

## Research Article

# An Efficient Clustering Algorithm in Wireless Sensor Networks Using Cooperative Communication

Shukui Zhang,<sup>1,2</sup> Jianxi Fan,<sup>1</sup> Juncheng Jia,<sup>1</sup> and Jin Wang<sup>1</sup>

<sup>1</sup> School of Computer Science and Technology, Soochow University, Suzhou 215006, China

<sup>2</sup> State Key Lab. for Novel Software Technology, Nanjing University, Nanjin 210093, China

Correspondence should be addressed to Shukui Zhang, zhangsk2000@163.com

Received 7 January 2012; Accepted 27 February 2012

Academic Editor: Hongli Xu

Copyright © 2012 Shukui Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Processing the gathered information efficiently is a key functionality for wireless sensor networks. In generally, the sensor networks often use in-network data aggregation and clustering to optimize network communication. The set of aggregating nodes forms a dominating set of the network graph. Finding the weakly connected dominating set (WCDS) is a promising approach for clustering the WSN. However, finding a minimum WCDS is NP-hard problem for most graphs, and a host of approximation algorithm has been proposed. The aim of the paper is to construct a minimum WCDS as a clustering scheme for WSN. Our clustering schemes construction algorithm includes two phases. First of all, we construct a maximal data aggregation tree (DAT) of the network. The second phase of the algorithm is to choose the nodes (called connectors) to make the WCDS connected. The correctness and performance of our algorithms are confirmed through theoretical analysis and comprehensive simulations.

## 1. Introduction

A wireless sensor network (WSN) is a multihop wireless communication network. In WSN, each node assumes the role of a router and relays the packets toward the final destinations if a source cannot directly send the packets to a final destination due to the limitation of the radio transmission range. In addition, the energy efficiency is one of the major constraints in WSN. The network topology may also change unpredictably due to node failure, running out of power, or adding new nodes into the network. Most topology changes are localized within a small area of the network. Therefore, it is desirable to abstract the network structure as local changes which need not be seen by the entire network. This is done by using logical substructures called clusters. It is believed that clustering can dramatically improve a network's broadband utilization and delivery ratio, extend network lifetime, and reduce packet retransmission [1]. A natural method for forming clusters is based on the idea of graph domination [2]. The most basic clustering methods that have been studied in ad hoc networks and WSN are based on the dominating sets (DSs). Moreover, among various existing

clustering schemes, dominating set-based clustering [3, 4] is a promising approach.

The main advantage of dominating set-based clustering is that it simplifies the clustering process to the one in a smaller subnetwork generated from the connected dominating set (CDS). The efficiency of this approach depends largely on the process of finding and maintaining a CDS and the size of the corresponding sub-network. In addition, the CDS formation algorithm should be localized (i.e., based on local information) for low overhead and fast convergence. The research that works on selecting a minimum CDS has never been interrupted because of its dramatic contributions to wireless networks. Unfortunately, finding a minimum CDS is NP complete for most graphs, even if global information is available and no constraint [5].

In addition, in wireless channels, packets are usually dropped when the channel goes into deep fade and thus an outage occurs. In particular, the outage happens when instantaneous channel capacity falls below the amount of information carried in the packet [6]. Recently, the cooperative communication technique was exploited to study energy management issues for ad hoc and sensor networks [7, 8].

Such as in [7], a network model using cooperative communication is developed to deal with broadcasting in ad hoc networks and WSN. Transmitting independent copies of a packet generates diversity and combats the effects of fading. The selected relay  $r_k$  cooperates with one another if the direct transmission fails to the final destinations  $D$ . Each relay will decide whether it can successfully decode the  $M$  sources' information based on its local channel information. The criterion used for successful decoding is that its local channel information can satisfy the condition

$$\sum_{m \in S} R_m \leq \log \left( 1 + \rho \sum_{m \in S} |h_{mr_k}|^2 \right), \quad \forall S \subseteq \{1, 2, \dots, M\}, \quad (1)$$

where  $\rho$  is the SNR [9].  $h_{mr_k}$  is the coefficient of the channels between  $m$ th source and  $k$ th relay. The above expression characterized the capacity region multiple access channels [10]. Assume that the  $Q$  relays  $R$  can satisfy the criterion and hence be able to decode the  $M$  sources' information correctly. By ordering the relay-destination channels, we denote the  $Q$ -qualified relays as  $R_1, R_2, \dots, R_Q$ , where  $|h_{R_n D}|^2 \geq |h_{R_{n+1} D}|^2$ . The study has shown that cooperative communication can potentially combine the following advantages: (1) the power saving provided by multihopping, (2) the spatial diversity provided by the antennas of separate mobile nodes, and (3) node cooperation can also lead to increased data rates [11, 12].

Motivated by cooperative communication in ad hoc networks and WSN, Alzoubi et al. proposed an algorithm for weakly connected dominating set (WCDS) based on a spanning tree [4]. In this scheme, a maximal independent set (MIS) is elected such that each node in the MIS can be connected to the spanning tree via an extra node. Chen and Liestman [13, 14] proposed a zonal algorithm, in which the graph is divided into regions, a WCDS is constructed for each region, and adjustments are made along the borders of the regions to produce a WCDS for the whole graph. Their algorithm for the partitioning phase is partly based on a minimum spanning tree (MST) algorithm of Gallager et al. [15]. Han and Jia [16] also proposed an area-based distributed algorithm for WCDS construction in ad hoc networks with constant approximation ratio, linear time, and message complexity. While it has a lower message complexity than the zonal algorithm proposed by Chen and Liestman, it outperforms the mentioned algorithm. Basagni et al. [17] presented a performance comparison of the protocols proposed for clustering and backbone formation in large scale ad hoc network. Wu [3] presented two distributed algorithms for finding a WCDS in ad hoc networks. The first algorithm was implemented by first electing a leader among the nodes, which was going to be the root of a spanning tree. The spanning tree is then traversed, and the dominator nodes are selected. But the distributed leader election is extremely expensive in practice and exhibits a very low degree of parallelism. The second algorithm first constructs a maximum independent set (MIS) by an iterative labeling strategy and then modifies the MIS by selecting

one intermediate node between each pair of dominators separated by exactly three hops.

At present, the study of WCDS is not more. As mentioned different above, we consider the WCDS as a better method for clustering [4] an ad hoc network and WSN. In this paper, based on the characteristics of communication under the cooperative communication, we extend the dominative capability of nodes in the corresponding network, and we turn the clustering scheme construction problem of a cooperative network into the WCDS problem in the graph model of cooperative communication. A novel algorithm (called DAT-WCDS) to find WCDS for clustering in ad hoc networks and WSN is proposed. And their good performance is confirmed by simulations.

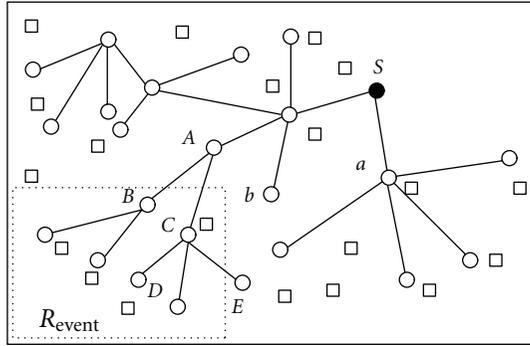
## 2. Preliminaries and Definitions

*2.1. A Network Environment.* In this paper, the aim of the proposed algorithm is to form a clustering scheme for the WSN by finding a connected dominating set problem. We consider a monitor area  $A$  with  $N$  wireless sensors, represented by the set  $S = (s_1, s_2, \dots, s_N)$  randomly deployed. Each sensor node is equipped to learn its location coordinates such as its location information  $(x_i, y_i)$  [18]. It is not the purpose of this paper to define mechanisms to find this location. Without loss of generality, let us assume that nodes in the set  $S$  belong to two dimensional planes as illustrated in Figure 1.

At first, the goal of the proposed algorithm is to construct the data aggregation tree (DAT) in this  $N$  nodes network, where DAT is consisted of  $N_t$  nodes called tree node, which is used to receive and aggregate data, the other  $(N - N_t)$  nodes are referred to as non-tree (NT) nodes. Each NT node senses its environmental parameter and reports it to its nearest tree node. The DAT is well spread over the entire WSN so that  $N_t$  tree nodes are uniformly distributed on the network. In this way, it ensures that the attribute readings sent by NT nodes to the corresponding tree node incur a smaller hop count. For simplicity, we use  $R_{\text{event}}$  (denoted by the dashed rectangle in Figure 1) to represent an event, and the event region is denoted by the area  $P_{\text{event}}$ , where  $R_{\text{event}} \subseteq R$ . Normally all the events are assumed to have already been sensed in the network by DAT.  $R'$  is defined as the portion of  $R$  not occupied by any event, that is,  $R' = R - R_{\text{event}}$ .

*2.2. Connected Dominating Set.* For simplicity, we assume a simple and yet general enough model that is widely used in the community. Wireless sensor networks are modeled as unit disk graphs  $G = (V, E)$ . Where, the vertices in  $V$  represent the communication nodes. Let  $V' \subseteq V$  be a subset of vertices in  $G = (V, E)$ . In the following, we use  $G[V']$  to denote the subgraph induced by  $V'$ . For a subgraph  $G'$  of  $G$ , we use  $V(G')$  and  $E(G')$  to refer to the vertices and edges of  $G'$ ; respectively, we denote by  $\Gamma(v)$  the closed neighborhood of a vertex  $v \in V$ , that is,

$$\Gamma(v) = \{u \in V \mid (u, v) \in E\} \cup \{v\}. \quad (2)$$



□ NT node      ○ Tree node  
 P\_event region

FIGURE 1: Network environment.

Analogously, for  $V' \subseteq V$ ,  $\Gamma(V') = \cup_{w \in V'} \Gamma(w)$  define the neighborhood of  $V'$ . In this context, we set  $\Gamma(\Phi) = \Phi$ , for  $k \in N$ , and we call  $\Gamma_k(v) = \Gamma(\Gamma_{k-1}(v))$  the recursively defined  $k$ th neighborhood of  $v \in V$ ,  $\Gamma_0(v) = \{v\}$ .

A normal transmission range  $r$ , using the Euclidean distance  $d(u, v)$ , denoting the number of hops on a shortest path in  $G$  between vertices  $u$  and  $v$ , where  $d(u, v)$  is also viewed as the transmission cost between  $u$  and  $v$ . This means that two vertices are connected by an edge if and only if  $u$ 's disk covers  $v$  and  $v$ 's disk covers  $u$ . Let  $p(u, v) = \{u, w_1, w_2, \dots, w_k, v\}$  be a shortest path between node  $u$  and  $v$ .

In graph theory, a dominating set (DS) of a graph  $G = (V, E)$  is a subset  $S \subseteq V$ , such that every vertex  $v \in V$  is either in  $S$  or adjacent to a vertex of  $S$ . A minimum DS (MDS) is a DS with the minimum cardinality  $\gamma(G)$ . A subset  $I \subseteq V$  is called independent if for every two vertices  $u, v \in I$ , there does not exist an edge  $(u, v) \in E$ . An independent set is called maximal if it cannot be extended by the addition of any other vertices from the graph. There is an important relationship between maximal independent sets and dominating sets in a graph; an independent set is also a dominating set if and only if it is a maximal independent set [4].

A CDS  $S$  of a given graph  $G$  is a dominating set whose induced subgraph, denoted  $G[S]$ , is connected, and a minimum CDS (MCDS) is a CDS with the minimum cardinality. A dominating set  $S$  is a weakly connected dominating set (WCDS) of a graph  $G$ , if the graph  $G[S] = (\Gamma(S), E \cap (\Gamma(S) \times S))$  is a connected subgraph of  $G$ . In other words, the weakly induced sub graph  $G[S]$  contains the vertex of  $S$ , their neighbors, and all edges with at least one endpoint in  $S$ .

Finding the minimum WCDS of the network graph is one of the most investigated methods for cluster formation in which a dominator node assumes the role of a cluster head, and its one-hop neighbors and 2-hop neighbors are assumed to be cluster members. The structure of the network graph can be simplified using WCDS and made more succinct for transmitting in ad hoc networks and WSN [15, 16].

In this paper, we focus on clustering mechanisms to elect a minimum and sufficient number of links to serve as

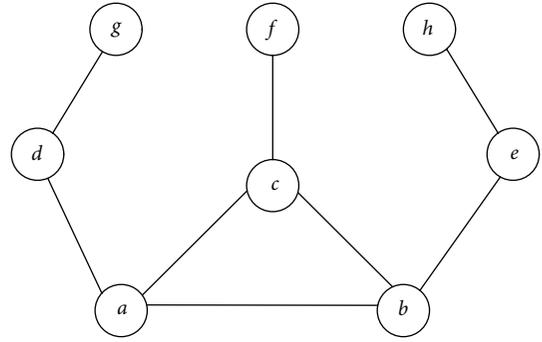


FIGURE 2:  $\{a, b\}$  is EDS.

the communication backbone of the network. Accordingly, the clustering approach to topology management can be modeled as the relevant minimum WCDS problem in graph theory.

**2.3. Dominating Set Extension.** In this subsection, we extend the dominative capabilities of nodes for finding a small WCDS for a WSN. Wu et al. proposed the notion of an extended dominating set (EDS) [12]. A subset  $S$  of  $V$  is an EDS if every node of  $V$  is (a) in the subset, (b) a regular neighbor of a node in  $S$ , or (c) 2-hop neighbor of  $k$  nodes in  $S$ .

Dominative capabilities extension of nodes: each node is extended such that it dominates not only itself and its 1-hop distance neighbors fully, but also its 2-hop distance neighbors partly. For example, in Figure 2, the node dominates not only itself and nodes  $d, c, b$  fully, but also nodes  $g, e, f$  partly. This extension extends the dominative capability of a node from its 1-hop neighbors to its 2-hop distance neighbors.

In [12], they used a notion of contribution; each forward node contributes 1 to all its 1-hop neighbors, and  $1/k$  to all its 2-hop neighbors. The effective contribution of  $u$  to  $v$  is  $u$ 's contribution to  $v$  before the signal energy of  $v$  reaches 1. The initial signal energy of each node is 0. A node is said to have the maximum effective contribution if it has the maximum total effective contribution to its neighbors and 2-hop neighbors. If we consider the contribution of each forward node as its dominative capability to all its neighbors, thus each forward node can fully dominate its 1-hop neighbors, and partly dominate its 2-hop neighbors. The following definitions will be used throughout the paper.

**Definition 1.** For any vertex  $u \in V$ ,  $v \in \Gamma_2(u)$ ,  $v$  is a dominator neighbor of  $u$  if  $v$  is a cluster-head (or dominator).

**Definition 2.** For a vertex  $v$ , the 2-hop-independent neighbors of  $v$  are  $P_2 \subseteq \{\Gamma_2(v) - \Gamma(v)\}$ , such that if  $v_1, v_2 \in P_2$ , then  $v_1$  and  $v_2$  are independent.

**Definition 3.** Let vertex  $w$  be called as connector if it is common neighbor between dominators  $u$  and  $v$ , where  $v$  is the 2-hop neighbor of  $u$ .

2-hop WCDS is also a CDS. It requires that, for any two nodes with distance equal to 2, there exists at least one shortest path between them, whose intermediate node should be included in 2-hop WCDS. The formal definition is shown in details as follows.

*Definition 4.* The 2-hop shortest path weakly connected dominating set problem (2-hop WCDS) is to find a minimum-size node set  $S \subseteq V$  such that

- (1)  $\forall u \in V \setminus S, \exists v \in S$  such that  $(u, v) \in E$ ,
- (2) the induced graph  $G[S]$  is connected, and
- (3)  $\forall u, v \in V$ , if  $d(u, v) > 2$ , then  $\exists p_i(u, v) \in p(u, v)$ ,  $p_i(u, v) \setminus \{u, v\} \subseteq S$ .

We do not consider the situation of  $d(u, v) = 1$ . The reason is that our WCDS aims to reduce transmission cost. When we select a WCDS, neighbors of  $\forall v \in V$  must be known to  $v$  during selecting process. As a result, when  $v$  has a packet destined to  $u$ ,  $v$  will not inform adjacent nodes in WCDS to help deliver the packet, because  $v$  knows that  $u$  can receive packets from  $v$  directly and no consecutive forwarding will happen. However, once  $d(u, v) > 1$ , consecutive forwardings are needed to deliver packages to the destination node. Thus, a good selection of forwarding nodes will influence on network performance greatly. We hope to select a CDS with minimum size, but keep the value of  $d(u, v), \forall u, v \in V$  through this CDS the same as that in original graph. It is the goal of WCDS. We redefine a node's degree in details as follows.

*Definition 5.* The degree of a node  $u$  is denoted by  $d(u)$ . Define the rank of node  $u$  to be an ordered pair  $(d_u, id_u)$ , where  $d_u$  is the node degree and  $id_u$  is the node ID of  $u$ . We say that a node  $u$  with rank  $(d_u, id_u)$  has a higher order than a node  $v$  with rank  $(d_v, id_v)$  if  $d_u > d_v$ , or  $d_u = d_v$  and  $id_u < id_v$ .

*Definition 6.* The "diameter"  $X$  of a set of nodes  $S$  in a graph  $G$  is the maximum of the pairwise shortest paths between these nodes  $X = \max_{i, j \in S} d(i, j)$ , where  $d(i, j)$  is the shortest number of hops needed to go from node  $i$  to node  $j$  in  $G$ .

When WCDS is constructed, only nodes in WCDS may forward data. In broadcasting [16], nodes in WCDS can help spread data to the whole network. In routing, data will be sent to WCDS and be delivered via nodes in WCDS. Thus, how to construct a WCDS is closely related to the performance of WCDS-based broadcasting and routing. Our approach to establishing a minimal WCDS is based on two phases that implement the data aggregation tree (DAT) and WCDS elections, respectively. We discuss the construction of WCDS in the following sections.

### 3. Algorithm Description

The aim of the proposed algorithm is to construct a minimum WCDS as a clustering scheme for WSN. We employ a CDS in this paper since it can behave as the virtual backbone of a sensor network. Our clustering schemes construction

algorithm includes two phases: DAT construction and then to select connectors to make the MIS nodes connected into a WCDS construction. In the first phase, we construct a maximal DAT of the network. The second phase of the algorithm is to choose the nodes (called connectors) to make the WCDS connected.

*3.1. Construction of Data Aggregation Tree.* We assume that each node knows the node ID and degree of all its 1-hop neighbors and 2-hop neighbors, this can be achieved through requiring each node to broadcast its node ID initially. After each node knows all its neighbors, it can broadcast its degree, one more round of "Hello" message is needed to construct 2-hop information.

Let the target region be  $A$ , and sensor node set in the region be

$$S = \{s_i(x_i, y_i) \mid s_i \in A\}, \quad (3)$$

where  $(x_i, y_i)$  is the position coordinate of the node  $s_i$ , external of the target region is set  $K = \{k_i(x_i, y_i) \mid k_i \notin A\}$ , and DAT is definite as  $T$  has

$$T(s_i - > K) = \bigcup_i \text{path}(s_i - > k_i), \quad k_i \in K, \quad (4)$$

where  $\text{path}(s_i - > k_i)$  is the greatest span path from node  $s_i$  to node  $k_i$  in graph  $G$ , and its length is diameter  $l$ . In this path, the minimum distance between each node is bigger or equals to the minimum distance  $d(s_i, k_i)$  in any other path from  $s_i$  to  $k_i$ , and the node number is the smallest in graph  $G$ .

Dynamic topology has a significant impact on DAT algorithms. Two actions of a node lead to network topology changes: withdrawing and joining. Withdrawing refers to the functional termination of a node in the network, and it happens when a node fails, runs out of power, or exits from the network. Joining refers to the functional start of a node in the network, and it happens when a new node is added, or a node recovers from a failure. Moving of a node can be treated as two separate actions of withdrawing and joining if the node can be assumed to stop receiving and transmitting messages when in motion. To cover a broader range of situations, this paper assumes no special notification sent from the withdrawing or joining node. Relying on such notification, even if possible, imposes high expectation on the ability of nodes. The neighbors of a changing node must rely on other mechanisms to detect the changes.

The changing neighborhood resulted from a node withdrawing or joining affects the generated DAT. Generally, there are two methods to handle it: recalculating and updating. With the recalculating method, a distributed DAT algorithm starts at a fixed interval or is triggered by some event (e.g., when disconnection of the dominating set is detected), and a new DAT is generated from scratch. With the updating method, the DAT is maintained by updating a portion of the existing dominating set according to the topology changes. A practical strategy may use the updating method most of the time and use the recalculating method

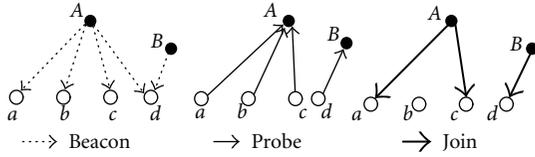


FIGURE 3: Exchange of message to construct the DAT.

when necessary. This paper only discusses the updating method.

Let depth of the tree  $T$  be  $p$ . Algorithm 1 constructs a DAT with given depth. When a node chooses its two children, it will choose the two biggest span nodes, ensuring that the tree  $T$  covers more target regions as far as possible. In the process of the multiple regressions, it can achieve the high accuracy. After the DAT is formed, in each subdomain all residual nodes send data to the nearest tree node away from themselves. In this paper, we used the literature [19] design method to construct the aggregation tree. That is, constructing process through three kinds of messages: Beacon, Probe, and Join. Figure 3 describes the process about the exchange of different signals to construct the tree. For more details, see literature [19].

After given data aggregation tree (DAT), a data communication operation consists of (possibly repeated) two phases: a propagation phase where the query demands are pushed down into the sensor network along the tree, and an transmission phase where the aggregated values are propagated up from the children to their parents.

**3.2. Clustering Formation.** In this section, a data aggregation tree-based algorithm (called DAT-WCDS) is proposed for clustering formation in WSN, which focuses on finding a WCDS problem in the network graph. In the algorithm, a special dominating set using a MIS of the network is constructed, and then a CDS is constructed to connect dominators and the other nodes.

Given  $T$  be a DAT and  $D$  is a dominating set of  $T$  containing. It suffices to determine an independent set  $J$  of vertices which is disjoint from  $D$  and contains a neighbor of every vertex in  $D$ , because a maximal independent set  $I$  which contains  $J$  but is disjoint from  $D$  is clearly a dominating set of  $T$ . A simple strategy to select the elements of  $J$  is to root  $T$  in some vertex  $x$  in  $D$  and to select a child of every vertex in  $D$  which itself is not contained in  $D$ .

If this strategy succeeds, then the selected vertices will clearly form an independent set. Nevertheless, this strategy fails in the presence of vertices  $u$  in  $D$  all children of which are also in  $D$ . For such a vertex, we have to choose its parent. Working out the consequences of this reasoning leads to Algorithm 1 in the following sections.

We will hope that there are some dominator(s) and some dominee(s) in maximal independent set of each layer of DAT. Here a connector node  $x$  (a dominee of a dominator  $u$ ) is said to be redundant for the dominator  $u$  if removing  $x$  will not disconnect any of the 2-hop dominators of  $u$  from  $u$ . For every dominee, it has at least one-dominator

neighbor in the same or upper level. Thus, every dominator (except the root) has at least one dominator in the upper level within 2 hops. Using this property, we can ensure that all the data in the dominators can reach the root finally if every dominator transmits its data to some dominator in upper level within two hops. From another point of view, considering dominators in the decreasing order of their levels, a dominator  $u$  in level  $L$  aggregates data from all dominators in level  $L + 1$  or  $L + 2$  that are within two hops of  $u$  leads to Algorithm 2 in the following sections.

In Algorithm 2, we only concentrate on communications between dominators. Since dominators cannot communicate directly, we have to rely on some dominates (NT node), each of which acts as a bridge between two dominators. The algorithm runs from lower level to upper level in DAT, every dominator will remain silent until the level where it locates begins running. When it is its turn, the dominator will try to gather all the data from other dominators in lower levels that have not been aggregated. If a dominator's data have been collected before, then it is unnecessary to be collected again. After the end of the second phase, the algorithm has identified MIS and the connectors. Iteratively, the dominator nodes are picked which connects independent set nodes in different components. The following phases are performed to establish and form clusters

Initially, the sink creates an empty cluster associated with an unclustered node of  $S$ . Each sensor  $\{s_1, s_2, \dots, s_N\}$  transmits its position  $(x_i, y_i)$  to the sink. To accomplish this step any efficient sensor routing algorithm can be used. Thus, the clustering algorithm is not bound to how the sink receives this information. If there is an unconnected node in the network, it cannot announce itself and thus will not be considered in the algorithm. Then, the sink finds the qualified unclustered nodes for joining to that first member. When no more nodes can be added to the cluster, the sink takes a new unclustered node and begins a new cluster. Then, each first member sends a packet to the members of his cluster notifying them about the cluster which they belong to. Each node is in one of the four states: unmarked, clustered-head (CH), cluster member (CM), and half-dominated. In the following, we describe the algorithm in detail.

Algorithm 3 is executed by the sink once upon deployment, and thus all nodes will become clustered. If a node joins to the network, it has to send its position  $(x_i, y_i)$  to the sink for announcing itself as a new node. The sink computes the highest rank of the new node and finds the first cluster that can accept it as a new member. Then, the sink sends a message to the first member in order that this node reorganizes the cluster with the new member. On the other hand, each node periodically sends a Hello message to the first member notifying that it is alive.

When a node dies, the first member will notify the rest of the members about the new cluster set and will reconfigure any parameter related to the cluster. The first member also periodically notifies to its cluster members about its availability. If a first-member dies, the cluster members will notify to the sink their availability to belong to another cluster or to create a new cluster. Note that the beacons among cluster

**Input:** a data aggregation tree  $T$  and threshold  $K, P$   
 /\*  $W$  be vertex set in tree  $T$ ,  $P$  be depth of the tree  $T$ ,  $K$  be stage number \*/  
**Output:** A maximal independent dominating set  
 Let  $D$  denote the dominating set constructed at stage  $K$  and be initially set to null  
 (1) begin  
 (2)  $D \leftarrow \Phi$   
 (3) While  $W \neq \Phi$   
 (4) Choose a vertex  $i \in W$   
 (5)  $D = D \cup \{i\}$   
 (6)  $W = W \setminus \Gamma(i)$   
 (7) End while  
 (8) Choose a vertex  $x \in D$  of degree  $d(x) = \min\{d(u) \mid u \in D\}$ ;  
 (9)  $J \leftarrow \Phi$  /\*  $J$  be an independent set \*/  
 (10) while  $\exists u \in D$  such that  $u \notin \Gamma(J)$  and all children of  $u$  inclusion in  $D \cap (\Gamma(J) \times J)$  do  
 (11) Let  $v$  be the parent of  $u$ ;  
 (12)  $J \leftarrow J \cup \{v\}$ ;  
 (13) partner( $u$ )  $\leftarrow v$ ;  
 (14) end  
 (15) while  $\exists u \in D$  such that  $u \notin \Gamma(J)$  do  
 (16) Choose a child  $v$  of  $u$  such that  $v \notin D \cup \Gamma(J)$ ;  
 (17)  $J \leftarrow J \cup \{v\}$ ;  
 (18) end  
 (19) Let  $I$  be a maximal independent set of  $T$  with  $J \subseteq I$  and  $D \cap I = \Phi$ ;  
 (20) Increment stage number  $k$   
 (21) Until depth of the tree  $T$  is greater than  $P$  or the stage number  $k$  is threshold  $K$   
 (22) end

ALGORITHM 1

**Input:** The DAT tree with root  $v_0$  and depth  $P$ , data  $d_i$  stored at each tree node  $v_i$ .  
 (1) Let  $T$  be the final data aggregation tree.  
 (2) Initially all independent set nodes form different components, each node in  $I$  broadcasts dominatees message so that dominatees can know of adjacent independent set nodes in different components.  
 (3) for  $i = P - 1, P - 2, \dots, 0$  do  
 (4) while a dominatee node  $v$  exists having  $i$ -adjacent independent nodes of  $I$  in different components do  
 (5) Choose all dominators, denoted as  $B_i$ , in level  $i$  of  $T$  tree.  
 (6) For every dominator  $u_i \in B_i$  do  
 (7) Node  $u_i$  broadcasts itself as the dominator.  
 (8) Node  $u_i$  finds the set  $\Gamma_2(u_i)$  of unmarked dominators that are within 2-hops of  $u$  in  $T$ , and in lower level  $i + 1$  or  $i + 2$ , mark all nodes in  $\Gamma_2(u_i)$ .  
 (9) Dominatees  $w_i$  on receiving this message keep a count of neighbouring dominators at level  $i + 1$  or  $i + 2$  and broadcasts the final count.  
 (10) Each level  $i + 1$  or  $i + 2$  dominators on receiving the counts from the potential connectors, select among them the node with highest bank as its connector and informs it.  
 (11) Node  $w_i$ , then becomes a connector;  $B_i \leftarrow B_i \cup \{w_i\}$ .  
 (12) Every node  $w$  in  $\Gamma_2(u_i)$  sends aggregated data to the parent node (a connector node) in  $T$ .  
 (13) Every node  $z$  that is a parent of some nodes in  $\Gamma_2(u_i)$  sends original data to node  $u_i$  (which is the parent of  $z$  in  $T$ ).  
 (14) End for  
 (15)  $i = i - 1$   
 (16) End for /\* The identified DAT nodes connect the dominator nodes. Thus, independent set nodes and DAT nodes forms the CDS of  $G$  \*/  
 (17) The root  $v_0$  sends the result to the sink using the shortest path.

ALGORITHM 2

Input: The DAT tree with has identified  $I$  and the connectors;  
 /\* Let  $V_p, V_q$  be the cluster,  $v \in V_p$  and  $v_p$  its cluster-head.  $H = I \cap \Gamma_2(v_p)$  be the set of 2-hop neighbors of  $v_p$  in  $I$ ;  $V_u$  be the cluster dominated by dominator node  $u$ ; \*/

- (1) begin
- (2) Initially, all nodes are unmarked.
- (3)  $u$  sends out CH message with the information of  $V_u$  to its neighbors,  $v \in V_u$ ,  $v$  sends its  $rank$  though CM message to dominator  $u$ .  $u$  receiving CM message from all its members computes its relative proximity as  $rank$ .  $u$  ranks each node  $v$  in  $V_u$  based on non-increasing ordering using proximity  $rank$ ;
- (4)  $u$  node with the highest  $rank$  among its unmarked 2-hop neighbors becomes a cluster-head and broadcasts CH messages to all its neighbors;
- (5) After receiving a CH message, for a node  $v$ ,  $v$  sends a message so that all available nodes in  $w \in \Gamma(u)$  become its dominates;
- (6) Each  $w$  node in turn sends another message to its 2-hop neighbors, making the available nodes as potential dominators;
- (7) For all  $V_h, v_h \in H$  and  $v_h$  is the cluster-head of  $V_h$  do
- (8)  $v$  becomes a cluster-member if it is a 1-hop neighbor of node  $u$ , and its current state is unmarked. If it is the first time that  $v$  receives a CH message,  $v$  will broadcast CM messages to all its neighbors;
- (9) If  $v$  is a 2-hop neighbor of node  $u$ , its current state is unmarked, and it is the first time that  $v$  receives a CH message,  $v$  becomes half-dominated;
- (10) If node  $v$  is a 2-hop neighbor of node  $u$  and its current state is half-dominated,  $v$  becomes a cluster-member if  $v$  receives a different CH message for the second time.  $V$  will broadcast CM messages to all its neighbors;
- (11) If node  $v$  is a 2-hop neighbor of node  $u$  and its current state is half-dominated,  $v$  becomes a cluster-head if  $v$  does not receive a different CH message again.  $v$  will broadcast CH messages to all its neighbors;
- (12) Dominator node  $v_p$  sends CH message to its neighboring dominator;  $v_q$  receiving CH message switched to become dominator in DAT tree and sends out CM message to the dominator node;
- (13) Connectors  $c_k \in V_q$  among its neighboring nodes are activated on receiving CM message and sends out CM message to its independent dominators;
- (14) Switching from  $v_p$  to  $v_q$  takes place through local messages;
- (15) The same procedure is repeated among the remaining nodes, until each node in DAT becomes either a cluster-head or cluster-member;
- (16) End for
- (17) End

ALGORITHM 3

members implies low overhead since cluster sizes have few nodes.

#### 4. Analysis of Algorithm

In the next subsections we first analyze the correctness of the algorithm and then analyze its complexity for running time and messages exchanged of the algorithm.

##### 4.1. Correctness of the Algorithm

**Theorem 7.** *The output of the proposed Algorithm 1 is a maximal independent set.*

*Proof.* By contradiction, we consider the first execution of the while-loop in line 11 for which the vertex  $u$  has no parent which does not belong to  $D$ ; that is, either  $u$  is the root  $x$  of  $T$  or the parent of  $u$  belongs to  $D$ .

Let  $D'$  denote the set of vertices  $u'$  from  $D$  which can be reached from  $u$  on a path  $P$  of the form

$$P : u_0 w_1 v_1 u_1 w_2 v_2 u_2 \dots \cdot w_l v_l u_l \quad (5)$$

with  $u_0 = u, u_l = u', l \in N, w_i \notin D$ , and partner  $(u_i) = v_i$  for  $1 \leq i \leq l$ . Note that  $w_1$  is a child of  $u$ . Let the set  $D''$

contain the parent of the parent of  $u'$ —the grandparent of  $u'$  for every vertex  $u'$  in  $D'$ . Let  $\check{D} = (D \setminus D' \cup \{u\}) \cup D''$ .

Let  $w''$  be a child of  $u$ . Clearly,  $w'' \notin J$ . If  $w'' \in D$ , then  $w'' \in \check{D}$ . If  $w'' \notin D$ , then  $w''$  has a child  $v''$  which belongs to  $J$ , and  $v''$  has a child  $u''$  which belongs to  $D$  such that partner  $(u'') = v''$ . Since  $uw''v''u''$  is a path as in (1), we obtain, by the definition of  $D'$ , that  $u'' \in D'$ . This implies that  $w'' \in D''$ , and hence  $w'' \in \check{D}$ . Therefore, in both cases,  $u, w'' \in \Gamma[\check{D}]$ , and all vertices which were dominated by  $u$  in  $D$  are still dominated by vertices in  $\check{D}$ .

Let  $u' \in D'$ . Let  $P$  be as in (1) with  $u' = u_l$ . Since  $w_l \in \check{D}$ , we have  $v_l \in \Gamma[\check{D}]$ . If  $w''$  is a child of  $u'$ , then exactly the same argument as above implies that  $w'' \in \check{D}$ . Hence again all vertices which were dominated by  $u''$  in  $D$  are still dominated by vertices in  $\check{D}$ .

Altogether, we obtain that  $\check{D}$  is a dominating set of  $T$  which contradicts the assumption that  $D$  is a minimum dominating set. By the claim, the while-loop in line 11 successfully adds to the set  $J$  the parents of vertices in  $D$  which do not belong to  $D$ . By the condition for the while-loop in line 11, just before the execution of the while-loop in line 16, the set  $J$  is independent, and every vertex  $u \in D$  with  $u \notin \Gamma(J)$  has at least one child which does not belong to  $D$

and is nonadjacent to the vertices in  $J$ . During the executions of the while-loop in line 16, only children of vertices in  $D$  are added to  $J$ , and this property is maintained throughout the remaining execution of Select. Hence, the while-loop in line 16 successfully adds to the set  $J$  the children of vertices in  $D$  which do not belong to  $D$  such that after the last execution of the while-loop in line 16, the set  $J$  is independent, disjoint from  $D$  and  $D \subseteq \Gamma(J)$ . By the above remarks, the set  $I$  defined in line 20 is an independent dominating set of  $T$  which completes the proof.  $\square$

**Theorem 8.** *After the above two phases, the constructed DS is a WCDS of the whole graph.*

*Proof.* After the first phase, Algorithm 1 constructs a DAT with the given depth. When a node chooses its two children, it will choose the two biggest span nodes, ensuring that the tree  $T$  covers more target regions as far as possible. It is possible that there exist two dominators that are apart by at least 2 hops in the graph. However, these dominators are apart by at most 3 hops. According to the definition of WCDS, we know that the IDS constructed in the second phase is a WCDS. Although the second phase reduces the size of dominators, the connectivity is not destroyed. Therefore, after the two phases, the constructed IDS is a WCDS of the whole graph.  $\square$

#### 4.2. Complexity Analysis

**Theorem 9.** *The algorithm DAT-WCDS has time complexity  $O(n)$  time and  $O(D)$  rounds, where  $D$  is the network diameter and message complexity of  $O(n \times d^2)$ , where  $d$  is max degree of node in  $G$ .*

*Proof.* Assume that in a given unit disk the size of an MIS is always less than maximum degree of a node in  $G$ ; therefore,  $|\text{MIS}| \leq d$ . Each node sends at most two messages to become dominatee and at most  $d$  messages per degree to update neighbor's information and  $d^2$  to get neighbors of the neighbor to become dominator. Thus, message complexity is  $O(n \times d^2)$ , where  $d$  is the maximum node degree.

While establishing the relationship between connectors and dominators, the message complexity is only size of CDS which is at most  $O(n)$ . Thus, in the message complexity of algorithm  $O(n \times d^2)$ , each node is explored one by one, so the time complexity  $O(n)$ . The number of synchronous rounds is  $O(D)$ , where  $D$  is network diameter, which is bounded by shortest distance of farthest node from a given leader.  $\square$

## 5. Simulation and Discussion

In simulations, all algorithms in discussion are implemented by using MATLAB, and all nodes are randomly deployed in a square area  $A$ . Every node uses a radio range  $r$  ( $r = 10, 60$  units). The network size and node density determine the number of nodes ( $N$ ) in the network. Node density  $p$  is defined as the average number of nodes per unit area. Relative node density is defined as the number of neighbors per node. For example, given node density  $p = 0.01$ ,

TABLE 1: Values of simulation parameters used.

Parameter	Value
Size of area $A$	$160 \times 160 \text{ m}^2$
Transmission range $r$	$r \in [10, 60] \text{ m}$
Number of nodes $N$	$N \in [20, 160]$
Density of network $p$	$p = A/N$
Event area $A_s$	$40 \times 40 \text{ m}^2$
Size of messages $m$	500 b

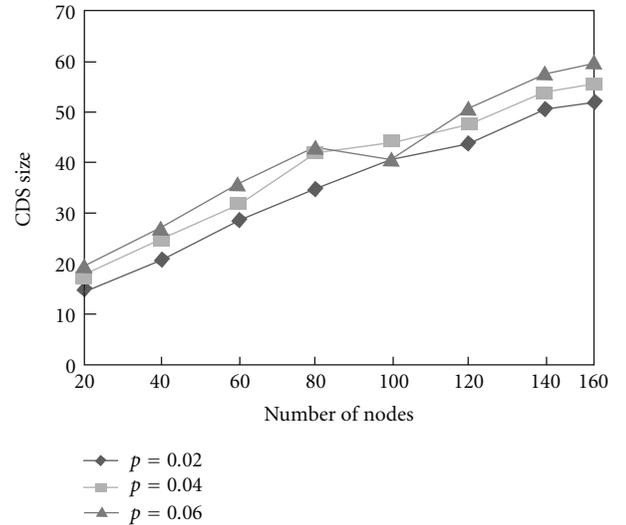


FIGURE 4: Impact of node density on CDS size.

$r = 10$ , the relative node density is  $\pi \times r^2 \times p = 3.14$ . Table 1 summarizes all the network configurations used in simulations.

**5.1. Node Density.** Node density determines how many neighbors a node can have. With a higher node density, a node has more neighbors to compete with to become a dominator. But after a node becomes a dominator, all its neighbors are covered as NT nodes. Usually, a node that can cover more neighbor nodes has a greater chance to become a dominator because of its greater degree. Thus, a new dominator will try to cover a new area of the network by given a connected network. Therefore, if the algorithm is well designed, the CDS size should be mainly determined by the network size and has less to do with the node density. Figures 4 and 5 show that DAT-WCDS generates CDS of almost the same size and the same diameter in networks with various node densities. But it takes longer time for the algorithm to converge in high-density networks (Figure 6).

**5.2. Size of WCDS.** Figure 4 shows the results when the node's transmission range is set as 30 units and the number of nodes in the networks ranges from 20 to 160. When the transmission range increases, as more nodes may be connected, the network becomes denser. In this case, the size of WCDS only increases slightly as the size of the network

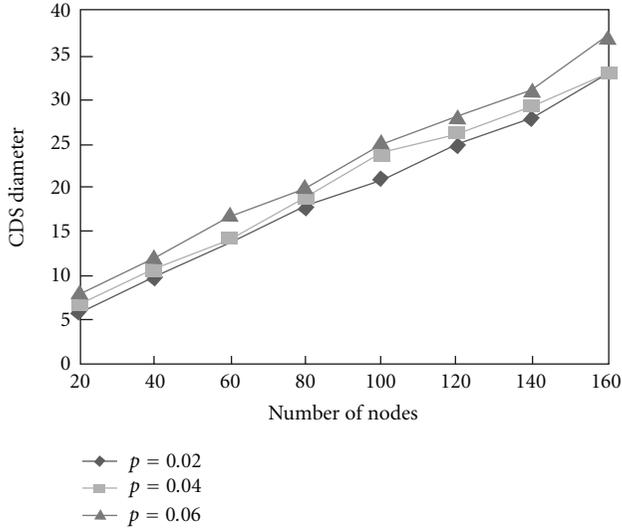


FIGURE 5: Impact of node density on CDS diameter.

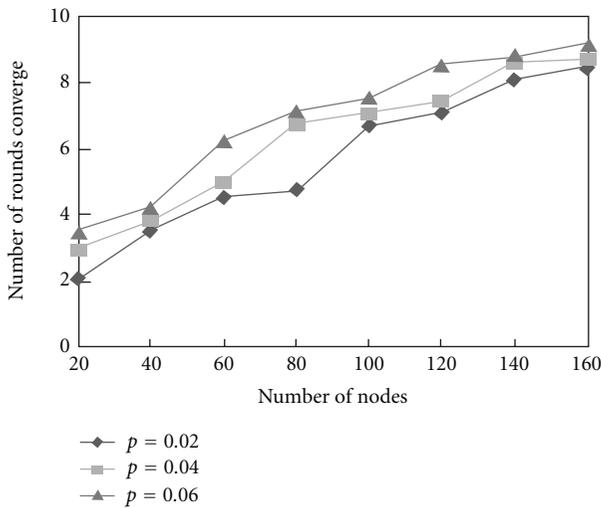


FIGURE 6: Impact of node density on convergence time.

increases. When the number of nodes in the network reaches 160, the number of nodes in the WCDS constructed by the DAT-WCDS algorithm is only about 31% of that constructed. The reason why our algorithm always outperforms is that for each pair of 2 hops is away cluster-heads adds one additional node to the WCDS, whereas our algorithm only “weakly connects” 2 hops away cluster-heads in different areas. We find that increasing the node’s transmission range can increase the coverage area of each node, and therefore, increasing the density of the network, which leads to a smaller size of the WCDS.

5.3. Comparison with Other Algorithms. The DAT-WCDS algorithm is compared with two multiple-phase CDS algorithms: ZS [20] and KM [21]. However, KM does not

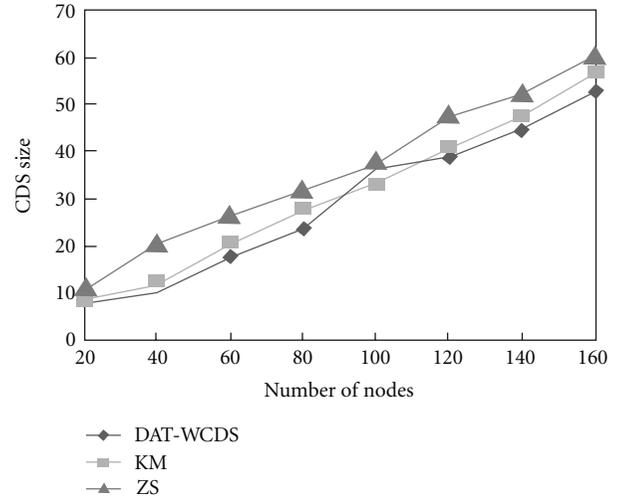


FIGURE 7: Comparison with other algorithms.

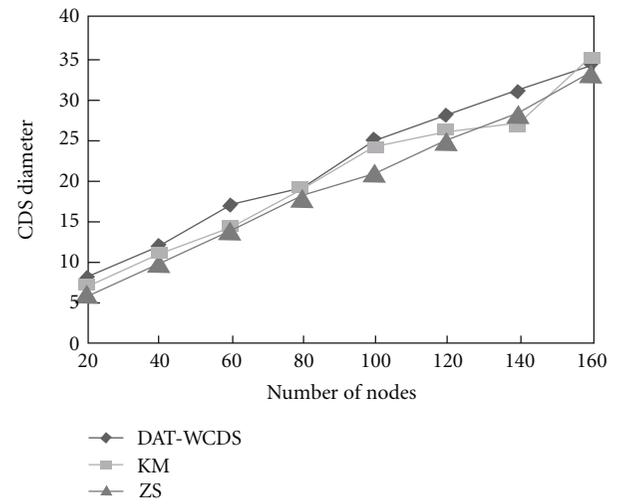


FIGURE 8: Comparison with other algorithms.

generate smallest size CDS, but it converges fast. Therefore, KM here serves as a good comparison candidate as we will show various aspects of algorithms at different performance levels.

Figure 7 shows that, in terms of CDS size, DAT-WCDS performs better than KM and ZS. The connected dominating sets built by KM have smaller diameters in large networks (Figure 8), but the tradeoff is much greater dominator population. DAT-WCDS converges much faster than ZS, as illustrated in Figure 9. The DAT-WCDS algorithm always converges in no more than 11 rounds for a wide range of network sizes in our simulations. Here, each round of ZS is the time for generating a new layer of dominators. The convergence time of ZS is mainly affected by the network size and the node radio range.

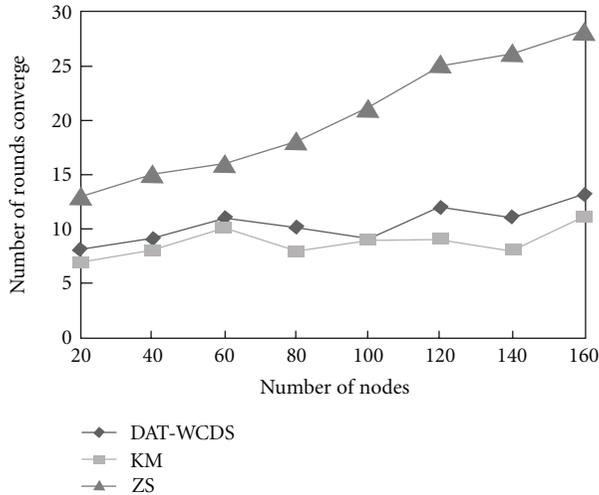


FIGURE 9: Comparison with other algorithms.

## 6. Conclusion

In this paper, we extend the dominative capabilities of nodes, and a data aggregation tree-based algorithm called DAT-WCDS is proposed for clustering formation in WSN, which focuses on finding a WCDS problem in the network graph. Our clustering schemes construction algorithm includes two phases: DAT is constructed and a special dominating set using a MIS of the network is constructed, then selecting connectors to make the MIS nodes connected into a WCDS construction. The correctness and performance of our algorithms are confirmed through theoretical analysis and comprehensive simulations.

## Acknowledgments

This work is supported by National Natural Science Foundation of China under Grants nos. 61070169 and 61170021, Natural Science Foundation of Jiangsu Province under Grant no. BK2011376 Specialized Research Foundation for the Doctoral Program of Higher Education of China no. 20103201110018, and Application Foundation Research of Suzhou of China no. SYG201118 and sponsored by Qing Lan Project.

## References

- [1] M. Agarwal, J. H. Cho, L. Gao, and J. Wu, "Energy efficient broadcast in wireless ad hoc networks with hitch-hiking," in *Proceeding of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 2096–2107, Hong Kong, March 2004.
- [2] R. Rajaraman, "Topology control and routing in ad hoc networks: a survey," *SIGACT News*, vol. 33, no. 2, pp. 60–73, 2002.
- [3] J. Wu, "Extended dominating-set-based routing in ad hoc wireless networks with unidirectional links," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 9, pp. 866–881, 2002.
- [4] K. M. Alzoubi, P.-J. Wan, and O. Frieder, "Message-optimal connected dominating sets in mobile ad hoc networks," in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc '02)*, pp. 157–164, Lausanne, Switzerland, June 2002.
- [5] K. M. Alzoubi, W. Peng-Jun, and O. Frieder, "Distributed heuristics for connected dominating sets in wireless ad hoc networks," *Journal of Communications and Networks*, vol. 4, no. 1, pp. 22–29, 2002.
- [6] L. Ding, W. Wu, J. Willson, H. Du, W. Lee, and D.-Z. Du, "Efficient algorithms for topology control problem with routing cost constraints in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1601–1609, 2011.
- [7] Y. Liang and V. V. Veeravalli, "Cooperative relay broadcast channels," *IEEE Transactions on Information Theory*, vol. 53, no. 3, pp. 900–928, 2007.
- [8] N. Jindal, U. Mitra, and A. Goldsmith, "Capacity of ad-hoc networks with node cooperation," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '04)*, pp. 271–272, July 2004.
- [9] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, 2004.
- [10] A. Nosratinia, T. E. Hunter, and A. Hedayat, "Cooperative communication in wireless networks," *IEEE Communications Magazine*, vol. 42, no. 10, pp. 74–80, 2004.
- [11] D. N. C. Tse, P. Viswanath, and L. Zheng, "Diversity-multiplexing tradeoff in multiple-access channels," *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 1859–1874, 2004.
- [12] J. Wu, M. Cardei, F. Dai, and S. Yang, "Extended dominating set and its applications in ad hoc networks using cooperative communication," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 8, pp. 851–864, 2006.
- [13] Y. P. Chen and A. L. Liestman, "Maintaining weakly-connected dominating sets for clustering ad hoc networks," *Ad Hoc Networks*, vol. 3, no. 5, pp. 629–642, 2005.
- [14] Y. P. Chen and A. L. Liestman, "A zonal algorithm for clustering ad hoc networks," *International Journal of Foundations of Computer Science*, vol. 14, no. 2, pp. 305–322, 2003.
- [15] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Transactions on Programming Languages and Systems*, vol. 5, no. 1, pp. 66–77, 1983.
- [16] B. Han and W. Jia, "Clustering wireless ad hoc networks with weakly connected dominating set," *Journal of Parallel and Distributed Computing*, vol. 67, no. 6, pp. 727–737, 2007.
- [17] S. Basagni, M. Mastrogiovanni, and C. Petrioli, "A performance comparison of protocols for clustering and backbone formation in large scale ad hoc networks," in *Proceeding of the IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MAHSS '04)*, pp. 70–79, October 2004.
- [18] K. K. Chintalapudi and R. Govindan, "Localized edge detection in sensor fields," in *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 59–70, May 2003.
- [19] S.-K. Zhang, Z.-M. Cui, S.-R. Gong, Q. Liu, and J.-X. Fan, "A data aggregation algorithm based on splay tree for wireless sensor networks," *Journal of Computers*, vol. 5, no. 4, pp. 492–499, 2010.

- [20] D. Zhou, M.-T. Sun, and T.-H. Lai, "A timer-based protocol for connected dominating set construction in IEEE 802.11 multihop mobile ad hoc networks," in *Proceeding of the 5th Symposium on Applications and the Internet (SAINT '05)*, pp. 2–8, February 2005.
- [21] K. M. Alzoubi, P.-J. Wan, and O. Frieder, "Maximal independent set, weakly connected dominating set, and induced spanners for mobile ad hoc networks," *International Journal of Foundations of Computer Science*, vol. 14, no. 2, pp. 287–303, 2003.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

