# Application of Dominating Sets in Wireless Sensor Networks

Amir Hassani Karbasi and Reza Ebrahimi Atani

*Department of Computer Engineering, Guilan University, Rasht, Iran*

*amirhassanikarbasi@gmail.com, rebrahimi@guilan.ac.ir*

***Abstract***

*Wireless sensor networks (WSNs) consists of spatially distributed autonomous sensor nodes to* monitor *physical or environmental conditions because of that they have widespread applications in human communities and existence world. Like other computer and telecommunication networks, wireless sensor networks are susceptible to regarding destructive threats and attacks and simple hardware of these electronic devices prevents applying defensive mechanisms called networks. Positioning key is of main performances of coding in all kinds of applications in which security is considered as an anxiety. Despite the nature of limited resources on sensor nodes limits using current routing and security mechanisms in wireless sensor networks. Secure routing protocol design because of being under the influence of resource limitations and physical nodes of the sensor is essential for wireless sensor networks. In this paper, we study topology control and virtual backbone based on dominating sets for wireless sensor networks. We deal with the performance evaluation of connected dominating set construction algorithms.*

***Keywords:*** *Wireless Sensor Networks, Dominating Sets, Routing Security, Topology Control, Virtual Backbone, CDS Construction Algorithms*

## 1. Introduction

Wireless Sensor Networks (WSNs) are becoming increasingly attractive for a variety of application areas, including industrial automation, security, weather analysis, and a broad range of military scenarios. Wireless sensor networks are dense wireless networks of sensor nodes collecting and disseminating environmental data. Sensor nodes are small low-power devices constrained severely in their computation, communication, and storage capabilities, usually for economical reasons. They may sense around themselves, communicate over wireless channels within short ranges, and frequently fall into the sleep mode for saving their power. A sensor node typically contains a power unit, a sensing unit, a processing unit, a storage unit, and a wireless transmitter / receiver. Figure 1 shows the hardware architecture of a sensor node [1].

Communication in WSNs usually occurs in ad hoc manner, and shows similarities to wireless ad hoc networks. Likewise, WSNs are dynamic in the sense that radio range and network connectivity changes by time. Sensor nodes dies and new sensor nodes may be added to the network. However, WSNs are more constrained, denser, and may suffer (or take advantage) of redundant information. WSN architectures are organized in hierarchical and distributed structures as shown in Figure 2 [2, 3].

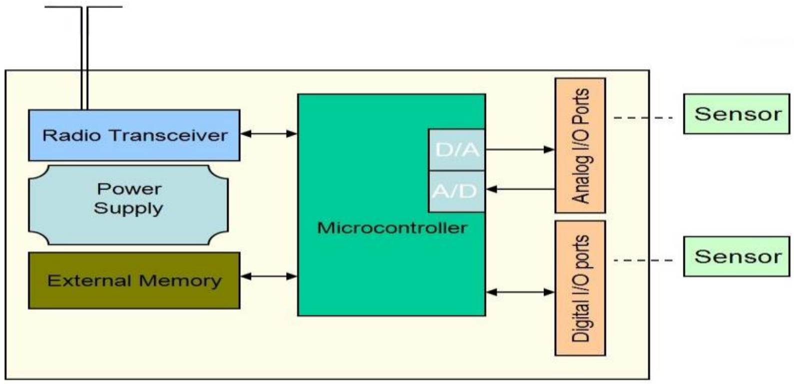


**Figure 1. Hardware Architecture of a Sensor Node**

A Hierarchical Sensor Network (HSN) is shown in Figure 2(a); there is a hierarchy among the nodes based on their capabilities: base stations, cluster heads and sensor nodes. Base stations are many orders of magnitude more powerful than sensor nodes and cluster heads. A base station is typically a gateway to another network, a powerful data processing / storage center, or an access point for human interface. Base stations collect sensor readings, perform costly operations on behalf of sensor nodes and manage the network. In some applications, base stations are assumed to be trusted and temper resistant. Thus, they are used as key distribution centers. Sensor nodes are deployed around one or more hop neighborhood of the base stations. They form a dense network where a cluster of sensors lying in a specific area may provide similar or close readings. Nodes with better resources, named as cluster heads, may be used collect and merge local traffic and send it to base stations. Transmission power of a base station is usually enough to reach all sensor nodes, but sensor nodes depend on the ad hoc communication to reach base stations.

A Distributed Sensor Network (DSN) is shown in Figure 2(b); there is no fixed infrastructure, and network topology is not known prior to deployment. Sensor nodes are usually randomly scattered all over the target area. Once they are deployed, each sensor node scans its radio coverage area to figure out its neighbors.

DSN is basically a wireless sensor network with a large number of sensors and large coverage area. It differs from the traditional wireless sensor network in the sense that, it contains considerably huge number of sensors which are intended to be deployed over hostile and hazardous areas where the communications among the sensors could be monitored, the sensors are under constant threat of being captured by the enemy or manipulated by the adversaries.

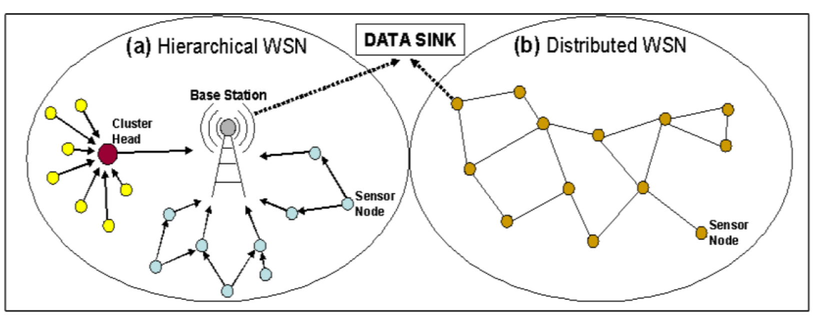


**Figure 2. Network Models: Hierarchical and Distributed Wireless Sensor Networks**

DSN is dynamic in nature in the sense that, new sensors could be added or deleted whenever necessary. These types of networks are suitable for covering large areas for monitoring, tracking, surveillance and moving object detection which are very crucial tasks in many military and public-oriented operations.

We assume that sensor nodes have no information on where they are located, that is, they are distributed in a random way. Once deployed, they operate unattended for extended periods without any movement. Accordingly, a large scale wireless sensor network is composed of a number of sensor nodes for covering wider area through multi-hop connections. Figure 3 shows the distributed wireless sensor network model. The challenge of designing these systems to be robust in the face of myriad security threats is a priority issue. Since sensor nodes are deployed in unattended fashions or even in hostile environments, they can readily be captured and tampered by adversaries as well as communication links are compromised. Thus, backbone establishment is necessary in the way of achieving perfect resilience to minimize a compromised range to one-hop distance only [4, 5].

This paper concerns dominating sets and other related subsets in graphs. Our goal is to present a self-contained, comprehensive treatment of this subject – an area of graph theory that is rich in applications of virtual backbone establishment and topology control in WSNs. This paper contains most of the necessary definitions and some preliminary graph theoretic results.

In wireless ad hoc and sensor networks (or simply wireless networks), autonomous nodes form self-organized networks without centralized control or infrastructure. These networks can be modelled as unit disk graphs [6], where two nodes are neighbors if they are within each other's transmission range. A unit disk is a disk with radius one. A unit disk graph (UDG) is associated with a set of unit disks in the Euclidean plane. Each vertex is the center of a unit disk. An edge exists between two vertices u and v if and only if $|uv| \leq 1$ where $|uv|$ is the Euclidean distance 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.

When all nodes are connected to each other, via a single-hop or multihop path, the WSN is said to have full connectivity. In most real applications, however, the UDG model cannot fully characterize the behavior of wireless links. This is mainly due to the transitional region phenomenon which has been revealed by many empirical studies [7-10]. Beyond the "always connected" region, there is a transitional region where pair of nodes are probabilistically connected. Such pairs of nodes are not fully connected but reachable via the so called lossy links [4]. As reported in [4], there are often much more lossy links than fully connected links in a WSN. Additionally, in a specific setup [5], more than 90% of the network links are lossy links. Therefore, their impact can hardly be neglected.

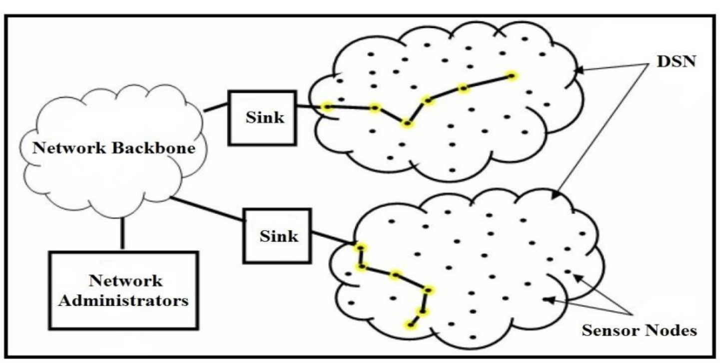


**Figure 3. Distributed Wireless Sensor Network Model**

The employment of lossy links in WSNs is not straightforward, since when the lossy links are employed, the WSN may have no guarantee of full network connectivity. When data transmissions are conducted over such topologies, it may degrade the node-to-node delivery ratio. Usually a WSN has large node density and high data redundancy, thus this certain degraded performance may be acceptable for many WSN applications. Therefore, as long as an expected percentage of the nodes can be reached, that is the node-to-node delivery ratio satisfies some preset requirement, lossy links are tolerable in a WSN. In other words, full network connectivity is not always a necessity. Some applications of dominating sets can trade full network connectivity for a higher energy-efficiency and larger network capacity [11].

In Section 2, we define some definitions of dominating sets in graphs. In Section 3, we introduce topology control in wireless sensor networks. Section 4 describes the constructing CDSs to serve as a virtual backbone for a WSN. Finally, we conclude this paper in Section 5.

## 2. Dominating Sets in Graphs

A number of definitions from graph theory are used in this section. Figure 4 can help to illustrate the following concepts:

- *Open Neighbor Set*, N(u) = {v | (u, v) $\in$ E}, is the set of nodes that are neighbors of u. In Figure 4, the open neighbor set of e is {d, f, g}.

- *Closed Neighbor Set*, N[u] = N(u) $\cup$ {u}, is the set of neighbors of u and u itself. In Figure 4, the closed neighbor set of e is {d, e, f, g}.

- *Maximum Degree*, $\Delta$, is the maximum count of edges emanating from a single node. The maximum degree of the graph in Figure 4 is three, and occurs at nodes c, e, and g.

- *Independent Set*, is a subset of V such that no two vertices within the set are adjacent in V . For example, {a, b, f, h} is an independent set in Figure 4.

- *Maximal Independent Set (MIS),* is an independent set such that adding any vertex not in the set breaks the independence property of the set. Thus, any vertex outside of the maximal independent set must be adjacent to some node in the set. The previous independent set {a, b, f, h} must have node d added to become an MIS.

- *Dominating Set*, S, is defined as a subset of V such that each node in V - S is adjacent to at least one node in S. Thus, every MIS is a dominating set. However, since nodes in a dominating set may be adjacent to each other, not every dominating set is an MIS. Finding a minimum-sized dominating set or MDS is NP-Hard [12].

Notice that if S is a dominating set of a graph G, then every superset S' $\supset$ S is also a dominating set. On the other hand, not every subset S" $\subset$ S is necessarily a dominating set. We will be interested in studying minimal dominating sets in graphs, where a dominating set S is a minimal dominating set if no proper subset S" $\subset$ S is a dominating set.

The set of all minimal dominating sets of a graph G is denoted by MDS(G). Figure 5 illustrates a graph having minimal dominating sets of cardinality three (the set {1,3,5}), four (the set {3,6,7,8}), and five (the set {2,4,6,7,8}).

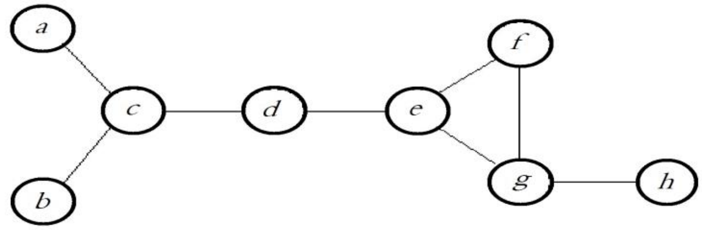


**Figure 4. Representation of a Wireless Network with Eight Nodes as a Graph**

- *Connected Dominating Set (CDS)*, C, is a dominating set of G which induces a connected subgraph of G. One approach to constructing a CDS is to find an MIS, and then add additional vertices as needed to connect the nodes in the MIS. A CDS in Figure 4 is {c, d, e, g}.

- *Minimum Connected Dominating Set (MCDS)* is the CDS with minimum cardinality. Given that finding minimum sized dominating set is NP-Hard, it should not be surprising that finding the MCDS is also NP-Hard [12]. In Figure 4, {c, g} is a minimum connected dominating set.

- *Weakly Connected Dominated Set (WCDS)*, S, is a dominating set such that N[S] induces a connected sub graph of G. In other words, the sub graph weakly induced by S is the graph induced by the vertex set containing S and its neighbors. Given a connected graph G, all of the dominating sets of G are weakly connected. Computing a minimum WCDS is NP-Hard [12].

- *Steiner Tree*, is a minimum weight tree connecting a given set of vertices in a weighted graph. After finding an MIS, connecting the nodes together could be formulated as an instance of the Steiner Tree problem. Like many of the other problems that arise in CDS construction, this problem is NP-Hard [12].

### 2.1. Applications of a CDS in Wireless Sensor Networks

As first noted by Ephremedis *et al.*, a CDS can create a virtual network backbone for packet routing and control [13]. Messages can be routed from the source to a neighbor in the dominating set, along the CDS to the dominating set member closest to the destination node, and then finally to the destination. This is termed dominating set based routing [14, 15], or Backbone based routing [16], or spine based routing [17, 18]. Restricting the routing to the CDS results in a significant reduction in message overhead associated with routing updates [19]. Furthermore, the dominating set can be organized into a hierarchy to further reduce control message overhead [20-22].

A CDS is also useful for location-based routing. In location-based routing, messages are forwarded based on the geographical coordinates of the hosts, rather than topological connectivity. Intermediate nodes are selected based on their proximity to the message's destination. With this scheme, it is possible for pa message to reach a local maximum, where it has been sent to an intermediate node whose neighbors are all further from the destination than itself. In this case, the routing must enter a recovery phase, where the route may backtrack to find another path. However, if messages are only forwarded to nodes in the dominating set, the inefficiency associated with this recovery phase can be greatly reduced [23].

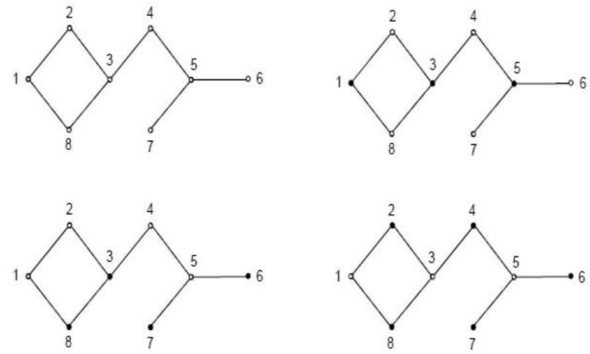


**Figure 5. Minimal Dominating Sets**

The efficiency of multicast / broadcast routing can also be improved through the utilization of CDSs. A big problem in multicast / broadcast routing is that many intermediate nodes unnecessarily forward a message. Nodes often hear the same message multiple times. This is the broadcast storm problem [24]. If the message is routed along a CDS, most of the redundant broadcasts can be eliminated [25-32].

Nodes in a wireless network often have a limited energy supply. CDSs play an important role in power management. They have been used to increase the number of nodes that can be in an sleep mode, while still preserving the ability of the network to forward messages [33-35]. They have also been used to balance the network management requirements to conserve energy among nodes [15, 30, 31, 32, 36, 37].

In large-scale dense sensor networks, sensor topology information extraction can be handled by CDS construction [34, 38]. Other than routing, the virtual backbone formed by dominating set can also be used to propagate "link quality" information for route selection for multimedia traffic [21], or to serve as database servers [39], *etc*.

Previous studies in this area have focused on finding a minimal CDS for higher efficiency. However, recent studies [40-44] suggested that it is equally important to maintain a certain degree of redundancy in the virtual backbone for fault tolerance and routing flexibility.

## 2.2. Bounds on the Domination Number

In general, when studying subsets of a given type, we are interested in finding either a smallest or a largest such set in a graph. For instance, one considers such problems as finding the minimum cardinality of a dominating set or a cover, or finding the maximum cardinality of an independent set or a packing. Since most of these subset problems are NP-complete for arbitrary graphs, it is natural to find bounds for these numbers. In this section, we describe bounds for the domination number $\gamma$ (G).

Enumerating all minimal dominating sets allows immediate solution of corresponding NP-hard optimization and counting problems. An upper bound was given in Fomin *et al.*, [45].

The number of minimal dominating sets in a graph on n vertices is at most 1.7159n.

Fomin *et al.*, also give a lower bound. There is a graph on n vertices with 15n/6 minimal dominating sets. This gives a lower bound of 1.5704n for the maximum number of minimal dominating sets.

## 3. Topology Control in Wireless Sensor Networks

One perhaps typical characteristic of wireless sensor networks is the possibility of deploying many nodes in a small area, for example, to ensure sufficient coverage of an area or to have redundancy present in the network to protect against node failures. These are clear advantages of a dense network deployment, however there are also disadvantages. In a relatively crowded network, many typical wireless networking problems are aggravated by the large number of neighbors: many nodes interfere with each other, there are a lot of possible routes, nodes might needlessly use large transmission power to talk to distant nodes directly (also limiting the reuse of wireless bandwidth), and routing protocols might have to recompute routes even if only small node movements have happened.

Some of these problems can be overcome by topology-control techniques. Instead of using the possible connectivity of a network to its maximum possible extent, a deliberate choice is made to restrict the topology of the network by Dominating sets. The topology of a network is determined by the subset of active nodes (Dominators) and the set of active links along which direct communication can occur. Formally speaking, a topology-control algorithm takes a graph G = (V,E) representing the network, where V is the set of all nodes in the network and there is an edge $(v_1, v_2) \in E \subseteq V^2$ if and only if nodes $v_1$ and $v_2$ can directly communicate with each other.

Hence all active node forms an induced graph $T = (V_T , E_T )$ such that $V_T \subseteq V$ and $E_T \subseteq E$.

### 3.1. Options for Topology Control

To compute an induced graph T out of a graph G representing the original network G, a topology control algorithm has a few options:

- The set of active nodes can be reduced ($V_T \subset V$ ), for example, by periodically switching off nodes with low energy reserves and activating other nodes instead, exploiting redundant deployment in doing so.

- The set of active links / the set of neighbors for a node can be controlled. Instead of using all links in the network, some links can be disregarded and communication is restricted to crucial links. When a distributed network topology (all nodes are considered equal) is desired, the set of neighbors of a node can be reduced by simply not communicating with some neighbors. There are several possible approaches to choose neighbors, but one that is obviously promising for a WSN is to limit the reach of a node's transmissions - typically by power control, but also by using adaptive modulations (using faster modulations is only possible over shorter distances) - and using the improved energy efficiency when communicating only with nearby neighbors. In essence, power control attempts to optimize the trade-off between the higher likelihood of finding a (useful) receiver at higher power values on the one hand and the increased chance of collisions / interference / reduced spatial reuse on the other hand.

- Active links / neighbors can also be rearranged in a hierarchical network topology where some nodes assume special roles. One example, illustrated in Figure 6, is to select some nodes as a Virtual Backbone (VB) for the network and to only use the links within this backbone and direct links from other nodes to the backbone. To do so, the backbone has to form a Dominating Set (DS). Then, only the links between nodes of the dominating set or between other nodes and a member of the active set are maintained. For a backbone to be useful, it should be connected. A related, but slightly different, idea is to partition the network into clusters, illustrated in Figure 7. Clusters are subsets of nodes that together include all nodes of the original graph such that, for each cluster, certain conditions hold (details vary). The most typical problem formulation is to find clusters with cluster heads, which is a representative of a cluster such that each node is only one hop away from its cluster head. When the (average) number of nodes in a cluster should be minimized, this is equivalent to finding a maximum (dominating) independent set (MIS). In such a clustered network, only links within a cluster are maintained (typically only those involving the cluster head) as also selected links between clusters to ensure connectivity of the whole network. Both problems are intrinsically hard and various approximations and relaxations have been studied.

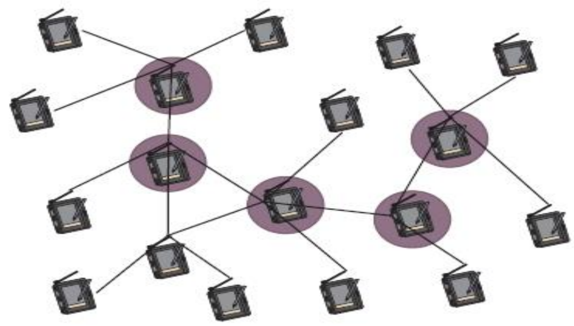

**Figure 6. Restricting the Topology by using Dominating Sets**

## 4. Constructing CDSs

The problem of constructing a minimum CDS (MCDS) is NP-complete [12]. Some algorithms (Table 1) were proposed in literature. According to the methods that all those algorithms are used, we classify them into two categories, Centralized and Distributed. Although the centralized approach can obtain a CDS with small size, it is not feasible to use in a real application since the topology of the whole network should be known in advance. Furthermore, for distributed algorithms we classify them into two subclasses, prune-based and MIS-based as [46] does (see Figure 8). The theoretical analysis and performance evaluation are based on the Unit Disk Graph (UDG), thus, the lemma is given in [47] as following for evaluation metrics.

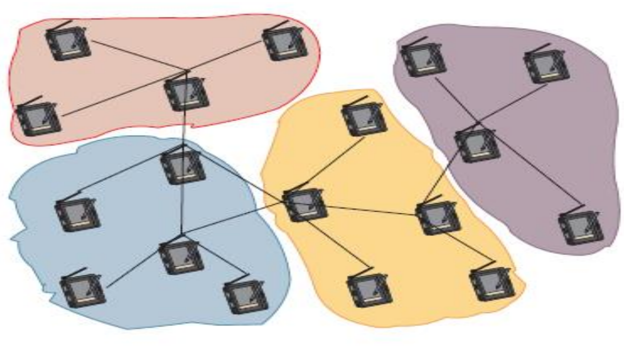

**Figure 7. Using Clusters to Partition a Graph**

**Lemma1.** The neighbor area of a vertex contains at most five independent vertices (Figure 9).

**Lemma 2.** The unit arc-triangle A cannot contain two independent vertices (Figure 10).

**Lemma 3.** The neighbor area of two adjacent vertices contains at most eight independent vertices (Figure 11).

**Lemma 4.** For any unit disk graph, there exists a minimum spanning tree such that every vertex has degree at most five.

**Lemma 5.** Every tree T with at least three vertices has a non-leaf vertex adjacent to at most one non-leaf vertex.

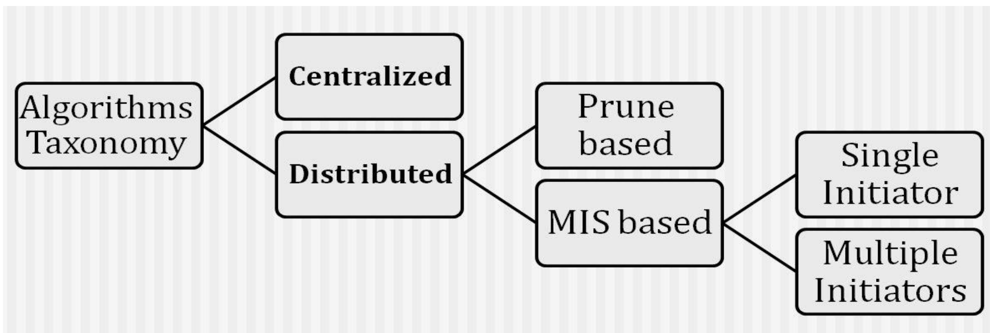


**Figure 8. CDS Construction Taxonomy**

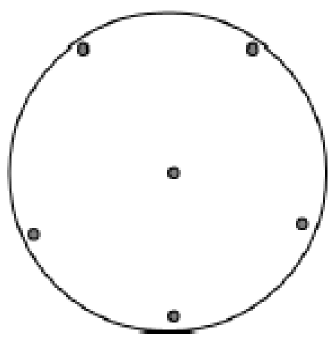

**Figure 9. A Neighboring Area with 5 Independent Nodes**

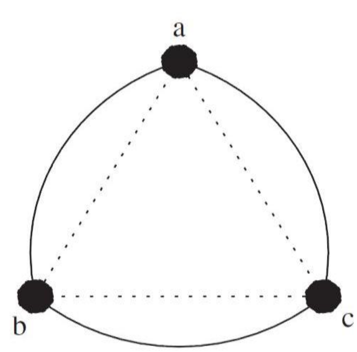


**Figure 10. Unit arc-triangle abc**

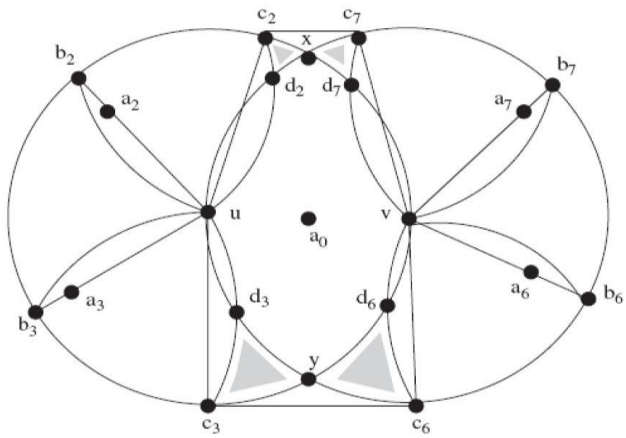


**Figure 11. A Neighboring Area with Two Adjacent Vertices contains 8 Independent Nodes**

### 4.1. Centralized Algorithms

Marathe *et al.*, [48]: A heuristic Connected Domination (CDOM) algorithm is firstly introduced by Marathe *et al.*, In CDOM, a vertex v is picked arbitrarily as root. Rooted from v, a breadth-first spanning tree T is constructed. Let $S_i$ be the nodes at level i. First, v is an MIS node and added to $IS_i$. Let $DS_i$ be the nodes which can be dominated by some vertex in IS. Then in each level i, an MIS of $S_i$-$DS_i$ is constructed. For each node in this MIS, a connector needs to be added to $NS_i$. This procedure is executed from the lowest level to the highest level. Finally all nodes in IS and NS form a CDS. The performance ratio of CDOM is 10 based on Lemma 1.

-Guha and Khuller [49]: The first algorithm begins by marking all vertices white. Initially, the algorithm selects the node with the maximal number of white neighbors. The selected vertex is marked black and its neighbors are marked gray. The algorithm then iteratively scans the gray nodes and their white neighbors, and selects the gray node or the pair of nodes (a gray node and one of its white neighbors), whichever has the maximal number of white neighbors. The selected node or the selected pair of nodes are marked black, with their white neighbors marked gray. Once all of the vertices are marked gray or black, the algorithm terminates. All the black nodes form a connected dominating set. This algorithm yields a CDS of size at most $2(1 + H(\Delta))$, where H is the harmonic function.

The second algorithm also begins by coloring all nodes white. A piece is defined to be either a connected black component, or a white node. The algorithm contains two phases. The first phase iteratively selects a node that causes the maximum reduction of the number of pieces. In other words, the greedy choice for each step in the first phase is the node that can decrease the maximum number of pieces. Once a node is selected, it is marked black and its white neighbors are marked gray. The first phase terminates when no white node left. After the first phase, there exists at most OPTIMAL number of connected black components. The second phase constructs a Steiner Tree that connects all the black nodes by coloring chains of two gray nodes black. The size of the resulting CDS formed by all black nodes is at most $(3 + \ln(\Delta))$.

-Ruan *et al.*, [50]: Proposed a one-step greedy approximation algorithm with performance ratio at most $3 + \ln(\Delta)$. Let C be the set containing all black nodes. Given a connected graph

G(V,E), define p(C) be the number of connected black components in the sub graph induced by $C \subset V$. Let D(C) be the set of all edges incident to vertices in C. Define q(C) be the number of connected components in the sub graph G(V,D(C)). Then the potential function is defined to be f(C) = p(C) + q(C). Initially f(C) = | V | since $C = \varnothing$. The first step chooses a node x with maximum degree. Every other step selects a node x such that $f(C) - f(C \cup \{x\})$ is maximized.

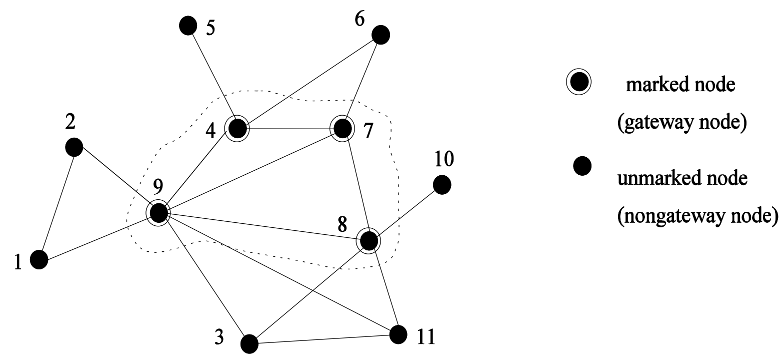


**Figure 12. A Sampled Wireless Sensor Network**

Color node x black and color all its white neighbors gray. The algorithm ends when f(C) = 2, where C is the resultant CDS.

### 4.2. Distributed Algorithms (Prune-based Algorithms)

-Wu *et al.*, [51, 52]: A node is marked true if it has two unconnected neighbors. A set of marked nodes (backbone or gateway nodes) V' form a connected dominating set (Figure 12).

-Chen *et al.*, [53]: Proposed Span, Span is proactive, thus each node periodically broadcasts HELLO messages. From these HELLO messages, each node constructs a list of the node's neighbors and coordinators and for each neighbor, a list of its neighbors and coordinators (Figure 13). Span has two procedures, coordinator announcement and coordinator withdrawal.

Coordinator announcement: Coordinator eligibility rule: A non-coordinator node should become a coordinator if it discovers, using only information gathered from local broadcast messages, that two of its neighbors cannot reach each other either directly or via one or two coordinators.

Announcement contention occurs when multiple nodes discover the lack of a coordinator at the same time, and all decide to become a coordinator Span resolves contention by delaying coordinator announcements with a randomized back-off delay (Figure 14).

Coordinator withdrawal: Each coordinator periodically checks if it should withdraw as a coordinator. A node should withdraw if every pair of its neighbors can reach each other either directly or via one or two other coordinators.

```
HELLO: <Id>,<isCoordinator>,<list of coordinators>,<list of neighbors>,<timeStamp>
```

```
Table:
<13>,<True>,<27>,<3,23>,<11934>
<25>,<False>,<2,23>,<3,45>,<11933>
<27>,< True >,<8,12>,<10,34>,<12001>
<43>,<False>,<13,56>,<3,34>,<11912>
```

**Figure 13. Data Structure**

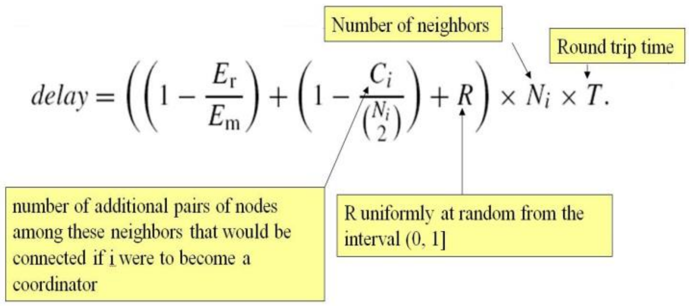


**Figure 14. Randomized back-off Delay**

### 4.3. Distributed Algorithms (MIS-based Algorithms)

#### 4.3.1. Single Initiator Algorithms

-Das *et al.*, [54-56]: (i) All nodes are initially colored white. (ii) The node with the maximum node degree is selected as the root and colored black. All the neighbors of the root are colored gray. (iii) Select a gray node that has the maximum white neighbors. The gray node is colored black and its white neighbors are marked gray. Repeat step (iii) until there is no more white node (Figure 15).

-Wan *et al.*, [47, 57]: Proposed a Single Initiator algorithm. They first choose a root by using distributed leader-election algorithm and build a spanning tree. The rank of each node is determined by a weight function which is the ordered pair (level, ID). After that, the root will move on the construction of the MIS by a color-marking process. All nodes are initially marked with white. The root first marks itself black and broadcasts a BLACK message. Upon receiving a BLACK message, a white node marks itself gray and broadcasts a GRAY message. If a white node receives all GRAY messages from its lower rank neighbors, it marks itself black and broadcast BLACK message. When a leaf node is marked with either gray or black, it transmits a MARK-COMPLETE message to it parent. When one node receives all MARK-COMPLETE messages from its children with high rank, it transmits a MARK-COMPLETE message to its parent. The second phase constructs a tree spanning all the MIS nodes, referred to as dominating tree. In this phase, the root first transfer it role to a gray neighbor which has the largest number of black neighbors. From this new root, a dominating tree is constructed. A gray node marks itself black when it can connect a black node to the dominating tree. All nodes in this dominating tree form a CDS (Figure 16).

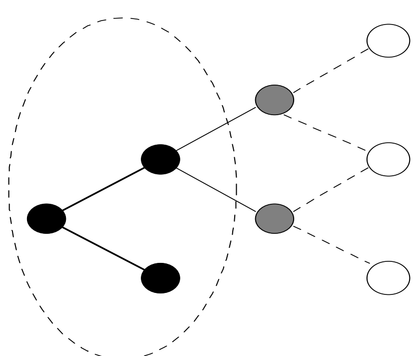

**Figure 15. Black Nodes are Connected Dominating Set (CDS)**

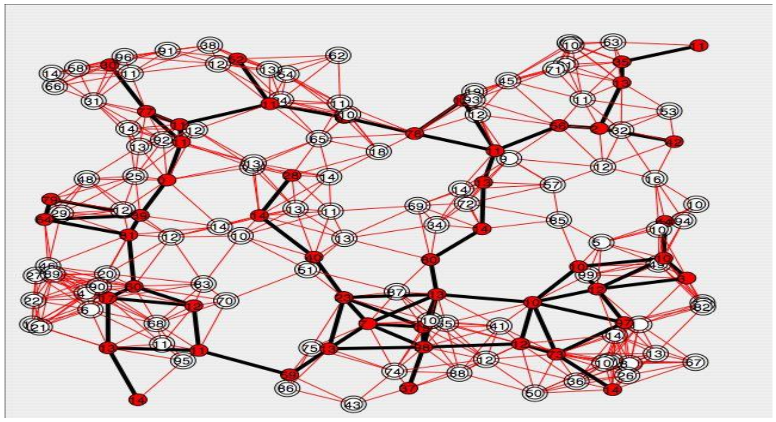

**Figure 16. A Wan *et al.*, Backbone**

-Cheng *et al.*, [58]: The first algorithm grows a spanning tree distributedly from the leader, with all non-leaf nodes form a CDS. Firstly, the leader colors itself black and broadcasts a dominator message. Upon receiving a dominator message, a white or yellow node colors itself gray and broadcasts a dominatee message, specifying itself as a dominatee. When a white node receives a dominatee message, it colors itself yellow. The yellow node whose id is the minimum among all of its one-hop yellow neighbors becomes a connector, colors itself black and broadcasts a dominator message.

-Cardei *et al.*, [59]: A 2-phase distributed algorithm is that the root does not need to wait for the arrival of the COMPLETE messages from the furthest nodes. The leader colors itself black and broadcasts a message DOMINATOR. Each White node receiving a DOMINATOR message the first time colors itself gray and broadcasts a message DOMINATEE. A white host receiving at least one DOMINATEE message becomes active. An active white nodes with highest ($d^*$, id) among all of its active white neighbors will color itself black and broadcast a message DOMINATOR, where $d^*$, denoted as effective degree, is the total number of white neighbors. Each gray node will broadcast a message NUMOFBLACKNEIGHBORS whenever it detects that none of its neighbors is white. The root initiates the connecting phase right after it receives NUMOFBLACKNEIGHBORSs from all of its neighbors. The black nodes generated in the first phase are connected through a Steiner tree. The main idea is to pick those gray nodes which can connect to many black neighbors as connectors. This process begins from the leader to the furthest node of the leader. The time complexity and message complexity are O(n) and O(n $\Delta$ ) respectively.

-Kim *et al.*, [60]: Proposed a CDS algorithm which is called Timer-based Energy aware Connected Dominating Set protocols. This algorithm has two phases. In the first phase an initiator is elected based on order pair (Energy, Degree) or (Degree, Energy). In the second phase, the CDS is constructed rooted from the initiator. In this phase, each node has an timer (DSTimer) whose time is determined by a function of uncovered neighbors and energy level. The initiator first broadcasts its status, and all the neighbors start DSTimers. When the DSTimer of one node expires, this node become a dominator node and also broadcasts its status. At the same time, one dominatee node will become a dominator whenever it finds there exist at least two unconnected dominator neighbors.

-Zeng *et al.*, [61]: Proposed a two phase algorithm. In the first phase, the initiator colors itself black and broadcasts a BLACK message. A white neighbor that receives the BLACK becomes grey and broadcasts GREY message. Upon receiving a GREY message, a white node broadcasts an INQUIRY message toward its neighbors to inquire their stats and weights. If this white node has the highest weight among all neighbors based on the replies of INQUIRY, it becomes black and also broadcasts a BLACK message. In the second phase, a dominating tree is used to form a CDS. However, the different between [61] with [47] is that a localized approximation of minimum spanning tree is used. They take a greedy approximation algorithm that every MIS node selects the non-MIS node with the highest weight which are equivalent to 2-hop to interconnect two or more MIS nodes, as a connector.

-Funke *et al.*, [62]: This algorithm also consider the message interference as well. They use the D2-coloring algorithm to find the time slot for each node, and then a CDS is formed by a distributed tree-based algorithm. The execution of this algorithm is divided into rounds. Each round consists of three phases. Initially, a leader is chosen by using a leader election algorithm and marks itself red. All other nodes are colored white. In each round red node u with minimum ID among its red neighbors joins a set I by sending APPLY-MSG. Then u marks itself black and broadcasts a CONFIRM-MSG(black) message. After that the colors of the relevant nodes are update accordingly. For example, all nodes that are the neighbors of nodes in I mark themselves blue. Therefore, whenever a red node joins I, its blue parent marks itself grey as a connector and joins I as well. Finally, all black nodes and grey nodes form a CDS.

### 4.3.2. Multiple Initiators Algorithms

-Parthasarathy *et al.*, [63]: Proposed two distributed algorithms in [63]. The first one has three stages. The first stage involves D2-coloring the nodes using a list of c colors. After that, all nodes in the network have a valid D2-coloring. An MIS is constructed in the second stage in which all nodes belonging to color i attempt to join the MIS during slot i. In the third stage, all MIS nodes exchange its information to all 3-hop neighbors by using PHASE-1 and PHASE-2 messages. Therefore, each MIS nodes can find the connectors to connect every other MIS node in its D3-neighborhood by using PHASE-4 and PHASE-5 messages. The main idea of the second algorithm is the same as the first one. However, the first one use D2-coloring to reduce the collision.

-Li *et al.*, [64]: Proposed a one phase distributed algorithm, r-CDS, with constant performance ratio 172. The difference is the method to choose the MIS. In r-CDS, a novel variable r is introduced which is the number of 2-hop-away neighbors - the number of 1-hop-away neighbors. The node with highest (r, deg, ID) among its neighbors is claimed as MIS node and broadcasts a BLACK message. Upon receiving a BLACK message from its neighbors v, a white node u marks itself grey and broadcasts GREY message. Upon receiving a GREY or BLACK message, a white node decrements its effective degree by 1. If a grey node finds that there exist two unconnected black nodes in its 2-hop neighbors, it marks itself as a connector. Although authors do not give any theocratical analysis that why r should be used, the simulation results show that r-CDS can obtain smaller CDS than others.

-Cheng *et al.*, [58]: In this algorithm, the node with smallest ID becomes an MIS node which is marked red. Then a node becomes black when it cannot be dominated by any red node. Therefore, all red and black nodes form an MIS. After changing at most two hop neighbor information, the connectors also are specified. They use different colors for different kinds of connectors. For example, blue color is used to specify a connector which has at least two MIS neighbors. Yellow color is used to specify a connector to connect two MIS nodes that are separated by 3 hops.

## 5. Conclusions

We briefly introduce the background knowledge of WSNs and the topology control techniques in WSNs. Since sensor nodes are tightly constrained in terms of energy, processing, and storage capacities, restricting topology in WSNs is very challenging due to these inherent characteristics that distinguish WSNs from other wireless networks. Due to such difference, many new algorithms have been proposed for controlling topology in WSNs. Connected Dominating Set (CDS) based topology control which is one kind of hierarchical methods has received more attention to reduce redundant and unnecessary communication overhead. Having such a CDS reduces network topology by restricting the main communication tasks to the dominators only. Then, we summarize the CDS constructing algorithms, especially; we focus on CDS-based including both centralized and distributed algorithms, for how to construct CDS. The size of CDS obtained from the centralized algorithms is smaller than those from distributed algorithms. However, the topology of the whole network should be known in advance for those kinds of centralized algorithms. Distributed algorithms are well suitable for large scale WSNs, and they are very energy and time efficient. According to their characteristic, those distributed algorithms can be classified to prune-based and MIS-based. The prune-based algorithm is efficient to maintain a CDS in dynamic network. However, a MIS-based algorithm can obtain a CDS with small size. Having such a CDS simplifies routing by restricting the main routing tasks to the dominators only. Unfortunately, a CDS only preserves 1-connectivity and it is therefore very vulnerable. Fault tolerance and routing flexibility are necessary for routing since nodes in WSNs are prone to failures and nodes may have mobility and turn on and off frequently. Thus, it is important to maintain a certain degree of redundancy in a CDS.

**Table 1. Comparison of the Presented CDS Algorithms**

| **Algorithms** | **Type** | **Time Complexity** | **Message Complexity** | **Performance ratio** |
|---|---|---|---|---|
| Marathe *et al.* | Centralized | - | - | 10 |
| Guha Khuller (1) | Centralized | - | - | $2(1+H(\Delta))$ |
| Guha Khuller (2) | Centralized | - | - | $\ln\Delta+3$ |
| Ruan *et al.* | Centralized | - | - | $3+\ln(\Delta)$ |
| Wu *et al.* | Prune -based | $O(\Delta^3)$ | $\Theta(n)$ | n/2 |
| Chen *et al.* | Prune -based | - | - | - |
| Das *et al.* | Single Initiator | $O(n^2)$ | $O(n^2)$ | $3H(\Delta)$ |
| Wan *et al.* | Single Initiator | O(n) | O(nlogn) | 8 |
| Cardei *et al.* | Single Initiator | O(n) | $O(n\Delta)$ | 8 |
| Cheng et al. | Single Initiator | O(n) | O(nlogn) | 8 |
| Kim *et al.* | Single Initiator | - | - | - |
| Zeng *et al.* | Single Initiator | O(n) | O(n) | 7.6 |
| Funke *et al.* | Single Initiator | O(n) | $O(n^2)$ | 6.91 |
| Parthasarathy *et al.* (1) | Multiple Initiators | $O(\Delta \log^2 n)$ | $O(n \log^2 n)$ | - |
| Parthasarathy *et al.* (2) | Multiple Initiators | $O(\log^2 n)$ | O(nlogn) | - |
| Li *et al.* | Multiple Initiators | $O(\Delta)$ | $O(n\Delta^2)$ | 172 |
| Cheng *et al.* (2) | Multiple Initiators | O(n) | O(n) | 147 |

## References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “A Survey on Sensor Networks”, IEEE communication Magazine, vol. 40, no. 8, **(2002)**, pp. 102-114.
[2] G. Wang, G. Cao and T. La Porta, “Movement-assisted sensor deployment”, IEEE INFOCOM, Hong Kong, China, **(2004)** March 7-11.
[3] Y. Zou and K. Chakrabarty, “Sensor deployment and target localization based on virtual forces”, IEEE INFOCOM, San Francisco, USA, **(2003)** April 1-3.
[4] L. Zhou and J. Z. Haas, “securing ad hoc networks”, IEEE Network Magazine, vol. 13, no. 6, **(1999)**, pp. 24-30.
[5] F. Stajano and R. Anderson, “The resurrecting duckling: security issues for ad-hoc wireless networks”, AT&T software symposium, Middletown, NJ, USA, **(1999)** October.
[6] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi and D. J. Rosenkrantz, “Simple heuristics for unit disk graphs Networks”, Networks Journal, vol. 25, no. 2, **(1995)**, pp. 59-68.
[7] M. Zuniga and B. Krishnamachari, “Analyzing the transitional region in low power wireless links”, IEEE SECON, Santa, Clara, CA, USA, **(2004)** October.
[8] G. Zhou, T. He, S. Krishnamurthy and J. A. Stankovic, “Impact of radio irregularity on wireless sensor networks”, ACM Mobisys, Boston, Massachusetts, USA, **(2004)** June 6-9.
[9] A. Cerpa, J. L. Wong, M. Potkonjak and D. Estrin, “Temporal properties of low power wireless links: Modeling and implications on multi-hop routing”, ACM MobiHoc, Urbana-Champaign, IL, USA, **(2005)** May 25-28.
[10] A. Cerpa, L. J. Wong, L. Kuang, M. Potkonjak and D. Estrin, “Statistical model of lossy links in wireless sensor networks”, IPSN, Los Angeles, CA, USA, **(2005)** April 25-27.
[11] Y. Liu, Q. Zhang and M. L. Ni, “Opportunity-Based Topology Control in Wireless Sensor Networks”, IEEE TPDS, vol. 21, no. 9, **(2010)**.
[12] R. M. Garey, S. D. Johnson, “Computers and Intractability: A guide to the theory of NP-completeness”, Freeman, San Francisco, **(1978)**.
[13] A. Ephremides, E. J. Wieselthier and J. D. Baker, “A design concept for reliable mobile radio networks with frequency hopping signaling”, Proceedings of the IEEE, vol. 75, no. 1, **(1987)**, pp. 56-73.
[14] B. Das and V. Bharghavan, “Routing in Ad-Hoc Networks Using Minimum Connected Dominating Sets”, International Conference on Communications, Montreal, Canada, **(1997)** June 12-12.
[15] J. Wu, M. Gao and I. Stojmenovic, “On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks”, Proc. of International Conference on Parallel Processing (ICPP), Valencia, Spain, **(2001)** September.
[16] B. Das, R. Sivakumar and V. Bharghavan, “Routing in Ad Hoc Networks Using a virtual backbone”, IEEE IC3N, Las Vegas, Nevada, USA, **(1997)** September 22-25.
[17] B. Das, R. Sivakumar and V. Bharghavan, “Routing in Ad Hoc Networks Using a Spine”, International Conference on Computers and Communications Networks, Las Vegas, NV, USA, **(1997)** September 22-25.
[18] R. Sivakumar, B. Das and V. Bharghavan, “An Improved Spine-based Infrastructure for Routing in Ad Hoc Networks”, IEEE Symposium on Computers and Communications, Athens, Greece, **(1998)** June.
[19] Y-L. Chang and Ch. Hsu, “Routing in wireless/mobile ad-hoc networks via dynamic group construction”, Mobile Networks and Applications, vol. 5, no. 1, **(2000)**, pp. 27-37.
[20] A. B. McDonald and T. Znati, “Mobility-based framework for adaptive clustering in wireless ad hoc networks”, IEEE Journal on Selected Areas in Communications, vol. 17, no. 1, **(1999)**, pp. 1466-1487.
[21] R. Sivakumar, P. Sinha and V. Bharghavan, “CEDAR: a core-extraction distributed ad hoc routing algorithm”, IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, **(1999)**, pp. 1454 -1465.
[22] P. Sinha, R. Sivakumar and V. Bharghavan, “Enhancing ad hoc routing with dynamic virtual infrastructures”, IEEE INFOCOM, vol. 55, no. 3, **(2001)**, pp. 1763-1772.
[23] S. Datta and I. Stojmenovic, “Internal Node and Shortcut Based Routing with Guaranteed Delivery in Wireless Networks”, Cluster Computing, vol. 5, no. 2, **(2002)**, pp. 169-178.
[24] S-Y. Ni, Y-Ch. Tseng, Y-Sh. Chen and J-P. Sheu, “The broadcast storm problem in a mobile ad hoc network”, Proc. ACM MOBICOM, Seattle, Washington, USA, **(1999)** August.
[25] X. Cheng, M. Ding, D. -Z. Du and X. Jia, “On the Construction of Connected Dominating Set in Ad Hoc Wireless Networks”, Appear in Special Issue on Ad Hoc Networks of Wireless Communications and Mobile Computing, Washington, USA, **(2004)**.
[26] H. Lim and C. Kim, “Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks”, Proc. 3rd ACM international workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Boston, MA, USA, **(2000)** August 20.
[27] H. Lim and C. Kim, “Flooding in wireless ad hoc networks”, Computer Communications Journal, vol. 24, no. 3-4, **(2001)**, pp. 353-363.

[28] I. Stojmenovic, M. Seddigh and J. Zunic, "Dominating Sets and Neighbor Elimination Based on Broadcasting Algorithms in wireless networks", Proc. IEEE Hawaii Int. Conf. on System Sciences, Los Alamitos, CA, USA, **(1988)** December 6-8.
[29] J. Wu and F. Dai, "Broadcasting in ad hoc networks based on self-pruning", IEEE INFOCOM, San Francisco, USA, **(2003)** April 1-3.
[30] J. Wu and B. Wu, "A Transmission Range Reduction Scheme for Power-Aware Broadcasting in Ad Hoc Networks Using Connected Dominating Sets", IEEE Semiannual Vehicular Technology Conference (VTC2003-fall), Orlando, Florida, USA, **(2003)** October 6-9.
[31] J. Wu, B. Wu and I. Stojmenovic, "Power-Aware Broadcasting and Activity Scheduling in Ad Hoc Wireless Networks Using Connected Dominating Sets", Proc. of IASTED International Conference on Wireless and Optical Communication (WOC 2002), Sydney, Australia, **(2002)** April 21-25.
[32] J. Wu, B. Wu and I. Stojmenovic, "Power-Aware Broadcasting and Activity Scheduling in Ad Hoc Wireless Networks Using Connected Dominating Sets", Wireless Communications and Mobile Computing, a special issue on Research in Ad Hoc Networking, Smart Sensing, and Pervasive Computing, vol. 3, no. 4, **(2003)**, pp. 425-438.
[33] B. Chen, K. Jamieson, H. Balakrishnan and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in Ad Hoc wireless networks", Wireless Networks, vol. 8, no. 1, **(2002)**, pp. 481-494.
[34] M. Ding, X. Cheng and G. Xue, "Aggregation tree construction in sensor networks", Proc. of IEEE VTC, Orlando, Florida, USA, **(2003)** October 6-9.
[35] Y. Xu, J. Heidemann and D. Estrin, "Geography-informed energy conservation for Ad Hoc routing", MobiCom, Rome, Italy
[36] J. Shaikh, J. A. Solano, I. Stojmenovic and J. Wu, "New Metrics for Dominating Set Based Energy Efficient Activity Scheduling in Ad Hoc Networks", Proc. of WLN Workshop (in conjunction to IEEE Conference on Local Computer Networks), Bonn, Germany, **(2003)** October 20-24.
[37] J. Wu, F. Dai, M. Gao and I. Stojmenovic, "On Calculating Power-Aware Connected Dominating Set for Efficient Routing in Ad Hoc Wireless Networks", Journal of Communications and Networks, vol. 5, no. 2, **(2002)**, pp. 169-178.
[38] B. Deb, S. Bhatnagar and B. Nath, "Multi-resolution State Retrieval in Sensor Networks", First IEEE Workshop on Sensor Network Protocols And Applications (SNPA), Alaska, USA, **(2003)** May 11.
[39] B. Liang and Z. J. Haas, "Virtual backbone generation and maintenance in ad hoc network mobility management", IEEE INFOCOM, Tel-Aviv, Israel, **(2000)** March 26-30.
[40] M. Bahramgiri, M. Hajiaghayi and V. S. Mirrokni, "Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multihop networks", Proceedings of ICCCN, Miami, Florida, USA, **(2002)** October 14-16.
[41] P. Basu and J. Redi, "Movement control algorithms for realization of fault tolerant ad hoc robot networks", IEEE Networks, vol. 18, no. 4, **(2004)**, pp. 36-44.
[42] M. Jorgic, I. Stojmenovic, M. Hauspie and D. Simplot-Ryl, "Localized algorithms for detection of critical nodes and links for connectivity in ad hoc networks", Proceedings of 3rd IFIP Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), Bodrum, Turkey, **(2004)** June 27-30.
[43] N. Li and J. C. Hou, "FLSS: a fault-tolerant topology control algorithm for wireless networks", Proceedings of MobiCom, Philadelphia, USA, **(2004)** September 26-October 1.
[44] X-Y. Li, P-J. Wan, Y. Wang and Ch-W. Yi Chi, "Fault tolerant deployment and topology control in wireless networks", Proceedings of MobiHoc, Maryland, USA, **(2003)** June 1-3.
[45] F. V. Fomin, F. Grandoni, A.V. Pyatkin and A. A. Stepanov, "Combinatorial bounds via measure and conquer: Bounding minimal dominating sets and applications", ACM Trans. Algorithms, vol. 5, no.1, **(2008)**.
[46] J. J. Blum, M. Ding, A. Thaeler and X. Cheng, "Handbook of Combinatorial Optimization", Springer, USA, **(2005)**.
[47] P-J. Wan, Kh. M. Alzoubi and O. Frieder, "Distributed construction of connected dominating set in wireless ad hoc networks", Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, New York, USA, **(2002)** June 23-27.
[48] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi and D. J. Rosenkrantz, "Simple heuristics for unit disk graphs", Networks, vol. 25, no. 1, **(1995)**, pp. 59-68.
[49] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets", Algorithmica, vol. 20, no. 1, **(1998)**, pp. 374-387.
[50] L. Ruan, H. Du, X. Jia, W. Wu, Y. Li and K. -I. Ko, "A greedy approximation for minimum connected dominating sets", Theoretical Computer Science, vol. 329, no. 1, **(2004)**, pp. 325-330.
[51] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks", in the 3rd ACM international workshop on Discrete algorithms and methods mobile computing and communications, Seattle, USA, **(1999)** August 20.

[52] J. Wu, B. Wu and I. Stojmenovic, "Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets", Wireless communications and Mobile Computing (WCMC), vol. 3, no. 1, **(2003)**, pp. 425-438.
[53] B. Chen, K. Jamieson, H. Balakrishnan and R. Morris, "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks", Wireless Networks, vol. 8, no. 1, pp. 481-494, **(2002)**.
[54] B. Das, R. Sivakumar and V. Bharghavan, "Routing in ad hoc networks using a spine", International Conference on Computers and Communications Networks, Las Vegas, NV, USA, **(1997)** September 22-25.
[55] B. Das and V. Bharghavan, "Routing in ad-hoc networks using minimum connected dominating sets", International Conference on Communications, Montreal, Canada, **(1997)** June 8-12.
[56] R. Sivakumar, B. Das and V. Bharghavan, "An improved spine-based infrastructure for routing in ad hoc networks", IEEE Symposium on Computers and Communications, Athens, Greece, **(1998)** June 29-July 2.
[57] Kh. M. Alzoubi, P-J. Wan and O. Frieder, "New distributed algorithm for connected dominating set in wireless ad hoc networks", The 35th Hawaii International Conference on System Sciences, Hawaii, USA, **(2002)** January 7-10.
[58] X. Cheng, M. Ding, D. H. Du and X. Jia, "Virtual backbone construction in multihop ad hoc Wireless networks", Wireless Communications and Mobile Computing (WCMC), vol. 6, no. 1, **(2006)**, pp. 183-190.
[59] M. Cardei, X. Cheng, X. Cheng and D-Zh. Du, "Connected domination in multihop ad hoc wireless networks", International Conference on Computer Science and Informatics, Durham, NC, USA, **(2002)** March 8-14.
[60] B. Kim, J. Yang, D. Zhouy and M-T. Sun, "Energy-aware connected dominating set construction in mobile ad hoc networks", Proceeding. 14th International Conference on Computer Communications and Networks, (ICCCN), San Diego, CA, USA, **(2005)** October 17-19.
[61] Y. Zeng, X. Jia and Y. He, "Energy efficient distributed connected dominating sets construction in wireless sensor networks", International Wireless Communications and Mobile Computing Conference (IWCMC), Vancouver, Canada, **(2006)** July 3-6.
[62] S. Funke, A. Kesselman, U. Meyer and M. Segal, "A simple improved distributed algorithm for minimum cds in unit disk graphs", ACM Transactions on Sensor Networks (TOSN), vol. 2, no. 1, **(2006)**, pp. 444-4537.
[63] S. Parthasarathy and R. Gandhi, "Fast distributed well connected dominating sets for ad hoc networks", University of Maryland, College Park, Tech. Rep. CS-TR-4559, **(2004)**.
[64] Y. Li, Sh. Zhu, M. T. Thai and D. -Z. Du, "Localized construction of connected dominating set in wireless networks", NSF International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor and Peer Networks (TAWN04), Chicago, USA, **(2004)** June.

## Authors

**Amir Hassani Karbasi** received his B.S. degree in applied mathematics from University of Tabriz, Tabriz, Iran, in 2010, and M.S. degree in information technology engineering from University of Guilan, Rasht, Iran, in 2013, During the years 2011-2012, he worked as a Network Engineer in Tabriz Institute of Technology, Tabriz, Iran. His research interests include key management in wireless sensor networks, cryptography, attack simulation and prevention, graph theory, computer networks security in general.

**Reza Ebrahimi Atani (corresponding Author)** was born in 1980. He received the B.S. degree in electrical engineering from the University of Guilan in 2002 and the M.Sc and PhD degrees in electronics from Iran University of Science and Technology in 2004 and 2010 respectively. Since 2010, he is an assistant professor in Computer Engineering Department at the University of Guilan. His current research interests include Computer and Network Security, cryptographic hardware and embedded system (CHES), and design of VLSI circuits.