Distributed Clustering-Based Aggregation Algorithm for Spatial Correlated Sensor Networks

Yajie Ma, Yike Guo, Xiangchuan Tian, and Moustafa Ghanem

***Abstract*—In wireless sensor networks, it is already noted that nearby sensor nodes monitoring an environmental feature typically register similar values. This kind of data redundancy due to the spatial correlation between sensor observations inspires the research of in-network data aggregation. In this paper, an α-local spatial clustering algorithm for sensor networks is proposed. By measuring the spatial correlation between data sampled by different sensors, the algorithm constructs a dominating set as the sensor network backbone used to realize the data aggregation based on the information description/summarization performance of the dominators. In order to evaluate the performance of the algorithm a pattern recognition scenario over environmental data is presented. The evaluation shows that the resulting network achieved by our algorithm can provide environmental information at higher accuracy compared to other algorithms.**

***Index Terms*—Clustering, data aggregation, dominating set, sensor networks, spatial correlation.**

I. Introduction

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions [1], [2]. It has no fixed or predefined topology, thus allowing nodes to become active and inactive at any time. Communication could occur instantaneously between any devices within communication range. Designing autonomous self-organizing WSNs still remains a challenge requiring the development of new methods to enable collective context-awareness and in-network information processing by the network nodes.

Early in 1970, Tobler's first law of geography was formulated to state that "Everything is related to everything else, but near things are more related than distant things" [3]. This statistical observation implies that data correlation increases with decreasing spatial separation. In WSNs, it is already noted that nearby sensor nodes monitoring an environmental feature (e.g., temperature or humidity) typically register similar values. This kind of data redundancy due to the spatial correlation between sensor observations inspires the research of in-network data aggregation. In recent years, cluster-based approaches for data aggregation have attracted wide attention [4]–[12]. In these approaches, some sensors are selected as the Cluster Heads (CHs). Nodes in a cluster transmit their data to CH and the CH is used to represent data from cluster members facilitating transfer of aggregated data to the sink.

A. Exploitation of Sensor Data Aggregation

Most existing clustering algorithms are realized through selecting CHs stochastically by probability [4] and constructing clusters considering the energy consumption [6], [7] or resource constraints including bandwidth [8], load balancing [9], and network topology structures [10], [12]. A survey of different methods is given in [13]. Most such approaches for data aggregation take spatial or network distance into account. They, however, typically ignore the data correlation (such as network topology based aggregation) or only assume ideal data correlations (such as defining the correlation as same readings in different sensors). Recent research exploiting data correlation in sensor networks reveals the effects of spatial correlation on energy consumption and routing protocol design [14]–[20]. Exploring such data correlation makes the design of large-scale multihop sensor networks feasible [21]. It also provides a basis for embedding collective context awareness across the nodes of the network.

Clustering techniques can provide an architectural framework for exploring data correlation in sensor networks. Results presented in [20] conclude that codes and routes must exist to enable every node in a WSN to have enough information to form an estimate of the sample available at every other node within a prespecified distortion value D. The result holds as long as $\mathrm{O}(R_{S1\cdots SN}(D)) \leq \mathrm{O}(L\sqrt{N})$ is satisfied, where $R_{S1\cdots SN}(D)$ is the joint rate/distortion function of all samples in the field; L is the finite capacity of links and N is the total number of nodes in the field. Based on this result, a cluster-based hierarchical network structure can be set up with the constraint of the total distortion being less than D, in which, some of the nodes can be chosen to represent their close neighbors thus reducing redundancy in both data processing and transmission.

B. Motivation and Contributions

Some recent research on clustering and data aggregation with respect to the spatial correlation in sensor networks, e.g., [22] investigates the nonuniform correlation of sensor data in enclosure spaces, such as a room. In such a highly correlated region (HCR), the correlations between sensors are determined

Manuscript received May 20, 2010; accepted June 14, 2010. Date of publication September 23, 2010; date of current version January 26, 2011. This work was supported in part by the EPSRC Project Mobile Environmental Sensing System Across a Grid Environment (MESSAGE) under Grant EP/E002102/1 and jointly funded by the Engineering and Physical Sciences Research Council and the Department for Transport. The associate editor coordinating the review of this paper and approving it for publication was Dr. Subhas Mukhopadhyay.

Y. Ma is with the College of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, 430081, China (e-mail: yajie.ma@ymail.com).

Y. Guo, X. Tian, and M. Ghanem are with the Department of Computing, Imperial College London, London, SW7 2BW, U.K. (e-mail: y.guo@imperial.ac.uk; x.tian@imperial.ac.uk; m.ghanem@imperial.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2010.2056916

predominantly only by there spatial distances. The CHs are elected stochastically at first and then adjusted according to the remaining battery levels. Using this scheme, the applications are restricted to HCR environments and the construction of clusters takes several rounds of message exchange and correlation computation. A similar scheme using the spatial distance of sensors as the judgment of correlation can be seen in [23]. The approach defines a weight for each sensor's data which depends on the distance from the sample position to the target position. Another method proposed in [24] and [25] addresses the spatial correlation measurement by calculating the offset between different sensor readings. This approach simply calculates the error between two readings. If the error is within a tolerable range, then these two readings are correlated. This kind of correlation judgment method can be used only in scenarios where the sampled data have only one dimension. An approach presented in [26] calculates spatial correlation using a probability distribution functions and correlation coefficients based on discrete readings. A high correlation coefficient of two readings means one is attacked by another. Then, the attacked reading can be replaced by a previous reading and the transmitted data are reduced. However, if none or few attacks are detected, the scheme is invalid or less efficient.

Most approaches described above considering the correlation-based data aggregation in sensor networks have two main shortcomings.

1) The judgment of the spatial correlation is based on geographic distance of sensors or one-dimension tolerance error of different sensors' readings. The applications of these schemes are restricted to environments of high sensor density and/or highly spatial correlated regions. None of them considers the spatial correlation measurement of multidimensional data.
2) The CHs are selected stochastically. The data correlation between a CH and its neighbors is not considered. Consequently, each CH may have no capability to provide precise description/summarization of its cluster members. Moreover, Each CH must collect data from its members and then execute complex data aggregation or fusion algorithms.

To overcome these shortcomings, we propose a new distributed clustering algorithm based on the dominating set theory to choose the CHs and construct clusters by measuring the spatial correlation between sensors. When the clusters are constructed, the data sampled in each CH have very high correlation with the data sampled in its cluster members. Consequently, only the CHs, but no cluster members, need to do the data sampling work, which means the sensor networks are aggregated to a CH backbone and the data transmitted in the sensor network are reduced remarkably without any extra data aggregation algorithm. In order to measure the efficiency of the algorithm for information description/summarization, we introduce a pattern recognition scenario based on practical data from a real environment.

The main contributions of this paper are: First, we define a weight to calculate the spatial correlation between sensor data. By using this weight, not only one-dimension data, but also multidimension data can be studied. Second, unlike the existing clustering approaches that the CHs must collect data from its members and then execute the data aggregation algorithm, our algorithm chooses the CHs that can represent the data features of its members. Thus, CHs only need to transmit the data sampled by themselves to the sink. Such an aggregated CHs backbone structure can provide very high efficiency in data aggregation. It also forms the basis for improved context-awareness within the network.

C. Paper Layout

The remainder of this paper is organized as follows. Section II discusses related work on clustering algorithms. Section III presents the α-local spatial clustering algorithm by describing each part of the algorithm in detail, including weight computation, CHs selection and the construction of clusters. In Section IV, we analyze the complexity of the algorithm based on an experiment to investigate the aggregation performance of the algorithm. Section V presents the simulation results of pattern recognition scenario of our algorithm. Finally, in Section VI, we provide our key conclusions and directions for future work.

II. Related Work

Many clustering algorithms have been proposed for ad-hoc and sensor networks recently. LEACH [4] is the most famous application-specific algorithm that uses clustering to prolong the network lifetime. As clustering is vital for efficient resource utilization and load balancing in large-scale sensor networks, it is not surprising that an increasing amount of research interest has been drawn towards clustering algorithms during the last few years. In general, such research can be classified primarily into two perspectives either finding a smallest set of the CHs based on Graph Theory, or finding an optimal set of the CHs based on residual energy of each node.

From the graph theory perspective, heuristic algorithms are always used to generate approximate results based on either a centralized or distributed model of operation. For a centralized model, Guha and Khuller propose two CDS (Connected Dominating Set) construction strategies in [27], which contain two greedy heuristic algorithms with bounded performance guarantees. Other algorithms, e.g., [28] and [29], are motivated by either of these two heuristics. However, for large-scale WSNs, a distributed CDS algorithm could be more effective due to the lack of a centralized administration. An example of the distributed implementations of [27] is provided in [30]. Other greedy heuristics including [31]–[33] have also come under investigation recently. Moreover, distributed CDS construction approaches have also been investigated based on Maximum Independent Set [34], multipoint relaying [35], and Spanning Tree [36].

From the residual energy perspective, LEACH [4] is a typical example. This algorithm uses randomized rotation of the CHs to distribute the energy load evenly among the sensor nodes in a network. The LEACH algorithm is employed in various systems, such as PEGASIS [37], TEEN [38], HEED [39], etc. The PEGASIS mechanism is a chain-based power efficient protocol. The chain can be constructed easily by using a greedy algorithm. In order to balance the overhead involved in communication between the chain leader and the base station, each node in the chain takes turn to be the leader. The TEEN algorithm is designed for time-critical applications and the sensor transmits the data to the sink only when the collected data is greater than a predefined threshold. In HEED, a variable called "cluster radius" is introduced to define the transmission power used for intracluster broadcast. In general, all the above methods require reclustering after a period of time because the CHs are always in high workload.

III. Algorithm Description

Our proposed algorithm is composed of two procedures: the distributed Cluster Head Selection procedure (CHS) and the Cluster Construction procedure (CC). In this section, we will first introduce the calculation of the *Weight*, which is a key variable for each procedure of the algorithm and then describe the detail of the algorithm itself.

A. Weight Calculation

The calculation of node's *Weight* tries to find out a measurement for each node to identify in what degree a node is correlated with other nodes within its communication radius.

Given an undirected graph G (we denote it as "graph G" in the rest of this paper), with all the vertexes set V and all the Edges set E. Let the number of nodes in V be $|V|$. For every $i \in V$, let i also be the sequence identification of this node $(1 \le i \le |V|)$. We have some fundamental definitions concerning the subsets of V.

Definition 1: **$\boldsymbol{\alpha}$-Neighbor Set $\boldsymbol{N}$**. For a predefined communication radius α of all the nodes, let $\Gamma\alpha(i)$ be the set of vertexes within the circle of the communication radius of i. For any $j \in \Gamma\alpha(i)$, if there is a transmission route from i to j, then j belongs to the α-Neighbor Set of i, denoted as $j \in N(i)$ $(N(i) \subseteq V)$.

For a measured random field, each sampled data is represented by an n-dimensional feature vector. Hereby, at a specified sample time stamp t, for a node $i \in V$ and any of its neighbor $j \in N(i)$, the sampled data in i and j can be denoted as $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n)$ respectively. We define the distance between the measurements of i and j by the Euclidean distance between X and Y as

$$d_{ij} = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \cdots + |x_n - y_n|^2}. \quad (1)$$

Then, the expected value of d_{ij} is

$$E(d_{ij}) = \overline{d_i} = \frac{1}{|N(i)|} \sum_{j \in N(i)} d_{ij} \quad (2)$$

where $|N(i)|$ is the number of nodes in $N(i)$.

The deviation of d_{ij} is

$$\begin{aligned} D(d_{ij}) &= E\left\{[d_{ij} - E(d_{ij})]^2\right\} = \frac{1}{|N(i)|} \sum_{j \in N(i)} (d_{ij} - \overline{d_i})^2 \\ &= E\left(d_{ij}^2\right) - [E(d_{ij})]^2 = \frac{1}{|N(x)|} \sum_{j \in N(i)} d_{ij}^2 - \overline{d_i}^2. \quad (3) \end{aligned}$$

Definition 2: Spatial Correlated Weight $\boldsymbol{w}$. For any node $i \in V$, $j \in N(i)$, $|N(i)|$ is the number of nodes in $N(i)$. The sampled data in i and j are $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n)$ respectively. The Spatial Correlated Weight w_i $(0 \le w_i \le 1)$[1] of node i is defined as

$$w_i = \frac{\left[\sum\limits_{j \in N(i)} |d_{ij} - \overline{d_i}|\right]^2}{|N(i)|^2 D(d_{ij})} = \frac{\left[\sum\limits_{j \in N(i)} |d_{ij} - \overline{d_i}|\right]^2}{|N(i)| \sum\limits_{j \in N(i)} (d_{ij} - \overline{d_i})^2}. \quad (4)$$

The definition of the Spatial Correlated Weight considers the average spatial distance deviation between each node i and its neighbors within a predefined communication range. Large value of w_i (or small value of $D(d_{ij})$) implies small distance variation between node i and its α-Neighbors, which means i has high spatial correlation with its neighbors.

B. Cluster Head Selection

We address the cluster head selection problem by a distributed algorithm based on the Weighted α-Dominating Set. Nodes belong to this set will become the CHs. Here, we give the definitions of α-Dominating Set and Weighted α-Dominating Set:

Definition 3: **$\boldsymbol{\alpha}$-Dominating Set**. Given a graph $G = (V, E)$, we say that a set $D \subseteq V$ is an α-dominating set if every node of G either belongs to D or is within a distance no more than α to one or more nodes of D.

Definition 4: **Weighted $\boldsymbol{\alpha}$-Dominating Set**. Given a graph $G = (V, E)$ together with a non-negative weight w_i for each node $i \in V$, the weight of a vertex set $D \subseteq V$ is defined as $w(D) = \sum_{i \in D} w_i$. With a predefined positive constant k, a Weighted α-Dominating Set D of G is an α-dominating set of G if $w(D) \le k$. If a node $i \in D$, then node i is a dominator; otherwise node i is a dominatee, which can be represented by a decision variable x_i:

$$x_i = \begin{cases} 1, & \text{if node } i \text{ is a dominator;} \\ 0, & \text{otherwise} \end{cases}.$$

Here, two possible situations are being considered when a node decides whether itself becomes a dominator or not:

1) A node has very low correlation with all its α-neighbors.
2) A node has very high correlation with most of its α-neighbors.

It is easy to note that nodes being in either of these two situations should be chosen as dominators. For the first situation, a node becomes a dominator without any cluster member. We call this kind of dominator the Isolated Dominator (ID); and for the second situation, a node becomes a dominator with at least one cluster member. We call this kind of dominator the General Dominator (GD). These two situations must be treated as the basic criteria of choosing dominators in the CHS procedure. Here, we give two definitions that will be used in the CHS procedure:

w^{LB}	a lower bound for all w_i, which is a predefined constant satisfying $0 < w^{LB} < 1$.
$w_i^{\max}$	for each node i, $w_i^{\max} = \max\{w_j \mid j \in N(i)\}$.

[1] According to Cauchy–Schwarz inequality, we get $[\sum_{j \in N(i)} (d_{ij} - \overline{d_i})]^2 \le |N(i)| \sum_{j \in N(i)} (d_{ij} - \overline{d_i})^2$. As the result, $0 \le w_i \le 1$.

The following pseudocode is the distributed cluster header selection algorithm applied in each node.

```
CHS Procedure

Input: $w^{LB}$ and the topology description of graph $G$
Output: Weighted $\alpha$-Dominating Set $D$
(Step1) $D = \Phi$;            /* $D$ is Null at the beginning */
(Step2) for each $i$: $i \in V$ pardo {   /* Parallel process for each $i$ */
        (2.1) node $i$ is a dominatee;
        (2.2) $w_i = \frac{\left[\sum_{j \in N(i)} |d_{ij} - \overline{d_i}|\right]^2}{|N(i)| \sum_{j \in N(i)} (d_{ij} - \overline{d_i})^2}$;
        (2.3) if $(w_i \leq w^{LB})$ $x_i = 1$;
        (2.4) if $(x_i == 1)$   node $i$ becomes an ID;
        } /* end for */
(Step3) for each $i$: $i \in V$ & (node $i$ is a dominatee) pardo {
        (3.1) calculate $w_i^{\max}$;
        (3.2) if $(w_i \geq w_i^{\max})$   $x_i = 1$;
        (3.3) else {
                $p_i = \log \frac{w_i^{\max} - w_i}{w_i - w^{LB}}$;
                $x_i = 1$ with probability $p_i$;
        }
        (3.5) if $(x_i == 1)$   node $i$ becomes a GD;
        } /* end for */
(Step4) for each $i$: $i \in V$ & (node $i$ is a dominatee) pardo {
        (4.1) if ($j$ is not a GD for all $j \in N(i)$)
                node $i$ becomes a GD;          } /* end for */
(Step5) $D = \{i | x_i = 1, i \in V\}$;
```

The key processing steps of the CHS procedure are steps 2, 3, and 4, where each node calculates the probability of becoming a dominator. In Step 2, the lower bound of the weight is used to find out the dominators that have very low correlations with their α-neighbors. In Step 3, the upper bound of the weight is used to make sure that the nodes having highest correlation with their α-neighbors are chosen to be the dominators. At the same time, probability function $p_i = \log((w_i^{\max} - w_i)/(w_i - w^{LB}))$ makes the nodes with comparatively lower correlation (smaller values of w_i) with their α-neighbors have higher probability to become dominators. While Step 4 is a complementary process which makes the set D satisfy the definition of α-neighbor dominating set. Step 4 can guarantee that, for any dominatee, if it is an α-neighbor of an ID, there must be at least one GD within the distance α of this dominatee.

C. Cluster Construction

After all the CHs are selected by the CHS procedure, each dominatee has to choose a cluster to join. The Euclidean distance is applied here to construct clusters. In CC procedure, if a dominatee can be dominated by several dominators, it must choose the nearest dominator (the Euclidean distance is smallest between them) to join. The details of the CC procedure are described as follows.

```
CC Procedure

1. Each GD $i$ ($i \in D$) broadcasts an INDICATOR message
   embedded with its identity to all its $\alpha$-neighbors $j$
   ($j \in N(i)$) to indicate its dominator status.
2. Each dominatee $j$ ($j \in N(i)$) chooses a cluster to join:
   a. If $j$ receives only one INDICATOR message from a
      dominator $i$, then it join the cluster of $i$ (denoted as
      $C_i$).
   b. If $j$ receives $n$ ($2 \leq n \leq$ |D|) INDICATOR
      messages from a set of dominators $S$ ($S \subseteq D$), then
      $j$ chooses a $C_i$ ($i \in S$) to join if it satisfies:
            $d_{ij} = \min\{d_{kj} | k \in S\}$
3. If dominatee $j$ ($j \in N(i)$) decides to join $C_i$, it sends a
   JOIN message embedded with its identity to $i$.
4. If a dominator $i$ receives a JOIN message from a
   dominatee $j$ ($j \in N(i)$), it sends back an ACK message
   to $j$. Then $i$ is the CH of $C_i$ and $j$ is a member of $C_i$.
```

IV. Performance Analysis

A. Complexity Analysis

For the time complexity, in each of the steps 2, 3, and 4 of CHS procedure, the time complexity is $O(n)$; in CC procedure, the time complexity is also $O(n)$. Therefore, the time complexity of the whole algorithm is $O(n)$.

For the message complexity, suppose the maximum degree of the sensor network topology graph is $\Delta = \max_{i \in V}\{|N(i)|\}$ $(1 < \Delta < n - 1)$. The message complexity is $O(\Delta n)$ in CHS procedure Step 4; in CC procedure Step 1 and 4, the message complexities are also $O(\Delta n)$; in CC procedure Step 3, the message complexity is $O(n)$. Therefore, the message complexity of the whole algorithm is $O(\Delta n)$.

B. Size of the Dominating Set

In this experiment, we calculate the average number of dominators $|D|$ as the size of the weighted α-dominating set, as well as evaluate the impact of different number of the IDs on $|D|$.

We use a topology generator to generate random topologies in an area. For different topology parameter values, the random graph is generated and simulated until a predefined confidence

TABLE I
RESULTS OF NETWORK AGGREGATION

Number of IDs (≈)	$5\%\lvert V\rvert$		$10\%\lvert V\rvert$		$15\%\lvert V\rvert$	
α	1 hop	2 hops	1 hop	2 hops	1 hop	2 hops
AAR	40.18 %	57.65 %	46.01 %	62.18 %	53.29 %	68.25 %

interval for the population mean is reached and, then, simulation results are measured by simply taking the average of all cases. Here, we achieve a precision of 1% with the 90% confidence interval of the dominating set. The values of w^{LB} is set specifically to ensure that the number of ID is about 5%, 10% and 15% of the total number of the sensors in each of the simulations. The communication radius α has two values: 1 hop and 2 hops, which means for a node i, the nodes within 1 hop or 2 hops of the communication distance are all its neighbors. We use a four-dimension air pollution dataset which has the average correlation coefficient $\overline{\rho} = 0.733$ with $\rho_{\max} = 0.998$ and $\rho_{\min} = 0.506$. Each node randomly picks up a data item from the dataset as its sampled data. Then these data are used to calculate the Spatial Correlated Weight w.

The experimental results are shown in Table I. The AAR is used to evaluate the network aggregation effect after the execution of our algorithm. If the total number of nodes in a sensor network is $|V|$ (in the experiments, the value of $|V|$ varies from 40 to 150), then the AAR can be calculated as

$$\text{AAR} = \left(1 - \frac{|D|}{|V|}\right) \times 100\%. \tag{5}$$

From Table I, we can see, under the α-local Spatial Clustering Algorithm, the sensor networks can be effectively aggregated. The communication radius affects the performance of the aggregation remarkably. Larger α always leads to smaller average number of dominators. When the number of IDs increase from $5\%|V|$ to $15\%|V|$, the values of AAR increase accordingly. In other words, the sizes of the weighted α-dominating set decrease. The reason is, for the CHS procedure, a GD can be chosen in either Step 3 or Step 4. However, in this algorithm, most of the GDs are chosen in Step 4 (this conclusion can be achieved from experiments and we omitted it because of the length of the paper). If we increase the number of IDs, a reasonably larger value of w^{LB} must be set. Thus, the values of p_i in CHS procedure Step 3.3 will be increased. Accordingly, the number of GDs generated by Step 3 will increase. Because Step 4 is a complementary process for the dominating set construction, as the result, the number of GDs generated by Step 4 will have a comparatively remarkable decrease. Hence, larger values of α and Number of IDs result higher network aggregation rate.

V. Simulation of Pattern Recognition Scenario

As the algorithm tries to achieve an aggregated sensor network without loosing the precise description/summarization of the research areas, we can evaluate the effectiveness of the α-local Spatial Clustering Algorithm by a variety of data mining methods, such as classification, pattern recognition, feature selection, and fuzzy sets, etc. We chose a pattern recognition scenario to group pollutants, measured in an urban area, into pollution clouds to study the distribution of the pollutants. This scenario gives us an easy and intuitive way to investigate the performance of our algorithm. In the scenario we use the classic K-means algorithm [40] to classify the data into K patterns. The purpose of the simulation is to reveal whether the pattern of the data sampled by a CH can be used to represent all the patterns of the data sampled by its members, and also to evaluate the accuracy of its operation.

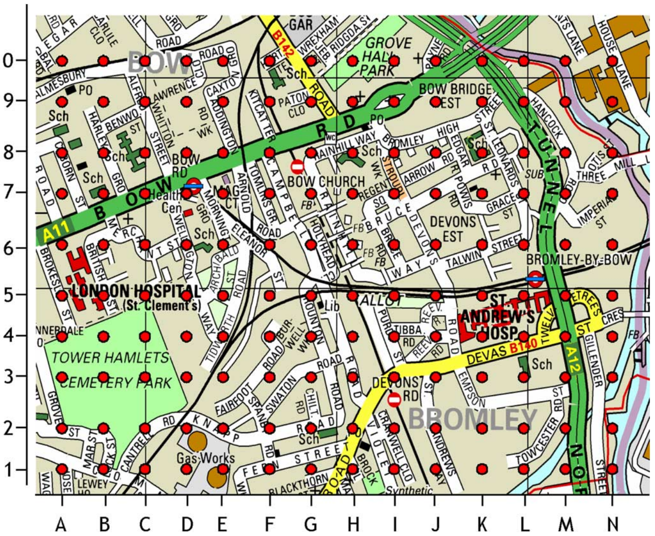


Fig. 1. One hundred forty sensors distributed in an area of east London.

The evaluation simulation consists of three steps.

1) Running the α-local Spatial Clustering Algorithm to generate the aggregated network.
2) Running the K-means algorithm in both of the original network and the aggregated network, then getting the pattern recognition results for both of the networks.
3) Generating the pollution pattern clouds according to the results in Step 2 for both the original network and aggregated network, then comparing the results.

The evaluation is based on our former research [41], [42] shown in Fig. 1. We use simulated air pollution data generated from a realistic scenario for the deployment of a sensor grid over a typical urban area in east London. The scenario is based on a distribution of 140 sensors (as the dots in Fig. 1) in the 1 km $*$ 1.4 km area collecting data over a typical day from 8:00 to 17:59 at 1-min intervals to monitor the pollution volumes of NO, NO_2, SO_2 and Ozone. Then, there are 600 data items for each node and totally 84000 data items for the whole network. Each data item is identified by a time stamp, a location, and a four-pollutant volume reading.

A. Pattern Recognition

In this step, we compare the pattern recognition results achieved by running the classic K-means algorithm [40] in both of the original network and the aggregated network. For the original network, we let the data collected from all the 140 sensors be the input of the K-means algorithm, and in the aggregated network generated by α-local Spatial Clustering Algorithm, as described above, only the data collected by the CHs are used for the pattern recognition calculation. At

TABLE II
ACCURACY PERFORMANCE OF PATTERN RECOGNITION

Number of IDs (≈)	$5\%\|V\|$		$10\%\|V\|$		$15\%\|V\|$	
α	1 hop	2 hops	1 hop	2 hops	1 hop	2 hops
APMM	95.64 %	83.64 %	93.57 %	81.71 %	90%	77.07 %

any measuring time stamp, the sampled data in node i is a four-dimension record with the readings of NO, NO_2, SO_2, and Ozone.

After running the pattern recognition algorithm, there will be K patterns for all the nodes participating the operation, and each node is assigned a pattern ID k $(k = 0, \ldots, K-1)$. The nodes with the same ID are in a pattern group, which means they have similar pollution features to each other, whereas different from other nodes.

B. Generating the Pollution Clouds

In this step, we visually use a different shapes to mark each pattern. For the original network, since each of the 140 nodes has a pattern identity, it is easy to generate the pollution clouds by marking the geographic area around each node with corresponding shape of its pattern identity.

For the aggregated network, only the dominators/CHs have the pattern identities after pattern recognition and all the members in a cluster will get same pattern identity as their CH. As each dominator may have m $(m >= 0)$ members, the pollution clouds for the aggregated network can be generated as follow: for each dominator, mark the entire geographic area of its cluster with corresponding shape according to the pattern identity of the dominator.

C. Accuracy Performance

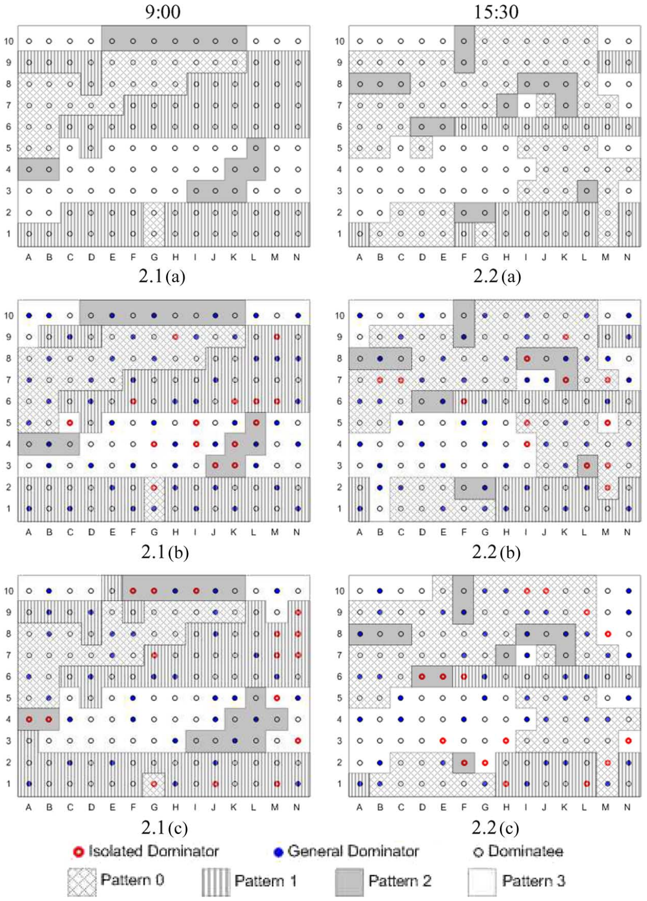
In this section, we evaluate the accuracy of the information description/summarization by the aggregation network. We investigate in how much degree the pattern recognition results of the aggregated network can match those of the original network.

Assign the value of $K = 4$ for K-means algorithm, which means 4 patterns will be generated after the pattern recognition and the pattern IDs are $k = 0$, 1, 2, and 3 for each pattern. Suppose the total number of nodes be $|V|$ and let X^i denote the dataset at node i. Let $L^i_{km}(x)$ and $L^i(x)$ denote the cluster membership of sample x $(x \in X^i)$ at node i under K-means algorithm in [40] and our algorithm, respectively. We define the Average Percentage Membership Match (APMM) as

$$\mathrm{APMM} = \frac{1}{|V|}\sum_{i=1}^{|V|}\frac{\left|\{x \in X^i : L^i(x) = L^i_{km}(x)\}\right|}{|X^i|} \times 100\%. \tag{6}$$

The APMM values in different scenarios are listed in Table II.

From this table, we can see when α is 1 hop, the values of APMM for different number of IDs are always above 90%. When α is 2 hops, the values of $APMM$ decrease by about 12% in comparison with each corresponding value of 1 hop. This is because if we increase α, then the dominating area of a dominator is enlarged, the correlation between the data collected from the dominator and its dominatees will decrease. Hence, the capability of a dominator to describe/summarize its dominatees will decrease as well. As the number of IDs increases from 5% to 15%, the value of APMM decreases. This result indicates that, as the number of the IDs increases, the AAR increases and the size of the weighted α-dominating set decreases (as Table I shows). However, the accuracy of the information represented by the aggregated network reduced.



2.1 (a)
2.2 (a)
2.1 (b)
2.2 (b)
2.1 (c)
2.2 (c)

Fig. 2. Pattern recognition results (number of IDs ≈ $10\%|V|$).

In order to give a more intuitive understanding, we draw the pattern clouds in Fig. 2 for the case of number of IDs ≈ $10\%|V|$, as well as the pattern clouds of the original network for comparison.

Here, we pick up two different time snapshots, 9:00 and 15:30 within a day. The two columns in Fig. 2 are the pattern recognition results of two time snapshots, respectively. While the first row is the results of the original network; the second and third rows are the results of the aggregated network with $\alpha = 1$ and $\alpha = 2$, respectively. Four different shapes are used to show four patterns clouds. We also mark different types of nodes. In the first row, as our clustering algorithm is not applied, only the original nodes are shown in the figures. For the aggregated network in the second and third rows, there are three types of nodes, ID, GD, and Dominatee.

From Fig. 2, we can see for each time snapshot, the aggregated network always presents very similar pollution patterns to the original network. The precision for $\alpha = 1$ is slightly higher than that for $\alpha = 2$, which just matches the calculation result in Table II.

TABLE III
ACCURACY PERFORMANCE COMPARISON

Algorithms	α-local spatial clustering	LEACH	EADAT	GCR
APMM	93.57%	76.07%	72.5%	70.36%

D. Comparison With Other Algorithms

In this section, an accuracy performance experiment is made to compare our algorithm with cluster-based (LEACH) [4], tree-based (EADAT) [43], and grid-based [44] data aggregation algorithms.

In our algorithm, the Number of IDs $\approx 10\%|V|$ and $\alpha = 1$ hop.

In LEACH, according to the analysis of the optimum number of clusters, if we apply equation (19) in [4] with total number of nodes $N = 140$ in a 1 km $*$ 1.4 km region (as shown in Fig. 1), while keeping all the other parameter values, the optimum value of expected number of CHs k is between 14 and 87. We choose $k = 70$ (this number is close to the value of $|D|$ in our algorithm in this case). To simplify the experiment, we only use the cluster result of one round to calculate the APMM.

For EADAT, we suppose the sink is located in the boundary of the area. Each node randomly generates a scalar between 0 and 50 to be its residual power. Each leaf node chooses a nonleaf node for data aggregation that requires shorter path and lower communication energy. This scenario is similar to $\alpha = 1$ in our algorithm when considering neighbors' data for aggregation.

For the grid-based algorithm of [44] (we refer to it as GCR), each node is randomly assigned an ID between 1 and 140. The grid width is set to 200 m. This grid size is also similar to $\alpha = 1$ in our algorithm when considering neighbors' data for aggregation.

All the above algorithms execute the same data aggregation scheme as our algorithm, which means, the pattern of the data sampled by a cluster head/nonleaf node/grid coordinator is used to represent all the patterns of the data sampled by its cluster members/leaf nodes/grid members.

Table III is the result of APMM for each of the algorithm. We can see that the $APMM$s of cluster-based, tree-based and grid-based algorithms are all less than 77%, which is much lower than $APMM$ of our algorithm. Therefore, our algorithm has better accuracy performance in data description/summarization.

VI. Conclusion and Future Work

In this paper, we proposed an α-local spatial clustering algorithm for WSNs. The algorithm can construct a dominating set as the sensor network backbone to realize the data aggregation, as well as consider the performance of the dominators in terms of their information description/summarization perspective. We discussed the time and message complexities of the algorithm, with the analysis of the size of the aggregated networks.

A pattern recognition scenario was also presented to investigate the information description/summarization capability of our algorithm. The experimental results show that the aggregated network can provide the environmental information in very high accuracy in comparison with the original network. Thus, this algorithm is useful for the applications such as the environmental surveillance where the sensors are always distributed in very high density.

Direct further work on the algorithm development includes analyzing its performance when jointly considering the size of the aggregated networks, the information representation capability, and the energy consumption. It also includes its extension to consider temporal correlation properties.

Our longer-term research in this area focuses on the development of a class of efficient distributed algorithms that support an elastic computation model for on-demand and real-time resource organization in sensor networks. Our aim is to develop methods that can focus the attention of the network resources only on relevant information for the task at hand. Such an approach not only helps in optimizing the performance characteristics of the network, but also helps in avoiding information overload when monitoring and analyzing large data sets.

References

[1] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 54–61, 2004.

[2] T. Haenselmann, "Sensornetworks," in *GFDL Wireless Sensor Network,* Aug. 2006. [Online]. Available: http://www.informatik.uni-mannheim.de/~haensel/sn_book

[3] W. R. Tobler, "A computer movie simulating urban growth in the Detroit region," *Economic Geography*, vol. 46, no. 2, pp. 234–240, 1970.

[4] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor network," *IEEE Trans. Wireless Commun.*, vol. 1, pp. 660–670, 2002.

[5] A. Manjeshwar, Q. Zeng, and D. P. Agrawal, "An analytical model for information retrieval in wireless sensor networks using enhanced APTEEN protocol," *IEEE Trans. Paralell and Distri. Syst.*, vol. 13, no. 12, pp. 1290–1302, 2002.

[6] O. Younis and S. Fahmy, "Heed: A hybrid, energy-efficient, distributed clustering approach for ad-hoc sensor networks," *IEEE Trans. Mobile Computing*, vol. 3, no. 4, pp. 660–669, 2004.

[7] S. Ghiasi, A. Srivastava, X. Yang, and M. Sarrafzadeh, "Optimal energy aware clustering in sensor networks," *Sensors*, vol. 2, pp. 258–269, 2002.

[8] M. Gerla, T. J. Kwon, and G. Pei, "On demand routing in large ad hoc wireless networks with passive clustering," in *Proc. IEEE Wireless Commun. Networking Conf. (WCNC2000)*, Chicago, IL, Sep. 2000, vol. 1, pp. 100–105.

[9] R. Virrankoski and A. Savvides, "TASC: Topology adaptive spatial clustering for sensor networks," in *Proc. IEEE Int. Conf. Mobile Adhoc and Sensor Systems*, Washington, DC, Nov. 2005, pp. 605–614.

[10] G. Gupta and M. Younis, "Performance evaluation of load-balanced clustering of wireless sensor networks," in *Proc. 10th Int. Conf. Telecommun.*, Tahiti, Papeete, French Polynesia, Mar. 2003, vol. 2, pp. 1577–1583.

[11] D. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," *Tran. Commun.*, vol. 29, no. 11, pp. 1694–1701, 1981.

[12] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A weighted clustering algorithm for mobile ad hoc networks," *J. Cluster Computing (Special Issue on Mobile Ad Hoc Networks)*, vol. 5, pp. 193–204, 2002.

[13] A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Commun.*, vol. 30, pp. 2826–2841, 2007.

[14] M. Lotfinezhad and B. Liang, "Effect of partially correlated data on clustering in wireless sensor networks," in *Proc. 1st Annu. IEEE Commun. Soc. Conf. Sensor and Ad Hoc Commun. Networks*, Santa Clara, CA, Oct. 2004, pp. 172–181.

[15] R. Cristescu and B. Beferull-Lozano, "Lossy network correlated data gathering with high-resolution coding," *IEEE Trans. Informat. Theory*, vol. 52, no. 6, pp. 2817–2824, 2006.

[16] J. Barros, C. Peraki, and S. D. Servetto, "Efficient network architectures for sensor reachback," in *Proc. IEEE Int. Zurich Seminar on Commun.*, 2004, pp. 184–187.

[17] M. Enachescu, A. Goel, R. Govindan, and R. Motwani, "Scale free aggregation in sensor networks," *Theoretical Comput. Sci.*, vol. 344, no. 1, pp. 15–29, 2004.

[18] D. Marco, E. Duarte-Melo, M. Liu, and D. L. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *Proc. Int. Workshop on Information Processing in Sensor Networks*, Palo Alto, CA, 2003, pp. 1–16.

[19] S. Pattem, B. Krishnamachari, and R. Govindan, "The impact of spatial correlation on routing with compression in wireless sensor networks," in *Proc. Int. Workshop on Information Processing in Sensor Networks*, Berkeley, CA, 2004, pp. 28–35.

[20] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," in *Proc. 8th ACM Int. Conf. Mobile Computing and Networking*, Altanta, GA, Sep. 2002, pp. 140–147.

[21] A. Scaglione, "Routing and data compression in sensor networks: Stochastic models for sensor data that guarantee scalability," in *Proc. IEEE Int. Symp. Informat. Theory*, Yokohama, Japan, Jun. 2003, p. 174.

[22] D. Maeda, H. Uehara, and M. Yokoyama, "Efficient clustering scheme considering non-uniform correlation distribution for ubiquitous sensor networks," *IEICE Trans. Fundamentals of Electron., Commun. Comput. Sci.*, vol. E90-A, no. 7, pp. 1344–1352, 2007.

[23] L. Badia, E. Fasolo, A. Paganini, and M. Zorzi, "Data aggregation algorithms for sensor networks with geographic information awareness," in *Proc. Int. Conf. Wireless Personal Multimedia Commun.*, San Diego, CA, Sep. 2006, pp. 1214–1218.

[24] S. Yoon and C. Shahabi, "The Clustered AGgregation (CAG) technique leveraging spatial and temporal correlations in wireless sensor networks," *ACM Trans. Sensor Networks*, vol. 3, no. 1, 2007.

[25] G.-Y. Jin and M.-S. Park, "CAC: Context adaptive clustering for efficient data aggregation in wireless sensor networks," *Networking*, vol. 3976, pp. 1132–1137, 2006.

[26] P. Schaffer and I. Vajda, "CORA: Correlation-based resilient aggregation in sensor networks," in *Proc. 10th ACM/IEEE Int. Symp. Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Chania, Crete, Greece, Oct. 2007, pp. 373–376.

[27] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

[28] X. Cheng, "Routing issues in ad hoc wireless networks," Ph.D. dissertation, Dept. Comput. Sci., Univ. Minnesota, Minneapolis, MN, 2002.

[29] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed construction of connected dominating sets in wireless ad hoc networks," *Discrete Algorithms and Methods for Mobile Comput. Commun.*, vol. 9, no. 2, pp. 141–149, 2004.

[30] B. Das and V. Bharghavan, "Routing in ad-hoc networks using minimum connected dominating sets," in *Proc. Int. Conf. Commun.*, Montreal, Canada, Jun. 1997, pp. 376–380.

[31] Y. Ma, Y. Guo, and M. Ghanem, "RECA: Referenced energy-based CDS algorithm in wireless sensor networks," *Int. J. Commun. Syst.*, vol. 23, no. 1, pp. 125–138, 2010.

[32] Z. Yuanyuan, J. Xiaohua, and H. Yanxiang, "A distributed algorithm for constructing energy-balanced connected dominating set in wireless sensor networks," *Int. J. Sensor Networks*, vol. 2, no. 1/2, pp. 68–76, 2007.

[33] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic, "A dominating sets and target radius based localized activity scheduling and minimum energy broadcast protocol for ad hoc and sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 6, pp. 536–547, 2006.

[34] W. Wu, H. Du, X. Jia, Y. Li, and S. C.-H. Huang, "Minimum connected dominating sets and maximal independent sets in unit disk graphs," *Theoretical Comput. Sci.*, vol. 352, no. 1, pp. 1–7, 2006.

[35] C. Adjih, P. Jacquet, and L. Viennot, "Computing connected dominated sets with multipoint relays," *Ad Hoc Sensor Networks*, vol. 1, no. 1, pp. 27–39, 2005.

[36] J. Cartigny, F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic, "Localized LMST and RNG based minimum-energy broadcast protocols in ad hoc networks," *Ad Hoc Networks*, vol. 3, no. 1, pp. 1–16, 2004.

[37] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proc. 2002 IEEE Aerosp. Conf.*, Big Sky, MT, 2002, pp. 1123–1130.

[38] A. Manjeshwar and D. P. Agrawal, "TEEN: A routing protocol for enhanced efficiency in wireless sensor network," in *Proc. IEEE 15th Int. Parallel Distrib. Process. Symp.*, San Francisco, CA, 2001, pp. 2009–2015.

[39] O. Younis and S. Fahmy, "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 4, pp. 366–379, 2004.

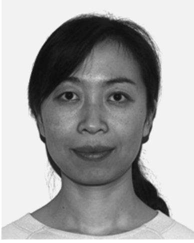
[40] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probability*, Berkeley, CA, 1967, pp. 281–297.

[41] M. Richards, M. Ghanem, M. Osmond, Y. Guo, and J. Hassard, "Grid-based analysis of air pollution data," *Ecological Modelling*, vol. 194, no. 1–3, pp. 274–286, 2006.

[42] Y. Ma, M. Richards, M. Ghanem, Y. Guo, and J. Hassard, "Air pollution monitoring and mining based on sensor grid in London," *Sensors*, vol. 8, pp. 3601–3623, 2008.

[43] M. Ding, X. Cheng, and G. Xue, "Aggregation tree construction in sensor networks," in *Proc. IEEE VTC'03*, Orlando, FL, Oct. 2003, pp. 2168–2172.

[44] R. Akl and U. Sawant, "Grid-based coordinated routing in wireless sensor networks," in *Proc. 4th IEEE Consumer Commun. Networking Conf.*, Las Vegas, NV, 2007, pp. 860–864.

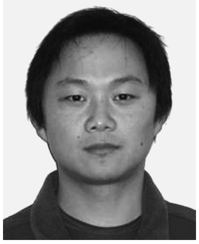


Yajie Ma received the B.Sc. degree in automatic control engineering from Wuhan University of Science and Technology, Wuhan, China, in 1996, and the M.Sc. degree in circuit and system in 2000 and the Ph.D. degree in communication and information engineering in 2005 from the Huazhong University of Science and Technology, Wuhan, China.

She worked in the Department of Computing, Imperial College of London, from 2006 to 2009, as a Research Associate. Now, she is an Associate Professor at the College of Information Science and Engineering, Wuhan University of Science and Technology. Her research focuses on wireless sensor networks, topology optimization, and data mining.

Yike Guo received the B.Sc. degree in computer science from TsingHua University, Beijing, China, and the Ph.D. degree in computational logic declarative programming from Imperial College London, London, U.K.

He is currently a Professor of Computing Science at the Department of Computing, Imperial College London. His research is in the area of parallel applications and network computing including parallel data mining algorithms, distributed systems, decision support systems, and grid computing.



Xiangchuan Tian received the B.Sc. degree in computing science from Huazhong University of Science and Technology, China, in 1998 and the M.Sc. degree in electronic and engineering from the University of Liverpool, Liverpool, U.K., in 2004.

Now, he works as a Research Assistant at the Imperial College London. His research interests are in distributed data mining, cloud computing, and workflow systems for e-Science applications.

Moustafa Ghanem received the M.Sc. and Ph.D. degree in high-performance computing from Imperial College London, London, U.K.

He is now a Research Fellow at the Department of Computing, Imperial College London. His current research interests are in large-scale informatics applications, including large-scale data and text mining applications and infrastructures, grid and cloud computing and workflow systems for e-Science applications.