Role-based Hierarchical Self Organization for Wireless Ad hoc Sensor Networks [*]

Manish Kochhal
Dept. of Electrical and
Computer Engineering
Wayne State University
Detroit, MI 48202
manishk@wayne.edu

Loren Schwiebert
Dept. of Computer Science
Wayne State University
Detroit, MI 48202
loren@cs.wayne.edu

Sandeep Gupta
Dept. of Computer Science
and Engineering
Arizona State University
Tempe, AZ 85287
sandeep.gupta@asu.edu

ABSTRACT

Efficiently self organizing a network hierarchy with specific assignment of roles (or tasks) to sensors based on their physical wireless connectivity and sensing characteristics is an important and challenging problem. In this paper, we extend the hierarchical connected dominating set (CDS) construction algorithm, proposed by Jie Wu, to develop our role-based hierarchical self organization algorithm for wireless sensor networks. The resulting self organized sensor network establishes a network-wide infrastructure consisting of a hierarchy of backbone nodes, and sensing zones that include sensor coordinators, and sensing collaborators (or sensing zone members). Our paper identifies the need for organizing a sensor network according to the tasks appropriate for each sensor node based on their initial deployment in the network. Past research in group-based (or hierarchical) sensor networks have ignored the possibility of utilizing both the physical communication and sensing characteristics to assign roles to sensor nodes. We demonstrate the effectiveness of our design, which considers both, through simulations.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Distributed Networks, Network Communications, Network Topology; F.2.2 [**Nonnumerical Algorithms and Problems**]: Routing and layout.

General Terms

Algorithms, Management, Performance, Design.

Keywords

Sensor networks, self organization hierarchy, sensing coordinators, sensing zones, backbone nodes, sensing proximity value (SPV), cumulative sensing degree (CSD), and connected dominating set (CDS).

1. INTRODUCTION

A wireless sensor network (WSN) is an autonomous ad hoc multihop system with a large number of sensor nodes communicating among each other using wireless radios. Wireless sensor networks are made possible by the continuing improvements in embedded sensor, VLSI, and wireless radio technologies [16].

WSNs have many possible applications in the scientific, medical, commercial, and military domains. Examples of these applications include environmental monitoring, smart homes and offices, surveillance, intelligent transportation systems, and many others. Wireless sensor networks randomly deploy tens to thousands of sensor nodes, with each sensor node having integrated sensors, processors, and radios. The sensor nodes then self organize to form an ad hoc network so as to monitor (or sense) target events, gather various sensor readings, manipulate this information, coordinate with each other, and then disseminate the processed information to an interested data-sink or a remote base station. This dissemination of information typically occurs over wireless links via other nodes using a multihop path [8] [1].

One of the crucial design challenges in sensor networks is energy efficiency. This is because individual sensor nodes use a small battery as a power source and recharging or replacing batteries in a remote environment is not trivial. In some cases sensors may also use solar cells that provide limited power. Thus, to achieve a longer network lifetime, one has to tackle energy efficiency at all levels of the sensor network infrastructure. Since the wireless radio is the major energy consumer in a sensor node, systematic management of network communications becomes critical. Sensor network tasks like routing, gathering, or forwarding sensing data to a nearby data sink or a remote base station requires network communication. In order to effectively coordinate these activities, one has to address the problems of sensor network organization and the subsequent reorganization and maintenance. It would be desirable that the initial sensor network organization take advantage of the underlying physical sensing and topological characteristics

---

[*] This material is based upon work supported by the National Science Foundation under Grant ANI-0086020.

WSNA'03, September 19, 2003, San Diego, California, USA.

so as to assign responsibilities to nodes that are best suited to perform certain sensor network duties. This preliminary assignment of duties would facilitate future reorganization to easily adapt to the traffic pattern existing in the sensor network. Thus, the *resilience* of the initial sensor network configuration would decide the energy savings achieved by subsequent sensor network coordination and maintenance.

We propose a new role-based self organization algorithm that extends the hierarchical connected dominating set (CDS) architecture proposed in [24] [25] for scalable operation of the network. We assign routing and sensing roles to sensor nodes depending upon their connectivity and sensing capabilities, respectively. In order to attribute sensing activities, we have developed two sensing metrics known as the sensing proximity value (SPV) and the cumulative sensing degree (CSD) along with other intermediate sensing parameters. We use these sensing metrics to partition the sensor network into several sensing zones. These sensing zones individually act as an aggregate consisting of sensor nodes collaborating to achieve a common sensing objective with a certain *sensing quality of service (sQoS)*. We elect sensing coordinators that act as leaders for their respective sensing zones. The sensor coordinators systematize collaboration among members in the sensing zone. They also support network reorganization and maintenance. Finally, we perform simulations to compare our sophisticated approach with a simple LEACH-based [11] randomized cluster organization.

2. RELATED WORK

The problem of self organization (or self configuration) has been a hot topic of research in wireless ad hoc networks including mobile and sensor networks. Self organization involves abstracting the communicating entities into an easily controllable network infrastructure. Cluster or connected dominating set (CDS), tree, grid, or mesh based organizations are typical. An excellent discussion of various algorithms supporting cluster-based organizations is furnished in [22].

In mobile ad hoc networks, self organization essentially involves maintaining some form of network organization to support routing infrastructure in the presence of random uncontrollable node mobility. Some interesting research in this area include the ZRP protocol [10], and the terminodes based [3] approach. For mobility management, ZRP uses zones that are similar to clusters whereas the terminodes based approach uses the concept of self organized virtual regions. Routing in both these approaches involves two different schemes, a proactive routing scheme for nodes within a local virtual-region or zone, and a reactive scheme for nodes located in remote virtual-regions or zones. Since in mobile-ad hoc networks the availability of the network is dependent on each user's discretion, an incentive for cooperation by way of virtual money called nuglets is employed in terminodes.

Sohrabi, Pottie, *et al.* [20] [21] [6] have introduced in detail the problem of self organization in wireless sensor networks. They point out the differences in various related wireless network models (e.g. MANET, Cellular networks, Bluetooth, and HomeRF) and the WSN w.r.t. the desired network performance objectives. [6] gives a detailed description of the top-level design components of a self organization protocol for WSN. [21] [20] propose a self organization protocol for WSN that trades available network bandwidth in order to save energy. The self organizing algorithm includes a suite of protocols designed to meet various phases of network self organization. Their self organizing algorithm forms a flat topology as opposed to our approach of forming a hierarchy of backbone nodes that also connect sensing zones based on the sensor network connectivity and sensing characteristics. Their protocol forms an on-demand minimum-hop spanning tree to a central node (CN) elected among neighboring (near-field or far-field) sensors that sense environmental stimuli. In our case we already have groups of sensors organized as approximate stand-alone sensing zones with little or no overlap among neighboring sensing zones. This initial organization can then be reorganized easily to adapt to the changing sensor network traffic.

Subramanian and Katz [23] propose a self configuration architecture that leads to a hierarchical network with address auto-configuration and a number of other useful properties. Their self organizing algorithm lists four phases of operation. These are the discovery phase, organizational phase, maintenance phase, and self reorganization phase. Chevallay *et al.* [5] build on this architecture by proposing a hierarchical cluster-based organization of a network of wireless sensors. The clusterhead election is based primarily on the energy level and processing capability of each sensor node. Our proposal differs from the above as we use both sensing-based and connectivity-based metrics to elect sensor coordinators and backbone nodes. Moreover, our proposal addresses mainly the second phase (i.e. the organizational phase); the third and the fourth phases are left as future work.

Mirkovic *et al.* [15] organize a large-scale sensor network by maintaining a dynamic multicast tree-based forwarding hierarchy that allows multiple sinks to obtain data from a (sensor) source. Their algorithm does not need a globally unique ID for every participating sensor node. Thus address auto-configuration is not one of their self organization objectives as it is for [23] and [5]. In our proposal we assume the existence of a globally unique ID for each sensor node.

Krishnan and Starobinski [12] present two algorithms that produce clusters of bounded size and low diameter by having nodes allocate local *growth budgets* to neighbors. Unlike the expanding ring approach [17], their algorithms do not involve the initiator (or clusterhead) in each round and do not violate the specified upper bound on the cluster size at any time, thus having a low message overhead as compared to [17]. In our approach we use localized communication among neighbors during the self organization phase. In order to limit the membership of the sensing zones as well as the number of sensing zones, our protocol uses two specified minimum and maximum sensing zone membership limits. However, in the final stages of the algorithm, orphan nodes will join any nearest neighboring sensor coordinator or a sensing zone member. This is done to cover the maximum possible number of nodes in the organized hierarchy.

Meguerdichian *et al.* [14] [13], have formulated the exposure and coverage properties of sensor networks using computational geometry based techniques like the Voronoi diagram and the Delaunay Triangulation. The sensor models used in their analysis include two concepts. One is that the sensing ability diminishes with increasing distance. Second is that noise bursts diminish the sensing ability but this effect of noise can be minimized by allowing sensors to sense over longer time periods (exposure). However, a distributed and localized algorithm measuring coverage and exposure of a sensor node or of a region of WSN deployment is not discussed.

Tian and Georganas [7] propose to increase the system lifetime and at the same time preserve original sensing coverage by using a node scheduling scheme that turns off redundant sensor nodes in a network of wireless sensors. This scheme allows nodes in the network to autonomously turn themselves on/off using local neighbor information. This local neighbor information is used to find out if a node needs to be ON so as to cover a region of some neighbor that is not being covered by any other neighbors. Sensing coverage determination employs geometrical techniques that calculate shared neighboring sectors modelled from a circular sensing region with central angles being interpreted from the AoA (Angle of Arrival) of incoming signals. AoA measurements need a multi-directional antenna, which is still sophisticated hardware in sensor technology. Nodes that find themselves redundant w.r.t. sensing coverage advertise status advertisement messages (SAM) to neighbors. This SAM advertisement employs a random back-off timer to avoid having all neighbors turning themselves off, in turn leaving a blind spot. This randomization may sometimes lead to a situation where neighboring nodes will come to know of a blind spot only after some time has elapsed.

Slijepcevic and Potkonjak [19] propose a heuristic that organizes the sensor network by selecting mutually exclusive sets of sensor nodes that together completely cover the monitored area. However our sensing zones are based on cumulative (rather than individual) sensing coverage. In this way, for any target event (genuine or spurious event) we have sensing zones that either report individually or collaboratively with some degree of fault tolerance. Since the sensors are deployed randomly rather than deterministically, there may be regions of the monitored area that are covered by a higher number of sensors. We believe that this redundancy in sensing coverage could be utilized to save energy if the energy required for continuous sensing is comparable to that consumed in message transmission. A thorough analysis of increased energy savings achieved by utilizing redundant sensing coverage w.r.t. the overall lifetime of the network and the fault tolerant sensing is needed.

3. ROLE BASED HIERARCHICAL SELF ORGANIZATION

3.1 Design Philosophy

Wireless sensor network operations include data discovery, which is achieved by way of sensing application specified target events. Additionally, the sensor network needs to process this information in a distributed manner and then forward it to an interested data sink or a remote base station. These sensor network tasks can be managed individually by a sensor node or they may be collaborated upon by several nodes simultaneously. An intuitive analysis of the sensor network activities leads to mapping tasks to roles as follows:

1. Sensing Collaborator
2. Sensing Coordinator
3. Routing or backbone nodes

Since all sensor nodes in the network are essentially deployed to collaboratively sense target events, all nodes assume the role of a sensing collaborator. However, some of these nodes are also requested to assume the role of either a routing node or a sensing coordinator. The routing role as the name suggests, supports a network-wide routing functionality for both application specific sensing queries and the sensing data gathered by the sensors. A sensing application may need to query for a target event in a certain interested region of sensor network deployment. On the other hand, target events sensed by some sensors in a certain region may need to be solicited by some other sensors acting as data sinks or sensor coordinators. These sensor coordinators not only take the responsibility of coordinating the sensing activities in their neighboring region (also known as a sensing zone) but also aggregate and forward the information to any remote data sink or the base station. The task of coordination is not a simple one and it is also not a short term job. In order to provide instantaneous sensing and reporting capability (dependent upon sensing applications) each sensor coordinator may need to systematically rotate its responsibilities transparently among neighboring nodes without much communication overhead. A hierarchical network organization would also be needed to provide scalability for a dense deployment of a large number of sensors.

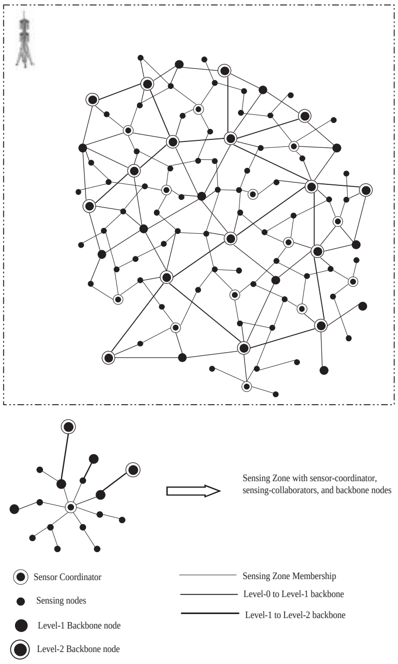


Figure 1: Role based hierarchical self-organization

Figure 1 illustrates these design principles of our proposal. A two-level CDS hierarchy is shown to support routing infrastructures throughout the network. One advantage of having multiple levels of hierarchy is that as the hierarchy increases fewer nodes are involved in routing, which leads to paths with fewer hops within the network. Depending upon the requirements of the sensing application as well as the topology of the sensor network deployment, one may be able to provide certain levels of guarantees with respect to routing queries or routing data to the base station. Thus, it would be desirable that the reorganization phase of the self configuring algorithm preserve the lifetime of these higher level hierarchy nodes and hence preserve the capability of providing prompt delivery of services in the face of changing sensor network traffic patterns.

In the next section, we will be briefly outlining the CDS construction algorithm proposed in [24] [25].

3.2 CDS based Network Organization

Cluster-based organizations partition the entire network into groups (or clusters). Each cluster is formed by selecting some nodes based upon some quality metric (say connectivity or distance) [4] [2] as cluster members and a group leader (known as the cluster-head) is also selected (using some metric, say maximum energy) to manage that cluster. These cluster-heads, when connected, form a virtual backbone (or spine [18]) or a set of connected dominating nodes. Related to clustering is the problem of finding a minimum connected dominating set ($MCDS$) of the nodes. An $MCDS$ satisfies two properties: (1) each node is either a backbone node or is connected (one hop) to a backbone node, and (2) the backbone nodes are connected. There are several algorithms [9] available in the literature that engineer virtual backbone based network configurations satisfying the $MCDS$ properties.

In our proposal, we will be using the distributed localized algorithm for constructing a hierarchical connected dominating set (CDS) presented in [24] [25]. Our main reason for selecting this algorithm is its inherent distributed and simple nature. Ideally, it requires only local information and a constant number of iterative rounds of message exchanges among neighboring hosts. The algorithm for CDS formation involves two processes, the marking process and the dominating set reduction process. We also assume the following network model.

3.2.1 Network Model

We represent the ad hoc wireless network by a simple graph $G = (V, E)$, where V represents a set of wireless nodes and E represents a set of edges. An edge between host pairs (v, u) indicates that both hosts v and u are within each others wireless transmitter ranges. We assume that all the wireless nodes are homogeneous, i.e. their wireless transmitter ranges are the same. In other words, if there is a edge $e = (v, u)$ in E, it indicates that u is within v's range and v is within u's range. Thus, the corresponding graph is an undirected graph also known as a unit graph, in which connections to hosts are determined by their geographical distances.

3.2.2 Marking Process

The marking process as described in [24] [25] is a localized algorithm in which hosts only interact with others only in a restricted vicinity. Each host performs exceedingly simple tasks such as maintaining and propagating information markers. Collectively, these hosts achieve a desired global objective, i.e. finding a small connected dominating set. The marking process marks every vertex in a given connected and simple graph, $G = (V, E)$. $m(v)$ is a marker for vertex $v \in V$, which is either T (marked) or F (unmarked). Initially, it is assumed that all the vertices are unmarked and that each vertex v has its open neighbor set as $N(v) = \{u \mid (v, u) \in E\}$. The marking process can thus be summarized as follows:

1. Initially, assign marker F to each v in V.
2. Each v exchanges its open neighbor set $N(v)$ with all its neighbors.
3. Each v assigns its marker $m(v)$ to T if there exist two unconnected neighbors.

In the example depicted in figure $2(a)$, $N(u) = \{v, y\}$, $N(v) = \{u, w, y\}$, $N(w) = \{v, x\}$, $N(y) = \{u, v\}$, and $N(x) = \{w\}$. After step 2 of the marking process, vertex u has $N(v)$ and $N(y)$; v has $N(u)$, $N(w)$ and $N(y)$; w has $N(v)$ and $N(x)$; y has $N(u)$ and $N(v)$; and x has $N(w)$. Based on step 3, only vertices v and w are marked T.

3.2.3 Dominating Set Reduction Process

In order to reduce the connected dominating set (CDS) generated from the marking process, two rules are proposed. Assuming that each vertex v in G' is assigned a distinct ID, $id(v)$, it then calculates its closed neighbor set $N[v]$ as $N[v] = N(v) \bigcup v$.

Rule 1: Consider two vertices v and u in G'. If $N[v] \subseteq N[u]$ in G and $id(v) < id(u)$, change the marker of v to F if node v is marked; i.e. G' is changed to $G' - v$.

Rule 2: Assume that u and w are two marked neighbors of vertex v in G'. If $N(v) \subseteq N(u) \cup N(w)$ in G and $id(v) = \min\{id(v), id(u), id(w)\}$, then change the marker of v to F.

In Figure $2(b)$, since $N[v] < N[u]$, vertex v is removed from G' if $id(v) < id(u)$ and vertex u is the only dominating node in the graph. In $2(c)$, since $N[v] = N[u]$, either v or u can be removed from G'. To ensure one and only one is removed, the node with the smallest ID is removed. Finally, in figure $2(d)$, $N(v) \subseteq N(u) \cup N(w)$ applies. If $id(v) = \min\{id(v), id(u), id(w)\}$, vertex v can be removed from G' based on Rule 2.

In [24] [25], the above rules were extended to include a combination of metrics like energy level (EL) and node degree (ND) along with ID to break ties. In this paper, we will be discussing our proposed sensing-based metrics, which can also be incorporated for the rules for dominating set reduction.

3.2.4 Hierarchical Dominating Sets

The dominating set reduction process can be reapplied on an already reduced dominating set of nodes to generate another set of dominating nodes. This process can be repeated until no further reductions are possible.

3.3 Sensing Attributes or Metrics

3.3.1 Sensing Model

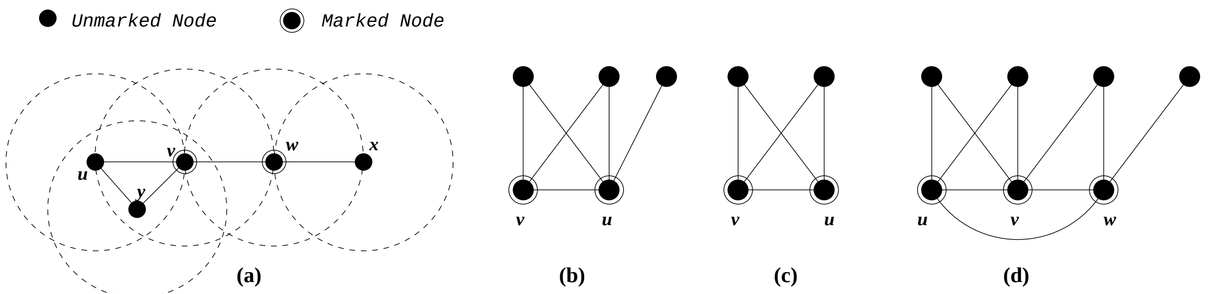
A sensor is a device that produces a measurable response to a change in a physical condition, such as temperature, light, voice, or magnetic field. We assume the same sensing model as that of [14] [13]. We also assume that the sensing region of a sensor is a circle with the sensing range[b] specified as S_R distance units.

3.3.2 Sensing Coverage Approximation

Figure 3 shows three sensors (say, seismic sensors) reporting the detection of a target event (say, an enemy tank) in a battlefield scenario. Since the target i.e. the tank is at a variable sensing proximity or distance (also denoted

---

[a] Example figure reproduced from [24] [25].



(a) (b) (c) (d)

Figure 2: (a) Initial marking process, (b, c, d) 3 examples of dominating set reduction[a]

as SPV) from each of the sensors, the degree of fault tolerance sensing (denoted as CSD) for this event is proportional to the cumulative proximity of the three sensors to the target event i.e. the tank[b]. In order to comprehend the maximum cumulative fault tolerant sensing capability of these three sensors, it may be necessary to calculate the amount of shared coverage overlap between these sensors. The sensing coverage is approximated as a circle with sensing range as its radius. Thus the problem of finding the cumulative sensing coverage gets transformed into finding the overlapping sectors between the neighboring sensors, which is a complicated approximation as discussed earlier. We simplify this approximation by dividing the circular sensing area of each sensor into square sensing cells. The dimension of the sensing cell determines the closeness of the coverage approximation. The sensing cell dimension (denoted as d) is also known as the *application specified sensing accuracy*. We assume that the three sensors, S_1, S_2, and S_3, know the positions of each other. Thus calculating combined sensing coverage would amount to finding the common overlapping sensing cells among the neighbors and also subsequently updating these sensing cell's cumulative sensing proximity values (denoted as $CSPV$) by accounting the relative distance of the neighboring sensors to the cell(s) in question. In other words, $CSD_{S_1,S_2,S_3} = K * F(3, d, overlap)$, where K is some sensing constant for the sensors (in our case, $K = 1$), and F is the function that calculates the cumulative sensing degree (CSD) by accounting for the number of cooperating sensors, the sensing cell dimension (d), and the cumulative sensing proximity value ($CSPV$) of the overlapping sensing cells between them. Figure 4 illustrate the approximation used in our sensing coverage calculations. In the next sections, we will be defining in detail the SPV, $CSPV$, and CSD sensing parameters.

3.3.3 Sensing Proximity Value (SPV)

SPV for a sensing cell denotes how close that cell is to a particular sensor. The SPV may vary from the best value of 1 to some max value, say SPV_{max} (dependent upon sensing range S_R). The lower the value of SPV for a cell, the better its sensing performance or sensitivity. For calculating the

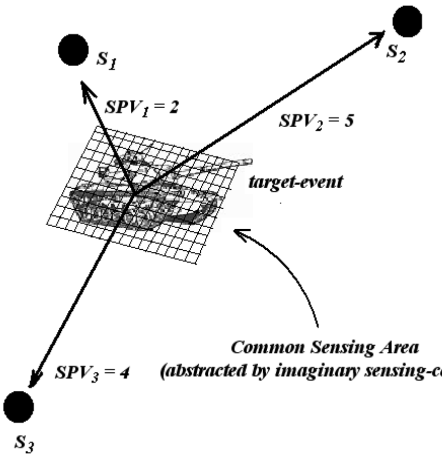


Figure 3: Sensing proximity concept

SPV of a sensing cell i for a sensor node, say n, we need the location of the sensor n i.e. (x_n, y_n), the sensing range, S_R, and the application specified sensing accuracy, d. We calculate the minimum distance between the sensor n and the center of its closest sensing cell and denote it as d_{csmin}.

$d_{csmin} = d/\sqrt{2}$

We also calculate the distance between the sensor node and the center of the square sensing cell i with coordinates as (x_{cell}, y_{cell}) and denote it as d_{cs}.

$d_{cs} = \sqrt{(x_n - x_{cell})^2 + (y_n - y_{cell})^2}$

Finally, SPV is calculated as the ratio of d_{cs} to d_{csmin} and is rounded to the nearest integer.

$SPV_i \approx \lceil d_{cs}/d_{csmin} \rceil$

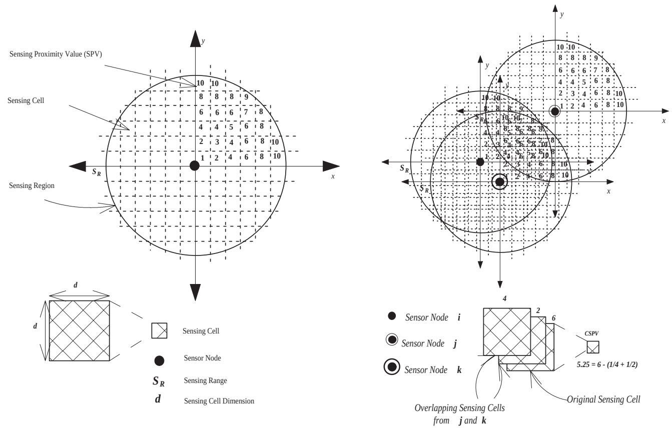
In order to evaluate the cumulative sensing coverage of a shared region commonly monitored by neighboring sensors, we introduce two more sensing parameters, *cumulative sensing proximity value* (CSPV) of a sensing cell and the *cumulative sensing degree* (CSD) of a sensor node.

3.3.4 Cumulative Sensing Proximity Value ($CSPV$)

$CSPV$ for a sensing cell is the cumulative SPV of all the overlapping sensing cells from the neighboring nodes covering that cell. Thus, if SPV_x is the SPV of a sensing cell say x and if n sensing cells having SPV values $SPV_1, SPV_2, SPV_3, ..., SPV_n$ overlap this cell x, then $CSPV_x$ is calculated using the reciprocal reduction technique which

---

[b] The sensing range may depend upon the dimensions of the observed target, e.g. a seismic sensor can detect a tank at a larger distance than it can detect a soldier on foot. For ease of discussion, we assume the sensing range to be same for both the tank and the soldier [19]. We can modify our self organizing algorithm based on the sensing range for a given application.



Sensing Proximity Values

Calculating Cumulative Sensing Coverage

Figure 4: Sensing coverage approximation

is formulated as

$$CSPV_x = SPV_x - \sum_{i=1}^{n} 1/SPV_i$$

Thus, the more sensing cells overlap, the lower the final value of $CSPV$. Moreover, if cells having equal SPV values (say $spv = y$) overlap, then a total reduction of y will be adjusted toward the final $CSPV$ value only if y or more cells overlap. Finally, the $CSPV$ value is always adjusted to be within a range of 1 and SPV_{max}.

3.3.5 *Cumulative Sensing Degree* (CSD)

CSD describes the degree of cumulative fault tolerance sensing for a common area monitored collaboratively by some sensors. We calculate the sensing coverage of a sensor node as the average sum of the $CSPVs$ of all sensing cells covering its sensing area. Ideal sensing coverage would imply that all the $CSPV$ values for the sensing cells of a sensor are 1 (i.e. each sensor node is covered by a maximum possible neighbors), whereas solitary sensing coverage would mean just the average of the sum of $CSPVs$ of a sensor node having no neighbors. Finally, CSD is calculated as percentage coverage and is given by the formula:

$CSD_{sensor} = (1.0-((avg(\sum CSPVs)-ideal)/(solitary-ideal))) \times 100$

3.4 Proposed Self Organization Algorithm

We assume the existence of a neighbor discovery stage that precedes our self organizing algorithm. In this stage, each sensor acquires knowledge of its neighbors and their positions. An example of 15 sensor nodes deployed in a hostile area to detect military tanks is shown in figure 5(a) which after neighbor discovery forms the network shown in figure 5(b). We construct a CDS hierarchy using the hierarchical CDS construction algorithm outlined in sections 3.2.2, 3.2.3, and 3.2.4. We also use the following metrics in order, along with the rules to break the ties. These metrics are energy level (EL), sensing-based metric known as CSD, connectivity-based metric or node degree (ND) and finally ID of the sensor node to break the tie. During the initial marking process, each sensor node exchanges one-hop neighbor information with its neighbors. This results in sensors gathering two-hop neighbor information, and also their corresponding location information. Figure 5(c) shows the CSD of the sensors and also the result of the initial marking process. The percentage CSD value calculated during the initial marking process is used in the subsequent hierarchical dominating set reduction processes. Figure 6(a) shows the 3-level CDS hierarchy formed after performing the dominating set reduction three times.

Our objective of having sensing zones is to have a self sufficient collaborative group of sensor nodes that would need as few sensing inputs as possible from sensors outside the group to reach consensus on any target events sensed. Such a group organization would need to be coordinated by a sensor coordinator. The sensor coordinator is that sensor node that has the maximum cumulative percentage coverage in the neighborhood. This implies that the chances of an event to get detected by a sensing zone in its region coordinated by a node with higher CSD would be higher than any of its neighboring sensing zones. The sensor coordinator would then initiate a consensus among its sensor collaborators to rule out the possibility of a spurious event or noise. This leads us to another interesting $QoS\ Sensing$ (or $sQoS$) metric for a sensing zone that can be specified as the minimum percentage coverage or CSD of a certain region of WSN deployment. Figure 6(b) illustrates all these sensing zone concepts.

From figure 6(a), we can see that as we go up the CDS hierarchy the number of dominating nodes reduces. We can naively select the dominating nodes at any higher level of the hierarchy to act as sensor coordinators. But as mentioned earlier, dominating nodes in the CDS hierarchy are essentially used as backbone nodes to route application specific sensing queries to the sensors and/or sensing data from the sensors to a data-sink. An intuitive suggestion is to select sensor coordinators from the lowest level of the CDS hierarchy i.e. level-0. We also know that the hierarchical dominating reduction process is a recursive process that uses marked nodes (or dominating nodes or backbone nodes) from the previous level to form the next level of the hierarchy. This means that our suggestion to use level-0 marked nodes as sensor coordinators has to be revised to include only those level-0 marked nodes that get removed during the dominating set reduction process to form level-1. In other words, our self organization algorithm chooses sensor coordinators from level-0 marked nodes (but level-1 unmarked) as these nodes will not be acting as backbone nodes in upper levels of the CDS hierarchy. We note one more advantage of selecting sensor coordinators from level-0 marked nodes (and level-1 unmarked) they are the majority of available nodes in comparison to any other levels. Thus we have a bigger pool of nodes from which we can select sensor coordinators. In order to have sensor coordinators at level-0. (i.e. nodes with maximum percentage CSD) the algorithm uses an adaptive sensing-based metric. This means that during level-0 marking (or dominating set reduction) we eliminate those nodes (during tie breaker stage) that have the lower percentage CSD. This results in all level-0 dominating (or marked) nodes as nodes that have maximum percentage CSD within their one-hop neighborhood. Finally, from level-1 onwards, we eliminate those nodes that have the higher percentage CSD, which again leaves higher percentage CSD marked nodes at level-0 (but level-1 unmarked). The overall effect is

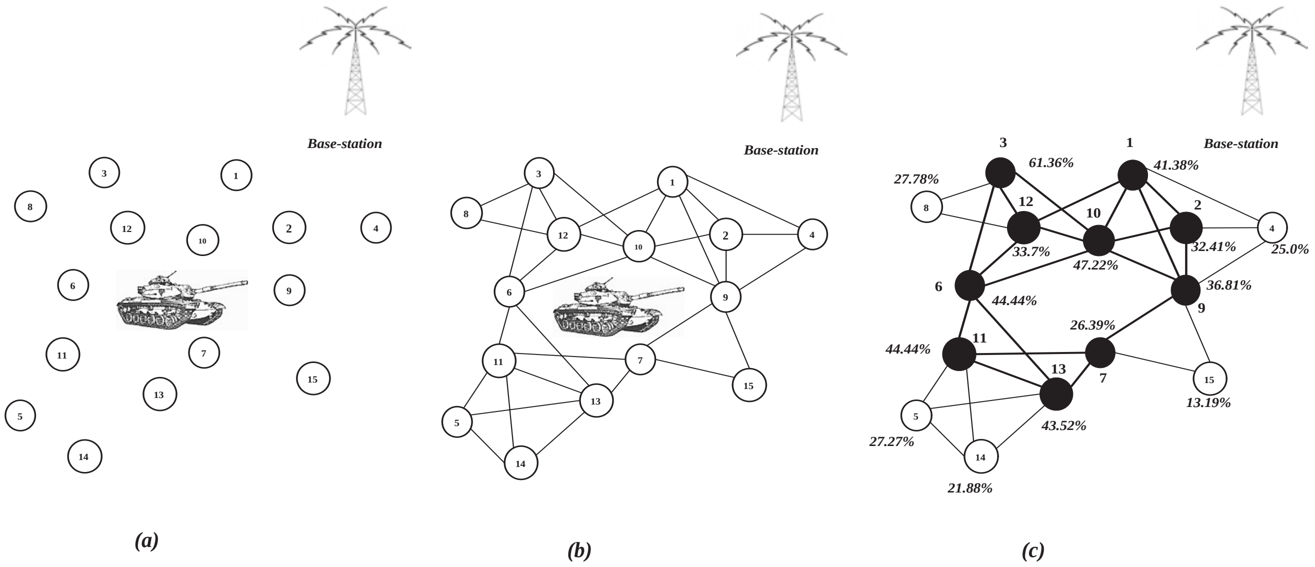


Figure 5: (a) Example 15 sensor nodes deployment, (b)WSN after $NEIGHBOR_{DISCOVERY}$ stage, and (c)WSN after marking stage

that we make sure that during both level-0 and level-1 marking, all the unmarked nodes at level-1 (but level-0 marked) have the maximum possible percentage CSD. This leaves a larger crowd of eligible sensor coordinators at level-0 nodes who are not dominating at any other higher level. Figure 6(c) shows the selected sensor coordinators.

In order to reduce from a list of probable sensor coordinators, we select only those nodes that have a higher percentage CSD than their level-0 marked (and level-1 unmarked) neighbors. If there is a tie, then we break it by the number of marked level-1 neighbors an eligible sensor coordinator may have. If there is still a tie, then we use sensor node ID as the final tie-breaker. An eligible sensor coordinator that passes the above three selection criteria would then advertise to its neighbors with its maximum percent CSD value. Sensors hearing this advertisement join the nearest soliciting sensor coordinator. However, in order to limit the overhead of sensing zone coordination and maintenance, we limit the group membership within a certain minimum and maximum number of sensor collaborators (or sensing zone members). Sensing zones with less than the specified minimum sensing zone membership will *merge* with neighboring *accepting sensing zones*. The reverse case applies for zones having membership larger than the maximum. In this case, sensor coordinators of these crowded sensing zones will ask distant members to find another neighboring sensing zone. These dismissed sensor nodes will join their nearest neighboring accepting sensing coordinator or sensing zone member. A sensor coordinator would *accept* a node as its zone member only if it has some space left to accommodate that node. Finally, all those nodes that were refused zone membership by their respective neighboring sensor coordinators due to zone size problems are considered as *orphan nodes*. Similarly, all those nodes who could not find any neighboring sensor coordinators due to a limited number of neighbors (or with sparse connectivity) will also consider themselves as *orphan nodes*. These orphan nodes will finally join any closest sensor coordinator or sensing zone member who would ultimately acquiesce to their join demands.

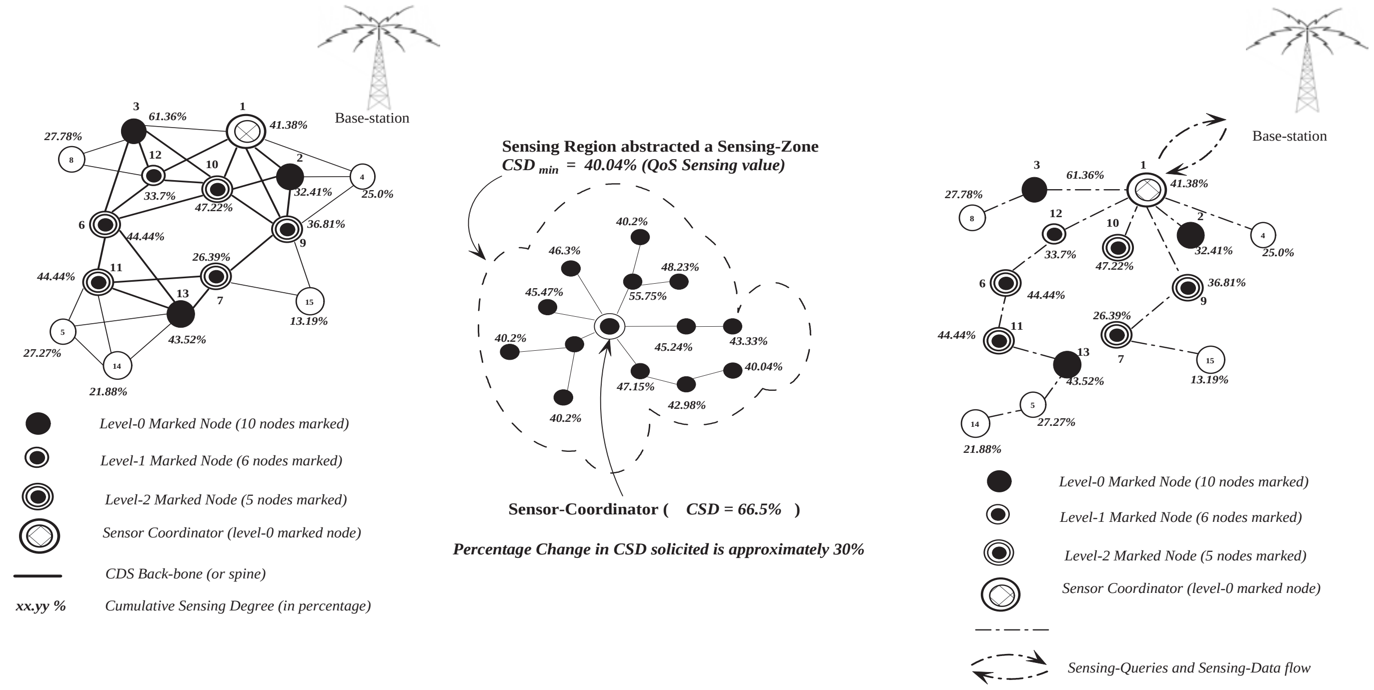
4. SIMULATION

The role-based hierarchical self organization protocol has been simulated using Java (JDK 1.3). The simulator can also be used to view the topology generated by the initial self organization algorithm. A comparison between Leach and our approach is possible if we have the same number of clusters or sensing zones. To achieve this, the simulator takes the number of sensing zones generated from our protocol as input to the cluster based protocol. The simulator assumes no packet collisions. It also assumes that there are no packet errors during transmission and reception. In other words, we assume a perfect wireless channel. Figures 7 and 8 show the results of an example simulation run with the following simulation parameters:

1. Number of nodes = 150.
2. Maximum X, Y boundary coordinates of region of WSN deployment = 400 meters.
3. Maximum wireless radio range and sensing range = 64 meters.
4. Application specified sensing accuracy (d) = 8 meters.

We have performed 100 simulation runs on four different sets of topologies:

1. 100 nodes in an area of 300 × 300 meters.
2. 300 nodes in an area of 600 × 600 meters.
3. 600 nodes in an area of 1400 × 1400 meters.
4. 1000 nodes in an area of 2200 × 2200 meters.



(a) *(b)* *(c)*

Figure 6: (a) CDS hierarchy with sensor coordinator, (b) Sensing zone formation, and (c) Sensing zone organization

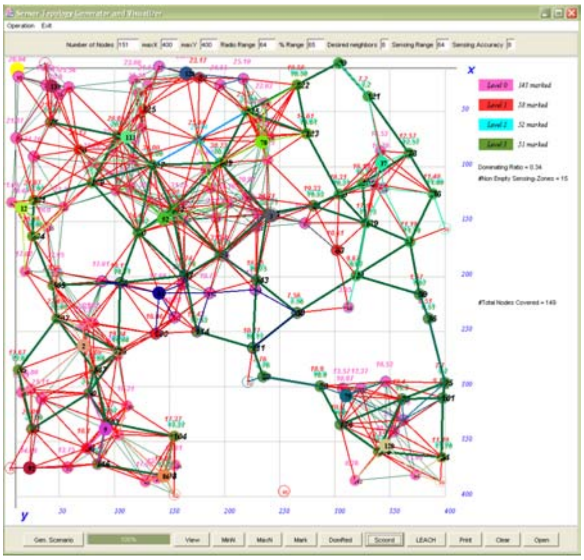


Figure 7: Our self organized infrastructure

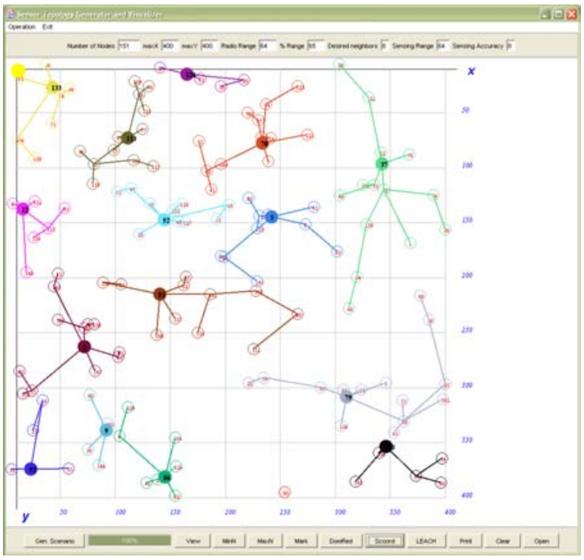


Figure 8: 150 nodes with 15 sensing coordinators

Table 1: Average group leader-member distances ($d = 8$)

Network size	Leach						Our Protocol					
	Mean			Standard deviation			Mean			Standard deviation		
	MaxDist	MinDist	AvgDist	MaxDist	MinDist	AvgDist	MaxDist	MinDist	AvgDist	MaxDist	MinDist	AvgDist
100 (300x300)	80.81	17.02	47.11	9.73	3.05	4.49	74.64	12.02	41.36	6.67	1.93	2.72
300 (600x600)	90.70	18.50	52.14	6.93	2.46	3.16	80.65	12.53	43.80	6.61	1.42	2.48
600 (1400x1400)	228.24	27.98	121.28	21.57	2.85	9.08	144.81	17.52	73.94	10.65	1.89	4.42
1000 (2200x2200)	443.24	32.54	219.53	58.27	3.43	23.89	150.16	18.52	76.83	11.75	1.73	5.17

Table 5: Average group membership sizes ($d = 8$)

Network size	Average Membership			
	Leach		Our Protocol	
	Mean	Std dev	Mean	Std dev
100 (300x300)	10.66	1.11	10.82	1.07
300 (600x600)	10.93	0.78	10.72	0.65
600 (1400x1400)	21.18	2.74	13.35	1.01
1000 (2200x2200)	46.28	9.23	12.80	0.86

For all topologies, we have set the radio range and the sensing range to 64 meters. The minimum and maximum sensing zone (or cluster) membership size is set to 4 and 12, respectively. Finally, the application specified sensing accuracy or the sensing cell dimension (d) is set to values 8, 12, and 16 for the above simulation scenarios. Tables 1 – 5 compare our protocol with the Leach-based protocol.

During the analysis of the simulation results, we will be using the terms *clusters* or *sensing zones* or *groups* interchangeably. From table 1, it can be seen that our self organizing protocol organizes sensors into sensing zones with less distance variation as compared to Leach. Moreover this distance variation becomes more pronounced as the topology becomes more sparse with an increasing number of nodes deployed in a larger area. Since the Leach protocol selects the clusterheads randomly rather than deterministically, most of the times it results in suboptimal selection of clusterheads. This in turn results in situations where sensors having distant soliciting clusterheads will extend their radio range in order to join any nearby less crowded clusters. It should be noted that the objective of any self organizing algorithm is to abstract the random topology into an easily controllable network infrastructure. Thus, any group based self organizing algorithm will try in a best effort manner to include each sensor node in at least one group. In pursuing such a goal, Leach ends up having larger group memberships as compared to our approach. This can be clearly understood from table 5 where the group size for our approach remains within 13 members whereas for Leach it may be up to 55 members as network size increases from 100 to 1000 sensors.

Tables 2, 3, and 4 analyze the effectiveness of our organization and Leach w.r.t. the cumulative sensing degree (CSD) metric. Table 2 shows the mean of the maximum, minimum, and average CSD of a sensor node assuming that it has all its neighbors in its group. We compare these *static* CSD values with the *current* CSD values obtained after the groups have been formed by the self organizing algorithm. It can be seen from table 4 that our protocol always results in sensor nodes retaining most of their static CSD values, whereas Leach results in an appreciable loss in node CSD due to suboptimal selection of clusterheads. However this difference is also negligible due to the fact that in Leach we also have orphan nodes select any nearby distant suboptimal clusterhead in order to be registered in some group. Table 3 shows the mean of the average CSD values of the clusterheads or the sensor coordinators (also referred to here as *leaders*). It can be clearly seen that due to deterministic leader selection our self organizing protocol has higher average leader CSD values as compared to Leach. One interesting result in all these tables is the dependency of our CSD approximation on the sensing cell dimension (or application specified sensing accuracy) i.e. d. We can easily conclude that lower values of d yield a better CSD approximation.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we present a role-based hierarchical self organization algorithm for wireless sensor networks. The algorithm groups sensors into sensing zones that are coordinated by a sensor coordinator. We also propose a sensing based metric CSD (known as *Cumulative Sensing Degree*) to form sensing zones. In order to form a hierarchy of back bone nodes we extend the CDS formation algorithm proposed in [24] [25]. The resulting self organized network consists of sensing zones that are connected to each other by the hierarchy of backbone nodes.

The simulation results show how a randomized cluster based organization performs worse as network size increases. Since our algorithm selects sensor coordinators deterministically, we have shorter distances between sensing zone members and the sensor coordinator. From figure 8 it can be seen that there is still some overlap between neighboring sensing zones. The amount of overlap among neighboring sensing zones reflect the actual dependency for events occurring at the border of the sensing zones. In our future work, we propose to include further protocol enhancements or optimizations to reduce such dependency or overlap.

Our algorithm is essentially an initial approach to network organization after neighbor discovery. We propose to extend our work to implement both the maintenance and the reorganization phase (similar to [23]) of a complete self organization algorithm for wireless sensor networks. In most of the previous research literature it is assumed that a hierarchical organization is too static (or rigid) to be reorganized with respect to the ultimate traffic pattern that may run on top of this self organized network architecture. It is also believed that concentrating specific responsibilities on specific nodes will result in such nodes becoming likely points of failure, thus making such a hierarchical network inherently less fault tolerant. We, however, believe that with sufficient network density, both of these problems can be resolved efficiently by systematically rotating roles among neighboring nodes in a localized manner without much overhead.

A detailed analysis of extra energy savings achieved by turning off redundant sensors within a sensing zone is needed. Moreover, we want to develop a MAC protocol to support efficient medium access [20] [21] [6] for sensors in our communication architecture. We also need to analyze our self organization for even larger sensor networks. The energy utilized during initial self organization for forming both the sensing zones and the backbone hierarchy also needs to be analyzed by using real node characteristics and practical sensor network traffic scenarios.

6. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless Sensor Networks: A Survey. *Computer Networks (Elsevier)*, March 2002.

[2] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh. Max-Min D-Cluster Formation in Wireless Ad Hoc Networks. In *Proceedings of IEEE INFOCOM'2000, Tel Aviv*, March 2000.

[3] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J. Hubaux, and J. Le Boudec. Self-organization in mobile ad-hoc networks: the approach of terminodes. *IEEE Communications Magazine*, pages 166–174, June 2001.

[4] G. Chen, F. Garcia, J. Solano, and I. Stojmenovic. Connectivity Based k-hop Clustering in Wireless Networks. In *Proceedings of the IEEE Hawaii Int. Conf. System Science*, Jan 2002.

[5] C. Chevallay, R. E. Van Dyck, and T. A. Hall. Self-organization Protocols for Wireless Sensor Networks. In *Thirty Sixth Conference on Information Sciences and Systems*, March 2002.

Table 2: Average static sensor *CSD*

Network size	*d*	Average static node *CSDs*					
		Mean			Standard deviation		
		MaxCSD	MinCSD	AvgCSD	MaxCSD	MinCSD	AvgCSD
100 (300x300)	8	29.27	14.08	22.54	2.19	1.59	1.46
	12	59.85	32.39	47.64	3.02	2.80	2.44
	16	80.09	47.59	66.43	2.87	3.65	2.56
300 (600x600)	8	26.47	12.96	20.46	1.37	1.08	1.03
	12	55.81	29.97	44.48	2.66	2.04	2.17
	16	75.93	44.11	62.34	2.01	2.42	2.02
600 (1400x1400)	8	12.91	1.52	7.20	0.51	0.34	0.38
	12	29.70	4.89	18.21	1.13	0.88	0.80
	16	45.16	9.14	28.93	1.36	1.26	0.96
1000 (2200x2200)	8	11.27	1.08	5.96	0.53	0.25	0.34
	12	26.59	3.52	15.57	1.15	0.75	0.79
	16	41.42	7.26	25.67	1.21	1.10	1.06

Table 3: Average leader *CSDs*

Network size	*d*	Average Leader CSD			
		Leach		Our Protocol	
		Mean	Std dev	Mean	Std dev
100 (300x300)	8	12.77	1.91	16.53	1.49
	12	28.45	4.47	35.79	2.64
	16	41.52	5.00	51.21	3.86
300 (600x600)	8	11.50	1.06	15.07	0.89
	12	26.46	1.95	33.83	1.67
	16	39.07	2.91	49.14	1.91
600 (1400x1400)	8	4.95	0.68	10.46	0.54
	12	13.17	1.41	24.79	1.04
	16	21.49	2.05	38.05	1.26
1000 (2200x2200)	8	3.13	0.59	9.51	0.48
	12	9.80	1.49	23.06	1.14
	16	16.52	2.32	35.90	1.29

Table 4: Current organized average sensor *CSD*

Network size	*d*	Leach						Our Protocol					
		Mean			Standard deviation			Mean			Standard deviation		
		MaxCSD	MinCSD	AvgCSD	MaxCSD	MinCSD	AvgCSD	MaxCSD	MinCSD	AvgCSD	MaxCSD	MinCSD	AvgCSD
100 (300x300)	8	21.02	4.14	9.98	2.34	1.12	1.29	21.85	5.07	11.60	1.94	0.95	1.14
	12	44.22	11.02	22.89	3.78	2.66	2.92	46.45	12.98	26.48	3.57	1.96	2.24
	16	60.93	17.10	33.88	5.08	3.59	3.68	63.98	20.41	38.95	4.37	2.64	2.72
300 (600x600)	8	18.77	3.09	8.42	1.71	0.63	0.66	20.29	4.01	10.08	1.04	0.57	0.58
	12	40.20	8.36	20.16	2.68	1.33	1.20	43.98	11.10	23.90	2.21	1.17	1.30
	16	57.87	13.31	30.32	3.91	1.84	1.75	61.26	17.87	35.86	2.52	1.60	1.38
600 (1400x1400)	8	9.66	0.16	3.66	0.72	0.11	0.29	11.40	0.78	5.17	0.52	0.18	0.35
	12	23.43	0.47	10.13	1.39	0.29	0.62	26.62	2.69	13.91	1.01	0.51	0.70
	16	36.36	0.95	16.79	1.71	0.48	0.85	40.87	5.25	22.82	1.23	0.90	0.87
1000 (2200x2200)	8	9.21	0.01	2.53	0.85	0.01	0.23	10.08	0.64	4.40	0.48	0.14	0.29
	12	23.02	0.08	7.48	1.21	0.05	0.44	24.43	2.25	12.40	1.04	0.46	0.72
	16	36.30	0.08	13.18	1.86	0.13	0.72	37.93	4.60	20.98	1.13	0.93	0.95

[6] L. P. Clare, G. J. Pottie, and J. R. Agre. Self-Organizing Distributed Sensor Networks. In *Proc. SPIE, Unattended Ground Sensor Technologies and Applications*, volume 3713, pages 229–237, 1999.

[7] D.Tian and N.D.Georganas. A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks. In *Proceedings of ACM Workshop on Wireless Sensor Networks and Applications (WSNA'02), Atlanta*, October 2002.

[8] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *ACM MOBICOM'99*, 1999.

[9] S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. Technical Report 3660, Univ. of Maryland Inst. for Adv. Computer Studies, Dept. of Computer Science, Univ. of Maryland, College Park, June 1996.

[10] Z.J. Haas, M.R. Pearlman, and P. Samar. The Zone Routing Protocol (ZRP) for Ad Hoc Networks. Internet Draft draft-ietf-manet-zone-zrp-04.txt, Internet Engineering Task Force, July 2002.

[11] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In *International Conference on System Sciences*, January 2000.

[12] R. Krishnan and D. Starobinski. Message-Efficient Self-Organization of Wireless Sensor Networks. In *IEEE WCNC 2003*, 2003.

[13] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. Coverage Problems in Wireless Ad-Hoc Sensor Networks. In *IEEE INFOCOM '01*, volume 3, pages 1380–1387, April 2001.

[14] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak. Exposure In Wireless Ad Hoc Sensor Networks. In *ACM SIGMOBILE (Mobicom)*, July 2001.

[15] J. Mirkovic, G. P. Venkataramani, S. Lu, and L. Zhang. A Self Organizing approach to Data Forwarding in Large Scale Sensor Networks. In *IEEE International Conference on Communications (ICC'01), Helsinki, Finland*, June 2001.

[16] G. J. Pottie and W. Kaiser. Wireless Sensor Networks. *Communications of the ACM*, 43(5):51–58, May 2000.

[17] C. V. Ramamoorthy, A. Bhide, and J. Srivastava. Reliable Clustering Techniques for Large, Mobile Packet Radio Networks. In *Proceedings of IEEE INFOCOM'87*, pages 218–226, 1987.

[18] R. Sivakumar, B. Das, and V. Bharghavan. An Improved Spine-based Infrastructure for Routing in Ad hoc Networks. In *IEEE Symposium on Computers and Communications '98, Athens, Greece*, June 1998.

[19] S. Slijepcevic and M. Potkonjak. Power Efficient Organization of Wireless Sensor Networks. In *IEEE International Conference on Communications (ICC'01), Helsinki, Finland*, pages 472–476, June 2001.

[20] K. Sohrabi, V. Ailawadhi J. Gao, and G. Pottie. Protocols for Self Organization of a Wireless Sensor Network. In *IEEE Personal Communication Magazine*, volume 7, pages 16–27, October 2000.

[21] K. Sohrabi and G. Pottie. Performance of a Novel Self-Organization Protocol for Wireless Ad hoc Sensor Networks. In *50th IEEE Vehicle Technology Conference, The Netherlands*, September 1999.

[22] M. Steenstrup, B. Beranek, and Newman. *Ad hoc Networking*, chapter Cluster-Based Networks, pages 75–135. Addison-Wesley, first edition, December 2000.

[23] L. Subramanian and R. H. Katz. An architecture for building self-configurable systems. In *IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC 2000), Boston*, August 2000.

[24] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 7–14, August 1999.

[25] Jie Wu. *Handbook of Wireless and Mobile Computing*, chapter Dominating Set Based Routing in ad hoc wireless networks, pages 425–450. John Wiley, first edition, February 2002.