ABSTRACT

PARK, SANGJOON. Performance Analysis of Data Aggregation Schemes for Wireless Sensor Networks. (Under the direction of Associate Professor Injong Rhee).

Wireless sensor networks are suitable for applications in which sensors detect moving targets in the area of interest. In such applications, one of the key challenges is how to design efficient data aggregation protocols which reduce redundant packet transmissions in the sensor network. Towards this goal, centralized, tree-based, static-cluster, and dynamic-cluster aggregation schemes have been proposed. However, each scheme has its share of benefits and corresponding costs, and it is difficult to say with certainty whether a particular scheme is always better than others. In this paper, our goal is to compare all of the above mentioned aggregation schemes with comprehensive theoretical analysis, simulation and real experiments and attempt to give clear scenarios where a particular scheme may be more beneficial compared to others. Along the way, we also propose two cluster-based aggregation algorithms, which are simple enough to be implemented on resource-constrained sensor networks. As a first step towards this goal, we model the sensor network environment under certain simplifying assumptions and then derive closed form expressions for the total number of packet transmissions incurred by each aggregation scheme. Next, we complement the assumptions made for the analysis by performing extensive simulations under different environmental conditions, such as channel capacity (1Mbps, 250Kbbps) and

MAC (B-MAC, IEEE 802.11) protocol. Finally, we test the aggregation schemes on a real sensor network testbed comprising of 31 MICA2 sensors. Various metrics, such as total number of packet transmissions, aggregation ratio, average energy consumption, network lifetime, average end-to-end delay, and packet delivery ratio, are used to evaluate the performance of aggregation schemes.

The results show that the performance of data aggregation is highly dependant on setup overhead, node density, sensing range, and the distance from sources to the sink. When the sources are close to the sink and the sensing range is short, tree-based aggregation, with long aggregation delay, achieves better performance in all metrics, except end-to-end delays. However, as the distance from sources to the sink and the sensing range increase, dynamic-cluster aggregation shows the best performance over other schemes, because a large number of generated packets are reduced by the local data aggregation.

Performance Analysis of Data Aggregation Schemes for Wireless Sensor Networks

by

Sangjoon Park

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computer Science

Raleigh

2006

Approved By:

Dr. Khaled Harfoush	Dr. George N. Rouskas
Dr. Injong Rhee Chair of Advisory Committee	Dr. Robert Fornaro

To my parents and my wife,

who made all of this possible, for their endless encouragement and patience.

And also to my children,

who seem to be growing into two wonderful boys, in spite of the fact that their father was less available than he should have been.

Biography

Sangjoon Park received the B.S. and M.S. degree in electronics from Kyungpook National University, Korea, in 1988 and 1990 respectively. From 1990 to 2001, he was senior researcher in Agency for Defense Development in Korea. He has 12 years research experience and his background includes design and development of real-time multi-processing computer, Fire Control System, multi-sensor data fusion algorithm, Threat evaluation and Weapon Assignment algorithm, and tactical communication system of Surface to Air Missile system. He was a member of the Ministry of National Defense of SAM-X Project (Next generation air defense system) working group and Air-Force SAM-X Test and Evaluation team. From 2001, he is pursuing his Ph.D. in computer science at the North Carolina State University. His current research interest is multi-sensor data fusion, ad-hoc network and wireless sensor network.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless sensor networks have been deployed extensively for applications such as event/target detection, monitoring and tracking. In such applications, a large number of sensors are densely deployed, in a deliberate or ad-hoc manner, to cover the area required to be observed. When a target or event is detected, multiple sensors collaborate with each other to perform local data processing operations which include target detection, classification, multi sensor data fusion and tracking. The results of such data processing operations need to be communicated back to the sink in an energy-efficient and resource-friendly manner. Towards this goal, the paradigm of *data aggregation* has been put forward [13]. The key idea of data aggregation is to combine multiple data packets from different sensor nodes based on the redundancy present in the packets. When several spatially close sensors report an event, the report packets may contain highly correlated information, which could be combined into a single packet by intermediate forwarding nodes. This significantly cuts down on the number of packet transmissions, consequently reducing collisions, idle listening, redundant transmissions and other such energy depleting operations.

The aggregation schemes proposed till now can be categorized into *centralized aggregation*, *tree-based aggregation*, *static-cluster aggregation* and *dynamic-cluster aggregation*. In centralized aggregation, all the data packets travel to the sink without aggregation at forwarding nodes, or aggregation nodes, and the sink performs data aggregation. In tree-based aggregation [13], [17], [14], [10], [22] forwarding nodes merge the multiple data packets coming from different sources and transmit an aggregated data packet to the sink. In static-cluster aggregation [9], [8], [23] nodes are statically grouped into multiple clusters during the initial setup. Each cluster consists of one cluster head and one or more member nodes. When member nodes detect a target, they report the sensing information to their cluster heads, and then the cluster heads transmit an aggregated packet to the sink. Unlike the static-cluster aggregation, the dynamic-cluster aggregation [4], [16], [28], [5], [26] does not need initial setup to create the clusters. Data aggregation is triggered by the event. When sensors detect target, they collaborate with each other to elect the cluster head among them. The elected cluster head performs local aggregation and transmits an aggregated data packet to the sink.

Centralized aggregation represents the worst case for data aggregation, where all packet transmissions reach the sink without any aggregation along the way. This could happen if the sensor nodes are incapable of complex data aggregation operations, or if there is insufficient redundancy for data aggregation to occur in the first place. Tree-based aggregation relies on established routing paths. Therefore, it does not need to pay additional costs for creating a data aggregation backbone. However, note that in the worst case, tree-based aggregation could end up incurring the same number of packet transmissions as centralized aggregation, if the routes from the sources to the sink do not converge at any point [17], [12]. In static-cluster aggregation, cluster heads form a backbone to transfer event reports from sensors back to the sink. The cluster backbone may not be necessarily capacity preserving, and the routes computed over the backbone may be sub-optimal in terms of path length or throughput. Dynamic-cluster aggregation forces aggregation to be performed near the event sensing area. But this scheme need to pay costs for electing the cluster head on-the-fly.

While each scheme has its benefits and corresponding costs, it is difficult to say with certainty whether any one scheme dominates the rest, or whether the choice of the scheme depends on particular network configurations like network density, diameter, etc. **Our goal in this paper is to compare all of the above aggregation schemes with comprehensive theoretical analysis, simulation and real experiments and attempt to give clear scenarios where a particular scheme may be more beneficial compared to others.** Also, along the way, we propose two simple algorithms – one for implementing static-cluster, and another for dynamic-cluster aggregation. The algorithms are simple enough to be implemented on the resource-constrained sensor network environment. We do not claim that these algorithms are novel or optimal, in fact several ideas are similar to the existing clustering and topology control algorithms. Instead, our focus in this paper is on simplicity and ease of implementation and of evaluation.

Our evaluation approach is the following. Assuming an idealized Medium Access Control (MAC), no channel losses or collisions, we first analyze the single most important metric in data aggregation – the total number of packet transmissions for each aggregation scheme. To complement the assumptions made for the theoretical analysis, we next compare the schemes using extensive simulation under different bandwidths ($1Mbps$, $250Kbps$) MAC layer protocols (B-MAC [20], 802.11 [19]) and performance metrics such as number of packet transmissions, aggregation ratio, average energy consumptions, network lifetime, average end-to-end delay, and packet delivery ratio. Finally, we study the performance of each scheme in a real 31-node mote testbed.

The results show that the performance of data aggregation schemes is highly sensitive to different network parameters such as setup overhead, node density, sensing range, and the distance from sources to the sink. When the sources are close to the sink and the sensing range is short, tree-based aggregation, which has longer aggregation delay, achieves better performance in all metrics, except end-to-end delays. At the same time static-cluster and dynamic-cluster aggregation cannot take advantage of data aggregation because the amount of overhead exceeds the packet reduction by data aggregation. However, as the distance from sources to the sink and the sensing range increase, cluster-based(static and dynamic) aggregation show better performance. Under such condition, dynamic-cluster aggregation shows the best performance over the other schemes, because a large number of generated packets are reduced by the local data aggregation.

The remainder of the article is organized as follows. Section 2 has discussion on related work, section 3 describes modeling of sensor network, section 4 describes characteristics and detail algorithms of various aggregation schemes, section 5 shows the analytical results on the grid topology and 6 shows the analytical results on the mesh topology, section 7 reports a comprehensive simulation results, section 8 shows the experimental results on a MICA2 testbed. Finally, concluding remarks are made in section 9.

Chapter 2

Related Work

In directed diffusion [13], aggregation is performed in pre-established low-latency paths. There is no explicit effort to share routing paths, and hence in the worst case, there may be no reduction in the number of packet transmissions, compared to the base case of centralized aggregation [12]. In [14], the authors propose a greedy incremental tree to increase path sharing by modifying the path establishment and maintenance algorithm of directed diffusion [13]. In [17], [12] the authors show that the greedy incremental tree achieves significant energy savings over the other tree-based aggregation schemes when the number of sources is large. But this scheme cannot apply to an application in which events are moving frequently because this algorithm has to pay high overhead to reorganizing the greedy aggregation tree whenever target moves. Thus, the additional cost of reconstructing a greedy incremental tree could be an overkill.

In static-cluster aggregation, cluster creation algorithm is one of the dominant factors to affect the performance of data aggregation. In LEACH, k nodes are randomly selected as cluster heads and the other nodes select one of the closest cluster heads as their cluster head. In LEACH, the cluster heads are selected randomly without knowledge of their neighbors, which could result in unbalanced clusters in terms of cluster area and number of nodes per cluster. The LEACH assumes that all nodes can control transmission power to reach the sink and cluster heads can directly communicate with the sink. Soheil et al.[8] propose an optimal balanced k-clustering scheme such that each cluster is balanced (in terms of the number of member nodes) and the total distance between sensor nodes and its cluster heads is minimized. When each cluster head is constrained to use only a certain number of communication channels, this scheme does not necessary allow a member node to choose the closest cluster head. In HEED[25], the clustering process requires a number of iterations. During an iteration, a node becomes a cluster head with a certain probability. All other nodes, which are not cluster head, select the cluster head which has lowest intra-cluster communication cost. Unlike LEACH, HEED creates well-balanced clusters. HEED uses two radio transmission power levels, one for intra cluster communication and the other for inter cluster communication. HEED relies on the assumption that cluster heads can communicate with each other and form a connected graph – realizing this assumption in practical deployments could be tricky.

In [27], the authors propose a dynamic convoy tree-based collaboration (DCTC) framework to detect and track the mobile target. When sensors detect a target, they construct a local aggregation tree, called *convoy-tree*, by selecting one of node to be the root of the convoy-tree. As the target moves, the tree is dynamically expanded and pruned. As the target mobility increases, so does the overhead required to create, expand, prune and reconfigure the convoy tree. In [28], the authors address the information driven approach to dynamically collaborate multiple sensors for target tracking applications. Their idea is to determine for a network the participants in a sensor collaboration which is based on information constrains as well as constraints on cost and resource consumption. In a moving target tracking scenario, one leader node estimates the target information as well as the next best leader sensor and then performs a hand-off of this tracking information to the next leader node. Chen et al. [5] devise a decentralized dynamic clustering algorithm for target tracking. This scheme assumes a hierarchical sensor network that is composed of a static backbone of sparsely placed high-capability sensors and moderately-to-densely populated low-end sensors. The high-capability sensors perform the role of cluster head and the low-end sensors provide the sensing information upon request of their cluster heads. The clusters are created on-the-fly when the cluster heads are triggered by events. When the acoustic signal strength detected by a cluster head exceeds a pre-defined threshold, the active cluster head broadcasts an information solicitation packet, asking sensors in its vicinity to join the cluster and provide their sensing information. After receiving a sufficient number of replies from sensors, the cluster head applies a localization method to estimate the location of the target and sends a report to the base station. Yu et al.[26] study a cluster-based lightweight middle-ware framework that consists of a cluster control layer and a resource management layer. In their scheme, the cluster layer is responsible for dynamic cluster formation around the target phenomena. This paper only outlines design issues and challenges in the cluster control layer instead of describing a detail algorithm.

Chapter 3

Sensor Network Model

Before describing in detail each aggregation scheme, we first describe how we model the environment that the sensors interact with. We consider an environment where N homogenous sensor nodes are deployed in a deliberate or ad-hoc manner, in an A x A area. The nodes are assumed to be stationary after deployment. The nodes are responsible for detecting a moving target. Nodes are location-aware by means of GPS or some other localization algorithm. All nodes are equipped with target detection, classification and data fusion facilities.

3.1 Sensor deployment model

Contemporary sensor deployments usually fall under two categories – pre-planned and random deployment. For applications that detect hostile targets in battle fields, the sensing area is difficult to access. In such a situation, sensors may be deployed by air drops or launched via artillery in the battle field, thus resulting in random placement of sensors. On the other hand, when the sensing area is easy to access and has a friendly environment, sensors can be placed manually, thus resulting in regular pattern of sensor placement. Monitoring and control of indoor environment, and structural heath monitoring of bridges applications can be good examples of pre-planned deployment.

3.2 Sensor detection model

Crossbow's wireless security mote MSP410 is designed for use in a wide variety of wireless intrusion detection applications. MSP410 mote combines 433 MHz MICA2 processor/radio boards with passive infrared(PIR), magnetic, and acoustic sensors in a package. The PIR sensor detects occupancy and human body motion in a range of 25-50 ft when coupled with a good discrimination algorithm. The magnetic sensor detects vehicles at 20-30 ft depending on the strength of magnetic field, while the acoustic sensor can detect a sound source at a range of 100 ft.

The signal strength of targets at sensor is dependent on the type of the target objects, the distance between the target and sensor, and the amount of measurement noise. The received signal strength, R_i , of target at each sensor is defined as

$$R_i = \alpha \cdot dist[T, S_i]^{-\gamma} + \omega_i \qquad 1 \leq i \leq N \tag{3.1}$$

where N is the total number of sensor and α is the constant factor dependant on the types of objects and it can be determined empirically. The $dist[T, S_i]$ is the Euclidean distance between target position T and sensor S_i. The γ is the known attenuation coefficient(2 or 3) which is affected by obstacles and terrain conditions, ω_i is the white Gaussian noise of sensor i, which has zero mean and variance σ^2.

In our research, we consider simple *binary detection*. In this model, each sensor has target sensing radius, r, when the Euclidian distance between sensor and targets is less than the target sensing range, it is called *detected*. The equation 3.2 shows the binary detection model that describes the detection $D_{xy}(S_i)$ of target by the sensor S_i.

$$D_{xy}(S_i) = \begin{cases} 1 & \text{if } dist[T, S_i] < r \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

3.3 Radio channel model

Since sensor nodes are stationary, the *free space* or the *two-ray propagation model* can be used to model a radio channel between transmitter and receiver [21]. The free space propagation model can be used only if the radio channel between transmitter and receiver has a line-of-sight path. But in the real world, a ground reflected path impacts the performance of radio channel. The two-ray propagation model considers both directed path and a ground reflected propagation path between transmitter and receiver. In this model, the received power at a distance d from a transmitter node, $P_r(d)$, is given by

$$P_r(d) = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4} \tag{3.3}$$

where P_t is transmitted power, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, h_t is the height of the transmitter, and h_r is the height of the receiver. The received signal power$P_r(d)$, is used to determine the connectivity between nodes. If the the received power at a distance d from a transmitter node, $P_r(d)$, is greater than the threshold value(γ), it is called *connected*. Thus combining the equation 3.3 and the threshold, the distance d is given by

$$d \leq \sqrt[4]{\frac{P_t G_t G_r h_t^2 h_r^2}{\gamma}} \tag{3.4}$$

Note from equation 3.4 that the radio coverage of nodes in the two-ray ground propagation model is a disk of radius $\sqrt[4]{\frac{P_t G_t G_r h_t^2 h_r^2}{\gamma}}$ centered at the transmitter. In our research, we consider simple radio connectivity model and equation 3.5 shows the connectivity C of nodes between node S_i and S_j

$$C = \begin{cases} 1 & \text{if } d(S_i, S_j) \leq \sqrt[4]{\frac{P_t G_t G_r h_t^2 h_r^2}{\gamma}} \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where $d(S_i, S_j)$ is the Euclidean distance between sensor S_i and S_j.

The two-ray model has been found to be reasonably accurate for predicting the stationary radio channel. ns-2 also uses two-ray ground propagation model to model the wireless radio channel.

3.4 Data Aggregation Model

In our research, we assumed that each node has homogeneous sensing devices and each node generates a single packet which is made up of R bits of sensing information. Data aggregation may take several forms, depending on the specific requirements and features of the application, but in our study we consider a very fundamental form of data aggregation. Let the generated packet $\lambda_{i-incoming}$ in node i is comprised of the local sensing traffic at node i (λ_i) and the sum of packets coming from neighbor nodes λ_j which detect the same events, and is given by

$$\lambda_{i-incoming} = \lambda_i + \sum_{j \in S_j}^{k} \lambda_j = (k+1) \times R, \quad i = 1, \ldots, N. \tag{3.6}$$

where $S_j, j = 1, \ldots, k.$, is the set of neighbor nodes which detect same events. The total amount of outgoing data $\lambda_{i-outgoing}$ at node i is given by

$$\lambda_{i-outgoing} = \alpha \times \lambda_{i-incoming} = \alpha \times (k+1) \times R \tag{3.7}$$

where α is the *aggregation factor* in the range from $\frac{1}{k+1}$ to 1. When $\alpha = \frac{1}{k+1}$, multiple data packets are aggregated into a single outgoing data packet and this is called *perfect aggregation*. When $\alpha = 1$, the amount of outgoing packets are the same as the amount of generated packets since the aggregation node does not perform any data aggregation. Therefore, the number of packet transmissions ρ_i at node i is given by

$$\rho_i = \lceil \alpha \times (k+1) \rceil \quad packets \tag{3.8}$$

where $\lceil \ \rceil$ indicates the ceiling of $\alpha \times (k+1)$.

3.5 Routing Model

In this study, we assume that nodes run a proactive routing protocol, for instance Mint [24], and create a many-to-one routing topology for transmitting event reports to the sink.

Chapter 4

Data Aggregation Schemes

4.1 Centralized aggregation

In centralized aggregation, all sensing nodes transmit their sensing data to the sink along the routing paths without aggregation at forwarding nodes. Once all the data packets have arrived, the sink performs data aggregation. Centralized aggregation is applicable for sensor networks where the sensing nodes do not have enough computing capability to aggregate the data packets coming from multiple sensor nodes. In this scheme, sensor nodes do not need to implement the complex data aggregation algorithm but a large number of data packets are transmitted in a network. This may result in an congestion, packet loss, queuing delay and high energy consumption.

4.2 Tree-based aggregation

In tree-based aggregation, all sensing nodes transmit their sensing data to the sink along the established routing paths. When two or more routing paths merge at a single forwarding node, also called *aggregation node*, performs the data aggregation operations. Such aggregation nodes wait for fixed period of time, called *aggregation delay*, to merge the data packets coming from multiple sources. Even though several sensors detect one target simultaneously, individually transmitted data packets from sensors may not arrive at the same time at an aggregation point because they may travel through different routing paths and may incur different network delays. As the aggregation delay is longer, data packets have a higher chance of being merged at an aggregation point during forwarding to the sink, but end-to-end network delays from the sources to the sink get longer. The impact of aggregation delay on the performance of tree aggregation will be further discussed in the simulation section.

Note that the event packets generated by a single event at different sources may travel on completely disjoint paths towards the sink, consequently there may be no data aggregation performed at all [17], [12]. Figure 4.1 illustrates three possible data aggregation scenarios where the structure of the underlying routing topology and the location of targets impact the aggregation performance.

In the first scenario, the event data packets generated due to target T1 are merged at node B, which is several hops away from the target sensing area. In the second

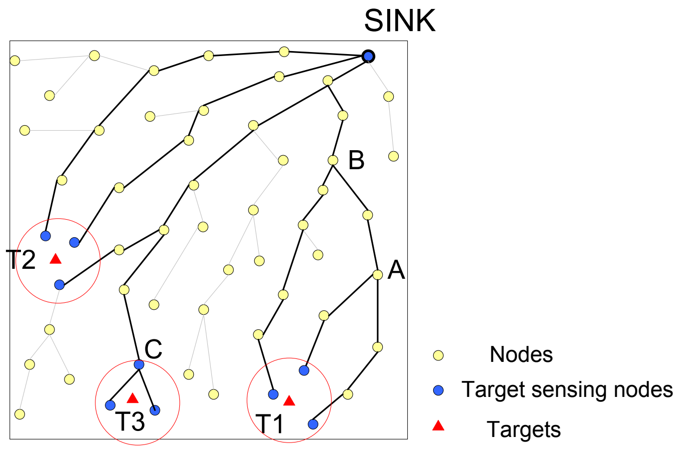


Figure 4.1: Various data aggregation scenarios in an aggregation tree

scenario, the event data packets generated due to target T2 are never merged at any forwarding node. In the final scenario, event data packets generated due to target T3 are aggregated at the forwarding node C, which has the minimum distance to the sink, and this node transmits aggregated packet to the sink.

For the purpose of analysis and simulation, we devise the following simple tree-based aggregation algorithm.

Tree aggregation algorithm:

1. Nodes which detect a target transmit the corresponding event data packet to their routing parent.

2. At intermediate nodes, data aggregation is performed on all event data packets reporting the same event which arrive during the aggregation delay period. As described in Section 3, depending on the aggregation factor α, the node then transmits ρ number of aggregated data packets to the sink.

4.3 Static-cluster aggregation

In the static-cluster aggregation scheme, the key idea is to proactively distribute the network into smaller *clusters* which consist of one node responsible for data aggregation called *cluster head*, and remaining nodes responsible for sensing and event reporting, called *member nodes*. Whenever the member nodes sense an event, they transmit event data packets to their respective cluster head, who then performs data aggregation operations and depending on the aggregation factor α, transmits ρ aggregated data packets toward the sink. Note that the clusters are proactively created, and are not influenced by the position or regularity of events. Hence we denote such aggregation as static cluster aggregation. In Section 4.4 we will look at *dynamic-cluster* aggregation, where clusters are created reactively, in response to events.

Ideally, all nodes sensing an event belong to a single cluster, in which case the number of packets transmitted would be minimum. However it is possible that the nodes detecting a target may belong to different clusters. This is illustrated in Figure 4.2, where a single target is sensed by member nodes in clusters 2, 5, and 6. In such cases multiple cluster heads will transmit information regarding the same event, reducing the benefits of data aggregation by some amount.

We now present a simple algorithm for implementing static-cluster aggregation. Our algorithm consists of a *setup phase* and an *operational phase*. In brief, during the setup phase, a fraction of nodes in the network are elected to be cluster heads,

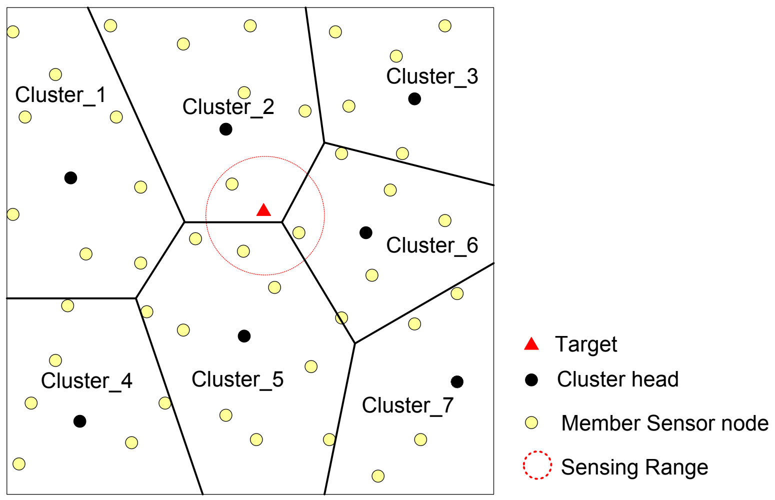


Figure 4.2: Static clustering scheme using Voronoi diagram

which are responsible for data aggregation and during the operational phase nodes are involved in sensing, event detection, event reporting and data aggregation. Since cluster heads tend to dissipate energy at a higher rate than member nodes, it is necessary to cycle through the setup and operational phase (as shown in Figure 4.3) so that different nodes are elected as cluster heads each time. This balances out the energy of the nodes in the system, which helps in extending the system lifetime.

The setup phase runs the following operations in sequence: Time Synchronization, Neighbor Discovery, and Clustering as shown in Figure 4.3. We now describe each operation in detail.

Time Synchronization Phase: Nodes in the system need to maintain time synchronization, so that they start the clustering and subsequent re-clustering phases together. We use a lightweight time synchronization scheme [7] which requires only localized time synchronization.

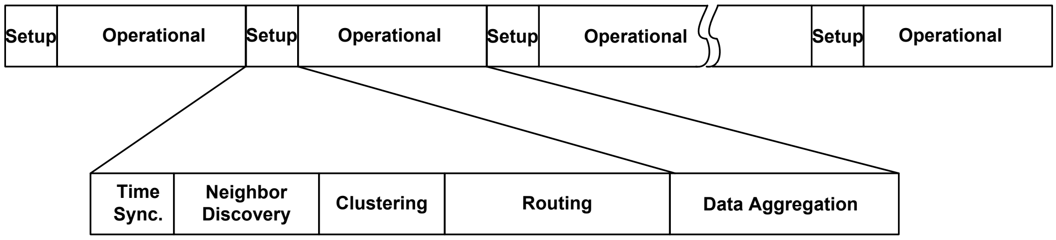


Figure 4.3: Operations of static-clustering algorithm

Neighbor Discovery Phase: During this phase, nodes periodically broadcast neighbor discovery *beacons*. A node receiving such beacon messages uses them to generate link quality estimates to its neighbors. This information is useful later during the clustering phases where member nodes decide the cluster head.

Clustering Phase: During this phase, a fraction of nodes in the network are selected as *cluster heads*. Non-cluster head nodes adjacent to a cluster head become *members* of the cluster head. The algorithm proceeds in rounds as follows:

1. **Cluster head candidate selection**: At each round, a *covered* node is a node which has become either a cluster head or a member node in one of the previous rounds. At the beginning of round i, an uncovered node in the network becomes a cluster head candidate with probability $k^i p_0$. Where p_0 is the initial probability of being a cluster head in round 0 and the value of p_0 must be selected small enough so that only one node within an circular area of radio range, also called *cluster area*, becomes cluster head candidate. Both parameters k and p_0 decide the number of rounds required for the whole network to finish clustering, as well as the fraction of nodes in the network elected as cluster heads.

2. **Cluster head advertisement** : If a node becomes a cluster head candidate, it starts a timer. The length of the timer is inversely proportional to the residual energy of the node; if the node has more residual energy, the timer expires earlier. When the timer expires, the node broadcasts a *Cluster Head Advertisement* (CHA) message and becomes cluster head. The use of such a timer allows nodes with more residual energy to be elected as cluster heads. The CHA message contains the node's ID, and the value of the node's residual energy. Once a node becomes a cluster head in round i, it retransmits CHA message in three subsequent rounds, to increase the probability that its neighbors receive its message in lossy environments.

3. **Cluster head advertisement message reception**: When an uncovered node receives a CHA message from another node, it becomes a member node. When a candidate cluster head receives a CHA message from another node; it compares the residual energy value in the message to its own residual energy; if sender's residual energy is more than its own, the node becomes a member and suppresses its timer. If a node is neither a cluster head nor a member node at the end of a round, it begins a new round and repeats the above steps.

4. **Cluster head decision**: After a finite number of rounds, all nodes are covered. A member node, which has received multiple CHA messages from $CH_1, CH_2, ..., CH_n$ cluster heads, decides to become a member of only one cluster among the n clus-

ters. This decision is based on the link quality estimates to the n cluster heads, which it obtained during the neighbor discovery phase.

Operational Phase: In the operational phase, nodes perform the following operations:

1. When a node detects a target, it reports sensing messages to its cluster head.
2. Cluster heads transmit an aggregated data packet to the sink along the routing path.

Figures 4.4 shows the clustering result of our devised static clustering algorithm on a network of randomly deployed 200 nodes.

4.4 Dynamic-cluster aggregation

In dynamic-cluster aggregation, cluster formation is triggered by a target detection and clusters are created around the target area on-the-fly. Multiple sensing nodes collaborate with each other to elect a cluster head and the elected cluster head transmits an aggregated data packet to the sink.

Prior research [28], [5], [26] cites the following desirable features of dynamic-cluster aggregation:

- When an event occurs, only the spatially close nodes participate in cluster

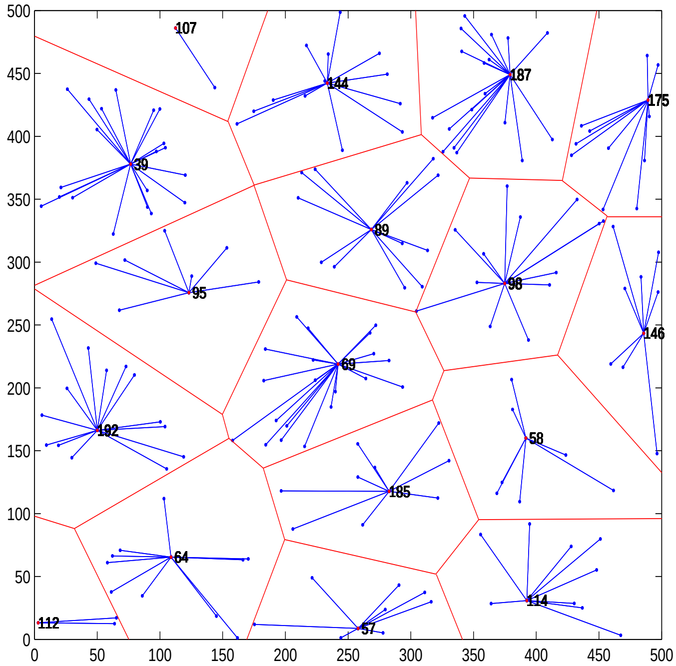


Figure 4.4: Clustering result of static-cluster aggregation on a network of randomly deployed 200 nodes

formation. The rest of the network need not perform energy-depleting cluster setup and maintenance operations.

- If the radio transmission range of a node is more than twice its sensing range, then all nodes which sense an event are within radio range of each other, and hence can agree on one single cluster head to perform data aggregation. Note this is unlike static clustering, where it is possible that multiple clusters may transmit information regarding the same event, as was illustrated in Figure 4.2.

Not much work has been done on the mechanism of dynamically electing an aggregation node for data aggregation. In our study, we devise a target oriented dynamic aggregation algorithm which is simple and practical to implement on real sensors. In our scheme, when sensors detect a target, they exchange their sensing information. The sensing node which has higher residual energy and which receives more sensing reports has a higher probability of being selected as a cluster head. We outline the algorithm in detail below:

Target Oriented Dynamic Aggregation (TODA) algorithm:

1. *Report sensing information*: When a sensor detects a target, it broadcasts a *Sensing Report(SR) message* to its neighbors. This message contains its node ID, residual energy and information regarding the target.

2. *Cluster head candidate election*: If a sensing node receives an SR message from its neighbors and its residual energy is greater than the average residual energy of sensing nodes(including itself), the node becomes a cluster head candidate and starts a *Cluster Head Advertisement* (CHA) timer for the transmission of a CHA message. The expiry time of this timer is generated from two values – (1) a value inversely proportional to the number of received SR messages, (2) a random value, which is smaller than first value. The second value is used to break ties between nodes that receive the same number of SR messages. Thus nodes which have received more SR messages and which have higher residual energy tend to become cluster head candidates. This is useful, since nodes which receive more SR messages can generate more robust and multidimensional view of target information.

3. *Cluster head advertisement*: When the CHA timer of a cluster head candidate expires, it becomes a cluster head and broadcasts a CHA message to its neighbors. This message contains its node ID and the CHA timer delay value.

4. *Cluster head advertisement message reception*: When a cluster head candidate receives a CHA message, it compares the CHA timer delay value in the message to its own CHA timer delay value – if the sender's value is smaller than its own, the node cancels its CHA timer.

5. *Multi cluster head resolution*: Due to unreliable wireless link and message collisions, it is possible that multiple cluster heads could be elected in response to an event. To reduce such occurrences, we employ a simple procedure. An elected cluster head starts a timer called *multiple cluster head resolution* timer, which is used to delay the transmission of the aggregated data packet to the sink. If the cluster head receives another CHA message before the multiple cluster head resolution delay timer expires, it sends a SUPPRESS message to the sender.

6. *Data packet transmission*: When the multiple cluster head resolution timer expires, a cluster head performs data aggregation and transmits aggregated data packets to the sink.

Chapter 5

Analytical Results for Pre-planned Deployment

The main goal of data aggregation is to reduce the total number of packet transmissions in the network. The ideal case is where multiple nodes detect an event, but only one node reports it back to the sink. We denote this as the *reference case* for data aggregation. Given this reference case, we define a new metric – *aggregation ratio* – to compare different aggregation schemes:

Aggregation Ratio: Consider a system which uses an aggregation scheme ϕ. Without loss of generality, assume perfect aggregation at every node. Hence, the aggregation ratio for this scheme AR_ϕ, is defined to be the ratio of the average number of packet transmissions generated by the reference case to the average number of packet transmissions generated by the aggregation scheme ϕ, $0 < AR_\phi \leq 1$.

Intuitively, the aggregation ratio of an aggregation scheme tells us how close the scheme is to the reference case; higher values of the aggregation ratio indicate that the scheme performs data aggregation well. Our goal in this section is to derive closed form expressions of the aggregation ratio for each aggregation scheme described in Section 4.

To keep the analysis tractable, we make the following assumptions:

- Suppose $N \times N$ sensors are placed in a grid topology as shown in Figure 5.1. Distance between each sensor is denoted by D and node density, d, in the network area is:

$$d = \frac{N^2}{(ND)^2} = \frac{1}{D^2}. \tag{5.1}$$

- The sink, S, is located at the top-right corner. All nodes maintain, the shortest route to the sink. In a grid topology, a node can only communicate with its immediate horizontal and vertical neighbors. Therefore, as shown in Figure 5.1, all nodes on the diagonal line have the same hop count from nodes to the sink.

- The radio channel is loss-free.

- The MAC protocol is an idealized MAC, in that it avoids all packet collisions.

- The radio present in each node follows the two ray propagation model, as described in Section 3.3 with transmission range R_r, and $R_r > 2R_s$.

- The sensors present in each node follow the binary detection model, as described

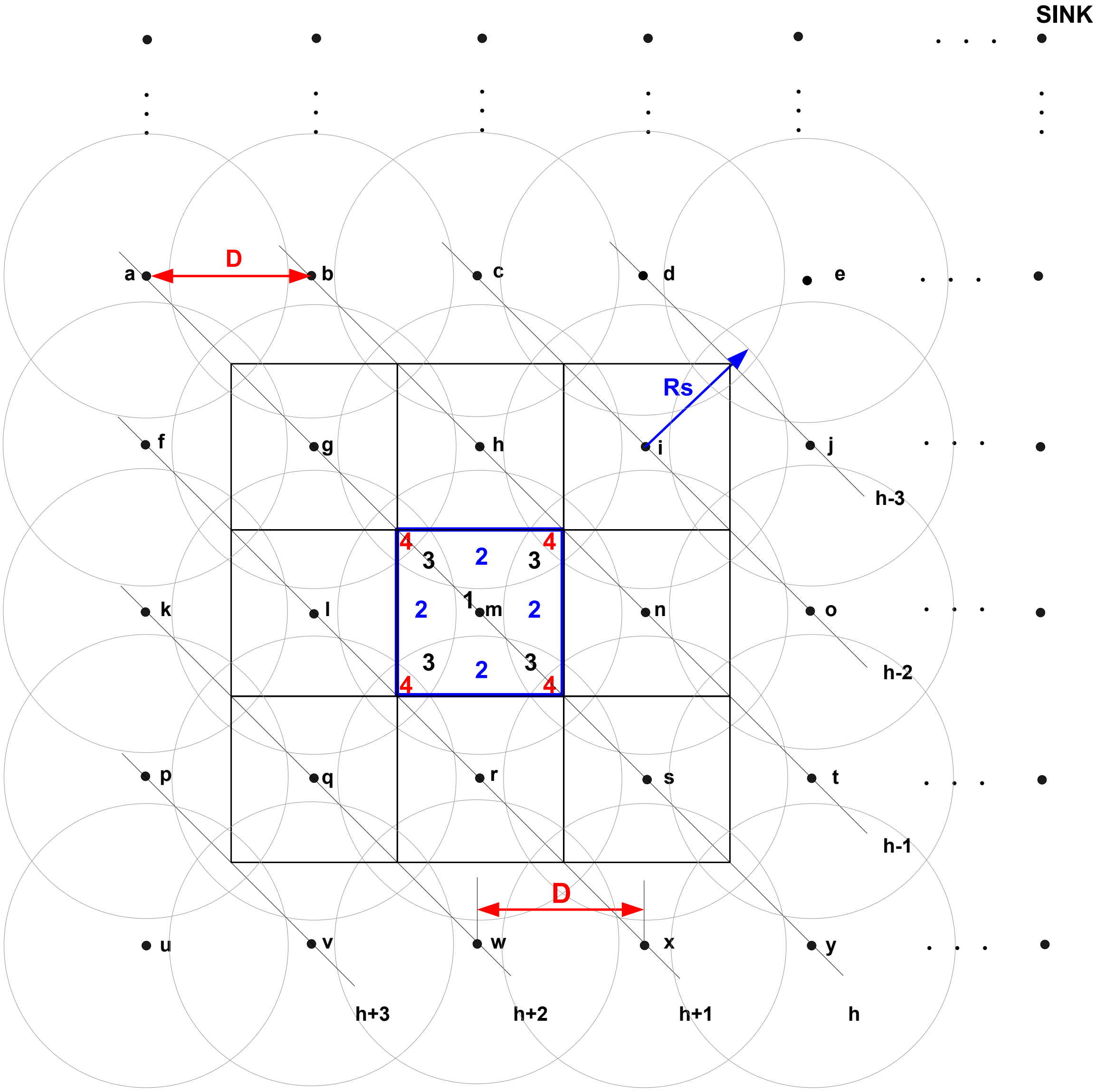


Figure 5.1: Network layout for analysis

in Section 3.2 with sensing range R_s.

Under these conditions, the entire sensor network can be divided into four zones, labeled 1, 2, 3 and 4, as shown in Figure 5.1. If we assume that the target occurs randomly in the entire sensor network, then these labeled zones represent the characteristics of the entire sensor network. The label of zone k has the meaning that this area is covered by k sensors, or if the target is generated in this zone, it will be detected by k sensors. Hence the probability that a target is detected by k sensors in the sensor network can be obtained from:

$$\begin{aligned} Pr(k=1) &= \frac{\text{Area(indexed 1)}}{D^2} \\ Pr(k=2) &= \frac{\text{Area(indexed 2)}}{D^2} \\ Pr(k=3) &= \frac{\text{Area(indexed 3)}}{D^2} \\ Pr(k=4) &= \frac{\text{Area(indexed 4)}}{D^2} \end{aligned} \tag{5.2}$$

where Area(indexed k) is sum of areas labeled k, $k = 1, \ldots, 4$ in Figure 5.1 and is obtained by means of geometric analysis. These areas are given by:

$$
\begin{aligned}
\text{Area(indexed 4)} = {} & 4{R_s}^2\cos^{-1}\Big(\sqrt{\frac{D\sqrt{{R_s}^2 - \frac{D^2}{4}} + {R_s}^2}{2{R_s}^2}}\Big) \\
& - 4\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2 - \frac{D^2}{4}} + 2{R_s}^2} - \frac{D^5 - 2D^4\sqrt{{R_s}^2 - \frac{D^2}{4}}}{2D + 4\sqrt{{R_s}^2 - \frac{D^2}{4}}} + \frac{D^4}{4}} \\
& - 2D\sqrt{{R_s}^2 - \frac{D^2}{4}} - D^2 + \frac{2D^3 - 4D^2\sqrt{{R_s}^2 - \frac{D^2}{4}}}{D + 2\sqrt{{R_s}^2 - \frac{D^2}{4}}}
\end{aligned}
\tag{5.3}
$$

$$
\begin{aligned}
\text{Area(indexed 3)} = {} & 4{R_s}^2\cos^{-1}\Big(\frac{D}{\sqrt{2}R_s}\Big) - 2\sqrt{2}D\sqrt{{R_s}^2 - \frac{D^2}{2}} \\
& - 8{R_s}^2\cos^{-1}\Big(\sqrt{\frac{D\sqrt{{R_s}^2 - \frac{D^2}{4}} + {R_s}^2}{2{R_s}^2}}\Big) \\
& - 8\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2 - \frac{D^2}{4}} + 2{R_s}^2} - \frac{D^5 - 2D^4\sqrt{{R_s}^2 - \frac{D^2}{4}}}{2D + 4\sqrt{{R_s}^2 - \frac{D^2}{4}}} + \frac{D^4}{4}} \\
& - 4D\sqrt{{R_s}^2 - \frac{D^2}{4}} - 2D^2 + \frac{4D^3 - 8D^2\sqrt{{R_s}^2 - \frac{D^2}{4}}}{D + 2\sqrt{{R_s}^2 - \frac{D^2}{4}}}
\end{aligned}
\tag{5.4}
$$

$$
\begin{aligned}
\text{Area(indexed 2)} = {} & 4{R_s}^2\cos^{-1}\Big(\frac{D}{2R_s}\Big) - 2D\sqrt{{R_s}^2 - \frac{D^2}{4}} \\
& - 8{R_s}^2\cos^{-1}\Big(\frac{D}{\sqrt{2}R_s}\Big) + 4\sqrt{2}D\sqrt{{R_s}^2 - \frac{D^2}{2}}
\end{aligned}
\tag{5.5}
$$

$$\begin{aligned}
\text{Area(indexed 1)} &= 4D^2 - 4{R_s}^2\cos^{-1}(\frac{D}{2R_s}) + 4{R_s}^2\cos^{-1}(\frac{D}{\sqrt{2}R_s}) \\
&+ 4{R_s}^2\cos^{-1}(\sqrt{\frac{D\sqrt{{R_s}^2 - \frac{D^2}{4}} + {R_s}^2}{2{R_s}^2}}) \\
&+ 8D\sqrt{{R_s}^2 - \frac{D^2}{4}} - 2\sqrt{2}D\sqrt{{R_s}^2 - \frac{D^2}{2}} \\
&+ 12\sqrt{\frac{D^4R^2}{2D\sqrt{{R_s}^2 - \frac{D^2}{4}} + 2{R_s}^2} - \frac{D^5 - 2D^4\sqrt{{R_s}^2 - \frac{D^2}{4}}}{2D + 4\sqrt{{R_s}^2 - \frac{D^2}{4}}} + \frac{D^4}{4}} \\
&- \frac{6D^3 - 12D^2\sqrt{{R_s}^2 - \frac{D^2}{4}}}{D + 2\sqrt{{R_s}^2 - \frac{D^2}{4}}}
\end{aligned} \tag{5.6}$$

5.1 Centralized aggregation

In centralized aggregation, all sensor nodes transmit their sensing information to the sink along their shortest paths without aggregation at forwarding nodes. When a sensor generates data packet at a distance of h hops from the sink, h transmissions are required to deliver data packet to the sink. If k sensors are located at h hops away from the sink, the total number of data packets transmitted in the network is k*h. Thus, the total number of packet transmissions, ρ_c, in a centralized aggregation is defined to be $\rho_c = \sum_{i=1}^{k} d(S_i, S)$, where $d(S_i, S) = d_i, i = (1, \ldots, k)$, is the distance in terms of hop count between a target sensing nodes S_i and the sink, S. k is the number of sensors detecting the target. The expected total number of packet transmissions is defined to be

$$E[\rho_c] = E[\sum_{i=1}^{k} d_i], i \in (1, \ldots, k). \tag{5.7}$$

since target sensing area is divided into four zones where a target is detected by the k sensors

$$\begin{aligned} E[\rho_c] = {} & E[d_1|k=1] \times Pr(k=1) \\ & + E[d_1+d_2|k=2] \times Pr(k=2) \\ & + E[d_1+d_2+d_3|k=3] \times Pr(k=3) \\ & + E[d_1+d_2+d_3+d_4|k=4] \times Pr(k=4) \end{aligned} \tag{5.8}$$

where $E[d_1|k=1]$ is expected hop count of sensing node to the sink when the target is detected in zone 1. In Figure 5.1, only node m, which is at a h hop count distance from the sink, detects the target. Thus expected hop count of sensing node to the sink when the target is detected in zone 1 is

$$E[d_1|k=1] = h \tag{5.9}$$

$E[d_1+d_2|k=2]$ is expected value of two sensing nodes' hop count when target is detected in zone 2. In Figure 5.1, there are four equal size areas labeled 2 in which the target is detected by both node m and n, or both node m and r, or both node m and l, or both node m and h. Each area has equal probability, thus expected value of two sensing nodes' hop count when target is detected in zone 2

$$\begin{aligned}E[d_1 + d_2 | k = 2] &= \frac{1}{4}[(h + (h-1)) + (h + (h+1)) + (h + (h+1)) + (h + (h-1))] \\ &= \frac{1}{4}(8h) = 2h\end{aligned} \tag{5.10}$$

By similar reasoning, the expected value of three sensing nodes' hop count when target is detected in zone 3, $E[d_1 + d_2 + d_3 | k = 3]$, and the expected value of four sensing nodes' hop count when target is detected in zone 4, $E[d_1 + d_2 + d_3 + d_4 | k = 4]$ is given by

$$\begin{aligned}E[d_1 + d_2 + d_3 | k = 3] &= \frac{1}{4}[(h + 2(h-1)) + (h + 2h) + (h + 2(h+1)) + (h + 2h)] \\ &= \frac{1}{4}(12h) = 3h\end{aligned} \tag{5.11}$$

$$\begin{aligned}E[d_1 + d_2 + d_3 + d_4 | k = 4] &= \frac{1}{4}[(h + 2(h-1) + (h-2)) + (2h + (h+1) + (h-1)) \\ &\quad + (h + 2(h+1) + (h+2)) + (2h + (h+1) + (h-1))] \\ &= \frac{1}{4}(16h) = 4h\end{aligned} \tag{5.12}$$

Hence, the expected total number of packet transmissions in centralized aggregation is

$$\begin{aligned} E[\rho_c] &= h \times Pr(k=1) + 2h \times Pr(k=2) + 3h \times Pr(k=3) + 4h \times Pr(k=4) \\ &= h \times [Pr(k=1) + 2Pr(k=2) + 3Pr(k=3) + 4Pr(k=4)] \end{aligned} \tag{5.13}$$

The aggregation ratio of centralized aggregation is obtained by:

$$AR_c = \frac{1 \times h}{E[\rho_c]} = \frac{1}{Pr(k=1) + 2Pr(k=2) + 3Pr(k=3) + 4Pr(k=4)} \tag{5.14}$$

5.2 Tree-based aggregation

In tree-based aggregation, sensing nodes send data packets to the sink along the established routing paths. When routing paths are merged together, packets are aggregated at forwarding nodes. As explained in section 4.2, in the worst case scenario, data packets can not aggregated until they reach the sink. In this case, the number of packet transmissions is the same as centralized aggregation. Thus, the total number of packet transmissions is bounded by

$$E[\rho_T] \leq E[\rho_c] \tag{5.15}$$

In [17], the authors show that if the maximum pair-wise hop count between the sensing nodes is $Y \geq 1$, the total number of packet transmissions required for the optimal tree-based aggregation, ρ_t, satisfies the following bounds:

$$min(d_i) + (k-1) \leq \rho_t \leq (k-1)Y + min(d_i) \tag{5.16}$$

where k is the number of sensing nodes and $min(d_i)$ is the hop count (between target sensing node S_i and the sink) of an aggregation node which has minimum hop count among sensing nodes. Note that since $R_s < 2R_r$, $Y = 1$. Thus the expected total number of packet transmissions, $E[\rho_t]$, is bounded by:

$$E[\rho_t] \geq E[min(d_i) + (k-1)], i \in (1, \ldots, k) \tag{5.17}$$

since target sensing area is divided into four zones where a target is detected by the k sensors

$$
\begin{aligned}
E[\rho_t] \geq E[(min(d_i)|k = 1] * Pr(k = 1) \\
+ [E[(min(d_i)|k = 2] + 1] * Pr(k = 2) \\
+ E[(min(d_i)|k = 3] + 2] * Pr(k = 3) \\
+ E[(min(d_i)|k = 4] + 3] * Pr(k = 4)
\end{aligned}
\tag{5.18}
$$

where $E[min(d_i)|k = 1], E[(min(d_i)|k = 2], E[(min(d_i)|k = 3], E[(min(d_i)|k = 4]$ is the expected hop count of an aggregation node when the target is detected in zone $k, k = (1, \ldots, 4)$ and whose values are

$$E[min(d_i)|k = 1] = h \tag{5.19}$$

$$
\begin{aligned}
E[min(d_i)|k = 2] &= \frac{1}{4}[(h-1) + (h-1) + (h) + (h)] \\
&= \frac{1}{4}(4h - 2) = h - \frac{1}{2}
\end{aligned}
\tag{5.20}
$$

$$
\begin{aligned}
E[min(d_i)|k = 3] &= \frac{1}{4}[(h-1) + (h-1) + (h) + (h-1)] \\
&= \frac{1}{4}(4h - 3) = h - \frac{3}{4}
\end{aligned}
\tag{5.21}
$$

$$
\begin{aligned}
E[min(d_i)|k = 4] &= \frac{1}{4}[(h-2) + (h-1) + (h) + (h-1)] \\
&= \frac{1}{4}(4h - 4) = h - 1
\end{aligned}
\tag{5.22}
$$

Hence, the expected total number of packet transmissions in tree-based aggregation is

$$
\begin{aligned}
E[\rho_t] \geq h \times Pr(k=1) + (h+\frac{1}{2}) \times Pr(k=2) \\
+ (h+\frac{5}{4}) \times Pr(k=3) + (h+2) \times Pr(k=4) \\
= h + \frac{1}{2} \times Pr(k=2) + \frac{5}{4} \times Pr(k=3) + 2 \times Pr(k=4)
\end{aligned}
\tag{5.23}
$$

The aggregation ratio of tree-based aggregation is bounded as follows:

$$
\frac{1 \times h}{E[\rho_c]} \leq AR_t \leq \frac{1 \times h}{h + \frac{1}{2} \times Pr(k=2) + \frac{5}{4} \times Pr(k=3) + 2 \times Pr(k=4)} \tag{5.24}
$$

5.3 Static-cluster aggregation

In static-cluster aggregation, members report their sensing information to the cluster heads, and the cluster heads transmit an aggregated packet to the sink along the routing paths. As explained in section 4.3, when targets are located in the clusters' boundary areas, there is a good chance that sensors in the multiple clusters detect these targets. As a result, multiple cluster heads transmit the aggregated data packets to the sink. The total number of packet transmissions in the static-cluster aggregation is consists of packet transmissions from the cluster heads to the sink and the packet transmissions from members to their cluster heads. Therefore, the number of packet transmissions, ρ_{sc}, in static-cluster aggregation is defined as $\rho_{sc} = \sum_{i=1}^{l} d(C_i, S) + E[m]$, where $d(C_i, S) = d_i, i = (1, \ldots, l)$ is the distance, in terms of hops, between a cluster head CH_i and the sink, S. m is the total number of packet transmissions from the members to the cluster head, l is the total number of cluster heads that transmit the packet to the sink.

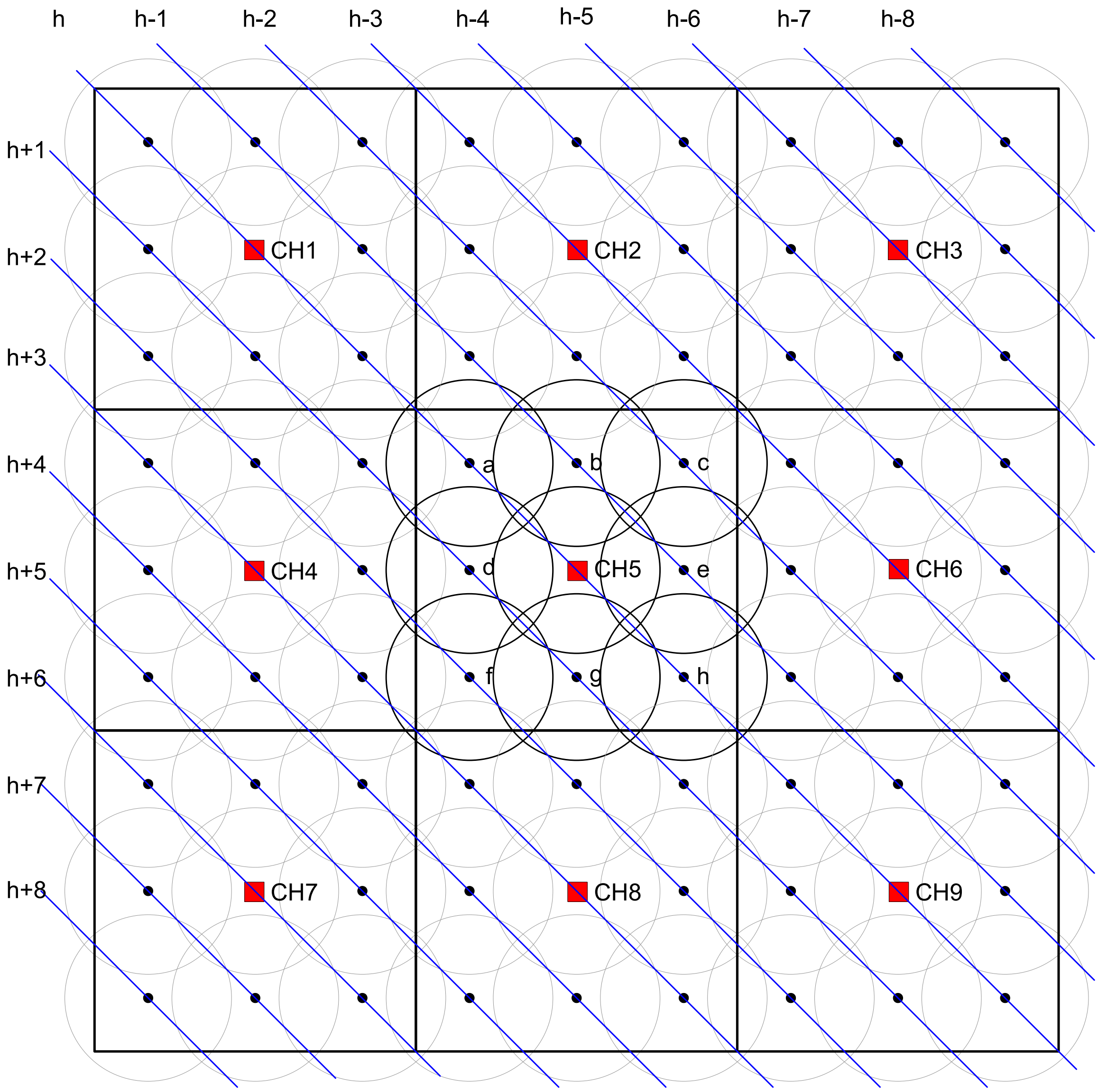


Figure 5.2: Modeling of static-cluster aggregation

Cluster structure can be described by the use of Voronoi cells. We model the

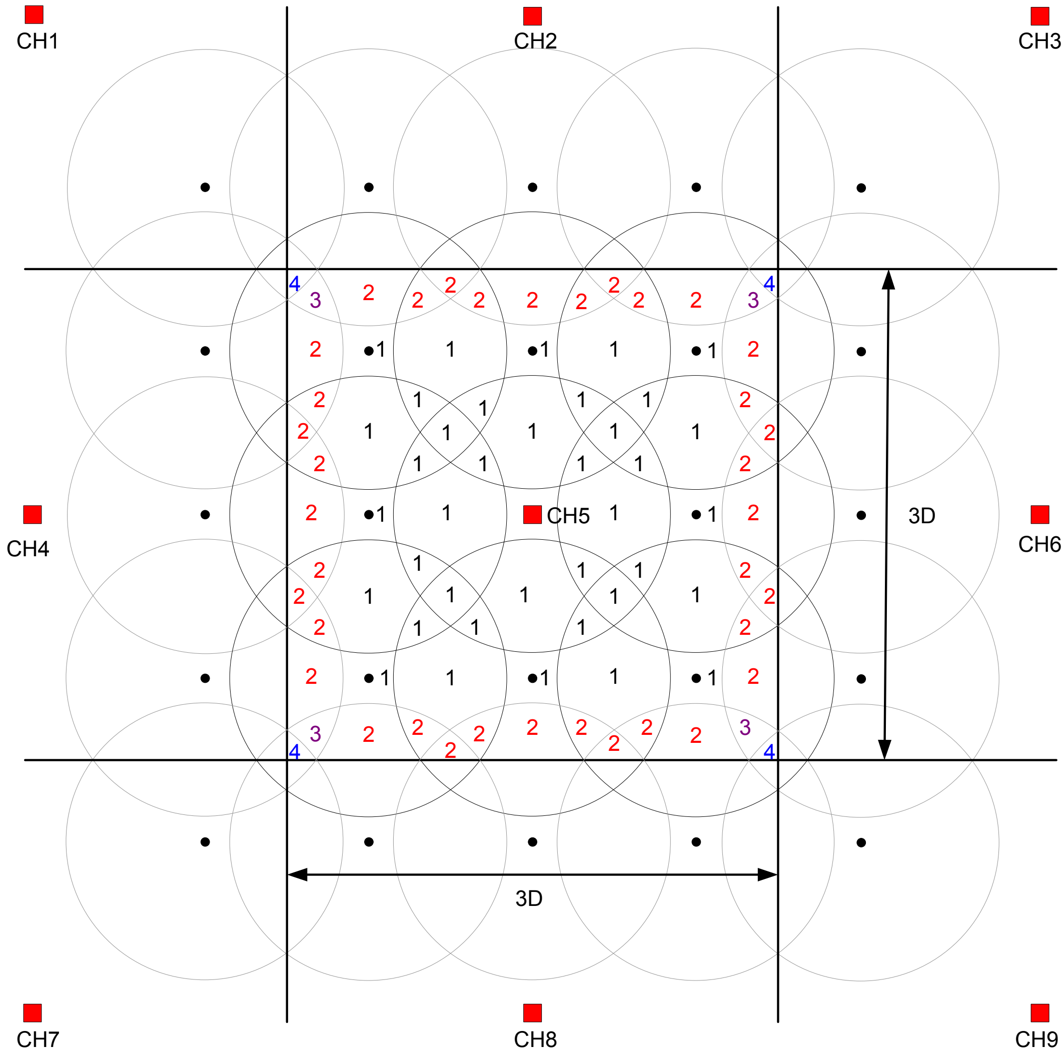


Figure 5.3: Total number of cluster heads which transmit a packet to the sink when a target is detected in each zone

clusters by square shaped Voronoi cells that have equal size. We also assume that each cluster head is located at the center of cluster. Figure 5.2 shows the modeling result of this clustering. Each cluster consists of one cluster head and eight member nodes. To keep the analysis tractable we using the same assumptions which are mentioned at the beginning of this section.

Under these conditions, when we assume that targets occur randomly in the entire sensor network, a single cluster of size $3D^2$ represents the characteristics of the entire sensor network as shown in Figure 5.3. In this Figure, the area belonging to cluster 5 has been divided into four zones, labeled 1, 2, 3 and 4. Each label indicates that if a target is detected in zone l, it will be detected by member nodes of l clusters. Hence l cluster heads will transmit the packet to the sink. From the Figure 5.3, by calculating the portion of each zone areas, we can obtain the probability that l cluster heads transmit the packets to the sink and is given by

$$\begin{aligned} Pr(l=1) &= \frac{\text{Area(indexed } l\text{=1)}}{(3D)^2} \\ Pr(l=2) &= \frac{\text{Area(indexed } l\text{=2)}}{(3D)^2} \\ Pr(l=3) &= \frac{\text{Area(indexed } l\text{=3)}}{(3D)^2} \\ Pr(l=4) &= \frac{\text{Area(indexed } l\text{=4)}}{(3D)^2} \end{aligned} \tag{5.25}$$

where Area(indexed l) is sum of areas labeled $l = 1, \ldots, 4$ and these areas can be obtained by means of geometric analysis. These areas are given by:

$$\begin{aligned}
\text{Area(indexed } l\text{=4)} = {} & 4{R_s}^2\cos^{-1}\Bigg(\sqrt{\frac{D\sqrt{{R_s}^2-\frac{D^2}{4}}+{R_s}^2}{2{R_s}^2}}\Bigg) \\
& - 4\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2-\frac{D^2}{4}}+2{R_s}^2}-\frac{D^5-2D^4\sqrt{{R_s}^2-\frac{D^2}{4}}}{2D+4\sqrt{{R_s}^2-\frac{D^2}{4}}}+\frac{D^4}{4}} \\
& - 2D\sqrt{{R_s}^2-\frac{D^2}{4}}-D^2+\frac{2D^3-4D^2\sqrt{{R_s}^2-\frac{D^2}{4}}}{D+2\sqrt{{R_s}^2-\frac{D^2}{4}}}
\end{aligned} \tag{5.26}$$

$$\begin{aligned}
\text{Area(indexed } l\text{=3)} = {} & 4{R_s}^2\cos^{-1}\Big(\frac{D}{\sqrt{2}R_s}\Big)-2\sqrt{2}D\sqrt{{R_s}^2-\frac{D^2}{2}} \\
& - 8{R_s}^2\cos^{-1}\Bigg(\sqrt{\frac{D\sqrt{{R_s}^2-\frac{D^2}{4}}+{R_s}^2}{2{R_s}^2}}\Bigg) \\
& - 8\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2-\frac{D^2}{4}}+2{R_s}^2}-\frac{D^5-2D^4\sqrt{{R_s}^2-\frac{D^2}{4}}}{2D+4\sqrt{{R_s}^2-\frac{D^2}{4}}}+\frac{D^4}{4}} \\
& - 4D\sqrt{{R_s}^2-\frac{D^2}{4}}-2D^2+\frac{4D^3-8D^2\sqrt{{R_s}^2-\frac{D^2}{4}}}{D+2\sqrt{{R_s}^2-\frac{D^2}{4}}}
\end{aligned} \tag{5.27}$$

$$
\begin{aligned}
\text{Area(indexed } l\text{=2)} = {} & 12{R_s}^2 \cos^{-1}(\frac{D}{2R_s}) - 8{R_s}^2 \cos^{-1}(\frac{D}{\sqrt{2}R_s}) \\
& - 16{R_s}^2 \cos^{-1}(\sqrt{\frac{D\sqrt{{R_s}^2 - \frac{D^2}{4}} + {R_s}^2}{2{R_s}^2}}) \\
& - 30D\sqrt{{R_s}^2 - \frac{D^2}{4}} - 12D^2 + 4\sqrt{2}D\sqrt{{R_s}^2 - \frac{D^2}{2}} \\
& + \frac{24D^3 - 48D^2\sqrt{{R_s}^2 - \frac{D^2}{4}}}{D + 2\sqrt{{R_s}^2 - \frac{D^2}{4}}} \\
& - 48\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2 - \frac{D^2}{4}} + 2{R_s}^2} - \frac{D^5 - 2D^4\sqrt{{R_s}^2 - \frac{D^2}{4}}}{2D + 4\sqrt{{R_s}^2 - \frac{D^2}{4}}} + \frac{D^4}{4}}
\end{aligned}
\tag{5.28}
$$

$$
\begin{aligned}
\text{Area(indexed } l\text{=1)} = {} & 24D^2 - 12{R_s}^2 \cos^{-1}(\frac{D}{2R_s}) - 4{R_s}^2 \cos^{-1}(\frac{D}{\sqrt{2}R_s}) \\
& + 20{R_s}^2 \cos^{-1}(\sqrt{\frac{D\sqrt{{R_s}^2 - \frac{D^2}{4}} + {R_s}^2}{2{R_s}^2}}) \\
& - 36D\sqrt{{R_s}^2 - \frac{D^2}{4}} - 2\sqrt{2}D\sqrt{{R_s}^2 - \frac{D^2}{2}} \\
& + \frac{-30D^3 + 60D^2\sqrt{{R_s}^2 - \frac{D^2}{4}}}{D + 2\sqrt{{R_s}^2 - \frac{D^2}{4}}} \\
& + 60\sqrt{\frac{D^4{R_s}^2}{2D\sqrt{{R_s}^2 - \frac{D^2}{4}} + 2{R_s}^2} - \frac{D^5 - 2D^4\sqrt{{R_s}^2 - \frac{D^2}{4}}}{2D + 4\sqrt{{R_s}^2 - \frac{D^2}{4}}} + \frac{D^4}{4}}
\end{aligned}
\tag{5.29}
$$

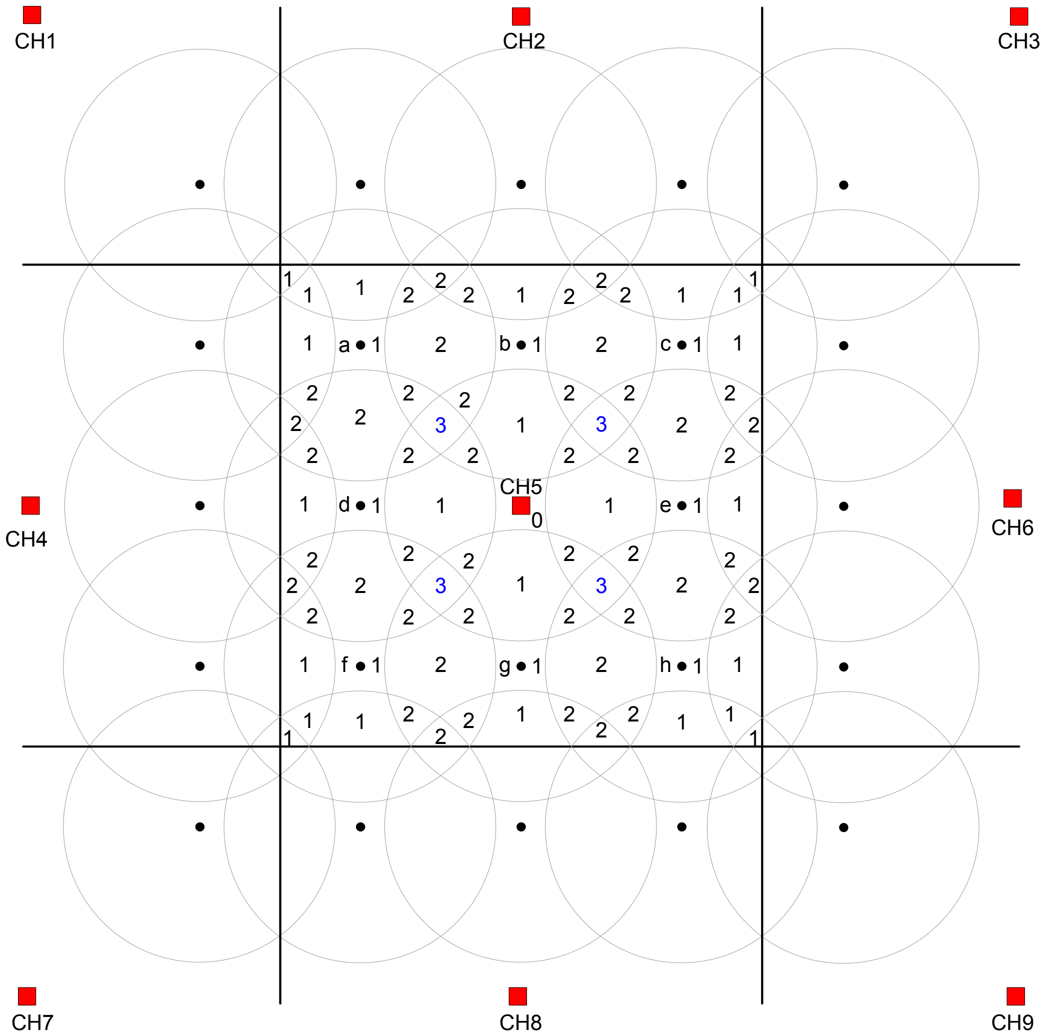


Figure 5.4: Total number of sensing nodes(members) when a target is detected in each zone

In Figure 5.4, areas indexed with $m = 0, \ldots, 3$, indicate that if a target detected in these areas, m members will detect the target, and thus m packets are transmitted to the cluster head. For example, an area indexed with 3, between nodes b, c, e, and CH5, indicates that this area is covered by three members(node b, c, and e), and these three nodes report their sensing information to the cluster head 5 whenever they detect a target. From Figure 5.4, we can calculate the conditional probability that m members transmit a packet to the cluster head given that a target is detected in zone l and is given

$$\begin{aligned}
Pr(m=0|l=1) &= \frac{\text{Area(indexed } m\text{=0} \mid l\text{=1)}}{\text{Area(indexed } l\text{=1)}} \\
Pr(m=1|l=1) &= \frac{\text{Area(indexed } m\text{=1} \mid l\text{=1)}}{\text{Area(indexed } l\text{=1)}} \\
Pr(m=2|l=1) &= \frac{\text{Area(indexed } m\text{=2} \mid l\text{=1)}}{\text{Area(indexed } l\text{=1)}} \\
Pr(m=3|l=1) &= \frac{\text{Area(indexed } m\text{=3} \mid l\text{=1)}}{\text{Area(indexed } l\text{=1)}} \\
Pr(m=1|l=2) &= \frac{\text{Area(indexed } m\text{=1} \mid l\text{=2)}}{\text{Area(indexed } l\text{=2)}} \\
Pr(m=2|l=2) &= \frac{\text{Area(indexed } m\text{=2} \mid l\text{=2)}}{\text{Area(indexed } l\text{=2)}} \\
Pr(m=1|l=3) &= 1 \\
Pr(m=1|l=4) &= 1
\end{aligned} \tag{5.30}$$

where the areas, Area(indexed $m \mid l$), can be obtained as follows using Equations 5.3 $\sim$ 5.6:

$$
\begin{aligned}
\text{Area(indexed } m\text{=0} \mid l\text{=1)} &= \text{Area(indexed 1)} \\
\text{Area(indexed } m\text{=1} \mid l\text{=1)} &= 8 \times \text{Area(indexed 1)} + 2 \times \text{Area(indexed 2)} \\
\text{Area(indexed } m\text{=2} \mid l\text{=1)} &= 4 \times \text{Area(indexed 2)} + 4 \times \text{Area(indexed 3)} \\
\text{Area(indexed } m\text{=3} \mid l\text{=1)} &= 4 \times \text{Area(indexed 4)} \qquad (5.31) \\
\text{Area(indexed } m\text{=1} \mid l\text{=2)} &= 3 \times \text{Area(indexed 2)} \\
\text{Area(indexed } m\text{=2} \mid l\text{=3)} &= 4 \times \text{Area(indexed 3)} + 4 \times \text{Area(indexed 4)}
\end{aligned}
$$

Using the Equations 6.15 $\sim$ 5.30, the expected number of packet transmissions of static-cluster aggregation is defined to be

$$
\begin{aligned}
& E[\rho_{sc}] = \\
& \quad [E[d_1|l=1] + \sum_{m=1}^{3} m \times P_r(M=m|l=1)] \times Pr(l=1) + \\
& \quad [E[d_1+d_2|l=2] + \sum_{m=1}^{2} m \times P_r(M=m|l=2)] \times Pr(l=2) + \\
& \quad [E[d_1+d_2+d_3|l=3] + 1 \times P_r(M=1|l=3)] \times Pr(l=3) + \\
& \quad [E[d_1+d_2+d_3+d_4|l=4] + 1 \times P_r(M=1|l=4)] \times Pr(l=4) \qquad (5.32)
\end{aligned}
$$

where $E[d_1|l=1], E[d_1+d_2|l=2], E[d_1+d_2+d_3|l=3], E[d_1+d_2+d_3+d_4|l=4]$ is the expected hop count of cluster heads when a target is detected in zone l, $l = (1,\ldots,4)$ and these values are

$$
\begin{aligned}
E[d_1|k=1] &= h \\
E[d_1+d_2|l=2] &= \frac{1}{4}[(h+(h-3))+(h+(h+3))+(h+(h+3))+(h+(h-3)] \\
&= \frac{1}{4}(8h) = 2h \\
E[d_1+d_2+d_3|l=3] &= \frac{1}{4}[(h+(h-3)+(h-3))+(h+(h-3)+(h+3)) \\
&\quad +(h+(h+3)+(h+3))+(h+(h+3)+(h-3)) \\
&= \frac{1}{4}12h = 3h \qquad (5.33) \\
E[d_1+d_2+d_3+d_4|l=4] &= \frac{1}{4}[(h+2(h-3)+(h-6))+(2h+(h-3)+(h+3)) \\
&\quad +(h+2(h+3)+(h+6))+(2h+(h+3)+(h-3))] \\
&= \frac{1}{4}16h = 4h
\end{aligned}
$$

If we plug in Equation 5.33 to the Equation 5.32, then the expected number of packet transmissions of static-cluster aggregation is

$$
\begin{aligned}
E[\rho_{sc}] = \\
&[h+\sum_{m=1}^{3} m \times P_r(M=m|l=1)] \times Pr(l=1) \\
&+[2h+\sum_{m=1}^{2} m \times P_r(M=m|l=2)] \times Pr(l=2) \\
&+[3h+1\times P_r(M=1|l=3)] \times Pr(l=3) \\
&+[4h+1\times P_r(M=1|l=4)] \times Pr(l=4) \qquad (5.34)
\end{aligned}
$$

The aggregation ratio of static-cluster aggregation is obtained by

$$AR_{sc} = \frac{1 \times h}{E[\rho_{sc}]} \tag{5.35}$$

5.4 Dynamic-cluster aggregation

In dynamic-cluster aggregation, sensors broadcast their sensing information to their neighbors and dynamically elected cluster head transmits an aggregated packet to the sink. Thus, total number of packet transmissions in dynamic-cluster aggregation, ρ_{dc}, is defined to be $\rho_{dc} = avg(d(S_i, S)) + (k+1)$, where $d(S_i, S) = d_i, i = (1, \ldots, k)$ is hop count between a sensing node S_i and the sink, S, $avg(d(S_i, S))$, or d_{avg}, is the average hop count of sensing nodes. $(k+1)$ is the sum of k sensing reports and one cluster head advertisement message from the elected cluster head. The expected total number of packet transmissions of dynamic-cluster aggregation, $E[\rho_{dc}]$, is

$$E[\rho_{dc}] = E[avg(d(S_i, S)) + (k+1)], i \in (1, \ldots, k) \tag{5.36}$$

As shown in Figure 5.1, the entire area can be divided into four zones in which the target is detected by the 1, 2, 3, or 4 sensors. The expected total number of packet transmissions becomes

$$
\begin{aligned}
E[\rho_{dc}] = [E[d_{avg}|k=1]+2] \times Pr(k=1) \\
+ [E[d_{avg}|k=2]+3] \times Pr(k=2) \\
+ [E[d_{avg}|k=3]+4] \times Pr(k=3) \\
+ [E[d_{avg}|k=4]+5] \times Pr(k=4)
\end{aligned}
\tag{5.37}
$$

where $E[d_{avg}|k=1], E[d_{avg}|k=2], E[d_{avg}|k=3], E[d_{avg}|k=4]$ is the expected average hop count of the sensing nodes in which the target is detected in zone $k,\ k = (1, \ldots, 4)$

$$
\begin{aligned}
&E[d_{avg}|k=1] = h \\
&E[d_{avg}|k=2] = \frac{1}{4} \times \frac{1}{2}[(h+(h-1)) + (h+(h+1)) \\
&\qquad + (h+(h+1)) + (h-(h-1))] = h \\
&E[d_{avg}|k=3] = \frac{1}{4} \times \frac{1}{3}[(h+2(h-1)) + (h+(h-1)+(h+1)) \\
&\qquad + (h+(h+1)+(h+1)) + (h+(h+1)+(h-1))] = h \\
&E[d_{avg}|k=4] = \frac{1}{4} \times \frac{1}{4}[(h+2(h-1)+(h-2)) + (2h+(h-1)+(h+1)) \\
&\qquad + (h+2(h+1)+(h+2)) + (2h+(h+1)+(h-1))] = h
\end{aligned}
\tag{5.38}
$$

Hence, the expected total number of packet transmissions of the dynamic-cluster aggregation becomes

$$\begin{aligned} E[\rho_{dc}] &= (h+2) \times Pr(k=1) + (h+3) \times Pr(k=2) \\ &\quad + (h+4) \times Pr(k=3) + (h+5) \times Pr(k=4) \\ &= (h+2) + 1 \times Pr(k=2) + 2 \times Pr(k=3) + 3 \times Pr(k=4) \end{aligned} \tag{5.39}$$

The aggregation ratio of dynamic-cluster aggregation is

$$AR_{dc} = \frac{1 \times h}{(h+2) + 1 \times Pr(k=2) + 2 \times Pr(k=3) + 3 \times Pr(k=4)} \tag{5.40}$$

5.5 Numerical result

In this section, numerical results are presented. Table 5.1 shows the summary of the expected total number of packet transmissions of the various aggregation schemes. We assume that the sensor network has a square grid layout as shown in Figure 5.1. We set the distance of each sensor, D, to 100m and vary the sensing range of nodes from 72m to 100m. To examine the effect of hop count, we test under two values of h, 2 and 10.

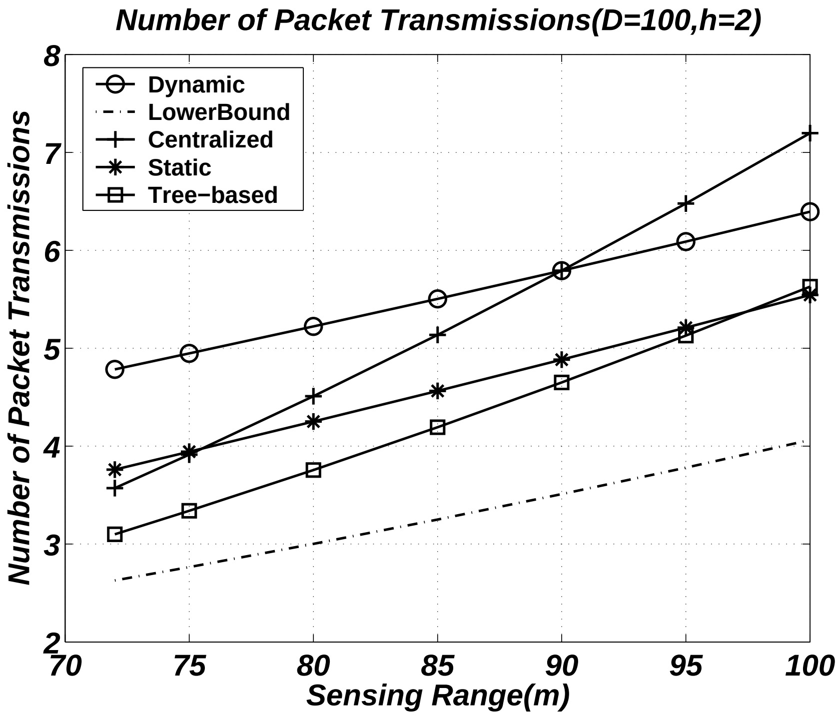
Figures 5.5 show the number of packet transmissions and aggregation ratio as a function of sensing range, when $D = 100$ and $h = 2$. In this condition, tree-

Table 5.1: The expected total number of packet transmissions

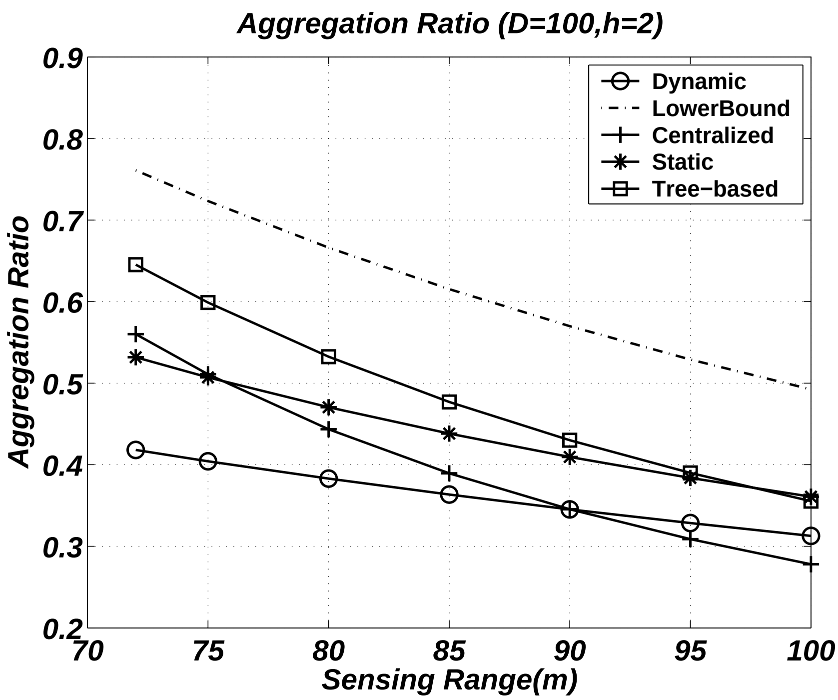
Schemes	Total number of packet transmissions
Centralized	$E[\rho_c] = h \times [Pr(k=1) + 2Pr(k=2) + 3Pr(k=3) + 4Pr(k=4)]$
Tree-based	$h + \frac{1}{2} \times Pr(k=2) + \frac{5}{4} \times Pr(k=3) + 2 \times Pr(k=4) \leq E[\rho_t] \leq E[\rho_c]$
Static	$E[\rho_{sc}] = [h + \sum_{m=1}^{3} m \times P_r(M=m \mid l=1)] \times Pr(l=1)$ $+ [2h + \sum_{m=1}^{2} m \times P_r(M=m \mid l=2)] \times Pr(l=2)$ $+ [3h + 1 \times P_r(M=1 \mid l=3)] \times Pr(l=3)$ $+ [4h + 1 \times P_r(M=1 \mid l=4)] \times Pr(l=4)$
Dynamic	$E[\rho_{dc}] = (h+2) + 1 \times Pr(k=2) + 2 \times Pr(k=3) + 3 \times Pr(k=4)$

based aggregation shows the best performance among others until the sensing range is greater than 98m. When sources are only two hops away from the sink, complex aggregation algorithms, such as static-cluster and dynamic-cluster aggregation, could not reduce packet transmissions. Moreover, these aggregation algorithms work as overhead. In static-cluster aggregation, sensing information has to deliver to the sink always via the cluster head and dynamic aggregation need additional packet transmissions to elect the cluster head dynamically. Furthermore, when the sensing range is short, centralized scheme shows the better performance than static-cluster and dynamic-cluster aggregation. As the sensing range increases, static-cluster and dynamic aggregation transmits less packets than centralized aggregation because the number of packet transmissions is reduced by the aggregation.

Figures 5.6 show the number of packet transmissions and aggregation ratios as a function of the sensing range when $D = 100$ and $h = 10$. When the sources are 10 hops away from the sink, static-cluster and dynamic-cluster aggregation show better performance than centralized and tree-based aggregation since these schemes

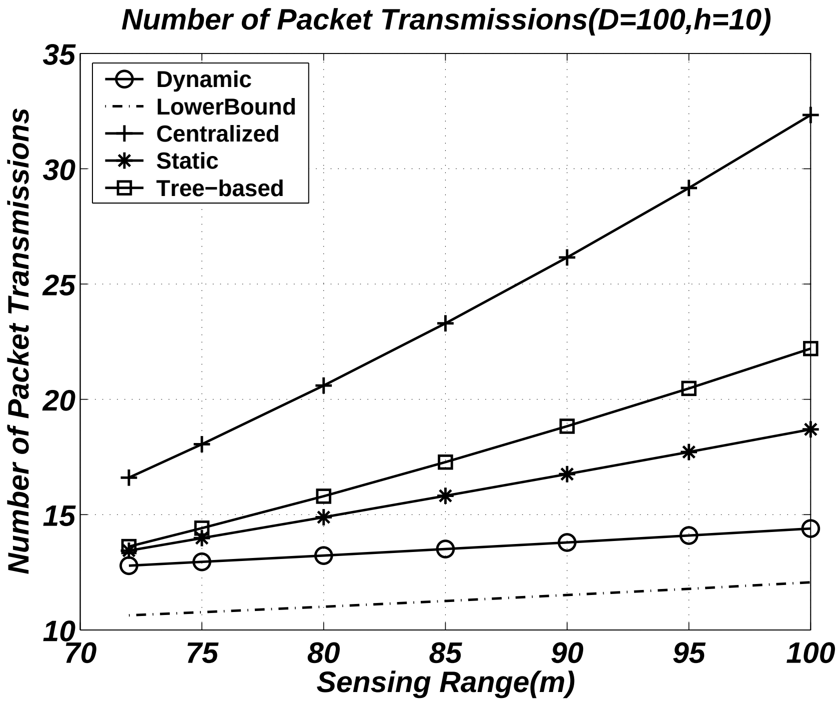


(a) Number of packet transmissions

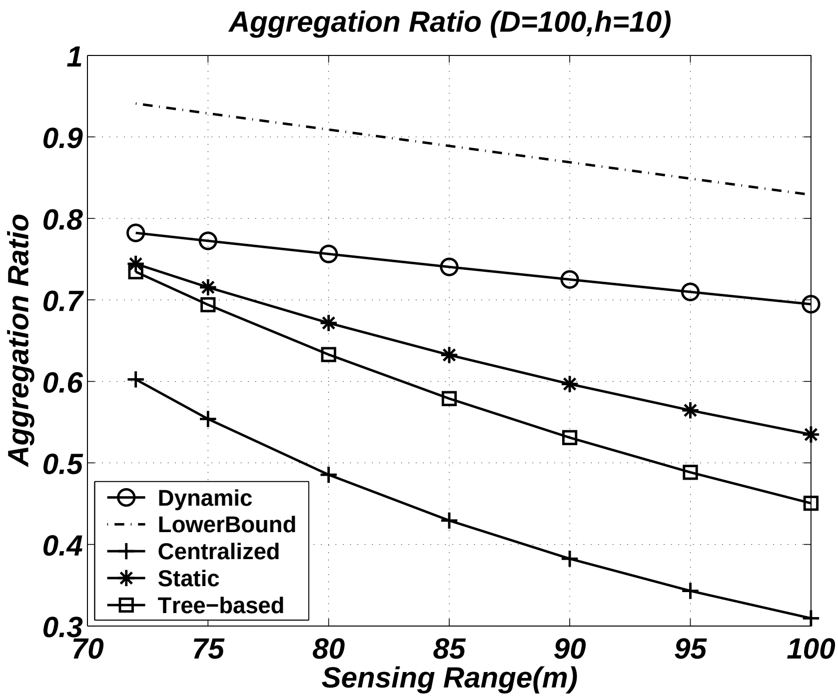


(b) Aggregation ratio

Figure 5.5: Number of packet transmissions and aggregation ratios when D=100 m and h=2



(a) Number of packet transmissions



(b) Aggregation ratio

Figure 5.6: Number of packet transmissions and aggregation ratios when D=100 m and h=10

can reduce a large amount of generated packets, near the event sensing area. Note that dynamic-cluster aggregation achieves significant performance improvement over the others and its performances are less sensitive to the sensing range.

Chapter 6

Analytical Results for Random Deployment

In this section, we derive the closed form expressions of the total number of packet transmissions and aggregation ratio for each scheme on a randomly deployed sensor network. To keep the analysis tractable, we make the following assumptions:

- N sensor nodes are distributed in a $A \times A$ area following the homogenous Poisson Point Process with intensity $\lambda = \frac{N}{A^2}$.
- The sensors present in each node follow the binary detection model, as described in Section 3.2 with sensing range R_s.
- The radio present in each node follows the two ray propagation model, as described in Section 3.3 with transmission range R_r, and $R_r > 2R_s$.

- Since $R_s < R_r/2$, we assume that all nodes sensing a target have the same distance (in terms of hop count) to the sink.
- The radio channel is loss-free.
- The MAC protocol is an idealized MAC, in that it avoids all packet collisions.

Given the Poisson Point Process distribution [15] of the nodes, and the binary detection model with sensing range R_s, the average number of sensor nodes detecting a target is given by:

$$\mu = \lambda \pi {R_s}^2 \tag{6.1}$$

Similarly, given the Poisson Process distribution of the nodes, and the two-ray propagation model with transmission range R_r, the average number of sensor nodes within radio range of an arbitrary node, is given by

$$N_R = \lambda \pi R_r^2 \tag{6.2}$$

6.1 Centralized aggregation

In centralized aggregation, all sensor nodes transmit their sensing information to the sink along their routing paths without aggregation at forwarding nodes. When a sensor generates a packet h hops away from the sink, h transmissions are required to deliver the packet to the sink. Since sensing radius is R_s, on average μ sensors detect the target. Thus the average number of data packet transmissions in the network is:

$$\rho_c = \mu \times h = \lambda \pi {R_s}^2 h \tag{6.3}$$

In the reference case, a single sensor will transmit a packet h hops to the sink. Hence the number of packet transmissions for the reference case is $h \times 1 = h$. Thus, the aggregation ratio of the centralized scheme is given by:

$$AR_c = \frac{1 \times h}{\lambda \pi {R_s}^2 \times h} = \frac{1}{\lambda \pi {R_s}^2} \tag{6.4}$$

6.2 Tree-based aggregation

In tree-based aggregation, sensing nodes transmit data packets to the sink along the established routing paths, and aggregation occurs at the points where routing paths merge. Hence, in the worst case, none of the routing paths merge, and no data aggregation occurs, as illustrated in 4.2. In this case, the number of data packet transmissions is the same as centralized aggregation. When the average hop counts of μ nodes is h, the average number of packet transmissions from the μ nodes is bounded by

$$\rho_t \leq \lambda \pi {R_s}^2 h \tag{6.5}$$

Krishnamachari et. al [17] show that if the maximum pair-wise hop count between the sensing nodes is $Y \geq 1$, the total number of data packet transmissions required for the optimal tree-based aggregation satisfies the following bounds:

$$min(d_i) + (\mu - 1) \leq \rho_t \leq (\mu - 1)Y + min(d_i) \tag{6.6}$$

where μ is the average number of sensing nodes and $min(d_i)$ is the hop count (between the target sensing nodes and the sink) of an aggregation node which has minimum hop count among sensing nodes. Note that since $R_s < 2R_r$, $Y = 1$. When the average hop count of aggregation nodes is h, the average number of packet transmissions, ρ_t, of tree-based aggregation is bounded by

$$\rho_t \geq h + (\mu - 1) = \lambda\pi {R_s}^2 + (h - 1) \tag{6.7}$$

The aggregation ratio of tree-based aggregation, AR_t, is

$$\frac{1}{\lambda\pi {R_s}^2} \leq AR_t \leq \frac{h}{\lambda\pi {R_s}^2 + (h - 1)} \tag{6.8}$$

6.3 Static-cluster aggregation

To find the aggregation ratio of the static-cluster aggregation, we need to find the average number and area of the clusters created by our clustering algorithm. According to our clustering algorithm in Section 4.3, clusters are formed at the end of the N_{ROUND} rounds. A node becomes cluster head candidate at round i with probability $k^i p_0$, where p_0 is the initial probability of being a cluster head in round 0. If node becomes cluster head candidate, it starts timer and broadcasts CHA message after the timer expired. If a node receives CHA message, it becomes a member node.

We define $P_{cov}(i)$ to be the probability that an arbitrary node *covered* in round i hears a cluster head CHA message, and becomes a member node, *or* becomes a cluster head with probability $k^i p_0$. Hence, in a round i, the probability that a node is still *uncovered* would be given by:

$$P_{uncov}(i) = 1 - \sum_{s=0}^{i-1} P_{cov}(s) \tag{6.9}$$

In round i, the number of neighbor nodes within the radio range R_r of an arbitrary node which are still uncovered is $N_{uncover-nbr}(i) = P_{uncov}(i)(N_R - 1)$. Let X be the random variable that denotes the total number of cluster heads. In our clustering algorithm, a node about to become a cluster head in a round, may suppress its own cluster head announcement message if it hears a cluster head announcement message from one of its neighbors in the same round. Therefore only one node becomes a cluster head in the same round. An arbitrary node becomes *covered* if: (1) it becomes a cluster head with probability $k^i p_0$ and none of the uncovered neighbors become cluster heads., or (2) it does not become a cluster head in round i with probability $(1 - k^i p_0)$, but one of its $N_{uncover-nbr}(i)$ uncovered neighbors becomes a cluster head, with probability $P_{nbr-ch}(X = 1, i)$.

Hence, the probability that an arbitrary node becomes covered, $P_{cov}(i)$, can be written as

$$P_{cov}(i) = P_{uncov}(i-1)[k^i p_0(1-P_{nbr-ch}(X \geq 1, i))+(1-k^i p_0)P_{nbr-ch}(X = 1, i)] \quad (6.10)$$

where $P_{nbr-ch}(X \geq 1, i)$ is the probability that at least one of its uncovered neighbors become a cluster head and is given by

$$\begin{aligned} P_{nbr-ch}(X \geq 1, i) &= P_{nbr-ch}(X = 1, i) + P_{nbr-ch}(X = 2, i) + \cdots \\ &= 1 - (1 - k^i p_0)^{N_{uncover-nbr}(i)} \end{aligned} \quad (6.11)$$

In [25], the authors show that the probability of more than two nodes becoming cluster heads in an area of radio range is negligible. Hence $P_{nbr-ch}(X = 1, i)$ is approximately equal to $P_{nbr-ch}(X \geq 1, i)$. Therefore the probability that an arbitrary node becomes covered, in equation 6.10, becomes

$$P_{cov}(i) = P_{uncov}(i-1)[k^i p_0(1-P_{nbr-ch}(X \geq 1, i))+(1-k^i p_0)P_{nbr-ch}(X \geq 1, i)] \quad (6.12)$$

When the sum of $P_{cov}(i)$ is greater than 1, all nodes are covered and become either cluster heads or members. Thus the number of rounds that is needed to cover all nodes is given by

$$N_{ROUNDS} = MIN\{s : \sum_{i=0}^{i=N_{ROUNDS}-1} P_{cov}(i) \geq 1\} \tag{6.13}$$

We now derive the expression for the expected number of cluster heads generated at the end of the clustering phase. A node would become a cluster head in round i with probability:

$$P_{CH}(i) \;\; = \;\; P_{unconv}(i)[k^i p_0(1 - P_{nbr-ch}(i))]$$

The total probability that a node becomes a cluster head in N_{ROUNDS} rounds is:

$$P_{CH} \;\; = \;\; \sum_{i=0}^{N_{ROUNDS}-1} P_{CH}(i)$$

Hence the expected number of cluster heads generated in N_{ROUNDS} rounds is:

$$\begin{aligned} E[\#CH] &= \sum_{i=1}^{N}[P_{CH} \times 1] \\ &= NP_{CH} \\ &= N \sum_{i=0}^{N_{ROUNDS}-1} k^i p_0\{1 - P_{nbr-ch}(i)\}\{1 - \sum_{t=0}^{i-1} P_{cov}(t)\} \end{aligned}$$

We assume that an entire area of the sensor network is much larger than a cluster area and the sensor network is divided into multiple clusters which have the same size. By dividing the entire area $A \times A$ into the average number of cluster heads, the average size of the clusters is given by

$$E[Area_{cluster}] = \frac{A^2}{E[\#CH]} \tag{6.14}$$

Cluster structures can be described by the use of Voronoi cells. Planar Voronoi cells are convex polygons. From the properties of the Voronoi diagram [11], [3] the average number of Voronoi edges per Voronoi polygon is 6. Since we assumed that clusters have the same size, we can model the Voronoi cells as hexagonal polygons. As shown in Figure 6.1, randomly deployed sensors are divided into hexagonal shaped clusters and cluster heads, CH1 $\sim$ CH7, are located at the center of each cluster.

In static-cluster aggregation, the total number of packet transmissions consists of two components: (1) packet transmissions from members to the sink and (2) packet transmissions of the aggregated packet from the cluster heads to the sink.

Packet Transmissions from the Cluster Heads to the Sink: We derive the number of packet transmissions from the cluster heads to the sink by means of geometric analysis. We divide the area belonging to a cluster into three zones. As shown in Figure 6.2(a), the area belonging to cluster head 1 has been divided into three

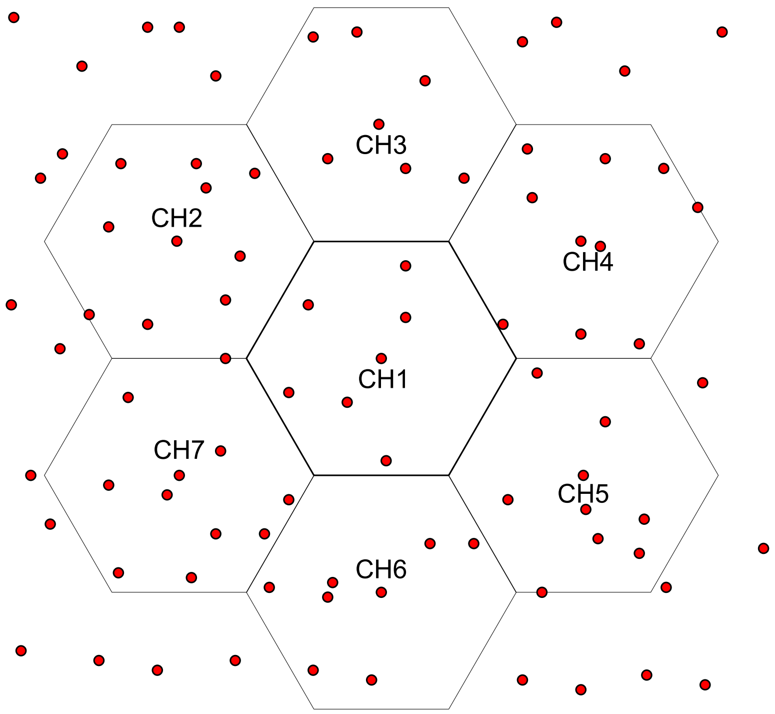


Figure 6.1: Modeling of sensor network using the hexagonal shaped Voronoi cells

zones, labeled 1, 2 and 3. The label of a zone has the following meaning: if a target appears in zone l, it will be detected by member nodes of *at most* l clusters. Since member nodes detecting events transmit the event data packets to their respective cluster heads, the label l on a zone indicates that at most l cluster heads will receive event data if a target/event appears in zone l. Hence the probability that up to l cluster heads transmit the packet to the sink is given by:

$$\begin{aligned} Pr(l=1) &= \frac{\text{Area(indexed } l\text{=1)}}{E[Area_{cluster}]} \\ Pr(l=2) &= \frac{\text{Area(indexed } l\text{=2)}}{E[Area_{cluster}]} \\ Pr(l=3) &= \frac{\text{Area(indexed } l\text{=3)}}{E[Area_{cluster}]} \end{aligned} \tag{6.15}$$

where Area(indexed l) is the size of the zones indexed with the $l = 1, \ldots, 3$ in a cluster.

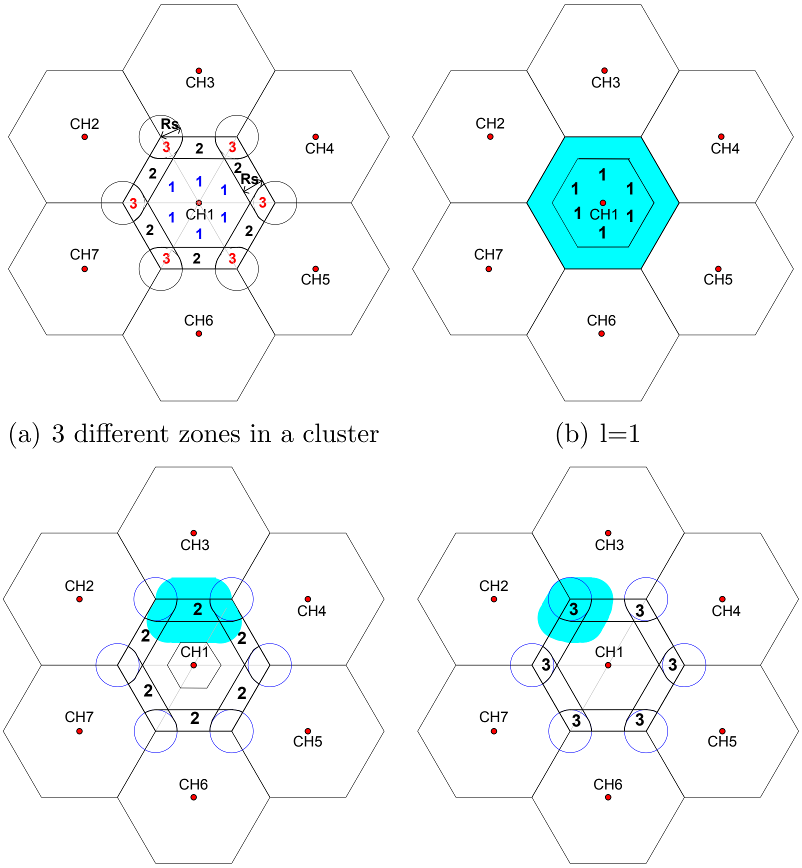
Consider the case when a target appears in zone 1. In Figure 6.2(b), the shaded area shows the possible location of nodes that detect this target. Clearly, only nodes in cluster 1 can detect this target. Hence, the number of nodes detecting this target is given by μ. The probability that *at least* one node detects this target and transmits a packet to the cluster head 1 is given by:

$$Pr(Cluster_1|l=1) = \begin{cases} 1 & \text{if } \mu \geq 1 \\ \mu & \text{otherwise} \end{cases} \tag{6.16}$$

Hence the average number of packet transmissions from a cluster head to the sink when a target is detected in zone 1 is given by:

$$\rho_{CH}(l=1) = h \times Pr(Cluster_1|l=1) \times P_r(l=1) \tag{6.17}$$

where h is the number of hop counts from cluster head to the sink.



(a) 3 different zones in a cluster (b) l=1

(c) l=2 (d) l=3

Figure 6.2: Location of sensing nodes that detect targets in zones indexed with 1, 2, and 3

Now consider the case when the target appears in zone 2. In Figure 6.2(c), the shaded area shows the possible location of nodes that detect this target. Clearly, some nodes in cluster 3 in addition to those in cluster 1 may detect the target. Hence, out of the μ nodes that detect the target, the probability that *at least one* node belongs to cluster 1 is given by:

$$Pr(Cluster_1|l=2) = \begin{cases} 1 & \text{if } \mu \times \frac{Area(Cluster_1|l=2)}{Area(Total|l=2)} \geq 1 \\ \mu \times \frac{Area(Cluster_1|l=2)}{Area(Total|l=2)} & \text{otherwise} \end{cases} \tag{6.18}$$

where $Area(Total|l = 2)$ is the entire shaded area in Figure 6.2(c) and $Area(Cluster_1|l = 2)$ is the portion of this shaded area which lies inside cluster 1.

By a similar reasoning, the probability that *at least one* node in the cluster 3(neighbor of cluster 1) detects the target is given by:

$$Pr(Neighbor|l=2) = \begin{cases} 1 & \text{if } \mu \times \frac{Area(Neighbor|l=2)}{Area(Total|l=2)} \geq 1 \\ \mu \times \frac{Area(Neighbor|l=2)}{Area(Total|l=2)} & \text{otherwise} \end{cases} \tag{6.19}$$

where $Area(Neighbor|l = 2)$ is the portion of the shaded area in Figure 6.2(c) which lies inside cluster 3.

Hence, for the case when the target appears in zone 2, the average number of packet transmissions is given by:

$$\rho_{CH}(l=2) = h \times [Pr(Cluster_1|l=2) \times 1 + Pr(Neighbors|l=2) \times 1] \times P_r(l=2) \tag{6.20}$$

Now consider the case when the target appears in zone 3. In Figure 6.2(d), the shaded area shows the possible location of nodes that detect this target. In this case, sensors in three clusters may detect the target. Some nodes in cluster 2, cluster 3 and cluster 1 may detect the target. By applying similar geometric arguments we can derive the expression for the probability that *at least one* node belongs to cluster 1, and two neighbors are given by:

$$Pr(Cluster_1|l=3) = \begin{cases} 1 & \text{if } \mu \times \frac{Area(Cluster_1|l=3)}{Area(Total|l=3)} \geq 1 \\ \mu \times \frac{Area(Cluster_1|l=3)}{Area(Total|l=3)} & \text{otherwise} \end{cases} \tag{6.21}$$

$$Pr(Neighbor_1|l=3) = \begin{cases} 1 & \text{if } \mu \times \frac{Area(Neighbor_1|l=3)}{Area(Total|l=3)} \geq 1 \\ \mu \times \frac{Area(Neighbor_1|l=3)}{Area(Total|l=3)} & \text{otherwise} \end{cases} \tag{6.22}$$

$$Pr(Neighbor_2|l=3) = \begin{cases} 1 & \text{if } \mu \times \frac{Area(Neighbor_2|l=3)}{Area(Total|l=3)} \geq 1 \\ \mu \times \frac{Area(Neighbor_2|l=3)}{Area(Total|l=3)} & \text{otherwise} \end{cases} \tag{6.23}$$

Hence the average number of packet transmissions when the target appears in zone 3, denoted by $\rho_{CH}(l=3)$.

$$\rho_{CH}(l=3) = h \times [Pr(Cluster_1|l=3) \times 1 + Pr(Neighbor_1|l=3) \times 1 + Pr(Neighbor_2|l=3) \times 1] \times P_r(l=3) \tag{6.24}$$

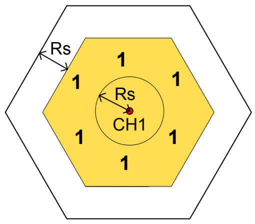


Figure 6.3: Two different zones when a target is detected by the cluster 1

Packet Transmissions from Members to the Cluster Head: Again, consider the case where the target appears in zone 1. Clearly, μ member nodes will detect the target and transmit event data packets to the cluster head. But note that the cluster head itself is a special case, since it clearly does not transmit a packet to itself. Hence zone 1 can be further divided into two zones as shown in Figure 6.3: a circular area centered at the cluster head 1 with the radius R_s and remaining area. When a target is detected in the circular area, cluster head 1 is always included in the average number of sensors that detect the target. Hence the average number of members in this zone becomes $\mu - 1$. Therefore when the target appears in zone 1, the average number of packet transmissions from members to the cluster head can be written as the following weighted sum:

$$\begin{aligned}\rho_m(l=1) = [(\mu-1)&\frac{\pi {R_s}^2}{\text{Area(indexed } l\text{=1)}} \\ &+ \mu\frac{\text{Area(indexed } l\text{=1)} - \pi {R_s}^2}{\text{Area(indexed } l\text{=1)}}] \times P_r(l=1)\end{aligned} \tag{6.25}$$

Now consider the case where the target appears in zone 2. Then the average number of packet transmissions from nodes which are members of cluster 1 can be written as:

$$\rho_m(l=2) = [\mu \times \frac{Area(Cluster_1|l=2)}{Area(Total|l=2)}] \times P_r(l=2) \tag{6.26}$$

where $Area(Cluster_1|l = 2)$ is the shaded area on the cluster 1 area and the $Area(Total|l = 2)$ is the total shaded area in Figure 6.2(c). A similar reasoning can be applied to find the average number of packet transmissions for the case where the target appears in zone 3, denoted by $\rho_m(l = 3)$ and is given by:

$$\rho_m(l=3) = [\mu \times \frac{Area(Cluster_1|l=3)}{Area(Total|l=3)}] \times P_r(l=3) \tag{6.27}$$

where $Area(Cluster_1|l = 3)$ is the shaded area on the cluster 1 area and the $Area(Total|l = 3)$ is the total shaded area in Figure 6.2(d).

Finally, the total number of packet transmissions in the system would be the sum of the two components:

$$\rho_{sc} = \sum_{k=1}^{3} (\rho_{CH}(l = k) + \rho_m(l = k)) \tag{6.28}$$

The aggregation ratio of static-cluster aggregation is

$$AR_s = \frac{h}{\rho_{sc}} \tag{6.29}$$

6.4 Dynamic-cluster aggregation

In dynamic-cluster aggregation, sensors broadcast their sensing information to their neighbors and a dynamically elected single cluster head transmits the aggregated packet to the sink. Hence, the average number of packet transmissions in dynamic-cluster aggregation can be written as:

$$\rho_{dc} = avg(d(S_i, S)) + (\mu + 1) \approx h + \lambda \pi {R_s}^2 + 1 \tag{6.30}$$

where $avg(d(S_i, S)) = h$, is the average distance from the nodes that detect the target to the sink. $(\mu + 1)$ is the sum of μ sensing reports from nodes that detect target and one cluster head advertisement message from the elected cluster head.

Hence, the aggregation ratio of dynamic-cluster aggregation is given by

$$AR_d = \frac{h}{h + \lambda\pi {R_s}^2 + 1} \tag{6.31}$$

6.5 Numerical result

In this section, we present numerical results of the analysis presented in the previous section. Figures 6.4 and 6.5 show the total number of packet transmissions and the consequent aggregation ratio obtained when using centralized, tree-based, static-cluster and dynamic-cluster aggregation schemes while varying the sensing range R_s from 20m to 45m. We set the network density to $\lambda = 0.0008$, and radio range $R_r = 150$m (Note that for all cases, $R_r > 2R_s$). In this experiment, we set the event to occur at a distance of $h = 10$ hops away from the sink.

When the sensing range is small, centralized and tree-based aggregation show better performance(in terms of the number of packet transmissions being small) than static-cluster and dynamic-cluster aggregation. In this condition, static-cluster and dynamic-cluster aggregation cannot take advantage of data aggregation because the amount of overhead exceeds the packet reduction by data aggregation.

As the sensing range increases, cluster-based aggregations begin taking advantage of data aggregation. When the sensing range is $45m$, dynamic-cluster aggregation achieves significant performance improvement over the others. Note that the perfor-

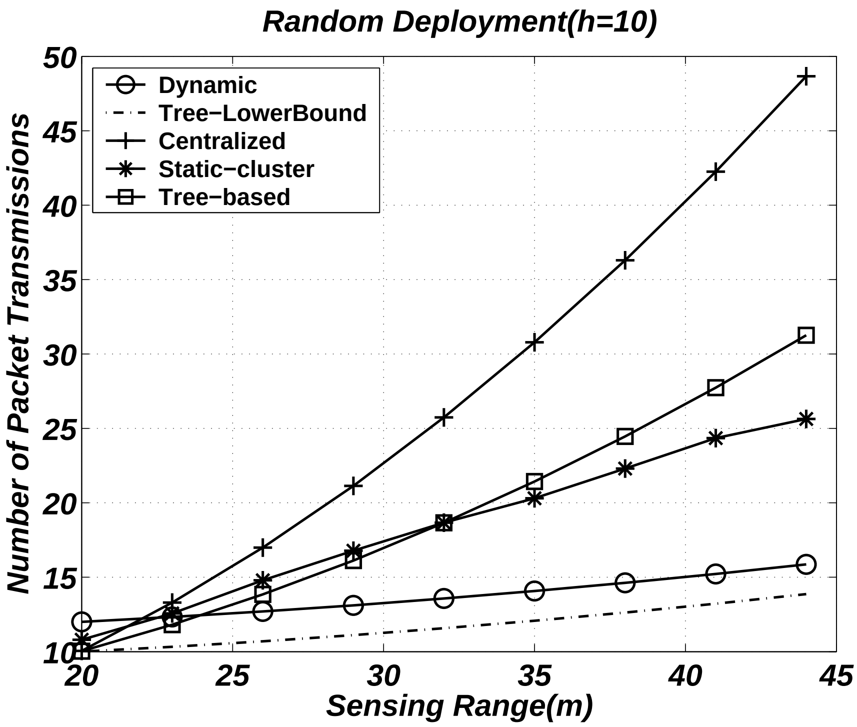


Figure 6.4: Total number of packet transmissions when h=10

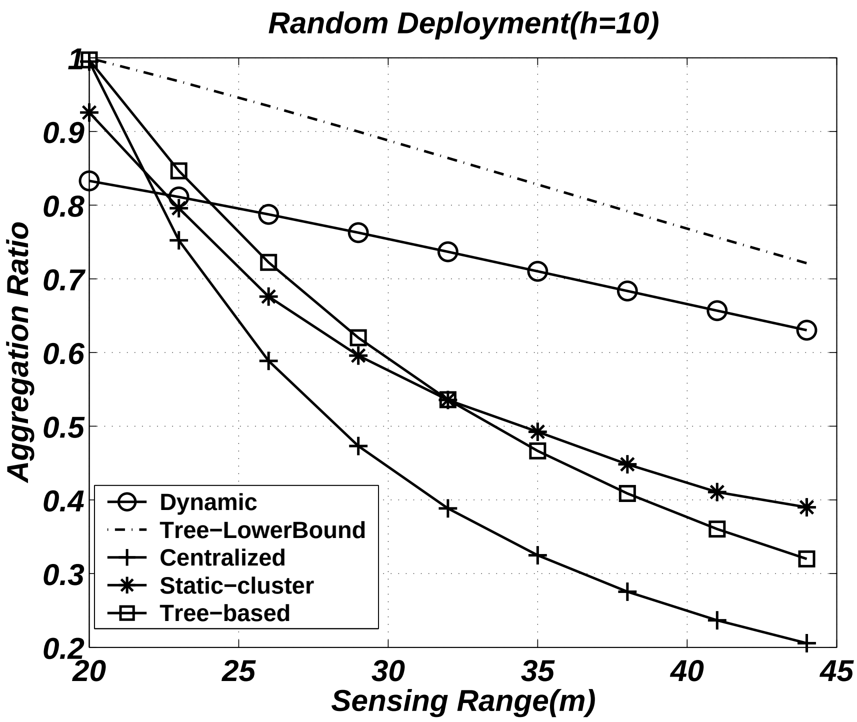


Figure 6.5: Aggregation ratio when h=10

mance of dynamic-cluster aggregation remains high regardless of the sensing range, while the performance of other aggregation schemes deteriorates. This is due to the fact that regardless of the sensing range, the dynamic-cluster aggregation always generates just one cluster head, which transmits one aggregated data packet to the sink per event.

Somewhat disappointingly, static-cluster aggregation does not perform superior to the tree-based aggregation. The reason for this is as follows. First, as the sensing range increases, the probability that an event is detected by member nodes belonging to more than one cluster increases. Second, routes computed over the cluster backbone may be sub-optimal in terms of number of hops. Hence the number of packets transmitted to the sink increases correspondingly.

Chapter 7

Simulations

In the previous section, we analyzed the performance of data aggregation under several simplifying assumptions. In particular, we assumed a loss-free channel, a collision-free MAC and a binary sensor detection model. Our analysis of tree-based aggregation was limited to upper and lower bounds and we did not consider the effect of aggregation delay. Finally, our analysis of static-cluster aggregation relied on assumptions about the shape and size of clusters and the position of the cluster head to simplify the geometric analysis.

In this section we study the performance of the data aggregation schemes in a simulated environment. We have the following goals: (1) Compare the performance of data aggregation algorithms with various metrics (2) Complement the limitations of our analytical results (3) Validate our proposed static-cluster and dynamic-cluster aggregation algorithms (4) Examine the impact of the bandwidth and MAC layer on the performance of the data aggregation. To conduct the simulations, we use NRL-SENSORSIM [6] which is a sensor network extension of ns-2 [2]. In NRL-SENSORSIM, sensor network events are simulated by the broadcast of event packets, also called PHENOM packets, by particular nodes called PHENOM nodes. Note that varying the transmission range of such PHENOM nodes is equivalent to varying the sensing range of the sensor nodes.

7.1 Performance Metrics

We select following metrics to evaluate the performance of various data aggregation schemes:

1. **Aggregation ratio:** The aggregation ratio for a particular aggregation scheme, as defined in Section 6.

2. **Average radio energy consumption:** The average radio energy consumption per node is measured as the total radio energy consumption in the network divided by the total number of nodes.

3. **Network lifetime:** The network lifetime is defined to be the time duration from the beginning of the simulation to the time until at least one node in the network depletes its battery capacity to 10% of its initial capacity.

4. **Average end-to-end delay:** The average end-to-end delay measures the average one-way latency from the time that an event occurs till the event report

data packet reaches the sink.

5. **Data packet delivery ratio:** For the centralized aggregation scheme, the data delivery ratio is defined to be simply the number of packets which arrive at the sink to the total number of packets sent by the sources. However, for tree and cluster-based schemes, we need to define data delivery ratio in a different manner. Consider, for instance a node X in tree-based aggregation, which receives packets a and b from sources A and B reporting the same event. Hence, assuming perfect aggregation, X will *aggregate* a and b and create a new packet x, which it will then transmits toward the sink. Hence, at node X, one redundant packet has essentially been dropped. For book-keeping purposes, every node which performs such data aggregation maintains the count of such redundant packets that it drops and records it inside the outgoing aggregated packet as a field called *fusing count.* If a packet undergoes multiple aggregations along the path to the sink, the fusing count is incrementally updated at each aggregation node. Finally, at the sink, the data delivery ratio is defined to be the ratio of the sum of the number of packets arriving at the sink and the respective fusing count inside each arrived packet, to the total number of packets sent by the sources.

7.2 Simulation details

We create 30 different network topologies by randomly placing 200 nodes in a $500m \times 500m$ area. The sink is always located at the top-right corner of this area. We use the directed diffusion [13] algorithm to establish low-latency routing paths from sources to the sink. To examine the performance of aggregation schemes under different traffic loads, we change the power level of PHENOM nodes from $0.1mW$ to $1.3mW$. This is equivalent to varying the sensing range of nodes from $13m$ to $48m$. As the sensing range increases, more sensors detect the target and the average number of sensors detecting the event varies from 0.35 to 5.12.

Table 7.1: Simulation Environment

Parameter	value
Routing Protocol	Diffusion
MAC Protocol	IEEE 802.11b
Radio Transmission Power	0.375mW
Radio Reception Power	0.325mW
Radio Idle Power	0.0375mW
Sensing Power	0.0325mW
Radio Propagation Model	Two-Ray
Packet Size	100 bytes
Data Rate	1Mbps
Radio Range	90 meter
Sensing Range	$13 \sim 48$ meter
Area	$500 \times 500\ m^2$
Number of nodes	200

We generate events in the following manner. Every second, 8 targets are generated at random positions in the $500m \times 500m$ area, which remain at those positions for 1 second. Hence, when the sensing range is $48m$, about 41 nodes among 200 nodes detect the targets, and act as data sources in our experiment. The data rate is set to $250Kbps$. To examine the impact of different data rates, we also test under $1Mbps$ data rate. We use IEEE 802.11b as MAC protocol. For the purpose of investigating the impact of the MAC layer, we also test under B-MAC [20] which is the default MAC protocol used in the TinyOS [18] operating system for sensor networks. The energy dissipation rate at each node follows the specification for MICA2 motes [1]. Each experiment runs for 100 seconds and we report the average and 95% confidence intervals. The parameters we used for our simulations are presented in Table 7.1.

7.3 Aggregation Ratio

The number of packet transmissions represents how many packets are transmitted from the entire sensor nodes during the simulation time. Figure 7.1 shows the number of packet transmissions as we vary the sensing range. In this Figure, when the sensing range is short, it is not easy to compare the performance with others because their graphs are overlapped. But aggregation ratios, shown in Figure 7.2, make the comparison easy.

When the sensing range is short, tree-based aggregation with 50 *msec* aggregation delay(Tree-50), shows the best performance than others. In this condition, routing paths of static-cluster aggregation are sub-optimal since sensing information has to be delivered to the sink via the cluster heads. Therefore static-cluster aggregation transmits more packets than tree-based aggregation. Dynamic-cluster aggregation

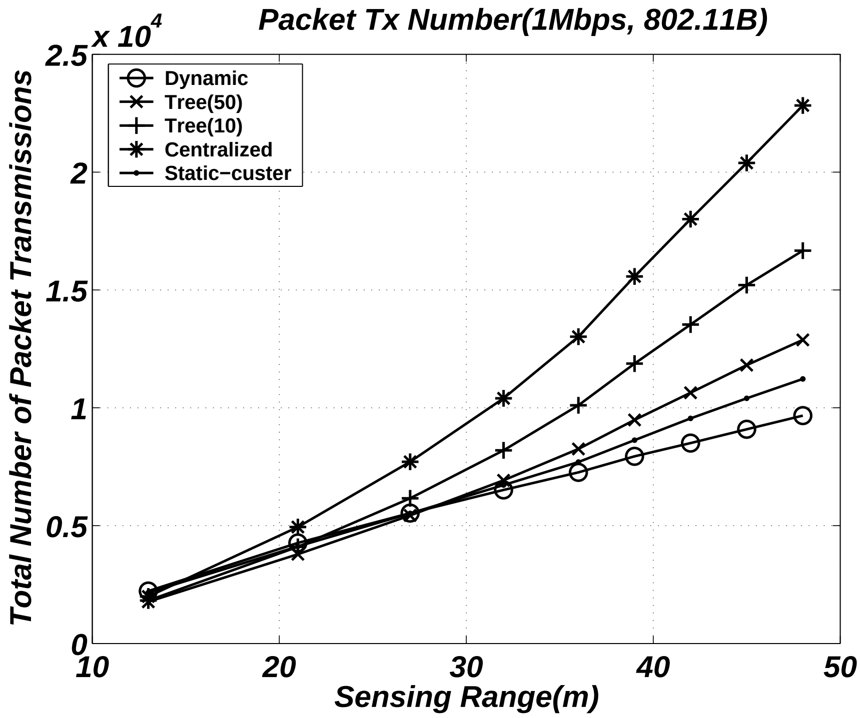


Figure 7.1: Number of packet transmissions(1 Mbps, IEEE 802.11)

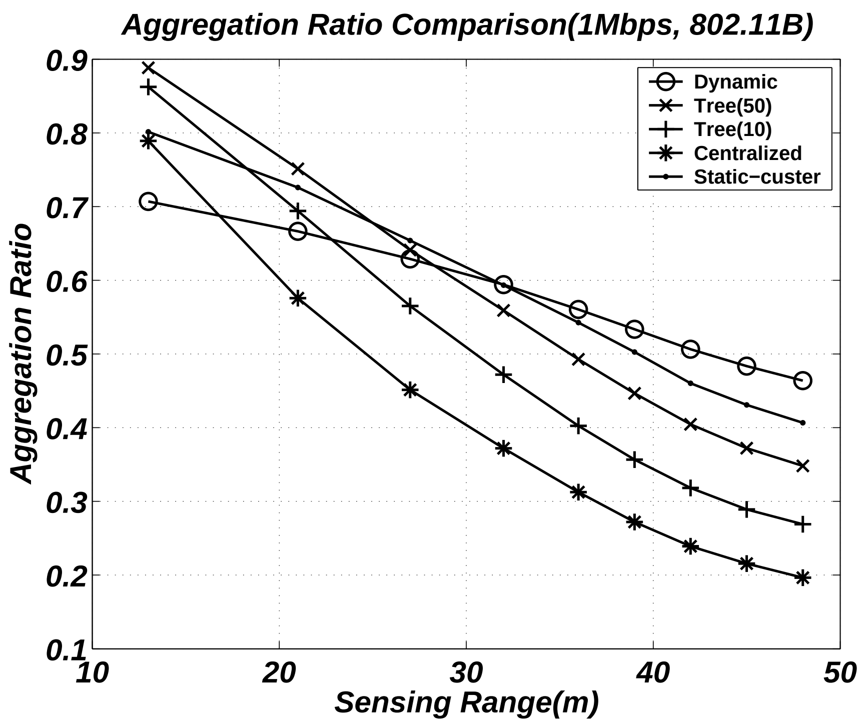


Figure 7.2: Data Aggregation Ratio(1 Mbps, IEEE 802.11)

also transmits more packets than tree-based aggregation under this condition because this scheme needs packet transmissions for the cluster head election process without take advantage of data aggregation benefits. As the sensing range increases, Tree-50 aggregation reduces the packet transmissions up to 25% than Tree-10 aggregation because the longer aggregation delay provides the more chances of aggregation to the forwarding nodes. Under this condition, static-cluster and dynamic-cluster aggregation can further reduce packet transmissions by efficient data aggregation. Static-cluster and dynamic-cluster aggregation show up to 13% and 25% of packet reduction respectively, over the Tree-50 aggregation. When the sensing range is $48m$, dynamic cluster aggregation achieves up to 0.4638 aggregation ratio, static cluster aggregation achieves up to 0.4067, Tree-50 and Tree-10 aggregation achieve up to 0.3481 and 0.269, respectively.

7.4 Average energy consumption

Figure 7.3 shows the average energy consumption per node as a function of the sensing range. Note that here the curve for static-cluster aggregation includes the energy for cluster setup. This overhead causes the static-cluster aggregation scheme to lose some of its performance for short sensing ranges. As the sensing range increases, static-cluster aggregation begins to perform better than tree-based aggregation. While it may seem that the tree-50 aggregation scheme provides good performance without the need for clustering, this performance is at the cost of end-to-end

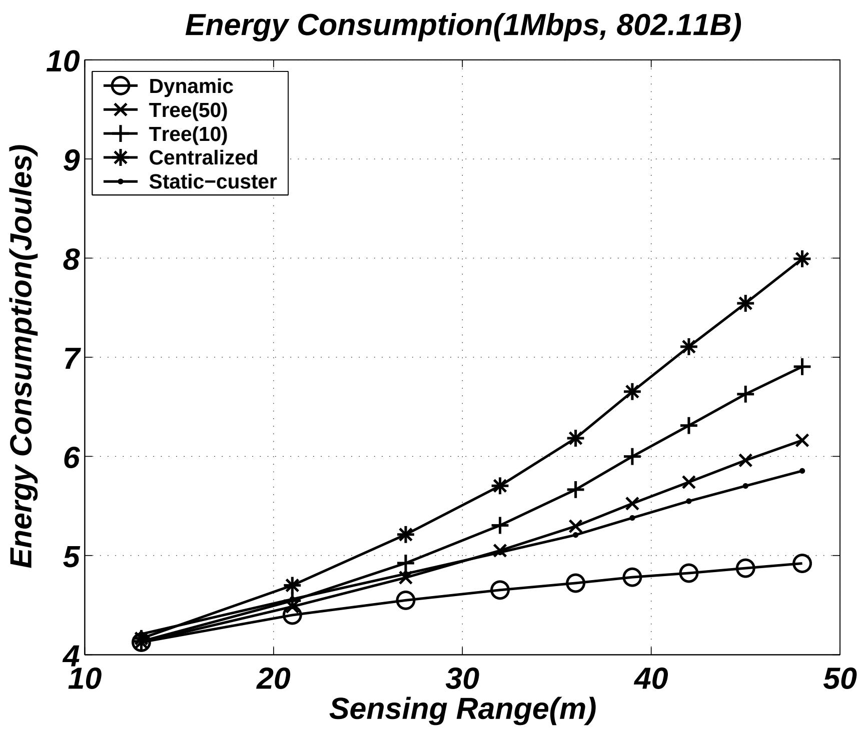


Figure 7.3: Average energy consumption(1 Mbps, IEEE 802.11)

delay, as shall be seen in Section 7.6.

The dynamic-cluster aggregation scheme achieves up to 15% to 38% less average energy consumption than others. Again, we notice that the performance of dynamic-cluster is less sensitive to the sensing range.

7.5 Network lifetime

The network lifetime is influenced by not only the average energy consumption but also energy balancing among nodes. The energy balancing of nodes can be measured by the standard deviation of average energy consumption among sensor nodes. Figure

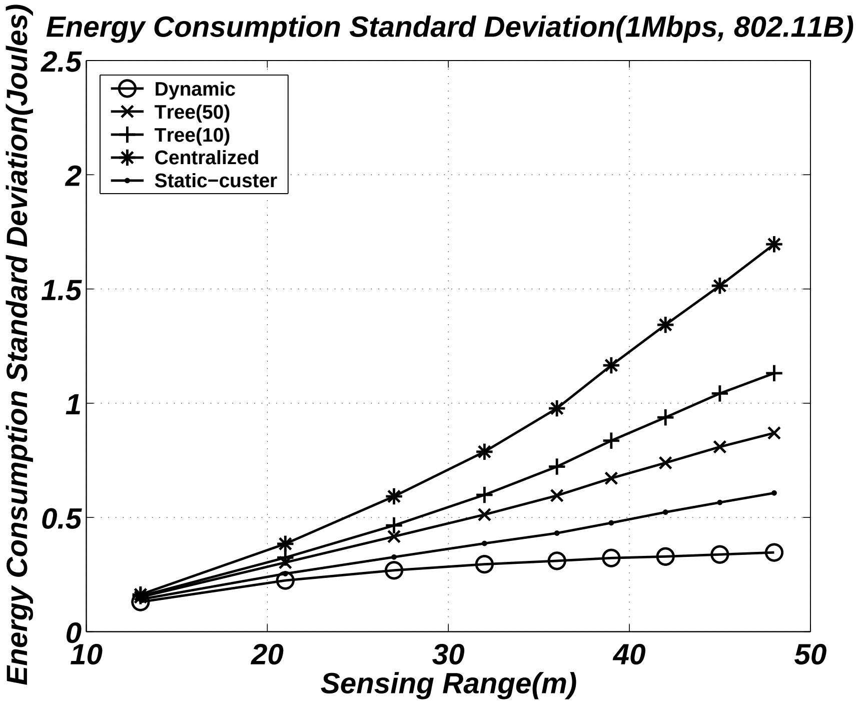


Figure 7.4: Standard deviation of energy consumption(1 Mbps, IEEE 802.11)

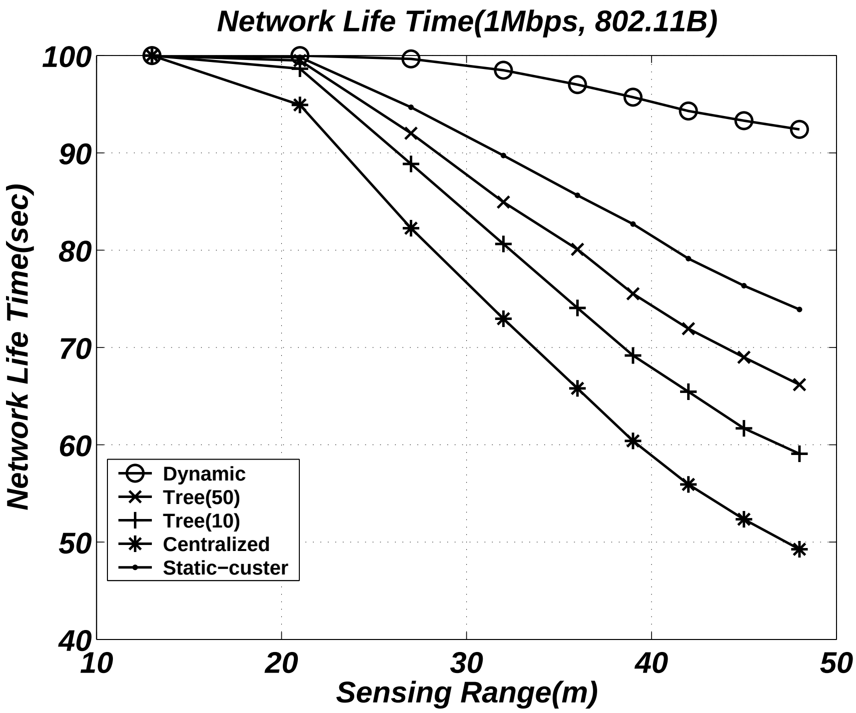


Figure 7.5: Network lifetime(1 Mbps, IEEE 802.11)

7.4 shows the standard deviation of the average energy consumption. Note that centralized aggregation shows the highest deviation. In centralized aggregation, when events are detected by sensors, they transmit packets along routing packets without aggregation at intermediate nodes. Hence nodes along routing paths experience high energy depletion, while the rest of the nodes remain more or less unaffected, which brings about high deviation.

This unbalance is mitigated to some extent if aggregation is performed at intermediate nodes and only aggregated packets are sent along routing paths, which is essentially what tree-based aggregation does. However, note that routing paths may not merge at any intermediate aggregation point, or may merge several hops away from the region where the event occurs, or worse, the data packets reporting an event from multiple sources may reach an aggregation point separated by delay more than the aggregation delay, rendering aggregation impossible. Because of these factors, tree-aggregation still exhibits high deviation.

By using clustering, we force aggregation to be performed near the region where the event occurs, hence we can see a corresponding decrease in deviation. These same factors affect the network lifetime as shown in Figure 7.5, allowing dynamic-cluster aggregation to achieve up to 26% more lifetime than static-cluster aggregation, and up to 39% more than Tree-50 aggregation, and up to 88% more than centralized aggregation.

7.6 Average end-to-end delay

The end-to-end delay is affected by four factors – first, under network congestion, data packets may incur high queueing delay at intermediate forwarding nodes; second, for the case of tree-based aggregation, delay is artificially induced due to the aggregation delay at aggregation points, as described in Section 4.2; third, for the case of dynamic-cluster aggregation, delay is induced due to the cluster head election process, and fourth, for the case of static-cluster aggregation, a cluster head needs to wait a fixed amount of time so that it is assured of receiving all sensing data packets from its members before performing data aggregation.

The effect of queueing delay can be seen for the case of centralized aggregation in Figure 7.6. Since there is no aggregation at intermediate nodes, as the sensing range increases, the number of packet transmissions also increase, causing high congestion, and consequently long delay. A similar curve is seen for tree-based aggregation with aggregation delay 10ms, which indicates that $10ms$ is not enough to aggregate all packets for an event, causing congestion and hence queueing delay. With $50ms$ aggregation delay, tree-aggregation shows less sensitivity to sensing range, which indicates that the the aggregation delay is enough to buffer all packets for an event and perform data aggregation, however, this leads to high end-to-end delay of about $250ms$. In the case of dynamic cluster aggregation, cluster head election process delay is about $150ms$ in our simulation, hence this results in end-to-end delay of about $185ms$. In static-cluster aggregation, due to the queuing delay and the cluster head waiting time(50ms in our simulation), end-to-end delay is 86ms to 116ms.

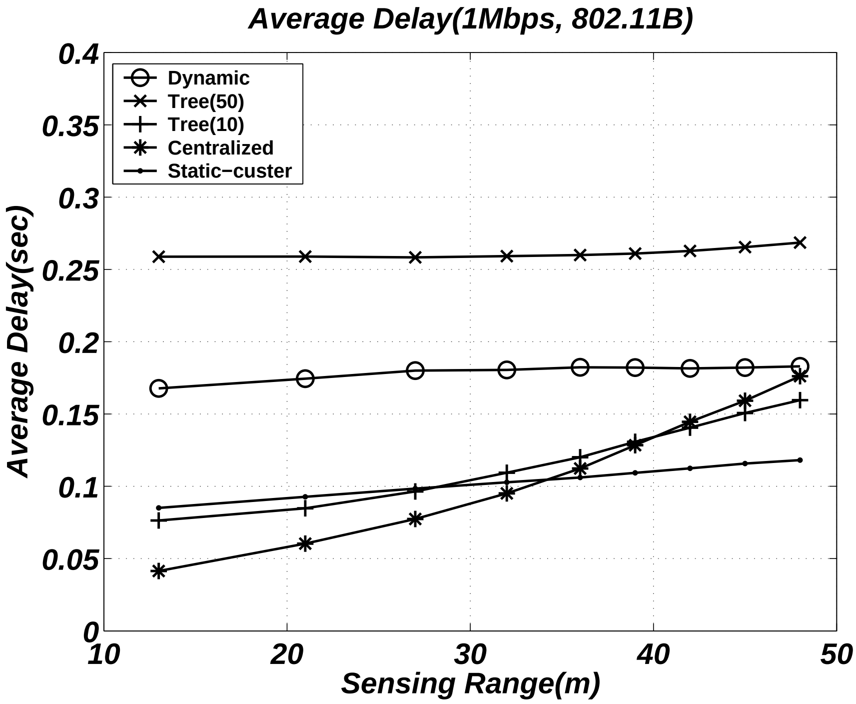


Figure 7.6: Average end-to-end delay (1 Mbps, IEEE 802.11)

Interesting results are that clustering gives less sensitive to sensing range. This is because the events are aggregated close to the event occurrence region, and only small number of packets are sent to the sink, reducing congestion and hence queueing delay. When the sensing range is greater than 35m, end-to-end delay of the static-cluster shows the best performance among various aggregation schemes.

7.7 Data packet delivery ratio

The data packet delivery ratio indicates how much target information is reliably delivered to the sink without packet loss. Figure 7.7 shows the data packet delivery ratio for the $1Mbps$ bandwidth experiment. The packet delivery ratio is highly dependant on the total number of data packet transmissions. As the sensing range increases, the number of packet transmissions also increase, causing high congestion and collision, consequently decreasing the data delivery ratio.

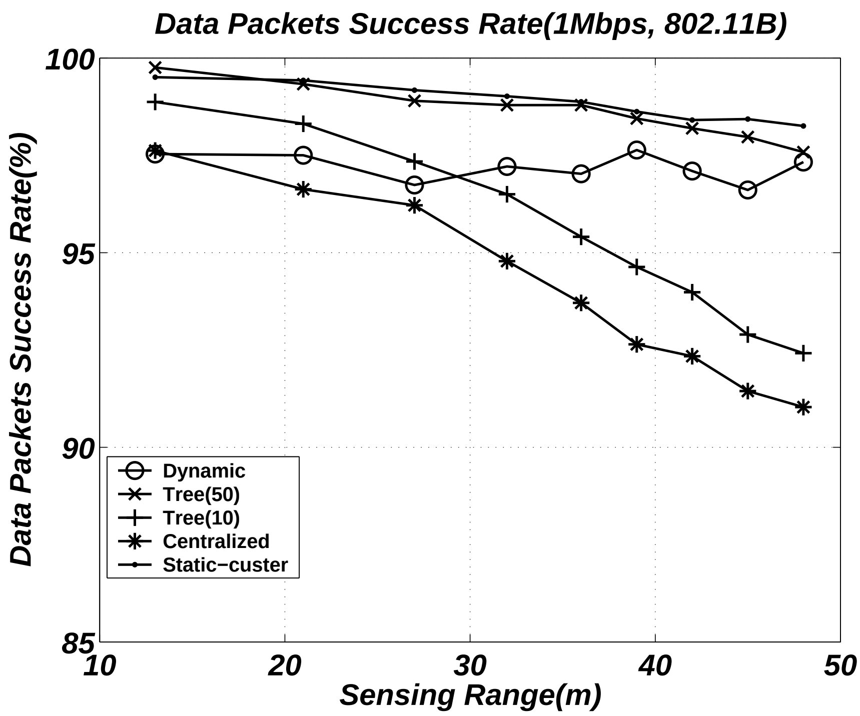


Figure 7.7: Data packet delivery ratio(1 Mbps, IEEE 802.11)

Intuitively, dynamic-cluster aggregation should give high data delivery ratio, since it has the least number of packet transmissions, as observed in Figure 7.1. However, surprisingly, its data delivery ratio turns out to be relatively less than that of static-cluster and tree-based aggregation schemes. This is because dynamic-cluster aggregation relies on broadcast extensively, for the initial sensing report message exchanges as well as the cluster head announcement messages. Since broadcast messages are not protected by RTS/CTS exchanges, this results in some packet loss due to hidden terminal collisions. On the other hand, tree-based and static-cluster aggregation schemes use unicast with RTS/CTS for their messages, hence they do not suffer as much from the hidden terminal problem. Table 7.2 shows the summary of simulation results when the sensing range is $48m$ and bandwidth is $1Mbps$.

Table 7.2: Summary of simulation result when the sensing range is $48m$ and bandwidth is 1 Mbps

Metrics	Centralized	Tree(10)	Tree(50)	Static	Dynamic
Packet Tx	22825	16664	12881	11024	9666
Aggregation Ratio	0.1965	0.269	0.3481	0.4067	0.4638
Energy(Joules)	7.9925	6.9043	6.1624	5.8541	4.9208
Energy-STD(Joules)	1.6953	1.1308	0.8694	0.6066	0.3466
Lifetime(sec)	49.27	59.074	66.186	73.903	92.413
Delay(sec)	0.1761	0.1596	0.2685	0.1181	0.183
Delivery Ratio(%)	91.035	92.42	97.585	98.254	97.33

7.8 Impact of the bandwidth

To investigate the impact of the bandwidth on the performance of data aggregation, we test under two data rates – $1Mbps$ and $250Kbps$. In this section, we examine the corresponding effect on metrics – data packet delivery ratio, end-to-end delay, and the energy consumption. Due to the reduction of bandwidth, overall performances are slightly degraded. Figures 7.8 and 7.9 show the comparison of packet delivery ratio and average end-to-end delay under two different data rates. The reduction of bandwidth increases congestion, queueing delay and packet loss, consequently about

$2\% \sim 5\%$ of packets are less delivered to the sink and about $25ms \sim 35ms$ of queuing delays are increased. Note that the packet delivery ratio and average end-to-end delay of cluster-based aggregation schemes are less sensitive to the bandwidth reduction than centralized and tree-based aggregation schemes. when the sensing range is greater than $42m$, end-to-end delay of dynamic-cluster aggregation is shorter than centralized and tree-based aggregation schemes. This is because the dynamic-cluster aggregation is less sensitive to queuing delay.

Figures 7.10 shows the comparison of energy consumptions under two different data rates. Due to the reduction of bandwidth, nodes need more transmission time to deliver a packet to the nodes also need more reception time to receive a packet from the nodes, hence increase of corresponding amount of energy consumptions are inevitable for each scheme. Also note that bandwidth reduction does not have a measurable effect on the energy consumption of dynamic-cluster aggregation. When the sensing range is $48m$, only 4% of additional energy is consumed in dynamic-cluster aggregation. But other aggregation schemes show 17% to 23% of additional energy consumption.

7.9 Impact of the MAC protocols

So far, we used 802.11 as the MAC protocol in our simulation. In this section we study how the performance of data aggregation schemes is affected by changing the MAC protocol from 802.11 to a more conventional sensor network MAC protocol –

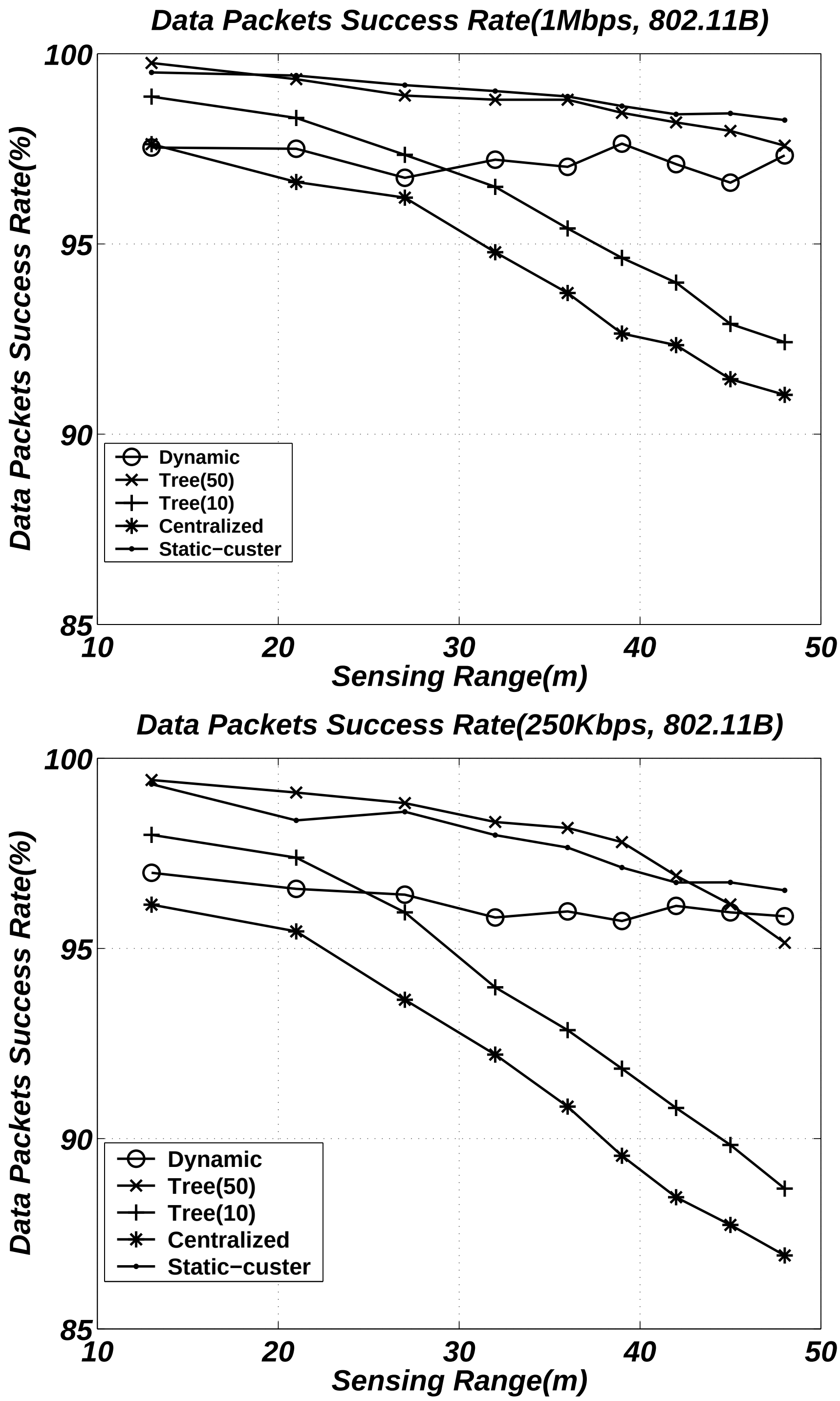


Figure 7.8: Comparison of Data Packet Delivery Ratio(1 Mbps vs. 250 Kbps)

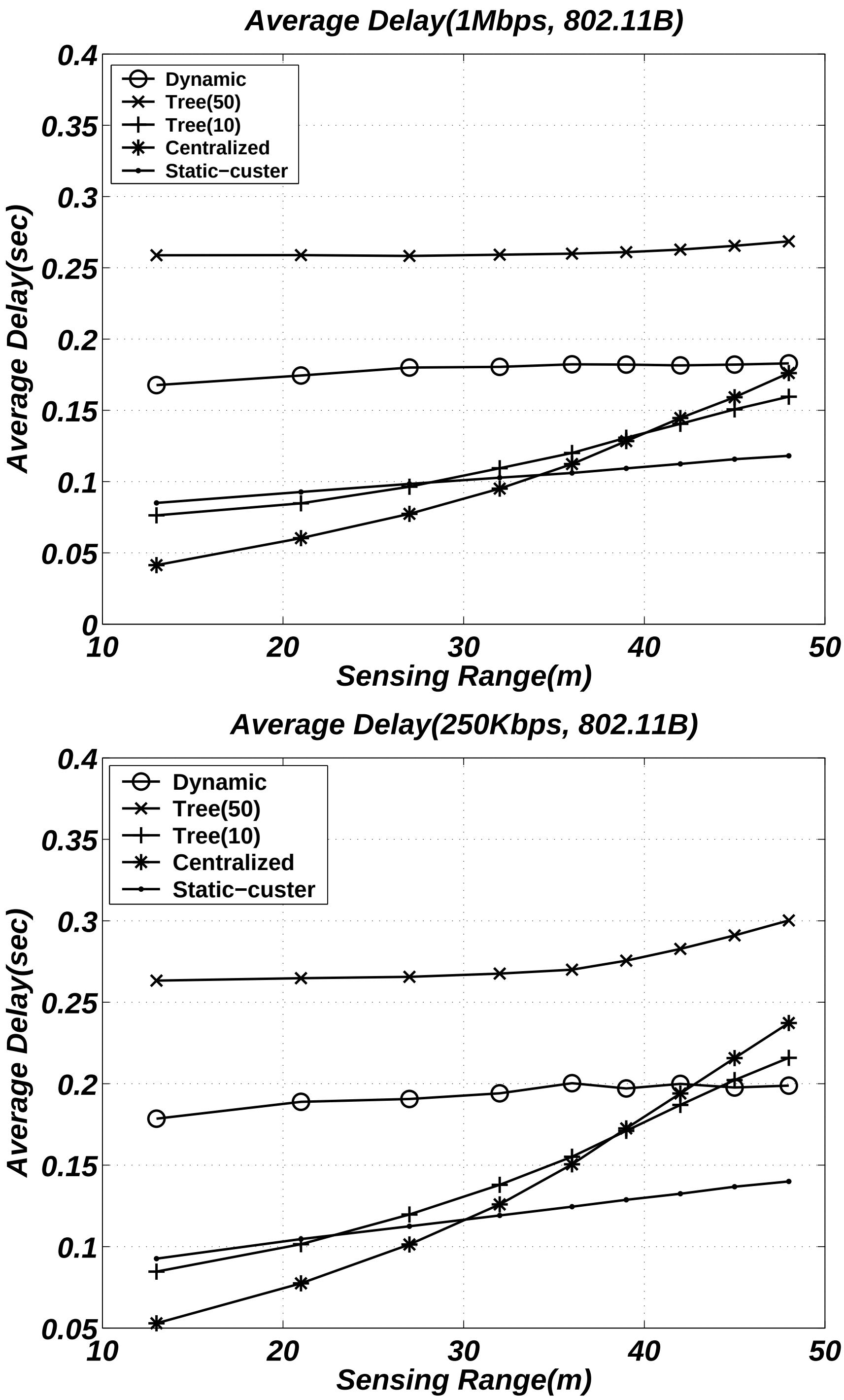


Figure 7.9: Comparison of Average end-to-end Delay(1 Mbps vs. 250 Kbps)

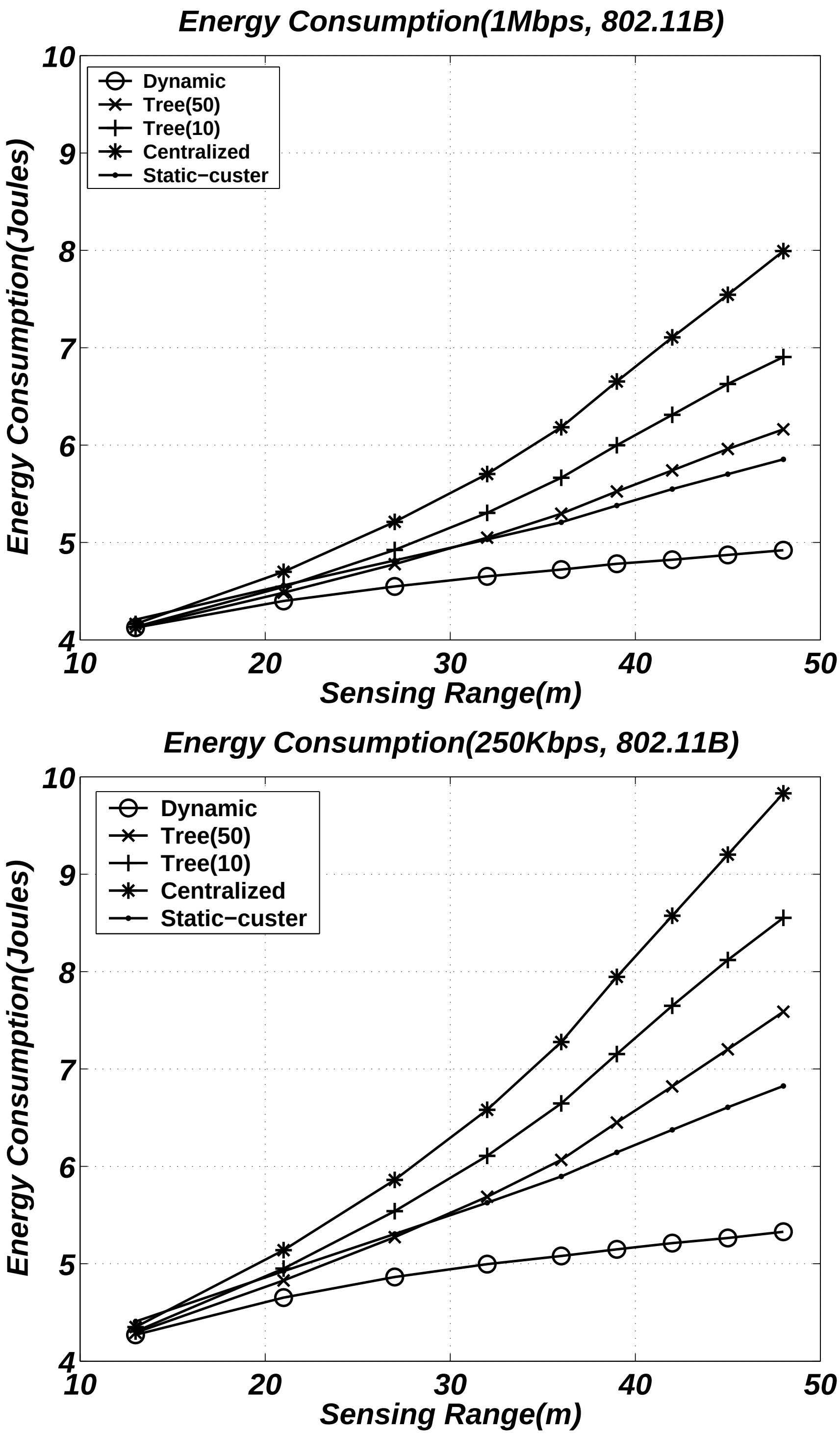


Figure 7.10: Comparison of Average Energy Consumption(1 Mbps vs. 250 Kbps)

B-MAC. B-MAC is the default MAC protocol for the TinyOS operating system for sensor networks. B-MAC is a CSMA-based MAC protocol, as is 802.11, but unlike 802.11, B-MAC does not perform binary exponential backoffs. Instead, when it finds the channel busy, it performs a back-off within a fixed window. For more details about B-MAC, we refer the readers to the original paper [20]. For our simulations, we do not enable link-layer ACK which is optional for B-MAC. When we generate 8 targets per second in the $500m$ x $500m$ area, the use of B-MAC leads to very low packet delivery ratios for all aggregation schemes. Hence we reduce the number of targets per second to 5 in this section.

Figure 7.11, 7.12 show the results of the number of packet transmissions and energy consumption under the two MAC protocols. 802.11 uses ACK, hence if a packet is not acknowledged, the 802.11 protocol tries several retransmissions before giving up and discarding the packet. In B-MAC, on the other hand, we disable ACK. As a result, B-MAC tries transmitting every packet just once. This is reflected in the lesser number of transmissions for B-MAC compared to 802.11. The lesser number of transmissions also translate to lesser energy consumption. However, as seen in Figure 7.13, this is at the cost of reduced data delivery ratio for B-MAC.

Due to the low packet delivery ratio under B-MAC, we could not fully investigate the impact of MAC protocols on the other metrics such as aggregation ratio, energy consumption, lifetime, etc. If we use optional link layer acknowledgement mechanism in B-MAC, and set the retransmission try number to large, ideally we could achieve

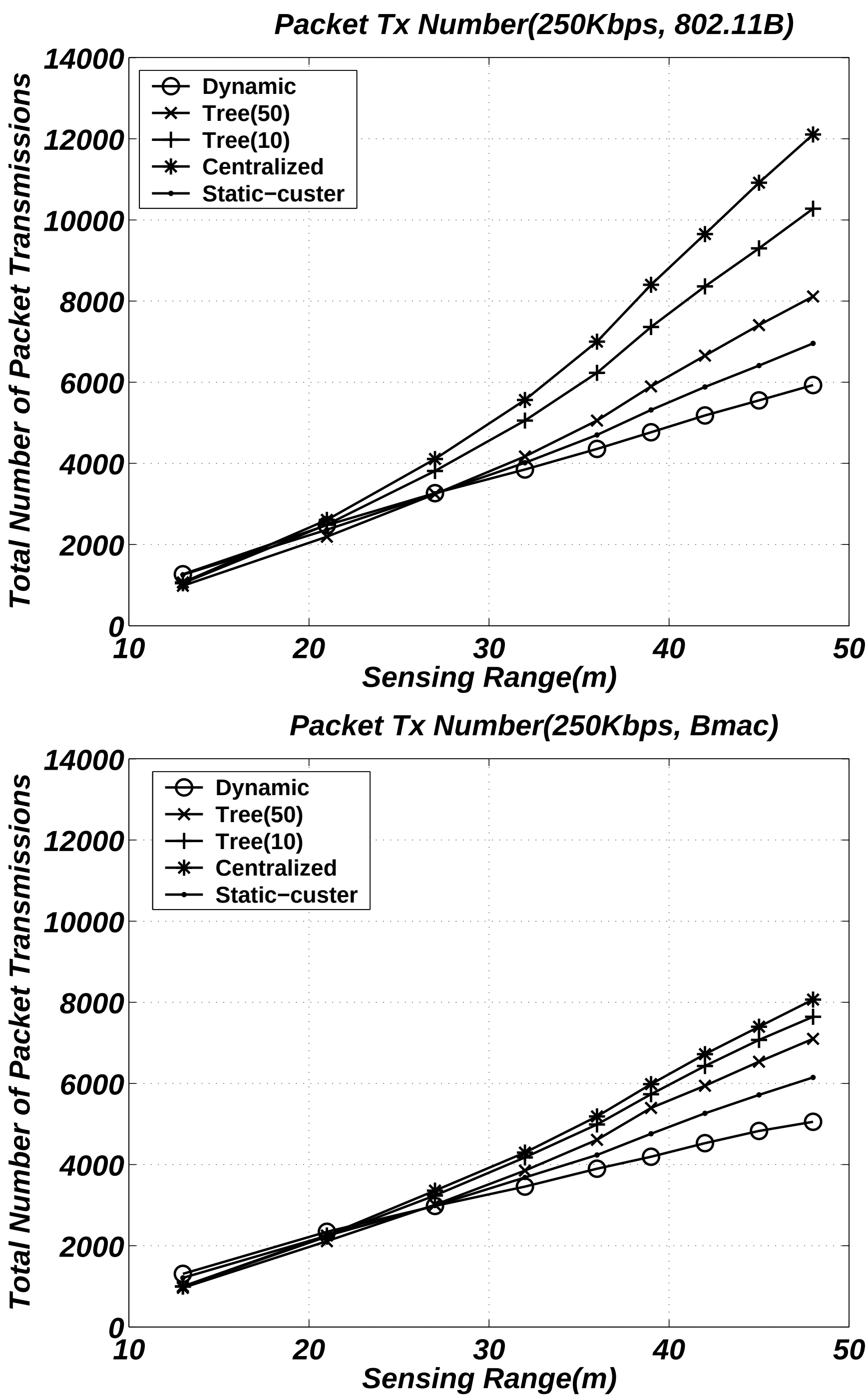


Figure 7.11: Comparison of Number of Packet Transmissions (802.11 vs. B-MAC)

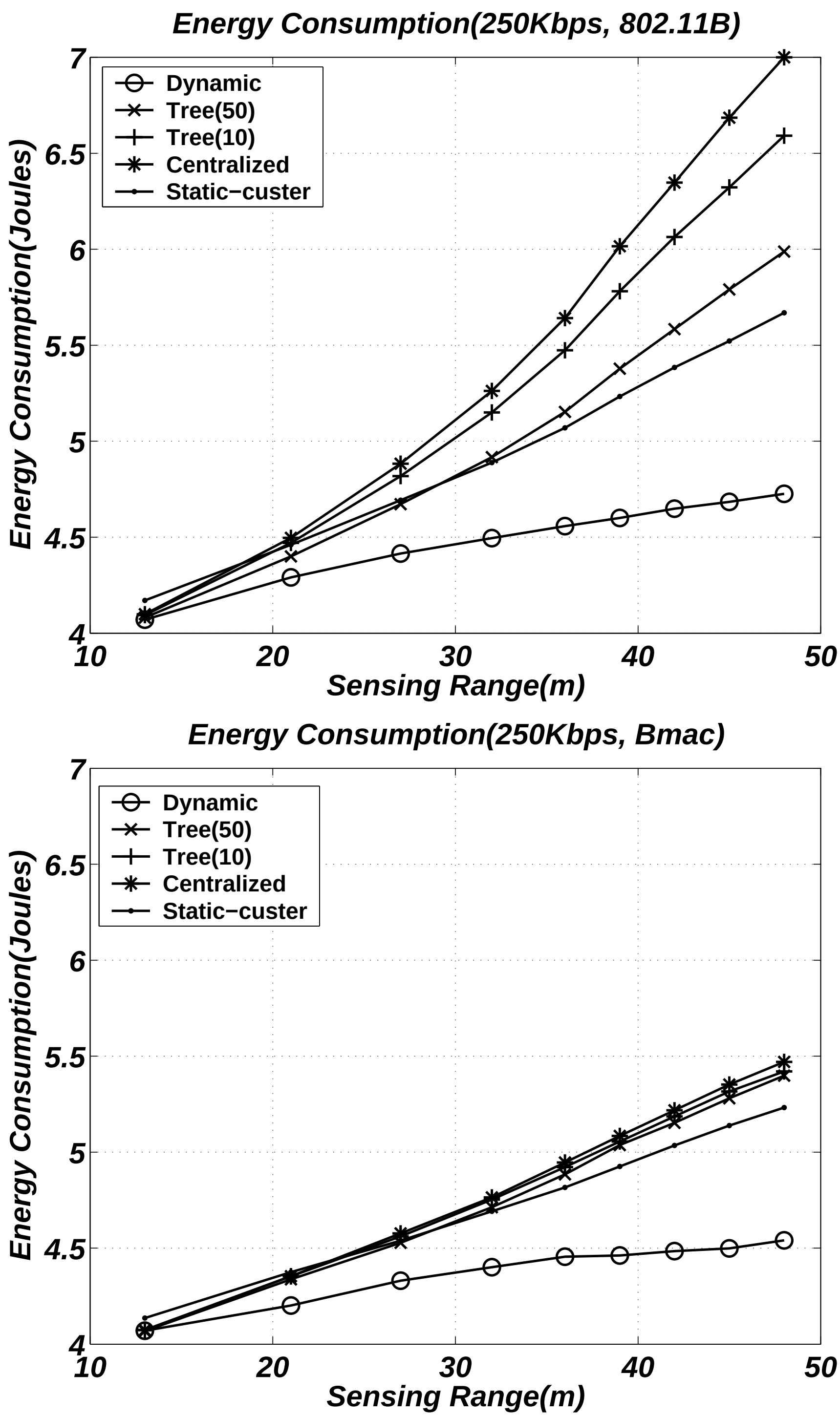


Figure 7.12: Comparison of Average Energy Consumption (802.11 vs. B-MAC)

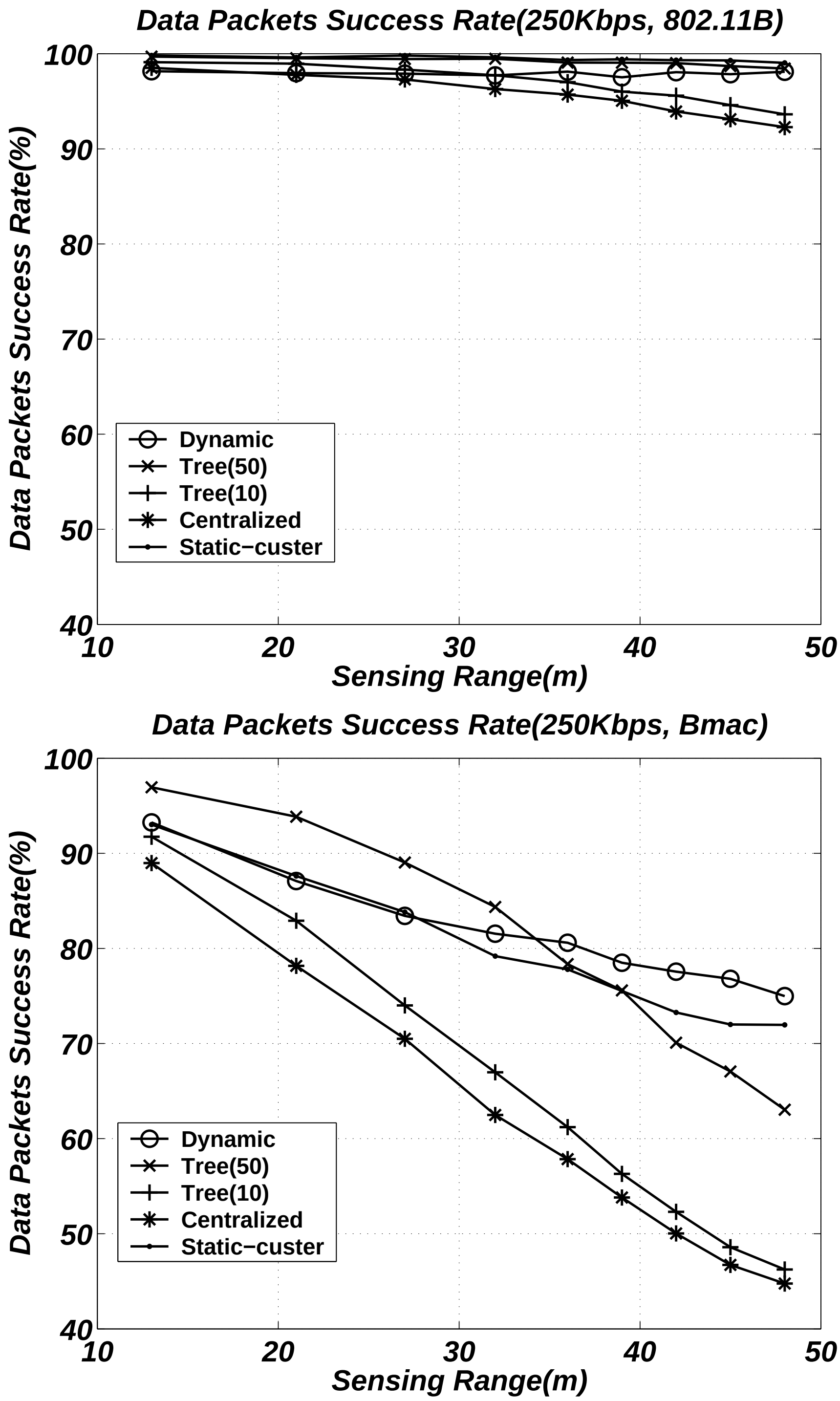


Figure 7.13: Comparison of Data Packet Delivery Ratio (802.11 vs. B-MAC)

100% packet delivery ratio. In order to predict the total number of packet transmissions of each scheme when it achieves 100% packet delivery ratio, we divide the total number of packet transmissions to data packet delivery ratio. Figure 7.14 shows the predicted total number of packet transmissions when 100% of generated packets are successfully delivered to the sink under B-MAC. From the Figure, we can predict 33% to 50% more number of packet transmissions than 802.11 because of the packet retransmissions which are caused by high packet loss. In the next section, we will further investigate the effect of link layer acknowledgement of B-MAC in real sensor network environment.

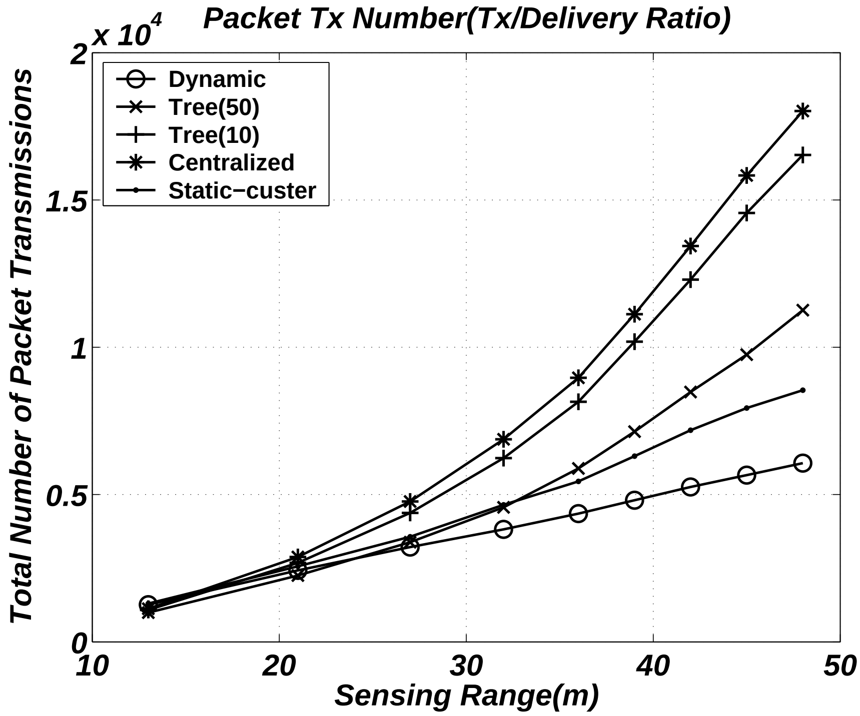


Figure 7.14: The predicted total number of packet transmissions when 100 % of generated packets are delivered to the sink under B-MAC

Chapter 8

Experiment on MICA2 testbed

To evaluate the performance of data aggregation schemes on real sensor network environment, we perform experiments on a 31 MICA2 testbed. Each node is placed in faculty and student offices and labs of Monteith Research Center in Centennial Campus. Figure 8.1 shows the locations of 31 nodes which are placed in a $25m \times 18m$ area.

8.1 Implementation Details

We are interested in the performance comparison of data aggregation schemes in a target detection application. In our testbed, nodes do not have a sensor board, hence we emulate the detection of targets as follows. We define a set of xy axes, with node 1 chosen to be the origin. Every node is then assigned an x and y co-ordinate based on its location on the testbed. We then generate 1 target every second at a

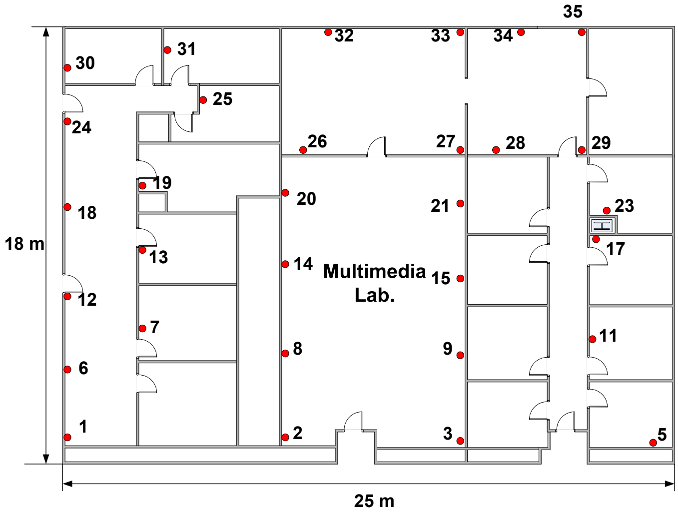


Figure 8.1: 31 MICA2 test bed in Monteith Research Center

new random location using a pseudo-random number generator on a PC, and store the time and the corresponding target location in a file. This file is fed to each node on node initialization. In the beginning of each second, each node reads the target location from this file, if the Euclidian distance between the target location and its own x,y co-ordinates is less than its assigned sensing range, the node considers the target to be detected. The sensing range of nodes is varied in the experiments from 1m to 6m.

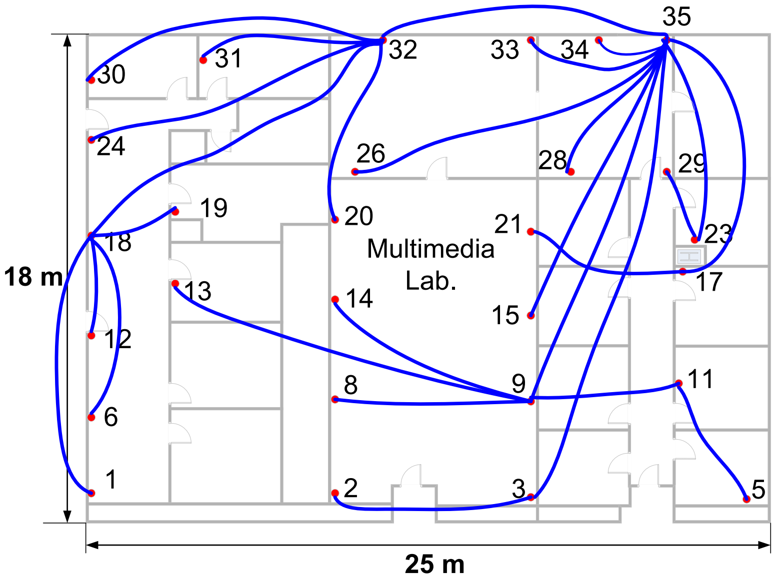
To see the effect of data aggregation, we require a dense network with routing paths comprising of several hops to the sink node. Unfortunately, due to the physical limitations of our testbed, it is not possible to create this kind of network. As an example, we show the snapshot of routing paths generated on our testbed when all nodes use a transmit power of $1dBm$ and node 35 in the top right corner is selected as the sink node. In spite of the low transmit power, the average hop count of nodes to the sink turns out to be 1.8 with a diameter of 3 hops.

Hence, in order to create a better multi hop sensor network, we apply two topology control techniques in our test bed: grid topology and random topology control. In grid topology control, a node can only communicate with its immediate horizontal and vertical neighbors. A node filters out packets from non neighbor nodes. In the random topology control, a node randomly select d nodes as neighbors and packets coming from non neighbor nodes are filtering out.

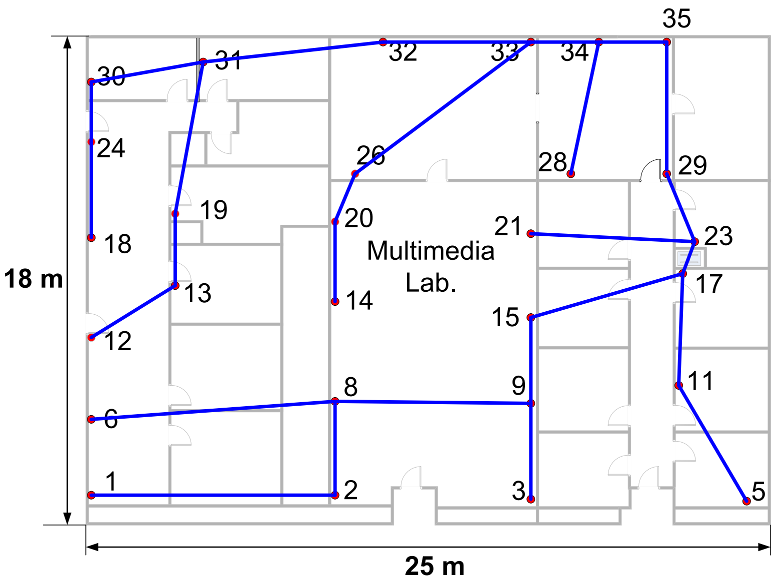
Nodes find their routing paths on top of those topology controlled sensor network. To provide better link connectivity between nodes, we set the radio power to the maximum, or $10dBm$. Figure 8.2(b) shows a snapshot of the routing paths on top of grid topology controlled sensor network with node 35(right-top of the test bed) selected as the sink node. By applying a topology control we achieve up to 8 hop counts from sources to the sink and the average hop counts and median hop counts of all nodes are 4.3 and 5, respectively.

8.2 Test results without link layer acknowledgement

Figure 8.3 shows the test results on MICA2 testbed without link layer acknowledgement mechanism. The trend seen here is quite similar to that seen in the simulation results in section 7.9. Due to the lack of reliable packet transmission mechanism



(a) Routing paths created by Mint with radio power control(1dBm)



(b) Routing paths created by Mint with grid topology control

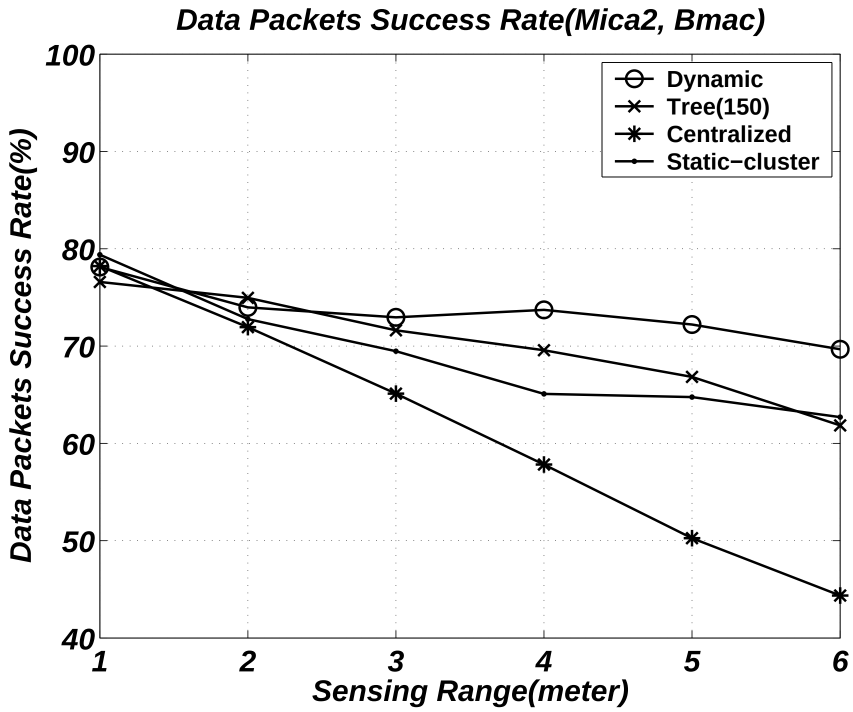
Figure 8.2: Routing paths created with power control and grid topology control

of B-MAC and high link loss rate in a real testbed, a large amount of generated packets could not be delivered to the sink. As shown in Figure 8.3(a), only 44.5 % to 71.2 % of the packets are successfully delivered to the sink. As mentioned in the simulation section 7.9, due to the low packet delivery ratio of each aggregation schemes, we could not compare the performance of other metrics using this results. In order to increase the packet delivery ratio, we enable the link layer acknowledgement mechanism in the following section.

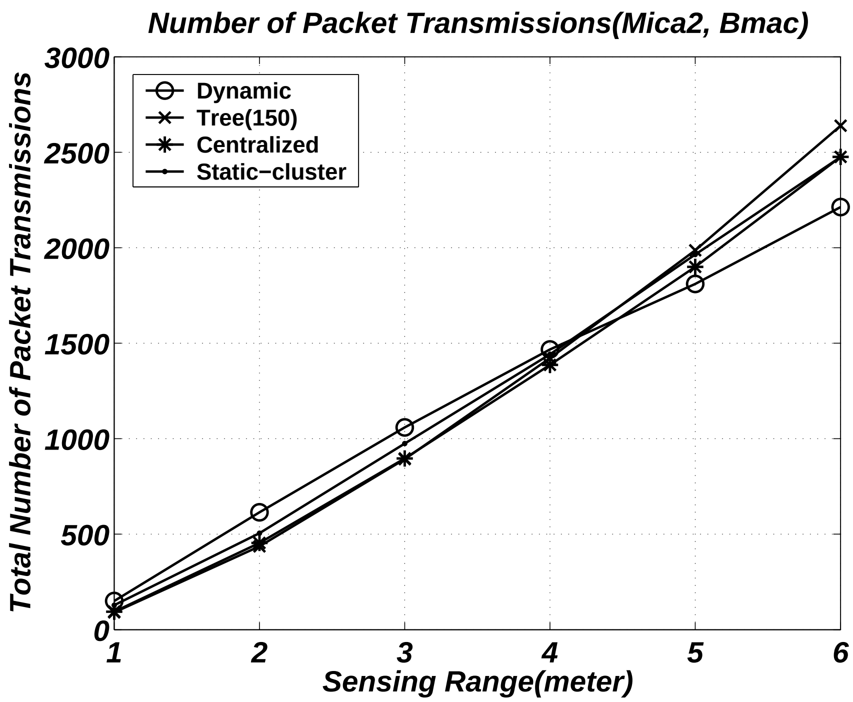
8.3 Test results with link layer acknowledgement

In this section, we show the test results on MICA2 testbed when we enable the link layer acknowledgement mechanism in B-MAC. We set the maximum number of retransmissions to 5. In this test, we define the lifetime of the network to be the time duration from the beginning of the simulation to the time until at least one of nodes in the network consumes 1.5 joules. Based on empirical result, we set the aggregation delay to 150ms for tree-based aggregation, set the cluster head waiting time to 150ms for static-cluster aggregation, set the cluster head election processing delay to 400ms for dynamic-cluster aggregation.

As a result of the link layer acknowledgement, as shown in Figure 8.4, overall packet delivery ratios are considerably improved compared to without link layer acknowledgement. Regardless of the sensing range, the dynamic-cluster aggregation guarantees 99% of packet delivery ratio. On the other hand, other schemes show



(a) Average Packet Delivery Ratio



(b) Number of packet transmissions

Figure 8.3: Results on MICA2 test bed without enabling the link layer acknowledgement

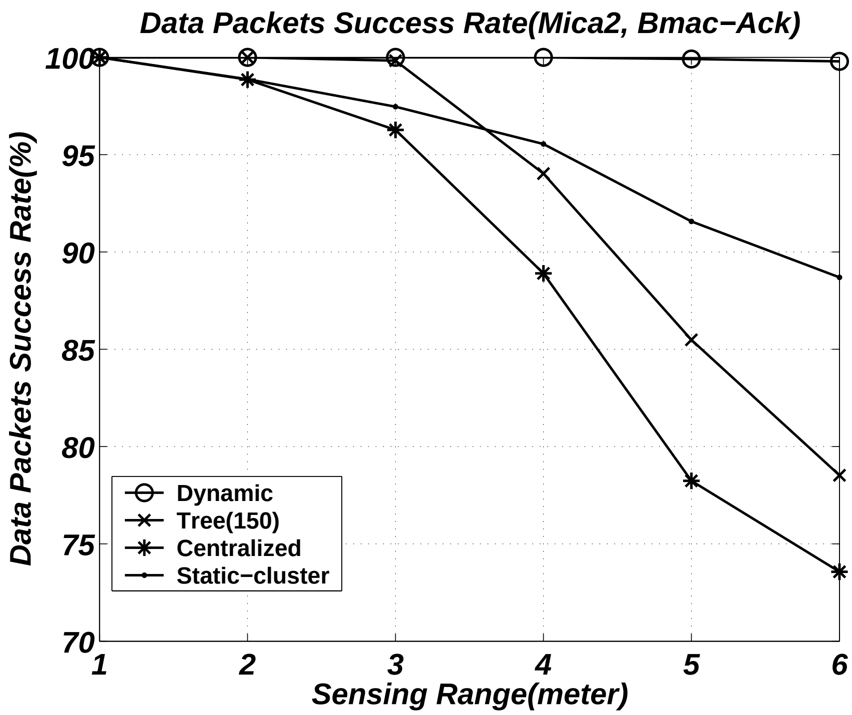


Figure 8.4: Average Packet Delivery Ratio

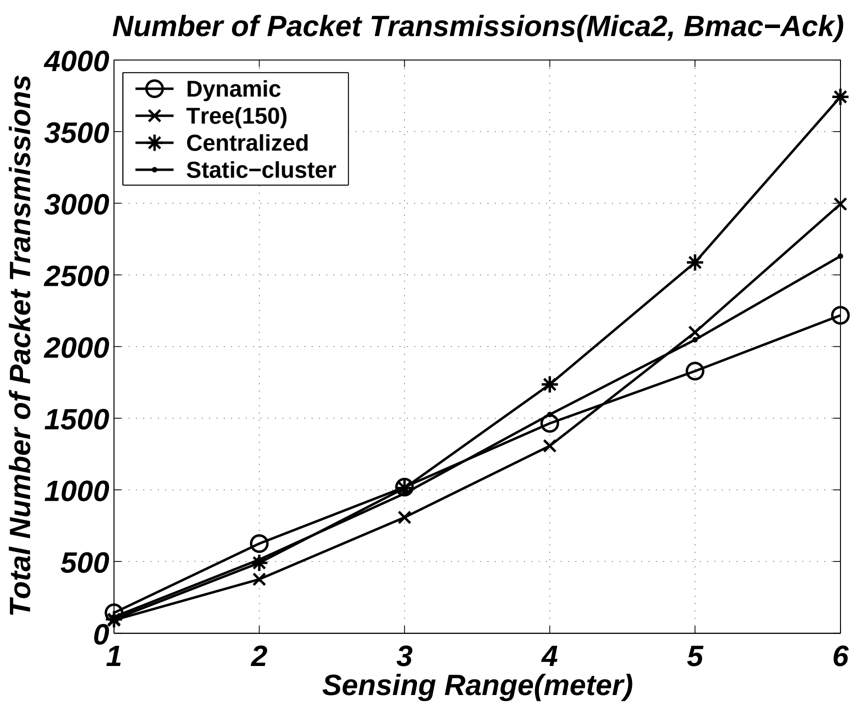


Figure 8.5: Number of packet transmissions

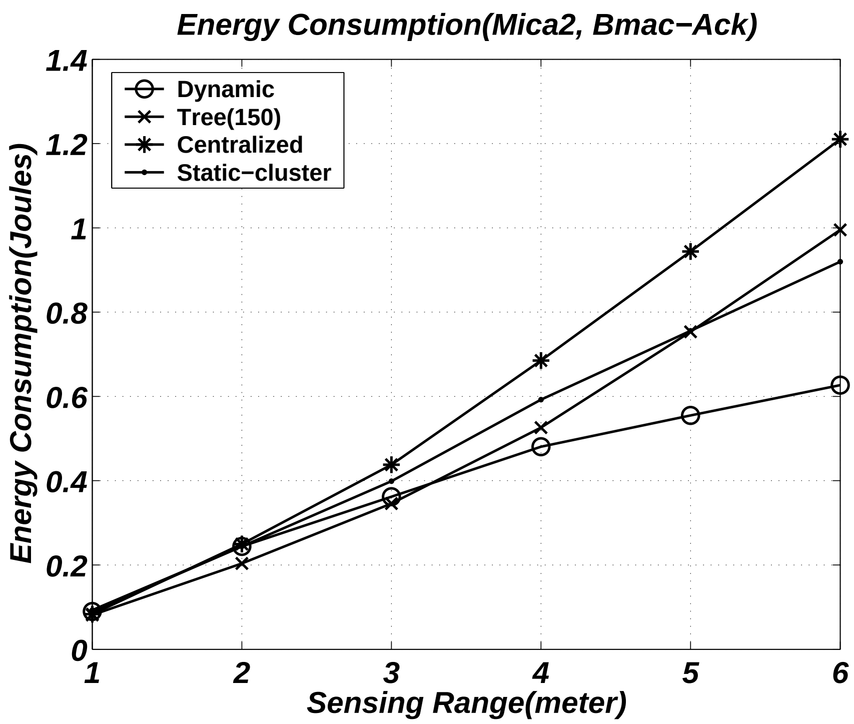


Figure 8.6: Average Energy consumptions

73% to 88% packet delivery ratios, also their delivery ratios are highly affected by the sensing range. Note that centralized, tree-based and static-cluster aggregation schemes could not deliver all generated packets to the sink. This affects the number of packet transmissions, energy consumption and network lifetime as seen in Figure 8.5, 8.6, 8.7, respectively. In spite of the different packet delivery ratio of each scheme, the trend of experiment results is quite similar to the results of analytical and simulation study. When the sensing range is short, centralized and tree-based aggregation show better performance than cluster-based aggregation schemes, but as the sensing range increases, static-cluster and dynamic-cluster aggregation schemes start performing better than centralized and tree-based aggregation schemes. Figure

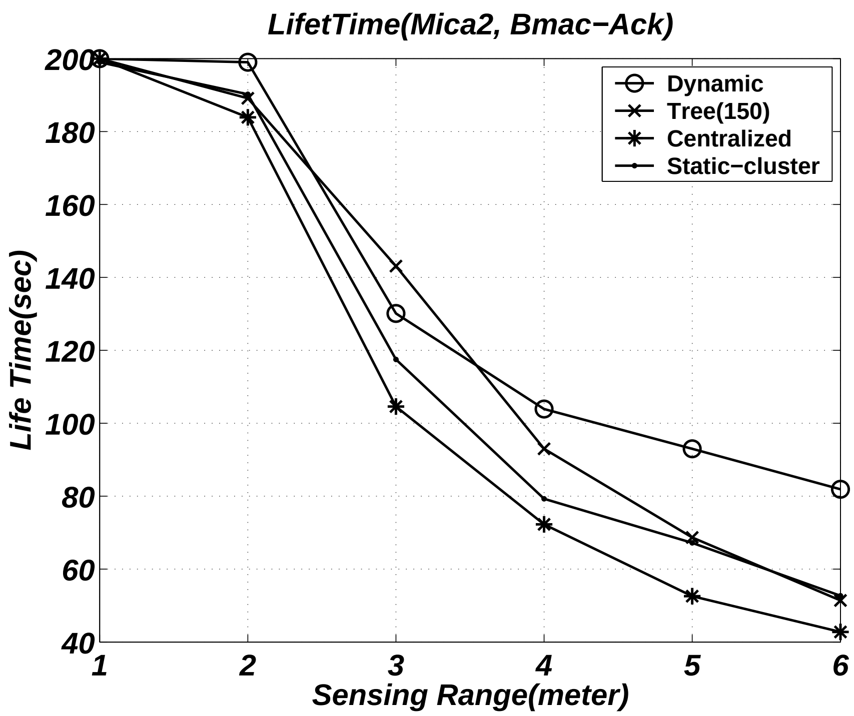


Figure 8.7: Network Lifetime

8.8 shows the aggregation ratio of each scheme.

As mentioned before, the end-to-end delay is affected by four factors: (1) queueing delay at forwarding nodes, (2) aggregation delay at forwarding nodes in tree-based aggregation, (3) cluster head aggregation delay in static-cluster aggregation, (4) dynamic cluster head election processing time in dynamic-cluster aggregation. The effect of these factors is shown in Figure 8.9. Note that centralized, tree-based, and static-cluster aggregation schemes exhibit sensitivity to the sensing range, which indicates that queueing delay comprises a significant component of the total end-to-end delay. However dynamic-cluster aggregation exhibits no sensitivity to sensing range. This is because the number of packet transmissions are significantly reduced by the

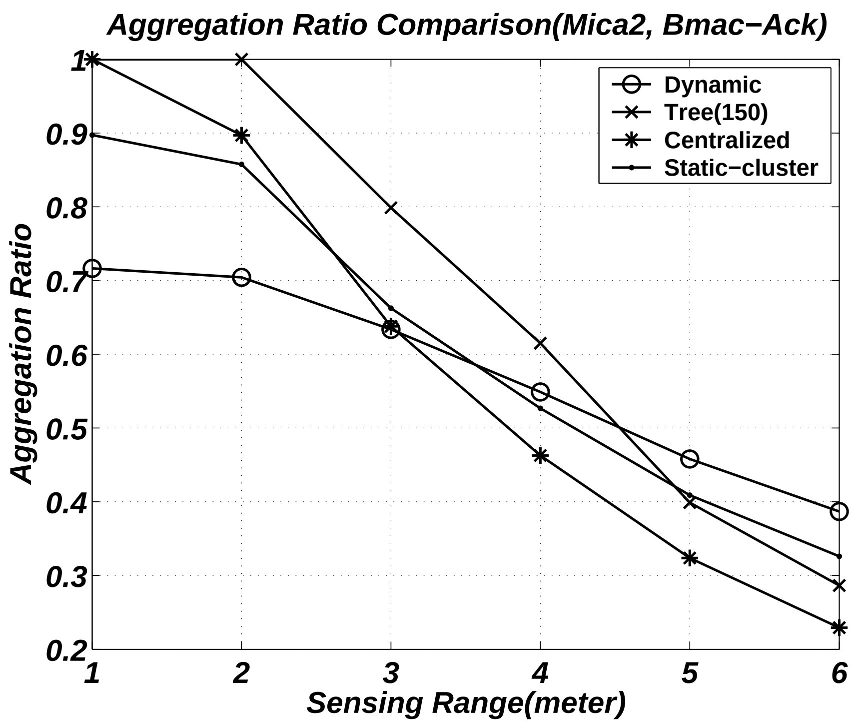


Figure 8.8: Aggregation Ratio

use dynamic clustering, reducing queueing delay.

Dynamic-cluster aggregation shows good performance in the real sensor network environment. However, there is one caveat. Our dynamic-cluster aggregation relies on broadcast extensively, for the cluster head election process. Since broadcast message cannot take advantage of link layer acknowledgements, under lossy environments, this leads to the election of multiple cluster heads. Hence some amount of redundant packet transmissions is inevitable in a real sensor network environment. Figure 8.10 shows the total number of elected cluster heads and the number of events that occurred throughout the simulation time. When the sensing range is $6m$, 234 nodes are elected as cluster heads while 182 events are generated during the simulation time.

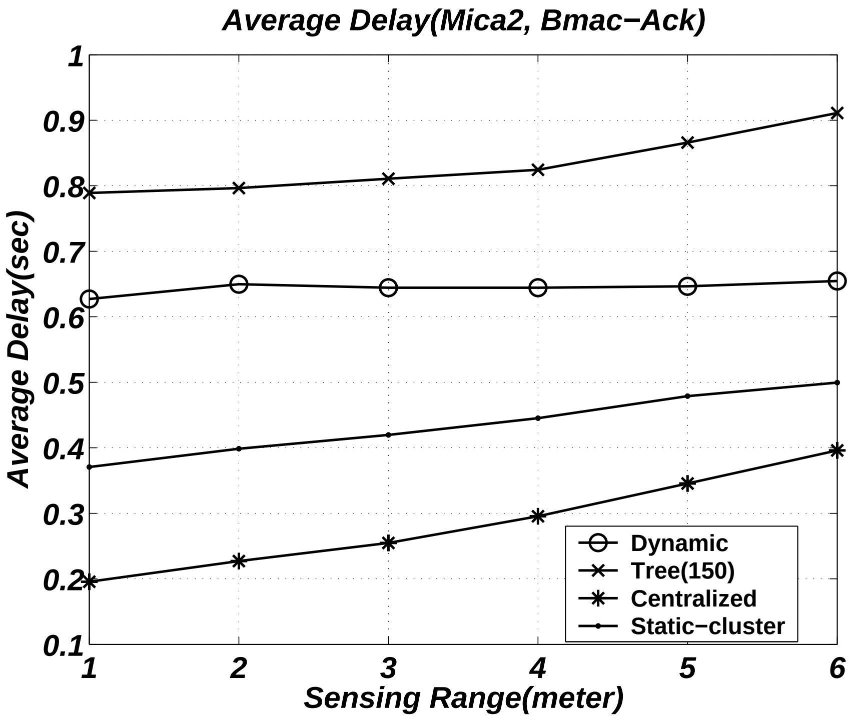


Figure 8.9: Average end-to-end Delay

About 52 nodes are additionally elected as cluster heads.

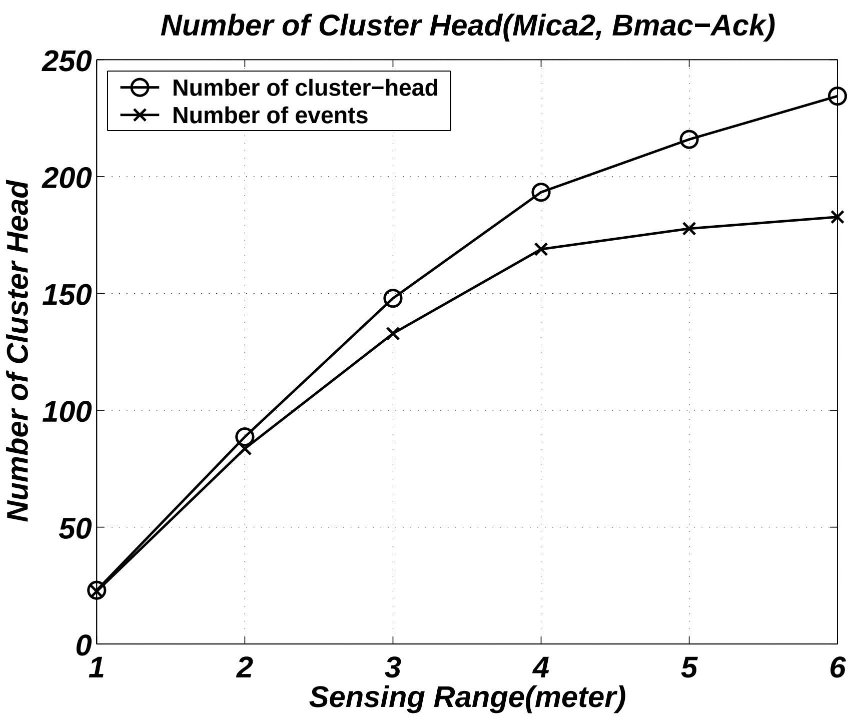


Figure 8.10: Multiple cluster heads election problems of dynamic-cluster aggregation in a real sensor network environment

Chapter 9

Conclusion

In this paper, we compare centralized, tree-based, static-cluster, and dynamic-cluster aggregation schemes with comprehensive theoretical analysis, simulation and real experiments. We attempt to give clear scenarios where a particular scheme may be more beneficial compared to others. Along the way, we also propose two cluster-based aggregation algorithms, which are simple enough to be implemented on resource-constrained sensor networks. The results show that the performance of data aggregation is highly dependant on setup overhead, node density, sensing range, and the distance from sources to the sink, and we report the following findings:

- In centralized aggregation, all packet transmissions reach the sink without any aggregation along the way. Therefore a large number of packet transmissions in the sensor network result in high congestion, packet loss and energy depletions and long delay. However when the sensing range is short and/or the sources are close to the sink, centralized aggregation shows the better performance than others since other schemes can not take advantage of data aggregation under this condition.

- In tree-based aggregation, aggregation is performed at intermediate nodes and only aggregated packets are sent along routing paths. Tree-based aggregation relies on established routing paths. Therefore, it does not need to pay additional costs for creating a aggregation backbone. However, note that routing paths may not merge at any intermediate aggregation point, or may merge several hops away from the region where the event occurs. Because of these factors, tree-based aggregation shows limited aggregation performance compare to other cluster-based aggregation schemes. In this scheme, longer aggregation delay at the forwarding nodes shows better aggregation performance while paying an increasing of end-to-end delay.

- In static-cluster aggregation, aggregation is performed on a cluster backbone which is created during the initial set-up time. There is significant overhead involved in creating this backbone and the routes computed over the backbone may be sub-optimal in-terms of path length or throughput. Because of these factors, this scheme shows poor performance when the sensing range is short and/or the sources are close to the sink. However, as the sensing range and distance from the sources to the sink increases, this scheme shows better aggregation performance than tree-based aggregation by taking advantage of data aggregation at the cluster heads.

- Dynamic-cluster aggregation forces aggregation to be performed in an event sensing area. Hence when the sensing range and distance from sources to the sink are long, this scheme shows the best aggregation performance over other schemes. However when the sensing range is short and/or the sources are close to the sink, this scheme shows the worst performance due to the high costs for dynamic cluster head election process. Dynamic-cluster aggregation shows some performance degradation under high loss link environment; a case of multiple cluster heads election has broken out in an event area, this result in redundant packet transmissions in the sensor network.

Bibliography

[1] Mica2 mote data sheet.

[2] The network simulator: ns-2.

[3] Kokichi Sugihara Sung Nok Chiu Atsuyuki Okabe, Barry Boots. Spatial tessellations: Concepts and applications of voronoi diagrams, 2nd edition. Wiley Series in Probability and Statistics, 2000.

[4] Stefano Basagni. Distributed clustering for ad hoc networks. In *ISPAN '99: Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN '99)*, page 310, Washington, DC, USA, 1999. IEEE Computer Society.

[5] WeiPeng Chen, Jennifer C. Hou, and Lui Sha. Dynamic clustering for acoustic target tracking in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 3:258 – 271, July-Aug 2004.

[6] I. Downard. Simulating sensor networks in ns-2. *NRL/FR/5522–04-10073, Naval Research Laboratory, Washington, D.C.*, 2004.

[7] Saurabh Ganeriwal, Ram Kumar, and Mani B. Srivastava. Timing-sync protocol for sensor networks. In *Proc. 1st Intl. Conf. on Embedded Networked Sensor Systems (SenSys)*, pages 138–149, Los Angeles, CA, November 2003. ACM.

[8] S. Ghiasi, A. Srivastava, X. J. Yang, and M. Sarrafzadeh. Optimal energy aware clustering in sensor networks. *Special Issue: Special section on sensor network technology and sensor data managment*, 2:258–269, July 2004.

[9] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An application-specific protocol architectures for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4), October 2002.

[10] J. Hellerstein, W. Hong, S. Madden, and K. Stanek. Beyond average: Towards sophisticated sensing with queries. *In Workshop on Information Processing In Sensor Networks (IPSN)*, 2003.

[11] H. J. Hilhorst. The perimeter of large planar voronoi cells: a double-stranded. random walk. *Journal of Statistical Mechanics: Theory and Experiment*, Feb. 2005.

[12] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann. Impact of net-

work density on data aggregation in wireless sensor networks. In *Technical Report 01-750, University of Southern California*, November 2001.

[13] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*, 11:2–16, February 2003.

[14] Chalermek Intanagonwiwat, Deborah Estrin, Ramesh Govindan, and John Heidemann. Impact of network density on data aggregation in wireless sensor networks. In *ICDCS '02: Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02)*, page 457, 2002.

[15] J. F. C. Kingman. Poisson processes. In *volume 3 of Oxford Studies in Probability.* The Clarendon Press Oxford University Press, New York, 1993. Oxford Science Publications., 1976.

[16] P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K. Pradhan. A cluster-based approach for routing in dynamic networks. *ACM SIGCOMM Computer Communication Review*, 27:49 – 65, April 1997.

[17] Bhaskar Krishnamachari, Deborah Estrin, and Stephen B. Wicker. The impact of data aggregation in wireless sensor networks. In *ICDCSW '02: Proceedings of the 22nd International Conference on Distributed Computing Systems*, pages 575–578, 2002.

[18] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. Tinyos: An operating system for wireless sensor networks, 2004.

[19] LAN MAN Standards Committee of the IEEE Computer Society. Wireless lan medium access control (mac) and physical layer (phy) specification. New York, NY, USA. IEEE.

[20] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, New York, NY, USA, 2004. ACM Press.

[21] Theodore S. Rappaport. Wireless communications: Principles and practice, 2nd edition. Prentice Hall PTR, 2002.

[22] A. Sharaf, Jonathan Beaver, Alexandros Labrinidis, and K. Chrysanthis. Balancing energy efficiency and quality of aggregate data in sensor networks. *The VLDB Journal*, 13(4):384–403, 2004.

[23] S. Soro and W. Heinzelman. Prolonging the lifetime of wireless sensor networks via unequal clustering. In *Proceedings of the 5th International Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (IEEE WMAN '05)*, April 2005.

[24] Alec Woo, Terence Tong, and David Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 14–27, New York, NY, USA, 2003. ACM Press.

[25] Ossama Younis and Sonia Fahmy. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 3:366 – 379, Oct.-Dec 2004.

[26] Yang Yu, Bhaskar Krishnamachari, and V.K. Prasanna. Issues in designing middleware for wireless sensor networks. *IEEE Network*, 18:15 – 21, Jan/Feb 2004.

[27] Wensheng Zhang and Guohong Cao. Dctc: dynamic convoy tree-based collaboration for target tracking in sensor networks. *IEEE Transactions on Wireless Communications*, 3, Sept. 2004.

[28] Feng Zhao, Jaewon Shin, and J. Reich. Information-driven dynamic sensor collaboration. *IEEE Signal Processing Magazine*, 19:61 – 72, March 2002.