Self-Organized Distributed Compressive Projection in Large Scale Wireless Sensor Networks

Mihaela I. Chidean, Eduardo Morgado, Julio Ramiro, and Antonio J. Caamaño
Signal Theory and Communications Departament, Rey Juan Carlos University
Camino del Molino, s/n, 28943, Fuenlabrada, Spain, Phone: +34 91 488 8759
Email: {mihaela.chidean, eduardo.morgado, julio.ramiro, antonio.caamano}@urjc.es

***Abstract*—The optimal configuration for a Large Scale Wireless Sensor Networks (LS-WSN) is the one that minimizes the sampling rate, the CPU time and the channel accesses (thus maximizing the network lifetime), with a controlled distortion in the recovered data. Initial deployments of LS-WSN are usually not able to adapt to changing environments and rarely take into account either the spatial or temporal nature of the sensed variables, both techniques that optimize the network operation. In this work we propose the use of Self-Organized Distributed Compressive Projection (SODCP) in order to let the nodes to form clusters in a distributed and data-driven way, exploiting the spatial correlation of the sensed data. We compare the performance of this innovative technique, using actual data from the LUCE LS-WSN, with two different baselines: Centralized Compressive Projection (CCP) and Distributed Compressive Projection (DCP). The former uses no clustering, whereas the latter makes use of an a priori clustering that favors proximity and balances the number of nodes in each cluster. We show that SODCP outperforms DCP (in terms of Signal-to-Noise vs. Compression Rate). We also show that the performance of SODCP converges to that of CCP for relatively high compression rates of 55%.**

***Index Terms*—Compressive Projection, Self-Organization, Wireless Sensor Networks, Data compression, Distributed algorithms.**

I. Introduction

The operation of Large Scale Wireless Sensor Networks (LS-WSN) measuring and transmitting large datasets is an open research area due to the operational tradeoffs involved. An unrestricted measure-and-transmit policy inevitably leads to the collapse of the network [1] by interference, blocking or both. Hierarchical organization of the network redefines the limit of nodes to maintain the network in operation [2]. An additional policy of in-network processing, where only the most relevant features of the measured data in local groups of nodes are computed and transmitted. Thus *maintaining the network operative while controlling the loss of information involved* [3]. Techniques such as Distributed Source Coding seem promising [4]. However, their computational complexity is still an issue in an energy-limited environment [5]. Compressed Sensing (CS) techniques have been applied to datasets obtained from actual WSNs in order to perform signal reconstruction [6]. The sparse projections used in CS may potentially help in the reduction of transmissions from wireless sensor nodes.

In this work, we focus on the combination of two complementary strategies, 1) in-network processing and 2) network hierarchization. Regarding in-network processing, we use Compressive-Projections Principal Component Analysis (CPPCA) [7], a complexity-controlled feature extraction method that transfers the computational burden to the Data Fusion Center (DFC). With respect to the network hierarchization, the Self-Organized Distributed Compressive Projection (SODCP) algorithm, a novel data-driven self-organization clustering technique exploiting spatio-temporal correlation of the data sensed in different nodes, is proposed. The idea of using the structure of the sensed data in order to cluster wireless nodes in a WSN is not new. For example, Le et al. [9] used a dissimilarity measure to quantify the difference between the actual and the average sensed data, as the growth cluster criteria. More recently, Wang et al. [10] used an entropy-based divergence measure criterion to aggregate nodes in clusters, in order to increase the global compression gain in the network.

The main goal of this work is to show that *the proposed SODCP algorithm efficiently scales the network hierarchy with respect to the spatio-temporal correlations of the sensed field whilst controlling the fidelity of the reconstructed field*. We also show that the combination of both in-network processing and network hierarchization leads to transmission reduction (thus increasing WSN lifetime) and to relevant feature preservation at controlled complexity (thus preserving the measured information at controlled cost). The utility of CPPCA in LS-WSN and its independence with the nature of the data are assessed, through a comparison with Principal Component Analysis (PCA) and the utilization of both LS-WSN temperature data and hyperspectral imagery, respectively. Afterwards, the original CPPCA method is adapted to a clustered LS-WSN scenario and its performance is analyzed, showing the benefits of the SODCP algorithm with respect to the baselines considered (plain network and *a priori* homogeneous clustering).

Following, in section II, CPPCA and the WSN application scenarios used are described. The clustering algorithm is explained in section III and section IV is dedicated to the computer simulations. Finally we make some concluding remarks in section V.

II. System Setting

In this section, we first summarize the CPPCA method, in order to provide basic definitions. For a complete explanation refer to [7]. This technique is used both in a centralized and distributed forms.

978-1-4577-1348-4/13/$31.00 ©2013 IEEE

A. Compressive-Projections PCA

In the CPPCA framework, the encoder gathers M measurements from N sensors into the dataset $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_M]$, being $\mathbf{x}_m \in \mathbb{R}^N$ zero-mean vectors. $\mathbf{X}$ is then divided into J partitions $\mathbf{X}^{(j)}$ with $j = 1, \ldots, J$, each corresponding with an orthonormal $K \times N$ Compressed Projection (CP) matrix $\mathbf{P}^{(j)}$ to a K-dimensional subspace. Finally, it transmits the projected vectors $\tilde{\mathbf{Y}}^{(j)} = \mathbf{P}^{(j)}\mathbf{X}^{(j)}$ to the decoder, where the projection operators $\mathbf{P}^{(j)}$ are known *a priori*.

Using the Rayleigh-Ritz (RR) procedure [8] and $\tilde{\mathbf{Y}}^{(j)}$, the CPPCA decoder calculates $\tilde{\mathbf{\Sigma}}^{(j)}$, the covariance matrix of $\tilde{\mathbf{Y}}^{(j)}$. With a set of the Ritz vectors obtained from $\tilde{\mathbf{\Sigma}}^{(j)}$, the POCS (Projections Onto Convex Sets) [7] optimization is performed in order to resolve the first L eigenvectors and assemble them into a $N \times L$ matrix $\mathbf{\Psi}$. Matrix $\mathbf{\Psi}$ is an approximation of the L-component PCA transform, with $L \leqslant K$. Once $\mathbf{\Psi}$ is obtained, the PCA coefficients are recovered solving $\tilde{\mathbf{Y}}^{(j)} = (\mathbf{P}^{(j)})^\top \mathbf{\Psi} \check{\mathbf{X}}^{(j)}$.

The parameter that impacts the most in the performance of the LS-WSN is K/N. Low values of K/N provide high sparsity, therefore reducing energy consumption and transmissions by removing redundant features in the sensed data.

Therefore, the CPPCA method is a light-encoder/heavy-decoder system architecture that allows the reduction of the dimensionality of transmitted data. CPPCA is suitable for WSN, since it involves a low computational burden for the sensors and the number of transmissions towards the DFC decreases proportional to K (along with the resulting accesses to the wireless channel). In addition, decoding is performed by the DFC, a smarter and powerful node in the WSN.

B. Centralized CPPCA (CCP)

The first WSN setting is the centralized CPPCA, where the N nodes forming the WSN contribute to the same coding. Figure 1(a) shows an example for a CCP scenario. We assume error-free transmissions. Sensor nodes send their temperature measurements to the Cluster Head (CH) node, which has the additional role of CPPCA encoder. The computational cost of CCP coding has order of

$$O\left(KNM\right), \tag{1}$$

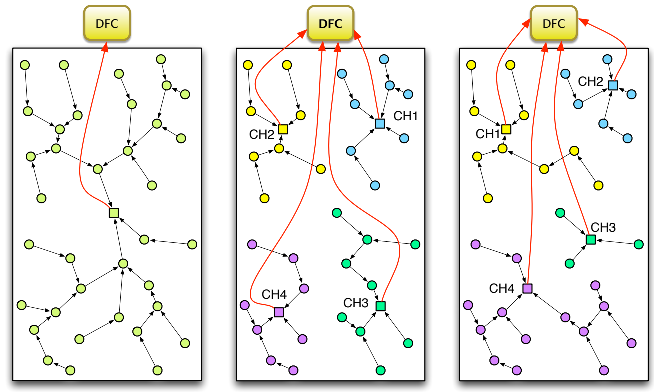
so a higher data compression configuration (i.e. low K/N) lessens the energy consumption for the encoders. Afterwards, the encoded data are sent to the DFC, which is responsible for the data reconstruction.

C. Distributed CPPCA (DCP)

The first stage for distributing the CPPCA coding in a WSN is to partition the network and implement CPPCA locally in each CH. The main advantage is that the computational cost of the coding task has order of

$$O\left(K_D M \sum_{i=1}^{N_c} N_i\right) = O\left(K_D M N\right). \tag{2}$$

where N_c is the number of clusters and N_i is the number of non-overlapping nodes in the i-th cluster, with $i = 1, ..., N_c$.



(a) (b) (c)

Fig. 1. Example of system architecture for (a) CCP, (b) DCP and (c) SODCP. Circles and squares indicate the location of the WSN sensors and CH, respectively, and arrows indicate the transmission direction.

In addition, K_D is the corresponding K for the cluster with $N_i^{max} = \max N_i \ \forall i$. Figure 1(b) shows a clustered version of Figure 1(a), with $N_c = 4$. The square nodes represent the CHs, meaning the CPPCA encoders.

Considering equal sparsity for both centralized and distributed settings, the total computational burden of the CCP coding is always higher than that of distributed one, as $K_D < K$, thus $O\left(K_D M N\right) < O\left(KMN\right)$.

III. Self-Organized Clustering

In order to provide with meaningful clusters of nodes for DCP to work we develop a data-driven self-organized distributed clustering technique. The joint use of both DCP and this clustering algorithm is named Self-Organized Distributed Compressed Projection (SODCP). We review the Fast Subspace Decomposition (FSD) metric, as it will be the basis for the clustering decisions. A homogeneous clustering method is also presented, to serve as a baseline.

A. Self-Organized Distributed FSD clustering

In the present work, we present a novel data-driven clustering algorithm that is based in the determination of the minimum dimension of the signal subspace from the measures of a cluster of nodes. As stated in section II, both CCP an DCP rely in the determination of the true signal subspace dimension in order to achieve successful compression. The determination of the signal subspace dimension has been the object of extensive research in the last decades. Specifically, Xu and Kailath [11] developed the FSD algorithm, which was oriented to iteratively estimate the $\hat{d}$ first RR eigenvectors and eigenvalues of the signal subspace up to the $\hat{d} = d$ iteration, where d is the true dimension. In each step of the algorithm, an FSD statistic is computed and it is shown that, for $\hat{d} \geqslant d+1$, the statistic

$$\varphi_{\hat{d}} = M(N-\hat{d}) \log \left[\frac{\sqrt{\frac{1}{N-\hat{d}}\left(||\tilde{\mathbf{\Sigma}}||^2 - \sum_{n=1}^{N}\theta_n^2\right)}}{\frac{1}{N-\hat{d}}\left(\mathrm{Tr}\tilde{\mathbf{\Sigma}} - \sum_{n=1}^{N}\theta_n\right)} \right] \tag{3}$$

is χ^2 distributed with $(1/2)(N-\hat{d})(N-\hat{d}+1)-1$ degrees of freedom, where θ_n are the RR eigenvalues. Finally, it is shown that, for a given M, the following inequality holds

$$\varphi_{\hat{d}} \leqslant \gamma_{\hat{d}}\ c(M) \tag{4}$$

In Eq. (4), $\gamma_{\hat{d}}$ can be computed from the aforementioned χ^2 distribution and $c(M)$ is a function s.t.:

$$\lim_{M\to\infty} \frac{c(M)}{M} = 0 \quad \text{and} \quad \lim_{M\to\infty} \frac{c(M)}{\log\log M} = \infty \tag{5}$$

In practice, functions such as $c(M) = \log(M)$ or $c(M) = \sqrt{\log(M)}$ can be used.

Moreover, matrix $\tilde{\mathbf{\Sigma}}$ experiences a *phase transition* for a number of samples M obtained from N sensors s.t. $M/N \geqslant \sigma^4/||\mathbf{v}||^4$, where σ^2 and $||\mathbf{v}||^2$ are, respectively, the noise power and the modulus of the first Principal Component of $\tilde{\mathbf{\Sigma}}$ (Eq. 2.19 in [12]). This phase transition is characterized by the "collapse" of the $\hat{d}$ largest eigenvalues from noise to signal subspace eigenvalues. A typical value of $\sigma^4/||\mathbf{v}||^4 \approx 4$ holds true for a wide range of signals. Thus, at least $M \gtrsim 4 \times N$ samples are needed to reliably extract the $\hat{d}$ signal eigenvectors using M measurements from N nodes.

We base the clustering procedure in the following hypothesis: *for an N_i-node cluster with $M_i = 4 \times N_i$ measurements per node, and for $\hat{d} \leqslant N_i$ s.t. $\varphi_{\hat{d}} \leqslant \gamma_{\hat{d}}\ c(M)$, the signal subspace from that specific cluster holds sufficient data to ensure DCP convergence.* Therefore, we propose a two phase algorithm which 1) seeds CH in the LS-WSN randomly and 2) lets the clusters grow until sufficient data is gathered to ensure DCP convergence. The first stage of the algorithm written in pseudocode can be seen in Algorithm 1.

The second stage, shown in Algorithm 2, is the most important and innovative one. Now, each CH is in charge of data analysis for all nodes belonging to its cluster and decisions about fusions. Each CH gathers an adequate amount of data (M_i) in order to properly compute the subspace dimension, using the FSD method. If $\hat{d} < N_i$, the CH is able to compress the data using CPPCA with $K = \hat{d}$. If not, the cluster must grow.

B. Homogeneous Clustering

In order to establish a baseline for comparison of the self-organized distributed FSD clustering with other clustering algorithms, we propose a homogeneous clustering method. The clustering of the network is performed a priori meeting two criteria: 1) minimization of Euclidean distance between nodes and 2) balance in the number of nodes per cluster. CH for each cluster is located such that distance to the members of the cluster is globally minimized. Thus, we ensure the capture of spatial correlations of up to the range from the farthest nodes in the cluster and balance computational complexity per cluster. For simplicity, we will refer to DCP using this clustering technique simply as DCP. In parallel, the joint use of self-organized distributed FSD clustering and DCP is named SODCP.

Algorithm 1 First stage: CLUSTER SEEDING

Decide if CH
if CH **then**
    REQUEST to first neighbors (only free nodes answer)
    UNION message to favorite neighbors
end if

Algorithm 2 Second stage: CLUSTER GROWING

$M_i = 4 \times N_i$
if CH **then**
    WAIT for M_i measurements per node
    $\hat{d} \leftarrow$ FSD dimension estimation of the cluster data
    if $\hat{d} \geqslant N_i$ **then**
        FUSION with nearest CH and decision of new CH
        if new CH **then**
            Gather all data from the other CH and update N_i
        end if
    end if
else
    Send data to CH
end if

IV. SIMULATION RESULTS

In this section, first we describe the dataset used in the computer simulations. Second, we will examine the reconstruction performance of a pure flat in-network processing technique (CCP), and compare it with CPPCA applied to hyperspectral data, in order to set a baseline for coding/decoding method in LS-WSN. In addition, the joint use of in-network processing and network partitioning is analyzed by comparing results for both DCP (*a-priori* partitioning) and SODCP (data-driven self-organized clustering), emphasizing the benefits in both scaling and reconstruction performance of the latter.

A. LUCE Dataset

The datasets used are obtained from an actual LS-WSN data, in particular temperature data gathered from the LUCE deployment from the SensorScope project [13]. The $N = 47$ sensors plotted in Figure 2 with red dots are selected from the original set (50% of the total) with data availability and height level in mind, forming a 2D surface.

In order to have sufficient data, we use $M = 10000$ values for each sensor, measured every minute for approximately one week. For now, the robustness of the method to corrupted data is out of scope of this work, so missing data and outliers are artificially replaced by the previous accurate value.

Due to the nature of the measured data (outdoor temperature), both trend and seasonal components are estimated and subtracted in the sensors. Afterwords, the dynamic range of the modified dataset is normalized to the support $[-1, 1]$.

B. Network partitioning

Two different network partitioning techniques are used in this work: 1) DCP and 2) SODCP. For the first technique, the $N = 47$ sensors clustering is done using an Euclidean

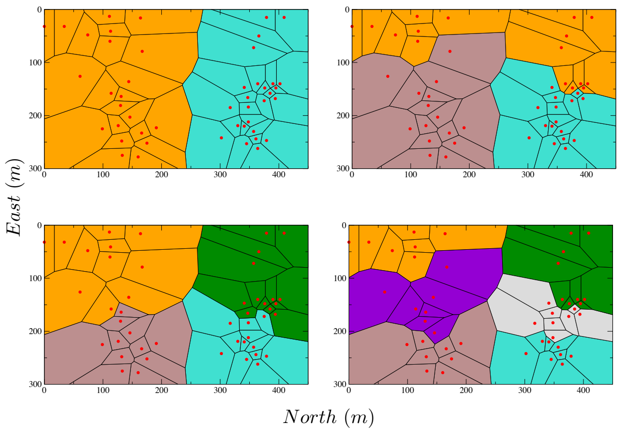


Fig. 2. Geographical situations for the LUCE sensors used (red dots) along with the homogeneous partitions for $N_c = 2, 3, 4$ and 6 clusters.

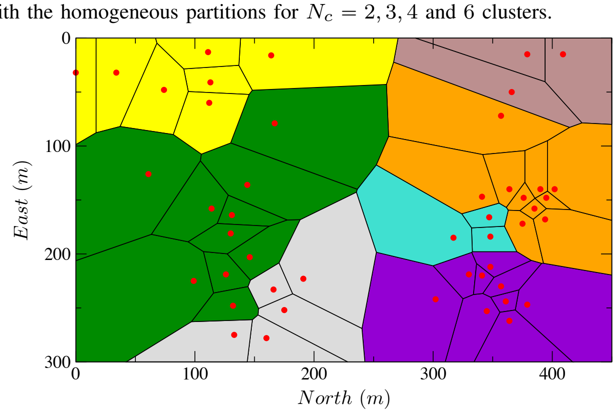


Fig. 3. Geographical situations for the LUCE sensor used (red dots) along with the self-organized distributed FSD clustering.

distance criterion and it is also attempted that all clusters have similar N_i (see Figure 2). On the other hand, the self-organized clustering scheme used is an outcome of the distributed algorithm presented in this work (see Figure 3). Now the cluster size depends on the subspace dimensionality of the sensor's measurements, determined by the spatial correlation present.

C. Flat LS-WSN in-network processing (CCP)

We examine the performance of CPPCA reconstruction with the dataset previously presented. Parameter $J = 20$ is used since it is a good tradeoff between accuracy and computational burden [7]. To simulate different data sparsities, the K value is changed in order to vary K/N. Finally, for each K/N, L is chosen to maximize the average SNR between the reconstruction and the original data, as the heuristic method proposed in [14] is focused only for hyperspectral imagery data.

In Figure 4, continuous line represents the average SNR obtained by applying the centralized CPPCA scheme to the LUCE dataset. In the same figure the dashed line is for PCA applied to the LUCE dataset, using the same amount of eigenvectors as for the CPPCA case, given by the parameter L. The dissimilarity between both is little enough to encourage the implementation of CPPCA in WSN. In addition, we perform a comparison with the "Cuprite" dataset used in original studies about CPPCA, e.g. [7], [14]. For a fair comparison, the

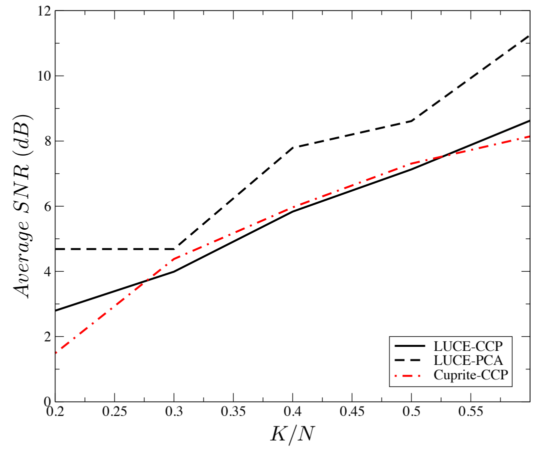


Fig. 4. Reconstruction performance of CCP (average SNR vs K/N) with LUCE dataset (straight line) compared to the hyperspectral imagery Cuprite dataset (dash-dotted line). Performance comparison with PCA (dashed line), using the same number of principal components as for CPPCA, is shown. Note that $K/N = 1$ for PCA; representation in the horizontal axis is only shown for comparison purposes.

same parameters are used (except L, for which their optimal values are used) and, regarding the dataset, the mean vector is removed and the dynamic range is set to $[-1, 1]$.

The preprocessing procedure makes both datasets as close to a gaussian distribution as possible. Due to PCA discrimination capacity against the variance of data, the analysis performed offers almost identical results for both datasets. Moreover, as expected, as the sparsity of the data decreases (K/N increases), the reconstruction performance increases.

D. Joint in-network process and clustering (DCP vs. SODCP)

We now proceed to analyze the performance of the distributed CPPCA for both clustering techniques considered. Regarding the parameters, M and J are the same as in the centralized configuration. K_D is selected for N_i^{max} and L is chosen to maximize the average SNR considering all clusters. Both K_D and L are equal for all clusters, with the exception of the smallest clusters in SODCP, where it can happen that $N_i < K_D$ or $N_i < L$. In this particular cases $K_D = L = N_i$, so no compression is realized for that clusters.

The reconstruction performances for DCP and SODCP are presented in Figure 5. The CCP configuration is also included in order to facilitate the comparison. This outcomes suggest that exists a lower bound for data compression. Therefore, for reasonable values of K/N ($\geqslant 0.3$), the results obtained encourage the network partitioning as a low reconstruction performance degradation is obtained.

As observed, the $N_c = 2$ case for DCP outperforms the centralized case. This interesting fact occurs due to the node location and the network partition considered. The N nodes selected from the LUCE deployment are located in such way that a gap is formed in the central strip of the WSN borders. Therefore, the $N_c = 2$ case captures in a better way the spatial correlations, which are expected to be different between the two clusters. Similar arguments may be used for the $N_c = 3$ vs. $N_c = 4$ discussion.

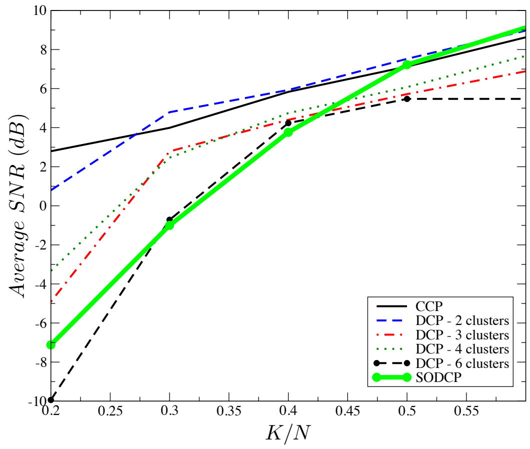


Fig. 5. Reconstruction performance (average SNR vs. K/N) for different partitions of the $N = 47$ LUCE sensor network.

For $K/N \sim 0.2$ the reconstruction performance is dominated by contributions from small clusters and therefore, by small-scale spatio-temporal features. However, as K/N increases, larger clusters become relevant and large-scale spatio-temporal features contribute to the average SNR. Therefore, SODCP reconstruction behavior for small K/N should be (and is, see Figure 5) similar to that of DCP with $N_c = 6$. Finally, for smaller compression ratios ($K/N \geqslant 0.5$) SODCP should exhibit similar performance as DCP for $N_c = 4$ as the larger nodes of the former are similar to those of the latter ($N_i = 10$). However, as SODCP exploits the spatio-temporal structure of the data for clustering, its efficiency in reconstruction performance in the cluster exceeds those of DCP with even larger clusters.

E. Transmission reduction

The impact of the CPPCA encoding in the amount of transmissions between the CHs and the DFC is analyzed here. This quantity is lessened due to the gap between the dimensions of the vector spaces of the encoded data (K) and that of the measurements (N). Moreover, by reducing the amount of transmissions, the accesses to the wireless channel are decreased, with the consequent energy efficiency.

In order to analyze this fact, the ratio between the amount of transmissions with DCP encoding and with no in-network processing, is computed as

$$\text{ratio} = \frac{(K \times M) + 5N_i^{max} + 1}{N_i^{max} \times M} \times 100 \qquad (6)$$

The ratio is the upper bound for all CHs, as is obtained using the corresponding largest N_i.

As expected, the reduction in the percentage of transmitted values towards the DFC is proportional to K/N. The same expression applies to SODCP with the caveat that unbalance in transmissions between larger and smaller cluster is present. Therefore, the benefit of transmission reduction in the reconstruction are present both in DCP and SODCP. However, the latter has a greater potential over the former for better performance in reconstruction fidelity as K/N increases.

V. Conclusions

In this work we have introduced SODCP that jointly performs 1) a self-organized network partitioning for LS-WSNs that forms the clusters based on the spatio-temporal correlations of the sensed data and, 2) compressed projections of the gathered data in each cluster in order to reduce energy consumption and transmissions. We have proven it to be a suitable for WSN as it is a scalable light-encoder/heavy-decoder system architecture. We have shown that this data-driven method represents a step forward in the self organizing clustering in LS-WSN problem by using two baselines: flat network organization (CCP) and a priori homogeneous clustering (DCP). In addition, we have confirmed that the amount of transmissions between the encoder nodes and the DFC can be significantly reduced by use of SODCP, with negligible loss of performance over CCP for reasonable compression rates (average SNR loss $\leqslant 1$ dB for $K/N \geqslant 0.4$). Therefore, it enables considerable transmission reduction at a controlled performance loss.

Acknowledgment

The authors are thankful to Dra. Inmaculada Mora-Jiménez for valuable suggestions during the course of this work.

This work was supported in part by the Spanish Government under Projects TEC2009-12098 and TEC2010-19263 and PhD Research Grant AP2012-2981.

References

[1] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. on Information Theory*, vol. 46, no. 2, 2000.
[2] S. Zhao and D. Raychaudhuri, "Scalability and Performance Evaluation of Hierarchical Hybrid Wireless Networks," *IEEE/ACM Trans. on Networking*, vol. 17, no. 5, 2009.
[3] A. Giridhar and P. R. Kumar, "Computing and communicating functions over sensor networks," *IEEE Journal on Sel. Areas in Comms.*, vol. 23, no. 4, pp. 755–764, 2005.
[4] S. Ramaswamy, K. Viswanatha, A. Saxena, and K. Rose, "Towards large scale distributed coding," in *IEEE Intern. Conf. on Acoustics Speech and Signal Processing*, 2010, pp. 1326–1329.
[5] C. Caione, D. Brunelli, and L. Benini, "Distributed Compressive Sampling for Lifetime Optimization in Dense Wireless Sensor Networks," *IEEE Trans. on Industrial Informatics*, vol. 8, no. 1, pp. 30–40, 2012.
[6] D. Guo, X. Qu, M. Xiao, and Y. Yao, "Comparative Analysis on Transform and Reconstruction of Compressed Sensing in Sensor Networks," in *WRI Internat. Conf. on Comms. and Mobile Computing, 2009. CMC '09.*, 2009, pp. 441–445.
[7] J. E. Fowler, "Compressive-Projection Principal Component Analysis," *IEEE Trans. on Image Proc.*, vol. 18, no. 10, pp. 2230–2242, 2009.
[8] B. Parlett, *The symmetric eigenvalue problem*. SIAM, 1980, vol. 7.
[9] T. D. Le, N. D. Pham, and H. Choo, "Towards a Distributed Clustering Scheme Based on Spatial Correlation in WSNs," in *Intern. Wireless Comms. and Mobile Comp. Conf.*, 2008.
[10] P. Wang, R. Dai, and I. F. Akyildiz, "Collaborative Data Compression Using Clustered Source Coding for Wireless Multimedia Sensor Networks," in *Proc. IEEE INFOCOM, 2010*, 2010, pp. 1–9.
[11] G. Xu and T. Kailath, "Fast subspace decomposition," *IEEE Trans. on Signal Processing*, vol. 42, no. 3, pp. 539–551, Mar. 1994.
[12] B. Nadler, "Finite sample approximation results for principal component analysis: A matrix perturbation approach," *The Annals of Statistics*, vol. 36, no. 6, pp. 2791–2817, Dec. 2008.
[13] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope," *ACM Trans. on Sensor Networks*, vol. 6, no. 2, pp. 1–32, 2010.
[14] W. Li and J. E. Fowler, "Decoder-Side Dimensionality Determination for Compressive-Projection Principal Component Analysis of Hyperspectral Data," in *18th IEEE Intern. Conf. on Image Proc. (ICIP)*, 2011, pp. 321–324.