Energy Efficiency and Quality of Data Reconstruction through Data-Coupled Clustering for Self-Organized Large-Scale WSNs

Mihaela I. Chidean, Eduardo Morgado, Margarita Sanromán-Junquera, Julio Ramiro-Bargueño, Javier Ramos, and Antonio J. Caamaño

***Abstract*—Energy efficiency has been a leading issue in Wireless Sensor Networks (WSNs) and has produced a vast amount of research. Although the classic tradeoff has been between quality of gathered data versus lifetime of the network, most works gave preference to an increased network lifetime at the expense of the data quality. A common approach for energy efficiency is partitioning the network into clusters with correlated data, where representative nodes simply transmit or average measurements inside the cluster. In this work, we explore the joint use of in-network processing techniques and clustering algorithms. This approach seeks both high data quality with a controlled number of transmissions using an aggregation function and an energy efficient network partition, respectively. The aim of this combination is to increase energy efficiency without sacrificing the data quality. We compare the performance of the Second-Order Data-Coupled Clustering (SODCC) and Compressive-Projections Principal Component Analysis (CPPCA) algorithm combination, in terms of both energy consumption and quality of the data reconstruction, to other combinations of state of the art clustering algorithms and in-network processing techniques. Among all the considered cases, the SODCC+CPPCA combination revealed a perfect balance between data quality, energy expenditure and ease of network management. The main conclusion of this paper is that the design of WSN algorithms must be processing-oriented rather than transmission-oriented, i.e., investing energy on both clustering and in-network processing algorithms ensures both energy efficiency and data quality.**

***Keywords*—Wireless sensor networks, Data-coupled clustering, Energy efficiency, Data quality**

I. Introduction

In recent years, the promise of large scale Wireless Sensor Network (WSN) applications have been seen as already achieved or just about to be realized. However, large problems remain unsolved as WSN performance metrics are highly application dependent. Although there exist energy harvesting technologies, their efficiency is highly dependent on the node usage. Therefore, sensor nodes may have no alternative for the power source and energy efficiency is required for extended lifetime. Thus, we observe that the most largely addressed issue in the scientific literature is the energy operation [1], especially in the design of clustering algorithms for WSN.

Network partitioning using a clustering algorithm improves the total capacity of the network and also provides scalability [2], [3]. Most clustering algorithms base their decisions on seeking low energy consumption during the formation of the clusters, leaving aside what happens afterwards, i.e. data measurement and their transmission to the Data Fusion Center (DFC). In other words, using little energy to cluster the network does not guarantee a high energy efficiency during the complete WSN lifetime.

Several in-network processing algorithms have been developed for WSN, with the aim of efficiently aggregating measured data such that the data recovered by the DFC is as similar as possible to the measured data. In this case, the tradeoff between the data quality and the amount of transmitted bytes to the DFC (i.e. the compression ratio) has to be considered. Usually less compression leads to higher data quality, but also leads to higher amount of transmissions and more energy consumption. Data transmission is the most expensive task in a WSN from an energy point of view [4].

Recently, the authors proposed the Second-Order Data-Coupled Clustering (SODCC) algorithm, specifically designed to attend the needs of the in-network processing algorithm [5], [6]. The decision criterion of SODCC is based on measured data statistics (i.e. second order moments), to obtain the best cluster configuration for an in-network processing algorithm that uses the same data statistics. In this way, the data processing is more efficient as the autocorrelation matrix of the data measured by the cluster is well posed.

The purpose of this work is to further analyze SODCC and to determine whether the design of a clustering algorithm attending to the needs of the in-network processing algorithm, in general, and the design of the SODCC algorithm, in particular, are beneficial both from the energy and the data quality points of view. We simulate the operation of a WSN with several clustering and in-network processing algorithms and show that SODCC approach is worth of further research as we obtain promising results both in energy consumption and data quality.

In the following, Section II relates the state of the art of clustering and in-network processing algorithms for WSN. Section III describes the network model and the different algorithms used in this work. The performed computer simulations and the obtained outcomes are explained in Section IV, and the obtained results are analyzed in Section V. Finally, Section VI

The authors are with the Department of Signal Theory and Communications, Rey Juan Carlos University, Camino del Molino, s/n, 28943, Fuenlabrada, Spain (email: {mihaela.chidean, eduardo.morgado, margarita.sanroman, julio.ramiro, javier.ramos, antonio.caamano}@urjc.es).

This work has been partially supported by the Research Projects S2013/MAE-2835 from the Autonomous Community of Madrid. Mihaela I. Chidean is supported by the FPU Research Grant AP2012-2981 from the Spanish Ministry of Education, Culture and Sports.

summarizes the conclusions of this work.

II. State of the art

During the last decade, there has been significant research in clustering techniques for WSNs and their benefits in terms of energy efficiency. Multiple available algorithms include the objective of saving energy during the formation of the clusters [1], [7]. Even though different metrics such as algorithmic computational complexity, scalability, load balancing, ... are usually examined as possible tradeoffs to energy efficiency, the quality of the data obtained at the DFC is not a common metric for comparison.

The quality of the retrieved data is addressed by some clustering algorithms, along with energy saving properties. Using the correlation of the measurements, other authors [8] propose a combination of distributed in-network processing algorithms and centralized heuristics that effectively selects correlated small clusters. It is shown that it is possible to obtain a reconstruction error lower than 10% during the WSN lifetime. But the energy efficiency of their proposal is only evaluated in terms of reduction in number of transmitted messages, and the energy cost of in-network processing is not taken into account. In [9], a similar approach is adopted to perform distributed WSN clustering by exploiting the spatial correlation of the measured data. Simulations with synthetic scenarios show that the spatial patterns of measured data are recognized with high accuracy ($> 90\%$) in comparison with other clustering techniques such as Low Energy Adaptive Clustering Hierarchy (LEACH) [10], [11]. However, the authors fail to evaluate the energy efficiency despite the apparent high algorithmic complexity of the approach. An alternative approach that exploits time correlation is proposed in [12], where cluster scheduling is optimized to adjust surveillance time series and, therefore, save energy in the transmission process. A detailed evaluation of the quality of the retrieved data is performed, and a realistic architecture for the nodes is proposed, but no systematic analysis of energy efficiency is performed. Geostatistics data reconstruction techniques such as Kriging are considered in [13] by means of the Spatial Kluster Analysis by Tree Edge Removal (SKATER) algorithm [14]. SKATER is analyzed in terms of reconstruction error, but again energy efficiency analysis is omitted.

In-network processing algorithms for WSN also try to take advantage of the spatial correlation of the data in order to increase the networks lifetime. Representative examples are the Approximate Data Collection or Approximate Data Gathering (ADG) approaches [15], [16], where representative nodes are selected for each cluster, and higher reconstruction error is traded for higher energy efficiency and WSN lifetime. The reconstruction error is generally dealt with prediction techniques, e.g. in Adaptive Sampling Approach (ASAP) [17] where the data is extracted from the network using model-based prediction and reducing the amount of transmissions. Synthetic examples are used to assess both energy consumption and data quality, but the cost of this centralized selection of representative nodes is not analyzed. A similar prediction-oriented approach is adopted in [18] but with a less thorough approach in terms of reconstruction error.

In the present work, both the energy efficiency and the quality of the reconstructed data are assessed for different clustering and in-network processing algorithms combinations. With this analysis we show that an energy-saving oriented design for an algorithm does not lead to overall energy efficiency as multiple factors not considered *a priori* act as energy sinks. For example, the main drawback for LEACH is the requirement of single-hop communications, while for ADG the issues arise if the representative nodes are not properly selected.

III. System Model

The network model used in this work is first described. In short, the WSN is modeled by a graph, the nodes are clustered using a specific clustering algorithm and a local data processing is performed in each cluster. Following, in separate subsections, we describe all the clustering and in-network processing algorithms used in this work.

A. Network model

Consider a WSN with N nodes and diverse wireless connection between them. During the WSN operation, each of the N nodes takes a measurement every T_s time instants, obtaining a total of M data measurements per node.

Regarding the network organization, the WSN is divided into clusters, to avoid the capacity limitations of a flat WSN [2]. The trend in this field is that of self-organized algorithms, that usually try to cluster the network "on-the-fly" while seeking to reduce at least one of the following limited resources: 1) energy; 2) bandwith; or 3) computational capabilities.

In addition, each cluster of the WSN performs data processing in order to reduce the number of data transmissions. This local processing can range from the simplest possible (i.e. calculation of the average value of a group of nodes [19]) to more sophisticated algorithms (i.e. discovery of local data correlation [15]). The tradeoff between the use of the computational resources and the improvement of the transmission efficiency has to be included in the choice of the processing algorithm, as an unfortunate choice can lead to network flooding and high energy consumption.

B. Clustering algorithms

We use three different clustering algorithms, selected for the following reasons: 1) LEACH [10], [11], designed to be energy-efficient and to increase the network lifetime; 2) Persistent [20], designed to balance the cluster sizes while the amount of control messages transmitted is low; and 3) SODCC [5], [6], designed to cluster the network using the second-order statistics of the measured data, obtaining a data-coupled clustering configuration.

1) LEACH: is one of the most popular clustering techniques for WSNs [1], [21]. The most important feature of LEACH is that the energy consumption is evenly distributed among all the nodes in the network. The operation is the following: (i) nodes independently decide to become Cluster Head (CH)

with an *a priori* known probability p and broadcast their decision; (ii) all non-CH nodes join the cluster whose CH is reached with the least communication energy. The CHs are rotated and the cluster configuration is changed periodically (each "round"), in order to balance the load and the energy consumption. A node that is CH in a given round cannot be again CH in the next $1/p$ rounds. One of the most significant constraints of LEACH is the fact that all communications have to be single-hop, meaning that all nodes have to be able to communicate to all the other nodes in the WSN. Moreover, as all nodes are susceptible to be CH, all nodes have to be able to communicate in a single-hop communication with the DFC. These facts make LEACH unsuitable for large networks, as the consumed energy to communicate the two farthest nodes may be unaffordable. Based on its design, the expected benefits for LEACH are the very low energy consumption and the simple implementation, as it only involves some random decisions and several control message interchanges.

2) Persistent: is categorized as message-efficient, as it uses few control messages in the cluster formation. Its main goal is to form clusters with equal cluster size B. The operation of Persistent is the following: (i) the initiator nodes know the *a priori* set parameter or budget B; (ii) each initiator evenly distributes the remaining budget $B-1$ among its neighbours; (iii) the nodes that receive those messages re-distribute their assigned budget among their neighbours except for the parent node ; (iv) the messages propagate until the budget is exhausted or there are no more available neighbours; (v) each node informs its parent about the size of the formed subtree; (vi) if the budget was not exhausted, it is re-distributed among unexplored neighbours or those have already met the previously allocated budget, in order to find new nodes to include in this cluster; (vii) the algorithm terminates when the initiator consumes the budget B or when no further growth is possible. Due to the latter, Persistent does not guarantee that all cluster sizes are equal to B, as if it is not able to find sufficient nodes, the cluster is smaller. In addition, Persistent needs $O(B^2)$ control messages to form a cluster of size B [20], which can be a quite high value depending on the multipliers ignored in the notation $O(\cdot)$. A highly restrictive assumption of Persistent is that the budget of B is a design parameter. Depending on the chosen value, the clusters may be too large to overcome the capacity issues or may be too small and difficult the management task of the WSN.According to the design of the Persistent algorithm, the expected benefits are homogeneous cluster configuration and low amount of control messages transmitted through the network.

3) SODCC: is a self-organized clustering algorithm developed for WSN that uses the statistics of the measured data as significant part of the decision criteria for the cluster formation. Specifically, this algorithm uses second-order statistics to compute the minimum number of linearly independent components in the cluster, i.e., the number of principal components that explain most of the variance in the data or, similarly, the dimension of the signal subspace. With this approach, the SODCC algorithm ensures that most of the variance in the data is captured within that cluster. It means that, if the cluster is small, only the data from a small number of stations are needed to extract the eigenvalues of the largest principal components that explain their variance. Thus, the cross-correlation of the data among those stations is high. On the other hand, if the cluster is large, the data from more stations are needed to explain the same percentage of the variance. Therefore, SODCC ensures that closely correlated data series are clustered by using the dimension of the signal subspace.

The operation of SODCC is the following: (i) nodes randomly decide according to an *a priori* set probability P to become CH and form the first clusters in the network, preferably small-sized clusters (the seed of the final clusters); (ii) once all nodes belong to a cluster, either as sensor nodes or CHs, the sensor nodes measure and send their data to the CH; (iii) once the CH has gathered enough data, it estimates the signal subspace dimension $\hat{d}$ of the data using the Fast Subspace Decomposition (FSD) procedure [22]; (iv) if $\hat{d}$ is smaller than the cluster size N_c, that cluster is stable; (v) if $\hat{d}$ is greater than the cluster size, that cluster is not stable, meaning that signal and noise subspaces are not separable. In this last case, the cluster initiates a fusion process with a neighbour cluster in order to adapt the clustering configuration to these large-scale correlations of the data that are found in this area of the WSN. The main drawbacks of the SODCC algorithm are the additional computational burden needed to apply the FSD algorithm and the transmission of the data between sensor nodes and CH during the clustering. The expected benefits of SODCC are a data-coupled clustering configuration, meaning that the statistics of the measured data are as similar as possible within any given cluster. Moreover, as SODCC performs the clustering in terms of second-order statistics, a processing algorithm that also uses second-order statistics will obtain better quality results.

C. Processing algorithms

It is well accepted in the scientific community that the measurements made by nearby sensors in a real environment usually have spatial correlation. Therefore, we use three processing algorithms that take advantage of this assumption (some are more effective than others) to reduce the amount of transmissions through the WSN.

1) Approximate Data Gathering - Cluster Head (ADG-CH): is the simplest possible in-network processing algorithm for a WSN and transmits the lowest amount of bytes through the WSN. The key issue of this algorithm is that, as nearby nodes should have high spatial correlation [23], [24], a unique sample should be sufficient to represent all the measurements made by a group of nearby nodes. As the WSN is partitioned by some clustering algorithm it is reasonable to assume that the measurement from any node belonging to that cluster could represent the measurements from the entire cluster in the DFC. It is also reasonable to assume that the representative node should be the CH.

2) Approximate Data Gathering - Mean (ADG-M): it is an improvement of ADG-CH, as the bytes transmitted to the DFC are a better representation of the measurements made by a cluster. This algorithm transmits the average value of the data measured by the cluster to the DFC. The disadvantages are

that all the nodes have to transmit their measurements to the CH, and that the CH spends additional energy computing the average of the measurements. On the other hand, the main benefits of ADG-M are both that the DFC receives a better approximation of the values measured by the nodes and that the values received by the DFC are not susceptible to errors of the electronics of the CH. This type of data processing was also used in [19], [25], [26], where either CHs or specialized aggregator nodes performed an average aggregation function.

3) Compressive-Projections Principal Component Analysis (CPPCA): is an algorithm based on Principal Component Analysis (PCA) that was first proposed for satellite hyperspectral imagery compression [27]. This is the processing algorithm with the highest computational burden that we use in this work. The operation of the algorithm is the following: (i) the measured data by a group of N_i nodes (i.e. the cluster) are gathered by a representative node (i.e. the CH); (ii) the CH assembles the data matrix $\mathbf{X}$ of size $N_i \times M$ and it is encoded by performing an orthonormal projection to a lower-dimension random space $\mathcal{P}$ of dimension K, with the orthonormal compressed projection matrix $\mathbf{P}$; (iii) the representative nodes transmits the encoded data to the DFC; (iv) the DFC estimates the covariance matrix of the received data $\tilde{\mathbf{\Sigma}}$ using the Rayleigh-Ritz (RR) procedure [28]; (v) the DFC uses the estimated matrix and performs a Projections Onto Convex Sets [27] optimization to resolve the L_i principal eigenvectors; and (vi) the DFC uses the eigenvectors to recover the PCA coefficients and obtain the highest possible information about the measured data. The relation between $\tilde{\mathbf{\Sigma}}$ and $\mathbf{\Sigma}$ of the measured data is established by means of matrix $\mathbf{P}$:

$$\tilde{\mathbf{\Sigma}} = \mathbf{P}^\top \mathbf{X}\mathbf{X}^\top \mathbf{P}/M = \mathbf{P}^\top \mathbf{\Sigma}\mathbf{P} \tag{1}$$

CPPCA can be adapted to the capacity needs of the WSN, as the amount of bytes transmitted are controlled by the ratio K/N. A lower K/N ratio leads to a higher data compression and a lower amount of bytes transmitted to the DFC. The disadvantage of this algorithm is the high computational load and the memory requirements of the sensor nodes to be able to store the gathered data, and to perform the projection onto a different subspace (i.e. matrix multiplication). In addition, the amount of bytes transmitted between the nodes and the CH, and between the CH and the DFC can be significantly higher than the previous two algorithms. On the other hand, the CP-PCA algorithm allows for the adjustment of the compression ratio (K/N) to control the amount of transmitted bytes and the quality of the data decoded by the DFC. Moreover, as CPPCA is PCA-based, the second order statistics of the data decoded by the DFC are ensured to be as similar as possible with the one of the measured data. This type of data analysis was also used in [27] to recover hyperspectral images or in [5] and [6] as an in-network processing algorithm.

The computational cost of the CPPCA procedure scales as $O(L_i^2 \times M)$, mainly due to the inherent cost of the RR procedure. Moreover, this algorithm lacks of adaptability in a changing environment. In the present work, although we use the original version of the algorithm proposed in [6], we can formulate a low-complexity adaptive version of the CPPCA algorithm where both the extraction of the eigenpairs and the calculation of the dimension d of the signal subspace can be done by an alternative procedure: the extended PASTd [29]. This modification could be also used by SODCC to determine the signal subspace dimension d instead of the FSD [22] statistic.

The covariance matrix is substituted by the exponentially weighted sample covariance matrix, defined as

$$\mathbf{\Sigma}(t) = \beta \mathbf{\Sigma}(t-1) + \mathbf{X}\mathbf{X}^\top(t) \tag{2}$$

where $0 < \beta < 1$ is a forgetting factor, and the index t is used to indicate the multiples of M data samples that are used to update the weighted sample covariance matrix. Therefore, in batches of M, the data are processed and new eigenpairs and dimension d are extracted, either by means of the FSD procedure or the PASTd procedure.

The low-complexity version of the CPPCA can be implemented substituting the RR estimation by the PASTd estimation [29], using either the Akaike Information Criterion (AIC) or the Minimum Descripton Length (MDL) criterion [30] as substitutes for the FSD criterion for rank subspace estimation.

The resulting computational complexity of the PASTd-CPPCA scales as $O(L_i \times M)$, therefore achieving high gains in terms of operations and energy expenditure at identical error thresholds for the eigenpair recovery. However, even in quasi-static environments (in terms of eigenpair variation) such as the ones explored in the present work, the convergence times almost double those of the RR adaptive version of CPPCA [31], [32]. Therefore, in the following we will use the CPPCA version presented in [27] and previously used for WSN in [6].

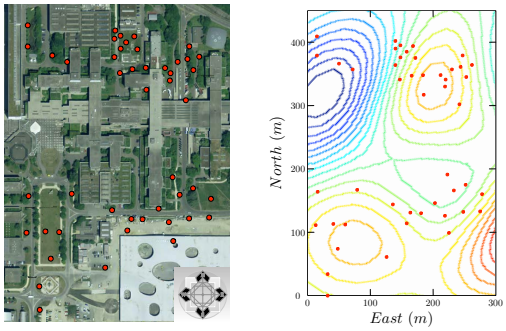
IV. Experiments

In this work multiple independent computer simulations using actual WSN temperature data and different network settings where performed. Each of these network settings used different combinations of clustering and in-network processing algorithms. These experiments allow us to assess the network management facility, calculate the energy consumption of multiple network settings and analyze the tradeoffs among transmission energy saving, processing energy consumption, and improvement of the reconstruction error. In this section, we describe the dataset used in our simulation, the network setting, and the outputs of our experiments.

A. Dataset

The dataset contains air temperature data gathered from the LUCE deployment of the Sensorscope project [33]. Figure 1(a) shows the location of $N = 47$ sensor nodes, selected from uniform height and that form a 2D network. Figure 1(b) represent the schematic view of this WSN. This figure includes an example of a generic temperature field represented as isothermal lines. Computer simulations involve data transmission to a fixed DFC; without loss of generality and for simplicity's sake, the DFC is located at $[150, 500]$.

The dataset contains $M = 10^4$ samples per sensor, measured during the last week of April 2007 ($T_s = 60s$). The preprocessing includes : (1) imputation of the missing data



(a) (b)

Figure 1. (a) Location of the $N = 47$ sensor nodes used for the computer simulations. Reproduced by permission of swisstopo (BA13063). (b) Schematic representation of the WSN with a temperature field example. The red dots represent the sensor nodes.

(4.08% of all samples) and the outliers (0.14% of all samples) by the previous available value, (2) subtraction of both trend and seasonal components and, (3) normalization of dynamic range to the support $[-1, 1]$. The total computational burden of this preprocessing is $O(M)$ due to the operations needed for the second part, which are light enough to be performed by the sensor nodes of a WSN.

This dataset has been previously used to test the performance of CPPCA in a WSN with no clustering algorithm [5] and SODCC+CPPCA algorithms in a WSN [6]. However, these previous works are focused based on error reconstruction results.

B. Network settings

Several network settings are used in the computer simulations, each containing a clustering and an in-network processing algorithm. Figure 2 shows the schematic representation of the operation of each network setting as graphical examples, in order to clearly distinguish the fundamental differences between each case. The amount of independent simulations performed is 10^3 for LEACH and 10^5 for both Persistent and SODCC algorithms. Several additional considerations have to be taken into account regarding the particular configuration used, i.e.:

1) The LEACH + CPPCA configuration is not used, as CPPCA [27] is formulated for static node set and LEACH does not allow for this feature.
2) For a fair comparison, all network settings that use ADG-CH or ADG-M report the data through the DFC in successive rounds. This fact is motivated by the LEACH algorithm which performs a new clustering configuration in each round. Therefore the CH (the node that transmits data to the DFC) is different in each round. As a result, although Persistent and SODCC maintain the clustering configuration through all the simulation, the representative node is rotated among the nodes belonging to the cluster.
3) The $p = 0.35$ parameter for LEACH is selected to be equal to the $P = 0.35$ parameter for SODCC, used in all the simulations performed with this second algorithm.
4) The $p = 0.5$ parameter for LEACH is selected to analyze the performance of these network settings when the amount of transmitted data is half the amount of measured data. These cases are similar to the network settings that use CPPCA with $K/N = 0.5$.
5) The network settings that use CPPCA with $K/N = 0.3$ are selected to analyze the performance of the WSN with high data compression, i.e. the amount of transmitted data is very low with respect to the amount of measured data. These cases are similar to the network settings that use LEACH with $p = 0.35$.
6) The network settings that use CPPCA with $K/N = 0.8$ are selected to analyze the performance of the WSN with low data compression, i.e. the amount of transmitted data is similar to the amount of measured data.

C. Outcomes

The outcomes that we evaluate for each configuration in Section V are: (1) average number of clusters, to assess the clustering algorithm management issues; (2) Signal to Noise Ratio (SNR), to analyze the data reconstruction quality; (3) energy consumption, to evaluate the network lifetime. The calculation of each of these outcomes is explained in the following.

1) Number of clusters: To evaluate the ease of management of the clustering configuration once the WSN is clustered, we compute the average number of clusters obtained in each simulation. Obviously, a WSN with a constant number of nodes N divided into more clusters leads to clusters with smaller sizes. The management of a WSN with many small clusters is more complicated as it requires a higher amount of signaling between CHs and between CH and the DFC.

2) SNR: To assess the data reconstruction quality, we use the SNR to quantify the relation between the power of the signal of interest, i.e. the original sample data x, and the reconstruction error or noise, i.e. the difference between x and the data obtained at the DFC $\tilde{x}$. The SNR is calculated for each independent simulation with a particular combination of parameters as:

$$\text{SNR}_{sim} = \frac{\mathbb{E}\left[(x - \mathbb{E}[x])^2\right]}{\mathbb{E}\left[(x - \tilde{x})^2\right]} \tag{3}$$

and for each network configuration as:

$$\text{SNR} = \mathbb{E}\left[\text{SNR}_{sim}\right] \tag{4}$$

where $\mathbb{E}[\cdot]$ represents the expected value of a random variable.

3) Energy consumption: To estimate the consumed energy we differentiate between the amount of transmitted bits, received bits and, performed operations. The value of these estimations is an approximation in simulations, and it depends highly on hardware specifics, but we tried to remain accurate allowing for generalization. The consumed energy in each case is computed as:

- Transmission: $Energy\ (J) = c_t \times$ total transmitted bits $\times\ distance^{\eta}$
- Reception: $Energy\ (J) = c_r \times$ total received bits
- Process: $Energy\ (J) = c_p \times$ total performed operations

being c_t, c_r, c_p three constants (dependent on electronic devices) for the amount of energy needed to transmit and receive one bit, and to process one operation, respectively.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2016.2551466, IEEE Sensors Journal

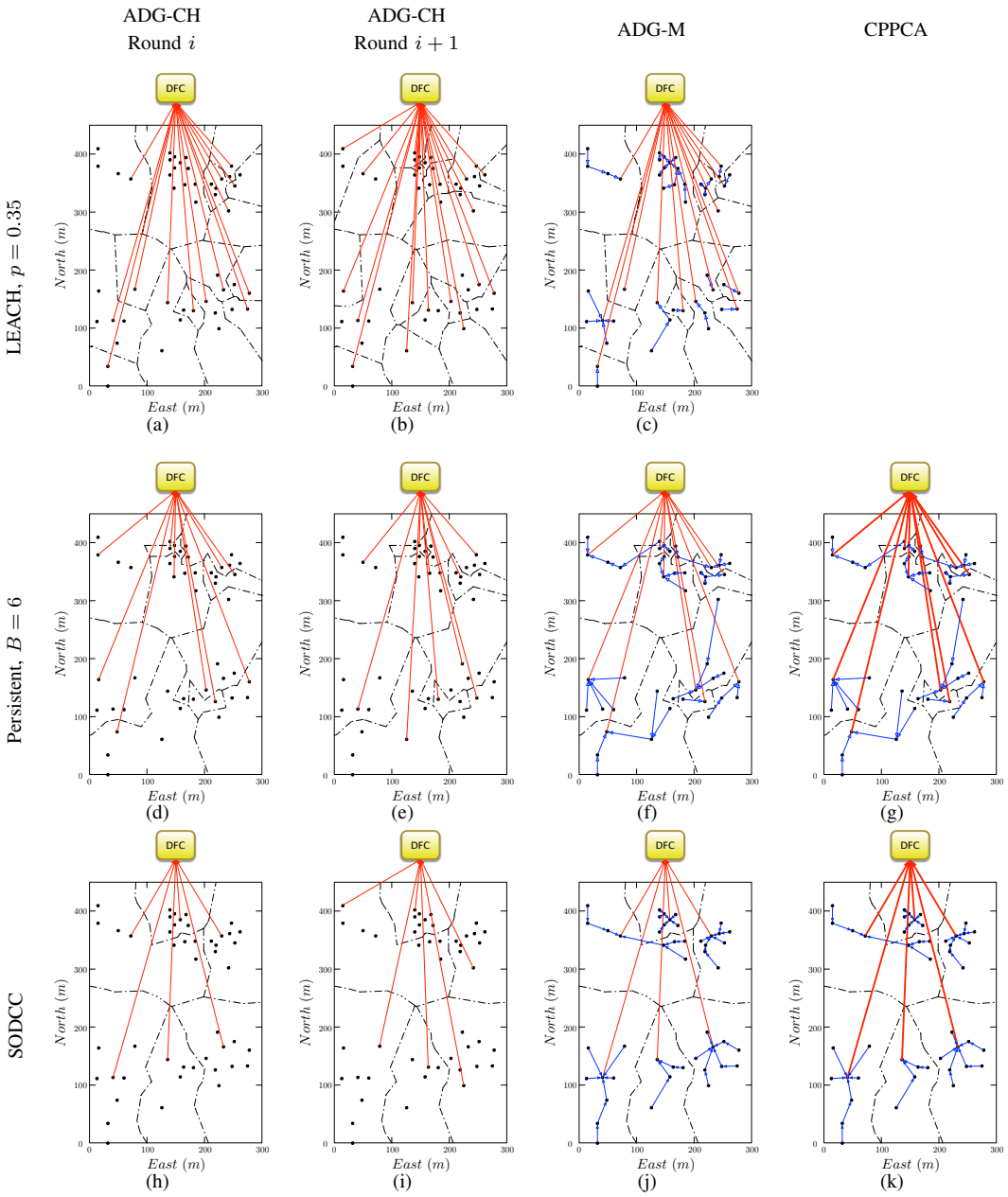


Figure 2. Schematic representation for each clustering and in-network processing algorithms combinations used in the computer simulations. Rows are for different clustering algorithms and columns are for in-network processing algorithms. The first two columns are both for ADG-CH, representing two consecutive rounds. Black dots represent the sensor nodes, discontinuous black line represent the cluster frontiers, blue arrows represent the intra-cluster packet transmission and red arrows represent the CH to DFC transmission. Thicker red arrows represent higher amount of packet transmissions.

The path loss exponent is set to $\eta = 3$, a good tradeoff between the free space propagation ($\eta = 2$) and indoor propagation ($\eta \sim 4$) [34]. Given that energy consumption is greatly influenced by particular hardware implementations, we provide relative energy consumption outcomes. As a guideline we follow previus accepted procedures [4], [35], i.e.: 1) Reception usually has an energy cost an order of magnitude lower than to Transmission, as Reception does not change transmitting power and, 2) Transmission has an energy cost two orders of magnitude higher than Process, due to the low energy efficiency of RF power amplifiers. These differences can increase up to three orders of magnitude for specific hardware.

The estimation of the number of operations required for each algorithm is also performed. SODCC involves the estimation of the signal subspace dimension with FSD, which is performed for all intermediate and final clusters. Intermediate clusters are by definition unstable (see Section III-B), thus cluster fusion

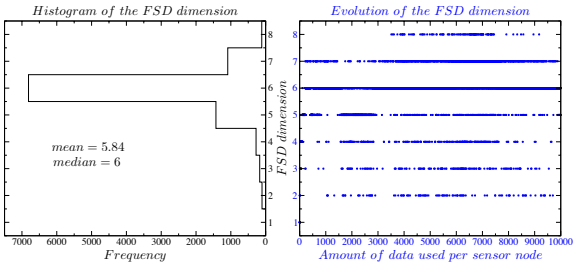


Figure 3. Histogram (left) and time evolution (right) of the signal subspace dimension of the dataset estimated with FSD using an increasing amount of samples per node.

must be performed. Final clusters do not require any cluster fusion as they are stable and are the outcome of the clustering algorithm. Therefore, the amount of operations needed by the SODCC algorithm is $\hat{d} \times (4N_c)^2$ for each intermediate and final cluster. CPPCA involves matrix multiplications, so the total amount of operations is $2 \times KN_cM$. Finally, as the computational burden of LEACH, Persistent and ADG-CH is much lower [26], we neglect the amount of operations needed in these last three cases.

V. Results

In the present Section we show and analyze the results (see Section IV-C) obtained for all the fourteen network settings (see Section IV-B). Firstly, we assess the time structure of the data and the network synchronization issue, in order to better understand the results obtained for each case.

A. Structure of data

The spatio-temporal structure of the sensed data in the WSN is irrelevant to the performance of both LEACH and Persistent. However, it plays a decisive role in the second phase of SODCC. The clusters formed by SODCC are located around sets of proximal nodes whose data variance can be explained with a minimal signal subspace dimension of the set [36]. Thus, the structure of the data affects both the clusters locations, and their sizes.

Figure 3 shows the time structure of the temperature dataset, obtained by performing a temporal analysis of the FSD statistic. The most probable dimension of the signal subspace in the whole dataset is $\hat{d} = 6$. Thus, it can be expected that configurations with an average size of 6 nodes per cluster have better performance, from the data quality point of view, as the spatio-temporal correlations of the temperature field will be better captured. This value also guided the set of the configuration parameter $B = 6$ for the Persistent algorithm, required prior to its initialization.

B. Network synchronization

Node synchronization is crucial depending on the measurements carried out by the WSN. In the present work, it is presumed that the synchronization has been independently acquired (e.g. with a centralized synchronization configuration

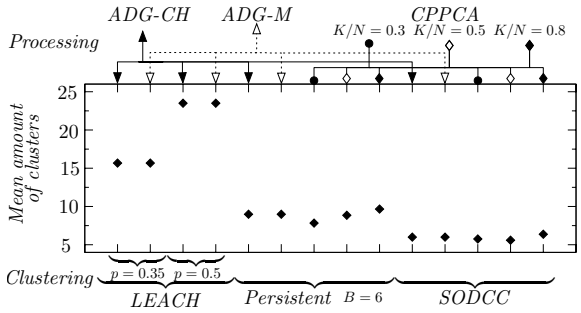


Figure 4. Mean amount of clusters in the WSN, computed for each combination of clustering and in-network processing algorithms.

or a broadcasting setting) from the clustering protocols. Therefore, the additional overhead can be considered identical for all the presented algorithms.

However, if we presume that synchronization is performed at cluster level, local message interchange to acquire synchronization depends on the total number of nodes in the network. As the authors previously pointed out [6, Eq. (15)], the average cluster size function for the SODCC algorithm scales as

$$\langle N_i \rangle = \frac{2}{\tau - 1} + \frac{2N^{2-\tau}}{2 - \tau} \tag{5}$$

where $\tau = 1.13$ in the present case. Thus, the SODCC algorithm exhibits scalability as $\langle N_i \rangle$ grows at a similar rate as N. Therefore, the growth of the synchronization overhead is moderate versus other clustering algorithms that do not exhibit scalability.

C. Number of clusters

The average number of obtained clusters, and therefore the ease of the network management, shows huge differences between the fourteen considered strategies (see Figure 4). For LEACH strategies, approximately one in every three nodes are CH for $p = 0.35$, leading to an average amount of clusters close to one third of the network size ($N = 47$). Similarly, for $p = 0.5$, the average amount of clusters is close to half of the network size. Also due to the pre-selected configuration parameters, strategies that use Persistent lead to an average amount of clusters close to 8, meaning that approximately one sixth of the nodes in the network are CH.

The behaviour of the SODCC algorithm is slightly different. From Figure 3 we deduce that the expected average cluster size should be close to 6 and that the average amount of clusters should be close to 8. However, it seems counterintuitive that the average amount of clusters is close to 6 and the average cluster size is close to 8. This fact reveals that, although 6 is a highly probable number of nodes per cluster, larger clusters exist. The minor variations in the number of clusters for the same clustering algorithm is due to the different initial random conditions. These variations are statistically negligible.

The relation between the network management facility and the average amount of clusters is reversed. Therefore, under this assumption, we can conclude that the network management for a WSN with N nodes clustered with SODCC is easier, compared to the LEACH and Persistent algorithms.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2016.2551466, IEEE Sensors Journal

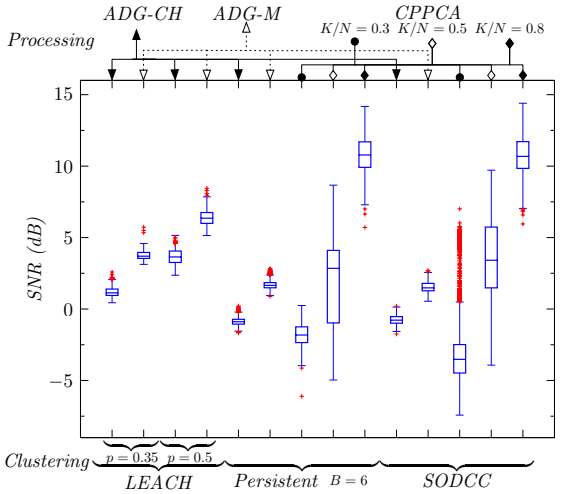


Figure 5. Boxplot of the SNR computed for each combination of clustering and in-network processing algorithms considered.

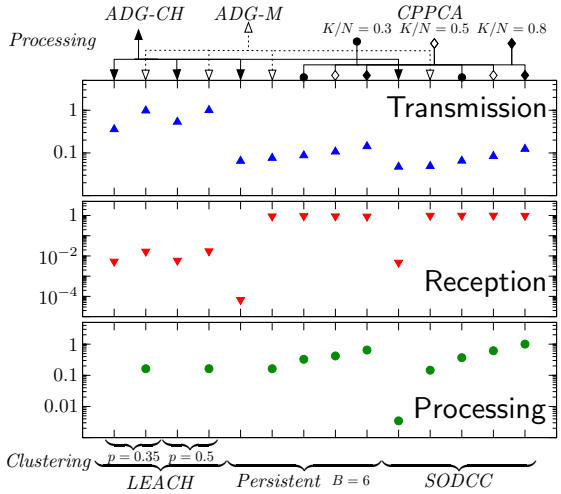


Figure 6. Energy consumption for each of the clustering and in-network processing algorithms combinations. The total energy consumption is divided in energy for transmission (top), reception (middle) and processing (bottom). In each case, the energy consumed is normalized with respect to the combination with higher energy consumption.

D. Data reconstruction quality

The fidelity of the reconstruction is a generally overlooked aspect for WSN performance evaluation. Figure 5 shows the assessment of the data quality by means of the SNR of the dataset reconstruction in the DFC. The LEACH+ADG-M strategy predictably fares better than its ADG-CH counterpart, for both values of p. The statistical dispersion of the SNR obtained for all LEACH+ADG strategies is minimal, due to the high temporal and spatial granularity of the nodes involved in the measuring process. For example, for LEACH+ADG-CH with $p = 0.5$, the DFC receives on average data from half of the nodes in one round and from the other half of the nodes in the next round. The low dispersion in SNR for both ADG algorithms is also apparent in all other combinations, with a loss of 2 dB approximately.

Strategies involving CPPCA are distinguished by a large dispersion in the reconstruction performance at any compression ratio. Moreover, the reconstruction performance gets exponentially better as the compression ratio K/N is reduced, obtaining improvements of 10 to 12 dB compared with Persistent+ADG and SODCC+ADG strategies and of 5 to 7 dB with LEACH strategies.

Thus, although the LEACH strategies result in fair and reliable performance figures, the usage of CPPCA holds in store a huge potential for improvement in performance. No clear advantage in reconstruction performance can be seen between the Persistent and SODCC strategies involving CPPCA. But bear in mind that, until SODCC is self-organized in terms of nodes-per-cluster, Persistent needs this proportions to be preset. With the optimal setting of Persistent to $B = 6$ nodes per cluster, the difference in SNR is neglectful. However, this should not be the case in a general setting where SODCC is bound to outperform Persistent as it operates without prior settings.

E. Energy consumption

Figure 6 shows the total energy consumption of the network for Transmission (top subfigure), Reception (middle) and Process (bottom), relative to the maximum energy consumption in each case. Note that there are clear differences between the fourteen network strategies.

For the Transmission case, all LEACH strategies consume more energy than any other considered strategy, mainly due to the single-hop communication with the DFC. In more detail, LEACH+ADG-M requires more energy due to higher data traffic between sensor nodes and the CH. Strategies involving either Persistent or SODCC are one order of magnitude below those with LEACH, with minimal differences among any processing technique.

On the other hand, the energy consumption for LEACH strategies in Reception is, in general, two orders of magnitude below those of the other clustering algorithms. However, the Persistent+ADG-CH strategy allows us to observe an additional impact of the LEACH cluster formation in the energy efficiency. The corresponding result indicates a significantly lower consumption when compared with LEACH+ADG-CH. This in-network processing strategy does not involve communication between sensor nodes and the CH, and all packet receptions are caused by the cluster formation process. Therefore, the energy required by LEACH to operate is two orders of magnitude higher than the one required by Persistent. As SODCC includes the calculation of the FSD statistic, which requires communication between sensor nodes and the CH, the previous analysis can not be performed with the same assumptions.

Regarding Processing, the assumption that ADG-CH has negligible computational burden enables the analysis of the energy invested by SODCC estimating the FSD signal subspace dimension. By using the results of the SODCC+ADG-CH setting, it is clear that the cost associated with the FSD calculation is minimal. The rest of the strategies have energy consumptions that span differences in less than an order of magnitude. Based on these results, we confirm that it

is always preferable to use processing-oriented strategies to transmission-oriented ones with respect to the power budget.

F. Discussion

In summary, we see that, the LEACH strategies perform reliably in terms of measurement field reconstruction, but they are not competitive with respect to Persistent or SODCC in terms of the energy consumption for both the Transmission (the most power consuming subsystem in a WSN) and Reception cases. This seems to be contradictory with the claims that LEACH is an energy efficient WSN clustering algorithm. Typically, LEACH implementations have used low values of $p \approx 0.1$ [10], resulting in larger clusters that, in conjunction with the ADG algorithms, assuredly degrade the fidelity of the measured field reconstruction. Furthermore, there is only a small margin for improvement in LEACH strategies in terms of data quality as more sophisticated in-network processing requires critical changes in the LEACH operation.

By comparing the self-organizing algorithms Persistent and SODCC, no significant differences can be perceived either in terms of reconstruction fidelity or in power consumption. However, as noticed in the previous subsections, while Persistent requires the *a priori* set budget B, i.e. the desired cluster size, no prior adjustable parameter is needed for SODCC. The optimal selection of B for a high data quality has to consider both the node deployment density and the spatial correlations of the data [6]. But, while the node density can be easily known, the spatial correlation of the data is considerably more difficult to obtain or estimate.

The network management once the cluster configuration is finished seems to be easier in the case of SODCC, as a lower amount of clusters is obtained. This result represents an additional benefit of SODCC over Persistent, and an additional support for the use of data-coupled clustering algorithms. Cluster configuration, which is coupled to the measured field, results in the lowest amount of clusters with the minimum cluster size to allow an efficient data processing. Therefore, these approaches are able to offer a better usage of the overall resources of the WSN.

The usage of CPPCA significantly contributes to the reduction of Transmissions energy consumption. Furthermore, the potential to increase the fidelity of the reconstruction (increasing K/N) with limited energy consumption is remarkable. Therefore, from the combined restrictions of reconstruction fidelity, minimum energy consumption and no prior required knowledge or settings, the SODCC+CPPCA setting outperforms any of the other examined strategies.

The previous discussion refers to the performed experiments, with temperature data. The advantages of temperature data are the slow temporal variation and that the second order statistics are suitable data representatives. However, for different environments where other physical variables are measured, the second order statistics may be not so suitable. Some examples of these kind of data include WSNs that monitors the opening of doors and windows for security applications or WSNs for rainfall monitoring. For these environments, where the second order statistics of the data are not the best guides for network clustering, LEACH+ADG strategies would be more suitable choices, even at the expense of a higher energy consumption.

VI. Conclusions

In the present work we have presented a thorough evaluation of the performance of the SODCC+CPPCA network setting, in terms of both the data quality and the energy efficiency, as they are the usual tradeoffs. We have shown that energy efficiency is not sacrificed for diminishing the distortion in the reconstructed field. Moreover, we have shown that the network management is facilitated by SODCC, as the cluster configuration is consistent with the internal structure of the measured data. We have also performed a fair comparison with other network settings that also combine clustering and in-network processing algorithms. Strategies with single-hop communications, as LEACH, exact a penalty on energy efficiency and strategies that used a representative node, as ADG-CH, achieved energy efficiency at the cost of losing relevant data from the correlated nodes in the cluster. In addition, the requirement of the *a priori* determination of the budget B for Persistent, a self-organized clustering algorithm, was critical for its optimal behaviour. Only the SODCC+CPPCA strategy achieved a perfect balance between quality of reconstruction, controlled by the compression ratio, and the energy expenditure of the data gathering process. Thus, a change in the design trend for WSN algorithms from transmission-oriented to processing-oriented ones is encouraged.

As future extensions, the adaptive version of the SODCC algorithm that allows for the reconfiguration of the clusters following the evolution of the characteristics of the measured data is being developed. In this case, the energy expenditure will also depend on the rate of variation of the measured field. Devising robust algorithms to preserve the balanced performance of the SODCC algorithm in rapidly changing environments will also be the focus of future research.

References

[1] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Comput Commun*, vol. 30, no. 14-15, pp. 2826–2841, 2007.

[2] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE T Inf Theory*, vol. 46, no. 2, 2000.

[3] H. El Gamal, "On the scaling laws of dense wireless sensor networks: the data gathering channel," *IEEE T Inf Theory*, vol. 51, no. 3, pp. 1229–1234, 2005.

[4] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw*, vol. 7, no. 3, pp. 537–568, 2009.

[5] M. I. Chidean, E. Morgado, J. Ramiro, and A. J. Caamano, "Self-Organized Distributed Compressive Projection in Large Scale Wireless Sensor Networks," in *Proc PIMRC*, 2013, pp. 2000–2004.

[6] M. I. Chidean, E. Morgado, E. del Arco, J. Ramiro-Bargueño, and A. J. Caamaño, "Scalable Data-Coupled Clustering for Large Scale WSN," *IEEE T Wireless Commun*, vol. 15, pp. 4681–4694, 2015.

[7] X. Liu, "A Survey on Clustering Routing Protocols in Wireless Sensor Networks," *Sensors*, vol. 12, pp. 11 113–11 153, 2012.

[8] H. Gupta, V. Navda, S. Das, and V. Chowdhary, "Efficient gathering of correlated data in sensor networks," *ACM Tran on Sensor Networks*, vol. 4, no. 1, pp. 1–31, Jan. 2008.

[9] Y. Ma, Y. Guo, X. Tian, and M. Ghanem, "Distributed Clustering-Based Aggregation Algorithm for Spatial Correlated Sensor Networks," *IEEE Sensors J*, vol. 11, no. 3, pp. 641–648, 2011.

[10] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc Hawaii Intl Conf on System Sciences*, 2000.

[11] ——, "An application-specific protocol architecture for wireless microsensor networks," *IEEE T Wireless Commun*, vol. 1, no. 4, pp. 660–670, 2002.

[12] C. Liu, K. Wu, and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," *IEEE T Parallel Distrib Syst*, vol. 18, no. 7, pp. 1010–1023, 2007.

[13] A. C. Frery, H. S. Ramos, J. Alencar-Neto, E. Nakamura, and A. A. F. Loureiro, "Data Driven Performance Evaluation of Wireless Sensor Networks," *Sensors*, vol. 10, no. 3, pp. 2150–2168, 2010.

[14] R. M. Assunção, M. C. Neves, G. Câmara, and C. da Costa Freitas, "Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees," *Intl J of Geogr Inf Sci*, vol. 20, no. 7, pp. 797–811, 2006.

[15] C. Wang, H. Ma, Y. He, and S. Xiong, "Adaptive Approximate Data Collection for Wireless Sensor Networks," *IEEE T Parallel Distrib Syst*, vol. 23, no. 6, pp. 1004–1016, 2012.

[16] C.-C. Hung, W.-C. Peng, and W.-C. Lee, "Energy-Aware Set-Covering Approaches for Approximate Data Collection in Wireless Sensor Networks," *IEEE T Knowl Data Eng*, vol. 24, no. 11, pp. 1993–2007, 2012.

[17] B. Gedik, L. Liu, and P. S. Yu, "ASAP: an adaptive sampling approach to data collection in sensor networks," *IEEE T Parallel Distrib Syst*, vol. 18, no. 12, pp. 1766–1783, 2007.

[18] H. Jiang, S. Jin, and C. Wang, "Prediction or not? An energy-efficient framework for clustering-based data collection in wireless sensor networks," *IEEE T Parallel Distrib Syst*, vol. 22, no. 6, pp. 1064–1071, 2011.

[19] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc INFOCOM*, 2003, pp. 1713–1723.

[20] R. Krishnan and D. Starobinski, "Efficient clustering algorithms for self-organizing wireless sensor networks," *Ad Hoc Netw*, vol. 4, no. 1, pp. 36–59, 2006.

[21] F. Chiti, R. Fantacci, E. Dei, and Z. Han, "Context aware clustering in vanets: A game theoretic perspective," in *IEEE Int Conf on Commun (ICC)*. IEEE, 2015, pp. 6584–6588.

[22] G. Xu and T. Kailath, "Fast subspace decomposition," *IEEE T Signal Process*, vol. 42, no. 3, pp. 539–551, 1994.

[23] M. C. Vuran, Ö. B. Akan, and I. F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks," *Comput Netw*, vol. 45, no. 3, pp. 245–259, 2004.

[24] A. Jindal and K. Psounis, "Modeling spatially correlated data in sensor networks," *ACM Tran on Sensor Netw*, vol. 2, no. 4, 2006.

[25] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TAG: a Tiny AGgregation service for ad-hoc sensor networks," *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, 2002.

[26] V. Erramilli, I. Malta, and A. Bestavros, "On the interaction between data aggregation and topology control in wireless sensor networks," in *Proc. SECON*, 2004, pp. 557–565.

[27] J. E. Fowler, "Compressive-Projection Principal Component Analysis," *IEEE Tran Image Process*, vol. 18, no. 10, pp. 2230–2242, 2009.

[28] B. Parlett, *The symmetric eigenvalue problem*. SIAM, 1980, vol. 7.

[29] B. Yang, "An extension of the PASTd algorithm to both rank and subspace tracking," *IEEE Signal Process Lett*, vol. 2, no. 9, pp. 179–182, 1995.

[30] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Tran on Acous, Speech, and Signal Process*, vol. 33, no. 2, pp. 387–392, 1985.

[31] A. J. Caamano, D. Segovia-Vargas, and J. Ramos, "Blind adaptive krylov subspace multiuser detection," in *IEEE Veh Technology Conf*, vol. 4. IEEE, 2001, pp. 2338–2341 vol.4.

[32] A. J. Caamano, R. Boloix-Tortosa, J. Ramos, and J. J. Murillo-Fuentes, "Hybrid higher-order statistics learning in multiuser detection," *IEEE Tran on Syst, Man, Cybern, C, Appl and Rev*, vol. 34, no. 4, pp. 417–424, 2004.

[33] G. Barrenetxea, F. Ingelrest, G. Schaefer, and M. Vetterli, "Wireless Sensor Networks for Environmental Monitoring: The SensorScope Experience," *IEEE Intl Zurich Seminar on Commun*, pp. 98–101, 2008.

[34] A. F. Molisch, *Wireless Communications, 2nd Edition*. Wiley, 2010.

[35] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun of the ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[36] M. I. Chidean, J. Muñoz-Bulnes, J. Ramiro-Bargueño, A. J. Caamaño, and S. Salcedo-Sanz, "Spatio-temporal trend analysis of air temperature in Europe and Western Asia using data-coupled clustering," *Global and Planetary Change*, vol. 129, pp. 45 – 55, 2015.