A survey on routing protocols for wireless sensor networks

Kemal Akkaya *, Mohamed Younis

Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, Baltimore, MD 21250, USA

Abstract

Recent advances in wireless sensor networks have led to many new protocols specifically designed for sensor networks where energy awareness is an essential consideration. Most of the attention, however, has been given to the routing protocols since they might differ depending on the application and network architecture. This paper surveys recent routing protocols for sensor networks and presents a classification for the various approaches pursued. The three main categories explored in this paper are data-centric, hierarchical and location-based. Each routing protocol is described and discussed under the appropriate category. Moreover, protocols using contemporary methodologies such as network flow and quality of service modeling are also discussed. The paper concludes with open research issues.
© 2003 Elsevier B.V. All rights reserved.

Keywords: Sensor networks; Energy-aware routing; Routing protocols; Classification of protocols

1. Introduction

Recent advances in micro-electro-mechanical systems and low power and highly integrated digital electronics have led to the development of micro-sensors [1–5]. Such sensors are generally equipped with data processing and communication capabilities. The sensing circuitry measures ambient conditions related to the environment surrounding the sensor and transforms them into an electric signal. Processing such a signal reveals some properties about objects located and/or events happening in the vicinity of the sensor. The sensor sends such collected data, usually via radio transmitter, to a command center (sink) either directly or through a data concentration center (a gateway). The decrease in the size and cost of sensors, resulting from such technological advances, has fueled interest in the possible use of large set of disposable unattended sensors. Such interest has motivated intensive research in the past few years addressing the potential of collaboration among sensors in data gathering and processing and the coordination and management of the sensing activity and data flow to the sink. A natural architecture for such collaborative distributed sensors is a network with wireless links that can be formed among the sensors in an ad hoc manner.

Networking unattended sensor nodes are expected to have significant impact on the efficiency of many military and civil applications such as combat field surveillance, security and disaster management. These systems process data gathered

---

* Corresponding author.
  E-mail addresses: kemall@csee.umbc.edu (K. Akkaya), younis@csee.umbc.edu (M. Younis).

from multiple sensors to monitor events in an area of interest. For example, in a disaster management setup, a large number of sensors can be dropped by a helicopter. Networking these sensors can assist rescue operations by locating survivors, identifying risky areas and making the rescue crew more aware of the overall situation. Such application of sensor networks not only can increase the efficiency of rescue operations but also ensure the safety of the rescue crew. On the military side, applications of sensor networks are numerous. For example, the use of networked set of sensors can limit the need for personnel involvement in the usually dangerous reconnaissance missions. In addition, sensor networks can enable a more civic use of landmines by making them remotely controllable and target-specific in order to prevent harming civilians and animals. Security applications of sensor networks include intrusion detection and criminal hunting.

However, sensor nodes are constrained in energy supply and bandwidth. Such constraints combined with a typical deployment of large number of sensor nodes have posed many challenges to the design and management of sensor networks. These challenges necessitate energy-awareness at all layers of networking protocol stack. The issues related to physical and link layers are generally common for all kind of sensor applications, therefore the research on these areas has been focused on system-level power awareness such as dynamic voltage scaling, radio communication hardware, low duty cycle issues, system partitioning, energy-aware MAC protocols [6–10]. At the network layer, the main aim is to find ways for energy-efficient route setup and reliable relaying of data from the sensor nodes to the sink so that the lifetime of the network is maximized.

Routing in sensor networks is very challenging due to several characteristics that distinguish them from contemporary communication and wireless ad hoc networks. First of all, it is not possible to build a global addressing scheme for the deployment of sheer number of sensor nodes. Therefore, classical IP-based protocols cannot be applied to sensor networks. Second, in contrary to typical communication networks almost all applications of sensor networks require the flow of sensed data from multiple regions (sources) to a particular

sink. Third, generated data traffic has significant redundancy in it since multiple sensors may generate same data within the vicinity of a phenomenon. Such redundancy needs to be exploited by the routing protocols to improve energy and bandwidth utilization. Fourth, sensor nodes are tightly constrained in terms of transmission power, on-board energy, processing capacity and storage and thus require careful resource management.

Due to such differences, many new algorithms have been proposed for the problem of routing data in sensor networks. These routing mechanisms have considered the characteristics of sensor nodes along with the application and architecture requirements. Almost all of the routing protocols can be classified as data-centric, hierarchical or location-based although there are few distinct ones based on network flow or quality of service (QoS) awareness. Data-centric protocols are query-based and depend on the naming of desired data, which helps in eliminating many redundant transmissions. Hierarchical protocols aim at clustering the nodes so that cluster heads can do some aggregation and reduction of data in order to save energy. Location-based protocols utilize the position information to relay the data to the desired regions rather than the whole network. The last category includes routing approaches that are based on general network-flow modeling and protocols that strive for meeting some QoS requirements along with the routing function. In this paper, we will explore the routing mechanisms for sensor networks developed in recent years. Each routing protocol is discussed under the proper category. Our aim is to help better understanding of the current routing protocols for wireless sensor networks and point out open issues that can be subject to further research.

The paper is organized as follows. In the balance of this section, we will briefly summarize the system architecture design issues for sensor networks and their implications on data routing. We then set our work apart from prior surveys on sensor networks. In the Section 2, data-centric routing approaches are covered. Section 3 summarizes hierarchical routing protocols. Location-based routing in sensor networks is discussed in Section 4. In Section 5, we describe other routing approaches that are based on network flow or QoS

modeling. Finally, Section 6 concludes the paper with a comparative summary of the surveyed approaches and points out open research problems.

1.1. System architecture and design issues

Depending on the application, different architectures and design goals/constraints have been considered for sensor networks. Since the performance of a routing protocol is closely related to the architectural model, in this section we strive to capture architectural issues and highlight their implications.

1.1.1. Network dynamics

There are three main components in a sensor network. These are the sensor nodes, sink and monitored events. Aside from the very few setups that utilize mobile sensors [11], most of the network architectures assume that sensor nodes are stationary. On the other hand, supporting the mobility of sinks or cluster-heads (gateways) is sometimes deemed necessary [12]. Routing messages from or to moving nodes is more challenging since route stability becomes an important optimization factor, in addition to energy, bandwidth etc. The sensed event can be either dynamic or static depending on the application [13]. For instance, in a target detection/tracking application, the event (phenomenon) is dynamic whereas forest monitoring for early fire prevention is an example of static events. Monitoring static events allows the network to work in a reactive mode, simply generating traffic when reporting. Dynamic events in most applications require periodic reporting and consequently generate significant traffic to be routed to the sink.

1.1.2. Node deployment

Another consideration is the topological deployment of nodes. This is application dependent and affects the performance of the routing protocol. The deployment is either deterministic or self-organizing. In deterministic situations, the sensors are manually placed and data is routed through pre-determined paths. However in self-organizing systems, the sensor nodes are scattered randomly creating an infrastructure in an ad hoc manner [2,14–16]. In that infrastructure, the posi-

tion of the sink or the cluster-head is also crucial in terms of energy efficiency and performance. When the distribution of nodes is not uniform, optimal clustering becomes a pressing issue to enable energy efficient network operation.

1.1.3. Energy considerations

During the creation of an infrastructure, the process of setting up the routes is greatly influenced by energy considerations. Since the transmission power of a wireless radio is proportional to distance squared or even higher order in the presence of obstacles, multi-hop routing will consume less energy than direct communication. However, multi-hop routing introduces significant overhead for topology management and medium access control. Direct routing would perform well enough if all the nodes were very close to the sink [14]. Most of the time sensors are scattered randomly over an area of interest and multi-hop routing becomes unavoidable.

1.1.4. Data delivery models

Depending on the application of the sensor network, the data delivery model to the sink can be continuous, event-driven, query-driven and hybrid [13]. In the continuous delivery model, each sensor sends data periodically. In event-driven and query-driven models, the transmission of data is triggered when an event occurs or a query is generated by the sink. Some networks apply a hybrid model using a combination of continuous, event-driven and query-driven data delivery. The routing protocol is highly influenced by the data delivery model, especially with regard to the minimization of energy consumption and route stability. For instance, it has been concluded in [17] that for a habitat monitoring application where data is continuously transmitted to the sink, a hierarchical routing protocol is the most efficient alternative. This is due to the fact that such an application generates significant redundant data that can be aggregated on route to the sink, thus reducing traffic and saving energy.

1.1.5. Node capabilities

In a sensor network, different functionalities can be associated with the sensor nodes. In earlier

works [5,18,19], all sensor nodes are assumed to be homogenous, having equal capacity in terms of computation, communication and power. However, depending on the application a node can be dedicated to a particular special function such as relaying, sensing and aggregation since engaging the three functionalities at the same time on a node might quickly drain the energy of that node. Some of the hierarchical protocols proposed in the literature designate a cluster-head different from the normal sensors. While some networks have picked cluster-heads from the deployed sensors [14,20,21], in other applications a cluster-head is more powerful than the sensor nodes in terms of energy, bandwidth and memory [11,15]. In such cases, the burden of transmission to the sink and aggregation is handled by the cluster-head.

Inclusion of heterogeneous set of sensors raises multiple technical issues related to data routing [22]. For instance, some applications might require a diverse mixture of sensors for monitoring temperature, pressure and humidity of the surrounding environment, detecting motion via acoustic signatures and capturing the image or video tracking of moving objects. These special sensors either deployed independently or the functionality can be included on the normal sensors to be used on demand. Reading generated from these sensors can be at different rates, subject to diverse quality of service constraints and following multiple data delivery models, as explained earlier. Therefore, such a heterogeneous environment makes data routing more challenging.

1.1.6. Data aggregation/fusion

Since sensor nodes might generate significant redundant data, similar packets from multiple nodes can be aggregated so that the number of transmissions would be reduced. Data aggregation is the combination of data from different sources by using functions such as *suppression* (eliminating duplicates), *min, max* and *average* [23]. Some of these functions can be performed either partially or fully in each sensor node, by allowing sensor nodes to conduct in-network data reduction [18,20,24]. Recognizing that computation would be less energy consuming than communication [14], substantial energy savings can be obtained

through data aggregation. This technique has been used to achieve energy efficiency and traffic optimization in a number of routing protocols [18,20,24–27]. In some network architectures, all aggregation functions are assigned to more powerful and specialized nodes [11]. Data aggregation is also feasible through signal processing techniques. In that case, it is referred as *data fusion* where a node is capable of producing a more accurate signal by reducing the noise and using some techniques such as *beamforming* to combine the signals [14].

1.2. Related work

The growing interest in wireless sensor networks and the continual emergence of new architectural techniques inspired some previous efforts for surveying the characteristics, applications and communication protocols for such a technical area [1,13]. In this subsection we highlight the features that distinguish our survey and hint the difference in scope.

The goal of [1] is to make a comprehensive survey of design issues and techniques for sensor networks describing the physical constraints on sensor nodes and the protocols proposed in all layers of network stack. Possible applications of sensor networks are also discussed. That survey is a good introductory for readers interested in the broad area. Although a number of routing protocols for sensor networks are covered, the paper does not make a classification for such routing protocols and the list of discussed protocols is not meant to be complete given the scope of the survey. Our survey is more focused and can serve those who like deeper insight for routing issues and techniques in wireless sensor networks. To the best of our knowledge, our paper is the first work to make a classification of routing protocols in sensor networks. Moreover, our work reflects the current state of art in routing research by including a comprehensive list of recently proposed routing protocols.

Taxonomy of the different architectural attributes of sensor networks is developed in [13]. This work gives a high-level description of what is considered a typical sensor network architecture

along with its components. Sensor networks are classified by considering several architectural factors such as network dynamics and the data delivery model. Such classification is helpful for a designer to select the appropriate infrastructure for his/her application. However, the paper neither describes any routing protocol nor talks about the potential effects of infrastructure design on route setup. Our work is a dedicated study of the network layer, describing and categorizing the different approaches for data routing. In addition, we summarize different architectural design issues that may affect the performance of routing protocols.

2. Data-centric protocols

In many applications of sensor networks, it is not feasible to assign global identifiers to each node due to the sheer number of nodes deployed. Such lack of global identification along with random deployment of sensor nodes make it hard to select a specific set of sensor nodes to be queried. Therefore, data is usually transmitted from every sensor node within the deployment region with significant redundancy. Since this is very inefficient in terms of energy consumption, routing protocols that will be able to select a set of sensor nodes and utilize data aggregation during the relaying of data have been considered. This consideration has led to data-centric routing, which is different from traditional address-based routing where routes are created between addressable nodes managed in the network layer of the communication stack.

In data-centric routing, the sink sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data. SPIN [25] is the first data-centric protocol, which considers data negotiation between nodes in order to eliminate redundant data and save energy. Later, Directed Diffusion [18] has been developed and has become a breakthrough in data-centric routing. Then, many other protocols have been proposed either based on Directed Diffusion [26–28] or following a similar concept [16,24,29,30]. In this

section, we will describe these protocols in details and highlight the key ideas.

2.1. Flooding and gossiping

Flooding and gossiping [31] are two classical mechanisms to relay data in sensor networks without the need for any routing algorithms and topology maintenance. In flooding, each sensor receiving a data packet broadcasts it to all of its neighbors and this process continues until the packet arrives at the destination or the maximum number of hops for the packet is reached. On the other hand, gossiping is a slightly enhanced version of flooding where the receiving node sends the packet to a randomly selected neighbor, which picks another random neighbor to forward the packet to and so on.

Although flooding is very easy to implement, it has several drawbacks, see Figs. 1 and 2 redrawn from [25]. Such drawbacks include *implosion* caused by duplicated messages sent to same node, *overlap* when two nodes sensing the same region send similar packets to the same neighbor and *resource blindness* by consuming large amount of energy without consideration for the energy constraints [25]. Gossiping avoids the problem of implosion by just selecting a random node to send the packet rather than broadcasting. However, this cause delays in propagation of data through the nodes.
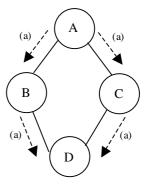

Fig. 1. The implosion problem. Node A starts by flooding its data to all of its neighbors. D gets two same copies of data eventually, which is not necessary.
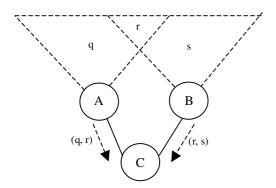
Fig. 2. The overlap problem. Two sensors cover an overlapping geographic region and C gets same copy of data form these sensors.

2.2. Sensor protocols for information via negotiation

SPIN [25] is among the early work to pursue a data-centric routing mechanism. The idea behind SPIN is to name the data using high-level descriptors or meta-data. Before transmission, meta-data are exchanged among sensors via a data advertisement mechanism, which is the key feature of SPIN. Each node upon receiving new data, advertises it to its neighbors and interested neighbors, i.e. those who do not have the data, retrieve the data by sending a request message. SPIN's meta-data negotiation solves the classic problems of flooding such as redundant information passing, overlapping of sensing areas and resource blindness thus, achieving a lot of energy efficiency. There is no standard meta-data format and it is assumed to be application specific, e.g. using an application level framing. There are three messages defined in SPIN to exchange data between nodes. These are: ADV message to allow a sensor to advertise a particular meta-data, REQ message to request the specific data and DATA message that carry the actual data. Fig. 3, redrawn from [25], summarizes the steps of the SPIN protocol.

One of the advantages of SPIN is that topological changes are localized since each node needs to know only its single-hop neighbors. SPIN gives a factor of 3.5 less than flooding in terms of energy dissipation and meta-data negotiation almost halves the redundant data. However, SPIN's data advertisement mechanism cannot guarantee the delivery of data. For instance, if the nodes that are interested in the data are far away from the source node and the nodes between source and destination are not interested in that data, such data will not be delivered to the destination at all. Therefore, SPIN is not a good choice for applications
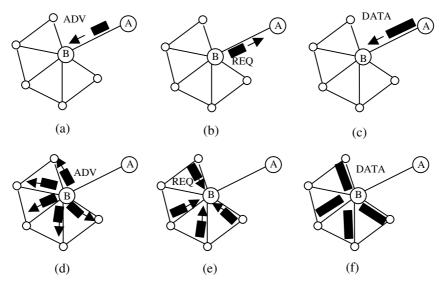

Fig. 3. SPIN protocol. Node A starts by advertising its data to node B (a). Node B responds by sending a request to node A (b). After receiving the requested data (c), node B then sends out advertisements to its neighbors (d), who in turn send requests back to B (e–f).
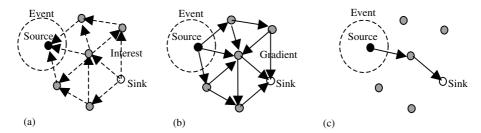
Fig. 4. Directed Diffusion protocol phases. (a) Interest propagation, (b) initial gradients setup, (c) data delivery along reinforced.

such as intrusion detection, which require reliable delivery of data packets over regular intervals.

2.3. Directed Diffusion

Directed Diffusion [18,19] is an important milestone in the data-centric routing research of sensor networks. The idea aims at diffusing data through sensor nodes by using a naming scheme for the data. The main reason behind using such a scheme is to get rid of unnecessary operations of network layer routing in order to save energy. Direct Diffusion suggests the use of attribute-value pairs for the data and queries the sensors in an on demand basis by using those pairs. In order to create a query, an interest is defined using a list of attribute-value pairs such as name of objects, interval, duration, geographical area, etc. The interest is broadcast by a sink through its neighbors. Each node receiving the interest can do caching for later use. The nodes also have the ability to do in-network data aggregation, which is modeled as a minimum Steiner tree problem [23]. The interests in the caches are then used to compare the received data with the values in the interests. The interest entry also contains several gradient fields. A gradient is a reply link to a neighbor from which the interest was received. It is characterized by the data rate, duration and expiration time derived from the received interest's fields. Hence, by utilizing interest and gradients, paths are established between sink and sources. Several paths can be established so that one of them is selected by reinforcement. The sink resends the original interest message through the selected path with a smaller interval hence reinforces the source node on that path to send data more fre-

quently. Fig. 4, redrawn from [18], summarizes the Directed Diffusion protocol.

Path repairs are also possible in Directed Diffusion. When a path between a source and the sink fails, a new or alternative path should be identified. For this, Directed Diffusion basically reinitiates reinforcement by searching among other paths, which are sending data in lower rates. Ganesan et al. [32] suggest employing multiple paths in advance so that in case of a failure of a path, one of the alternative paths is chosen without any cost for searching for another one. There is of course extra overhead of keeping these alternative paths alive by using low data rate, which will definitely use extra energy but more energy can be saved when a path fails and a new path should be chosen.

Directed Diffusion differs from SPIN in terms of the on demand data querying mechanism it has. In Directed Diffusion the sink queries the sensor nodes if a specific data is available by flooding some tasks. In SPIN, sensors advertise the availability of data allowing interested nodes to query that data. Directed Diffusion has many advantages. Since it is data centric, all communication is neighbor-to-neighbor with no need for a node addressing mechanism. Each node can do aggregation and caching, in addition to sensing. Caching is a big advantage in terms of energy efficiency and delay. In addition, Direct Diffusion is highly energy efficient since it is on demand and there is no need for maintaining global network topology.

However, Directed Diffusion cannot be applied to all sensor network applications since it is based on a query-driven data delivery model. The applications that require continuous data delivery to the sink will not work efficiently with a

query-driven on demand data model. Therefore, Directed Diffusion is not a good choice as a routing protocol for the applications such as environmental monitoring. In addition, the naming schemes used in Directed Diffusion are application dependent and each time should be defined a priori. Moreover, the matching process for data and queries might require some extra overhead at the sensors.

2.4. Energy-aware routing

Shah and Rabaey [29] proposed to use a set of sub-optimal paths occasionally to increase the lifetime of the network. These paths are chosen by means of a probability function, which depends on the energy consumption of each path. Network survivability is the main metric that the approach is concerned with. The approach argues that using the minimum energy path all the time will deplete the energy of nodes on that path. Instead, one of the multiple paths is used with a certain probability so that the whole network lifetime increases. The protocol assumes that each node is addressable through a class-based addressing which includes the location and types of the nodes. There are 3 phases in the protocol:

1. Setup phase: Localized flooding occurs to find the routes and create the routing tables. While doing this, the total energy cost is calculated in each node. For instance, if the request is sent from node N_i to node N_j, N_j calculates the cost of the path as follows:

$$C_{N_j,N_i} = \text{Cost}(N_i) + \text{Metric}(N_j, N_i).$$

Here, the energy metric used captures transmission and reception costs along with the residual energy of the nodes. Paths that have a very high cost are discarded. The node selection is done according to closeness to the destination. The node assigns a probability to each of its neighbors in routing (forwarding) table (FT) corresponding to the formed paths. The probability is inversely proportional to the cost, that is:

$$P_{N_j,N_i} = \frac{1/C_{N_j,N_i}}{\sum_{k\in\text{FT}_j} 1/C_{N_j,N_k}};$$

N_j then calculates the average cost for reaching the destination using the neighbors in the forwarding fable (FT_j) using the formula:

$$\text{Cost}(N_j) = \sum_{i\in\text{FT}_j} P_{N_j,N_i} C_{N_j,N_i}.$$

This average cost for N_j is set in the cost field of the request and forwarded.

2. Data communication phase: Each node forwards the packet by randomly choosing a node from its forwarding table using the probabilities.
3. Route maintenance phase: Localized flooding is performed infrequently to keep all the paths alive.

The described approach is similar to Directed Diffusion in the way potential paths from data sources to the sink are discovered. In Directed Diffusion, data is sent through multiple paths, one of them being reinforced to send at higher rates. On the other hand, Shah and Rabaey select a single path randomly from the multiple alternatives in order to save energy. Therefore, when compared to Directed Diffusion, it provides an overall improvement of 21.5% energy saving and a 44% increase in network lifetime. However, such single path usage hinders the ability of recovering from a node or path failure as opposed to Directed Diffusion. In addition, the approach requires gathering the location information and setting up the addressing mechanism for the nodes, which complicate route setup compared to the Directed Diffusion.

2.5. Rumor routing

Rumor routing [26] is another variation of Directed Diffusion and is mainly intended for contexts in which geographic routing criteria are not applicable. Generally Directed Diffusion floods the query to the entire network when there is no geographic criterion to diffuse tasks. However, in some cases there is only a little amount of data requested from the nodes and thus the use of flooding is unnecessary. An alternative approach is to flood the events if number of events is small and number of queries is large. Rumor routing is be-

tween event flooding and query flooding. The idea is to route the queries to the nodes that have observed a particular event rather than flooding the entire network to retrieve information about the occurring events.

In order to flood events through the network, the rumor routing algorithm employs long-lived packets, called agents. When a node detects an event, it adds such event to its local table and generates an agent. Agents travel the network in order to propagate information about local events to distant nodes. When a node generates a query for an event, the nodes that know the route, can respond to the query by referring its event table. Hence, the cost of flooding the whole network is avoided. Rumor routing maintains only one path between source and destination as opposed to Directed Diffusion where data can be sent through multiple paths at low rates.

Simulation results have shown that rumor routing achieves significant energy saving over event flooding and can also handle node's failure. However, rumor routing performs well only when the number of events is small. For large number of events, the cost of maintaining agents and event-tables in each node may not be amortized if there is not enough interest on those events from the sink. Another issue to deal with is tuning the overhead through adjusting parameters used in the algorithm such as time-to-live for queries and agents.

2.6. Gradient-based routing

Schurgers et al. [27] have proposed a slightly changed version of Directed Diffusion, called Gradient-based routing (GBR). The idea is to keep the number of hops when the interest is diffused through the network. Hence, each node can discover the minimum number of hops to the sink, which is called height of the node. The difference between a node's height and that of its neighbor is considered the gradient on that link. A packet is forwarded on a link with the largest gradient.

The authors aim at using some auxiliary techniques such as data aggregation and traffic spreading along with GBR in order to balance the traffic uniformly over the network. Nodes acting

as a relay for multiple paths can create a data combining entity in order to aggregate data. On the other hand, three different data spreading techniques have been presented:

- Stochastic scheme: When there are two or more next hops with the same gradient, the node chooses one of them at random.
- Energy-based scheme: When a node's energy drops below a certain threshold, it increases its height so that other sensors are discouraged from sending data to that node.
- Stream-based scheme: The idea is to divert new streams away from nodes that are currently part of the path of other streams.

The data spreading schemes strives to achieve an even distribution of the traffic throughout the whole network, which helps in balancing the load on sensor nodes and increases the network lifetime. The employed techniques for traffic load balancing and data fusion are also applicable to other routing protocols for enhanced performance. Through simulation GBR has been shown to outperform Directed Diffusion in terms of total communication energy.

2.7. CADR

Constrained anisotropic diffusion routing (CADR) [28] is a protocol, which strives to be a general form of Directed Diffusion. Two techniques namely information-driven sensor querying (IDSQ) and constrained anisotropic diffusion routing are proposed. The idea is to query sensors and route data in a network in order to maximize the information gain, while minimizing the latency and bandwidth. This is achieved by activating only the sensors that are close to a particular event and dynamically adjusting data routes. The major difference from Directed Diffusion is the consideration of information gain in addition to the communication cost. In CADR, each node evaluates an information/cost objective and routes data based on the local information/cost gradient and end-user requirements. The information utility measure is modeled using standard estimation theory.

IDSQ is based on a protocol in which the querying node can determine which node can provide the most useful information while balancing the energy cost. While IDSQ provides a way of selecting the optimal order of sensors for maximum incremental information gain, it does not specifically define how the query and the information are routed between sensors and the sink. Therefore, IDSQ can be seen as a complementary optimization procedure.

Since CADR diffuses queries by using a set of information criteria to select which sensors to get the data, simulation results confirmed that it is more energy efficient than Directed Diffusion where queries are diffused in an isotropic fashion, reaching nearest neighbors first.

2.8. COUGAR

A data-centric protocol that views the network as a huge distributed database system is proposed in [24]. The main idea is to use declarative queries in order to abstract query processing from the network layer functions such as selection of relevant sensors etc. and utilize in-network data aggregation to save energy. The abstraction is supported through a new query layer between the network and application layers.

COUGAR proposes an architecture for the sensor database system where sensor nodes select a leader node to perform aggregation and transmit the data to the gateway (sink). The architecture is depicted in Fig. 5, which is redrawn from [24]. The gateway is responsible for generating a query plan, which specifies the necessary information about the data flow and in-network computation for the incoming query and send it to the relevant nodes. The query plan also describes how to select a leader for the query. The architecture provides in-network computation ability for all the sensor nodes. Such ability ensures energy-efficiency especially when the number of sensors generating and sending data to the leader is huge.

Although COUGAR provides a network-layer independent solution for querying the sensors, it has some drawbacks: First of all, introducing additional query layer on each sensor node will bring extra overhead to sensor nodes in terms of
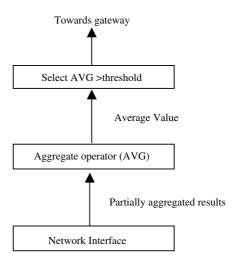

Fig. 5. Query plan at a leader node: the leader node gets all the readings, calculates the average and if it is greater than a threshold sends it to the gateway (sink).

energy consumption and storage. Second, in-network data computation from several nodes will require synchronization, i.e. a relaying node should wait every packet from each incoming source, before sending the data to the leader node. Third, the leader nodes should be dynamically maintained to prevent them from failure.

2.9. ACQUIRE

A fairly new data-centric mechanism for querying sensor networks is ACtive QUery forwarding In sensoR nEtworks (ACQUIRE) [30]. As in [24], the approach views the sensor network as a distributed database and is well-suited for complex queries which consist of several sub queries. The querying mechanism works as follows: the query is forwarded by the sink and each node receiving the query, tries to respond partially by using its pre-cached information and forward it to another sensor. If the pre-cached information is not up-to-date, the nodes gather information from its neighbors within a look-ahead of d hops. Once the query is being resolved completely, it is sent back through either the reverse or shortest-path to the sink.

One of the main motivations for proposing ACQUIRE is to deal with one-shot, complex

queries for data where a response can be provided by many nodes. Since, the data-centric approaches such as Directed Diffusion uses flooding-based query mechanism for continuous and aggregate queries; it would not make sense to use the same mechanism for one-shot complex queries due to energy considerations. ACQUIRE mechanism provides efficient querying by adjusting the value of parameter d. Note that if d is equal to network size, then the protocol behaves similar to flooding. On the other hand, the query has to travel more hops if d is too small.

A mathematical modeling has been derived for the energy cost of the ACQUIRE approach and been compared to both flooding and ring search, i.e. gradual increase in number of hops. An optimal value of parameter d is calculated for a grid of sensors where each node has four immediate neighbors. However, there is no validation of results through simulation and the reception costs have not taken into account during calculations.

The problem of selecting the next node for forwarding the query, which ACQUIRE addresses, has been studied in CADR [28] and rumor routing [26]. In CADR, as described earlier, the querying nodes uses IDSQ mechanism to determine which node can provide most useful information by using estimation theory. Rumor routing [26] tries to forward query to a node, which knows the path to the searched event. Since the nodes become aware of events through the event agents, the heuristic for defining the route of an event agent highly affects the performance of next hop selection. In ACQUIRE, the next node to forward the query is either picked randomly or the selection is based on maximum potential of query satisfaction [30].

3. Hierarchical protocols

Similar to other communication networks, scalability is one of the major design attributes of sensor networks. A single-tier network can cause the gateway to overload with the increase in sensors density. Such overload might cause latency in communication and inadequate tracking of events. In addition, the single-gateway architecture is not scalable for a larger set of sensors covering a wider area of interest since the sensors are typically not capable of long-haul communication. To allow the system to cope with additional load and to be able to cover a large area of interest without degrading the service, networking clustering has been pursued in some routing approaches.

The main aim of hierarchical routing is to efficiently maintain the energy consumption of sensor nodes by involving them in multi-hop communication within a particular cluster and by performing data aggregation and fusion in order to decrease the number of transmitted messages to the sink. Cluster formation is typically based on the energy reserve of sensors and sensor's proximity to the cluster head [33,34]. LEACH [14] is one of the first hierarchical routing approaches for sensors networks. The idea proposed in LEACH has been an inspiration for many hierarchical routing protocols [16,20,21,35], although some protocols have been independently developed [11,15]. We explore hierarchical routing protocols in this section.

3.1. LEACH

Low-energy adaptive clustering hierarchy (LEACH) [14] is one of the most popular hierarchical routing algorithms for sensor networks. The idea is to form clusters of the sensor nodes based on the received signal strength and use local cluster heads as routers to the sink. This will save energy since the transmissions will only be done by such cluster heads rather than all sensor nodes. Optimal number of cluster heads is estimated to be 5% of the total number of nodes.

All the data processing such as data fusion and aggregation are local to the cluster. Cluster heads change randomly over time in order to balance the energy dissipation of nodes. This decision is made by the node choosing a random number between 0 and 1. The node becomes a cluster head for the current round if the number is less than the following threshold:

$$T(n) = \begin{cases} \dfrac{p}{1 - p * (r \bmod 1/p)} & \text{if } n \in G, \\ 0 & \text{otherwise,} \end{cases}$$

where p is the desired percentage of cluster heads (e.g. 0.05), r is the current round, and G is the set of nodes that have not been cluster heads in the last $1/p$ rounds.

LEACH achieves over a factor of 7 reduction in energy dissipation compared to direct communication and a factor of 4–8 compared to the minimum transmission energy routing protocol. The nodes die randomly and dynamic clustering increases lifetime of the system. LEACH is completely distributed and requires no global knowledge of network. However, LEACH uses single-hop routing where each node can transmit directly to the cluster-head and the sink. Therefore, it is not applicable to networks deployed in large regions. Furthermore, the idea of dynamic clustering brings extra overhead, e.g. head changes, advertisements etc., which may diminish the gain in energy consumption.

3.2. PEGASIS and Hierarchical-PEGASIS

Power-efficient GAthering in Sensor Information Systems (PEGASIS) [20] is an improvement of the LEACH protocol. Rather than forming multiple clusters, PEGASIS forms chains from sensor nodes so that each node transmits and receives from a neighbor and only one node is selected from that chain to transmit to the base station (sink). Gathered data moves from node to node, aggregated and eventually sent to the base station. The chain construction is performed in a greedy way. As shown in Fig. 6 node $c0$ passes its data to node $c1$. Node $c1$ aggregates node $c0$'s data with its own and then transmits to the leader. After node $c2$ passes the token to node $c4$, node $c4$ transmits its data to node $c3$. Node $c3$ aggregates node $c4$'s data with its own and then transmits to the leader. Node $c2$ waits to receive data from both neighbors and then aggregates its data with its neighbors' data. Finally, node $c2$ transmits one message to the base station.
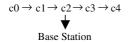
The difference from LEACH is to use multi-hop routing by forming chains and selecting only one node to transmit to the base station instead of using multiple nodes. PEGASIS has been shown to outperform LEACH by about 100–300% for different network sizes and topologies. Such performance gain is achieved through the elimination of the overhead caused by dynamic cluster formation in LEACH and through decreasing the number of transmissions and reception by using data aggregation. However, PEGASIS introduces excessive delay for distant node on the chain. In addition the single leader can become a bottleneck.

Hierarchical-PEGASIS [21] is an extension to PEGASIS, which aims at decreasing the delay incurred for packets during transmission to the base station and proposes a solution to the data gathering problem by considering energy×delay metric. In order to reduce the delay in PEGASIS, simultaneous transmissions of data messages are pursued. To avoid collisions and possible signal interference among the sensors, two approaches have been investigated. The first approach incorporates signal coding, e.g. CDMA. In the second approach only spatially separated nodes are allowed to transmit at the same time.

The chain-based protocol with CDMA capable nodes, constructs a chain of nodes, that forms a tree like hierarchy, and each selected node in a particular level transmits data to the node in the upper level of the hierarchy. This method ensures data transmitting in parallel and reduces the delay significantly. Since the tree is balanced, the delay will be in O(lg N) where N is the number of nodes. For instance, in Fig. 7 redrawn from [21], node $c3$

$$c0 \rightarrow c1 \rightarrow c2 \rightarrow c3 \rightarrow c4$$
$$\downarrow$$
Base Station

Fig. 6. Chaining in PEGASIS.

BS
$$\uparrow$$
c3

$$c3 \leftarrow c7$$

$$c1 \rightarrow c3 \rightarrow c5 \rightarrow c7$$

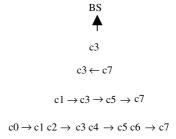$$c0 \rightarrow c1 \; c2 \rightarrow c3 \; c4 \rightarrow c5 \; c6 \rightarrow c7$$

Fig. 7. Data gathering in a chain based binary scheme.

is the designated leader for round 3. Since, node c3 is in position 3 (counting from 0) on the chain, all nodes in an even position will send to their right neighbor. Nodes that are receiving at each level rise to next level in the hierarchy. Now at the next level, node c3 is still in an odd position (1). Again all nodes in an even position will aggregate its data with its received data and send to their right. At the third level, node c3 is not in an odd position, so node c7 will aggregate its data and transmit to c3. Finally, node c3 will combine its current data with that received from c7 and transmit the message to the sink.

The non-CDMA based approach creates a three-level hierarchy of the nodes and interference effects is reduced by carefully scheduling simultaneous transmissions. Such chain-based protocol has been shown to perform better than the regular PEGASIS scheme by a factor of about 60.

Although the PEGASIS approaches avoid the clustering overhead of LEACH, they still require dynamic topology adjustment since sensor's energy is not tracked. For example, every sensor needs to be aware of the status of its neighbor so that it knows where to route that data. Such topology adjustment can introduce significant overhead especially for highly utilized networks.

3.3. TEEN and APTEEN

Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [16] is a hierarchical protocol designed to be responsive to sudden changes in the sensed attributes such as temperature. Responsiveness is important for time-critical applications, in which the network operated in a reactive mode. TEEN pursues a hierarchical approach along with the use of a data-centric mechanism. The sensor network architecture is based on a hierarchical grouping where closer nodes form clusters and this process goes on the second level until base station (sink) is reached. The model is depicted in Fig. 8, which is redrawn from [16].

After the clusters are formed, the cluster head broadcasts two thresholds to the nodes. These are hard and soft thresholds for sensed attributes. Hard threshold is the minimum possible value of
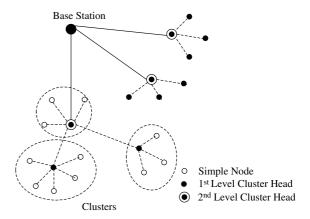

Fig. 8. Hierarchical clustering in TEEN and APTEEN.

an attribute to trigger a sensor node to switch on its transmitter and transmit to the cluster head. Thus, the hard threshold allows the nodes to transmit only when the sensed attribute is in the range of interest, thus reducing the number of transmissions significantly. Once a node senses a value at or beyond the hard threshold, it transmits data only when the value of that attribute changes by an amount equal to or greater than the soft threshold. As a consequence, soft threshold will further reduce the number of transmissions if there is little or no change in the value of sensed attribute. One can adjust both hard and soft threshold values in order to control the number of packet transmissions. However, TEEN is not good for applications where periodic reports are needed since the user may not get any data at all if the thresholds are not reached.

The Adaptive Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) [35] is an extension to TEEN and aims at both capturing periodic data collections and reacting to time-critical events. The architecture is same as in TEEN. When the base station forms the clusters, the cluster heads broadcast the attributes, the threshold values, and the transmission schedule to all nodes. Cluster heads also perform data aggregation in order to save energy. APTEEN supports three different query types: historical, to analyze past data values; one-time, to take a snapshot view of the network; and persistent to monitor an event for a period of time.

Simulation of TEEN and APTEEN has shown them to outperform LEACH [14]. The experiments have demonstrated that APTEEN's performance is between LEACH and TEEN in terms of energy dissipation and network lifetime. TEEN gives the best performance since it decreases the number of transmissions. The main drawbacks of the two approaches are the overhead and complexity of forming clusters in multiple levels, implementing threshold-based functions and dealing with attribute-based naming of queries.

3.4. Energy-aware routing for cluster-based sensor networks

Younis et al. [15] have proposed a different hierarchical routing algorithm based on a three-tier architecture. Sensors are grouped into clusters prior to network operation. The algorithm employs cluster heads, namely gateways, which are less energy constrained than sensors and assumed to know the location of sensor nodes. Gateways maintain the states of the sensors and sets up multi-hop routes for collecting sensors' data. A TDMA based MAC is used for nodes to send data to the gateway. The gateway informs each node about slots in which it should listen to other nodes' transmission and slots, which the node can use for its own transmission. The command node (sink) communicates only with the gateways.

The sensor is assumed to be capable of operating in an active mode or a low-power stand-by mode. The sensing and processing circuits can be powered on and off. In addition both the radio transmitter and receiver can be independently turned on and off and the transmission power can be programmed based on the required range. The sensor nodes in a cluster can be in one of four main states: sensing only, relaying only, sensing-relaying, and inactive. In the *sensing state*, the node probes the environment and generates data at a constant rate. In the *relaying state*, the node does not sense the target but its communications circuitry is on to relay the data from other active nodes. When a node is both sensing and relaying messages from other nodes, it is considered in the *sensing-relaying state*. Otherwise, the node is con-
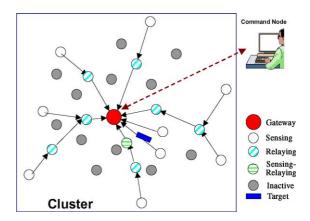

Fig. 9. A typical cluster in a sensor network.

sidered *inactive* and can turn off its sensing and communication circuitry.

Fig. 9, redrawn from [15], shows an example of the state of sensors and routes within a typical cluster for a target-tracking application. A cost function is defined between any two nodes in terms of energy consumption, delay optimization and other performance metrics. Using this cost function as the link cost, a least-cost path is found between sensor nodes and the gateway. The gateway will continuously monitor the available energy level at every sensor that is active in data processing, sensing, or in forwarding data packets, relaying. Rerouting is triggered by an application-related event requiring different set of sensors to probe the environment or the depletion of the battery of an active node.

A variant of this routing approach has been proposed in [36]. The algorithm constrains the minimum transmission range in order to limit the delay. Simulation results have demonstrated that such approach consistently performs well with respect to both energy-based metrics, e.g. network lifetime, as well as contemporary metrics, e.g. throughput and end-to-end delay. The results also have indicated that combining the routing approach with the time-based medium arbitration can further increase the life of the network by an order of magnitude. However, such approach assumes simple propagation model, which might require the deployment of many gateways to ensure high sensor coverage. The approach is further

extended in [37] to overcome ambiguity in signal propagation by introducing an additional tier to the network. Basically nodes that are not reachable are assigned an agent sensor to convey commands from the gateway and pass nodes status back to the gateway.

3.5. Self-organizing protocol

Subramanian and Katz [11] not only describe a self-organizing protocol but develop taxonomy of sensor applications as well. Based on such taxonomy, they have proposed architectural and infrastructural components necessary for building sensor applications. The architecture supports heterogeneous sensors that can be mobile or stationary. Some sensors, which can be either stationary or mobile, probe the environment and forward the data to designated set of nodes that act as routers. Router nodes are stationary and form the backbone for communication. Collected data are forwarded through the routers to more powerful sink nodes. Each sensing node should be reachable to a router node in order to be part of the network.

A routing architecture that requires addressing of each sensor node has been proposed. Sensing nodes are identifiable through the address of the router node it is connected to. The routing architecture is hierarchical where groups of nodes are formed and merge when needed. In order to support fault tolerance, local Markov loops (LML) algorithm, which performs a random walk on spanning trees of a graph, is used in broadcasting. The algorithm for self-organizing the router nodes and creating the routing tables consists of four phases:

- Discovery phase: The nodes in the neighborhood of each sensor are discovered.
- Organization phase: Groups are formed and merged by forming a hierarchy. Each node is allocated an address based on its position in the hierarchy. Routing tables of size $O(\log N)$ are created for each node. Broadcast trees that span all the nodes are constructed.
- Maintenance phase: Updating of routing tables and energy levels of nodes is made in this phase.

Each node informs the neighbors about its routing table and energy level. LML are used to maintain broadcast trees.
- Self-reorganization phase: In case of partition or node failures, group reorganizations are performed.

The proposed algorithm utilizes the router nodes to keep all the sensors connected by forming a dominating set. Such approach is similar to the idea of virtual grid used in GAF [38], which will be discussed later under location-based protocols. Both approaches achieve energy saving through utilization of a limited subset of nodes. Since sensor nodes can be addressed individually in the routing architecture, the proposed algorithm is suitable for applications such as parking-lot networks where communication to a particular node is required [11]. The major advantage of using the algorithm is the small cost of maintaining routing tables and keeping routing hierarchy being strictly balanced. Moreover, the energy consumed for broadcasting a message is less than that consumed in SPIN protocol [25] due to the broadcast trees utilized in the algorithm. Fault tolerance is also achieved by using LML in broadcast trees.

The disadvantage is in the organization phase of algorithm, which is not on-demand, therefore introducing extra overhead. Another possible problem is in case of hierarchy forming when there are many cuts in the network. This will be expensive since network-cuts increase the probability of applying reorganization phase.

4. Location-based protocols

Most of the routing protocols for sensor networks require location information for sensor nodes. In most cases location information is needed in order to calculate the distance between two particular nodes so that energy consumption can be estimated. Since, there is no addressing scheme for sensor networks like IP-addresses and they are spatially deployed on a region, location information can be utilized in routing data in an energy efficient way. For instance, if the region to be sensed is known, using the location of sensors, the

query can be diffused only to that particular region which will eliminate the number of transmission significantly. Some of the protocols discussed here are designed primarily for mobile ad hoc networks and consider the mobility of nodes during the design [38–40]. However, they are also well applicable to sensor networks where there is less or no mobility.

It is worth noting that there are other location-based protocols designed for wireless ad hoc networks, such as Cartesian and trajectory-based routing [41,42]. However, many of these protocols are not applicable to sensor networks since they are not energy aware. In order to stay with the theme of the survey, we limit the scope of coverage to only energy-aware location-based protocols.

4.1. MECN and SMECN

Minimum energy communication network (MECN) [39] sets up and maintains a minimum energy network for wireless networks by utilizing low power GPS. Although, the protocol assumes a mobile network, it is best applicable to sensor networks, which are not mobile. A minimum power topology for stationary nodes including a master node is found. MECN assumes a master-site as the information sink, which is always the case for sensor networks.

MECN identifies a *relay region* for every node. The relay region consists of nodes in a surrounding area where transmitting through those nodes is more energy efficient than direct transmission. The relay region for node pair (i, r) is depicted in Fig. 10, redrawn from [39]. The *enclosure* of a node i is then created by taking the union of all relay regions that node i can reach. The main idea of MECN is to find a sub-network, which will have less number of nodes and require less power for transmission between any two particular nodes. In this way, global minimum power paths are found without considering all the nodes in the network. This is performed using a localized search for each node considering its relay region. The protocol has two phases:

1. It takes the positions of a two-dimensional plane and constructs a sparse graph (enclosure
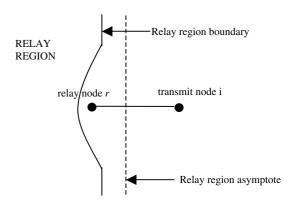

Fig. 10. Relay region of transmit-relay node pair (i, r) in MECN.

graph), which consists of all the enclosures of each transmit node in the graph. This construction requires local computations in the nodes. The enclose graph contains globally optimal links in terms of energy consumption.

2. Finds optimal links on the enclosure graph. It uses distributed Belmann–Ford shortest path algorithm with power consumption as the cost metric. In case of mobility the position coordinates are updated using GPS.

MECN is self-reconfiguring and thus can dynamically adapt to node's failure or the deployment of new sensors. Between two successive wake-ups of the nodes, each node can execute the first phase of the algorithm and the minimum cost links are updated by considering leaving or newly joining nodes.

The small minimum energy communication network (SMECN) [40] is an extension to MECN. In MECN, it is assumed that every node can transmit to every other node, which is not possible every time. In SMECN possible obstacles between any pair of nodes are considered. However, the network is still assumed to be fully connected as in the case of MECN. The sub-network constructed by SMECN for minimum energy relaying is provably smaller (in terms of number of edges) than the one constructed in MECN if broadcasts are able to reach to all nodes in a circular region around the broadcaster. As a result, the number of hops for transmissions will decrease. Simulation results show that SMECN uses less energy than

MECN and maintenance cost of the links is less. However, finding a sub-network with smaller number of edges introduces more overhead in the algorithm.

4.2. GAF

Geographic adaptive fidelity (GAF) [38] is an energy-aware location-based routing algorithm designed primarily for mobile ad hoc networks, but may be applicable to sensor networks as well. GAF conserves energy by turning off unnecessary nodes in the network without affecting the level of routing fidelity. It forms a virtual grid for the covered area. Each node uses its GPS-indicated location to associate itself with a point in the virtual grid. Nodes associated with the same point on the grid are considered equivalent in terms of the cost of packet routing. Such equivalence is exploited in keeping some nodes located in a particular grid area in sleeping state in order to save energy. Thus, GAF can substantially increase the network lifetime as the number of nodes increases. A sample situation is depicted in Fig. 11, which is redrawn from [38]. In this figure, node 1 can reach any of 2, 3 and 4 and nodes 2, 3, and 4 can reach 5. Therefore nodes 2, 3 and 4 are equivalent and two of them can sleep.

Nodes change states from sleeping to active in turn so that the load is balanced. There are three states defined in GAF. These states are *discovery*, for determining the neighbors in the grid, *active* reflecting participation in routing and *sleep* when the radio is turned off. The state transitions in GAF are depicted in Fig. 12. Which node will
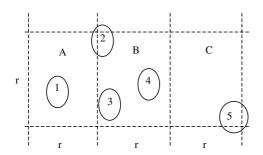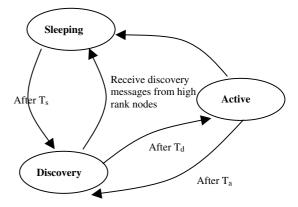

Fig. 12. State transitions in GAF.

sleep for how long is application dependent and the related parameters are tuned accordingly during the routing process. In order to handle the mobility, each node in the grid estimates its leaving time of grid and sends this to its neighbors. The sleeping neighbors adjust their sleeping time accordingly in order to keep the routing fidelity. Before the leaving time of the active node expires, sleeping nodes wake up and one of them becomes active. GAF is implemented both for non-mobility (GAF-basic) and mobility (GAF-mobility adaptation) of nodes.

GAF strives to keep the network connected as in [11,39], by keeping a representative node always in active mode for each region on its virtual grid. While such connectivity is ensured by self-organizing the router sensors in [11], MECN maintains an enclosure graph of the network by dynamically changing the transmitting range assignment of the nodes [39]. Simulation results show that GAF performs at least as well as a normal ad hoc routing protocol in terms of latency and packet loss and increases the lifetime of the network by saving energy. Although GAF is a location-based protocol, it may also be considered as a hierarchical protocol, where the clusters are based on geographic location. For each particular grid area, a representative node acts as the leader to transmit the data to other nodes. The leader node however, does not do any aggregation or fusion as in the case of other hierarchical protocols discussed in this paper.

Fig. 11. Example of virtual grid in GAF.

4.3. GEAR

Yu et al. [43] have suggested the use of geographic information while disseminating queries to appropriate regions since data queries often includes geographic attributes. The protocol, namely geographic and energy-aware routing (GEAR), uses energy aware and geographically informed neighbor selection heuristics to route a packet towards the target region. The idea is to restrict the number of interests in Directed Diffusion by only considering a certain region rather than sending the interests to the whole network. GEAR compliments Directed Diffusion in this way and thus conserves more energy.

In GEAR, each node keeps an estimated cost and a learning cost of reaching the destination through its neighbors. The estimated cost is a combination of residual energy and distance to destination. The learned cost is a refinement of the estimated cost that accounts for routing around holes in the network. A hole occurs when a node does not have any closer neighbor to the target region than itself. If there are no holes, the estimated cost is equal to the learned cost. The learned cost is propagated one hop back every time a packet reaches the destination so that route setup for next packet will be adjusted. There are two phases in the algorithm:

1. Forwarding packets towards the target region: Upon receiving a packet, a node checks its neighbors to see if there is one neighbor, which is closer to the target region than itself. If there is more than one, the nearest neighbor to the target region is selected as the next hop. If they are all further than the node itself, this means there is a hole. In this case, one of the neighbors is picked to forward the packet based on the learning cost function. This choice can then be updated according to the convergence of the learned cost during the delivery of packets.
2. Forwarding the packets within the region: If the packet has reached the region, it can be diffused in that region by either recursive geographic forwarding or restricted flooding. Restricted flooding is good when the sensors are not densely deployed. In high-density networks, recur-
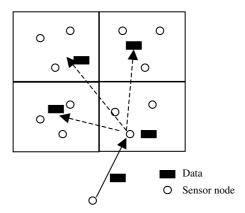

Fig. 13. Recursive geographic forwarding in GEAR.

sive geographic flooding is more energy efficient than restricted flooding. In that case, the region is divided into four sub regions and four copies of the packet are created. This splitting and forwarding process continues until the regions with only one node are left. An example is depicted in Fig. 13, which is redrawn from [43].

GEAR is compared to a similar non-energy-aware routing protocol GPSR [44], which is one of the earlier works in geographic routing that uses planar graphs to solve the problem of holes. In case of GPSR, the packets follow the perimeter of the planar graph to find their route. Although GPSR decrease the number of states a node should keep, it has been designed for general mobile ad hoc networks and requires a location service to map locations and node identifiers. GEAR not only reduces energy consumption for the route setup, but also performs better than GPSR in terms of packet delivery. The simulation results show that for an uneven traffic distribution, GEAR delivers 70–80% more packets than (GPSR) For uniform traffic pairs GEAR delivers 25–35% more packets than GPSR.

5. Network flow and QoS-aware protocols

Although most of the routing protocols proposed for sensor networks fit our classification, some pursue somewhat different approach such as

network flow and QoS. In some approaches, route setup is modeled and solved as a network flow problem. QoS-aware protocols consider end-to-end delay requirements while setting up the paths in the sensor network. We discuss sample of these protocols in this section.

5.1. Maximum lifetime energy routing

Chang and Tassiulas [45] presents an interesting solution to the problem of routing in sensor networks based on a network flow approach. The main objective of the approach is to maximize the network lifetime by carefully defining link cost as a function of node remaining energy and the required transmission energy using that link. Finding traffic distribution is a possible solution to the routing problem in sensor networks and based on that, comes the name "maximum lifetime energy routing". The solution to this problem maximizes the feasible time the network lasts. In order to find out the best link metric for the stated maximization problem, two maximum residual energy path algorithms are presented and simulated. The two algorithms differ in their definition of link costs and the incorporation of nodes' residual energy. Rather than using e_{ij}, the energy consumed when a packet is transmitted over link i–j, the following link costs are used:

$$c_{ij} = \frac{1}{\underline{E_i} - e_{ij}} \quad \text{and} \quad c_{ij} = \frac{e_{ij}}{\underline{E_i}}$$

where $\underline{E_i}$ is the residual energy at node i.

By using Bellman–Ford shortest path algorithm for the above link costs, the least cost paths to the destination (gateway) are found. The least cost path obtained is the path whose residual energy is largest among all the paths. The algorithms utilizing these link costs are compared to Minimum transmitted energy (MTE) algorithm, which considers e_{ij} as the link cost. Simulation results show that the proposed maximum residual energy path approach has better average lifetime than MTE for both link cost models. This is due to the absolute residual energy metric that MTE uses. The newly proposed metrics are concerned with relative

residual energy that reflect the forecasted energy consumption rate.

5.2. Maximum lifetime data gathering

Kalpakis et al. [46] models the data routes setup in sensor networks as the maximum lifetime data-gathering problem and presents a polynomial time algorithm. The lifetime "T" of the system is defined as the number of rounds or periodic data readings from sensors until the first sensor dies. The data-gathering schedule specifies for each round how to get and route data to the sink. A schedule has one tree for each round, which is directed from the sink and spans all the nodes in the system. The system lifetime depends on the duration for which the schedule remains valid. The aim is to maximize the lifetime of the schedule. An algorithm called maximum lifetime data aggregation (MLDA) is proposed. The algorithm considers data aggregation while setting up maximum lifetime routes. In this case, if a schedule "S" with "T" rounds is considered, it induces a flow network G. The flow network with maximum lifetime subject to the energy constraints of sensor nodes is called an optimal admissible flow network. Then, a schedule is constructed by using this admissible flow network. The paper presents an algorithm for generating such schedule.

A variant of the problem is also considered for the applications where data aggregation is not possible, i.e. steams from video sensors. In this case, the scenario is called maximum lifetime data routing (MLDR) and is modeled as a network flow problem with energy constraints on sensors. A solution with integer programming is presented.

Both MLDR and MLDA are implemented and compared to hierarchical-PEGASIS [21] in terms of system lifetime. The results show that both of them perform significantly better than hierarchical-PEGASIS. However, in MLDA the delay for a data packet is slightly greater than hierarchical-PEGASIS. While MLDA is performing better than the other protocols in terms of system lifetime, the algorithm is computationally expensive for very large sensor networks. Therefore, the authors propose another solution (CMLDA)

based on clustering to ensure scalability of the algorithm in [47].

5.3. Minimum cost forwarding

Minimum cost forwarding protocol [48] aims at finding the minimum cost path in a large sensor network, which will also be simple and scalable. The protocol is not really flow-based, however since data flows over the minimum cost path and the resources on the nodes are updated after each flow, we have included it in this section.

The cost function for the protocol captures the effect of delay, throughput and energy consumption from any node to the sink. There are two phases in the protocol. First phase is a setup phase for setting the cost value in all nodes. It starts from the sink and diffuses through the network. Every node adjusts its cost value by adding the cost of the node it received the message from and the cost of the link. Such cost adjustment is not done through flooding. Instead, a back-off based algorithm is used in order to limit the number of messages exchanged. The forwarding of message is deferred for a preset duration to allow the message with a minimum cost to arrive. Hence, the algorithm finds optimal cost of all nodes to the sink by using only one message at each node. Once these cost fields are set, there will be no need to keep next hop states for the nodes. This will ensure scalability.

In the second phase, the source broadcasts the data to its neighbors. The nodes receiving the broadcast message, adds its transmission cost (to sink) to the cost of the packet. Then the node checks the remaining cost in the packet. If the remaining cost of the packet is not sufficient to reach the sink, the packet is dropped. Otherwise the node forwards the packet to its neighbors. The protocol does not require any addresses and forwarding paths. Simulation results show that the cost values for each node obtained by the proposed protocol is same as flooding. As a consequence, optimal forwarding is achieved with minimum number of advertisement messages. The average number of advertisement messages in flooding could be reduced by a factor of 50 using the back off based algorithm with a proper setting of the back off timer.

5.4. SAR

Sequential assignment routing (SAR) is the first protocol for sensor networks that includes the notion of QoS in its routing decisions [1,2]. It is a table-driven multi-path approach striving to achieve energy efficiency and fault tolerance. The SAR protocol creates trees rooted at one-hop neighbors of the sink by taking QoS metric, energy resource on each path and priority level of each packet into consideration. By using created trees, multiple paths from sink to sensors are formed. One of these paths is selected according to the energy resources and QoS on the path. Failure recovery is done by enforcing routing table consistency between upstream and downstream nodes on each path. Any local failure causes an automatic path restoration procedure locally. Simulation results show that SAR offers less power consumption than the minimum-energy metric algorithm, which focuses only the energy consumption of each packet without considering its priority. SAR maintains multiple paths from nodes to sink. Although, this ensures fault-tolerance and easy recovery, the protocol suffers from the overhead of maintaining the tables and states at each sensor node especially when the number of nodes is huge.

5.5. Energy-aware QoS routing protocol

A fairly new QoS aware protocol for sensor networks is proposed by Akkaya and Younis [22]. Real-time traffic is generated by imaging sensors. The proposed protocol extends the routing approach in [15] and finds a least cost and energy efficient path that meets certain end-to-end delay during the connection. The link cost used is a function that captures the nodes' energy reserve, transmission energy, error rate and other communication parameters.

In order to support both best effort and real-time traffic at the same time, a class-based queuing model is employed. The queuing model allows
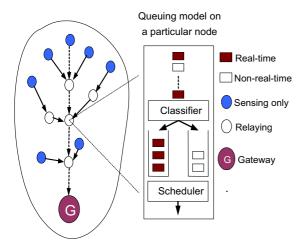
Fig. 14. Queuing model in a particular sensor node.

service sharing for real-time and non-real-time traffic. The bandwidth ratio r, is defined as an initial value set by the gateway and represents the amount of bandwidth to be dedicated both to the real-time and non-real-time traffic on a particular outgoing link in case of a congestion. As a consequence, the throughput for normal data does not diminish by properly adjusting such "r" value. The queuing model is depicted in Fig. 14, which is redrawn from [22]. The protocol finds a list of least cost paths by using an extended version of Dijkstra's algorithm and picks a path from that list which meets the end-to-end delay requirement.

Simulation results show that the proposed protocol consistently performs well with respect to QoS and energy metrics. However, the r-value is set initially same for all the nodes, which does not provide flexible adjusting of bandwidth sharing for different links. The protocol is extended in [49] by assigning a different r-value for each node in order to achieve a better utilization of the links.

5.6. SPEED

A QoS routing protocol for sensor networks that provides soft real-time end-to-end guarantees is described in [50]. The protocol requires each node to maintain information about its neighbors and uses geographic forwarding to find the paths. In addition, SPEED strive to ensure a certain speed for each packet in the network so that each application can estimate the end-to-end delay for the packets by dividing the distance to the sink by the speed of the packet before making the admission decision. Moreover, SPEED can provide congestion avoidance when the network is congested.

The routing module in SPEED is called stateless geographic non-deterministic forwarding (SNFG) and works with four other modules at the network layer, as shown in Fig. 15, redrawn from [50]. The beacon exchange mechanism collects information about the nodes and their location. Delay estimation at each node is basically made by calculating the elapsed time when an ACK is received from a neighbor as a response to a transmitted data packet. By looking at the delay values, SNGF selects the node, which meets the speed requirement. If such a node cannot be found, the *relay ratio* of the node is checked. The neighborhood feedback loop module is responsible for providing the relay ratio which is calculated by looking at the miss ratios of the neighbors of a node (the nodes which could not provide the desired speed) and is fed to the SNGF module. If the relay ratio is less than a randomly generated number between 0 and 1, the packet is dropped. And finally, the backpressure-rerouting module is used to prevent voids, when a node fails to find a next hop node, and to eliminate congestion by sending messages back to the source nodes so that they will pursue new routes.

When compared to dynamic source routing (DSR) [51] and ad hoc on-demand vector routing

Fig. 15. Routing components of SPEED.

(AODV) [52], SPEED performs better in terms of end-to-end delay and miss ratio. Moreover, the total transmission energy is less due to the simplicity of the routing algorithm, i.e. control packet overhead is less, and to the even traffic distribution. Such load balancing is achieved through the SNGF mechanism of dispersing packets into a large relay area [50]. As explained earlier, similar energy saving technique is used in GBR [27] by spreading traffic uniformly through the network. SPEED does not consider any further energy metric in its routing protocol. Therefore, for more realistic understanding of SPEED's energy consumption, there is a need for comparing it to a routing protocol, which is energy-aware.

6. Conclusion and open issues

Routing in sensor networks has attracted a lot of attention in the recent years and introduced unique challenges compared to traditional data routing in wired networks. In this paper, we have summarized recent research results on data routing in sensor networks and classified the approaches into three main categories, namely data-centric, hierarchical and location-based. Few other protocols followed the traditional network flow and QoS modeling methodology. However, we have also observed that there are some hybrid protocols that fit under more than one category as shown in Table 1. The table summarizes the classification of the protocols covered in this survey. We also included in the table whether the protocol is utilizing data aggregation or not, since it is an important consideration for routing protocols in terms of energy saving and traffic optimization.

Protocols, which name the data and query the nodes based on some attributes of the data are categorized as data-centric. Many of the researchers follow this paradigm in order to avoid the overhead of forming clusters, the use of specialized nodes etc. However, the naming schemes such as attribute-value pairs might not be sufficient for complex queries and they are usually dependent on the application. Efficient standard naming schemes are one of the most interesting future research direction related to this category.

Table 1
Classification of routing protocols in sensor networks

Routing protocol	Data-centric	Hierarchical	Location-based	QoS	Network-flow	Data aggregation
SPIN	✔					✔
Directed Diffusion	✔					✔
Rumor routing	✔					✔
Shah and Rabaey	✔		✔			
GBR	✔					✔
CADR	✔					
COUGAR	✔					✔
ACQUIRE	✔					
Fe et al.					✔	
LEACH		✔				✔
TEEN and APTEEN	✔	✔				✔
PEGASIS		✔				✔
Younis et al.		✔	✔			
Subramanian and Katz		✔				✔
MECN and SMECN			✔			
GAF		✔	✔			
GEAR			✔			
Chang and Tassiulas		✔			✔	
Kalpakis et al.			✔		✔	
Akkaya et al.		✔		✔		
SAR				✔		
SPEED			✔	✔		

On the other hand, cluster-based routing protocols group sensor nodes to efficiently relay the sensed data to the sink. The cluster heads are sometimes chosen as specialized nodes that are less energy-constrained. A cluster-head performs aggregation of data and sends it to the sink on behalf of the nodes within its cluster. The most interesting research issue regarding such protocols is how to form the clusters so that the energy consumption and contemporary communication metrics such as latency are optimized. The factors affecting cluster formation and cluster-head communication are open issues for future research. Moreover, the process of data aggregation and fusion among clusters is also an interesting problem to explore.

Protocols that utilize the location information and topological deployment of sensor nodes are classified as location-based. The number of energy-aware location-based approaches found in the literature is rather small. The problem of intelligent utilization of the location information in order to aid energy efficient routing is the main research issue. Spatial queries and databases using distributed sensor nodes and interacting with the location-based routing protocol are open issues for further research.

Although the performance of these protocols is promising in terms of energy efficiency, further research would be needed to address issues such as quality of service posed by video and imaging sensors and real-time applications. Energy-aware QoS routing in sensor networks will ensure guaranteed bandwidth (or delay) through the duration of connection as well as providing the use of most energy efficient path. QoS routing in sensor networks have several applications including real time target tracking in battle environments, emergent event triggering in monitoring applications etc. Currently, there is very little research that looks at handling QoS requirements in a very energy constrained environment like sensor networks.

Another interesting issue for routing protocols is the consideration of node mobility. Most of the current protocols assume that the sensor nodes and the sink are stationary. However, there might be situations such as battle environments where the sink and possibly the sensors need to be mo-

bile. In such cases, the frequent update of the position of the command node and the sensor nodes and the propagation of that information through the network may excessively drain the energy of nodes. New routing algorithms are needed in order to handle the overhead of mobility and topology changes in such energy constrained environment.

Other possible future research for routing protocols includes the integration of sensor networks with wired networks (i.e. Internet). Most of the applications in security and environmental monitoring require the data collected from the sensor nodes to be transmitted to a server so that further analysis can be done. On the other hand, the requests from the user should be made to the sink through Internet. Since the routing requirements of each environment are different, further research is necessary for handling these kinds of situations.

References

[1] I.F. Akyildiz et al., Wireless sensor networks: a survey, Computer Networks 38 (4) (2002) 393–422.

[2] K. Sohrabi et al., Protocols for self-organization of a wireless sensor network, IEEE Personal Communications 7 (5) (2000) 16–27.

[3] R. Min et al., Low power wireless sensor networks, in: Proceedings of International Conference on VLSI Design, Bangalore, India, January 2001.

[4] J.M. Rabaey et al., PicoRadio supports ad hoc ultra low power wireless networking, IEEE Computer 33 (7) (2000) 42–48.

[5] R.H. Katz, J.M. Kahn, K.S.J. Pister, Mobile networking for smart dust, in: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.

[6] W.R. Heinzelman et al., Energy-scalable algorithms and protocols for wireless sensor networks, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00), Istanbul, Turkey, June 2000.

[7] R. Min et al., An architecture for a power aware distributed microsensor node, in: Proceedings of the IEEE Workshop on signal processing systems (SIPS'00), October 2000.

[8] A. Woo, D. Culler. A transmission control scheme for media access in sensor networks, in: Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'01), Rome, Italy, July 2001.

[9] W. Ye, J. Heidemann, D. Estrin, An energy-efficient MAC protocol for wireless sensor networks, in: Proceedings of IEEE Infocom 2002, New York, June 2002.

[10] E. Shih et al., Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks, in: Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'01), Rome, Italy, July 2001.

[11] L. Subramanian, R.H. Katz, An architecture for building self configurable systems, in: Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing, Boston, MA, August 2000.

[12] F. Ye et al., A two-tier data dissemination model for large-scale wireless sensor networks, in: Proceedings of Mobicom'02, Atlanta, GA, September, 2002.

[13] S. Tilak et al., A taxonomy of wireless microsensor network models, Mobile Computing and Communications Review 6 (2) (2002) 28–36.

[14] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless sensor networks, in: Proceeding of the Hawaii International Conference System Sciences, Hawaii, January 2000.

[15] M. Younis, M. Youssef, K. Arisha, Energy-aware routing in cluster-based sensor networks, in: Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002), Fort Worth, TX, October 2002.

[16] A. Manjeshwar, D.P. Agrawal, TEEN: a protocol for enhanced efficiency in wireless sensor networks, in: Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[17] W. Heinzelman, Application specific protocol architectures for wireless networks, PhD Thesis, MIT, 2000.

[18] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00), Boston, MA, August 2000.

[19] D. Estrin et al., Next century challenges: scalable coordination in sensor networks, in: Proceedings of the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.

[20] S. Lindsey, C.S. Raghavendra, PEGASIS: power efficient gathering in sensor information systems, in: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, March 2002.

[21] S. Lindsey, C.S. Raghavendra, K. Sivalingam, Data gathering in sensor networks using the energy*delay metric, in: Proceedings of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[22] K. Akkaya, M. Younis, An energy-aware QoS routing protocol for wireless sensor networks: in: Proceedings of the IEEE Workshop on Mobile and Wireless Networks (MWN 2003), Providence, RI, May 2003.

[23] B. Krishnamachari, D. Estrin, S. Wicker, Modeling data centric routing in wireless sensor networks, in: Proceedings of IEEE INFOCOM, New York, June 2002.

[24] Y. Yao, J. Gehrke, The cougar approach to in-network query processing in sensor networks, in: SIGMOD Record, September 2002.

[25] W. Heinzelman, J. Kulik, H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, in: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.

[26] D. Braginsky, D. Estrin, Rumor routing algorithm for sensor networks, in: Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, October 2002.

[27] C. Schurgers, M.B. Srivastava, Energy efficient routing in wireless sensor networks, in: The MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force, McLean, VA, 2001.

[28] M. Chu, H. Haussecker, F. Zhao, Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks, The International Journal of High Performance Computing Applications 16 (3) (2002) 293–313.

[29] R. Shah, J. Rabaey, Energy aware routing for low energy ad hoc sensor networks, in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Orlando, FL, March 2002.

[30] N. Sadagopan et al., The ACQUIRE mechanism for efficient querying in sensor networks, in: Proceedings of the First International Workshop on Sensor Network Protocol and Applications, Anchorage, AK, May 2003.

[31] S. Hedetniemi, A. Liestman, A survey of gossiping and broadcasting in communication networks, Networks 18 (4) (1988) 319–349.

[32] D. Ganesan et al., Highly resilient, energy efficient multipath routing in wireless sensor networks, Mobile Computing and Communications Review 5 (4) (2002) 11–25.

[33] A. Buczak, V. Jamalabad, Self-organization of a heterogeneous sensor network by genetic algorithms, in: C.H. Dagli et al. (Eds.), Intelligent Engineering Systems Through Artificial Neural Networks, vol. 8, ASME Press, New York, 1998, pp. 259–264.

[34] C.R. Lin, M. Gerla, Adaptive clustering for mobile wireless networks, IEEE Journal on Selected Areas in Communications 15 (7) (1997) 1265–1275.

[35] A. Manjeshwar, D.P. Agrawal, APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks, in: Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Ft. Lauderdale, FL, April 2002.

[36] M. Youssef, M. Younis, K. Arisha, A constrained shortest-path energy-aware routing algorithm for wireless sensor networks, in: Proceedings of the IEEE Wireless Commu-

nication and Networks Conference (WCNC 2002), Orlando, FL, March 2002.

[37] M. Younis, P. Munshi, E. Al-Shaer, Architecture for efficient monitoring and management of sensor networks, in: Proceedings of the IFIP/IEEE Workshop on End-to-End Monitoring Techniques and Services (E2EMON'03), Belfast, Northern Ireland, September 2003.

[38] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc routing, in: Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, July 2001.

[39] V. Rodoplu, T.H. Ming, Minimum energy mobile wireless networks, IEEE Journal of Selected Areas in Communications 17 (8) (1999) 1333–1344.

[40] L. Li, J. Y Halpern, Minimum energy mobile wireless networks revisited, in: Proceedings of IEEE International Conference on Communications (ICC'01), Helsinki, Finland, June 2001.

[41] G. Finn, Routing and addressing problems in large metropolitan-scale internetworks, University of Southern California, Tech. Rep. ISI Research Report ISI/RR-87-180, 1987.

[42] B. Nath, D. Niculescu, Routing on a curve, in: HOTNETS 1, Princeton, NJ, October 2002.

[43] Y. Yu, D. Estrin, R. Govindan, Geographical and energy-aware routing: a recursive data dissemination protocol for wireless sensor networks, UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001.

[44] B. Karp, H.T. Kung, GPSR: greedy perimeter stateless routing for wireless sensor networks, in: Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '00), Boston, MA, August 2000.

[45] J.-H. Chang, L. Tassiulas, Maximum lifetime routing in wireless sensor networks, in: Proceedings of the Advanced Telecommunications and Information Distribution Research Program (ATIRP'2000), College Park, MD, March 2000.

[46] K. Kalpakis, K. Dasgupta, P. Namjoshi, Maximum lifetime data gathering and aggregation in wireless sensor networks, in: Proceedings of IEEE International Conference on Networking (NETWORKS '02), Atlanta, GA, August 2002.

[47] K. Dasgupta et al., An efficient clustering-based heuristic for data gathering and aggregation in sensor networks, in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'03), New Orleans, LA, March 2003.

[48] F. Ye et al., A scalable solution to minimum cost forwarding in large scale sensor networks, in: Proceedings of International Conference on Computer Communications and Networks (ICCCN), Dallas, TX, October 2001.

[49] K. Akkaya, M. Younis, Energy and QoS aware routing in wireless sensor networks, Cluster Computing, in press.

[50] T. He et al., SPEED: a stateless protocol for real-time communication in sensor networks, in: Proceedings of International Conference on Distributed Computing Systems, Providence, RI, May 2003.

[51] D.B. Johnson et al., Mobile Computing, in: T. Imielinski, H. Korth (Eds.), Mobile Computing, Kluwer Academic Publishers, Dordrecht, 1996, pp. 153–181, Chapter 5.

[52] C. Perkins et al., Ad hoc on-demand distance vector (AODV) routing, in: Internet Draft draft-ietf-manet-aodv-11.txt, June 2002, work in progress.

Kemal Akkaya received his BS and MS degrees in Computer Science from Bilkent University, Ankara, Turkey in 1997 and Ortadogu Technical University (ODTU), Ankara, Turkey in 1999 respectively. He worked as a software developer at an automation project of Siemens and World Bank in Ankara, Turkey in 2000. In the same year, he joined to University of Maryland Baltimore County (UMBC) in Baltimore, MD as a research assistant. He is currently a PhD candidate and research assistant in Computer Science Department at UMBC. His research interests include energy aware routing, security and quality of service issues in ad hoc wireless and sensor networks.

Mohamed Younis received B.S. degree in Computer Science and M.S. in Engineering Mathematics from Alexandria University in Egypt in 1987 and 1992, respectively. In 1996, he received his Ph.D. in Computer Science from New Jersey Institute of Technology. He is currently an assistant professor in the Department of Computer Science and Electrical Engineering at the University of Maryland Baltimore County (UMBC). Before joining UMBC, he was with the Advanced Systems Technology Group, an Aerospace Electronic Systems R&D organization of Honeywell International Inc. While at Honeywell he led multiple projects for building integrated fault tolerant avionics, in which a novel architecture and an operating system were developed. This new technology has been incorporated by Honeywell in multiple products and has received worldwide recognition by both the research and the engineering communities. He also participated in the development the Redundancy Management System, which is a key component of the Vehicle and Mission Computer for NASA's X-33 space launch vehicle. His technical interest includes network architectures and protocols, embedded systems, fault tolerant computing and distributed real-time systems. He has four granted and three pending patents. He served on multiple technical committees and published over 40 technical papers in refereed conferences and journals.