

# Extended Dominating Set and Its Applications in Ad Hoc Networks Using Cooperative Communication \*

Jie Wu<sup>†</sup>, Mihaela Cardei<sup>†</sup>, Fei Dai<sup>‡</sup>, and Shuhui Yang<sup>†</sup>

<sup>†</sup>Department of Computer Science and Engineering

Florida Atlantic University

Boca Raton, FL 33431

<sup>‡</sup>Department of Electrical and Computer Engineering

North Dakota State University

Fargo, ND 58105

## Abstract

We propose a notion of extended dominating set where each node in an ad hoc network is covered by either a dominating neighbor or several 2-hop dominating neighbors. This work is motivated by cooperative communication in ad hoc networks whereby transmitting independent copies of a packet generates diversity and combats the effects of fading. We first show the NP-completeness of the minimum extended dominating set problem. Then, several heuristic algorithms, global and local, for constructing a small extended dominating set are proposed. These are non-trivial extensions of the existing algorithms for the regular dominating set problem. The application of the extended dominating set in efficient broadcasting is also discussed. The performance analysis includes an analytical study in terms of approximation ratio and a simulation study of the average size of the extended dominating set derived from the proposed algorithms.

**Keywords:** *Ad hoc network, connectivity, cooperative communication, dominating set, simulation.*

---

\*This work was supported in part by NSF grants ANI 0073736, CCR 0329741, CNS 0422762, CNS 0434533, and EIA 0130806. Contact address: jie@cse.fau.edu

# 1 Introduction

The nature of ad hoc networks makes them different from wireless infrastructure networks which typically include base stations that are not battery constrained. Energy management strategies can conserve the energy of the battery powered nodes by taking advantage of the energy available at base stations. In contrast, an ad hoc network consists of small, battery powered devices only. The development of new energy management techniques is critical for practical deployment of these networks.

Dominating set (DS) has been widely used in the selection process of an active node set. A set is dominating if every node in the network is either in the set or a neighbor of a node in the set. When active nodes form a dominating set, all nodes in the network are also said to be reachable. When a DS is connected, it is denoted as a CDS; that is, any two nodes in the DS can be connected through intermediate nodes from the DS. CDS as a connected virtual backbone has been widely used for broadcast process [28], searching in a reduced space, and point coverage in sensor networks [7]. Because of the promiscuous receiving mode of wireless sensors, when each node in a CDS forwards the packet once, all nodes in the network will receive the packet. In Fig. 1 (a), the dominating node set  $\{u, v, w\}$  forms a virtual backbone for efficient broadcasting, since only dominating nodes need to forward the packet and all remaining nodes can receive the packet without having to forward it.

Power saving techniques for ad hoc networks can be classified into two categories: *power saving protocols* and *power control for transmission*. The former aims to put wireless nodes into periodical sleep state in order to reduce power consumption. Power control for transmission manages energy consumption by adjusting transmission ranges. It was experimentally confirmed by Feeney and Nilsson in [15] that the difference in energy consumption between an idle node and a transmitting node is not major, while a major difference exists between idle and sleep states of nodes. Specifically, it is shown in Span [8] that the ratio of energy for transmit, receive, idle, and sleep is 13:9:7:1. In this paper, we focus on power saving protocols in which a small set of active nodes is maintained at any given period, although such a set can change over time. Recently, the *cooperative communication* (CC) technique [24] was exploited to study energy management issues for ad hoc and sensor networks. Such as in [2], a network model using CC is developed to deal with broadcasting in ad hoc networks. In CC, transmitting independent copies of a packet generates diversity and combats the effects of fading. In this way,  $k$  copies of the same packet can potentially reach a receiver outside the normal transmission range with the same baseline transmit power.

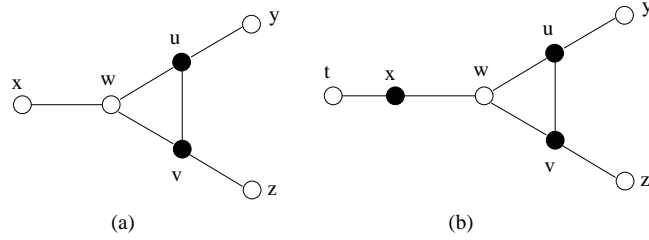


Figure 1: (a) A sample network:  $\{u, v, w\}$  forms a CDS and  $\{u, v\}$  forms an ECDS. (b) Another sample:  $\{x, u, v\}$  forms an EWCDS.

In this paper, we propose a notion of *extended dominating set* based on the cooperative communication. A DS is called an extended dominating set (EDS) if for every node in the network, it is in the set, it has a neighbor in the set, or it has  $k$  2-hop neighbors in the set. In Fig. 1 (a),  $u, v, w$  forms a CDS. If using CC, and  $k = 2$ , node  $x$  is covered twice by two 2-hop neighbors,  $u$  and  $v$ .  $w$  can be withdrawn and  $\{u, v\}$  forms an EDS. Since the set is connected, it is also called *extended connected dominating set* (ECDS). Later, we will define two notions of connectivity: strongly connected (ECDS by default) and weakly connected (EWCDS). The connectivity is defined in terms of the success of a broadcast process: a packet from a source in EDS will be received by all other nodes, given that each node in EDS forwards the packet once, after it receives the complete packet. In EWCDS, the broadcast will be successful for at least one source in EDS; whereas in ECDS the broadcast will be successful for any source in EDS. In Fig. 1 (b),  $\{u, v, x\}$  forms an EWCDS for  $k = 2$  since  $x$  can retrieve the complete packet when either  $u$  or  $v$  is the source, while neither  $u$  nor  $v$  can when  $x$  is the source. we focus on algorithms designed for *unit disk graphs*, which is the most popular model used in literature for wireless network analysis.

This paper focuses on some non-trivial extensions of various methods for ECDS/EWCDS formation and shows their applications in the broadcast process. More specifically, we will focus on the following technical issues related to ECDS/EWCDS: (1) The complexity of determining a minimum EDS, ECDS or EWCDS. We will show that these problems are NP-complete. (2) Heuristic solutions to the minimum ECDS/EWCDS problems. We will propose four types of solutions: global for EWCDS based on Guha and Khuller's MCDS, quasi-global for EWCDS based on Alzoubi, Wan, and Frieder's maximal independent set approach, quasi-local for ECDS based on the clustering approach, and local for ECDS based on Wu and Li's marking process. (3) Application of ECDS/EWCDS. We will focus on an application in efficient broadcasting. (4) Activity scheduling and rotation for local

solutions. We will discuss different ways to rotate and schedule active nodes under certain global constraints, including global coverage. (5) Performance analysis. We will conduct performance analysis, through analytical and simulation studies on the proposed solutions.

The remainder of the paper is organized as follows. Section 2 reviews the related work in the field. Section 3 gives a new geometric graph model from which the extended dominating set is defined. The NP-completeness of finding a minimum extended dominating set (EDS) and a minimum extended strongly/weakly connected dominating set (ECDS/EWCDS) are also proved in this section. Section 4 presents several non-local heuristic algorithms for EDS and EWCDS. Section 5 proposes the local solution for ECDS. Applications and related issues are discussed in Section 6. A performance study through simulation is conducted in Section 7. The paper concludes in Section 8.

## 2 Background and Related Work

### 2.1 Cooperative communication (CC)

Extensive research has been done in the area of *cooperative communication* (CC) [24, 21]. The basic idea is the use of single-antenna nodes in a multi-user scenario to share their antennas to create a virtual multiple-input multiple-output (MIMO) system. CC can potentially combine the following advantages: (1) the power savings provided by multi-hopping, (2) the spatial diversity provided by the antennas of separate mobile nodes, and (3) node cooperation can also lead to increased data rates [19]. There are several cooperative signaling methods [24], including detect and forward methods, amplify-and-forward methods, and coded cooperations. In our model, no synchronization is required, that is, the receiver can “assemble”  $k$  copies of the same packet received at different time.

In CC, there are two thresholds on the received signal’s SNR (signal-to-noise ratio):  $\gamma_p$  for successful decoding, and  $\gamma_{acq}$  timing acquisition. A packet received with SNR  $\gamma$  is (1) a failed reception, when  $\gamma < \gamma_{acq}$ , (2) a partial reception, when  $\gamma_{acq} \leq \gamma < \gamma_p$ , or (3) a full reception, when  $\gamma_p \leq \gamma$ . Suppose a node has two partial receptions of the same packet, with  $\gamma_1$  and  $\gamma_2$ . If  $\gamma_1 + \gamma_2 \geq \gamma_p$ , combining these two partial receptions achieves a full reception. This combining process can be extended to multiple partial receptions. The channel gain is often modelled as a power of distance, therefore,  $\gamma/\gamma_p = (r/d_{ij})^\alpha$ , where  $d_{ij}$  is the distance between nodes  $i$  and  $j$ ,  $r$  is the communication range of nodes, and  $2 \leq \alpha \leq 4$  is a communication medium dependant parameter. The header of a message is

coded in a different way that requires a lower SNR ( $\gamma_{acq}$ ) to decode. Therefore, the combiner knows which packet a partial reception belongs to. Signal combining can be performed whenever new partial reception is made in an incremental way. Many partial receptions of a packet do not require extra storage space.

Discussions of the CC technique and its applications in ad hoc networks can be found in [24, 18]. This technique (referred to as *hitchhiking*) has been exploited by Agarwal et al [2] to reduce the total energy consumption in a broadcast process. A heuristic algorithm was used to build a rooted tree that covers the entire network. Then local optimization steps were performed at each level of the tree, where the extra coverage provided by higher level transmissions is used for the transmission power reduction at the current level. Cardei et al [6] extended the work in [2] to address the topology control problem.

## 2.2 Dominating set and its extensions

Finding the minimum dominating set (DS) and minimum connected dominating set (CDS) is NP-complete in both regular graphs [17] and unit disk graphs [11]. Finding an extended (weakly) connected dominating set (ECDS and EWCDS), which takes the advantages of the CC technique, has not been exploited except in [2]. Note that an ECDS/EWCDS is different from a weakly connected dominating set (WCDS) [5, 9]. A WCDS  $W$  of a network is a set such that the subnetwork, consisting of all nodes in the network and all adjacent links to nodes in  $W$ , is connected. Basically, a WCDS is a DS such that it is connected when treating all nodes in the DS within two hops as adjacent. The difference between a WCDS and an ECDS/EWCDS is that WCDS cannot guarantee a full coverage under the CC in broadcasting. That is, in order to achieve full delivery in a broadcasting, both WCDS nodes and some non-WCDS nodes need to forward the broadcast packet. For example, in Fig. 1 (b),  $\{x, u, v\}$  is considered WCDS (and EWCDS) and  $\{x, u, z\}$  is also WCDS (but not EWCDS).

This subsection reviews existing CDS formation protocols for wireless ad hoc and sensor networks. Based on their efficiency in terms of forming a small CDS and overhead in terms of message and time complexity, these protocols were classified into four categories in [33]: global, quasi-global, quasi-local, and local. A global protocol assumes a central point in the network, where global information is available and CDS membership is computed based on this information. Global protocols usually yield the smallest CDS, but their application is limited due to the high maintenance cost. Das

et al [13] proposed a global protocol based on Guha and Khuller’s approximation algorithm [16]. This algorithm, MCDS, is based on “growing” a tree from a selected root until all nodes are covered. Non-leaf nodes form a CDS. MCDS has an  $O(\log \Delta)$  approximation ratio in regular graphs, where  $\Delta$  is the maximum number of neighbors of a node. Recently, Cheng et al [10] proposed a polynomial time approximation scheme (PTAS) for the minimum CDS in unit disk graphs. Given a network of size  $n$  and a small parameter  $s$ , Cheng et al designed a  $(1 + 1/s)$ -approximation with running time  $n^{O((s \log s)^2)}$ .

A quasi-global protocol relies on global coordination rather than global information. The computation, starting from a central point, is propagated in a sequential manner to the entire network. Then, a maximal independent set (MIS) is constructed from the tree. An independent set (IS) is a special type of DS where any two nodes in DS are not adjacent. These protocols usually have a small constant approximation ratio in unit disk graphs, but the high overhead of the global infrastructure makes them less attractive in dynamic networks. Alzoubi, Wan, and Frieder [4] have proposed a quasi-global algorithm (called AWF here) with an approximation ratio of 8 in unit disk graphs. Nodes in the DS are selected from a spanning tree so that when a gateway node is selected by each DS node, the DS becomes a CDS. AWF takes  $O(n)$  rounds to complete.

A quasi-local protocol assumes no central point. However, sequential propagation of information is still possible and, sometimes, expanded to the entire network. These protocols have a large constant approximation ratio in unit disk graphs, but moderate overhead since nodes are selected in parallel to form an MIS. A cluster-based quasi-local algorithm usually contains two phases as in the AWF algorithm. At first, the network is partitioned into clusters; a *clusterhead* is elected for each cluster. Then clusterheads are interconnected to form a CDS. Unlike in AWF, one or two gateways are needed to connect clusterheads separated by two and three hops. Several clustering algorithms have been proposed [14, 23] to elect clusterheads. In most approaches for gateway selection [14, 20, 33], neighboring clusterheads are connected via a mesh structure through a local selection by each clusterhead. These protocols have  $O(1)$  approximation ratios in unit disk graphs. In the worst case, it takes  $O(n)$  rounds for them to complete due to the sequential information propagation in the clustering process. But on average, the expected number of rounds is  $O(\log n)$ .

A local protocol relies only on local information, i.e., properties of nodes within its vicinity. In addition, there is no sequential propagation of any partial computation result. The status of each node depends on its  $l$ -hop topology only for a small constant  $l$ , and is usually determined after  $l$  rounds

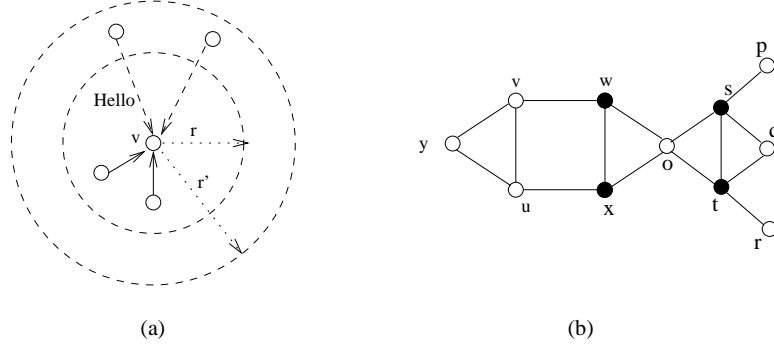


Figure 2: (a) Neighbor detection. (b) A sample extended marking process and extended pruning rule.

of information exchange among neighbors. Local protocols do not have a constant deterministic approximation ratio, but in random unit disk graphs, the expected size of the resultant CDS is  $O(1)$  times that of the minimum CDS. Wu and Li [30, 32] proposed the marking process and two pruning rules that select a few nodes to form a CDS. The approach uses 2-hop information and converges after two rounds of information exchange. Dai and Wu [12] further extended this scheme to use a more efficient pruning rule that ensures a probabilistic approximation ratio. Other local CDS algorithms include Span [8], Multipoint relay (MPR) [1, 25], and core-based approaches [26, 27].

### 3 Extended Dominating Set

#### 3.1 A new geometric graph model

Here we consider a special geometric graph to approximate the CC model: Given a set  $V$  of points in a 2D space, a normal transmission range  $r$ , and a CC range  $r'$ , we define a graph with vertex set  $V$  and an arc from vertex  $v$  to vertex  $u$  iff the Euclidean distance,  $d(v, u)$ , from  $v$  to  $u$  is no more than  $r$ . In addition, a quasi-arc from vertex  $v$  to vertex  $u$  iff  $r < d(v, u) \leq r'$ . When  $r' = 2r$ , the corresponding graph can be approximated by a single unit disk graph, where a quasi-arc exists between any two vertices (called quasi neighbors) that are separated by two hops. When  $r' \neq 2 \times r$  (normally  $r' < 2 \times r$ ), each vertex maintains its regular arcs (the corresponding node is called a regular neighbor, or simply, neighbor) and quasi-arcs (the corresponding node is called a quasi neighbor) through “Hello” messages (see Fig. 2 (a)). It should be stressed that once 1-hop neighborhood is

derived it is used to derive  $k$ -hop neighborhood without resorting to Euclidean distance information. For example, quasi neighbors are derived from 1-hop neighbors of neighbors.

It is assumed that when a quasi neighbor of  $v$  sends a packet,  $v$  can receive only a partial packet. However,  $v$  can “assemble” the complete packet after it receives  $k$  copies of the packet from  $k$  quasi neighbors. It is still assumed that when a regular neighbor of  $v$  sends a packet,  $v$  can receive the complete packet provided no collision occurs. It is also assumed that the network is sparse or moderately dense, where the communication and computation overhead of the proposed approaches is limited. If the network is extremely dense, the network should be first sparsified as discussed in [31].

### 3.2 Definition of extended dominating set

Our objective is to find a minimum connected backbone so that nodes outside the backbone can be put in a sleep state. The backbone construction using the CC feature can be formulated as extended dominating set (EDS) and extended connected dominating set (ECDS).

**Definition 1** *A subset of nodes is an EDS if every node is (a) in the subset, (b) a regular neighbor of a node in the subset, or (c) a quasi neighbor of  $k$  nodes in the subset.*

When  $r' = 2 \times r$  and  $k = 2$ , every node is (a) in the subset, (b) a 1-hop neighbor of a node in the subset, or (c) a two-hop neighbor of 2 nodes in the subset. Note that the regular dominating set (DS) is also an EDS. Under the regular physical model, the connectivity is defined as follows: Two nodes are connected if there exists a sequence (path) of regular neighbors. Under the CC model, nodes are connected not only via regular neighbors but also via quasi neighbors.

**Definition 2** *A set is strongly connected under the CC model if for any node  $u$  in the set sending a packet, the packet should be fully received by all other nodes. Only nodes with a fully received packet (including  $u$ ) are able to forward the packet once.*

If the connectivity condition holds at least for a particular node  $u$ , the set is called *weakly connected*. A strongly connected EDS is denoted as ECDS and a weakly connected EDS is denoted as EWCDS. In Fig. 1 (b),  $\{u, v, x\}$  forms an EWCDS while  $\{u, w, x\}$  forms an ECDS. It is known that the dominating set and connected dominating set problems in unit disk graphs (UD-DS and UD-CDS)



are NP-complete [11]. The following theorem shows that EDS, ECDS and EWCDS are NP-complete. The theorem can be proved by showing that these problems belong to the NP class and either UD-DS or UD-CDS is a special case for each problem.

**Theorem 1** *EDS, ECDS, and EWCDS problems are NP-complete.*

**Proof:** It is easy to see that EDS, ECDS and EWCDS belong to the NP-class. Given a vertex subset, it can be verified in polynomial time whether it forms an EDS, ECDS, or EWCDS with CC model. First, we show that UD-DS is a special case of EDS. Recall from our previous graph definition that any node  $v$  has associated a radius pair  $(r_v, r'_v)$  such that  $r_v \leq r'_v$ , and any quasi neighbor  $u$  satisfies the relation  $r_v \leq d(v, u) \leq r'_v$ . A subset of nodes is an EDS if any node is (a) in the subset, (b) has a neighbor in the subset, or (c) has  $k$  quasi neighbors in the subset. When  $r_v = r'_v$  for any node  $v$ , we will have no case of quasi neighbor or quasi neighbor domination. That is, any node is either part of the EDS or has a neighbor in the EDS. Hence, we say that the UD-DS is a special case of the EDS problem, for  $r_v = r'_v$  for  $\forall v \in V$ .

Next, we show that UD-CDS is a special case of ECDS and EWCDS problems when  $r_v = r'_v$  for  $\forall v \in V$ . When  $r_v = r'_v$  for  $\forall v \in V$ , a subset of nodes  $S$  is an ECDS if the nodes in  $S$  are connected and any node in  $V$  is either part of  $S$  or has a neighbor in  $S$ . Therefore, ECDS reduces to the UD-CDS problem. Also, for the case  $r_v = r'_v$  for  $\forall v \in V$ , a subset of nodes  $S$  is an EWCDS if connectivity holds for at least a node  $u$  in  $S$ , and any node in  $V$  is either part of  $S$  or has a neighbor in  $S$ . If the connectivity holds for node  $u$ , then there exists a path from  $u$  to any other node in  $S$  and this is equivalent with a connected dominating set  $S$ . Therefore, EWCDS reduces to the UD-CDS problem. Since UD-DS and UD-CDS are NP-complete and one of them is a particular case of the EDS, ECDS and EWCDS problems, and because EDS, ECDS, and EWCDS belong to the NP-class, we conclude that EDS, ECDS, and EWCDS are NP-complete problems.  $\square$

## 4 Non-Local Heuristic Solutions

### 4.1 Global solutions for EWCDS

First, we consider a centralized greedy solution, called extended MCDS (E-MCDS), similar to Guha and Khuller's MCDS [16] for the minimum EWCDS. However, the notion of *contribution* is used

here: each forward node contributes 1 to all its neighbors and  $1/k$  to all quasi neighbors. The problem is to find a minimum EWCDs so that all other nodes are *reachable* (i.e., each node has a signal energy of at least 1). The *effective contribution* of  $v$  to  $u$  is  $v$ 's contribution to  $u$  before the signal energy of  $u$  reaches 1. The initial signal energy of each node is zero. For example, suppose the signal energy of node  $u$  is 0.5 before  $v$  forwards the message and  $k = 4$ , the effective contribution of  $v$  to  $u$  is 0.5 if they are neighbors and is 0.25 if they are quasi neighbors. A node is said to have the *maximum effective contribution* if it has the maximum total effective contribution to its neighbors and quasi neighbors. In E-MCDS, the node with the maximum effective contribution is selected as a source to grow a tree. At each round, a neighbor of the tree with the maximum effective contribution is selected until the signal energy of each node in the network is at least 1. For ease of description, we assume  $r' = 2 \times r$  and  $k = 2$  in the following discussion. Nevertheless, all algorithms and theorems in this paper, after a minor revision, also apply to the general situation with any constant ratios  $r'/r$  and  $k$ .

---

**Algorithm 1** Extended MCDS (E-MCDS)

---

1. (Initialization) All nodes are initially colored white, except that the node with the maximum effective contribution is colored gray (and will be the root).
  2. Select the gray node that has the maximum effective contribution to its white neighbors (regular and quasi).
  3. Update the signal energy level for every regular or quasi neighbor of the selected gray node.
  4. The selected gray node is colored black and its white regular neighbors are marked gray. If the signal energy of a white quasi neighbor is at least 1, that neighbor is marked gray also.
  5. Repeat steps 2, 3, and 4 until the signal energy level of each node is at least 1.
- 

Although our simulation study shows that the E-MCDS has good performance in terms of producing a small EWCDs, it does not produce a constant approximation ratio. Note that the original MCDS algorithm by Guha and Khuller [16] does not have a constant approximation ratio either, since it was designed for general graphs, not for unit disk graphs. In order to provide a worst-case guarantee, we refine the first global algorithm using the concept of independent set. A set  $S$  of nodes is an independent set (IS) if all pairwise nodes in  $S$  are not adjacent. If  $S$  is also a dominating set, it is called a maximal independent set (MIS). Our second global algorithm is the same as the first algorithm with one important difference. Two black colors, *black1* and *black2* are used to mark nodes in the EWCDs. In addition, the following mutual exclusion rule must be observed.

**Mutual Exclusion Rule:** *When a gray node is added to the EWCDs, it has color black1 if it has no black1 neighbors and black2 if it has no black2 neighbors. A gray node with both black1 and black2*

neighbors cannot be added to the EWCDs.

Enforcing the above rule will not leave any white nodes uncovered; otherwise, let  $w$  be a white node with a gray neighbor  $g$ . Node  $g$  has at most one black neighbor. It can be legally colored as either *black1* or *black2* and cover  $w$ . If  $g$  has two black neighbors, their total contribution to  $w$  is 1, and  $w$  is already covered. Since a *black1* node cannot have *black1* neighbors, then all the *black1* nodes form an IS. Similarly, all *black2* nodes also form an IS. Let  $G_r = (V, E_r)$  be the unit disk graph with radius  $r$  without considering the CC model. Similarly,  $G_{2r}$  is the unit disk graph with radius  $2r$ , and  $DS^{2-hop}$  is a dominating set for graph  $G_{2r}$ .

**Lemma 1** *If  $S$  is an IS of  $G_r$ , then  $|S| \leq 25 \cdot |EDS_{opt}|$ , where  $EDS_{opt}$  is the optimal solution of the EDS problem.*

**Proof:** Let  $DS_{opt}^{2-hop} = \{v_1, v_2, \dots, v_{opt}\}$  be the optimal solution of the  $DS^{2-hop}$  problem, and  $S_i$  the set of nodes in  $S$  dominated by  $v_i$  in  $G_{2r}$ . It has been proved in [3] that the number of IS nodes in a circle with radius  $2r$  is at most  $\pi(2r + \frac{r}{2})^2 / \pi(\frac{r}{2})^2 = 25$ . That is,  $|S_i| \leq 25$  for  $i = 1, 2, \dots, opt$ . As  $DS^{2-hop}$  dominates  $S \subseteq V$ , we have  $S = S_1 \cup S_2 \cup \dots \cup S_{opt}$ . Therefore,  $|S| \leq |S_1| + |S_2| + \dots + |S_{opt}| \leq 25 \cdot |DS_{opt}^{2-hop}|$ . Finally,  $|DS_{opt}^{2-hop}| \leq |EDS_{opt}|$  because any EDS of  $G_r$  is also a DS of  $G_{2r}$ .  $\square$

**Lemma 2**  $\frac{1}{25} \cdot |DS_{opt}| \leq |EDS_{opt}| \leq |DS_{opt}|$ , where  $DS_{opt}$  and  $EDS_{opt}$  are the optimal solutions for the DS and EDS problems.

**Proof:** The relation  $|EDS_{opt}| \leq |DS_{opt}|$  is clear since any solution of the DS problem is also a solution of the EDS problem. Let us take an MIS,  $S$ . Then  $S$  is also a dominating set, resulting in  $|DS_{opt}| \leq |S|$ . Using Lemma 1,  $|S| \leq 25 \cdot |EDS_{opt}|$ . Therefore,  $|DS_{opt}| \leq 25 \cdot |EDS_{opt}|$ .  $\square$

**Theorem 2** *The extended MCDS algorithm with the mutual exclusive rule has a constant approximation ratio for the EWCDs problem.*

**Proof:** Let  $U$  be the set of black nodes,  $U_{black1}$  the set of *black1* nodes and  $U_{black2}$  the set of *black2* nodes. Both  $U_{black1}$  and  $U_{black2}$  are IS of  $G_r$  and  $|U| = |U_{black1}| + |U_{black2}|$ . From Lemma 1,  $|U_{black1}| \leq 25 \cdot |EWCDs_{opt}|$  and  $|U_{black2}| \leq 25 \cdot |EWCDs_{opt}|$ . Therefore,  $|U| \leq 50 \cdot |EWCDs_{opt}|$ .  $\square$

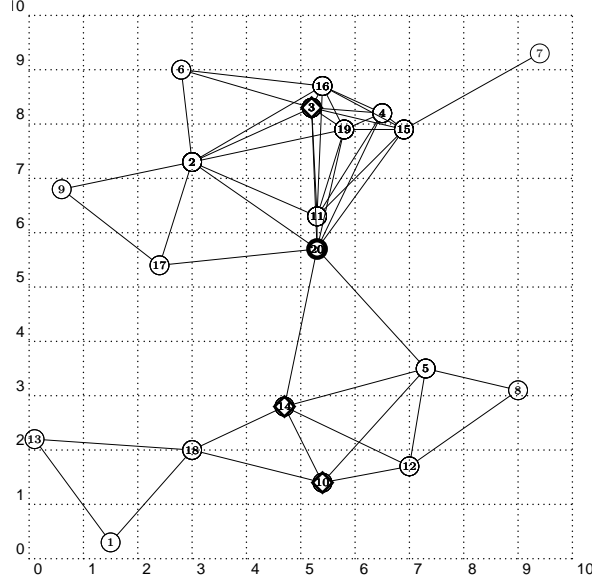


Figure 3: An example to illustrate EWCCS by E-MCDS.

The extended MCDS algorithm runs on a single central node, which collects global information and disseminates the resultant EWCCS to the entire network. Information collection and dissemination takes  $\Theta(H)$  rounds for network diameter  $H$ . The computation cost of the central node is  $O(nD \log n)$  for a network with  $n$  nodes and deployment density  $D$ . The algorithm selects at most  $n$  EWCCS nodes each of which affects effective contributions of up to  $\pi(4r)^2 D = O(D)$  nodes within 4 hops. The positions of these nodes in a sorted list need to be adjusted with  $O(\log n)$  cost each. Fig. 3 shows the EWCCS generated by the E-MCDS algorithm in a random 20 nodes connected graph. There are 4 nodes, 3, 10, 14, and 20, in the resultant EWCCS with the source 20. We can see that every other node has at least two different 2-hop paths or one 1-hop path to reach nodes in EWCCS. Fig. 4 (a) shows the EWCCS generated by the E-MCDS algorithm in a random 100 nodes connected graph. There are 13 nodes in the resultant EWCCS, and node 77 is the source. There are 19 CDS nodes (not shown in the figure) generated by MCDS of the same graph.

## 4.2 Quasi-global solutions for EWCCS

First, we give a simple version of the AWF algorithm proposed by Alzoubi, Wan, and Frieder [4] for CDS which is also a trivial solution for ECDS with a constant approximation ratio. Then, we

propose a solution for the EWCDs problem, by using the extended connectivity concept to reduce the dominating set.

---

**Algorithm 2** AWF Algorithm

---

1. (Topological sorting) A spanning tree is built via flooding from a pre-defined root. Each node  $v$  in the spanning tree is given a rank  $r_v = (l_v, id_v)$ , where  $l_v$  is  $v$ 's level (i.e., distance to root), and  $id_v$  is  $v$ 's ID. Node ranks form a total order.
  2. (Sequential clustering) Initially, all nodes are white. In the sequence from the lowest rank (the root) to the highest rank, each node determines its color. If a node has no neighboring black nodes with lower ranks, it becomes a black node.
  3. (Gateway designation) Each non-root black node selects a white neighbor with a lower rank as its gateway, which connects this black node to another black node with a lower rank.
- 

It has been proved in [4] that (1) all black nodes form an MIS, and (2) the set of black and gateway nodes is a CDS of the network. Obviously, the black node set also forms an EDS, and the black and gateway node set forms an ECDS. In fact, AWF has a constant performance ratio for the ECDS problem. Let  $U$  be the set of both black and gateway nodes, and  $S$  the MIS selected in step 2. The number of nodes used in step 3 is at most one fewer than the number of nodes in  $S$ . Therefore  $|U| \leq 2 \cdot |S| - 1$ . Based on the Lemma 1,  $|S| \leq 25 \cdot |EDS_{opt}|$  and since  $|EDS_{opt}| \leq |ECDS_{opt}|$  we obtain  $|U| < 50 \cdot |ECDS_{opt}| - 1$ .

In our extended AWF algorithm (E-AWF) for EWCDs, the first two steps (topological sorting and sequential clustering) are the same as in the original AWF algorithm. The third step is changed as follows:

**Extended Gateway Designation:** *Each non-root black node designates a neighbor with a lower rank as its gateway, only when it is not reachable from black and gateway nodes with lower ranks; otherwise, this black node does not designate a gateway.*

**Theorem 3** *The set of black nodes and gateways selected by the E-AWF algorithm is an EWCDs.*

**Proof:** Let  $S$  be the set of black nodes. Based on [4],  $S$  is an MIS and EDS of the network. To show the weak connectivity, we show that any black node in  $S$  is reachable from the root. The root is reachable by default. Suppose all black nodes with ranks lower than  $r_v$  are reachable (i.e., they have received the complete packet), we show that black node  $v$  is also reachable.  $v$  has designed a gateway

$g$  only when it is not reachable, then  $g$  is reachable from a black node  $b$  satisfying  $r_b < r_g < r_v$ . Therefore, gateway  $g$  will receive the packet from  $b$  and forward it to  $v$ .  $\square$

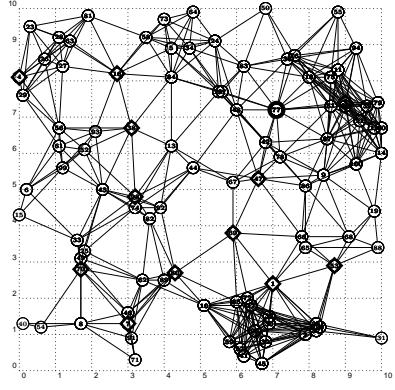
The E-AWF algorithm has a constant approximation ratio for the EWCDs problem. This is because the number of black nodes selected by the E-AWF is exactly the same as in the original AWF. The number of gateways selected by the extended gateway designation rule is at most the same as in the original AWF. Therefore, the size of the EWCDs formed by the E-AWF is no larger than that by the original AWF, which has a constant approximation ratio. The E-AWF algorithm takes  $O(n)$  rounds to converge in the worst case and  $O(H)$  rounds in the best case. The computation cost of each node is  $O(D)$ . We use the same example graph of Fig. 3 to illustrate E-AWF. Applying E-AWF to that graph, we will have the EWCDs =  $\{1, 2, 6, 7, 8, 9, 10, 15, 18, 20\}$ , in which  $\{2, 15, 18\}$  are the gateways and 1 is the source. Node 8 and 20 are connected to 1 through 10 and 18 using the connectivity concept under CC model. Otherwise, as the result of AWF, nodes 5 and 14 are both selected to make the set connected. Fig. 4 (b) shows the EWCDs generated by the E-AWF algorithm in a random 100-node connected graph. There are 23 nodes in the resultant EWCDs, and node 1 is the source. There are 37 CDS nodes generated by AWF for the same graph (not shown in the figure).

### 4.3 A quasi-local solution for EDS and ECDS

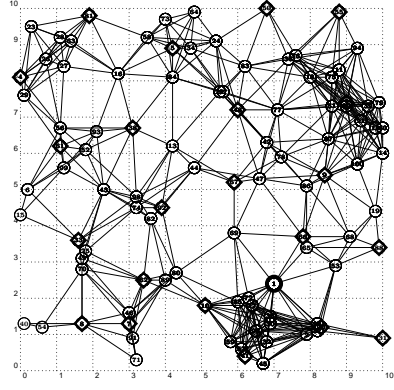
In this subsection, we consider a quasi-local solution for the minimum disconnected EDS and then extend the solution for ECDS. By a quasi-local solution, we mean the solution completes with a high probability in a small number of rounds with an occasional large number of rounds for completion.

This approach is similar to a clustering algorithm with two major differences: (1) The coverage is under the CC model. (2) Each node operates on its 2-hop neighborhood, rather than its 1-hop neighborhood in the regular clustering approach. Nodes in the network are classified into black (selected), gray (covered by a black node), partial gray (partially covered by one or more black nodes), and white (clean). The priority of each node is defined by either its node ID or its node degree (1-hop or 2-hop) as long as the priority is a total order. Therefore, in case of a tie in node degree, node ids can be used to break the tie. Initially, all nodes are colored white.

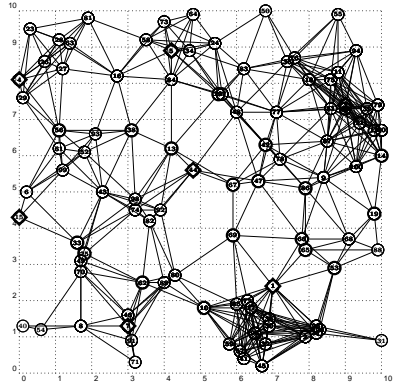
The black nodes (also called clusterheads) generated in the extended clustering form a disconnected EDS and an IS. From Lemma 1, the extended clustering algorithm has a constant approximation ratio for the EDS problem. To extend EDS to ECDS, we need each clusterhead to connect



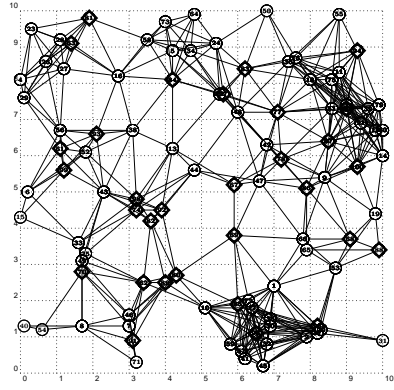
(a) EWCDs by E-MCDS, source is 77



(b) EWCDs by E-AWF, source is 1



(c) ECDS by E-Clustering with gateways



(d) ECDS by E-Rule  $K$

Figure 4: Sample ECDS or EWCDs in a random network with 100 nodes.

to neighboring clusterheads within 5 hops. To find a small number of gateways to connect all the neighboring clusterheads without resorting to global information, we use an extension of Li, Hou and Sha’s local minimum spanning tree (LMST) [22] algorithm on neighboring clusterheads. Unlike the traditional gateway designation algorithm, whereby each clusterhead is connected to all of its neighboring clusterheads and thus the CDS is a mesh structure (Cluster-Mesh), in LMST, each node constructs a local minimum spanning tree within its 1-hop neighborhood, and marks the links to its on-tree neighbors only. Note that the on-tree neighbor set is usually a subset of 1-hop neighbor set. It is proved that all the marked links together with all the nodes can form a connected graph. In our extension, the 1-hop neighborhood includes the current clusterhead CH and all clusterheads within 5 hops, along with their pairwise “virtual distance” in terms of hop count. The IDs of the two end nodes of a “virtual link” can be used to break a tie in hop count if needed. In this way, each pair

---

**Algorithm 3** Extended Clustering (E-Clustering)

---

1. A white node with the highest priority within its 2-hop white neighborhood is colored black.
  2. A partial gray node is colored black if (a) there is no white neighbor within its 2-hop neighborhood, and (b) it has the highest priority among all partial gray nodes in its 2-hop neighborhood.
  3. For any recently-turned black node, its neighbors are colored gray if they are either white or partial gray. Signal energy of quasi neighbors are adjusted and their colors are changed accordingly. That is, a white quasi neighbor is changed to partial gray. A partial gray node with signal energy level of at least 1 is changed to gray.
- 

of neighboring clusterheads has a virtual link with a virtual distance. When a virtual link is selected in LMST, i.e., a link connecting CH to a neighboring clusterhead, all nodes on the virtual link are selected as gateways.

---

**Algorithm 4** Localized Tree-based Gateway Designation (E-Cluster-LMST)

---

1. Each clusterhead constructs a local minimum spanning tree (LMST) among all the clusterheads within its 5-hop neighborhood rooted at itself, using virtual links.
  2. Each clusterhead selects the on-tree neighbors and marks all the intermediate nodes as gateways on the virtual links to these neighbors.
- 

**Theorem 4** *The EDS generated by extended clustering and gateway nodes together form an ECDS which has a constant approximation ratio.*

**Proof:** First, we prove the connectivity of resultant dominating set including clusterheads and gateways. We show that all the clusterheads are connected by virtual links. Arbitrarily select two clusterheads  $CH_u$  and  $CH_v$  and assume the shortest path between them in the original graph is  $(CH_u, u_1, u_2, \dots, u_k, CH_v)$ . For each  $u_i$  ( $1 \leq i \leq k$ ), its clusterhead  $CH_i$  is within 2 hops of  $u_i$ . Therefore, any two adjacent clusterheads in sequence  $CH_u, CH_1, CH_2, \dots, CH_k, CH_v$  are separated by at most 5 hops. When each clusterhead connects to all clusterheads within 5 hops through virtual links,  $CH_u$  and  $CH_v$  are connected. Based on [22], if the original virtual graph is connected, the sub-graph induced by the virtual links selected by LMST is still connected. Each virtual link consists of gateways used to connect two clusterheads. Therefore, clusterheads together with gateways appearing in the selected virtual links form a connected graph. Then, we prove the constant approximation ratio of the resultant ECDS. From previous discussion, we know that the size of clusterheads  $|U|$  that form



an EDS has a constant approximation ratio. In local tree-based gateway designation, each clusterhead at most has all the clusterheads within its 5-hop neighborhood as its LMST neighbors, which is bounded by  $O(1)$  [3]. Therefore, the gateways designated by each clusterhead is at most  $4 \cdot O(1)$  (4 gateways on a virtual link at most). We can have now the size of selected clusterheads and gateway nodes to be  $|U| + O(1) \cdot |U|$  in the worst case. Therefore, it has a constant approximation ratio.  $\square$

The clustering process takes  $O(n)$  rounds in the worst case, and  $O(\log n)$  rounds on average. The computation cost of clustering is  $O(D)$  for each node, and that of neighborhood designation is  $O(D)$  in building a distance vector of  $O(1)$  neighboring clusterheads, and  $O(1)$  in LMST construction. Applying E-Clustering and E-Cluster-LMST to the graph in Fig. 3, we have the EDS =  $\{1, 2, 7, 8\}$  and the ECDS =  $\{1, 2, 7, 8, 5, 10, 14, 15, 20\}$  respectively. Fig. 4 (c) shows the ECDS generated by E-Clustering with gateways in a random 100-node connected graph. The clusterheads are noted by diamonds, and the gateways by bold circles. There are 25 nodes in the resultant ECDS, of which 7 clusterheads are in the EDS. There are 35 CDS nodes generated by clustering with gateways on the same graph, of which 19 clusterheads form the DS (not shown in the figure).

## 5 Local Heuristic Solutions

### 5.1 A local solution for EDS and ECDS

In local backbone construction, each node maintains only 2-hop neighborhood information. The local solution consists of two steps: (1) use Wu and Li's marking process [32] and Dai and Wu's pruning rule [12] for constructing the CDS. Note that a CDS is also an ECDS. (2) Adopt an aggressive pruning rule to remove nodes from the CDS while still maintaining local coverage and connectivity. Specifically, Wu and Li's marking process is the following:

**Marking Process** [32]: *A node is temporarily marked if it has two neighbors that are not directly connected.*

It has been shown in [32] that the temporarily marked node set is a CDS. When the marking process is applied to the example in Fig. 2, the temporarily marked node set  $\{v, u, w, x, o, s, t\}$  forms a CDS. After a set of temporarily marked nodes is derived by applying the marking process, it can be further reduced via the following pruning rule.

**Pruning Rule  $K$**  [12]: *A temporarily marked node  $u$  can be unmarked if all its 1-hop neighbors ( $N_1(u)$ ) are also neighbors of any one of  $K$  coverage nodes that are connected and have higher priorities.*

When each node in  $N_1(u)$  is a neighbor of  $C$ ,  $u$  is said to be *covered* by  $C$ . In Fig. 2 (b),  $\{w, v\}$  covers  $u$  and, hence,  $u$  can be unmarked. A *restricted* version of the above pruning rule exists, where all coverage nodes must be 1-hop neighbors of the unmarked node. For example, node  $u$  in Fig. 2 (b) cannot be unmarked based on the restricted Rule  $K$ , because  $w$  is not a neighbor of  $u$ . It was proved in [12] that the reduced set of temporarily marked nodes is still a CDS after applying pruning Rule  $K$ , either restricted or non-restricted.

In the extended pruning rule (E-Rule  $K$ ), 2-hop neighborhood information, including temporary markers of all 2-hop neighbors, is needed. A temporarily marked node  $u$  can be unmarked if all its 2-hop neighbors, regular and quasi, can be covered by other temporarily marked nodes in the neighborhood, and the corresponding condition is called *coverage condition*. Let  $C$  be a set of temporarily marked nodes with higher priority than  $u$  within  $u$ 's 2-hop neighborhood. Again, the priority of a node can be node id and node degree. The neighbor set of node  $u$ ,  $N_2(u)$ , includes both regular and quasi neighbors. When the coverage condition holds,  $u$  is said to be (extendedly) covered by  $C$ . In Fig. 2 (b),  $\{w, x\}$  covers  $v$  and, hence,  $v$  can be unmarked.

**Coverage Condition:** *A temporarily marked node  $u$  can be unmarked if for each  $v \in N_2(u)$ , the collective energy contribution of  $C$  to  $v$  is at least 1.*

**Theorem 5** *The set derived by the pruning rule based on the coverage condition forms an EDS.*

**Proof:** The set derived from the marking process is a CDS which is clearly an EDS. Consider a single application of the coverage condition on  $u$  in an EDS, since  $u$  is covered by other higher priority nodes in the EDS. The removal of  $u$  from the EDS will not change its extended dominating set property. When there are simultaneous removals, since the node priority is a total order, no cyclic dependence among nodes in terms of coverage will occur; the remaining nodes form an EDS.  $\square$

Note that the pruning rule based on the coverage condition does not guarantee an ECDS even though the set is an ECDS initially. To ensure connectivity, we require  $C$  to be connected under the CC model. We call  $C$  an *extended component* if it is strongly connected (based on Definition 2). The fact that  $u$  is reachable from  $C$  is denoted as  $C \rightarrow u$  (i.e., the total energy contribution of  $C$  to  $u$  is at

least 1). If  $C'$  is a component (defined based on the regular connectivity),  $C$  can reach  $C'$ , denoted as  $C \rightarrow C'$ , if  $C \rightarrow u$  for a  $u$  in  $C'$ . Next we give a procedure for constructing an extended component: Given a set of components (based on the regular connectivity),  $C_1, C_2, \dots, C_m$ , the corresponding extended components are derived by iteratively merging two (regular and extended) components,  $C_i$  and  $C_j$ , whenever they satisfy  $C_i \rightarrow C_j$  and  $C_j \rightarrow C_i$ . In Fig. 2 (b),  $C_1 = \{w, x\}$  and  $C_2 = \{s, t\}$  form two regular components. Since  $C_1 \rightarrow C_2$  and  $C_2 \rightarrow C_1$ ,  $C_1$  and  $C_2$  can be merged into one extended component (based on Definition 2). In Fig. 2, the extended component  $\{w, x, s, t\}$  covers  $o$  and  $o$  is unmarked.

**Connectivity Condition:** *The coverage set  $C_u$  for  $u$  is an extended component. In addition, each marked node in  $N_2(u)$  is adjacent to a node in  $C_u$ .*

**Theorem 6** *A pruning rule that meets coverage and connectivity conditions ensures an ECDS when the given set is ECDS.*

**Proof:** We use the following process of *sequential removal* to emulate the application of coverage and connectivity conditions: Nodes that are unmarked by these two conditions are first sorted in an ascending order of node priority. Then, nodes in the sorted list are removed one by one, with one per round. At each round, the node with the smallest priority is removed from the list. Assume that vertex  $u$  is selected at round  $l$  and it is the first node such that the coverage set  $C$  is an ECDS before its removal but  $C' = C - \{u\}$  is no longer an ECDS after its removal. We prove by contradiction that such a  $u$  does not exist and, therefore, the set after the sequential removal is still an ECDS. If  $v (\neq u)$  in  $C$  sends a packet,  $u$  will receive the packet fully since  $C$  is an ECDS. There is a marked node  $w$  in  $N_2(u)$  that “excites”  $u$ . That is,  $w$  forwards the packet to make an energy contribution to  $u$ . Based on the connectivity condition,  $w$  will excite all nodes in  $C_u$  (the coverage set for  $u$  and a subset of  $C$ ), which in turn will cover all nodes in  $N_2(u)$  (and the coverage condition holds). In this case, all marked nodes in  $N_2(u)$  fully receive the packet. Because  $u$  cannot make a contribution to any node outside  $N_2(u)$ ,  $u$  can be removed which is a contradiction.  $\square$

Note that any given network before pruning is a trivial ECDS and the CDS derived from the marking process is also a trivial ECDS. A better, pragmatic approach starts from the CDS derived from Dai and Wu’s (restricted) pruning Rule  $K$ . The corresponding local solution is called (*restricted*) *extended Rule  $K$* . Notice the similarity between the pruning Rule  $K$  and the coverage and connectivity conditions. The major difference is that Rule  $K$  does not use the CC model. Therefore, the connectivity

and component are defined in a normal term. In addition, the coverage is on  $u$ 's 1-hop neighbor set and the connectivity condition is trivially satisfied in Rule  $K$ .

**Theorem 7** *The localized algorithm computes an ECDS of expected size  $O(1) \cdot |ECDS_{opt}|$ , where  $ECDS_{opt}$  is an optimal solution to the ECDS problem.*

**Proof:** The size of the ECDS is upper bounded by the number of temporarily marked nodes derived from the marking process and Rule  $K$ . It has been proved in [12] that the expected number of temporarily marked nodes after applying Rule  $K$  is  $O(1) \cdot |DS_{opt}|$  in unit disk graphs with both node IDs and locations being uniformly distributed, where  $DS_{opt}$  is the minimal DS. The resultant ECDS is of size  $O(1) \cdot |DS_{opt}|$ . By Lemma 2,  $|DS_{opt}| \leq O(1) \cdot |EDS_{opt}| \leq O(1) \cdot |ECDS_{opt}|$ .  $\square$

The localized algorithm converges after 4 rounds of ‘‘Hello’’ message exchanges: 2 rounds to collect the 2-hop information for the marking process and Rule  $K$ , and 2 additional rounds to disseminate temporary markers. ‘‘Hello’’ messages carrying 1-hop information are of size  $O(D)$ . The computation complexity is  $O(D^2)$  in constructing (extended) components and confirming coverage in a subgraph with  $O(D)$  nodes. Applying E-Rule  $K$  to the graph in Fig. 3, we have the ECDS = {14, 15, 16, 17, 18, 19, 20}. We can see that node 8 can reach the ECDS using the CC technology. Otherwise, node 12 will be added to make a CDS. Fig. 4 (d) shows the ECDS generated by the extended Rule  $K$  in a random 100-node connected graph. There are 31 nodes in the resultant ECDS. There are 35 CDS nodes generated by Rule  $K$  in the same graph (not shown in the figure).

## 6 Applications and Related Issues

### 6.1 Applications

The ECDS/EWCDS can be used as a virtual backbone under the CC model. Such a backbone can support an efficient broadcast process and a searching space reduction. We use the broadcast process as an example. EWCDS can be used for a specific node whereas ECDS can be applied for any node to carry out a broadcast process. A typical broadcast process involves the following steps: (1) (At the source node  $s$ ) If  $s$  is in EDS,  $s$  follows step 2; if  $s$  has a neighbor  $u$  in EDS, it forwards the packet to  $u$  and  $u$  then follows step 2; otherwise,  $s$  selects a neighbor  $v$  that has a neighbor  $u$  in EDS. In the last

case,  $s$  first forwards the packet and then  $u$  follows step 2 after the packet is relayed by  $v$ . (2) (At an intermediate node  $u$ ) If  $u$  is in EDS, it forwards the complete packet once; otherwise, it does nothing.

Unlike broadcasting using regular DS, the source node may need a relay node (not in EDS) to forward the packet to a node in EDS; otherwise, only the nodes in EDS need to forward the packet once. Suppose  $y$  is the source node in Fig. 1 (b), it forwards the packet to  $u$  in EDS. Each of  $u$  and  $v$  forwards once. After  $x$  assembles the two partial packets from  $u$  and  $v$ , it forwards the complete packet once to reach  $t$ . The construction of an EDS increases the overhead, but the impact is minimal compared with the benefit of reducing the number of forwarding nodes. From all distributed approaches, the localized solutions incur a minimal overhead. In addition, a localized solution is very efficient in a dynamic environment since it supports localized maintenance.

## 6.2 Activity scheduling/rotation

The aim of the activity scheduling/rotation is to provide a good trade-off between minimizing energy consumption in sensor monitoring and prolonging the life span of each individual node. The backbone approach minimizes overall energy consumption by putting the maximum number of nodes in a sleep state. However, this comes with significant energy consumption by the nodes in the backbone. We propose to rotate the role of dominating (active) and non-dominating (sleep) nodes based on energy level in backbone construction. The localized scheduling/rotation can be as follows:

---

### Algorithm 5 Localized Scheduling/Rotation

---

1. Apply the marking process and extended pruning rule to determine the *marker* (i.e., marked/unmarked status) of each node, so unmarked nodes can be put into sleep mode.
  2. Each active node can judiciously lower its priority in the next round of scheduling (rule application).
- 

After an active node has lowered its priority (called *tired*), its new priority is propagated to 2-hop neighbors. Any changes of temporary markers are also propagated to 2-hop neighbors. Here we assume an asynchronous wake up scheme [29, 34] for communication among neighbors, and the propagation delay of each hop is bounded by the scheduling frame  $T$ . Therefore, the rotation process takes a non-trivial time period to complete. In a real situation, sensor nodes may fail (called *off*) and new sensor nodes can be deployed (called *on*). An on/off operation can change the network topology and, hence, the corresponding ECDS needs to be modified, and the corresponding operation is called maintenance. In general, ECDS maintenance cannot be done in a localized way in non-local solutions

(such as the extended MCDS) without sacrificing performance (such as approximation ratio). On the other hand, an ECDS derived from the marking process and extended pruning rule can be maintained in a localized way, where only nodes in a small vicinity of on/off nodes need to modify their markers.

**Theorem 8** *In the restricted extended Rule  $K$ , only nodes within 3 hops of a tired/on/off node need to change their final markers.*

**Proof:** Let  $u$  be a tired, on, or off node. First consider the temporary marker of a node  $v$ . Based on definitions of the marking process and restricted Rule  $K$ , the temporary marker of  $v$  depends only on the list of  $v$ 's 1-hop neighbors, their priorities, and wireless links among them. As we have assumed that a wireless link does not break unless an end node switches off, the temporary marker of  $v$  remains the same if  $u$  is not a 1-hop neighbor of  $v$ . That is, only  $u$ 's 1-hop neighbors need to change temporary markers. Then consider the final marker of a node  $w$ . Based on the coverage and connectivity conditions, the final marker of a node  $w$  depends only on the list of  $w$ 's 2-hop neighbors, their priorities, their temporary markers, and wireless links among them. After excluding the impact of wireless links based on the previous assumption,  $w$ 's final marker changes only when (1)  $u$  is within 2 hops of  $w$ , or (2)  $u$  has a 1-hop  $v$  that is within 2 hops of  $w$  and has changed its temporary marker. Therefore,  $w$  changes its final marker only when it is within 3 hops of  $u$ .  $\square$

The above theorem shows that a tired/on/off node affects only nodes within 3 hops, and the process converges after three rounds of “Hello” exchanges, which means a handover interval of  $3T$ . For a smooth handover, the following rule is used to preserve an ECDS during a rotation process.

**Rotation Rule:** *All active nodes newly unmarked in a rotation process must stay active for additional three scheduling frames ( $3T$ ) before switching to the sleep mode.*

**Theorem 9** *The rotation rule preserves an ECDS during the rotation process.*

**Proof:** Let  $C(t)$  be the set of active nodes at time  $t$ . Assume the rotation process starts at  $t_0$ . By Theorem 6,  $C(t)$ , the set of marked nodes, is an ECDS for  $t \leq t_0$ . By Theorem 8, the rotation process converges no later than  $t_0 + 3T$ ; that is,  $C(t)$  is an ECDS for  $t \geq t_0 + 3T$ .  $C(t_0) \subseteq C(t)$  for  $t \in [t_0, t_0 + 3T]$  by rotation rule. Because  $C(t_0)$  is an ECDS,  $C(t)$  is an ECDS during this period.  $\square$

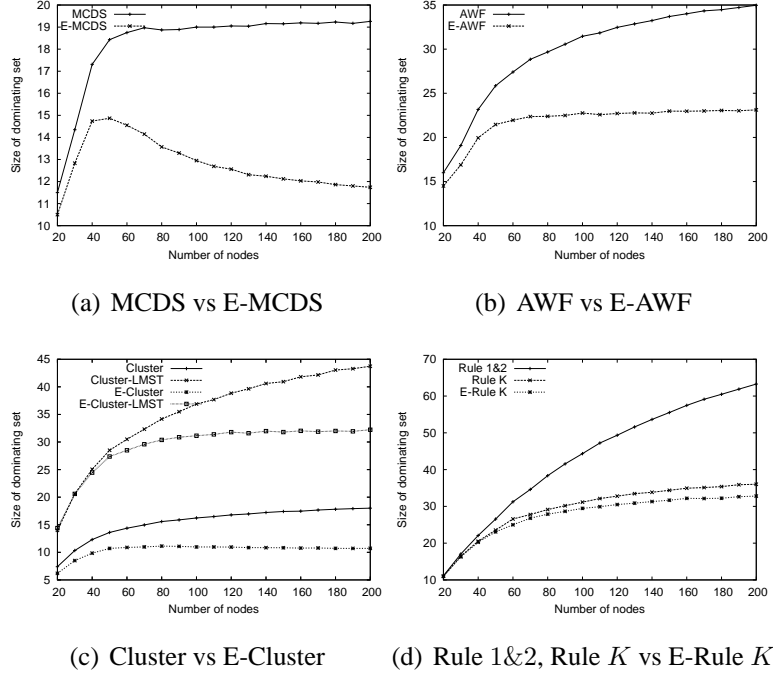


Figure 5: Different algorithms with various node number ( $r = 20$ ).

## 7 Simulation

This section presents results from our simulation. The efficiency of all proposed approaches are evaluated and compared. In the simulation, the extended MCDS (E-MCDS), which is a global solution for EWCDS, is compared with MCDS. The extended AWF (E-AWF), which is a quasi-global solution for EWCDS, is compared against AWF. The quasi-local solutions, extended clustering without gateways (E-Cluster) and with gateways (E-Cluster-LMST), are compared with regular clustering without gateways (Cluster) and regular clustering with gateways (Cluster-Mesh and Cluster-LMST), respectively. Note that in Cluster-LMST the new local method is used for gateway selection, but clusterheads are still selected based on the regular clustering. The extended Rule  $K$  (E-Rule  $K$ ) for ECDS is compared with Rule  $K$  and the original marking process and pruning rules (Rule 1&2). All extended dominating set algorithms using CC are also evaluated with different choices of simulation parameters.

All of the above approaches are implemented on a custom simulator. To generate a random network,  $n$  nodes are randomly placed in a restricted  $100 \times 100$  area. We assume all nodes have the same transmission range; therefore, all links between them are bidirectional. Networks that cannot form a

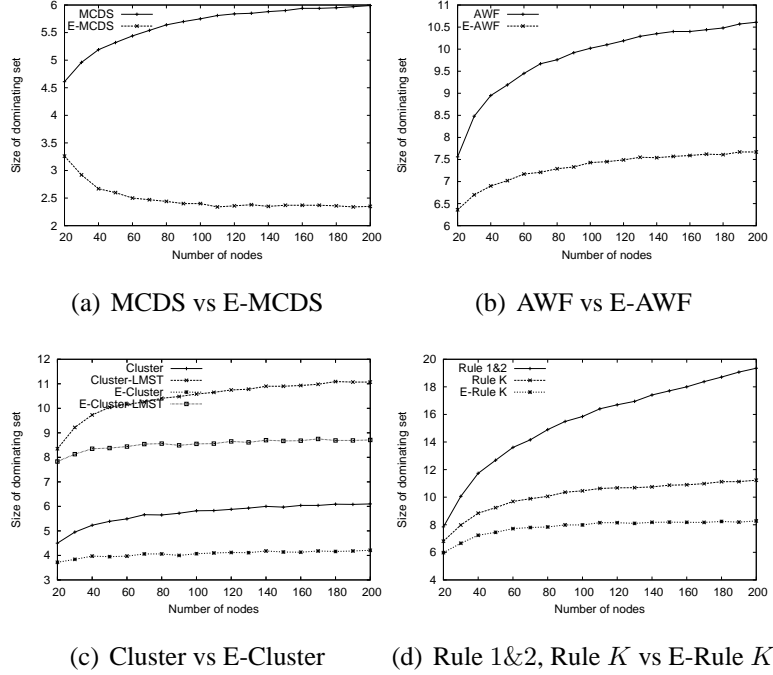


Figure 6: Different algorithms with various node number ( $r = 40$ ).

strongly connected graph are discarded. The tunable parameters in our simulation are: (1) The node number  $n$ . We change the number of deployed nodes from 20 to 200 to check the scalability of the algorithms. (2) The transmission range  $r$ . We use 20 and 40 as transmission ranges to witness the effect of link density on the algorithms. (3) The energy contribution parameter  $k$ .  $k$  controls the coverage contribution of a node to its quasi neighbors. We use 2, 3, 4 and 5 as its values. The performance metric is the number of nodes in the resultant (connected) dominating set or extended (connected or weakly connected) dominating set. For each tunable parameter, the simulation is repeated 1000 times or until the confidence interval is sufficiently small ( $\pm 1\%$ , for the confidence level of 90%).

Fig. 5 is the comparison of proposed algorithms in relatively sparse networks with transmission range 20. (a) compares the performance of E-MCDS, in terms of the size of resultant EWCDs, with MCDS. With CC and the new connectivity condition, E-MCDS has better performance (i.e., produces smaller EWCDs than the CDS of MCDS). The size of EWCDs increases with the number of nodes, but will decrease after the node count reaches 50. This is because, at first, more nodes need a larger EWCDs to cover them, but after the node number exceeds a threshold, the increased node density helps to select a smaller EWCDs in better positions. (b) shows the performance of E-AWF and AWF.



E-AWF has better performance, especially when node number is large, where the EWCDS has a stable size. (c) compares the performance of E-Clustering and Clustering with and without gateways. Clearly, E-Cluster-LMST and E-Cluster beat Cluster-LMST and Cluster, respectively. (d) presents the performance of Rule 1&2, Rule  $K$ , and E-Rule  $K$ . E-Rule  $K$  has the smallest size of ECDS, and reduces the size of CDS generated by Rule  $K$  by 7%.

Fig. 6 shows the comparison of these algorithms in relatively dense networks with transmission range 40. Every ECDS/EWCDS has smaller size than the corresponding CDS. Actually, the resultant figures are very close to the curves of Fig. 5 with a large node number. Therefore, the curve in Fig. 6 (a) for E-MCDS is monotonously decreasing. We can see that the extended algorithms have much better performance in dense networks. When the number of nodes is 200, E-Rule  $K$  can reduce the resultant dominating set of Rule  $K$  by 23%.

Fig. 7 is the comparisons of these algorithms under various transmission ranges and a fixed node number. When the node number is fixed, increasing transmission range results in a relatively dense network. It is quite the same procedure with increasing node number under a fixed transmission range. Therefore, in the following simulation, we vary only the number of nodes to test the scalability of the algorithms.

Fig. 8 (a) and Fig. 8 (b) show the performance comparison of algorithms to generate ECDS (E-Cluster-LMST and E-Rule  $k$ ) and CDS (Cluster-LMST and Rule  $K$ ). Rule  $K$  performs slightly better than Cluster-LMST although they are close when the network is dense and has relatively small diameter (as in (b)). One interesting observation is that when the network is dense and has relatively small diameter (as in (b)), E-Rule  $k$  still beats E-cluster-LMST, although E-Rule  $k$  uses only 2-hop information. When the network is sparse with relatively large diameter (as in (a)), E-Rule  $K$  and E-cluster-LMST stay very close. One explanation is that the additional neighborhood information used in E-cluster-LMST can take more advantage of the effect of CC in such a graph than the 2-hop information used in E-Rule  $K$ . However, E-Rule  $K$  is a local approach using 2 rounds, whereas E-cluster-LMST is a quasi-local approach using non-constant rounds and multiple-hop information for gateway selection. E-Rule  $K$  is clearly a better choice. Fig. 8 (c) and Fig. 8 (d) show the performance comparison of algorithms to generate EWCDS (E-MCDS and E-AWF) and CDS (MCDS and AWF). It is clear that E-MCDS and MCDS have better performance than E-AWF and AWF, respectively.

Fig. 9 is the performance comparison with different  $k$ ,  $r = 20$ . For both E-AWF and E-Rule  $K$ , the resultant EWCDS and ECDS increase gradually as  $k$  increases. For E-MCDS, the resultant

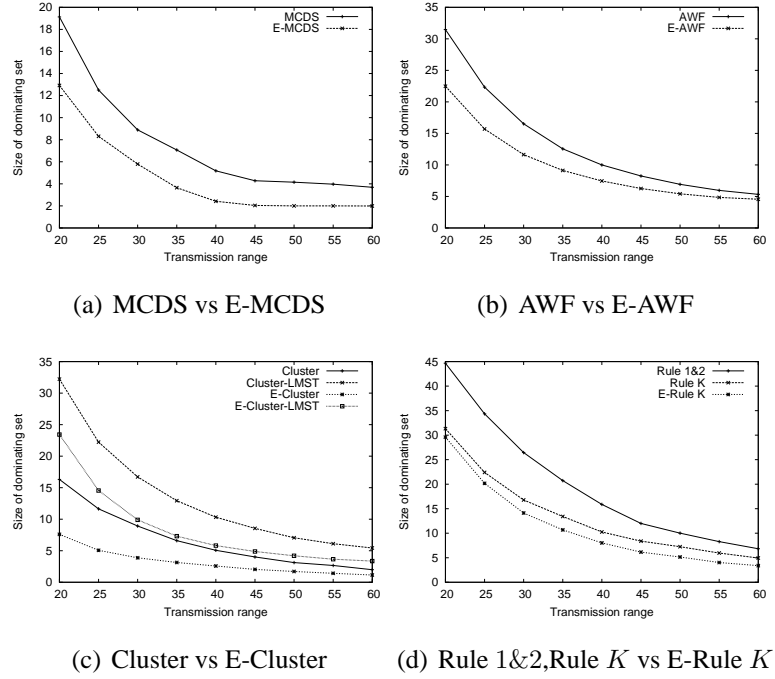


Figure 7: Different algorithms with various transmission range ( $n = 100$ ).

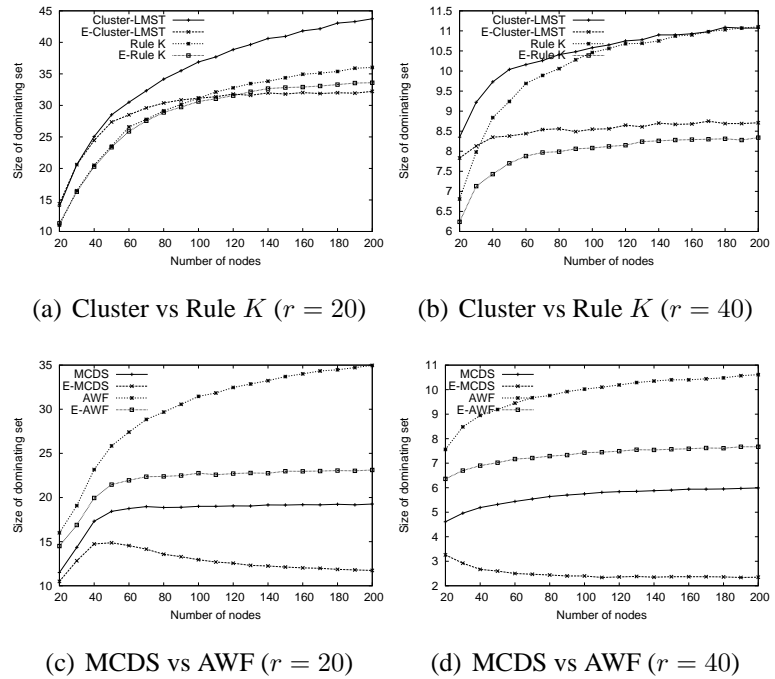


Figure 8: Comparison of Cluster and Rule  $K$ , MCDS and AWF.

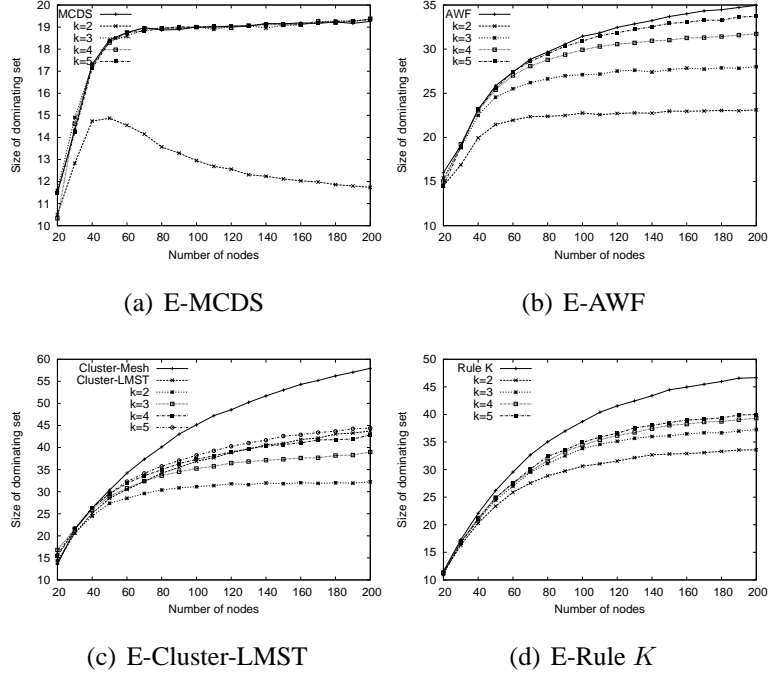


Figure 9: Extended CDS of different algorithms with  $k = 2, 3, 4, 5$ .

EWCDs degrades quickly to the CDS of MCDS when  $k$  is larger than 2. Cluster-LMST is better than Cluster-Mesh. On the other hand, Cluster-LMST uses more neighbor information. E-Cluster-LMST is better than Cluster-LMST for  $k = 2$  and  $k = 3$ . When  $k$  reaches 4, the resultant ECDS will be similar to that of CDS by Cluster-LMST, because as  $k$  increases, the contribution of a node to its quasi neighbors decreases and cannot offset the additional gateways introduced as a result of the longer distance between two neighboring clusterheads in extended clustering.

The simulation results can be summarized as follows: (1) All the proposed extended dominating set algorithms can generate smaller extended (connected or weakly connected) dominating sets than the corresponding (connected) dominating sets. (2) When the network is relatively dense, the extended dominating set algorithms have better performance and generate smaller extended (connected or weakly connected) dominating sets. (3) Among four proposed approaches, E-MCDS has the best performance for producing EWCDs and E-Rule  $K$  has the best performance for producing ECDS, although E-Rule  $K$  is just a local solution. (4) When  $k$  is more than 2, except for E-MCDS, the other approaches can still generate a smaller dominating set, although the size gradually increases as  $k$  increases and is close to the corresponding dominating sets under the regular model.

## 8 Conclusions

In this paper, we describe an extended dominating set (EDS) based on the cooperative communication model. Some non-trivial extensions of the methods for the regular dominating set are presented. The problems of finding a minimum EDS, ECDS (connected EDS) and EWCDS (weakly connected EDS) are shown as NP-complete. Several heuristic algorithms, global and local, are presented. The focus is on local solutions that can offer local maintenance. The efficiency of node reduction in dominating set is confirmed through simulation study for both sparse and dense graphs. The potential applications of ECDS/EWCDS for the broadcast process is also discussed. In future work, we will examine other local solutions for ECDS, such as the extension to multipoint relay (MPR) [25], which is a localized extended MCDS. Each node selects its backbone nodes to cover its 3-hop coverage area. Collectively, locally selected backbone nodes form an ECDS. In addition, an in-depth simulation using a complete protocol stack that can better reflect the CC model is needed for some real applications of ECDS/EWCDS.

## References

- [1] C. Adjih, P. Jacquet, and L. Viennot. Computing connected dominated sets with multipoint relays. Technical Report 4597, INRIA-Rapport de recherche, October 2002.
- [2] M. Agarwal, J. H. Cho, L. Gao, and J. Wu. Energy efficient broadcast in wireless ad hoc networks with hitch-hiking. In *Proc. of INFOCOM*, March 2004.
- [3] K. Alzoubi, X. Y. Li, Y. Wang, P. J. Wan, and O. Frieder. Geometric spanners for wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(5):408–421, 2003.
- [4] K. M. Alzoubi, P. J. Wan, and O. Frieder. Distributed heuristics for connected dominating set in wireless ad hoc networks. *Journal of Communications and Networks*, 4(1):22–29, 2002.
- [5] K. M. Alzoubi, P. J. Wan, and O. Frieder. Weakly-connected dominating set and sparse spanners in wireless ad hoc networks. In *Proc. of ICDCS*, March 2003.
- [6] M. Cardei, J. Wu, and S. Yang. Topology control in ad hoc wireless networks with hitch-hiking. In *Proc. of SECOM*, August 2004.
- [7] J. Carle and D. Simplot-Ryl. Energy-efficient area monitoring for sensor networks. *IEEE Computer*, 37(2):40–46, 2004.

- [8] 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 Journal*, 8(5):481–494, 2002.
- [9] Y. P. Chen and L. Liestmen. Approximating minimum size weakly-connected dominating sets for clustering mobile ad hoc networks. In *Proc. of MobiHoc*, June 2002.
- [10] X. Cheng, X. Huang, D. Li, W. Wu, and D. Du. Polynomial-time approximation scheme for minimum connected dominating set in ad hoc wireless networks. *Networks*, 42(4):202–208, 2003.
- [11] B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86:165–177, 1991.
- [12] F. Dai and J. Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transaction on Parallel and Distributed Systems*, 15(10):908–920, 2004.
- [13] B. Das, E. Sivakumar, and V. Bhargavan. Routing in ad-hoc networks using a virtual backbone. In *Proc. of IC3N*, September 1997.
- [14] A. Ephremides, J. E. Wjeselthier, and D. J. Baker. A design concept for reliable mobile radio networks with frequency hopping signaling. in *Proc. of IEEE*, 71(1):56–73, 1987.
- [15] L. M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proc. of INFOCOM*, April 2001.
- [16] S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387, 1998.
- [17] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater. *Foundamentals of Domination in Graphs*. A Series of Monographs and Text books. Marcel Dekker, Inc., 1998.
- [18] Y. W. Hong and A. Scaglione. Energy-efficient broadcasting with cooperative transmission in wireless sensor and ad hoc networks. *IEEE Transactions on Wireless Communications*, September 2004, accepted to appear.
- [19] N. Jindal, U. Mitra, and A. Goldsmith. Capacity of ad-hoc networks with node cooperation. In *Proc. of IEEE International Symposium on Information Theory (ISIT)*, pages 271–272, June 2004.
- [20] T. J. Kwon and M. Gerla. Efficient flooding with passive clustering (PC) in ad hoc networks. *ACM SIGCOMM Computer Communication Review*, 32(1):44–56, 2002.
- [21] N. Laneman, D. Tse, and G. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information Theory*, 50(12):3060–3080, 2004.
- [22] N. Li, J. Hou, and L. Sha. Design and analysis of an MST-based topology control algorithm. In *Proc. of INFOCOM*, March 2003.

- [23] C. R. Lin and M. Gerla. Adaptive clustering for mobile wireless networks. *IEEE Journal on Selected Areas in Communications*, 15(7):1265–1275, 1996.
- [24] A. Nosratinia, T. E. Hunter, and A. Hedayat. Cooperative communication in wireless networks. *IEEE Transactions on Communications*, 42(10):74–80, 2004.
- [25] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast message in mobile wireless networks. In *Proc. of HICSS-35*, January 2002.
- [26] P. Sinha, R. Sivakumar, and V. Bharghavan. CEDAR: a core-extraction distributed ad hoc routing algorithm. *IEEE Journal on Selected Areas in Communications, Special Issue on Ad Hoc Networks*, 17(8):1454–1465, 1999.
- [27] P. Sinha, R. Sivakumar, and V. Bharghavan. Enhancing ad hoc routing with dynamic virtual infrastructures. In *Proc. of INFOCOM*, March 2001.
- [28] I. Stojmenovic, M. Seddigh, and J. Zunic. Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 13(1):14–25, 2002.
- [29] Y. Tseng, C. Hsu, and T. Hsieh. Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks. In *Proc. of INFOCOM*, March 2002.
- [30] J. Wu. Extended dominating-set-based routing in ad hoc wireless networks with unidirectional links. *IEEE Transactions on Parallel and Distributed Systems*, 13(9):866–881, 2002.
- [31] J. Wu and F. Dai. A distributed formation of a virtual backbone in manets using adjustable transmission ranges. In *Proc. of ICDCS*, March 2004.
- [32] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proc. of DIALM*, August 1999.
- [33] J. Wu and W. Lou. Forward node set based broadcast in clustered mobile ad hoc networks. *Wireless Communications and Mobile Computing*, 3(2):141–154, 2003.
- [34] R. Zheng, J. C. Hou, and L. Sha. Asynchronous wakeup for ad hoc networks. In *Proc. of MobiHoc*, June 2003.