Energy Aware and Cluster-based Routing Protocols for Large-Scale Ambient Sensor Networks*

Anahit Martirosyan, Azzedine Boukerche and Richard W. Nelem Pazzi
PARADISE Research Laboratory
SITE - University of Ottawa, Canada
{amart013, boukerch, rwerner}@site.uottawa.ca

ABSTRACT

The Ambient Intelligence paradigm is built upon Ubiquitous Computing (UC), in which the computing devices are embedded in the environment with the purpose of enhancing the human experience at home, workplace/office, learning, health care etc. The UC applications aim at providing services to the users anywhere, anytime in an unobtrusive, seemingly invisible way. Wireless Sensor Networks (WSNs) have great potential for UC applications and are envisioned to revolutionize them. However, before WSNs become a commodity, several issues remain to be resolved. Among the major concerns of designing WSN for UC applications is the design of efficient energy-aware routing protocols. A large number of routing protocols has been proposed in the literature. They basically differ depending on the application and network architecture used in their design. In this paper, we present clustering energy-aware routing protocols for WSNs and highlight their features.

1. INTRODUCTION

Recent advances in micro-sensor development have resulted in the wide use of Wireless Sensor Networks (WSNs) for remotely monitoring tasks [1]. WSNs are used for physical environment monitoring, security surveillance, military applications, among others. WSNs have also a great potential for Ubiquitous Computing (UC) applications (such as pervasive health monitoring systems, smart homes etc.) and are envisioned to revolutionize them. UC applications aim at providing services to the users anywhere, anytime in an unobtrusive, seemingly invisible way. WSNs are composed of a large number of nodes that work together to accomplish a sensing task. Sensors communicate with each other to relay messages from the network to the Sink, which is the entity interested in monitoring the subject of interest.

*This work is partially sponsored by Grants from the NSERC, Canada Research Chairs Program, ORNEC, the Ontario Distinguished Researcher Award and the EAR Award.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
AMBI-SYS 2008, February 11-13, Quebec, Canada
Copyright © 2008 ICST 978-963-9799-16-5
DOI 10.4108/ICST.AMBISYS2008.2904

A large number of routing protocols for WSNs has been developed recently [2]. Due to the limited energy resources of sensor nodes, designing efficient energy-aware routing protocols has been one of the most challenging issues for WSNs. In recent years, clustering routing protocols have been developed in order to reduce the network traffic toward the Sink. Moreover, cluster heads have been used to enhance the efficiency of the energy-aware routing protocols. While clustering may introduce overhead due to the cluster configuration and their maintenance, earlier work has demonstrated that cluster-based protocols exhibit better energy consumption and performance when compared to flat network topologies for large-scale WSNs.

2. ENERGY-AWARE CLUSTERING ROUTING PROTOCOLS FOR WIRELESS SENSOR NETWORKS

In this section, we will present selected clustering routing protocols for WSNs.

2.1 Low-Energy Adaptive Clustering Hierarchy Protocol

The Low-Energy Adaptive Clustering Hierarchy protocol (LEACH) [6] is a well-known hierarchical routing protocol and it was one of the pioneering clustering approaches in the literature for WSNs. Basically, LEACH is a cluster-based protocol that utilizes the randomized rotation of cluster-heads to evenly distribute the energy load among the sensors in the network. In the LEACH protocol, the sensor nodes organize themselves into local clusters, with one node acting as the cluster head (CH). The LEACH mechanism includes the randomized rotation of the CH function among the sensor nodes in order to not drain the battery of a single node. It also performs local data aggregation in the CHs to reduce the amount of data being sent from the clusters to the Sink, which further reduces energy dissipation and enhances the network lifetime. The sensor nodes elect themselves to be CHs at any given time with a certain probability. These CH nodes broadcast their status to the other sensors in the network. Each sensor node determines which cluster it wants to belong to by choosing the cluster head that requires the minimum communication energy. Once all the nodes are organized into clusters, each CH maintains a schedule for the nodes in its cluster. This allows the radio components of each non-cluster-head node to be turned off at all times except during its transmit time, thus minimizing the energy dissipated in the individual sensors. The CH nodes are also

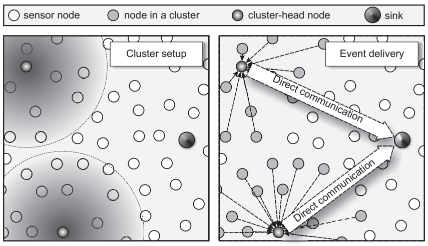

Figure 1: The LEACH protocol.

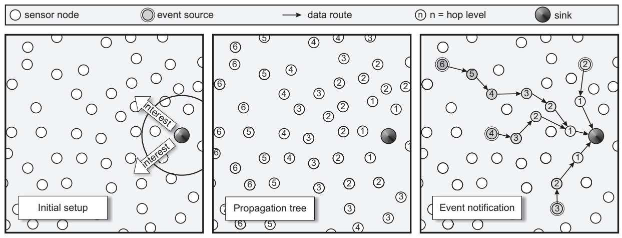

Figure 2: The steps of the PEQ algorithm

aggregator nodes.

According to the simulation results presented in [6], the sensor nodes reach energy depletion randomly and the dynamic clustering increases the network lifetime. The LEACH protocol is completely distributed and requires no global knowledge of network. However, it uses one-hop communications from sensor nodes to a CH and from a CH to the Sink, as exemplified in Figure 1. Therefore, the LEACH mechanism faces scalability problems if applied to large networks due to the long distance communications in order to relay the sensed data.

2.2 Threshold Sensitive Energy Efficient Sensor Network protocol

Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [8], is a hybrid of hierarchical clustering and data-centric protocols designed for time-critical applications. It is a responsive protocol to sudden changes of some of the attributes observed in the WSN (e.g., temperature). The algorithm first goes through cluster formation. The CHs then broadcast two thresholds to the nodes. Those are hard and soft thresholds for the sensed attributes. Hard threshold is the minimum possible value of the attribute, which will trigger the node to turn the radio on and transmit to the CH. Using the threshold will result in reducing the number of transmissions and thus saving energy. After the attribute's value reached the hard threshold, the node will transmit again only when the attribute's value changes by the soft threshold. By adjusting hard and soft thresholds, it is possible to achieve energy saving by decreasing the number of transmissions. However, TEEN cannot be applied for sensor networks where periodic sensor readings should be delivered to the Sink, since the values of the attributes may not reach the threshold at all. Another limitation is that the message propagation is accomplished by CHs only. If CHs are not in each other's transmission radius, the messages will be lost.

Adaptive threshold sensitive Energy Efficient sensor Network protocol (APTEEN) [9] is an extension of TEEN protocol. It can be used for both periodic and responsive data collection. The disadvantage of the two approaches is overhead and complexity of forming clusters. The Sink is engaged in forming the clusters, which obviously creates a lot of network traffic. Another drawback is maintaining the threshold-based functions.

2.3 Geographic Adaptive Fidelity Protocol

One of the location-based protocols is Geographic Adaptive Fidelity (GAF) [7], which was designed primarily for mobile ad hoc networks and can be used in WSNs as well. The idea of the protocol is to conserve energy by using sleeping mode for some of the nodes without affecting the connectivity of the network. The method forms a virtual grid for the nodes in the network and decides which nodes in the same grid (geographical area) are equivalent in terms of possibility to communicate with the nodes in the neighboring grid. GAF can achieve significant energy savings due to using sleeping mode to those nodes without affecting routing fidelity, especially when the network size grows. Although GAF is a location-based protocol, it resembles hierarchical protocols, in which the clusters are associated with the geographic location. In GAF, each node in the network uses GPS for indicating their position. However, GPS is an energy-expensive and costly tool to be combined with an energy-scarce sensor node. It also requires a line of sight.

2.4 Periodic, Event-driven and Query-based Routing Protocol (PEQ) for Wireless Sensor Networks

The PEQ protocol [3] consists of three steps: the construction of the hop tree, i.e., the dissemination tree; the propagation of subscriptions; and the data delivery to the Sink. The configuration of the hop tree is started by the Sink node by broadcasting a packet that contains a hop value, a time-stamp, and the source address. Each node will store, increment and transmit the hop level to its neighbor nodes. This process continues until the entire network is configured with hop levels. Each node will also learn and store the source address in a routing table that will be used later on to forward data to the Sink. This mechanism is depicted in Figure 2(a) and (b).

The protocol employs the Publish / Subscribe paradigm introduced by Eugster et. al. [5]. In the paradigm, the Sink subscribes to the data it wants. In the PEQ protocol, when a sensor node detects a phenomenon that matches the Sink's interests, the node sends the event packet to the forwarding address (learned during the flooding). After the neighbor node has received the packet, it performs the same algorithm and sends the data to its forwarding neighbor. This process continues until the data reaches the Sink. The mechanism is shown in Figure 2(c).

The PEQ protocol also implements a simple acknowledgement based repair mechanism. Each sensor node sets a timeout timer after it has forwarded a packet and waits for the neighbor's ACK. If no ACK is received, the node infers that the neighbor node is dead. In this case, the node broadcasts a SEARCH packet in order to find a new forwarding node. The neighbor nodes will reply with a packet that contains their hop level, and address. Then, the node elects the neighbor with the lowest hop level as the new forwarding node.

2.5 A Cluster-Based Approach: CPEQ

The CPEQ protocol [3] is based on the PEQ mechanisms and it adopts a cluster-based approach in which the nodes

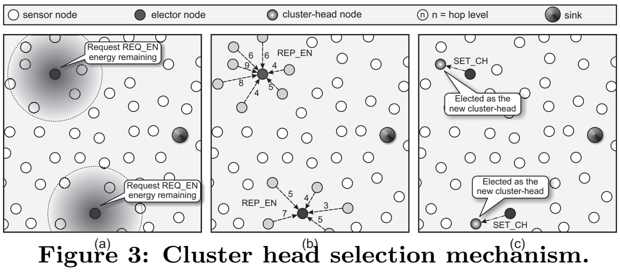

Figure 3: Cluster head selection mechanism.

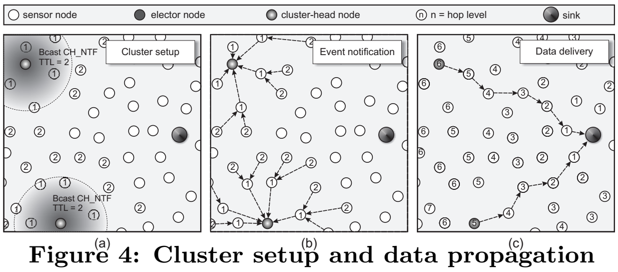

Figure 4: Cluster setup and data propagation

with more residual energy are selected as cluster head (CH) nodes. The CH node builds up a cluster and the nodes within its cluster send their events to the CH. Thereafter, the CH executes an aggregation function (e.g., the averaged temperature) or conceals redundant data, before relaying the data to the Sink.

The CH selection scheme is based on the idea presented in LEACH [6]. Any node in the network can become a CH with a certain probability p. Each node generates a random number and if it is less than the probability p, the node will initiate the CH selection mechanism. First, the "elector" node requests the energy level from its neighbors, as shown in Figure 3(a). After receiving the reply messages from the neighbors, the elector node selects the neighbor with more residual energy as the new cluster. A node remains in the CH state for a specific period of time until it expires. The CH selection scheme is executed periodically.

After the selection, each CH will build its cluster, as seen in Figure 4(a). The mechanism is the same as the initial configuration algorithm of PEQ. In order to limit the size of a cluster, a time-to-live is used. Thereafter, the data delivery scheme is the same employed by PEQ, in which nodes use their forwarding addresses that were learned during the configuration step, as depicted in Figure 4(b).

2.6 An Energy Efficient Inter-Cluster Communication based Routing Protocol for WSNs

The Energy Efficient Inter-Cluster Communication based Routing Protocol for WSNs (ICE) [4] also employs the Publish /Subscribe paradigm. In the setup phase, first cluster head nodes are selected and clusters are formed. The protocol adopts the idea presented in [6] based on the probability for each node in the network to become a cluster head(CH) node and selects 5% CH nodes. Next, neighboring clusters are discovered by means of the beacon nodes (these are the nodes that were used for selecting CH nodes). The beacon nodes broadcast the coordinates of the discovered CHs. Once a CH receives this information, it learns of its neighboring clusters and builds nearest neighbor (NN) tables to these clusters. To find NNs, every CH identified by the beacon node draws a logical line between itself and the other CHs and projects the nodes in its cluster to that line (e.g., nodes A and C in Figure 5). Thus, two neighboring clusters communicate with each other by means of nearest neighbor nodes as it ensures the shortest transmissions between two neighboring clusters. Then, in its table, the CH stores the first two NNs (i.e., the first NN and the second NN) to Cluster 2 (nodes A and B in Figure 5). This information is used by the CHs for routing as well as providing energy efficiency. The neighbors responsible for inter-cluster communication will be active in turns (e.g., nodes A and B in Cluster 1). Every CH will stay in that role for a certain period of time. In other words, the algorithm goes through rounds in order to cope with the dynamics of the network. Last, free nodes (the nodes that do not belong to any cluster) are discovered.

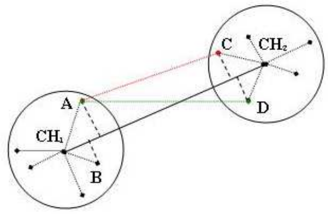

Figure 5: Finding nearest neighbors between clusters.

When a Sink sends its subscription message to the network, the message's propagation is accomplished by using the constructed clusters. Once the subscription message leaves a cluster, the nodes that are not NN nodes (i.e., they do not have the responsibility of communicating with neighboring clusters) and one node from the pair of NN nodes will go to sleep. The CH will broadcast a TDMA schedule for these nodes. The nodes will stay asleep unless they sense an event or wake up according to the TDMA schedule.

When a cluster node hears a message from the Sink, it forwards the subscription message to its CH. The CH has information about all of its nodes' coordinates. It identifies whether the subscription is intended for any of its nodes and any free node(s) that it knows about. If it is, the CH will either notify the cluster node or use its nodes to inform the free node about the subscription respectively.

Every time when a message leaves a cluster, a timeout is set up. If the sending node does not receive an ACK message from a node in the destination cluster, the algorithm uses intermediate nodes for the further relaying of the message. This can happen due to insufficient transmission radius. The message is then retransmitted with the flag "intermediate = ON" in its header. Every message going out from a cluster has its sender's and destination's *cluster IDs* (coordinates of the corresponding CH) as shown in Figure 6.

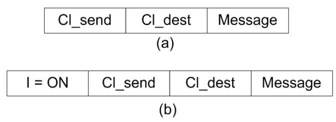

Figure 6: The format of interest subscription and event notification messages when using (a) nearest neighbors and (b) intermediate nodes.

When a message is relayed from $Cluster_i$ to $Cluster_j$, the NN node in $Cluster_j$ hears the message (the header contains the information about the sender) and relays it to its CH. The subscription message will be relayed further to neighboring clusters. The CH node takes care of relaying the message further to the neighboring clusters in a similar way. The CHs store information about every subscription message that passes them. It consists of the *eventID, SinkID* and the *clusterID*(s) of the clusters from which the message has arrived at the current cluster.

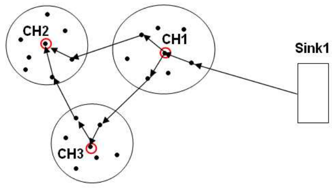

Figure 7: Interest subscription's propagation through a network.

In Figure 7, a subscription message is sent by Sink 1. It reaches CH2 node via two different routes: through Cluster 1 and Cluster 3. Then, the CH2's Table of Subscriptions will contain the information about the subscription message's *SinkID* and the cluster(s) from which the message has been received (1).

Table 1: Table of Subscriptions kept at CH2.

EventID	SinkID	Sender_clusterID
	Sink 1	Cluster 1, Cluster 3

As seen from the above, during the subscription propagation phase, CHs store the *senderID* of the cluster that has forwarded a subscription message to the current cluster. Once the event occurs, the CH assigns the node that is responsible for communicating to that cluster to forward the event notification message toward the Sink. Once the neighboring cluster's node receives that message, it will in turn forward it to its CH. By checking the *SinkID* field in the message, this CH will know to which cluster to forward the message so that it eventually reaches the Sink. The algorithm ensures that at any given point in time there is an active node from the NN table that is responsible for inter-cluster communication. A special case occurs when a free node senses an event. For sending the event notification, the node will use the reverse path to the path traversed by the subscription message.

When the beacon nodes explore the neighboring clusters, the network is divided into islands of clusters as shown in Figure 8. The overlapping CHs provide network connectivity between the islands of clusters. When there are no overlapping CHs in two islands (e.g., the beacon nodes B2 and B6 in the figure did not find overlapping CHs), communication is achieved by using intermediate nodes and messages with "intermediate" flag as described above.

To cope with the varying delays on alternative paths, the algorithm provides QoS in that it associates a cost with each of the paths and uses a path with the least cost for high-priority event notification messages. The cost of a path is defined as $C = Count_p/E_{res}$, where $Count_p$ is the hop count of the path p and E_{res} is the path's residual energy.

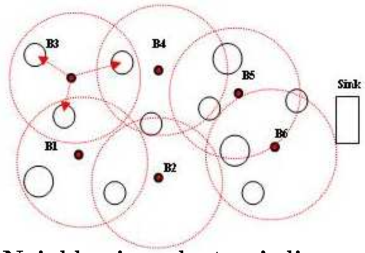

Figure 8: Neighboring clusters' discovery by using beacon nodes.

3. CONCLUSION

WSNs have been used for remotely monitoring tasks. They are being increasingly used for Ubiquitous Computing applications as well. The limited energy resources of sensor nodes pose challenging issues on the development of routing protocols for WSNs. Introducing clustering into the networks' topology has the goal of reducing the number of messages that need to be delivered from to the Sink in large-scale WSNs. It also provides energy efficiency as cluster heads maintain a schedule for low duty cycling of the cluster nodes. In this paper, we presented selected clustering protocols for WSNs and highlighted their features. The scalability limitation of an early clustering protocol LEACH has been addressed by the later clustering protocols. Multi-hop data propagation is employed by either using only cluster heads as relay nodes (such as TEEN and APTEEN) or the cluster heads, cluster nodes and the free nodes (such as CPEQ and ICE). Having the nodes other than the cluster heads participate in the message delivery has the objective of reducing the load on the cluster head nodes thus preserving their energy and prolonging the network lifetime.

4. REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless Sensor Networks: A Survey, Computer Networks, 2002, 393-422.

[2] K. Akkaya, M. Younis, A survey of routing protocols in wireless sensor networks, Elsevier Ad Hoc Network 3/3, 2005, 325-349.

[3] A. Boukerche, R. W. Pazzi, and R.B Araujo, Fault-tolerant wireless sensor network routing protocols for the supervision of context-aware physical environments, Journal of Parallel and Distributed Computing, Volume 66, Issue 4, Algorithms for Wireless and Ad-Hoc Networks, April 2006, 586-599.

[4] A. Boukerche and A. Martirosyan, An Energy-Aware and Fault Tolerant Inter-cluster Communication based Protocol for Wireless Sensor Networks, Globecom, Washington D.C., November 2007.

[5] P.T. Eugster, P.A. Felber, R. Guerraoui, and A.M. Kermarrec, The many faces of publish/subscribe. ACM Computing Surveys, 35(2), 114 - 131, June 2003.

[6] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, Energy-efficient communication protocols for wireless sensor networks. In Proceedings of the 33rd Annual Hawaii Int'l Conference on System Sciences (HICSS), 3005–3014, Big Island, Hawaii, USA, January 2000.

[7] Y. Xu, J. Heidemann, and D. Estrin, Geography-informed energy conservation for ad hoc routing, ACM MobiCom, 2001, 70-84.

[8] A. Manjeshwar and D. P. Agrawal, TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks, 1st Int'l Workshop on Parallel and Distr. Computing Issues in Wireless Networks and Mobile Computing, 2001, p.189.

[9] A. Manjeshwar and D. P. Agrawal, APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks, 2nd Int'l Workshop on Parallel and Distr. Computing Issues in Wireless Networks and Mobile Computing, 2002, 195-202.