

# Cluster-based Routing Algorithms Using Spatial Data Correlation for Wireless Sensor Networks

Chongqing Zhang

Department of Network Engineering  
Shandong University of Science and Technology  
Qingdao, China  
gongyouzhang@sina.com

**Abstract**—In densely deployed wireless sensor networks, spatial data correlations are introduced by the observations of multiple spatially proximal sensor nodes on a same phenomenon or event. These correlations bring significant potential advantages for the development of efficient strategies for reducing energy consumption. In this paper, spatial data correlations are exploited to design cluster-based routing algorithms of high data aggregation efficiency. We define the problem of selecting the set of cluster heads as the weighted connected dominating set problem. Then we develop a set of centralized approximation algorithms to select the cluster heads. Simulation results demonstrate the effectiveness and efficiency of the designed algorithms.

**Index Terms**—Wireless Sensor Networks (WSNs), Routing, Clustering, Dominating Set, Ant-colony Optimization

## I. INTRODUCTION

Recent advances in wireless communications and embedded computing have enabled the creation of wireless sensor networks. Due to the features of reliability, accuracy, flexibility, cost-effectiveness and ease of deployment, wireless sensor networks are promising to be used in a wide range of applications, such as environmental monitoring, target tracking, etc [1]. As a consequence, wireless sensor networks are receiving more and more attention from researchers. In wireless sensor networks, sensor nodes are usually powered by batteries that cannot be replaced in most cases. As a result, the energy constraint has significant effect on the network design and makes energy efficiency a major design challenge [1].

Environmental monitoring is a kind of typical applications of wireless sensor networks. In such kind of applications of wireless sensor networks, it is generally needed to deploy the sensor nodes densely in order to achieve satisfactory coverage [2]. Due to the high density of sensor nodes, spatially proximal sensor observations are highly correlated. The core operation of an environmental monitoring wireless sensor network is to collect and process data at the network nodes, and then transmit the necessary data to the base station for further analysis and processing. The correlations of sensor data can be exploited to develop efficient approaches for reducing energy consumption in data collecting.

Routing is an essential problem of wireless sensor networks. Due to the inherent constraints such as power, memory, and CPU processing capabilities of sensor nodes, routing in sensor networks is a challenging issue. Many routing protocols have been developed to make wireless sensor networks practical and efficient. These protocols can typically be classified into two types: (1) flat routing protocol, and (2) hierarchical routing protocol. In a wireless sensor network in which data correlation exists, the sensor nodes can perform data aggregation to avoid duplicated data transfers. [3]

By grouping sensor nodes into different clusters, clustering allows hierarchical structures to be built on the nodes and thus can improve the scalability of multi-hop wireless sensor networks [4]. Typically, a clustering algorithm divides the network into subsets of nodes, called clusters, each with one node serving as the cluster head. After the formation of clusters, sensor nodes transmit their data to the cluster heads for data aggregation, and then the aggregated data are further transmitted to the sink. Clustering provides an architectural framework for exploiting spatial data correlations to reduce energy consumption [4]. By selecting the cluster heads efficiently, hierarchical routing protocols can be developed to reduce the usage of consumption power and maximize the life time of the networks.

Various clustering algorithms have been proposed to organize sensor nodes in a wireless sensor network into clusters. In view of the energy constraint of such kind of network, many of these algorithms or protocols [5, 6, 7, 8, 9, 10, 11] have considered the issue of energy consumption or network lifetime. However, without considering spatial data correlations when organizing nodes into clusters, the cluster structures generated by these algorithms cannot provide effective support for the aggregation and compression to exploiting spatial data correlations to reduce energy consumption. Motivated by this, this paper tries to investigate how to integrate spatial data correlations with clustering algorithms so as to bring forth network structures which are able to gain high energy-efficiency by exploiting spatial data correlations to reduce energy consumption.

In this paper, we focus on designing routing algorithms to conserve energy by exploiting existing spatial data correlations which typically exist in sensor

networks in which sensor nodes are densely deployed. The targeted applications are monitoring applications that need to monitor a phenomenon over a geographic region covered by the sensor network. Such a sensor network is generally composed of two types of nodes: common sensor nodes and data sinks [12]. The data sink periodically gathers data values measured by common sensor nodes. By exploiting the spatial data correlations in the sensor data, our proposed algorithms select a small subset of sensor nodes which are called cluster heads. During data gathering, common sensor nodes first send their data to the cluster heads. Data compression is done by cluster heads, and then the compressed data is relayed to the sink. These cluster heads form a connected correlation-dominating set which means the resulting communication graph is connected. In this paper, based on defining the problem of selecting such a set of cluster heads as the weighted connected dominating set problem, we design several centralized and distributed algorithms for computing a weighted connected dominating set for a sensor network.

In addition, we propose another approach to tackle the clustering problem in a different way. The approach has two steps. The first step selects a set of cluster heads that form a dominating set. This dominating set needs not to be connected. In the second step, a set of nodes are selected and added to the above dominating set to make the resulting union set a connected set. This problem is modeled as a Steiner Tree problem. Then the ant-colony algorithm is adopted to solve the problem. The effectiveness of the above described approaches is verified by extensive simulations.

The rest of the paper is organized as follows: Section II formally defines the problem as a weighted connected dominating set problem. Section III presents the designed centralized and distributed algorithms, respectively. Section IV provides the numerical results to demonstrate the effectiveness of our clustering scheme through simulations. Finally, Section V gives concluding remarks and directions for future work.

## II. PROBLEM STATEMENT

### A. Problem Definition

This paper addresses the optimization problem that arises in wireless sensor networks with spatial data correlations. Given a wireless sensor network, select a set of cluster-heads  $C$  that satisfy (a) the selected cluster heads in  $C$  form a connected communication graph, (b) each sensor node that is not in  $C$  is a direct neighbor of a cluster head in  $C$ , and (c) the summation of the spatial data correlation degrees of all clusters is as large as possible. The first requirement for connectivity in the communication graph is due to the fact that the selected sensor set needs to collectively relay data to the sink. The second requirement makes sure that a node is either a cluster head or a direct neighbor of a cluster head. The third requirement is used to guarantee the resulting network structure is capable of make good use of the spatial data correlations to reduce energy consumption.

This problem can be defined formally as a weighted connected dominating set problem in the following text.

**Definition 1.** (Network Graph) Given a sensor network consisting of a set of sensor nodes  $V$ , the topology of the sensor network can be modeled as an undirected graph  $G(V, E)$  with  $V$  as the set of vertices and an edge  $(u, v)$  is included in the edges set  $E$  if two nodes,  $u$  and  $v$ , can communicate directly with each other. The network subgraph induced by a subset  $M$  of set  $V$  is the subgraph of  $G$  involving only the vertices and nodes in  $M$ .

**Definition 2.** (Correlation Degree) Let  $S$  be a subset of set  $V$  and  $s_1, s_2, \dots, s_n$  be the members of set  $S$ . Let  $X^t(S) = \{x_{i,t}^t, i = 1, 2, \dots, n\}$  be the vector of samples at time instant  $t$  returned by the  $n$  nodes in set  $S$ . The correlation degree  $D_t(S)$  among nodes in set  $S$  at time instant  $t$  is the spatial data correlation of  $X^t(S)$ . Let  $S$  be a cluster, then we can define the correlation degree of cluster  $S$  similarly.

### B. Computation of Correlation Degree

Given a sensor network consisting of a set of sensor nodes, we assume the spatial data correlation degree between two nodes is proportional to the distance between the two nodes. A model frequently encountered in practice is the Gaussian random field [13]. This model has the nice property that the dependence in data at different nodes is fully expressed by the covariance matrix, which makes it more suitable for analysis. Thus, we assume a *jointly Gaussian model* for the data measured at nodes, with an  $N$ -dimensional multivariate normal distribution  $G_N(\mu, K)$ :

$$f(X) = \frac{1}{\sqrt{2\pi} \det(K)^{1/2}} e^{-\frac{1}{2}(X-\mu)^T K^{-1}(X-\mu)} \quad (1)$$

where  $K$  is the covariance matrix (positive definite), and  $\mu$  is the mean vector. The diagonal elements of  $K$  are the variances  $k_{ii} = \sigma_i^2$ . The rest of  $K_{ij}$  depend on the distance between the corresponding nodes (e.g.  $K_{ij} = \sigma^2 \exp(-ad_{i,j}^2)$ ). Then, for any index combination  $I = \{i_1, \dots, i_k\} \in \{1, \dots, N\}, k \leq N, \mathbf{W} = (X_{i_1}, \dots, X_{i_k})$  is  $k$ -dimensional normal distributed. Its covariance matrix is the submatrix  $K[I]$  selected from  $K$ , with rows and columns corresponding to  $\{i_1, \dots, i_k\}$ .

Without loss of generality, we use differential entropy instead of entropy, since we assume that data at all nodes is quantized with the same quantization step, and differential entropy differs from entropy by a constant for uniformly quantized variables. The entropy of a  $k$ -dimensional multivariate normal distribution  $G_k(\mu, K)$  is:

$$h(G_k(\mu, K)) = \frac{1}{2} \log(2\pi e)^k \det K \quad (2)$$

Base on equation (2), we can compute the spatial data correlation degree of a set composed of a node and its close neighbors. After having computed the spatial data correlation degree, the clustering algorithms can pick out the cluster heads and group the nodes into different clusters. We will introduce the algorithms in the next section.

As for two correlated random data source  $X_i$  and  $X_j$ , let  $H(X_i)$  and  $H(X_j)$  be the entropies of  $X_i$  and  $X_j$ . Then  $X_i$  and  $X_j$  can code their data using  $H(X_i)$  and  $H(X_j)$  as their coding rate. If they can communicate with each other, they can jointly code their data using a coding rate  $H(X_i, X_j)$ . In [9], it has been proved that even  $X_i$  and  $X_j$  cannot communicate with each other, they can still jointly code their data using a coding rate  $H(X_i, X_j)$ . The prerequisite is their coding rates are equal to their conditional entropies  $H(X_i|X_j)$  and  $H(X_j|X_i)$ .

Above conclusion can be extended to multidimensional conditions. As for a data source set  $X = (X_0, X_1, X_2, \dots, X_n)$ , if the sources know the correlation structure, then the sources can use a joint coding rate  $H(X_0, X_1, X_2, \dots, X_n)$  to code their data even they do not communicate with each other. Assume the sources in set  $X$  are arranged according to their distances to  $X_0$ , then the coding rates are assigned as follows [14]:

$$\begin{aligned} R_0^* &= H(X_0) \\ R_1^* &= H(X_1 | X_0) \\ R_2^* &= H(X_2 | X_1, X_0) \\ &\dots \dots \dots \\ R_n^* &= H(X_n | X_{n-1}, X_{n-2}, \dots, X_1, X_0) \end{aligned} \quad (3)$$

For a cluster, let  $X_0$  be the cluster head and  $(X_1, X_2, \dots, X_n)$  are the members that are listed according to their distances to  $X_0$ . The coding rates can be assigned according to equation (3).

**C. Problem Definition**

After the formation of the clusters, the total communication cost incurred during the data gathering consists of two components,  $E_{total} = E_{clusters} + E_{tosink}$ , where  $E_{clusters}$  means the communication cost consumed by each member of a cluster while sending the data to the cluster-head, and  $E_{tosink}$  means the communication cost incurred by the cluster-heads while sending the data to the sink. When organizing sensor nodes into clusters, smaller number of clusters is preferred. This is because a smaller number not only means higher data compression rate, but also means the communication cost  $E_{tosink}$  is lower.

Let a set of sensor nodes  $C = \{c_1, c_2, \dots, c_m\}$  be the set of selected cluster heads. Our goal is to select the set  $C$  that holds the following conditions:

- 1) Minimize  $E_{total} = E_{clusters} + E_{tosink}$ .
- 2) For each sensor node  $u \in V$ , either  $u \in C$  or  $u \in V - C$  holds. For each node  $u \in V - C$ , there exists a cluster head  $v \in C$  and there exists an edge  $(u, v) \in E$ .
- 3) The communication subgraph induced by  $C$  is connected.
- 4) The sink is a member of the cluster heads set  $C$ .
- 5) The residual energy of a cluster head is above certain threshold.

To minimize the energy cost of data gathering, we need to do two sides of work. On one hand, we need to reduce  $E_{clusters}$ . On the other hand, we need to reduce

$E_{tosink}$ . Sometimes, the reduction of  $E_{clusters}$  maybe means the rise of  $E_{tosink}$ . Thus, a uniform energy cost is needed to unify  $E_{clusters}$  and  $E_{tosink}$ . In this paper, the average energy cost for collecting the data generated by one node in one period is adopted to meet above purpose.

As for a cluster  $C = \{c_0, c_1, c_2, \dots, c_m\}$  with  $c_0$  serves as the cluster head. Let  $T(c_i)$  be the energy consumed by node  $c_i$  for sending the data produced in one period by  $c_i$  to  $c_0$ .  $T(c_i)$  can be computed by following expression:

$$T(c_i) = 2E_{elec} + R_i^* (d(c_i, c_0)^\gamma \cdot E_t + E_r) \quad (4)$$

where  $E_{elec}$  is the energy cost by  $c_i$  to start the communication module,  $R_i^*$  is the coding rate of  $c_i$ ,  $\theta$  is the data gathering frequency,  $d(c_i, c_0)^\gamma \cdot E_t$  represents the energy consumed by  $c_i$  for sending a data unit,  $d(c_i, c_0)$  is the distance between  $c_i$  and  $c_0$ ,  $\gamma$  is the path loss exponent,  $E_t$  is the energy for sending a data unit under the condition that the communication distance is the standard reference distance, and  $E_r$  stands for the energy consumed by  $c_0$  for receiving a data unit.

After cluster head  $c_0$  receives the data from the cluster members, it packs the data into packets with size  $\lambda$  and forwards them to the base station. Let  $P(c_0)$  be the energy cost by  $c_0$  for sending a packet to the base station, then the energy consumed for collecting the data produced by cluster  $C$  can be expressed as:

$$Cost = \sum_{i=1}^m T(c_i) + \frac{\theta}{\lambda} \sum_{i=1}^m R_i^* \cdot P(c_0) \quad (5)$$

where  $\sum_{i=1}^m T(c_i)$  stands for  $E_{clusters}$ , and  $\frac{\theta}{\lambda} \sum_{i=1}^m R_i^* \cdot P(c_0)$  stands for  $E_{tosink}$ .

Therefore, the average energy cost for collecting the data generated by one node in one period can be expressed as:

$$Cost = \frac{1}{m} \sum_{i=1}^m T(c_i) + \frac{\theta}{m\lambda} \sum_{i=1}^m R_i^* \cdot P(c_0) \quad (6)$$

To minimize  $E_{tosink}$ , we need to minimize  $\sum_{i=1}^m R_i^*$ . In

another word, the data correlation degree of the cluster should be maximized. Notice that  $P(c_0)$  can only be computed after the completion of clustering operation.  $P(c_0)$  can be computed in an approximate way as follows:

$$P(c_0) = \frac{d(c_0, BS)}{d^*} (E_{elec} + \lambda(d^*)^\gamma \cdot E_t) + (\frac{d(c_0, BS)}{d^*} - 1)(E_{elec} + E_r) \quad (7)$$

where  $d^*$  stands for the optimum hop distance from  $c_0$  to the base station. Let  $x$  be the average hop distance from  $c_0$  to the base station, then the energy consumed by  $c_0$  for sending a packet to the base station can be expressed as:

$$f(x) = \frac{d(c_0, BS)}{x} (E_{elec} + \lambda x^\gamma E_t) + (\frac{d(c_0, BS)}{x} - 1)(E_{elec} + \lambda E_r) \quad (8)$$

By differentiating (8), we have

$$f'(x) = -\frac{d(c_0, BS) \cdot (2E_{elec} + E_r)}{x^2} + \frac{d(c_0, BS) \cdot \lambda x^{(\gamma-2)} E_t}{\gamma-1} \quad (9)$$

According to the condition of extremum, let expression (9) be equal to 0, then we get an equation. Solve the equation, then  $d^*$  can be computed.

$$d^* = \sqrt[\gamma]{\frac{(\gamma-1)(2E_{elec} + \lambda E_r)}{\lambda E_t}} \quad (10)$$

The above-discussed clustering problem can be modeled as a weighted connected dominating set problem [15]. A dominating set of a graph  $G = (V, E)$  is a node subset  $S \subseteq V$ , such that every node  $u \in V$  is either in  $S$  or adjacent to a node of  $S$ . A node of  $S$  is said to dominate itself and all adjacent vertices. If the nodes in a graph  $G$  are assigned with different weights, then graph  $G$  is called a weighted graph. A weighted connected dominating set is a connected dominating set in a weighted graph. Our clustering problem can be modeled as finding a weighted connected dominating set in which the total summation of the weights of the cluster heads is maximum. Unfortunately, this problem is a problem of NPC hardness [16].

### III. CLUSTERING ALGORITHMS

A set of approximation algorithms are proposed to select a weighted connected dominating set for a sensor network. The algorithms are based on the two algorithms proposed in [17] by Guha and Khuller.

To solve the connected dominating set problem discussed above, there are two strategies:

- 1) In the process of computing the dominating set, the connectivity of the graph induced from the interim dominating set is ensured by constraining the candidate nodes from which the dominating nodes are selected. This strategy can guarantee the connectivity of the graph induced by the final dominating set.
- 2) In the process of computing the dominating set, the candidate nodes from which the dominating nodes are selected are not constrained, thus the connectivity of the graph induced from the interim dominating set is not ensured. In order to guarantee the connectivity of the graph induced by the final dominating set, additional nodes may be needed to be added into the final dominating set.

The two strategies presented above are denoted as Strategy I and Strategy II in this paper. In the following two subsections, we first present three algorithms based on Strategy I, then we present one algorithm based on Strategy II. These algorithms are all centralized algorithms. The base station with high computing ability and sufficient power supply is the suitable computing device to run these algorithms. To run these algorithms, the base station needs to collect the position and residual energy information of the sensor nodes. After the computation, the results are broadcast to the sensor nodes.

#### A. Algorithms Based on Strategy I

The three centralized algorithms proposed in this subsection are sequential algorithms, which means that they run in cycles and a node is chosen as a cluster head in a cycle. The repetition continues until every node in the network is either a cluster head or a direct neighbor of a cluster head. Given a sensor network, each node is associated with a color, *white*, *gray*, or *black*. All nodes are initially *white* and change color as the algorithm progresses. The algorithm is essentially an iteration of the process of choosing a *white* or *gray* node to dye *black*. When a node is dyed *black*, any neighboring *white* nodes are changed to *gray*. At the end of the algorithm, the black nodes constitute a weighted connected dominating set. The meanings of three colors that a node  $u$  can have are explained as follows:

- *white* – node  $u$  is not in the dominating set, and  $u$  is not dominated by a node colored as *black*.
- *gray* – node  $u$  is not in the dominating set, but  $u$  is dominated by at least one node colored as *black*.
- *black* – node  $u$  is in the dominating set and acts as a cluster head.

We use the term *piece* to refer to a particular substructure of the network that is in the running process of an algorithm. A *white piece* is simply regarded as a *white piece*, and a *black piece* contains a maximal set of black vertices whose weakly induced subgraph is connected plus any gray vertices that are adjacent to at least one of the black vertices of the piece. Fig. 1 illustrates the definitions. The pieces are indicated by the dotted regions. Vertices  $a, b$ , and  $c$  are three white pieces. The other vertices are divided into 4 black pieces, which are marked as  $A, B, C$  and  $D$  in Fig. 1.

In each iteration cycle of the algorithm, a single *white* or *gray* node is chosen by the algorithm to dye black. The selection of the chosen node is based on following criteria:

- *residual energy*. If a node's residual energy is lower than a certain threshold  $E_{threshold}$ , then this node is deprived of the candidacy of becoming a cluster head.
- *compression rate*,  $D_u / N_u$ , where  $D_u$  is the spatial data correlation degree of the node set composed of node  $u$  and its close white neighbors, and  $N_u$  is the number of cluster members if  $u$  is selected as a cluster head. This value indicates to what degree node  $u$  compresses the data sent to it if node  $u$  is selected as a cluster head and acts as the compressor node. If multiple nodes tie on this item, then next

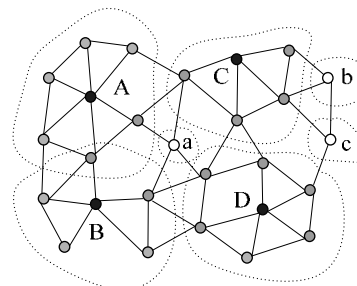


Figure 1. Illustration of Pieces

item is used to break the tie.

- *improvement*. The *improvement* of a node  $u$  is defined as the number of distinct pieces within the closed neighborhood of  $u$ . That is, the improvement of  $u$  is the number of pieces that would be merged into a single black piece if  $u$  were to be dyed black. If there are still multiple nodes tying on this item, then the residual energy is used to break the tie.

The *compression rate* of node  $u$  indicates to what degree that node  $u$  compresses the data sent from its neighbors. The *improvement* of node  $u$  indicates the number of pieces that will be reduced if node  $u$  was selected as a cluster head. The iteration goes on until there is only one piece left. In an iteration cycle, only nodes having more energy than  $E_{threshold}$  are candidates of cluster heads. The *compression rate* is then compared. The node with the highest compression rate is selected as the best node. Tie is broken by using the *improvement*. . If there are still multiple nodes tie on *improvement*, then a node with the most *residual energy* is chosen to be the target node.

The main differences between three algorithms lie in the number of black pieces and the candidate nodes set from which the target node is chosen. In Algorithm I, only one black piece is used and the best node is only chosen from all the gray nodes. This can insure the connectivity of the resulting dominating set. Like Algorithm I, only one black piece is used in Algorithm II too. Yet in Algorithm II, all gray nodes and white nodes that are one-hop to the black piece are candidates. If a white node  $u$  is selected as the target node, one gray node that connects  $u$  to a black node is also colored black. This can also insure the connectivity of the resulting dominating set. Different from Algorithm I and Algorithm II, multiple black pieces are used in Algorithm III. And in every iteration cycle, the best node is chosen from all gray and white nodes. The iteration cycle continues until two conditions are satisfied: 1) every node is either a member of the final dominating set or is dominated by another node; 2) the induced graph of the final dominating set is a connected graph.

Fig. 2 presents the pseudo code of Algorithm I. Two lists,  $L_1$  and  $L_2$ , are used to record nodes in the running process of the algorithm.  $L_1$  is used to record the nodes that are not included in the clusters, that is, nodes that are

not colored as black or gray. On the other hand,  $L_2$  is used to record the nodes that are already included in the clusters, that is, the nodes in the black piece. At the beginning, all nodes are included in  $L_1$  and  $L_2$  is empty. Then the algorithm begins its iteration process. In each iteration cycle of the algorithm, the spatial data correlation degrees are first calculated. Then a single white or gray node is chosen by the algorithm to dye black. When a white or gray node is dyed black, all white nodes adjacent to it are colored gray. When a node is dyed black, it is placed into the black piece along with all of its newly gray neighbors. The best candidate is chosen based on the comparison discussed above. The best candidate and all the nodes just colored gray are deleted from  $L_1$  and added to  $L_2$ . After all nodes are dyed black or gray, the algorithm judge if the sink is colored black or not. If the sink is colored gray, then the algorithm adds the sink to the weighted connected dominating set by coloring the sink black.

The processes of Algorithm II and Algorithm III are similar to the process of Algorithm I, so the pseudo code of Algorithm II and Algorithm III are not presented.

### B. Algorithm Based on Strategy II

The above-discussed algorithms can guarantee the final dominating set is a connected dominating set. However, the strategies used in these algorithms limit the selection range of nodes in every step, and this may make the clusters generated by these algorithms do not have the maximum correlation degrees.

The approach proposed in this subsection tackles the clustering problem in a different way. The approach has two steps. The first step is similar to the algorithms discussed above. This step selects a set of cluster heads that form a dominating set. The difference lies in the candidate nodes set from which the optimum node is selected. In this algorithm, the candidate nodes set includes all the gray and white nodes. Consequently, the dominating set selected here needs not to be a connected dominating set.

In the second step, a set of nodes are selected and added to the above dominating set to make the resulting union set a connected set. For a sensor network  $G = (V, E)$  and the cluster heads set  $S = \{s_1, s_2, \dots, s_n\}$ , some nodes of  $(V - S)$  can be selected and added to  $S$  to make the resulting set a connected tree with minimum communication cost. This problem can be modeled as the minimum Steiner Tree problem [18]. The minimum Steiner Tree of  $G(V, E)$  is a sub-graph  $G'(V_s, E_s)$  without cycles. The cost function of  $G'$  is defined as the summation of the costs of edges in  $E_s$ , that is:

$$W(S) = \sum_{(u,v) \in E_s} w(d^2(u,v) - d^{*2}) \quad (11)$$

Minimum Steiner Tree problem is a problem of NP-complete hardness. Ant-colony algorithm is adopted to solve the problem. Multiple ants cooperate in solving the minimum Steiner Tree problem. The Ant-colony algorithm is shown in Fig. 3.

```

Pseudocode of Algorithm I:
1.  add all nodes to  $L_1$ ;
2.  clear  $L_2$ ;
3.  while ( $L_1$  is not empty) {
4.      calculate the correlation degree of the nodes;
5.      choose the best candidate  $\Rightarrow u$ ;
6.      dye  $u$  black;
7.      delete  $u$  from  $L_1$ ;
8.      add  $u$  to  $L_2$ ;
9.      select node adjacent to  $u$  whose color are white  $\Rightarrow T$ ;
10.     color nodes in  $T$  gray;
11.     delete nodes in  $S$  from  $L_1$ ;
12.     add nodes in  $S$  to  $L_2$ ;
13. }
14. if (sink is gray)
15.     color sink black;

```

Figure 2. Pseudocode of Algorithm I

```

Pseudocode of the Ant Colony Algorithm
Input: WSN  $G = (V, E)$ , Clusterhead set  $C$ ;
Output: Steiner tree  $S$ ;
1. generate an ant for each node in  $C$  and add the ant to  $A$ ;
2. for (each edge  $(i, j)$  in  $E$ ) do
3.    $\tau_{ij} = \tau_0$ ;
4. for (each ant  $g$  in  $A$ ) do {
5.    $T^{(g)} = \{ \}$ ;
6.    $S^{(g)} = \{ \}$ ;
7. }
8. while ( $|A| > 1$ ) do {
9.   select an ant  $g$  in  $A$  at node  $i$ ;
10.  for (each edge  $(i, j)$  in  $E$ ) do
11.    compute  $P_{ij}$ ;
12.    determine next node  $j$  for ant  $g$ ;
13.     $T^{(g)} = T^{(g)} \cup \{j\}$ ;
14.     $S^{(g)} = S^{(g)} \cup \{(i, j)\}$ ;
15.    if ((exists  $g'$  in  $A$ ) and ( $j$  in  $T^{(g')}$ )) then {
16.       $T^{(g)} = T^{(g)} + T^{(g')}$ ;
17.       $S^{(g)} = S^{(g)} + S^{(g')}$ ;
18.       $A = A - \{g\}$ ;
19.    }
20.    if (location of  $g' = j$ ) then {
21.      compute  $X = \{x \in T^{(g')} \mid (\text{exists } x' \text{ not in } T^{(g')}, ((x, x') \text{ in } E))\}$ ;
22.      choose a node from  $X$  randomly and place  $g'$  in that node;
23.    }
24.  }
25.   $S = \text{Union}_{g \in A} S^{(g)}$ ;
26.  for (each edge  $(i, j)$  in  $E$ ) do
27.    update  $\tau_{ij}$ ;
28. }
    
```

Figure 2. Pseudocode of ant colony optimization algorithm

#### IV. NUMRICAL RESULTS

In this section the simulation results are presented to demonstrate the performance and effectiveness of our proposed algorithms. The number of nodes of the used networks ranges from 100 to 1000. The nodes are deployed uniformly on planes of different sizes to ensure all networks have a same node density. The network area is a square that is divided into sub-squares. The data of nodes in a same sub-square are strongly correlated, yet the data of nodes in different sub-squares are weakly correlated. Each sub-square follows Gaussian random field model.

The above-discussed four algorithms are compared in the experiments. Among them, three algorithms are based on strategy I. These algorithms are denoted as Algorithm I, II, III respectively. Another algorithm is the algorithm based on strategy II. This algorithm is denoted as Algorithm IV.

The networks work in a periodical way. In one period, a node produces 32 bytes data and codes them with the coding rate shown in expression (3), and then the data is sent to the cluster head. If a node and its cluster head lie in different sub-squares, the node sends the data directly to the cluster head without any processing. After a cluster head receives all the data from its cluster members, it reorganizes the data into packets with the size of 32 bytes and then forwards these packets to the base station.

Three energy costs are compared in the experiments. These costs are the energy cost consumed inside clusters, the energy costs consumed outside the clusters, and the total cost. These three costs are denoted as in-cluster cost, CH-BS cost and total cost respectively. Figs. 4, 5 and 6 show the comparisons of the three costs. It can be observed from the figures that Algorithm IV achieves the highest energy efficiency.

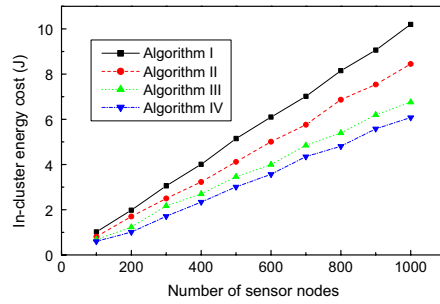


Figure 4. Comparison of in-cluster cost

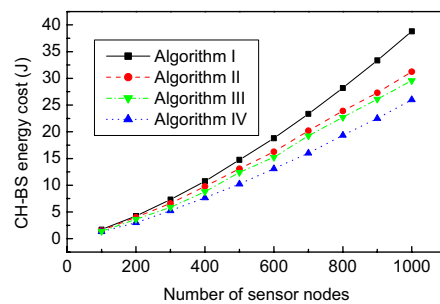


Figure 5. Comparison of CH-BS cost

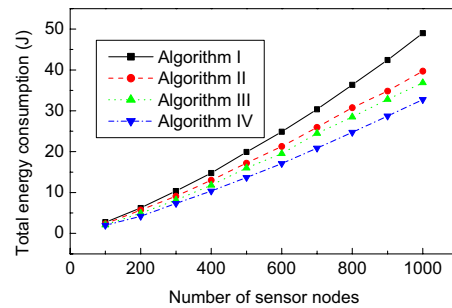


Figure 6. Comparison of total cost

#### V. CONCLUSION AND FUTURE WORK

In this paper, we have investigated how to exploit spatial data correlations in sensor data to develop efficient cluster-based routing algorithms for reducing energy consumption. We presented several centralized and distributed clustering algorithms that group sensor nodes into clusters of high data aggregation efficiency. These algorithms select the set of cluster heads for a sensor network by constructing a weighted connected dominating set. We showed the proposed weighted connected dominating set-based clustering approach an effective way for reducing energy consumption in collecting sensor data. As the future work, we plan to adopt different models to capture the spatial data correlations in our clustering approach. We also plan to exploit temporal and spatial data correlations jointly to refine our clustering algorithms.

## REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci. "A survey on sensor networks," *Computer Networks*, vol. 38, no. 4, pp. 393-422, March 2002.
- [2] M. C. Vuran, O. B. Akan, and I. F. Akyildiz, "Spatio-Temporal Correlation: Theory and Applications for Wireless Sensor Networks," *Computer Networks*, vol. 45, no 3, pp. 245-259, June 2004.
- [3] J. Ibric and I. Mahgoub, "Cluster-Based Routing in Wireless Sensor Networks: Issues and Challenges," in *Proceedings of the 2004 Symposium on Performance Evaluation of Computer Telecommunication Systems*, 759-766, July 2004.
- [4] Feng Zhao, Leonidas Guibas, "Wireless Sensor Networks: An Information Processing Approach," Boston: Elsevier-Morgan Kaufmann; 2004.
- [5] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad-hoc Sensor Networks". *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366-379, Oct-Dec 2004.
- [6] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-Configuring Sensor Networks Topologies," *Proc. IEEE INFOCOM*, Jun. 2002.
- [7] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor network," *IEEE Transaction on Wireless Communications*, vol. 1, pp. 660-670, Oct 2002.
- [8] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," *Proc. IEEE INFOCOM*, Apr. 2003.
- [9] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang, "PEAS: A robust energy conserving protocol for long-lived sensor networks," *Proc. IEEE ICDCS*, May. 2003.
- [10] V. Kawadia and P.R. Kumar, "Power Control and Clustering in Ad Hoc Networks," *Proc. IEEE INFOCOM*, Apr. 2003.
- [11] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *ACM Wireless Networks*, vol. 8, no. 5, Sept. 2002.
- [12] H. Gupta, V. Navda, S. R. Das, V. Chowdhary, "Efficient gathering of correlated data in sensor networks," *Proc. ACM MOBIHOC*, May. 2005.
- [13] M. Lotfinezhad and B. Liang, "Effect of partially correlated data on clustering in wireless sensor networks," *Proc. IEEE SECON*, Oct. 2004.
- [14] R. Cristescu, B. Beferull-Lozano, M. Vetterli. "Network Correlated Data Gathering," *Twenty-Third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2004, 2571-2582.
- [15] J. E. Dunbar, J. W. Grossman, J. H. Hattingh, S. T. Hedetniemi, and A. A. McRae, "On weakly-connected domination in graphs," *Discrete Math.*, 167/168:261-269, 1997.
- [16] M. L. Garey and D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," W. H. Freeman, San Francisco, 1979.
- [17] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, Vol. 20, No. 4, pp. 374-387, Apr. 1998.
- [18] Dingzhu Du and Xiaodong Hu, *Steiner Tree Problems In Computer Communication Networks*, World Scientific, 2008.



**Chongqing Zhang** received his Ph.D from the Department of Computer Science and Engineering at Shanghai Jiaotong University in 2007. He joined Shandong University of Science and Technology as an Assistant Professor in Fall 2007. His primary research interests include wireless networks and wireless sensor networks.