Chapter 1

SENSOR MANAGEMENT

Mark Perillo
University of Rochester
Rochester, NY 14627
perillo@ece.rochester.edu

Wendi Heinzelman
University of Rochester
Rochester, NY 14627
wheinzel@ece.rochester.edu

Abstract Sensors are deployed in a sensor network for the purpose of providing data about environmental phenomena to the sink node(s). As not all sensors may be able to transmit their data directly to the sink(s), sensors must also route other sensors' data. Therefore, sensors must assume roles of both data provider and router. However, there are often many more sensors in the network than are needed at a given time to accomplish these tasks. Sensor management is needed to assign roles to each sensor, so that nodes that are not needed at a given time can enter a sleep state to save energy until they are needed, thereby extending network lifetime. The area of sensor management, as defined here, includes topology control—choosing which nodes should be routers to ensure a connected network—and sensing mode selection—choosing which nodes should sense data to meet application requirements (application QoS). It is possible to maximize network lifetime by finding optimal schedules of when each node should perform each function. However, these optimizations are computationally intensive, require global knowledge, and are not robust to changes in network topology. Therefore, there is a need for distributed, robust, and computationally efficient sensor management protocols that extend network lifetime while ensuring that application goals are met. In this chapter, protocols for both topology control and sensing mode selection are described and a qualitative comparison of these protocols is given.

Keywords: Sensor management, topology control, sensing mode selection, application QoS, network lifetime

1. Introduction

Recent advances in micro-fabrication technology have spurred a great deal of interest in the use of large-scale wireless sensor networks. The sizes of these networks are expected to grow to large numbers of nodes (thousands) in the next several years as the cost of manufacturing the sensors continues to drop significantly. The goal of these large-scale sensor networks is to gather enough data to monitor the environment with an acceptable fidelity, or application quality of service (QoS). At the same time, these networks are expected to last for a long time (months or even years) without recharging the small batteries providing energy to the individual sensors. Sensor management, including topology control and sensing mode selection, is essential to ensure that application QoS is met while extending network lifetime.

A sensor network is essentially a distributed network of data sources that provide information about environmental phenomena to an end user or multiple end users. Typically, data from the individual sensors are routed via the other sensors to certain sink points in the network (base stations), through which the user accesses the data. Two of the essential services provided by each sensor are sensing the environment and routing other sensors' data. As sensor nodes are provided solely to support the sensor network application, each node should only be used for sensing or routing if this is the best role for that sensor to play to support the end goal.

Quite often, there are so many nodes deployed that not all of the nodes need to provide or route data. Sensor management protocols determine which sensors are needed to provide data (sensing mode selection) and which are needed to ensure a connected topology so data can reach the sink points (topology control). The goal of most sensor management protocols, both for topology control and for sensing mode selection, is to ensure energy efficiency to maximize network lifetime. If sensors are not needed at a given time to provide data or route other sensors' data, they can save energy by shutting down (i.e., going into "sleep" mode) or halting traffic generation until they are needed at a later time. Both topology control and sensing mode selection play critical roles in ensuring that necessary sensor data reach the sink(s) while removing any unnecessary redundancy in the network, resulting in network-wide energy efficiency and long network lifetime.

Topology Control. Topology control is used when sensors are deployed with density high enough that not all sensors are needed to route data to the sink(s). The goal of a topology control protocol is to ensure that enough nodes are activated to provide a connected network so all sensors that have data to send can get their data to the base station while turning off any unnecessary sensors to save energy. Oftentimes, topology control protocols aim to rotate active sensor nodes so that the energy drain of performing routing functions is distributed evenly among all the nodes in the network. In addition to achieving energy efficiency, topology control protocols should be fault tolerant so that the loss of one or a small number of sensors does not disconnect the network.

Sensing Mode Selection. The need for sensing mode selection arises when sensors are deployed with density high enough that activating every sensor in the network provides little more quality of service (QoS) to the sensor network application than what could have been provided with many fewer sensors (i.e., the marginal quality provided by many of the sensors is minimal). In fact, activating all of the sensors can be detrimental to the overall task of the sensor network if there is so much traffic on the network that congestion is noticeable [28]. In this case, network throughput can degrade significantly and important data may be dropped as packet queues at the sensor nodes overflow. Among the data packets that do reach the data sink(s), high packet delays may be introduced, rendering the data useless. The goal of sensing mode selection is to have only certain sensors gather data so that there is no unnecessary redundancy and the network can operate at a point where the cumulative sensor data quality is sufficient to meet the application's goals, network congestion is minimal, and energy-efficiency is achieved.

Optimizing Sensor Management. If an application is able to perform at an acceptable level using data from a number of different sensor sets, it is possible to schedule the sets so as to maximize the sum of the time that all sensor sets are used. The lifetime of an individual sensor is influenced by the amount of data that it routes as well as the amount that it generates. Thus, network lifetime can be greatly extended when sensor mode selection and routing are solved jointly.

This problem of when to schedule sensors to sense data and when to schedule them to route data can be solved optimally so that network lifetime is maximized for a particular application QoS [1, 22]. The constraints of the optimization problem include the sensor battery levels, which dictate the total amount of time any node can route other nodes' data and the total amount of time any node can be an active sensor, the fact that all data sent by a scheduled sensor must be forwarded through other sensors to reach the sink (conservation of data flow), and the minimum acceptable level of quality of service. The objective of the problem is to maximize the total time the network is operational, which is the sum of the times that each sensor set is operational, given these constraints. This problem can be formalized as a generalized maximum flow graph problem and solved via a linear program [22].

In some sensor network applications, data can be aggregated to reduce the total amount of data sent to the base station. Similar network flow optimization problems can be designed to choose not only sensing and routing roles but also data aggregator roles for each sensor such that application QoS is met while lifetime is maximized [1, 15]. The Maximum Lifetime Data Aggregation (MLDA) problem can be solved to find an optimal schedule of when sensors should transmit data and where data should be aggregated [15]. Similar work in [1] can be used to find the upper bound of attainable lifetime using sensor management algorithms that determine sensing, routing and data fusing roles for each sensor in the network.

The above optimization problems provide upper bounds on achievable lifetime, but they are typically not feasible in real sensor networks for three reasons. First, each of these optimization problems require global knowledge of every sensor's location and battery level. Second, the amount of computation required to solve these optimization problems becomes prohibitive for large-scale networks. Finally, the solution to these optimization problems is not robust to node failures or changes in network topology (e.g., addition of new nodes, node mobility, etc.). If, for example, a critical router node fails, data may not reach the sink(s) for the entire duration of that sensor's scheduled time. Therefore, there is a need for more feasible sensor management solutions, whose properties generally include:

- significant network lifetime extension, achieved by choosing sensor roles (sensing, routing) to reduce energy dissipation at each sensor whenever possible while meeting application QoS,
- scalability for large numbers of sensors,
- distributed control and decision-making,
- robustness to individual node failures, and
- low overhead and computational feasibility.

In the remainder of this chapter, currently proposed solutions for topology control and sensing mode selection that aim to achieve these important sensor management goals are described.

2. Topology Control Algorithms

The topic of topology control in general ad hoc networks has been studied extensively. The purpose of traditional topology control protocols has been to balance two contradictory goals—reducing energy consumption and maintaining high connectivity. Most early topology control protocols adjusted radio settings (e.g., transmission power [2, 23, 24, 29], beamforming patterns [16]) to maintain connectivity with an optimal set of neighbors. Because it is often more power-efficient to relay packets over several short hops than a single long hop, reducing transmission power is an effective means for reducing overall energy consumption. Reducing transmission power also allows the network to benefit from spatial reuse, possibly resulting in reduced congestion, higher throughput, and a reduction in the number of costly data packets that are unnecessarily overheard.

These methods may be very effective in sensor networks where energy consumption is dominated by the energy consumed in transmitting data packets. However, typical power models considered for sensor networks show that receive power and idle power are comparable to transmit power [26]. Based on this observation, further savings can surely be achieved by not only reducing transmission power, but also setting the sensors' radios into a sleep state whenever possible. Below, several topology control protocols that achieve energy efficiency through these means are described. While some of these protocols were originally designed for use in general ad hoc networks, the fact that nodes are often allowed to turn their radios off nevertheless makes them suitable protocols for sensor networks as well.

GAF: Geographic Adaptive Fidelity

The GAF protocol [31] takes advantage of the fact that neighboring nodes are often nearly identical from the perspective of data routing. In GAF, a virtual grid is formed throughout the deployed network, and each node is assigned to the virtual grid cell in which it resides. Only a single node from a cell in the virtual grid is chosen to be active at any given time (see Figure 1.1). Nodes implementing GAF initially enter a discovery state, where they listen for messages from other nodes within their cell. If the node determines that a more suitable node can handle the routing responsibilities for its cell, it falls into a sleep state, from which it periodically reenters the discovery state; otherwise, it enters the active state and participates in data routing. After a predetermined active period, active nodes fall back into the discovery state. As the density of a network implementing GAF increases, the number of activated

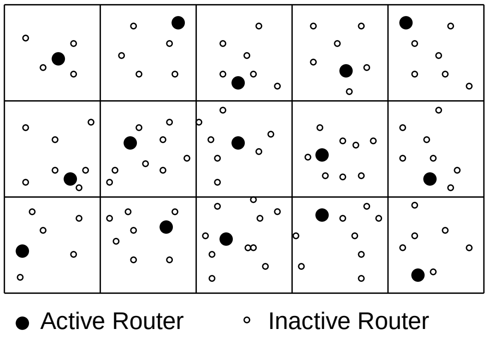


Figure 1.1. Example of a GAF virtual grid [31]. Only one node per cell is activated as a router.

nodes per grid cell remains constant while the number of nodes per cell increases proportionally. Thus, GAF can extend lifetime approximately linearly as a function of node density.

Span

Span [4] is a topology control protocol that allows nodes that are not involved in a routing backbone to sleep for extended periods of time. In Span, certain nodes assign themselves the position of "coordinator." These coordinator nodes are chosen to form a backbone of the network, so that the capacity of the backbone approaches the potential capacity of the complete network. Periodically, nodes that have not assigned themselves the coordinator role initiate a procedure to decide if they should become a coordinator. The criteria for this transition is if the minimum distance between any two of the node's neighbors exceeds three hops. To avoid the situation where many nodes simultaneously decide to become coordinators, backoff delays are added to nodes' coordinator announcement messages. The backoff delays are chosen such that nodes with higher remaining energy and those potentially providing more connectivity in their neighborhood are more likely to become a coordinator. To ensure a balance in energy consumption among the nodes in the network, coordinator nodes may fall back from their coordinator role if neighboring nodes can make up for the lost connectivity in the region.

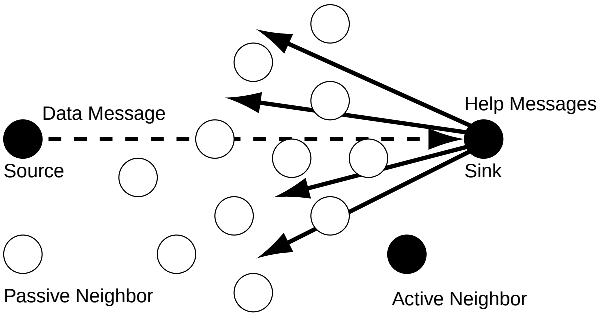
ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies

ASCENT [3] is similar to Span in that certain nodes are chosen to remain active as routers while others are allowed to conserve energy in a sleep state. In ASCENT, the decision to become an active router is based not only on neighborhood connectivity, but also on observed data loss rates, providing the network with the ability to trade energy consumption for communication reliability. Nodes running the ASCENT protocol initially enter a test state where they actively participate in data routing, probe the channel to discover neighboring sensors and learn about data loss rates, and send their own "Neighborhood Announcement" messages. If, based on the current number of neighbors and current data loss rates, the sensor decides that its activation would be beneficial to the network, it becomes active and remains so permanently. If the sensor decides not to become active, it falls into a passive state, where it gathers the same information as it does in the test state (as well as any "Help" messages from neighboring sensors experiencing poor communication links), but it does not actively participate in data routing. From this state, the node may reenter the test state if the information gathered indicates poor neighborhood communication quality, or enter the sleep state, turning its radio off and saving energy. The node periodically leaves the sleep state to listen to the channel from the passive state.

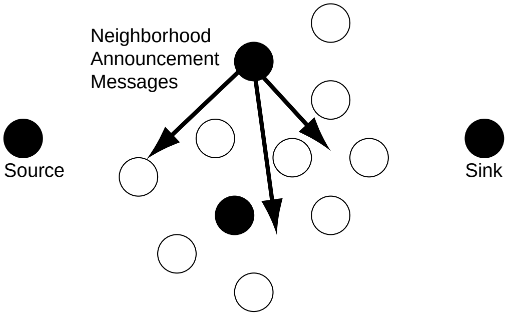
One procedure for node activation is shown in Figure 1.2. Figure 1.2a illustrates a situation in which the channel quality between a source and its sink becomes poor, prompting the sink to broadcast "Help" messages to its neighbors. A limited number of intermediate nodes that receive these messages enter the test state and broadcast "Neighborhood Announcement" messages (Figure 1.2b) to inform their neighbors that they intend to become active. Alternatively, this transition to the test state can be triggered by a node observing that it has a low number of active neighbors or high data loss rate. The network eventually stabilizes at a point in which the communication between the source and sink becomes adequately reliable, as shown in Figure 1.2c.

STEM: Sparse Topology and Energy Management

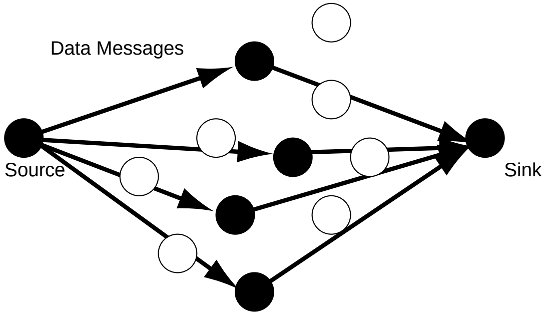
In the case of many sensor network applications, it is expected that nodes will continuously sense the environment but transmit data to a base station very infrequently or only when an event of interest has occurred. While GAF, Span, and ASCENT save energy by reducing the number of sensors used for routing, the energy consumption of the selected routers can be further reduced by exploiting the low traffic generation rates of the sensors. STEM [25] takes advantage of this by leaving all sensors in a sleep state while monitoring the environment but not sending data, which is assumed to be the majority of time. STEM is quite different from the rest of the topology control protocols described



(a) Communication Hole



(b) Transition



(c) Final State

Figure 1.2. One procedure for the activation of inactive routers in ASCENT [3]. When the channel quality is poor, a node will broadcast "Help" messages to its neighbors (a), some of whom will become active and broadcast "Neighborhood Announcement" messages (b). After these nodes are activated, the communication quality improves (c).

here in that it activates nodes reactively rather than proactively. When data packets are generated, the sensor generating the traffic uses a paging channel (separate from the data channel) to awaken its downstream neighbors. Two versions of STEM have been proposed—STEM-T, which uses a tone to wake neighboring nodes, and STEM-B, in which the traffic generating node sends beacons on the paging channel and sleeping nodes turn on their radios with a low duty cycle to receive the messages. A benefit of STEM is that it can be combined with any of the aforementioned protocols, with only the currently activated nodes required to listen on the paging channel.

3. Sensing Mode Selection

The previous section outlined some topology control protocols that limit energy consumption by reducing the number of nodes actively participating in data routing. To further exploit potential energy savings in a wireless sensor network, careful consideration should be taken in the selection of the sensing modes of the nodes in the network. This can be as simple as determining which sensors should be activated or deactivated or as complex as determining certain features of the sensing, such as sensing frequency and data resolution. By doing so, the designer is not only influencing how the data are routed, but also what traffic is generated in the first place. This aspect of sensor management has several implications. First, it may be absolutely necessary. In simple coverage applications, if the sensing range is significantly larger than the transmission range, some of the previously described topology control protocols may be suitable alone. However, if this is not the case, additional sensors besides the selected active routers will need to be activated. Also, with the use of an energy-efficient MAC protocol [8, 9, 34], the average energy consumption of a node when it is not transmitting or receiving data may not approach the levels of the transmit and receive power, as was the assumption for the topology control protocols. In this case, the amount of data generated on the network may influence network energy consumption (and consequently, network lifetime) in a more drastic manner. Furthermore, deactivating sensors can reduce any congestion that may arise in low bandwidth networks [28].

In this section, several protocols for sensing mode selection in wireless sensor networks are described. In most cases, sensing mode selection is as simple as deciding which sensors should send data to a base station or other nodes. The process of determining the subset of sensors chosen for activation should be influenced by QoS (or fidelity) requirements of the application. Given the many proposed applications for wireless sensor networks, these requirements can be quite wide-ranging. In this chapter, the focus is primarily on applications that require coverage of the entirety or a portion of a region where the sensors are deployed [19]. Examples of sensing mode selection in other applications include

- edge detection [18], where only a certain subset of the nodes is expected to change state when the edge moves,
- oversampling of bandlimited phenomena [12], where nodes can adjust the resolution of their data in dense regions without significant information loss,
- target localization/tracking [20, 35], where selection of only the subset of sensors in the current vicinity of the target is beneficial, and
- general target classification [5, 6].

The reader is referred to the cited literature for more detailed descriptions of this work.

The primary goal of the protocols described in this section is coverage preservation rather than network connectivity (as in Section 1.2). Coverage preserving protocols and algorithms have many potential applications, including intruder detection, biological/chemical agent detection, and fire detection. Also, these protocols and algorithms can be used in the initial stages of many target tracking architectures, where a more detailed description or location estimate of a phenomenon is required only when a "tripwire" threshold is crossed in the measurements of some of the active sensors [11].

PEAS: Probing Environment and Adaptive Sleeping

PEAS [33] is a protocol that was developed to provide consistent environmental coverage and robustness to unexpected node failures. Nodes begin in a sleeping state, from which they periodically enter a probing state. In the probing state, a sensor transmits a probe packet, to which its neighbors will reply after a random backoff time if they are within the desired probing range. If no replies are received by the probing node, the probing sensor will become active; otherwise, it will return to the sleep state. The probing range is chosen to meet the more stringent of the density requirements imposed by the sensing radius and the transmission radius. The probing rate of PEAS is adaptive and is adjusted to meet a balance between energy savings and robustness. Specifically, a low probing rate may incur long delays before the network recovers following an unexpected node failure. On the other hand, a high probing rate may lead to expensive energy waste. Basically, the probing rate of individual nodes should increase as more node failures arise, so that a consistent expected recovery time is maintained.

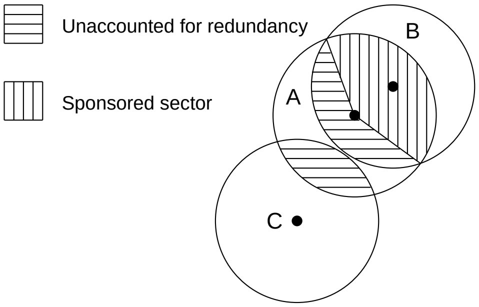


Figure 1.3. A sponsored sector, as defined by NSSS [27]. Sensor A admits the redundant coverage of sensor B in the vertically shaded regions. The additional redundancy of sensors B and C shown in the horizontally shaded regions is not accounted for.

NSSS: Node Self-Scheduling Scheme

A node self-scheduling scheme for coverage preservation in sensor networks is presented in [27]. In NSSS, a node measures its neighborhood redundancy as the union of the sectors/central angles covered by neighboring sensors within the node's sensing range. At decision time, if the union of a node's "sponsored" sectors covers the full 360° (see Figure 1.3), the node will decide to power off. It should be noted that additional redundancy may exist between sensors and that the redundancy model is simplified at a cost of not being able to exploit this redundancy. In NSSS, at the beginning of each round, there is a short self-scheduling phase where nodes first exchange location information and then decide whether or not to turn off after some backoff time. NSSS avoids scenarios of unattended areas due to the simultaneous deactivation of nodes by requiring nodes to double check their eligibility to turn off after making the decision.

Gur Game

In [13], the problem of optimal sensor selection for a given resolution was modeled as a Gur game. It is assumed that a base station wishes to receive packets from a predetermined number of sensors. Each sensor operates as a single chain finite state machine, where one side of the chain represents states in which the sensor sends a packet to the base station and the other side represents states in which the sensor does not (see Figure 1.4). After each round, the base station calculates a reward probability r from a function of the number of packets received. The reward function has its maximum value at the desired resolution

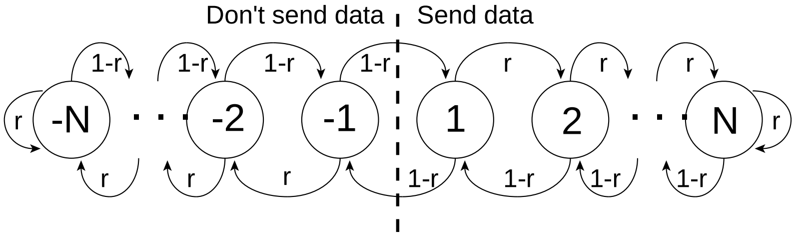


Figure 1.4. State machine for the Gur game at an individual node [13].

(number of packets received). The base station broadcasts the value of this reward function to the sensors, which move toward the edges of their state machine with probability r and toward the center with probability $(1 - r)$. The network settles at the desired resolution and is robust to sensor failures and additional sensor deployment. This work was further developed into a hierarchical model, providing better scalability, in [14].

Reference Time-based Scheduling Scheme

In the reference time-based scheduling scheme presented in [32], the environment is divided into a grid and coverage is maintained continuously at every grid point while minimizing the number of active sensors. During an initialization process, each node broadcasts a randomly chosen reference time (uniformly distributed on $[0, T)$, where T is the round length) to all neighboring sensors within twice its sensing radius. For each location in the grid that the sensor is capable of monitoring, a sensor sorts the reference times of all sensors capable of monitoring that grid point. For a given grid point, the sensor schedules itself to be active beginning halfway between its reference time and the reference time of the sensor immediately preceding it in the sorted list. Similarly, its scheduled slot for the grid point ends halfway between its reference time and the reference time of the sensor immediately after it in the sorted list (see Figure 1.5). The sensor remains active during the union of the scheduled slots calculated for each grid point within its sensing range. This algorithm is also enhanced to guarantee coverage by multiple sensors in selected areas as well as provide robustness to node failures.

CCP: Coverage Configuration Protocol

In CCP [30], an eligibility rule is proposed to maintain a certain degree of coverage (coverage at every location by a given number of sensors).

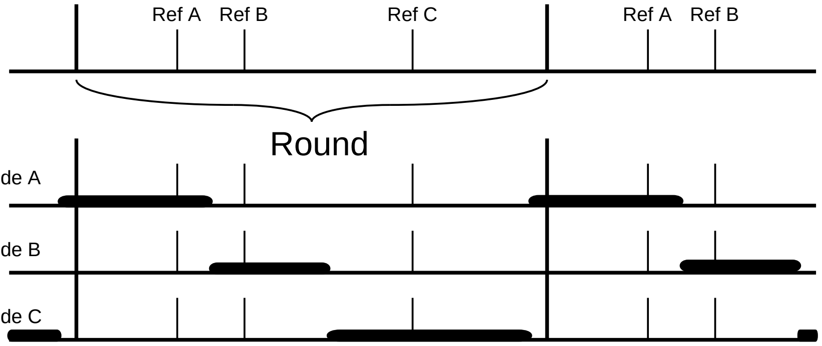
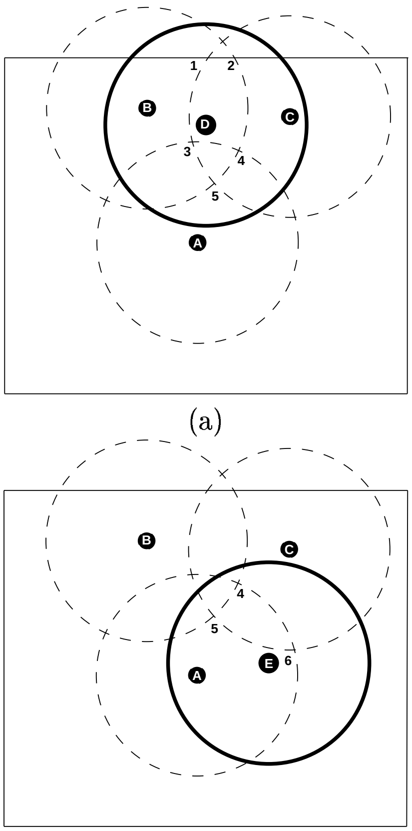


Figure 1.5. Schedule calculation for a single grid point, as proposed in the reference time-based scheduling scheme [32]. After all grid point schedules are calculated, the schedules are merged and a sensor's overall schedule is the union of all of its grid point schedules.

First, each node finds all intersection points between the borders of its neighbors' sensing radii and any edges in the desired coverage area. The CCP rule assigns a node as eligible for deactivation if each of these intersection points is K-covered, where K is the desired sensing degree. The CCP scheme assumes a Span-like protocol and state machine that can use the Span rule for network connectivity or the proposed CCP rule for K-coverage, depending on the application requirements and the relative values of the communication radius and sensing radius. An example of how the CCP rule is applied is given in Figure 1.6. In Figure 1.6a, node D, whose sensing range is represented by the bold circle, must decide whether it should become active in order to meet a coverage constraint of $K = 1$. It is assumed that D knows that A, B, and C, whose sensing ranges are represented by the dashed circles, are currently active. The intersection points within D's sensing range are found and enumerated 1-5 in the figure. Since B covers points 1 and 3, C covers points 2 and 4, and A covers point 5, D deduces that the coverage requirements have already been met and remains inactive. In the case of node E, illustrated in Figure 1.6b, there is an intersection point (labeled 6 in the figure) that is not covered by any of E's neighbors. Thus, E must become active and sense the environment.

4. Integrated Sensing Mode Selection and Routing Protocols

Sensing mode selection can potentially have implications on topology control and data routing. Accordingly, some sensor management

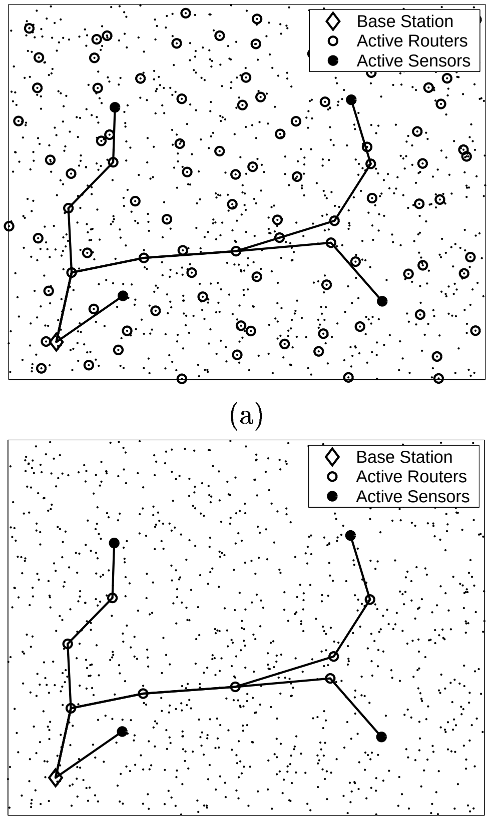


(a)

(b)

Figure 1.6. Illustration of the CCP activation rule for K-coverage, K = 1 [30]. Nodes D and E decide whether or not to become active in (a) and (b), respectively, knowing that neighbors A, B, and C, are already active. Node D may remain inactive since all of its intersection points are K-covered, but the CCP rule dictates that E must become active since intersection point 6 is not covered by any of E's neighbors.

protocols integrate these functions [7, 21]. To see why this is beneficial, consider that topology control protocols that do not use knowledge of traffic patterns may activate many more nodes than the necessary amount to accommodate the sensed data. For example, in the network shown in Figure 1.7, four nodes are required to send raw data to the base station. Without knowledge of traffic patterns and using the GAF topology control protocol discussed previously, the nodes selected as routers might be the set shown in Figure 1.7(a). Given that traffic will only be generated from the small subset of selected sensor nodes, many of these nodes should actually be allowed to sleep, and only the nodes used in



(a)

(b)

Figure 1.7. Example of a situation where knowledge of traffic patterns can potentially benefit topology control. In (a), many nodes have unnecessarily been activated as routers. In (b), only the necessary routers for the given traffic conditions are activated.

the optimally calculated routes, shown in Figure 1.7(b), need to remain active.

Connected Sensor Cover

The Connected Sensor Cover algorithm presented in [7] provides a joint sensing mode selection and topology control solution. The problem addressed in this work is to find a minimum set of sensors and additional routing nodes necessary in order to efficiently process a query over a given geographical region. In the centralized version of the algorithm, an initial sensor within the query region is randomly chosen, following which additional sensors are added by means of a greedy algorithm.

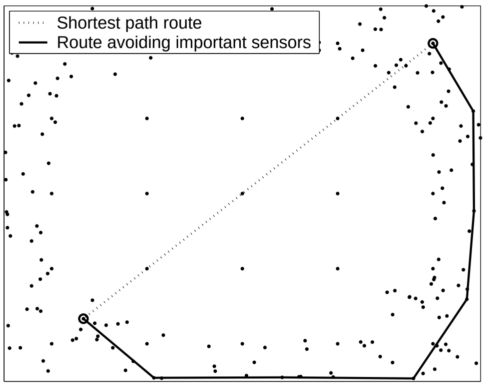


Figure 1.8. Calculated routes using shortest path and an approach that considers the importance of the sensors to the application.

At each step in this algorithm, all sensors that redundantly cover some area that is already covered by the current active subset are considered candidate sensors and calculate the shortest path to one of the sensors already included in the current active subset. For each of these candidate sensors, a heuristic is calculated based on the number of unique sections in the query region that the sensor and its routers would potentially add and the number of sensors on its calculated path. The sensor with the most desirable heuristic value and those along its path are selected for inclusion in the sensor set. This process continues until the query region is entirely covered. The algorithm has been extended to account for node weighting, so that low energy nodes can be avoided, and to be implemented through distributed means, with little loss in solution optimality compared with the centralized version.

DAPR: Distributed Activation with Pre-determined Routes

The integration of sensing mode selection and data routing/topology control can also be accomplished by choosing routes so as to avoid sensors that are critical for maintaining the desired coverage. Sensors that are more important to the sensing application and those whose potential active lifetime is shortest should not be chosen as routers over those who are less important to the application and those with potentially long active lifetimes (see Figure 1.8).

DAPR [21] is an algorithm for integrated sensing mode selection and data routing that follows this intuition. In DAPR, nodes assign themselves an application cost—a measure of their value to the application. The application cost is calculated based on a node's remaining energy and the remaining energy of the node's neighbors that provide redundant coverage. The application cost was chosen to consider a node's neighbors' energy as well as its own because neighbors covering the same region redundantly are equivalent from the application's perspective. This is because DAPR is less concerned with prolonging individual sensors' lifetime as it is with prolonging the time before a spot in the network is left uncovered. A round in DAPR begins with a route discovery phase, in which the base station broadcasts a query packet across the network and nodes discover their optimal routes based on smallest cumulative application cost along the path. Following this phase is a role discovery phase, consisting of an Opt In and an Opt Out portion. In the Opt In portion, nodes attempt to activate themselves after a backoff delay proportional to their cumulative route cost. If a sensor discovers that its neighborhood has already been covered by the time it wishes to broadcast its activation message, it withholds this message. In the Opt Out phase, nodes check to see if their neighborhood has been completely redundantly covered by other nodes, some of which may have been activated after their own activation message was broadcast. Nodes are given the opportunity to opt out in the reverse order in which they opted in, so that nodes with high cumulative route cost are still favored for deselection. After this set-up phase, activated sensors send data along the pre-calculated routes to the base station.

5. Discussion

The properties of the protocols and schemes described in this chapter are summarized in Table 1.1. It should be noted that the classifications and descriptions in this table are not straightforward and that simple modifications can often eliminate any undesirable properties of a protocol. For example, some topology control protocols could be used as sensing mode selection protocols with some simple modifications, and vice versa. Also, some of the protocols that require location information can be modified to operate without location information, but this usually comes at the cost of reduced performance (e.g., higher energy dissipation, no application QoS guarantees, etc.). For example, DAPR can be modified so that coverage quality is measured as the number of active neighbors that a node overhears rather than the actual calculation of area covered by the active neighbors. This approach removes the need for location information at the cost of not guaranteeing complete coverage. In general, the protocols are summarized in Table 1.1 as they were presented in their original literature.

This chapter has described methods for the efficient management of the scarce resources typically available in wireless sensor networks. Specifically, some proposed methods for choosing which sensors should act as routers (through topology control protocols) and which should actively sense the environment and send traffic to any data sinks in the network (through sensing mode selection) have been presented. The choice of which protocol to use in a specific network is subject to factors such as

- the application requirements,
- the choice of MAC protocol,
- the bandwidth resources, and
- the availability of certain network services (e.g., sensor node localization, time synchronization).

For example, in applications where data need to be sent only after an event of interest occurs, a reactive solution such as STEM [25] might be preferable. However, in applications that require constant streaming traffic and have little tolerance for delays, protocols such as GAF [31] or Span [4] that proactively maintain a connected network might be desirable. Application requirements also dictate what level of service is needed. For example, an application that requires an absolute guarantee that the entirety of a region is covered by at least one sensor may want to use an approach such as the node self-scheduling scheme in [27] or the reference time-based scheduling scheme in [32], whereas if the application requires only that a given number of sensors send data, an approach such as the Gur game modeling presented in [13] may be acceptable and require less communication overhead, enabling the sensor nodes to save even more energy and extend network lifetime.

When using a MAC protocol that consumes significant idle power, topology control protocols by themselves may be suitable since the power consumption is dominated by the idle power consumption of the nodes chosen as active routers and the traffic induced on the network may be of little consequence from an energy consumption standpoint. With a MAC protocol that allows nodes to sleep during periods of inactivity, however, sensing mode selection protocols can be much more effective in reducing power consumption and extending network lifetime. Also, sensing mode selection protocols may relieve network congestion in low bandwidth networks.

Finally, a cross-layer approach that selects routes in conjunction with choosing sensors to sense the environment may be the best approach in many applications with static sensor nodes and data sinks. However, in some situations, such as battlefield scenarios in which mobile soldiers and vehicles may act as sensors or data sinks, the network may consist of a dynamic topology. With constant route updates becoming necessary, a layered approach that handles sensing mode selection and topology control/routing separately may be necessary.

In addition to the important roles of sensing and routing, careful consideration should be taken when determining which nodes should assume other network roles such as cluster heads and data fusion points [10, 17]. While this chapter has not addressed this, some research has been devoted to this area.

Protocol	TC/MS	Loc?	D/C	Energy balancing	Robustness	Comp.	Overhead
GAF [31]	TC	Y	D	Activity rotation	Good	Low	Neighbor state exchange
Span [4]	TC	N	D	Activity rotation, energy weighting	Good	Low	Neighbor state exchange
ASCENT [3]	TC	N	D	No	Good	Low	Neighbor state exchange
STEM [25]	TC	N	D	N/A	Good	Low	Paging channel
PEAS [33]	TC,MS	Y	D	No	Good, dep. on probing frequency	Low	Neighbor state exchange
NSSS [27]	MS	Y	D	No	Dep. on round ln.	Low	Neighbor state exchange
Gur game [13]	MS	N	D	Dep. on N, reward function	Coverage not guaranteed	Low	BS queries
Ref. time-based scheduling [32]	MS	Y	D	Activity rotation	K-degree coverage	Low	Reference time exchange
CCP [30]	TC,MS	Y	D	Activity rotation	K-degree coverage	Low	Neighbor state exchange
Connected Sensor Cover [7]	TC,MS	Y	C/D	Energy weighting possible	Dep. on query frequency	Low	BS queries, neighbor state exchange (D)
DAPR [21]	MS	Y	D	Activity rotation, energy weighting	Dep. on round ln.	Low	BS queries, neighbor state exchange
Optimization	MS	Y	C	Yes	Poor	High	N/A

TC/MS:Topology Control/Mode Selection   Loc?:Location information required?   D/C:Distributed/Centralized

Table 1.1. Summary of protocols/schemes described.

References

[1] Bhardwaj, M. and A. Chandrakasan: 2002, 'Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments'. In: *Proceedings of the Twenty First International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM).*

[2] Blough, D., M. Leoncini, G. Resta, and P. Santi: 2003, 'The K-Neigh Protocol for Symmetric Topology Control in Ad Hoc Networks'. In: *Proceedings of MobiHOC.*

[3] Cerpa, A. and D. Estrin: 2002, 'ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies'. In: *Proceedings of the Twenty First International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM).*

[4] Chen, B., K. Jamieson, H. Balakrishnan, and R. Morris: 2000, 'Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks'. In: *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking.*

[5] Chu, M., H. Haussecker, and F. Zhao: 2002, 'Scalable Information-Driven Sensor Querying and Routing for Ad Hoc Heterogeneous Sensor Networks'. *International Journal of High Performance Computing Applications* **16**(3).

[6] Ertin, E., J. W. F. III, and L. C. Potter: 2003, 'Maximum Mutual Information Principle for Dynamic Sensor Query Problems'. In: *Proceedings of the Second International Workshop on Information Processing in Sensor Networks.*

[7] Gupta, H., S. Das, and Q. Gu: 2003, 'Connected Sensor Cover: Self-Organization of Sensor Networks for Efficient Query Execution'. In: *Proceedings of the Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc).*

[8] Gutierrez, J. A., M. Naeve, E. Callaway, M. Bourgeois, V. Mitter, and B. Heile: 2001, 'IEEE 802.15.4: A Developing Standard for Low-Power, LowCost Wireless Personal Area Networks'. *IEEE Network* **15**(5), 12–19.

[9] Haartsen, J., M. Naghshineh, J. Inouye, O. Joeressen, and W. Allen: 1998, 'Bluetooth: Vision, Goals, and Architecture'. *Mobile Computing and Communications Review* **2**(4), 38–45.

[10] Heinzelman, W., A. Chandrakasan, and H. Balakrishnan: 2002, 'An Application-Specific Protocol Architecture for Wireless Microsensor Networks'. *IEEE Transactions on Wireless Communications* **1**(4), 660–670.

[11] Hui, J., Z. Ren, and B. H. Krogh: 2003, 'Sentry-Based Power Management in Wireless Sensor Networks'. In: *Proceedings of the Second International Workshop on Information Processing in Sensor Networks.*

[12] Ishwar, P., A. Kumar, and K. Ramchandran: 2003, 'Distributed Sampling for Dense Sensor Networks: A Bit-Conservation Principle'. In: *Proceedings of the Second International Workshop on Information Processing in Sensor Networks.*

[13] Iyer, R. and L. Kleinrock: 2003a, 'QoS Control For Sensor Networks'. In: *Proceedings of the IEEE International Conference on Communications.*

[14] Iyer, R. and L. Kleinrock: 2003b, 'Scalable Sensor Network Resolution'. In: *Proceedings of Sensys.*

[15] Kalpakis, K., K. Dasgupta, and P. Namjoshi: 2002, 'Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks'. In: *Proceedings of the 2002 IEEE International Conference on Networking.*

[16] Li, Q. and D. Rus: 2000, 'Sending Messages to Mobile Users in Disconnected Ad-Hoc Wireless Networks'. In: *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking.*

[17] Lindsey, S., C. Raghavendra, and K. Sivalingam: 2002, 'Data Gathering Algorithms in Sensor Networks Using Energy Metrics'. *IEEE Transactions on Parallel and Distributed Systems* **13**(9), 924–935.

[18] Liu, J., P. Cheung, L. Guibas, and F. Zhao: 2002, 'A Dual-Space Approach to Tracking and Sensor Management in Wireless Sensor

Networks'. In: *Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications.*

[19] Meguerdichian, S., F. Koushinfar, M. Potkonjak, and M. Srivastava: 2001, 'Coverage Problems in Wireless Ad-hoc Sensor Networks'. In: *Proceedings of the Twentieth International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM).*

[20] Pattem, S., S. Poduri, and B. Krishnamachari: 2003, 'Energy-Quality Tradeoffs for Target Tracking in Wireless Sensor Networks'. In: *Proceedings of the Second International Workshop on Information Processing in Sensor Networks.*

[21] Perillo, M. and W. Heinzelman: 2003a, 'Sensor Management and Routing Protocols For Prolonging Network Lifetime (Technical Report)'. In: *University of Rochester Technical Report.*

[22] Perillo, M. and W. Heinzelman: 2003b, 'Simple Approaches for Providing Application QoS Trhough Intelligent Sensor Management'. *Elsevier Ad Hoc Networks Journal* **1**(2-3), 235–246.

[23] Ramanathan, R. and R. Hain: 2000, 'Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment'. In: *Proceedings of the Nineteenth International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM).*

[24] Rodoplu, V. and T. Meng: 1998, 'Minimum Energy Mobile Wireless Networks'. In: *Proceedings of the IEEE International Conference on Communications.*

[25] Schurgers, C., V. Tsiatsis, S. Ganeriwal, and M. Srivastava: 2002, 'Optimizing Sensor Networks in the Energy-Latency-Density Design Space'. *IEEE Transactions on Mobile Computing* **1**(1), 70–80.

[26] Stemm, M. and R. H. Katz: 1997, 'Measuring and reducing energy consumption of network interfaces in hand-held devices'. *IEICE Transactions on Communications* **E80-B**(8), 1125–31.

[27] Tian, D. and N. Georganas: 2003, 'A Node Scheduling Scheme for Energy Conservation in Large Wireless Sensor Networks'. *Wireless Communications and Mobile Computing Journal* **3**(2), 271–290.

[28] Tilak, S., N. Abu-Ghazaleh, and W. Heinzelman: 2002, 'Infrastructure Tradeoffs for Sensor Networks'. In: *Proceedings of the First*

ACM International Workshop on Wireless Sensor Networks and Applications.

[29] Tseng, Y., Y. Chang, and P. Tseng: 2002, 'Energy-Efficient Topology Control for Wireless Ad Hoc Sensor Networks'. In: *Proceedings of the International Computer Symposium.*

[30] Wang, X., G. Xing, Y. Zhang, C. Lu, R. Pless, , and C. Gill: 2003, 'Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks'. In: *Proceedings of Sensys.*

[31] Xu, Y., J. Heidemann, and D. Estrin: 2001, 'Geography-informed Energy Conservation for Ad Hoc Routing'. In: *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking.*

[32] Yan, T., T. He, and J. A. Stankovic: 2003, 'Differentiated Surveillance for Sensor Networks'. In: *Proceedings of Sensys.*

[33] Ye, F., G. Zhong, J. Cheng, S. Lu, and L. Zhang: 2003, 'PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks'. In: *Proceedings of the Twenty Third International Conference on Distributed Computing Systems.*

[34] Ye, W., J. Heidemann, and D. Estrin: 2002, 'An Energy-Efficient MAC Protocol for Wireless Sensor Networks'. In: *Proceedings of the Twenty First International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM).*

[35] Zou, Y. and K. Chakrabarty: 2003, 'Target Localization Based on Energy Considerations in Distributed Sensor Networks'. In: *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications.*