A SURVEY OF WIRELESS SENSOR NETWORK ARCHITECTURES

Almir Davis, Hwa Chang

Department of Electrical and Computer Engineering, Tufts University, Medford, MA USA

almir.davis@tufts.edu, hchang@ece.tufts.edu

ABSTRACT

A wireless sensor network (WSN) consists of spatially distributed autonomous sensors that cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion, or pollutants, at different locations. Recent advances in low-power highly-integrated electronics, advances in micro-electro-mechanical systems (MEMS), rapid growth in the type and quality of available sensors, and progress in communication have allowed WSNs to achieve an unprecedented growth in commercial, industrial and military applications. In order to better understand WSNs, we look at their network architectures. In this survey, we classify existing WSN architectures into specific groups based on WSN behaviour and data flow characteristics. Existing architectures are described and presented along with their advantages and disadvantages. The existing architectures are also evaluated in terms of most common WSN performance parameters such as network lifetime, latency, reliability, quality of service (QoS), fidelity, scalability, modularity, and ease of deployment.

KEYWORDS

Distributed sensing, Wireless sensor networks, network lifetime, survey of architectures

1. INTRODUCTION

Advancements in wireless communications, low-power electronics, battery technology, and power harvesting capabilities have enabled the development of low-cost WSNs. WSNs are characterized by limited power, unreliable communication, need for self-configuration and scalability, harsh environmental conditions, small size, cooperative network behaviour, data centricity (as opposed to address centricity), very small packet size, unattended operation, and random deployment. Given those characteristics, the most common WSN applications are environmental monitoring, health monitoring, terror threat detection, terrestrial and underwater habitat monitoring, military surveillance, seismic oil and gas explorations, inventory tracking, process monitoring, acoustic detections, object localization and tracking, homeland security protection, disaster prevention and disaster recovery, and pipelines corrosion detection. Figure 1. shows an example of WSN architecture. Each node consists of a sensing unit, a processing unit, a communication unit, a battery, and a power harvester. Akyildiz et al. [1] provide a detailed overview of various constraints that drive new WSN designs and present the WSN software stack, consisting of application layer, transport layer, network layer, data-link layer, and physical layer. While all of the WSN layers developed over time, considerable research attention has been given to network layer of the stack. In fact, most architecture acronyms in the literature are associated with the design and development of the WSN network layer.

In this survey, we describe various network layer architectures along with their advantages and disadvantages. Akkaya and Younis [2] surveyed the field of WSN architectures and grouped

them into data-centric, hierarchical, location-based, and network and QoS flow. Yang and Mohammed [3] define the same architectural groups as Akkaya and Younis but add additional architectures to each group. Singh et al. [4] add three architectural groups: mobility-based architectures, multi-path-based architectures, and heterogeneity-based architectures. Finally, Yick et al. [5] add geographical routing and anchor location service (ALS) to location-based architectures and security routing (SecRout) and secure cell relay (SCR) to the hierarchical group.

We group all architectures/protocols as follows:

- Data-centric architectures
- Hierarchical architectures
- Location-based architectures
- Mobility-based architectures
- Quality of Service (QoS) architectures
- Other Architectures
  - Network flow architectures
  - Multipath-based architectures
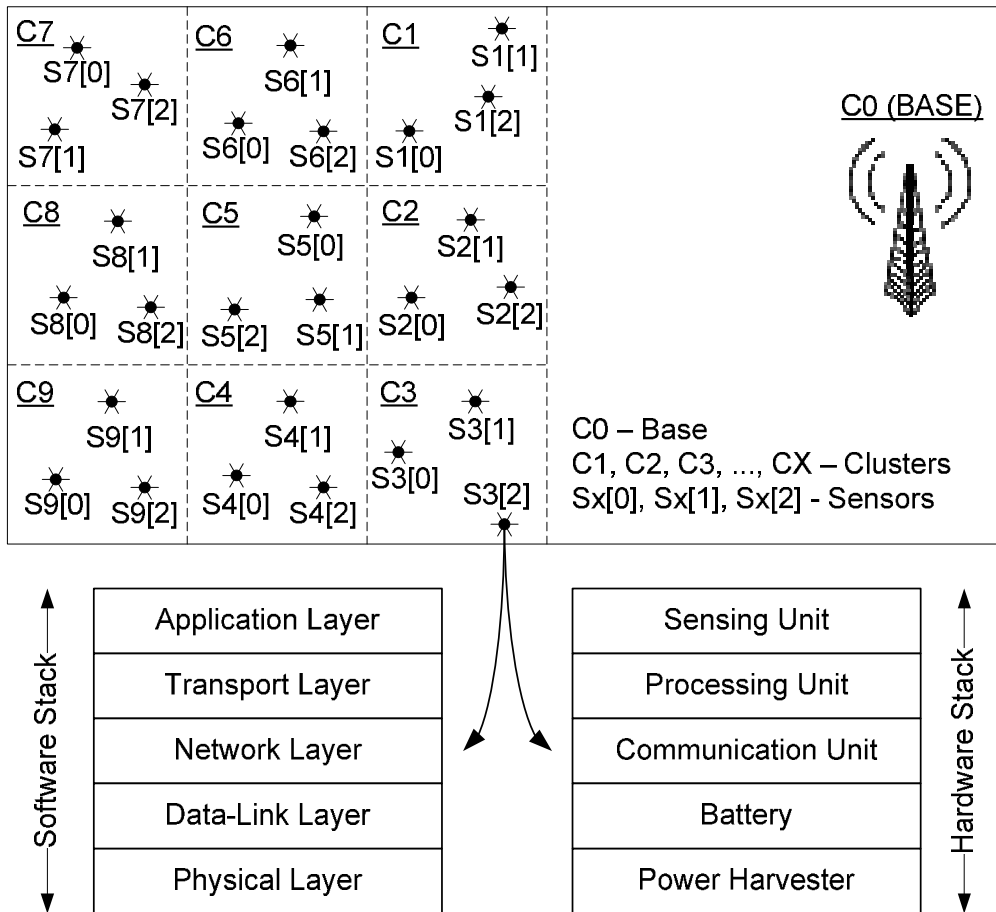 - Heterogeneity-based architectures

Figure 1. WSN Stack

Table 1 groups all major architectures.

Table 1. WSN Architectures Grouping

Group	Architectures/Protocols	
Data-centric	1. Flooding 2. Gossiping 3. SPIN 4. Directed Diffusion 5. Rumor Routing 6. Energy-aware routing for low-energy ad-hoc WSN 7. STCP	8. Gradient-based 9. COUGAR 10. ACQUIRE 11. Information dissemination by negotiation 12. EAD 13. Information-directed
Hierarchical	1. LEACH 2. PACT 3. HEED 4. PEGASIS 5. Hierarchical-PEGASIS 6. TEEN	7. APTEEN 8. Energy-Aware Routing for Cluster-based WSN 9. SecRout 10. SCR
Location-based	1. GAF 2. SPAN 3. GEAR 4. GeRaF 5. TBF	6. ALS 7. BVGF 8. MECN 9. SMECN 10. Geographic Routing in Lossy WSNs
Mobility-based	1. SEAD 2. TTDD 3. Joint Mobility and Routing 4. Data MULEs	5. Dynamic Proxy Tree-based dissemination 6. MMAC 7. MS-MAC 8. VBF
QoS	1. SAR 2. SPEED 3. Energy-Aware QoS	4. RL-MAC 5. MMSPEED 6. DAPR
Network flow	1. Max Lifetime Energy 2. Max Lifetime Data Gathering and Aggregation	3. Min Cost Forwarding
Multipath-based	1. Node-disjoint 2. Braided Path 3. N-to-1 Multipath Discovery	4. SEEM 5. REER 6. HMPR
Heterogeneity-based	1. CADR 2. IDSQ 3. CHR	4. HDMRP 5. SEP 6. EEHC

2. DATA-CENTRIC ARCHITECTURES

Data-centric architectures are characterized by a vast number of randomly deployed sensors that only communicate node-to-node without any global network identification. In these types of architectures, a sink node sends a request query through the network of nodes, and the source node responds to the query; alternatively, the source node sends an event query, and the sink node routes to the event. The goal in these architectures is to send the data through the most efficient route between the sink node and the source node. Data-centric networks tend to be power inefficient, because the entire network is involved in data transfer. Many data-centric protocols try to improve the power efficiency by creating dedicated source-to-sink routes so that the rest of the network can save power. Data aggregation is another popular approach, because it reduces the number of packets traversing through the network. Data-centric architectures do not require global clock synchronization.

Flooding and gossiping [6] are the most prominent representatives of data-centric architectures. In *flooding*, each sensor node sends data to all of its neighbors. The send packet is propagated throughout the entire network until it either reaches the destination or the number of maximum hops is reached. Flooding was one of the first data-centric protocols. The main flooding disadvantages are implosion and overlap. Figure 2 explains both problems using four nodes (A,
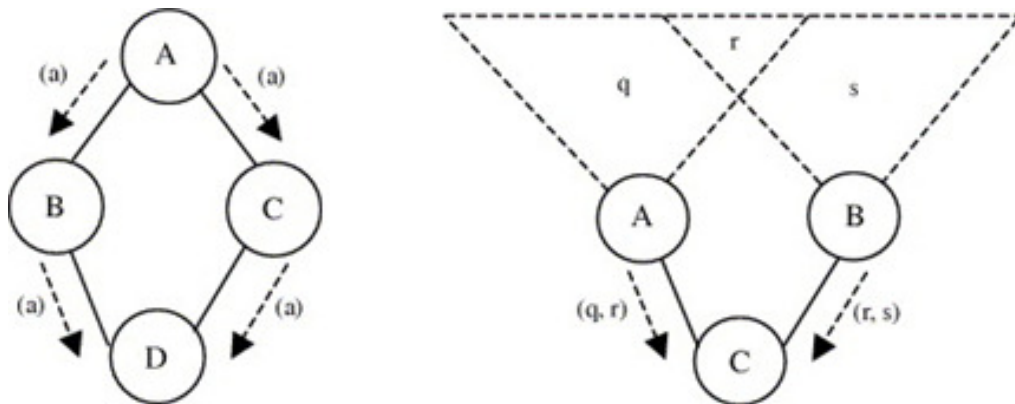

Figure 2. Flooding implosion and overlap problem.

B, C, and D). Implosion is where a packet originating from the same source (Node A) travels through different paths (Nodes B and C) but arrives as a duplicate at some other node (Node D) within the network. This creates power inefficiencies within the network. A similar case is the overlap problem, where data indeed originates from two different sources (Nodes A and B), but both sources cover the same overlapping area r, resulting in a data duplicate at the node neighboring both source nodes (Node C). Another disadvantage of the flooding protocol is the fact that all nodes must be on all the time to avoid missing packets. This is very power inefficient, leading to extremely short network lifetime.

The *gossiping* protocol is a more efficient version of flooding, because it uses a single, randomly selected neighbor to transfer each packet. Therefore, gossiping avoids the implosion problem by creating a single random path from the source to the sink. The overlap problem still exists in gossiping. Another disadvantage is that gossiping creates long propagation delays, because the selected random path might be suboptimal in terms of propagation latency. In fact, the propagation delay is not bounded within any limits, because the selected data path is random.

Flooding and gossiping protocol offer simple implementation, and there is no need for network level synchronization among nodes. There is also no need for global addressing scheme or a sophisticated clock distribution procedure.
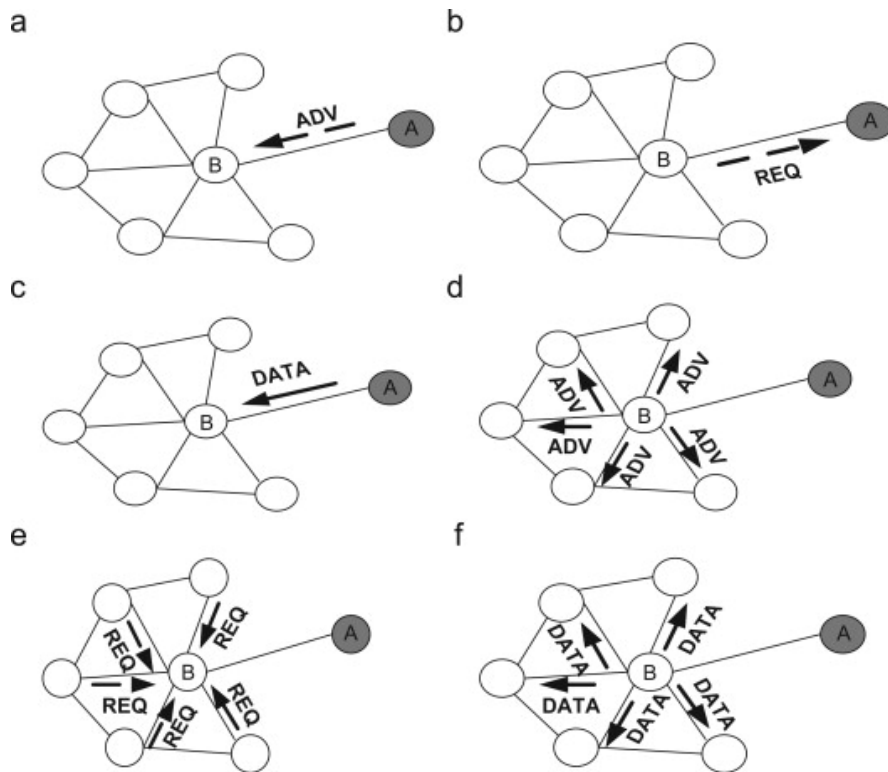
Figure 3. SPIN Protocol

Sensor Protocol for Information via Negotiation (SPIN) [*7*] starts with a source (Node A) advertising the availability of its data to neighboring nodes (Figure 3). The ADV message contains meta-data necessary for neighboring nodes to decide if they would like to acquire the data. The nodes interested in data submit their requests for data (REQ message); in return, they receive the data. The negotiations between nodes continue until the data reach its final destination—the sink node. The SPIN advantage is seen in the relative localization of topological changes. In other words, changes in the location of nodes only affect local negotiation but also the overall source to sink delivery. However, SPIN suffers from lack of quality of service (QoS) and cannot ensure that negotiation among nodes along the source-sink path will guarantee the final delivery of data (i.e., intermediate nodes might decide not to request data upon receiving the data advertising message).

The *directed diffusion* architecture [*8*] starts with sink advertising or requesting data and nodes responding to the request (Figure 4). The transfer starts with a source flooding the network with messages containing attribute-value request pairs. For example, a pair might be (location, 50°C) signifying the request for location of the place where the measured temperature exceeds 50°C. Once the flooded message arrives at the source node, the node issues the acknowledgment of data existence. While the acknowledgment message travels back to sink, the hop gradient, defined as the measure of hop distance between two nodes, is recorded and propagated along with the acknowledgment. Next, the sink nodes send the request for data through the most optimal path, effectively reinforcing the path through which it would like data to go. Finally, data are sent through the designated path. A major advantage of directed diffusion is the node-to-node communication without the need for a global network addressing mechanism. In addition, nodes can cache and aggregate data, which in turn saves overall network power consumption, and the data traverse the network only when requested, avoiding unnecessary

power consumption when the data are not needed. On the other hand, directed diffusion is not suitable for applications that require immediate reporting to a trigger event (such as military and homeland security applications). It also has poor QoS, because the latency between source and sink can vary greatly.
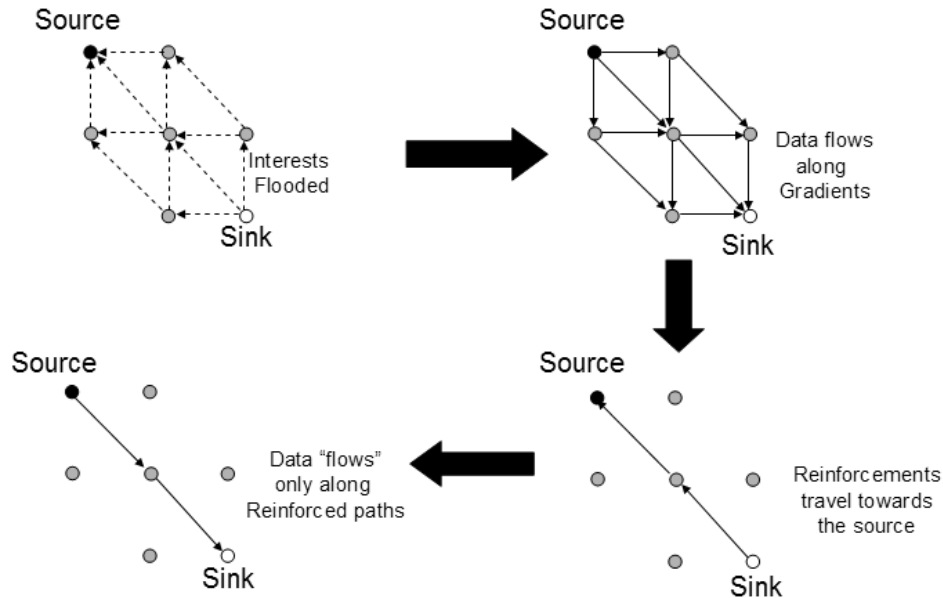

Figure 4. Directed Diffusion

Gradient-Based Routing (GBR) [9] defines the distance between the event source and individual nodes in hop counts. The minimum number of hops from the source to the user is called the node height. The difference between a node's height and that of its neighbor is called the link gradient. A packet is always sent via path with the largest gradient. For example, if the packet originating Node A has a hop count of 15 and its neighbors Node B and Node C have hop counts of 14 and 7, respectively, then grad(Node A, Node B) is 1, and grad(Node A, Node C) is 8. The higher gradient is grad(NodeA, NodeC); therefore, the packet will be sent from Node A to Node C. Since Node C's height is 7, the arriving packet's maximum number of additional hops will be 7 as opposed to 14 if the packet were sent via Node B. GBR is an improved version of the directed diffusion protocol, with an increased network lifetime of up to 90%. Like many other data-centric architectures, GBR also takes advantage of data compression and data fusion.

In *rumor routing* [10], flooding the network with queries or events is prevented through the use of packets called agents (Figure 5). In the case of an event, the event triggering node generates a packet called an event agent. The agent is then sent through several random paths advertising the event existence and the source of the event. All nodes that are not already familiar with the particular node log the agent into their routing tables. Eventually, the particular event's agent builds one or more event agent paths. The node that is interested in the event sends its own agent, the query agent, through multiple nodes until it reaches the node that knows how to route the particular query to the source node. Therefore, the query agent path and event agent path cross, and the path from the source to the sink is established. Rumor routing is energy efficient since it prevents sources and sinks from flooding the network with events and queries. On the other hand, it establishes only a single path between the source and the sink, creating a network reliability problem. In addition, rumor routing does not guarantee the discovery of a successful route between the source and the sink nodes, because it cannot ensure that the query agent's and the event agent's paths will cross in the discovery phase.
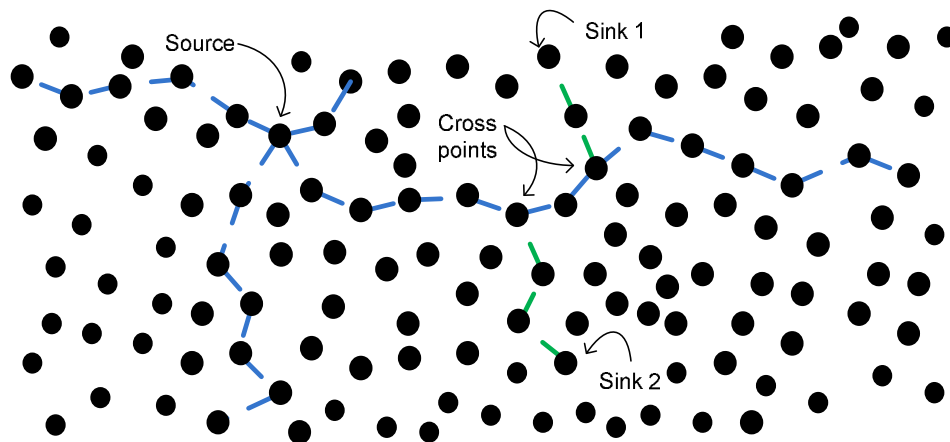
Figure 5. Rumor Routing

Energy Aware Routing for Low Energy Ad Hoc Sensor Networks [*11*] extends network lifetime by occasionally using power sub-optimal routes. Shah et al. argue that permanently using the most optimal minimum energy path will actually decrease the network lifetime, because the same nodes will experience a disproportionally high traffic compared to other less utilized nodes.

COUGAR architecture [*12*] is a software approach to solving the WSN power efficiency issue. In the COUGAR, the number of power costly data transmissions from individual nodes to the base (central data gathering place) is replaced with cheap local computation. In other words, the so-called declarative query generates an efficient optimized query plan that interrogates only necessary nodes and reduces their data load by using local processing. The result is minimum data communication with the base. However, the power savings from the reduced communication among nodes come at the expense of the more sophisticated nodes' communication stack. Recently, COUGAR introduced a new communication layer called query layer. In addition, leader nodes that generate declarative queries are disproportionally utilized, creating a power imbalance among nodes.

Another interesting data-centric architecture is *ACQUIRE - ACtive QUery forwarding in sensoR nEtworks* [*13*]. In ACQUIRE, the sink node sends a data request query that propagates throughout the network either randomly or through some directed means. Each time that an active query reaches an intermediate node, the node will try to resolve the query using either event cached information or information from its neighbors *d* hops away from the node. Once the query is fully resolved, the response to sink is sent via the most optimal intermediate nodes, and the sink-source path is established. Therefore, ACQUIRE relies on selecting the most power efficient path by each node on the path looking *d* hops ahead for the most optimal solution. For *d=1* case*,* ACQUIRE behaves as flooding architecture.

Information dissemination by negotiation [*14*] is an energy efficient data-centric approach that has three major advantages: a fully distributed network, high success rate for data retrieval, and capability to deal with mobile sensors in addition to static sensors. Energy efficiency is measured by the number of message transmissions required for the source node to advertise its data to all possible data consumers and by the number of hops of the path between the source node and the querying node for data transmission.

Additional data-centric power efficient architectures are Energy-Aware Data-Centric Routing (EAD) [*15*], information-directed routing [*16*], and sensor transmission control protocol [*17*].

3. HIERARCHICAL ARCHITECTURES

Data-centric network topologies are not suitable for large-scale sensor networks. Covering a large area without performance degradation is not possible with data-centric architecture. Moreover, in data-centric architectures, the reporting latency increases with the size of the network. The data-centric approach also causes significant power inefficiencies as the network grows.

The network scalability issue is addressed in hierarchical routing. The hierarchical routing's main goal is to efficiently maintain network power consumption even in large-scale networks. In other words, hierarchical routing allows the network to scale in a number of sensor nodes. Most hierarchical architectures consist of sensor nodes grouped into cluster heads. Cluster heads build intra-cluster communication with other nodes within the same cluster, but they also build inter-cluster communicate with other cluster heads. Cluster heads aggregate data obtained individual sensors and then transfer the same information mostly in a multi-hop approach to the base.

Low-Energy Adaptive Clustering Hierarchy (LEACH) [18] is one of the most popular hierarchical architectures. LEACH utilizes a randomized rotation of local cluster heads to evenly distribute the energy load among sensors in the network. It also minimizes the overall energy consumption by allowing each sensor node to determine which cluster it wants to join by choosing the cluster head that requires the minimum communication energy (typically, the cluster head closest to the sensor). However, LEACH architecture determines the percentage of cluster heads in the network and cluster switching frequency a priori. This approach may lead to a less than optimal number of cluster heads in the network at any point of time. It also leads to unnecessary overuse of cluster head switching and a waste of network power capacity associated with the switching overhead.

Power Aware Clustered TDMA (PACT) [19] uses a more efficient cluster head switching algorithm. The PACT architecture takes individual nodes (sensors) energy levels into account when selecting cluster heads. PACT also uses passive clustering [20] that limits the number of exchange control messages and therefore reduces the power overhead associated with cluster head switching. However, PACT cluster switching is still probabilistic and does not always lead to optimal network lifetime. Additionally, neither PACT nor LEACH uses ambient power harvesting methods to extend the network lifetime.

The *Hybrid Energy-Efficient Distributed (HEED)* hierarchical architecture [21] extends the network lifetime by taking into account the residual energy of each node (primary parameter) and considers intra-cluster communication cost (secondary parameter). The second parameter is based on AMRP (average minimum *reachability* power) and is a good measure of communication energy consumption if the node becomes a cluster head. Like LEACH, HEED selects a percentage of cluster heads a priori that does not always lead to an optimal number of cluster heads. Furthermore, HEED cluster head selection is probabilistic. The selection heavily relies on an a priori selected percentage of cluster heads allowed in the network and the a priori selected minimum ratio between the sensor residual and maximum energy.

Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [22] builds sensor node chains rather than clusters. Figure 6 explains the concept; each node (s1 through s4) sends its data to only one neighboring sensor. As sensors send data to only one of their neighbors, they build a chain of sensors, and only one node transmits data to the base or to another cluster. PEGASIS is very efficient, because it allows each node to have its transmitter turned on for only one TDMA slot and its receiver also for only one TDMA slot. In all other TDMA slots, the node can sleep and therefore conserve energy. PEGASIS is also power efficient, because the cluster setup phase is minimal. On the other hand, the biggest disadvantage is the excessive delay for distant nodes.

Hierarchical PEGASIS [*23*] improves traditional PEGASIS by decreasing the propagation delay (Figure 6). In hierarchical PEGASIS, each node sends data to only one neighbor; instead of building a chain of nodes, it builds a binary scheme of nodes. The binary scheme of nodes precludes the protocol with embedded CDMA coding, because multiple transmissions occur in the same TDMA slot. The use of CDMA code might significantly decrease the channel bandwidth and therefore cause network power inefficiencies. In addition, the binary scheme requires more carefully synchronized nodes.
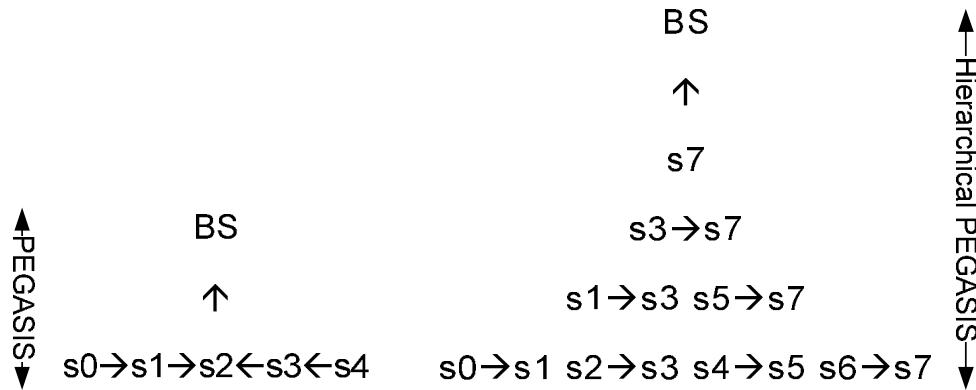

Figure 6. PEGASIS and Hierarchical PEGASIS routing

The *threshold sensitive energy efficient sensor network (TEEN)* [*24*] architecture relies on building two levels of cluster head nodes (Figure 7). Individual nodes are grouped into clusters, and clusters are represented via cluster heads. Cluster heads aggregate the data of all individual nodes and transfer them further toward the base. If a cluster head has an uplink path to at least one cluster head farther away from the base, then it is a second-level cluster head. Unlike first-level cluster heads, which only transfer cluster data down the link to the base, second-level cluster heads aggregate and transfer all data originating from directly linked cluster heads.

The feature that differentiates TEEN from any other hierarchical architecture is the use of hard and soft thresholds. The hard threshold defines when an individual node is allowed to send its data to the cluster head. In other words, if a particular node's attribute (e.g. temperature) reaches the threshold value, the node reports the event to cluster head once. The software threshold defines the incremental delta value that needs to be reached in order for node to report the event again. For example, if the hard threshold is $70^{o}C$ and the soft threshold is $5^{o}C$, the node will report the temperature rise at $70^{o}C$, $75^{o}C$, $80^{o}C$ and so on. Hard and soft thresholds help TEEN limit the number of intra-cluster messages by filtering small changes in measure attributes. TEEN does not perform well in applications where periodic reports are needed, since the user may not receive any data if the thresholds are not reached. If the collected data does not exceed hard threshold, the node does not transmit any sensed data. And if it does not exceed soft threshold, we cannot know about data changes after the default value is passed, especially if the data change is under the threshold value. Moreover, due to those thresholds it is hard to judge whether the nodes are alive or not.

The *Adaptive Threshold Sensitive Energy Efficient Sensor Network (APTEEN)* [*25*] architecture is an extension of the TEEN architecture. APTEEN addresses TEEN's shortcomings by capturing periodic data collections and reacting to time-critical events. In addition to all of TEEN's features, APTEEN also supports three different query types: historical, to analyze past data values; persistent, to deliver data on a regular basis; and one-time, to take a network snapshot. While APTEEN extends the number of applications in which it can be used, it is still considered one of the setup overhead heaviest architectures.
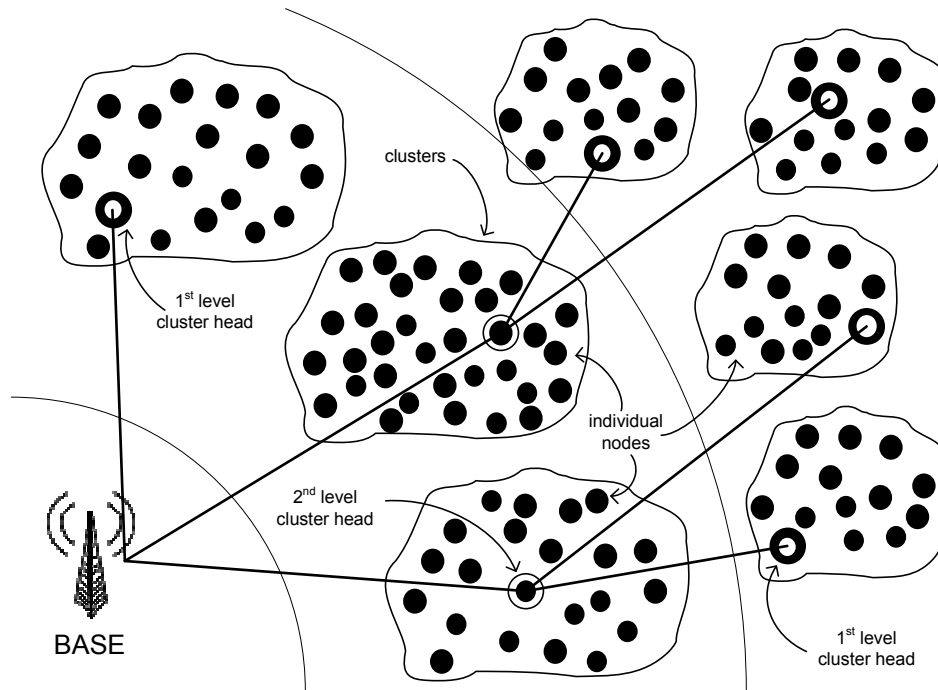
Figure 7. TEEEN and ATEEN architecture

Energy-aware Routing for Cluster-based Sensor Networks [*26*] is an interesting novel architecture since it takes into account the latest hardware features to save transmission power. This architecture relies on individual nodes being capable of adjusting their transmission power to account for the distance range between nodes. The main architectural structures are gateways (cluster heads) that program individual nodes with the exact TDMA schedule as well as precise functionality (sensing nodes, sensing-relaying nodes, relaying nodes and inactive nodes). An extended version of the same architecture is proposed in [*27*] where the algorithm constrains the minimum transmission range in order to limit the delay.

Security Routing (SecRout) [*28*] is cluster-based approach that emphasizes the secure delivery of packets from the source to the sink. This architecture employs standard hierarchical routing network elements such as individual nodes, clusters, cluster heads and sink. Packet transfers among sensor nodes are secured via symmetric cryptography. Each sensor is given a unique ID and a unique pre-loaded key. All individual sensors with a cluster use the cluster head's KEY to encrypt data. The cluster head decrypts and aggregates all data from all sensors within the same cluster. Then, it encrypts the aggregated packet and sends it back to the sink node. The sink node (base) is assumed to be trusted and power-rich. The sink node also contains all ID/KEY pairs from all sensor nodes and therefore can easily detect an attack.

Secure Cell Relay (SCR) [*29*] is an even more secure architecture providing security against the following attacks: Sybil, wormhole, sinkhole, selective forwarding, and hello flood. In SCR, the sink node distributes a global key that is used for initial neighborhood discovery and handshake communication. In the discovery phase, nodes use a three-way handshake protocol to establish a shared secret key between neighboring nodes. Therefore, each pair of neighboring nodes shares one unique secret key. Once the discovery phase is complete, the global shared key is no longer needed, so it is destroyed. Another SCR feature is that routing paths from the source to the sink are formed through a series of cells and cannot be altered via wormhole or sinkhole attacks.

4. LOCATION-BASED ARCHITECTURES

Location-based architectures along with their underlining routing algorithms rely on knowledge of nodes' positions to route packets. Nodes might obtain their positions using low-power, embedded GPS receivers, through triangulation techniques, or simply by being placed at the known location. In this section, we concentrate on location-based architectures with the primary goal of energy efficient data routing.

Geographic Adaptive Fidelity (GAF) [*30*] is an energy-aware location-based architecture that conserves energy by identifying routing equivalent nodes and then turning off the unnecessary nodes, keeping a constant level of routing fidelity Figure 8 depicts an example. Here, assume that each node from region 1 can talk to each node from region 2. Likewise, each node from region 2 can talk to each node in region 3. None of the nodes from region 1 can talk to any node from region 3. Since Nodes B, C, and D effectively cover the same region 2, two of them can go to sleep without affecting the overall routing scheme. Therefore, GAF saves energy by turning off two of the three nodes in region 2. Once the predetermined active time expires, a new node wakes up and takes over the responsibilities of the currently active node, which then goes to sleep. By activating and deactivating different nodes that cover the same region, GAF significantly extends the network lifetime. GAF can be implemented for non-mobility and mobility of nodes. The disadvantage is that the leading nodes do not aggregate, filter, or compress data.
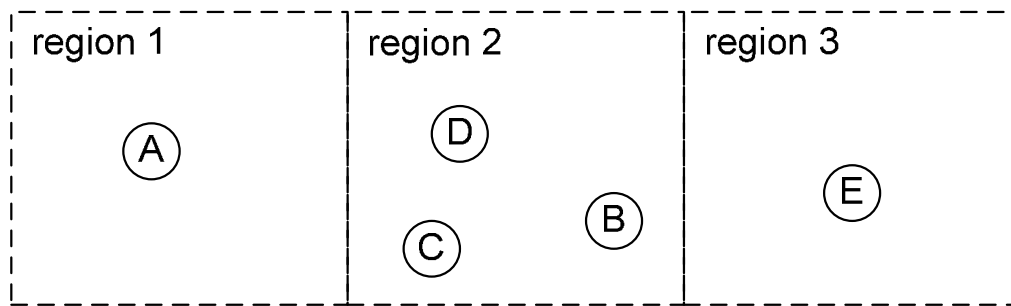

Figure 8. GAF architecture

SPAN architecture [*31*] [*32*] uses its routing algorithm to efficiently select a set of backbone nodes (coordinators) whose goal is to efficiently transfer packets between the sink node and the source node. All other nodes not currently used as coordinators can retain inactive (sleep) status and therefore save energy. SPAN aims to satisfy the following requirements:
* As many nodes as possible should be able to sleep most of the time.
* The reduced set of active nodes shall forward packets between any source and destination with minimally more delay than if all nodes were awake.
* The backbone capacity (throughput) should be as high as the capacity of the original network.
* There shall be seamless coordination, interoperability, and sleep mode support by all link-layer and physical layer protocols.

SPAN design achieves four goals. First, it ensures that enough coordinators are elected so that the network stays fully connected. Second, it rotates the coordinators in order to ensure the equal power burden among all nodes. Third, it attempts to minimize the number of nodes elected as coordinators, thereby increasing network lifetime, but without suffering a significant loss of capacity or an increase in latency. Fourth, it elects coordinators using only local neighbors' routing tables. SPAN performance increases with increased node density. A disadvantage is the relatively high overhead associated with coordinator selection and backbone routing algorithms. This is magnified in larger and denser WSNs.
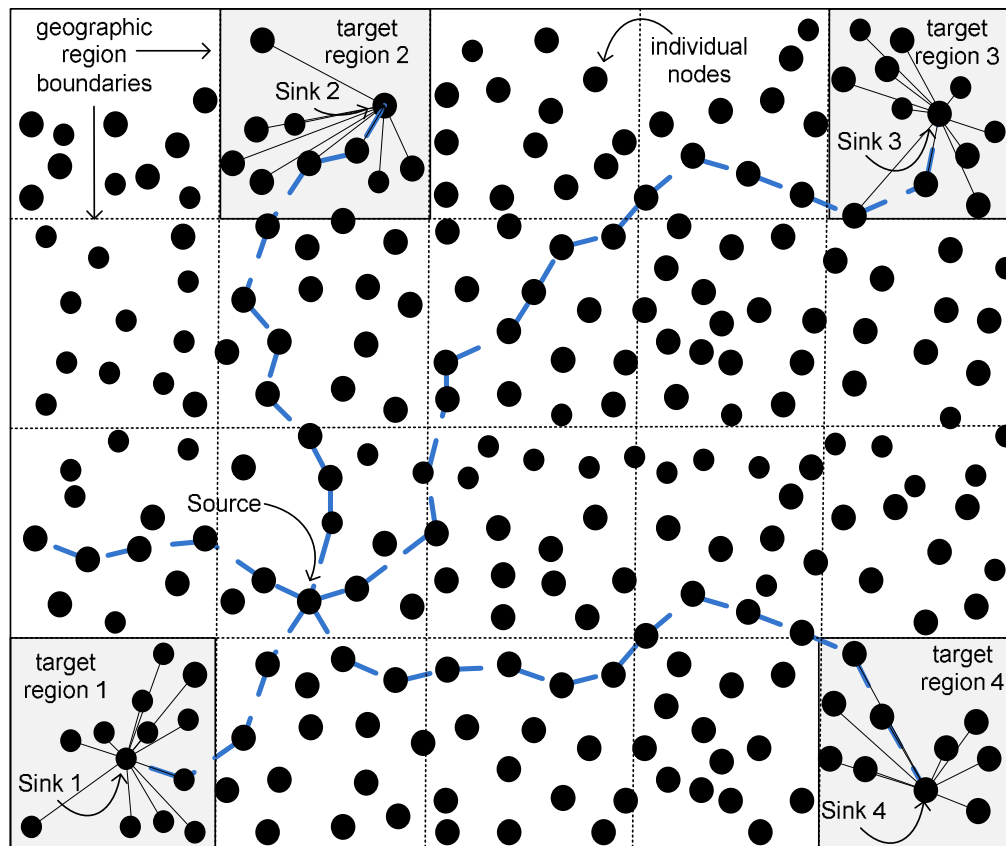
Figure 9. GEAR architecture

Geographic and Energy Aware Routing (GEAR) [*33*] is an energy-aware architecture that uses nodes' geographical information to route packets to the target area. In a sense, it is similar to directed diffusion, except the interest is sent to the specific target region. Figure 9 depicts the architecture. There are two phases in the algorithm. In the first phase, each packet is routed through the set of nodes that uses the nearest-neighbor-to-target-region approach to select the next hop. In the second phase, the packet that arrives in the target region is diffused using either recursive geographic forwarding or restricted flooding. Unlike other architectures, GEAR ensures that all nodes within the target region receive the packet. Like all other location-based architectures, GEAR requires the knowledge of nodes' positions to properly and energy-efficiently route a packet from the source to the target region (sink).

Geographic Random Forwarding (GeRaf) [*34*] is a novel greedy-based forwarding architecture that uses a randomly selected relay node from the region closest to the sink to route its data packets. The process starts with the source node sending a request-for-send (RTS) message to all nodes within a priority region. The priority region is known, because the source node uses location-based architecture that requires the location knowledge of neighboring nodes. If there is no clear-to-send (CTS) reply from the priority region, the source node sends an RTS to the region with the second most priority. If there is still no answer, the source node continues traversing the priority region until the CTS message is received, or the source node simply gives up and declares the packet as undeliverable (best-effort forwarding). However, if a CTS is received, the source node simply forwards the data packet to the CTS message originating node. The transaction is complete once the source node receives the data acknowledgment packet. If multiple relay nodes send CTS messages, a contention resolution algorithm is used to resolve the contention and to allow a single randomly chosen node to send a CTS back to the source

node. The same concept is applied to further advance the packet through relay nodes to the final destination (i.e. sink node). GeRaf is considered a greedy, best-effort forwarding architecture that does not guarantee packet delivery. This is a substantial disadvantage, especially when used in applications requiring a certain level of QoS. On the other hand, GeRaf is adaptable to network topology changes created by nodes changing their status from active to sleep and vice versa.

Trajectory-Based Forwarding (TBF) [*35*] uses node geolocation to forward packets via a predetermined, source-specified trajectory. The advantage of the approach is that the trajectory specifies the general direction or nature of the path, but it does not specify the exact nodes that need to participate in the packet forwarding. This feature allows the network to route or even reroute packets using the best available resources that are close to the projected path. Figure 10 depicts various trajectories descried in [*35*]. The TBF protocol is very flexible and can implement path redundancy by simply sending the same packet through two or more separate trajectories.
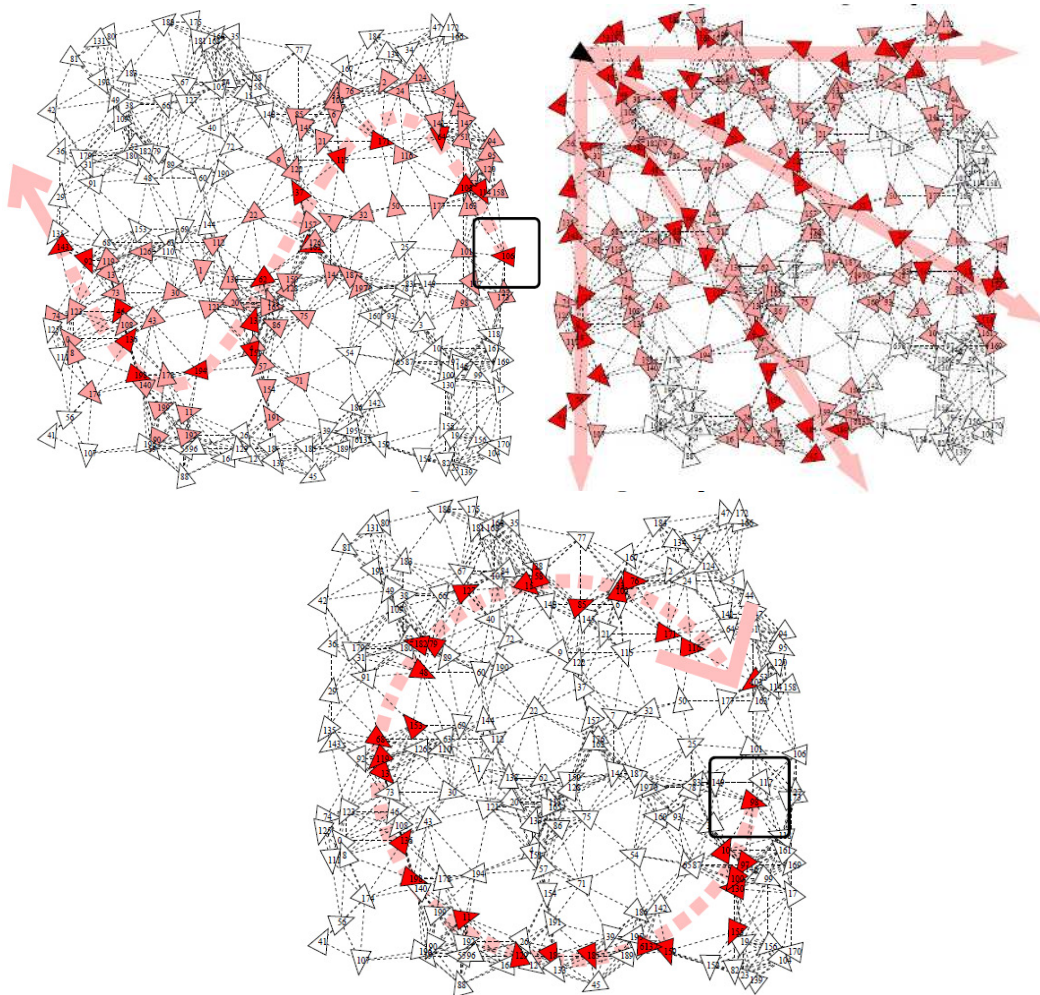

Figure 10. TBF architecture

Anchor Location Service (ALS) [*36*] is a power-efficient, location-based architecture that supports routing among multiple moving sources and destinations. Figure 11 depicts the basic ALS routing mechanism. ALS constructs a virtual Cartesian grid with a scalable number of ideal vertexes. Ideal vertexes are not real nodes but rather virtual positions of squares' vertexes that are used as the reference points for further location-based ALS routing steps. Each square is α*α. Therefore, all nodes belong to one of the squares that is (a*α, b*α) away from some reference geographic point in the coordinate system. In the initialization phase, the sensor node closest to the virtual grid ideal vertex is selected as the so-called grid node for that particular square. In Figure 11, virtual grid vertex G8 is used as the location reference for selecting Region-G8B's grid node. Therefore, each square in the coordinate system has one designated grid node. The same grid node also becomes a source or a sink agent node if the source or the sink node is found in the particular square.
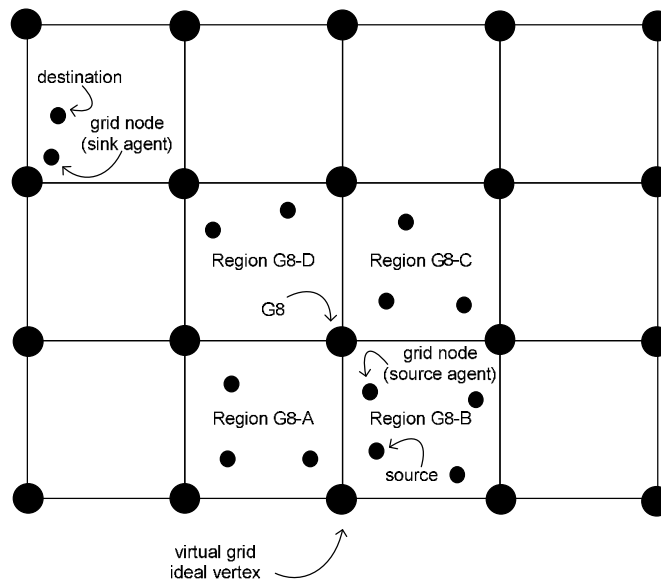

Figure 11. ALS geographical grid

The sink agent is responsible for distributing the information about the sink location via the anchor system of grid nodes. Each sink node builds its own anchor system. Having separate anchor systems, sink nodes are allowed to move within the network without losing the node's synchronization with the overall network's node structure. The source agent is responsible to find and attach to the sink's anchor system. Once the source agent discovers the sink's anchor system, ALS uses the location-based routing algorithm to find the most power-efficient path from the source to the sink. Implementation of ALS tends to be straightforward and less cumbersome than the average location-based routing scheme. Its advantage is the fact that it supports multiple moving sources and sinks with modest storage and communication power requirements. It is also scalable in terms of covered geographic space as well as the network density.

Other power efficient location-based architectures are Bounded Voronoi Greedy Forwarding (BVGF) [*37*], Minimum Energy Communication Network (MECN) [*38*], Small Minimum Energy Communication Network (SMECN) [*39*], and Geographic Routing in Lossy WSNs [*40*].

5. MOBILITY-BASED ARCHITECTURES

Mobility-based architectures assume that a source, a sink, or intermediate nodes change their positions over time. Some architectures also assume that there are multiple sources and multiple

sinks in the WSN field. Routing through a constantly moving set of nodes is a difficult problem that requires a lot of energy to keep the network well connected. Consequently, architectures presented below limit the problem to mobile sources and sinks moving within the stationary network of intermediate nodes. This approach is reasonable, because most applications have a stationary network structure in the field and expect only a source node or a sink node to move.

Scalable Energy-efficient Asynchronous Dissemination (SEAD) [*41*] supports moving data from a stationary source to a moving sink via a network of stationary nodes. The SEAD architecture starts with a source node that builds its own dissemination tree. In the case of multiple source nodes, there are multiple dissemination trees. The sink node is not a part of the tree; rather, it creates a relationship with the closest node belonging to the tree. The closest node to the sink becomes the sink's access node, which seeks to transfer source data via the dissemination tree. Once the data are available at the access node, the node simply transfers the data to the sink. SEAD is very flexible, because it allows the sink to move and change its access node. The access node changes once the distance threshold between the sink and its access node is reached. The value of this threshold allows trade-offs to be made between path delay and energy spent on reconstructing the tree. SEAD also allows limited network traffic reduction by being able to send data to multiple sink nodes.

Two-Tier Data Dissemination (TTDD) [*42*] is similar to SEAD in that it also relies on stationary source and network nodes while allowing multiple sinks to move. Figure 12 depicts TTDD data flow. It starts with each source building its grid structure (one tier). Once the grid structure is established, the data requesting sink node floods the local cell to find the cell's dissemination node (i.e. the node closest to the virtual cell vertex). Once it finds the dissemination node, the path from the sink to the source is established. Then, the sink requests the data, and the source responses with the data delivery. As Figure 12 depicts, TTDD allows for limited data traffic reduction by requesting only once identical requests originating from different sink nodes (see Node G).

Joint Mobility and Routing [*43*] defines mobile sink nodes (called base stations) very differently than SEAD and TTDD. In this architecture, the nodes closest to the sink (base) deplete their energy fastest, because all other distant network nodes route their data through those nodes
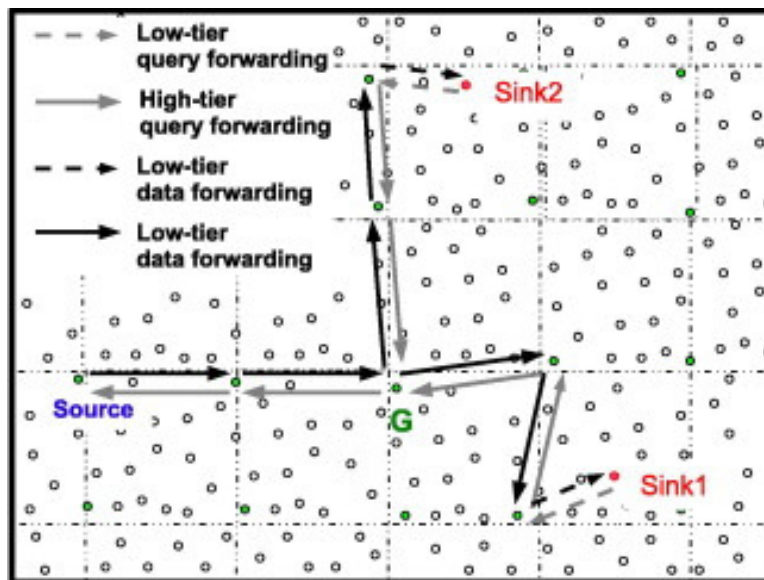

Figure 12. TTDD architecture

closest to the sink. In order to prolong the network lifetime, a mobile sink node constantly changes its position, which forces the network rotation of the closest nodes. Therefore, power consumption is evenly split among all network nodes. Sink mobility and the network routing protocol are developed jointly, so that the harmonious coexistence ensures even longer network lifetime. The main disadvantage is the fact that a mobile sink must be some kind of moving robot capable of traversing through long and sometimes unevenly distributed networks and terrains. This type of architecture might be suitable for underwater WSN applications.

Data MULEs [*44*] propose special, mobile nodes called MULEs (Mobile Ubiquitous LAN Extensions) that move through the network and pick up data from nodes found in close proximity. The close-range transfers can use promising communication technologies such as Ultra-Wideband (UWB) radios. Figure 13 conceptualizes the architecture proposed in this paper. By establishing data transfers from the source to the sink via mobile data MULEs, significant power savings can be achieved. On the other hand, there is substantial power loss due to the continuous listening needed to identify a passing MULE. Also, data latency is high due to the fact that sink nodes must first wait for source nodes to offload data and then wait for MULEs to deliver the data. However, a major advantage is the low cost of placing and maintaining the network.

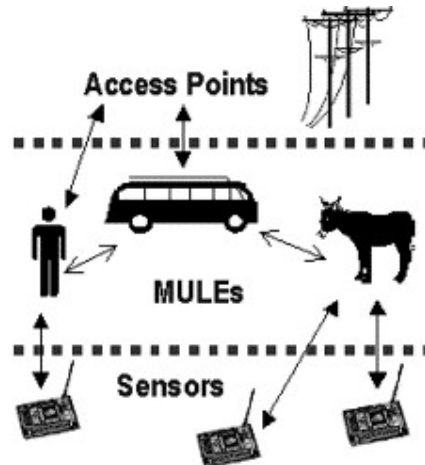*Dynamic Proxy Tree-based Data Dissemination* [*45*] is another mobile-based architecture that

Figure 13. The MULEs three-tier architecture

relies on the dynamic proxy tree-based framework. In this framework, each source is associated with a source proxy, and each sink node is associated with a sink proxy. Proxies related to the same source build a proxy tree, which is the facilitator of data movement. The source node disseminates data through its source proxy, which further propagates data to multiple sink proxies. The sink can then query its proxy to obtain data. The advantage of this architecture is the efficient reconfiguring of the proxy tree, as the proxies frequently change from one node to another.

Additional mobility-based architectures are *Mobility-adaptive Collision-free Medium Access Control (MMAC)* [*46*], *Mobility-aware MAC (MS-MAC)* [*47*], and *Vector-based Forwarding (VBF)* [*48*].

6. QUALITY OF SERVICE (QOS) ARCHITECTURES

QoS architectures are characterized by stringent requirements such as packet end-to-end network delay and packet end-to-end energy cost. QoS is usually needed for networks required to deliver real-time data or to deliver data with predefined reliability metrics. Generally, QoS architectures are complex with high network maintenance overhead, because WSNs are generally viewed as non-deterministic, randomly spread set of nodes with limited lifetime.

Sequential Assignment Routing (SAR) [*49*] creates multiple trees, each rooted from one-hop neighbor of the sink. The trees are created by taking into account the link cost between immediate neighbors. The tree creation algorithm avoids nodes with very low QoS and energy resources. All network nodes belong to multiple trees and can send data through multiple paths. Having multiple paths to the sink node, each sensor uses the SAR algorithm for path selection. The SAR algorithm takes into account energy resources, the path's QoS metric, and the priority of the packet to select the optimal routing path to the sink. While the SAR architecture includes QoS in terms of latency, robustness, and reliability, the same architecture suffers from high overhead in maintenance of routing tables.

SPEED [*50*] is a truly unique architecture centered on real-time packet delivery. SPEED differentiates three types of services: unicast (point to point packet delivery), area-multicast (delivery to all nodes within an area), and area-anycast (one node representing the whole area of nodes). The routing itself is a combination of feedback control and non-deterministic geographic forwarding. In other words, SPEED is capable of routing packets through the most optimal path based on prior communication history through various paths. At the same time, SPEED is fully capable of managing the immediate network congestions through the backpressure rerouting scheme. Backpressure rerouting allows SPEED to change the direction of incoming packets if severe congestion is detected. Another QoS feature embedded into SPEED is the Neighborhood Feedback Loop (NFL), which is responsible for maintaining an a priori set single hop relay speed by effectively dropping all backlogged packets if the delivery speed drops below the set point value.
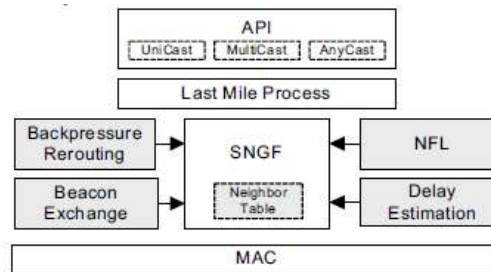

Figure 14. SPEED Protocol

Energy-Aware QoS Routing [*51*] provides QoS aware routing for video and imaging transmission. This architecture finds the least-cost, delay-constrained path for real-time data in terms of link cost that captures nodes' energy reserve, transmission energy and error rate. A novel feature is the capability to prioritize real-time and non-real-time data at sensor nodes. However, the coexistence of real-time and non-real-time data makes the routing problem extremely complex. In addition, this architecture also provides QoS meeting preset end-to-end delay requirements.

Additional QoS architectures are *Reinforcement Learning based MAC (RL-MAC)* [*52*]*, Multipath Multi-SPEED (MMSPEED)* [*53*]*,* and *Distributed Activation based on Predetermined Routes (DAPR)* [*54*].

7. OTHER ARCHITECTURES

Network flow, multi-path, and heterogeneity-based architectures are described in this section.
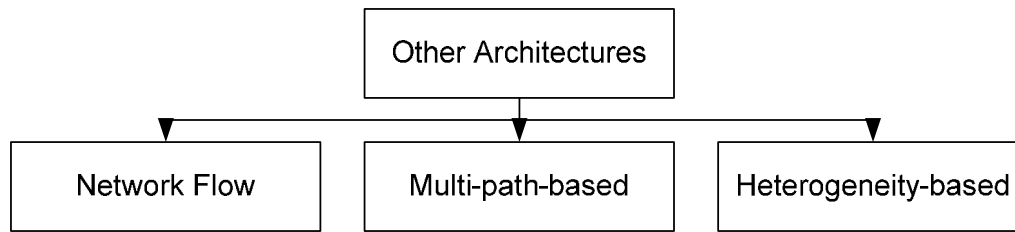

Figure 15. Summary of other architectures

Network flow architectures are defined by the optimization process across node links. Links are defined as node-to-node communication with certain cost parameters such as the power to transfer a bit of data, the latency to transfer a bit of data, or the communication delay between two nodes. Therefore, the goal is to find the optimal routing path across all links between the source and the sink given the predefined link cost metric. The network flow architectures are: *Maximum Lifetime Routing* [55], Maximum *Lifetime Data Gathering and Aggregation* [56], and *Minimum Cost Forwarding* [57].

Multipath-based architectures connect the source node and the sink node via multiple routes (paths). This approach allows data payload to be evenly distributed across the multiple paths. These types of architectures are also popular for real-time streaming data as well as data requiring a special level of reliability. These architectures tend to be power efficient, because they spread the energy load across multiple paths. The most prominent representatives of this type are *Node-Disjoint* and *Braided Path* architectures [58], *N-to-1 Multipath Discovery* [59], *Secure and Energy Efficient Multipath (SEEM)* [60], *Robust and Energy Efficient multipath Routing (REER)* [61], and *Hierarchy-based Multipath Routing Protocol (HMRP)* [62].

Heterogeneity-based architectures imply a network with multiple types of nodes. Nodes within the same network might be split into battery operated nodes vs. power-operated nodes, sensing nodes vs. communication nodes, or nodes with processing power vs. nodes with sensor units. In all cases, the goal is to optimize the network in order to best utilize each node's available resources. *Information-driven Sensor Querying (IDSQ), Constrained Anisotropic Diffusion Routing (CADR)* [63], *Cluster Head Relay (CHR)* [64], *Heterogeneous Disjoint Multipath Routing Protocol (HDMRP)* [65], *Stable Election Protocol (SEP)* [66], and *Energy Efficient Heterogeneous Clustered* scheme*(EEHC)* [67] fall into this category.

8. SUMMARY

This survey presented an overview of WSN architectures. We classified all major architectures into data-centric, hierarchical, location-based, mobility-based, and quality of service. Other architectures such as network flow, multi-path, and heterogeneity-based are also described in this survey. All architectures had in common the ability to extend the network lifetime. The network lifetime is the most relevant performance metric for this survey. All architectures, however, differed in other performance parameters such as packet latency, network security, quality of service, geographical awareness, and data and network centricity. This survey described each architecture separately, pointing out its advantages and disadvantages.

ACKNOWLEDGEMENTS

REFERENCES

[1]   I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, vol. 40, no. 8, pp. 102 -114, August 2002.

[2]   K Akkaya and M. Younis, "A survey on routing protocols for wireles ssensor networks," Elsevier - Ad Hoc Networks, vol. 3, no. 3, pp. 325-349, May 2005.

[3]   Z. Yang and Mohammed A., "A Survey of Routing Protocols of Wireless Sensor Networks," Blekinge Institute of Technology, Sweden, White Paper 2010.

[4]   S.K. Singh, M.P. Singh, and D.K Singh, "Routing Protocols in Wireless Sensor Networks – A Survey," International Journal of Computer Science & Engineering Survey (IJCSES), vol. 1, no. 2, pp. 63-83, November 2010.

[5]   J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," The International Journal of Computer and Telecommunications Networking, vol. 52, no. 12, pp. 2292-2330, August 2008.

[6]   S. Hedetniemi and A. Liestman, "A survey of gossiping and broadcasting in communication networks," Networks, vol. 18, no. 4, pp. 319-349, 1988.

[7]   W.R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in MobiCom '99 Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, 1999, pp. 174-185.

[8]   C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in MobiCom '00 Proceedings of the 6th annual international conference on Mobile computing and networking, 2000, pp. 56 - 67.

[9]   C. Schurgers and M.B. Srivastava, "Energy efficient routing in wireless sensor networks," in MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force, 2001, pp. 357-361 vol.1.

[10]   D. Braginsky and D. Estrin, "Rumor Routing Algorithm for Sensor Networks," in WSNA '02 Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, Atlanta, 2002, pp. 22-31.

[11]   R.C. Shah and J.M. Rabaey, "Energy aware routing for low energy ad hoc sensor networks," in 2002 IEEE WCNC2002. Wireless Communications and Networking Conference, 2002, pp. 350-355 vol.1.

[12]   Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," SIGMOD, vol. 31, no. 3, pp. 9-18, September 2002.

[13]   N. Sadagopan, B. Krishnamachari, and A. Helmy, "The ACQUIRE mechanism for efficient querying in sensor networks," in Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, 2003, pp. 149-155.

[14]   D Liu, X. Hu, and X Jia, "Energy efficient information dissemination protocols by negotiation for wireless sensor networks," Computer Communications, vol. 29, no. 11, pp. 2136-2149, July 2006.

[15]   A. Boukerche, X. Cheng, and J. Linus, "Energy-aware data-centric routing in microsensor networks," in MSWIM '03 Proceedings of the 6th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems, 2003, pp. 42-49.

[16]   J. Liu, Feng Z., and D. Petrovic, "Information-directed routing in ad hoc sensor networks," IEEE Journal on Selected Areas in Communications, vol. 23, no. 4, pp. 851-861, April 2005.

[17]   Y.G. Iyer, S. Gandham, and S. Venkatesan, "STCP: a generic transport layer protocol for wireless sensor networks," in Proceedings. 14th International Conference on Computer Communications and Networks, 2005, pp. 449-454.

[18]   W.B. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," in Proceedings of the 33rd Hawaii International Conference on System Sciences, vol. 1, October 2000, pp. 660 - 670.

[19]   G. Pei and C. Chien, "Low Power TDMA in Large Wireless Sensor Networks," in Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE, 2001, pp. 347 - 351 vol.1.

[20]   M. Garla, T.J. Kwon, and G. Pei, "On demend Routing in Large Ad Hoc Wireless Networks with Passive Clustering," in Proceedings IEEE WCNC 2000, Chicago, IL, 2000.

[21]   O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient Distributed Clustering Approach for Ad Hoc Sensor Networks," in IEEE Transactions on Mobile Computing, 2004, pp. 366-379 Vol.3 No.4.

[22]   S. Lindsey and C.S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in IEEE Aerospace Conference Proceedings, 2002, pp. 3-1125 - 3-1130 vol.3.

[23]   S. Lindsay, C.S. Raghavendra, and K.M. Sivalingam, "Data Gathering in Sensor Networks using the Energy Delay Metric," in IPDPS '01 Proceedings of the 15th International Parallel & Distributed , 2001, p. 188.

[24]   A. Manjeshwar and D.P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," in Proceedings 15th International Parallel and Distributed Processing Symposium, 2000, pp. 2009 - 2015.

[25]   A. Manjeshwar and D.P. Agrawal, "APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in Proceedings International Parallel and Distributed Processing Symposium, 2002, pp. 195 - 202.

[26]   M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," in 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems, 2002, pp. 129 - 136.

[27]   M.A. Youssef, M.F. Younis, and K.A. Arisha, "A constrained shortest-path energy-aware routing algorithm for wireless sensor networks," in IEEE Wireless Communications and Networking Conference, Orlando, FL, 2002, pp. 794-799 vol.2.

[28]   J. Yin and S. Madria, "SecRout: a secure routing protocol for sensor networks," in 20th International Conference on Advanced Information Networking and Applications, 2006, p. vol 1.

[29]   X. Du and F. Lin, "Secure cell relay routing protocol for sensor networks," in 24th IEEE International Performance, Computing, and Communications Conference, 2005, pp. 477 - 482.

[30]   Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for Ad Hoc routing," in Proceedings of the 7th annual international conference on Mobile computing and networking MobiCom, 2001, pp. 70-84.

[31]   B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in Proceedings ACM MobiCom'01, Rome, Italy, 2001, pp. 85-96.

[32]   B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," Wireless Networks, vol. 8, no. 5, pp. 481-494, September 2002.

[33]   Y. Yu, R. Govindan, and D. Estrin, "Geographical and Energy Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks," University of California Los Angeles , Los Angeles, Technical Report No. UCLA/CSD-TR-01-0023, 2001.

[34]   M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Mutlihop performance," IEEE Transactions on mobile Computing, vol. 2, no. 4, pp. 337-348, Oct.-Dec. 2003.

[35]   B. Nath and D. Niculescu, "Routing on a curve," ACM SIGCOMM Computer Communication Review, vol. 33, no. 1, pp. 155-160, January 2003.

[36]   R. Zhang, H. Zhao, and M.A. Labrador, "The anchor location service (ALS) protocol for large-scale wireless sensor networks," in Proceedings of the First International on Integrated Internet Ad hoc and Sensor Networks, Nice, France, 2006.

[37]   G. Xing, C. Lu, R. Pless, and Q. Huang, "On greedy geographic routing algorithms in sensing-covered networks," in Proceedings ACM MobiHoc'04, Tokyo, Japan, 2004, pp. 31-42.

[38]   V. Rodoplu and T.H. Meng, "Minimum energy mobile wireless networks," IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, pp. 1333-1344, August 1999.

[39]   L. Li and J.Y. Halpern, "Minimum-energy mobile wireless networks revisited," in IEEE International Conference on Communications, 2001, pp. 278-283.

[40]   K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari, "Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks," in Proceedings of the Sensys'04, Baltimore, MD, 2004.

[41]   H.S. Kim, T.F. Abdelzaher, and W.H. Kwon, "Minimum-Energy asynchronous dissemination to mobile sinks in wireless sensor networks," in Proceedings of the 1st international conference on Embedded networked sensor systems, 2003, pp. 193-204.

[42]   F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A two-tier data dissemination model for large-scale wireless sensor networks," in Proceedings of the 8th annual international conference on Mobile computing and networking, 2002, pp. 148-159.

[43]   J. Luo and J. P. Hubaux, "Joint mobility and routing for lifetime elongation in wireless sensor networks," in Proceedings IEEE INFOCOM'05, Miami, FL, 2005, pp. 1735-1746.

[44]   R.C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Modeling a three-tier architecture for sparse sensor networks," in Proceedings SNPA '03, Anchorage, AK, 2003, pp. 30-41.

[45]   W. Chang, G. Cao, and T. La Porta, "Dynamic proxy tree-based data dissemination schemes for wireless sensor networks," in Proceedings IEEE MASS'04, Fort Lauderdale, FL, 2004, pp. 21-30.

[46]   M. Ali, T. Suleman, and Z.A. Uzmi, "MMAC: a mobility-adaptive, collision-free MAC protocol for wireless sensor networks," in 24th IEEE International Performance, Computing, and Communications Conference, 2005, pp. 401-407.

[47]   H. Pham and S. Jha, "An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC)," in IEEE International Conference on Mobile Ad-hoc and Sensor Systems, 2004, pp. 558-560.

[48]   P. Xie, J.H. Cui, and L. Lao, "VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks," in Proceedings of IFIP Networking'06, vol. 3967, Coimbra, Portugal, 2006, pp. 1216-1221.

[49]   K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie, "Protocols for self-organization of a wireless sensor network," IEEE Personal Communications, vol. 7, no. 5, pp. 16-27, October 2000.

[50]   T. He, J. Stankovic, and C. Abdelzaher, T. Lu, "SPEED: A stateless protocol for real-time communication in sensor networks," in Proceedings of International Conference on Distributed Computing Systems, Providence, RI, 2003.

[51]   K. Akkaya and M. Younis, "An Energy-Aware QoS Routing Protocol for Wireless Sensor Networks," in Proceedings of the IEEE Workshop on Mobile and Wireless Networks, Providence, RI, 2003.

[52]   Z. Liu and I. Elhanany, "RL-MAC: A QoS-Aware Reinforcement Learning based MAC Protocol for Wireless Sensor Networks," in Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, 2006, pp. 768-773.

[53]   E. Felemban, Chang-Gun Lee, and E. Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks," IEEE Transactions on Mobile Computing, vol. 5, no. 6, pp. 738-754, June 2006.

[54]   M. Perillo and W. Heinzelman, "DAPR: a protocol for wireless sensor networks utilizing an application-based routing cost," IEEE Wireless Communications and Networking Conference, vol. 3, pp. 1540-1545, March 2004.

[55]   J.H. Chang and L Tassiulas, "Maximum lifetime routing in wireless sensor networks," IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 609-619, August 2004.

[56]   K. Kalpakis, K. Dasgupta, and P. Namjoshi, "Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks," Computer Networks, vol. 42, no. 6, pp. 697-716, August 2003.

[57]   F. Ye, A. Chen, S. Lu, and L. Zhang, "A scalable solution to minimum cost forwarding in large sensor networks," in Tenth International Conference on Computer Communications and Networks, 2001, pp. 304-309.

[58]   D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-Resilient, Energy-Effcient Multipath Routing in Wireless Sensor Networks," ACM SIGMOBILE, vol. 5, no. 4, pp. 11-25, October 2001.

[59]   W. Lou, "An efficient N-to-1 multipath routing protocol in wireless sensor networks," in IEEE International Conference on Mobile Adhoc and Sensor Systems, 2005, pp. 672- 8pp.

[60]   N. Nasser and Y. Chen, "SEEM: Secure and energy-efficient multipath routing protocol for wireless sensor networks," Computer Communications, vol. 30, no. 11-12, pp. 2401-2412, September 2007.

[61]   B. Yahya and J. Ben-Othman, "REER: Robust and Energy Efficient Multipath Routing Protocol for Wireless Sensor Networks," in GLOBECOM - Global Telecommunications Conference, 2009, pp. 1-7.

[62]   Y.H. Wang, H.J. Mao, C.H. Tsai, and C.C. Chuang, "HMRP: Hierarchy-Based Multipath Routing Protocol for Wireless Sensor Networks," in Embedded and Ubiquitous Computing - EUC 2005 Workshops, Nagasaki, Japan, 2005, pp. 6-9.

[63]   M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," International Journal of High Performance Computing Applications, vol. 16, no. 3, pp. 293-313, February 2002.

[64]   X. Du and F. Lin, "Improving routing in sensor networks with heterogeneous sensor nodes," in Proceedings IEEE VTC'05, Dallas, TX, 2005, pp. 2528-2532.

[65]   A. Hadjidj, A. Bouabdallah, and Y. Challal, "HDMRP: An Efficient Fault-Tolerant Multipath Routing Protocol for Heterogeneous Wireless Sensor Networks," in 7th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (Qshine), Houston, TX, 2010.

[66]   G. Smaragdakis, I. Matta, and A. Bestavros, "SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks," in International Workshop on SANPA, 2004.
[67]   D. Kumara, T. C. Aserib, and R.B. Patel, "EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks," Computer Communications, vol. 32, no. 4, pp. 662-667, March 2009.

Authors

Mr. Almir Davis received his BS and MS in electrical and computer engineering from Northeastern University, Boston USA and his Ph.D. from Tufts University, Medford USA. Mr. Davis has 12+ years of industry experience resulting in 13 awarded and 5 pending patents as well as numerous published papers in the area of wireless sensor networks. He is the co-founder and the associate director of Tufts Wireless Laboratory. Mr. Davis is recipient of reputable NU President's Award and Sears B. Condit Award for outstanding academic achievements. He is also a member of engineering honor societies Eta Kappa Nu, Phi Kappa Phi, and Tau Beta Pi. Mr. Davis' current research interests include wireless sensor networks, distributed sensing, and low-power electronics.

Professor C. Hwa Chang received his BS from National Cheng-Kung University (NCKU) his Master in CS from Montana State University, and his Ph. D. in Electrical and Computer Engineering from Drexel University. He majored in Architecture and later Engineering Science at NCKU.

Dr. Chang joined Tufts University as a faculty member is 1987. His interests are in the areas of computer architecture, communication protocols, wireless communications and RFID. He is the most senior faculty member of its Computer Engineering Program and the Director of Computer Engineering Program within the Department of Electrical and Computer Engineering. He has prepared and went through four times of ABET Accreditation and plays a major role in the 1999, 2006 and 2012 ABET accreditation and visits.

Professor Chang's research areas include Wireless Communications, Parallel Processing, Computer Networking, Distributed Computing, Multimedia Networks, Simulation, Programmable Logic Design, and Engineering Education. He has graduated four Ph. D. students and about 120 Master students. Presently his wireless research group has eight graduate students including three Ph. Ds working on various aspects of wireless communications. He is interested in wireless sensor networks. He founded the Tufts Wireless Lab at Tufts University.