlation matrix $\mathbf{R}_{y_c y_c}^{-1}$ should be replaced with $\mathbf{R}_{y_c y_c + k}^{-1} = E\{\left[\mathbf{y}_c^T y_k\right]^T \left[\mathbf{y}_c^H y_k^*\right]\}^{-1}$, which can be computed similarly to (12), i.e.

$$\boxed{\mathbf{R}_{y_c y_c + k}^{-1} = \left[\begin{array}{c|c} \mathbf{R}_{y_c y_c}^{-1} + \frac{1}{S_k} \mathbf{v}_k \mathbf{v}_k^H & -\frac{1}{S_k} \mathbf{v}_k \\ \hline -\frac{1}{S_k} \mathbf{v}_k^H & \frac{1}{S_k} \end{array} \right]} \tag{34}$$

where $\mathbf{v}_k$ denotes the k-th column of $\mathbf{V}$, and where S_k denotes the k-th diagonal element of $\mathbf{\Sigma}$. This has computational complexity $O(M^2)$, which is the same as the complexity of an estimator update according to (7). The new optimal LMMSE estimator can then be computed as

$$\boxed{\hat{\mathbf{w}}_{+k} = \mathbf{R}_{y_c y_c + k}^{-1} \left[\begin{array}{c} \mathbf{r}_{y_c d} \\ R_{y_k d} \end{array} \right]} \tag{35}$$

where $R_{y_k d}$ denotes the k-th entry in $\mathbf{r}_{y_e d}$.

4.3 Greedy sensor subset selection

The formulas (25) and (31) can be readily used in a greedy approach to efficiently determine a subset of sensor signals that yields a good estimator. This can be done in two different ways (with generally different end results). In the case of sensor addition, one starts by selecting the single sensor signal which results in the best single-channel estimator, and then in each cycle the sensor with highest utility is added to the estimation process (forward mode). In the case of sensor deletion, one starts by computing the optimal estimator using all sensor signals, and then in each cycle the sensor with lowest utility is deleted (backward mode). An adaptive greedy sensor subset selection (AGSSS) algorithm is described in more detail in the next section.

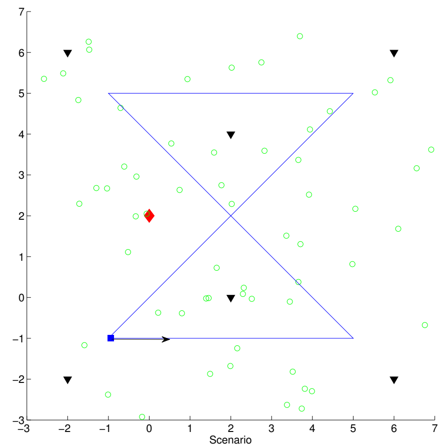

Figure 1: The simulated scenario, containing $M = 60$ sensors ($\circ$) with one reference sensor ($\diamond$), 6 noise sources ($\bigtriangledown$) and one moving target source ($\Box$).

5. SIMULATIONS

In this section, we present simulation results of an adaptive LMMSE signal estimation algorithm with adaptive greedy sensor subset selection. The scenario is depicted in Fig. 1. This is a toy scenario, and we do not attempt to model any practical setting or application. All signals are sampled with a sampling rate of 8kHz. The target source ($\Box$) moves at a speed of 0.5 m/s over the path indicated by the straight lines, and stops for 5 seconds at each corner. The target source signal is white and has a Gaussian distribution. There are six localized white Gaussian noise sources ($\bigtriangledown$) present, each with 25% of the power of the target source[5]. The WSN contains $M = 60$ randomly placed sensors ($\circ$), with one reference sensor ($\diamond$). The goal is to estimate the target source signal as it is sensed by this reference sensor (denoted by d). In addition to the spatially correlated noise, independent white Gaussian sensor noise, with 5% of the power of the target source, is added to each sensor signal. The individual signals originating from the target sources and the noise sources that are collected by a specific sensor are attenuated in power and summed. The attenuation factor of the signal power is $\frac{1}{r}$, where r denotes the distance between the source and the sensor. We assume that there is no time delay in the transmission path between the sources and the sensors[6]. The estimation performance will be assessed based on the instantaneous signal-to-error ratio, computed over $L = 1000$ samples:

$$SER[t] = 10 \log_{10} \left(\frac{\sum_{k=t-L+1}^{t} d[k]^2}{\sum_{k=t-L+1}^{t} (d[k] - \overline{d}[k])^2} \right). \tag{36}$$

The inverse correlation matrix $\mathbf{R}_{yy}^{-1}$ is updated according to (7) with a forgetting factor $\lambda = 0.9995$. The correlation vector $\mathbf{r}_{yd}$ is updated with the same forgetting factor. We use the clean desired signal d in the estimation of $\mathbf{r}_{yd}$, to isolate estimation errors. Notice that in practice, application-specific techniques are required to estimate this vector if d is not directly available[7] (see e.g. [2, 3]). During the first 3 seconds, the estimation algorithm estimates the required statistics of all sensor signals, and computes the optimal M-channel LMMSE estimator $\hat{\mathbf{w}}$ (3). After 3 seconds, an adaptive

[5]This is an arbitrary choice that yields practical SNR's at the sensors.

[6]Since there are no time delays, the spatial information is purely energy based in this case. Therefore, the fusion center cannot perform any beamforming towards specific locations by exploiting different delay paths between sources and sensors.

[7]In some applications, the signal d is directly available at certain moments in time. For example, in communications applications, known training sequences can be used to estimate $\mathbf{r}_{yd}$ during periodic training intervals.

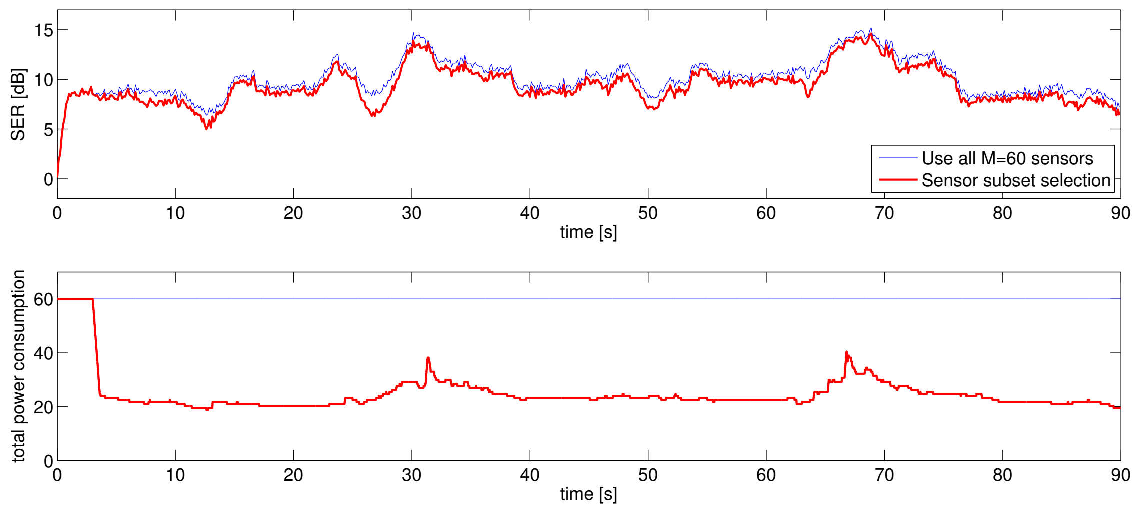

Figure 2: SER vs. time (above), and the corresponding total power consumed in the WSN (below).

greedy sensor subset selestion (AGSSS) algorithm starts running simultaneously with the adaptive LMMSE estimation process.

In the AGSSS, the utility of each currently used sensor signal is tracked using (25). If a sensor's utility drops below 1% of the MSE cost of the current estimator (computed with (5)), the sensor is put to sleep, and the inverse correlation matrix and the estimator are updated according to (15) and (19), respectively. Notice that this corresponds to a decrease in SER of maximum $10\log_{10}(1.01) = 0.043$dB for each sensor that is removed. The sensors that are put to sleep transmit their sensor signal only 25% of the time, reducing their power consumption with 75 %. The reason why sleeping sensors still transmit data, is to estimate the required statistics to compute their utility, based on (31). Once their utility exceeds 5% of the MSE cost of the current estimator, they are added again to the estimation process. This corresponds to an increase in SER of at least $-10\log_{10}(0.95) = 0.22$dB for each sensor that is added. The inverse correlation matrix and the estimator are updated according to (34) and (35), respectively.

The instantaneous SER of the resulting time-varying estimator is shown in Fig. 2, together with a plot of the total power consumption summed over all sensors. The active sensors have a power consumption of 1, and sleeping sensors have a power consumption of 0.25 (these numbers are unitless since they are not based on actual physical power consumption). The SER and power consumption of the optimal estimator that uses all $M = 60$ sensors is also added as a reference, which we will refer to as the full estimator. We observe that, due to the sensor subset selection, the SER slightly drops compared to the full estimator (on average, this is a decrease of 0.56 dB). However, due to the power saving of the sleeping sensors, the total average power consumption is only 41 % of the total power consumption of the full estimator. The average number of active sensors is 13.

6. CONCLUSIONS

In this paper, we have considered two aspects in linear MMSE signal estimation in wireless sensor networks, i.e. sensor subset selection and link failure response. We have first derived an efficient formula to compute the optimal fall-back estimator when the wireless link of one of the sensors fails. High efficiency is achieved by exploiting the knowledge of the inverse correlation matrix as used before the link failure. We have then derived an efficient formula to monitor the utility of each sensor signal in the current estimation process, which can be used for sensor deletion. We have also derived a formula to efficiently compute the potential utility of sensors that are not yet used in the estimation process, which can then be used for sensor addition. Both formulas can be used to perform an adaptive greedy sensor subset selection procedure. Simulation results of this greedy procedure in an adaptive LMMSE estimation algorithm demonstrate that a significant amount of energy can be saved, at the cost of a slight decrease in estimation performance.

REFERENCES

[1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 4, pp. 2033–2036 vol.4, 2001.

[2] A. Bertrand and M. Moonen, "Robust distributed noise reduction in hearing aids with external acoustic sensor nodes," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, Article ID 530435, 14 pages, 2009. doi:10.1155/2009/530435.

[3] S. Doclo, T. van den Bogaert, M. Moonen, and J. Wouters, "Reduced-bandwidth and distributed MWF-based noise reduction algorithms for binaural hearing aids," *IEEE Trans. Audio, Speech and Language Processing*, vol. 17, pp. 38–51, Jan. 2009.

[4] S. Srinivasan, "Noise reduction in binaural hearing aids: Analyzing the benefit over monaural systems," *The Journal of the Acoustical Society of America*, vol. 124, no. 6, pp. EL353–EL359, 2008.

[5] S. Srinivasan and A. C. den Brinker, "Rate-constrained beamforming in binaural hearing aids," *EURASIP Journal on Advances in Signal Processing, vol. 2009, Article ID 257197, 9 pages, 2009. doi:10.1155/2009/257197*.

[6] S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *Signal Processing, IEEE Transactions on*, vol. 50, pp. 2230 – 2244, Sep 2002.

[7] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New insights into the noise reduction Wiener filter," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, pp. 1218 –1234, July 2006.

[8] A. Bertrand and M. Moonen, "Distributed adaptive estimation of correlated node-specific signals in a fully connected sensor network," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP), Taipei, Taiwan*, April 2009.

[9] A. Bertrand and M. Moonen, "Distributed adaptive node-specific MMSE signal estimation in sensor networks with a tree topology," *Proc. of the European signal processing conference (EUSIPCO), Glasgow - Scotland*, August 2009.

[10] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks – part I: sequential node updating," *Accepted for publication in Signal Processing, IEEE Transactions on*, 2010.

[11] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks – part II: simultaneous & asynchronous node updating," *Accepted for publication in Signal Processing, IEEE Transactions on*, 2010.

[12] G. H. Golub and C. F. van Loan, *Matrix Computations*. Baltimore: The Johns Hopkins University Press, 3rd ed., 1996.