BLIND CHANNEL IDENTIFICATION AND EQUALIZATION IN DENSE WIRELESS SENSOR NETWORKS WITH DISTRIBUTED TRANSMISSIONS

Xiaohua(Edward) Li

Department of Electrical and Computer Engineering
State University of New York at Binghamton
Binghamton, NY 13902
Email: xli@binghamton.edu

ABSTRACT

In densely deployed wireless sensor networks, signals of adjacent sensors can be highly cross-correlated. This property is utilized for blind channel identification and equalization. Blind equalization can be performed with linear complexity and with robustness to ill channel conditions. Transmissions are more power and bandwidth efficient, which is especially important for wideband sensor networks. The cross-correlation property and the finite sample effect are analyzed. Simulations demonstrate the superior performance of the proposed method.

1. INTRODUCTION

Wireless sensor networks consist of a large amount of densely deployed sensors which are connected with each other wirelessly to form a dynamic network. Since the density may be very high, e.g., tens of sensors per square meter [1], signals from adjacent sensors are highly cross-correlated [2].

Sensors should be extremely power efficient because once deployed they may not be recharged or replaced. Since wireless transceivers consume a major portion of battery power [1], it is critical to improve their power efficiency. However, one of the major difficulties comes from the hash communication environment since sensors work in unknown environment with multipath propagation and severe fading [3]. Sophisticated techniques have to be used for reliable and efficient signal demodulation and detection. In particular, blind equalization is necessary to mitigate multipath propagation and to improve both bandwidth and energy efficiency. This is especially important for wideband sensor networks [4] such as those for acoustic location or video surveillance purposes. Unfortunately, it is mostly still an open area.

Since training sequences waste not only bandwidth but also power, blind equalization becomes more promising, especially those with the same complexity as training methods. However, many traditional blind methods may not be appropriate for sensor networks. For those based on single-input-single-output (SISO) framework, local and slow convergence [6] becomes a severe problem. In addition, many of them may not work for SISO channels with zeros on the unit circle, i.e., ill-conditioned channels.

On the other hand, for single-input-multiple-output (SIMO) methods, multi-antenna or over-sampling unnecessarily increases complexity and thus reduces power efficiency. The most severe problem is that most of them are not robust to non-ideal channel conditions such as those with common zeros among sub-channels [6].

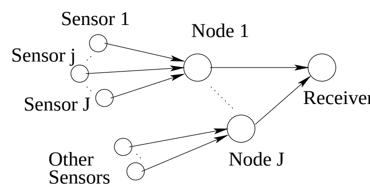

Fig. 1. System model of wireless sensor networks.

Traditional methods generally assume that signals from different users (or sensors) are uncorrelated. This is not true in densely deployed wireless sensor networks. Therefore, in this paper we show that the cross-correlation among sensors can be exploited for efficient and robust blind channel identification and equalization.

The organization of this paper is as follows. In Section 2, we introduce the communication system model. In Section 3, we derive the blind algorithms. In Section 4, we study cross-correlation property and the finite sample effect. Simulations are shown in Section 5 and conclusions are presented in Section 6.

2. PROBLEM FORMULATION

In wireless sensor networks with TDMA or similar channel access schemes where sensors transmit data packets in their own slots, we consider the case that a sensor receives signals from multiple other sensors, e.g., Node 1 receives signals from Sensor 1 to J, as illustrated in Fig. 1. For each sensor, the baseband symbols, denoted as $s_i(n)$, are transmitted through a wireless channel $\{h_i(n)\}$.

Let the received signal at the receiving node be $x_i(n)$. Define $\mathbf{x}_i^T(n) = [x_i(n), \cdots, x_i(n-N)]$, $\mathbf{s}_i^T(n) = [s_i(n), \cdots, s_i(n-N-L)]$, and $\mathbf{v}_i^T(n) = [v_i(n), \cdots, v_i(n-N)]$. Then we have

$$\mathbf{x}_i(n) = \mathcal{H}_i \mathbf{s}_i(n) + \mathbf{v}_i(n) \tag{1}$$

where the $(N+1) \times (N+L+1)$ channel matrix is

$$\mathcal{H}_i = \begin{bmatrix} h_i(0) & \cdots & h_i(L) & & \\ & \ddots & & \ddots & \\ & & h_i(0) & \cdots & h_i(L) \end{bmatrix}. \tag{2}$$

Channels of all sensors are assumed normalized.

The receiving node receives signals $x_i(n)$ of all $i = 1, \cdots, J$ sensors, from which it needs to estimate the channels $\{h_i(n)\}$, $i = 1, \cdots, J$, and perform symbol estimation.

We assume that the AWGN $v_i(n)$ is stationary with zero mean and variance $\sigma^2_{v_i}$, and is uncorrelated with symbols of all sensors. The symbols $\{s_i(n)\}$ are i.i.d. However, symbols from different sensors, e.g., $\{s_i(n)\}$ and $\{s_j(n)\}$, may have non-zero cross-correlation.

3. BLIND CHANNEL EQUALIZATION

3.1. Cross-correlation assumption of symbol sequences

Consider the transmitted signals of Sensor i and Sensor j, i.e., $s_i(n)$ and $s_j(n)$, $i \neq j$. Construct subsequences $\{s_i(n_{i\ell}), n_{i\ell} \in I_i\}$, where $I_i = \{n_{i1}, n_{i2}, \cdots, n_{iM}, \cdots\}$ and $n_{ik} < n_{im}$ if $k < m$, and similarly, $\{s_j(n_{j\ell}), n_{j\ell} \in I_j\}$.

Cross-correlation assumption: We assume that there exist indexes I_i, I_j and two sequences $\{w_i(n)\}$ and $\{w_j(n)\}$ such that

$$\begin{aligned} & E[w_i(n_{i\ell})s_i(n_{i\ell}+d_i)s_j^*(n_{j\ell}+d_j)w_j^*(n_{j\ell})] \\ & \quad = \quad r_{ij}\delta(d_i)\delta(d_j), \quad -N \leq d_i \leq N, \ -N \leq d_j \leq N, \end{aligned} \tag{3}$$

where $r_{ij} \neq 0$, $(\cdot)^*$ denotes complex conjugate.

Equation (3) defines the cross-correlation among the subsequences extracted from the transmitted symbol sequences $\{s_i(n)\}$ and $\{s_j(n)\}$. The cross-correlation becomes zero only for the subsequences defined on I_i and I_j or, in another word, with zero-shiftings $d_i = d_j = 0$.

There are several ways to assure (3) in practice. The simplest way is scrambling, a technique widely used in practical systems. If the sensors i and j need transmit two highly cross-correlated sequences $\{b_i(n)\}$ and $\{b_j(n)\}$, respectively, they can use pseudo-noise (PN) sequences $\{c_i(n)\}$ and $\{c_j(n)\}$ to scramble them. The transmitted symbols are then $s_i(n) = c_i(n)b_i(n)$ and $s_j(n) = c_j(n)b_j(n)$. The receiver can use $w_i(n) = c_i^*(n)$ and $w_j(n) = c_j^*(n)$ to perform descrambling, which gives $E[w_i(n_{i\ell})s_i(n_{i\ell} + d_i)s_j^*(n_{j\ell} + d_j)w_j^*(n_{j\ell})] = \delta(d_i)\delta(d_j)E[b_i(n_{i\ell} + d_i)b_j^*(n_{j\ell} + d_j)]$, where we assume that the PN sequences are with zero-mean and unit variance.

Some other possible ways include the application of random interleavers or direct-sequence spread-spectrum transmission [5]. To simplify the presentation, we consider the scrambling case only. In addition, we define $w_{ij}(\ell) \stackrel{\triangle}{=} w_i(n_{i\ell})w_j^*(n_{j\ell})$.

3.2. Blind channel estimation

With the knowledge about the position index I_i of Sensor i, $1 \leq i \leq J$, we choose the received sample vectors from (1) as $\mathbf{x}_i(n_{i\ell}+p) = \mathcal{H}_i\mathbf{s}_i(n_{i\ell}+p)+\mathbf{v}_i(n_{i\ell}+p)$. If p and N satisfy $L \leq p \leq N$, then the symbol $s_i(n_{i\ell})$ is corresponding to the $(p+1)^{\text{th}}$ column $\mathbf{h}_i(p)$ of the matrix $\mathcal{H}_i$ that contains all the channel coefficients (*c.f.* (2)),

$$\mathbf{h}_i(p) = [\mathbf{0}_{p-L}, h_i(L), \cdots, h_i(0), \mathbf{0}_{N-p}]^T, \tag{4}$$

where $\mathbf{0}_k$ is a k dimensional vector.

Proposition 1. Define the cross-correlation matrix between Sensor i and j as

$$\mathbf{R}_{ij} = E[\mathbf{x}_i(n_{i\ell}+p)\mathbf{x}_j^H(n_{j\ell}+p)w_{ij}(\ell)]. \tag{5}$$

If $L \leq p \leq N$ and $i \neq j$, we have

$$\mathbf{R}_{ij} = r_{ij}\mathbf{h}_i(p)\mathbf{h}_j^H(p). \tag{6}$$

Proof. From the cross-correlation assumption (3), the proof is readily available [5]. □

Consider estimating the channel of Sensor i with signals from all J sensors. From (5) we have an $(N+1) \times [(J-1)(N+1)]$ matrix

$$\mathbf{R}_i = [\mathbf{R}_{i1}, \cdots, \mathbf{R}_{i,i-1}, \mathbf{R}_{i,i+1}, \cdots, \mathbf{R}_{iJ}]. \tag{7}$$

Since from (6) each column in the matrix $\mathbf{R}_i$ is simply a weighted version of the column $\mathbf{h}_i(p)$, the matrix is with rank 1. The channel of Sensor i is available as the left eigenvector of $\mathbf{R}_i$ corresponding to its largest eigenvalue. Nevertheless, we can use the following two ways to estimate channels efficiently from (6) and (7).

The first way is to simply use a column in the matrix $\mathbf{R}_i$ with sufficiently large magnitude as channel estimation. The second way is to combine all the columns in $\mathbf{R}_i$ together recursively. To begin, we initialize with any non-zero column from $\mathbf{R}_i$, which can in fact be determined similarly as the first way. Let such a column be $\mathbf{R}_i(:, m)$, where we use the MATLAB notation to denote the m^{th} column. Then we can estimate the channel recursively as

$$\begin{aligned} \hat{\mathbf{h}}_i^{(0)} &= \mathbf{R}_i(:,m)\|\mathbf{R}_i(:,m)\|, \quad \text{if } \mathbf{R}_i(:,m) \neq \mathbf{0}, \\ \hat{\mathbf{h}}_i^{(k)} &= \begin{cases} \hat{\mathbf{h}}_i^{(k-1)} + \frac{\mathbf{R}_i(:,k)\mathbf{R}_i^H(:,k)\hat{\mathbf{h}}_i^{(k-1)}}{\|\hat{\mathbf{h}}_i^{(k-1)}\|}, & k \neq m. \\ \hat{\mathbf{h}}_i^{(k-1)}, & k = m. \end{cases} \end{aligned} \tag{8}$$

Proposition 2. The recursive procedure (8) converges to $\hat{\mathbf{h}}_i = \mathbf{h}_i(p)e^{j\theta}\sum_{0 \leq j \leq J,\ j \neq i}|r_{ij}|^2$, i.e., channel estimation with a scalar ambiguity, where θ is the phase.

Proof: See [5].

3.3. Blind equalization

Once channels are estimated blindly, we can estimate linear filter equalizers $\mathbf{f}_i$ by a constrained optimization

$$\arg\min_{\mathbf{f}_i} \ \|\mathbf{f}_i^H\tilde{\mathbf{x}}_i(n)\|^2, \quad s.t., \quad \mathbf{f}_i^H\tilde{\mathbf{h}}_i = 1, \tag{9}$$

where $\tilde{\mathbf{x}}_i(n)$ is constructed similarly as $\mathbf{x}_i(n)$ but with dimension no less than N, and $\tilde{\mathbf{h}}_i$ is an extended version of $\hat{\mathbf{h}}_i$ with zero-padding for proper equalization delay.

It is well known that (9) converges to the MMSE equalizer

$$\mathbf{f}_i = \mathbf{R}_{\tilde{x}}^{-1}\tilde{\mathbf{h}}_i(\tilde{\mathbf{h}}_i^H\mathbf{R}_{\tilde{x}}^{-1}\tilde{\mathbf{h}}_i)^{-1}, \tag{10}$$

where $\mathbf{R}_{\tilde{x}} = E[\tilde{\mathbf{x}}_i(n)\tilde{\mathbf{x}}_i^H(n)]$.

If the number of samples is limited or channels vary relatively fast, we can use batch processing to take better utilization of available samples. We first calculate correlation matrix $\mathbf{R}_i$ (5) (7), then estimate channel (8) and equalizer $\mathbf{f}_i$ (10). The matrix inversion formula can be used to avoid explicit inversion. The computational complexity is thus $O(N^2)$. It is affordable for systems with limited sample amount.

On the other hand, if the sample amount is sufficient, we can use the extremely efficient adaptive implementation. To avoid correlation matrix estimation, we use the first way, i.e., use only one column of the correlation matrix as the channel estimation

$$\hat{\mathbf{h}}_i^{(\ell)} = \beta\hat{\mathbf{h}}_i^{(\ell-1)} + \mathbf{x}_i(n_{i\ell}+p)x_j^*(n_{j\ell}+q)w_{ij}(\ell), \tag{11}$$

where β is used to track time-variation, and we need choose j and $q \in [0, L]$ online so that $\|\hat{\mathbf{h}}_i^{(\ell)}\|$ is sufficiently large. With the temporarily estimated channel, we adaptively implement (9) for equalizer estimation by the Frost's Algorithm

$$\mathbf{f}_i^{(n+1)} = \tilde{\mathbf{h}}_i^{(n)} + [\mathbf{I} - \tilde{\mathbf{h}}_i^{(n)}(\tilde{\mathbf{h}}_i^{(n)})^H][\mathbf{f}_i^{(n)} - \mu\tilde{\mathbf{x}}_i(n)\tilde{\mathbf{x}}_i^H(n)\mathbf{f}_i^{(n)}], \tag{12}$$

where $\tilde{\mathbf{h}}_i^{(n)}$ is an extended (with zero-padding) version of $\hat{\mathbf{h}}_i^{(\ell)}$, $n_{i\ell} \leq n < n_{i,\ell+1}$, and $\mathbf{I}$ is an identity matrix. The parameter μ is to adjust convergence. The computational complexity is $O(N)$.

In addition, thanks to the special cross-correlation property (3), the new algorithms are robust to non-ideal or ill channel conditions, as can be easily seen from (6) and (10).

4. CROSS-CORRELATION PROPERTIES

4.1. Source cross-correlation and symbol cross-correlation

In densely deployed wireless sensor networks, cross-correlations of the sensing values of adjacent sensors are high. It is determined by the source signals' signal-to-noise ratio (S-SNR). However, for the blind methods in Section 3, what we need is the cross-correlation among the transmitted symbols. In this section, we study quantitatively the relationship between the source signal cross-correlation and the symbol cross-correlation. To simplify the problem, we consider binary signaling and all variables are thus real.

Consider that the sensor i samples a source $z_i(m)$ with noise $u_i(m)$. The sampling values (before encoding to binary sequence) are $a_i(m) = z_i(m) + u_i(m)$, where $a_i(m)$, $z_i(m)$ and $u_i(m)$ are with zero mean, $u_i(m)$ is independent of $z_i(m)$, $u_i(m)$ and $u_j(m)$ are independent from each other if $i \neq j$. The S-SNR is defined as $\alpha = 10\log_{10}(E[z_i^2(m)]/E[u_i^2(m)])$.

Assume $|a_i(m)| \leq 2^{L_a-1}$. Then we define the normalized source signal cross-correlation between sensor i and j as

$$r_{ij}^a \triangleq \frac{1}{2^{2L_a-2}}E[a_i(m)a_j(m)], \tag{13}$$

where we consider them after synchronization, i.e., r_{ij}^a is the maximum cross-correlation. From the independence of noise, we have $r_{ij}^a = E[z_i(m)z_j(m)]/2^{2L_a-2} \leq E[z_i^2(m)]/2^{2L_a-2}$. Since $E[a_i^2(m)] = E[z_i^2(m)] + E[u_i^2(m)]$, we have

$$r_{ij}^a \leq \frac{E[a_i^2(m)]}{2^{2L_a-2}(1+10^{-\alpha/10})}. \tag{14}$$

Then, consider encoding $a_i(m) + 2^{L_a-1}$ into L_a-bit words $a_i(m) + 2^{L_a-1} = \sum_{k=0}^{L_a-1} b_{ik}(m)2^k$, where $b_{ik}(m) \in \{1, 0\}$. The symbol $s_{ik}(m)$ is $s_{ik}(m) = 2b_{ik}(m) - 1$.

Proposition 3. Assume $E[s_{ik}(m)] = E[s_{jk}(m)] = 0$. The cross-correlation of symbol sequences depends on that of source signals through

$$\sum_{k=0}^{L_a-1}\sum_{\ell=0}^{L_a-1} E[s_{ik}(m)s_{j\ell}(m)]2^{k+\ell-2L_a} = r_{ij}^a + 1 - (1-2^{-L_a})^2. \tag{15}$$

Proof. See [5].

To analyze (15), first, we skip all items within the double summation except the three ones with $E[s_{ik}(m)s_{jk}(m)]$, $k \in \{L_a - 3, L_a - 2, L_a - 1\}$, and second, we assume $E[s_{ik}(m)s_{jk}(m)] = r_{ij}^b$ for $k \in \{L_a - 3, L_a - 2, L_a - 1\}$, where r_{ij}^b is the symbol cross-correlation. Then from (15) we have

$$r_{ij}^b = \frac{64}{21}[r_{ij}^a + 1 - (1-2^{-L_a})^2]. \tag{16}$$

Since L_a is usually large enough, a thumb of rule about the relation between source signal cross-correlation and symbol cross-correlation can be obtained from (16) as

$$r_{ij}^b \approx 3r_{ij}^a. \tag{17}$$

Note that from (14) we have $r_{ij}^a \leq 1/3$.

Therefore, the symbol cross-correlation is high as long as the source cross-correlation is high. The results (16)-(17) have been verified through numerical experiments as shown in Fig. 2. $L_a = 12$. Noises are added to a random source sequence to generate sensors' sampling values. Then symbol cross-correlations are calculated by (16) for the analysis results, and by Monte-Carlo simulation for the simulated results. Since data fusion is an important concern on symbol cross-correlation, we evaluate also the cross-correlation after data fusion with a linearly constrained least squares data fusion method. Results in Fig. 2 show that the analysis results fit well to the simulated results.

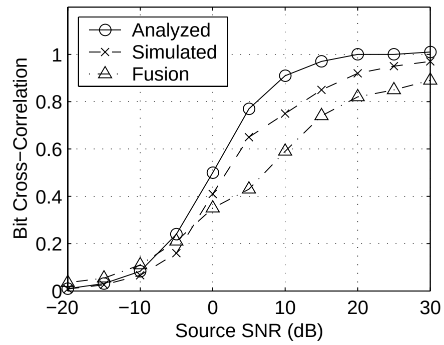

Fig. 2. Symbol (bit) cross-correlations as functions of source SNR. ○: analysis results from (16). ×: simulation results without data fusion. △: simulation results with data fusion.

4.2. Finite sample effect on blind channel estimation

In binary transmission, with finite number of samples, the cross-correlation becomes $\frac{1}{M}\sum_{\ell=1}^{M} s_i(n_{i\ell})s_j(n_{j\ell})w_{ij}(\ell) = \hat{r}_{ij}$. It is easy to show that $\hat{r}_{ij} - r_{ij}$ is a random variable with zero mean and variance

$$\sigma_{ss}^2 = \frac{E[w_{ij}^2(\ell)]}{M} \triangleq \frac{A}{M}. \tag{18}$$

For symbol cross-correlations with non-zero shiftings, we find that $\varepsilon_s(d_i, d_j) = \frac{1}{M}\sum_{\ell=1}^{M} s_i(n_{i\ell}+d_i)s_j(n_{j\ell}+d_j)w_{ij}(\ell)$ is a random variable with zero mean and variance σ_{ss}^2, where $d_i \neq 0$ or $d_j \neq 0$.

For noises $v_i(n)$ and $v_j(n)$, their cross-correlation $\varepsilon_v(d_i, d_j) = \frac{1}{M}\sum_{\ell=1}^{M} v_i(n_{i\ell}+d_i)v_j(n_{j\ell}+d_j)w_{ij}(\ell)$ is a random variable with zero mean and variance $\sigma_{v_i}^2\sigma_{v_j}^2 A/M$. Cross-correlation between symbols and noises, $\varepsilon_{sv}(d_i, d_j) = \frac{1}{M}\sum_{\ell=1}^{M} s_i(n_{i\ell}+d_i)v_j(n_{j\ell}+d_j)w_i(n_{i\ell})$ is a random variable with zero mean and variance $\sigma_{v_j}^2 E[w_i^2(n_{i\ell})]/M$, and $\varepsilon_{vs}(d_i, d_j) = \frac{1}{M}\sum_{\ell=1}^{M} v_i(n_{i\ell}+$

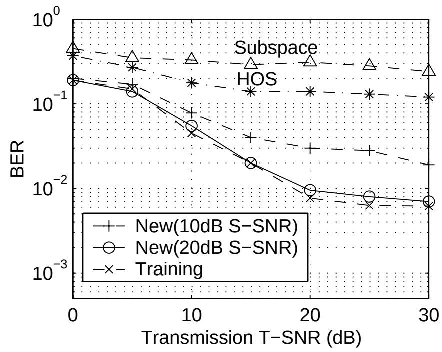

Fig. 3. BER of the batch algorithms as functions of T-SNR. 10 sensors.

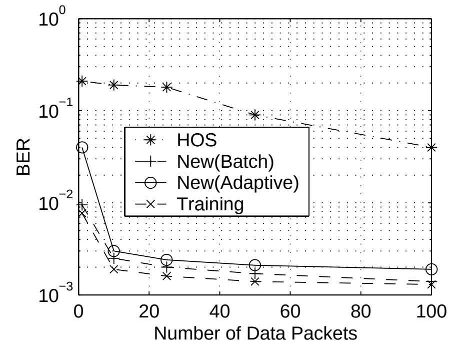

Fig. 4. BER as functions of number of data packets. S-SNR 20 dB, T-SNR 20 dB. 260 bits per data packet. 10 sensors.

$d_i)s_j(n_{j\ell}+d_j)w_j(n_{j\ell})$ is a random variable with zero mean and variance $\sigma_{v_i}^2 E[w_j^2(n_{j\ell})]/M$.

Then the estimation of cross-correlation matrix (5) becomes $\hat{\mathbf{R}}_{ij} = r_{ij}\mathcal{H}_i\mathbf{Z}_p\mathcal{H}_j^H + \mathcal{H}_i\mathbf{E}_s\mathcal{H}_j^H + \mathcal{H}_i\mathbf{E}_{sv}^H + \mathbf{E}_{vs}\mathcal{H}_j^H + \mathbf{E}_v$, where $\mathbf{Z}_p$ is diagonal with only one non-zero element which is 1 at the $(p,p)^{\text{th}}$ the entry. The $(m,n)^{\text{th}}$ elements in the matrices $\mathbf{E}_s$, $\mathbf{E}_{sv}$, $\mathbf{E}_{vs}$ and $\mathbf{E}_v$ are $\varepsilon_s(p-m,p-n)$, $\varepsilon_{sv}(p-m,p-n)$, $\varepsilon_{vs}(p-m,p-n)$ and $\varepsilon_v(p-m,p-n)$, respectively.

Consider the estimation of $\mathbf{h}_i(p)$. From (6) the best estimation is $\hat{\mathbf{h}}_i = \hat{\mathbf{R}}_{ij}\mathbf{h}_j(p)/r_{ij}$. Define the estimation mean square error (MSE) as $\text{MSE} \triangleq \sqrt{E[\|\hat{\mathbf{h}}_i - \mathbf{h}_i(p)\|^2]}$.

Proposition 4. If the signal-to-noise ratio during transmission (T-SNR$= 20\log_{10} E[|x_i(n)-v_i(n)|]/E[|v_i(n)|]$) is high, in order to have $\text{MSE} = \gamma$, the number of symbols used in correlation calculation should satisfy

$$\frac{A(N+1)}{\gamma^2 r_{ij}^2 (J-1)} \leq M \leq \frac{A(N+1)(2L+1)}{\gamma^2 r_{ij}^2 (J-1)}. \tag{19}$$

Proof. See [5].

Therefore, higher cross-correlation r_{ij}, smaller vector size N and more sensors all decrease M. In binary case, in order to have MSE$\leq \gamma$, we need $M > (N+1)/[\gamma^2(J-1)]$.

5. SIMULATIONS

We compared our new algorithms with the training-based MMSE equalizer, the cumulant-based blind algorithm (HOS) [6], the blind CMA, and the blind subspace method [5].

The source signals were some speech signals after compression by the GSM codec. BPSK was used. Channel length $L+1 = 5$. Channels for each sensor were randomly generated and could be ill-conditioned. For channel estimation, we used $N = 6$. The equalizer length was 15. For the subspace method, we assumed that each sensor had three receiving antennas.

Experiment 1. We used only one data packet (260 symbols) to evaluate the batch algorithms with finite sample amount. For our batch algorithm, we tried two S-SNR: 10 dB and 20 dB. In addition, we used $1/3$ of the symbols (i.e., 80 bits) to calculate cross-correlations. For the training method, we used 20% of the symbols, or 52 bits, for training. From Fig. 3, with 20 dB S-SNR, our blind method achieved almost the same performance as training method.

Experiment 2. We used more data packets to evaluate both our batch and adaptive algorithms, with $1/3$ of each data packet used for cross-correlation. As shown in Fig. 4, both the new batch and adaptive algorithms achieved the performance of the training method. The new adaptive algorithm rapidly converged within 10 data packets.

6. CONCLUSIONS

In this paper, we show that cross-correlation among sensors can be used to develop efficient blind channel estimation and equalization algorithms in densely deployed wireless sensor networks. Their superior performance is demonstrated by simulations. We have also analyzed the cross-correlation property of sensor signals and the effect of finite sample amount.

7. REFERENCES

[1] R. Min, M. Bhardwaj, S. Cho, N. Ickes, E. Shih, A. Sinha, A. Wang and A. Chandrakasan, "Energy-centric enabling technologies for wireless sensor networks," *IEEE Wireless Commun.*, pp. 28-39, August 2002.

[2] A. Scaglione and S. D. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," *Proc. Annual Inter. Conf. Mob. Comp. Net.*, MOBICOM, pp 140-147, 2002.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramianiam and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, pp. 102-114, Aug. 2002.

[4] H. Gharavi and K. Ban, "Multi-hop sensor network design for wide-band communications," *Proc. IEEE*, vol. 91, no. 8, pp. 1221-1234, Aug. 2003.

[5] X. Li, "Blind channel estimation and equalization in wireless sensor networks based on cross correlations among sensors," submitted to *IEEE Trans. Signal Processing*, 2003.

[6] J. Tugnait, L. Tong and Z. Ding, "Single user channel estimation and equalization," *IEEE Signal Processing Mag.*, vol. 17, no. 3, pp. 17-28, May 2000.