WCDS-DCR: An Energy-Efficient Data-Centric Routing Scheme for Wireless Sensor Networks

Nesrine Chakchouk, Bechir Hamdaoui, and Mounir Frikha[†]

School of EECS, Oregon State University
[†] Higher School of Communications, University of 7 November at Carthage, Tunisia
chakchon, hamdaoui@eecs.oregonstate.edu, m.frikha@supcom.rnu.tn

***Abstract*— Wireless sensor networks (WSNs) consist of battery-constrained sensors often deployed in harsh environments with little to no human control, thereby necessitating scalable and energy-efficient techniques. For this reason, self-organizing and maintenance mechanisms are very appealing to the design of WSNs. In this paper, we propose a routing scheme, called WCDS-DCR, that meets these design requirements. WCDS-DCR is a fully distributed, data-centric, routing technique that makes use of an underlying clustering structure induced by the construction of WCDS (Weakly Connected Dominating Set) to prolong network lifetime. It aims at extending network lifetime through the use of data aggregation (based on the elimination of redundant data packets) by some particular nodes. It also utilizes both the energy availability information and the distances (in number of hops) from sensors to the sink in order to make hop-by-hop, energy-aware, routing decisions. Simulation results show that our solution is scalable, and outperforms existing schemes in terms of network lifetime.**

Index Terms—Sensor networks, clustering, energy-aware routing, data aggregation.

I. INTRODUCTION

Recent advances in wireless technology as well as those in electronics enabled low-cost, low-data-rate, small, communications devices with various sensing capabilities, thereby making wireless sensor networks (WSNs) both possible and successful. WSNs are enabling a giant leap over current distributed control system paradigms, paving the way for large-scale, ubiquitous computing applications, ranging from environmental monitoring to building automation. WSNs are comprised of large numbers of nodes often in the order of thousands to million.

In a WSN, nodes collaborate to carry out certain information processing tasks. Collaboration among nodes typically occurs in the form of information fusion, where sensor nodes in the same geographic vicinity each performs desired measurements, processes the measured data, and transmits the data over a wireless channel to a base station, commonly referred to as the *sink*. The sink collects the data from all the nearby nodes, analyzes and fuses the information, then sends the fused data to a decision center/network, which uses it, along with others, to arrive at the necessary consensus decision.

By their design nature, sensor nodes have limited resources in terms of processing capability, storage space, and power. Because nodes are typically battery powered, and not accessible so as their batteries can be replaced, power consumption is of a paramount importance to WSNs as it is crucial to their operational longevity. Most of the node's energy resources are dissipated by its communication radio, and are primarily spent in handling main network tasks, such as synchronization, packet transmissions, and channel sensing [1]. It is therefore crucial that communication techniques for WSNs be designed with energy awareness so as to prolong network lifetime as much as possible.

As a result, there have been numerous studies on energy-aware techniques that address MAC- and network-related issues for WSNs. Most of the reported techniques do not consider cross-layer effects; they focus on issues that pertain to either MAC- or network-layer, but decoupled from one another. However, it is important that these techniques account for cross-layer coupling effects. Most of the functionalities supported at MAC- and network-layers perform more effectively when optimized jointly. This is especially the case in hierarchical networks, where network structure/organization has an impact on various functionalities at different layers, such as connectivity, synchronization, and routing.

In this paper, we develop an energy-efficient, data-centric, routing scheme for hierarchical WSNs. The proposed routing scheme uses our previously developed MAC protocol [2] as its underlaying MAC, a protocol that relies on the Weakly Connected Dominating Set (WCDS) to define its clustering structure. The proposed routing scheme, hereafter referred to as WCDS-DCR, takes advantage of WCDS structure to balance the overall energy consumption among nodes, thus prolonging network lifetime. WCDS-DCR aggregates sensed data by eliminating redundant information, thus reducing energy consumption and network congestion. WCDS-DCR relies on a distributed energy-aware mechanism that takes into account the nodes' remaining energy when making routing decisions, enabling a more balanced communication load among all nodes. The communication burden is shared among all nodes instead of a set of nodes only, which typically occurs under static routing methods. Static routes are inappropriate for WSNs since they lead to a quick energy depletion of some nodes, which in turn leads to fast network disconnection.

The remainder of this paper is organized as follows. Section II presents related works. Section III describes the system model and the motivation of our work. Section IV presents the detailed design of the proposed routing scheme. Section V evaluates the performance of the proposed scheme via simulations, and compares it with existing ones. Section VI concludes the paper.

II. RELATED WORK

In this section, we will briefly overview some related work on energy-aware routing for WSNs. The focus, however, will be on hierarchical, data-centric routing for clustered WSNs.

Heinzelman *et al.* [3] proposed a classic, cluster-based algorithm for sensor networks, called LEACH (Low-Energy Adaptive Clustering Hierarchy). In LEACH, cluster heads (CHs) are randomly chosen among all sensor nodes on a per-round basis. Nodes that are chosen once to be CHs are not allowed to perform the role of CH again unless all the other nodes perform it during one or more of the successive rounds. A sensor node that does not become a CH selects the closest CH to be the neighbor through which it communicates with the sink. Although LEACH is completely distributed and simple, its CH formation mechanism requires that all communications from sensor nodes to CHs as well as those from CHs to the sink be single hop, thus making it not too scalable.

In [4], the authors proposed an improvement to LEACH, called Threshold-Sensitive Energy-Efficient sensor Network (TEEN) protocol, which basically extends LEACH to address energy efficiency. Unlike LEACH, TEEN allows nodes to be switched off (to save power) whenever there is no data to be reported. TEEN introduces two thresholds: hard and soft. The hard threshold is the minimum value of the sensed attribute, and is what triggers the sensor to turn on its radio and report its sensed data. The soft threshold determines the minimum variation of the sensed attribute that could result in reporting this information to the sink. Moreover TEEN introduces multi-level CHs (CHs and Super CHs) to reduce the cost induced by long-distance transmissions in LEACH. The main drawback of this approach is that it introduces some additional overhead and complexity while forming clusters in multiple levels, and implementing threshold-based functions.

One common issue with both TEEN and LEACH is that sensor nodes that are located far away from their CHs are very susceptible to excessive energy consumption when transmitting packets to their CHs. To overcome this shortcoming, Lindsey and Raghavendra [5] proposed an algorithm, called Power-Efficient Gathering in Sensor Information Systems (PEGASIS), which addresses the above-mentioned issue through the construction of chain structures rather than multiple clusters. PEGASIS is shown to outperform LEACH in terms of energy consumption, especially for sparse networks where the distances between the sink and the other nodes are large.

PEGASIS is, however, more complex than each of the other two approaches as it requires that each sensor node acquire global knowledge of the network. In addition to its complexity, PEGASIS introduces excessive delays for distant nodes on the chain. Moreover, since all nodes in a chain transmit information to a single leader that will later aggregate the data packets and report them to the sink, this leader can easily become a bottleneck.

HEED (Hybrid Energy Efficient Distributed) is another cluster-based algorithm proposed in [6] to enhance LEACH's clustering structure. Unlike LEACH, HEED balances the distribution of CHs across all nodes. Basically, HEED proposed a new cluster formation scheme in which CHs are chosen periodically based on a hybrid of node's residual energy and topological parameters, such as node degree. HEED is shown to achieve an uniform CH distribution across the network, but not without incurring more overhead in terms of both energy consumption and control traffic due to the many iterations that are needed to build those balanced clusters.

The authors in [7] proposed EECS (Energy Efficient Clustering Scheme) for wireless sensor networks, which basically ensures balanced CH distribution with minimum overhead. Unlike HEED, EECS is distributed, and requires a fewer number of iterations. Despite of its features, EECS uses the single-hop communication mode, which still makes it consume large amounts of energy.

Unlike previous approaches, our proposed scheme, WCDS-DCR, addresses both energy consumption and load balancing problems by relying on an optimized MAC [2] designed for self-organizing multichannel WSNs. Our proposed scheme is based on multi-hop rather than single-hop routing, and it is built upon a WCDS-clustering structure which induces a minimal number of CHs that ensures global connectivity within the network.

III. NETWORK MODEL AND PROBLEM STATEMENT

A WSN is modeled as an undirected graph $G = (V, E)$, where V is the set of all nodes in the network, and E is the set of all possible links between pairs of nodes. Nodes are generated and placed randomly in a grid to form a mesh-like network. Connections between neighboring nodes are established through the broadcast of ”Hello” messages, which helps every node obtain a local knowledge of network. Once the network topology is fixed, the WCDS (weakly connected dominating set) component is then constructed progressively in a distributed way by means of the WCDS distributed heuristic [2]. WCDS is the set of nodes such that each node in the network either belongs to WCDS or is adjacent to a node that belongs to WCDS. Such a property helps divide the network into clusters and maintain global connectivity. Nodes that belong to WCDS are called cluster head (CH) nodes; the other nodes are called non cluster head (non CH) nodes. The WCDS construction is made such that there are no adjacent CH nodes, and for each obtained cluster (consisted of the CH and its neighbors), there is at least one non CH node, called bridge node, that belongs to more than one cluster at a time, thus ensuring network connectivity. Each cluster in the network uses a different logical channel: a combination of a time slot and an FHSS sequence for intra cluster communication. A bridge node uses as many logical channels as the number of intersecting clusters it belongs to in order to ensure inter-cluster communication. More details about the construction of WCDS can be found in [2].

In this work, we do not consider mobility, and instead, we assume a static network consisting of one base station (sink) and many sensor nodes. We also assume that nodes are location-unaware (i.e., nodes are not equipped with a GPS or any positioning system). We further assume that nodes can vary their transmission ranges by varying their transmission powers[1], and that each node is capable of using multiple physical channels and FHSS processing[2].

The objective of this work is to design a distributed routing scheme that improves energy consumption, scalability, and lifetime of WSNs. The proposed routing scheme, which will be referred to as WCDS-DCR, is built on top of our recently proposed energy-aware MAC [2],

---

[1] Feasible with Berkeley motes [8]

[2] Feasible with Coronis Motes [9]

and hence, it takes advantage of the MAC's optimized structure to reduce routing overhead and complexity, thus improving energy consumption, scalability, and network lifetime.

IV. ENERGY-EFFICIENT DATA CENTRIC ROUTING

We will first begin by providing a brief background on the underlaying MAC protocol, and then present our proposed energy-efficient, data centric routing scheme.

A. *Medium Access Control*

For completeness, we begin by providing a brief overview of our previously proposed MAC [2] since the proposed routing scheme uses it as its medium access control protocol (for more details, please refer to [2]). Hereafter, this MAC will be referred to as D-MAC.

After the neighbor discovery phase, D-MAC starts organizing the network through a fully distributed heuristic that constructs the WCDS (weakly connected dominating set) component over the network. More specifically, D-MAC divides the nodes into two sets: a WCDS whose nodes are referred to as cluster heads (CHs), and the rest of the nodes, referred to as non CHs. D-MAC aims at minimizing the number of clusters, and consequently, at reducing the synchronization cost as illustrated in [2]. This division process is based on a coloring algorithm that assigns different colors to nodes. It uses four colors, each of which is assigned a fixed weight and corresponds to a different node status: black for CH node status, gray for non CH node status, white for non assigned node status, and red for CH candidate node status. Initially, all the nodes of the network are colored white, except the sink which colors itself black (i.e., a CH). The sink declares then its color to its neighbors through a broadcast message, forcing them to become gray (i.e., non CH nodes). These gray nodes in turn broadcast a color declaration message to their respective neighbors. Among these neighbors, only those which are still colored white become colored red. The red color is an intermediary color which means that a node colored red is a candidate to become a CH node (i.e., could become a CH, and then colors itself black). Such decision or transformation of becoming a CH node is based on the aggregate color weight of the red node's neighbors. Indeed, among the red nodes, the one with the highest aggregate color weight value is chosen to be a CH. This elected CH colors itself black, announces its color to its neighbors, which become gray (i.e., non CH). This process repeats until all nodes in the network are either black (i.e., CHs) or gray (i.e., non CHs).

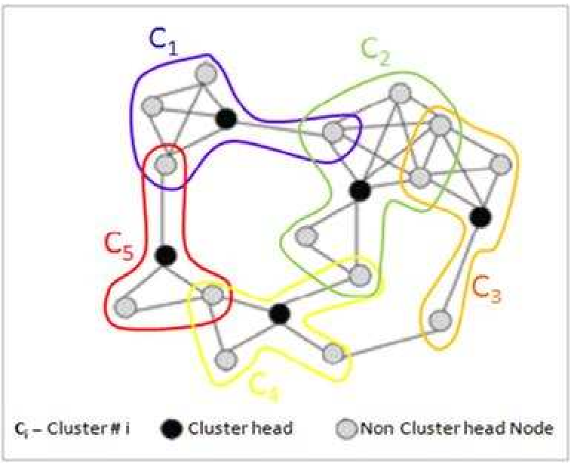


Fig. 1. CH and cluster formation via WCDS

Every CH and its neighbors (non CHs) constitute a cluster. Each cluster uses its own logical channel, consisting of an FHSS sequence and a Time Slot, for intra-cluster communication. We assume that for each cluster, the logical channel is derived from the CH's MAC address which is unique so that neighboring clusters use different logical channels. For this reason, each non CH node belonging to more than one cluster uses as many logical channels as intersecting clusters it belongs to in order to enable inter-cluster communication, thus ensuring network connectivity—these nodes are called Bridge nodes. Fig. 1 shows an example of a sensor network with 5 clusters.

Having defined the topological and the physical structure of our WSN, we will now present the logical organization. In other words, we will state the different node synchronization roles that help maintain connectivity as well as tight synchronization across all the network. We define three synchronization roles:

- **The Referencing role:** It is handled by the CH nodes and consists of periodically broadcasting an announcement message by the CH to maintain the logical channel that it shares with its neighbors. Such a periodic transmission is required to keep intra-cluster tight synchronization, and to overcome the clock drift problem arising from the intrinsic imperfections of the sensors themselves.

- **The Following role:** This is the role of non-CH nodes, which consists of receiving the periodic channel announcement/maintenance messages. The Referencing/Following roles here resemble the famous Master/Slave model used in communication theory.
- **The Sampling role:** This is played by both CH and non-CH nodes. Every check interval, each node senses physical channels for possible packet receptions. While non-CHs belonging to more than one cluster (Bridge nodes) sample multiple channels—those channels that are assigned to all clusters they belong to, all other nodes each samples one channel only.

This synchronization allows each node to know the active schedule in real-time as well as the FHSS sequence of each of its synchronized neighbors. Prior to transmitting a data packet to its neighbor, a node must send a preamble to ensure that the destination node receives the packet as soon as it wakes up. As a result of this established tight synchronization, WCDS-DCR requires that source nodes send short MAC preambles [10] only instead of long ones as in the case of traditional approaches, resulting in greater energy savings [2].

B. The Proposed Routing Scheme: WCDS-DCR

We will now present our proposed energy-efficient, data centric routing scheme: WCDS-DCR. We will first introduce the preparation phase that occurs at the beginning (i.e., after the network organization initiated by the underlying D-MAC is done). Then, we will present the details of the routing process itself, and show how the obtained network structure helps establish our proposed routing technique.

1) Preparation Phase: After building the WCDS component and allocating the logical channels to the different clusters [2], each node executes a preparation phase which consists of three steps: Bellman-Ford tree construction, father-son association, and routing table construction. Upon execution of these steps, nodes can then use the thus-obtained structures for routing and forwarding their packets to the sink. We now present each of these three steps.

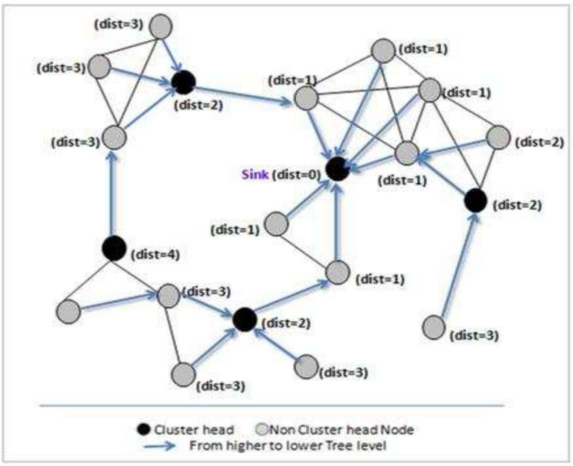


Fig. 2. Distributed Bellman-Ford tree for wireless sensor networks

a) Bellman-Ford tree construction: The construction of the Bellman-Ford tree is initiated by the sink, and is done distributively as follows. First, the sink broadcasts a packet that contains a ”distance from sink” attribute, which is initially set to zero. Each node receiving this packet increments this value by one, stores it in its structure, and then broadcasts the updated packet (containing the incremented value of ”distance from sink”) to all of its neighbors. These neighbors will in turn do the same thing: increment the value of the attribute by one, store it, update the packet, and rebroadcast it again. During this process, a node may receive two conflicting values of ”distance from sink”. When this happens, the node retains the minimum received value. This process ends when each node is assigned a ”distance from sink” that is different from zero. The ”distance from sink” attribute represents the minimum number of hops that separates it from the sink, and will be used later for making routing decisions. A network example is provided in Fig. 2 for illustration.

b) Father-son association: The clustering structure on which we will build our routing scheme is such that each cluster uses a different logical channel, and the nodes that belong to more than one cluster follow more than one CH so as to maintain global connectivity. This is taken care of by D-MAC. The network layer, on the other hand, requires that each non-CH node has a unique principal Reference or Father (i.e., CH). This is ensured via this distributed process, which occurs just after building the WCDS component and the tree: non CH nodes, also called followers, which have only one CH neighbor, take this CH as their main parent. On the other hand, each non CH node that has more than one CH neighbor selects among its CH neighbors those that have the least number of ”children”, then chooses the one with the smallest ID (for uniqueness).

c) Routing table construction: At the end of the first two preparation steps, each node acquires local, topology knowledge and information, such as its CH, its neighbor list, and its neighbors' distances to sink, that enable it to construct its routing table. This table contains the set of its neighbors ordered from nearest to farthest to the sink, their distance from the sink, and their remaining energies.

2) Energy-Efficient Data-Centric Routing: WCDS-DCR is a reactive, i.e. event driven, scheme. That is, data report or transmission to the sink occurs only when an event takes place in a certain area, and when this happens, only the nodes belonging to that area are in charge of delivering sensed data. Below are the details of WCDS-DCR.

a) Data aggregation: Whenever an event occurs in a given area ”X” of the network (e.g., temperature increases over a certain threshold), all nodes belonging to that area generate and send packets to the sink to report that event. Because it suffices that one among all nodes within a cluster report the sensed information, and in order to save energy resources, our proposed routing scheme relies on its WCDS structure to aggregate data, thus reducing both the consumed energy and the amount of generated traffic.

In the occurrence of an event in area X, each of the triggered, non-CH nodes belonging to that area sends a sensed-data packet to its unique father (whether this father is in the same area or not). Here, we assume that an event-data packet mapping table is handled and used by all nodes to map events to data packets, which will then be sent to the CH. Upon receiving all data packets, the CH will then perform data aggregation by eliminating redundant information. For example, a ”temperature is below 10 degrees at 2 pm” can be thought of as an event, which can be sensed by all nodes in a given cluster, mapped to data packets using their tables, and then sent to CH for further processing and aggregation. In this work, we assume that all nodes agreed beforehand on an event-data mapping table that will then be used to map events to data packets (event-data packet mapping table construction is not within the scope of this work; readers are referred to [11] and [12] for more details.)

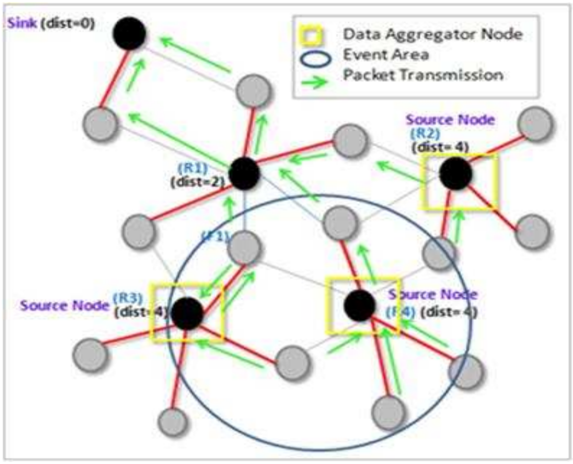


Fig. 3. Data aggregation in a WCDS-structured wireless sensor network

Each triggered CH stores the first packet it receives from one of its children about the event. Then, each time it receives a new packet from any of its other children, it compares it with the stored one, and discards it if carries the same information. Only CHs having performed data aggregation will report this data to the sink. Thus, within a triggered cluster of $|C|$ members, instead of sending $|C|$ packets, only one packet will be reported to the sink, thereby considerably reducing energy consumption. An example, illustrating this data aggregation process, is given in Fig. 3.

b) Energy aware packet forwarding: The following notations will be used throughout this section:

- CH_t = A triggered CH
- $N(CH)$= $\{N_i : N_i$ is a neighbor of $CH\}$
- $d(N)$= Shortest distance from node N to the sink
- $Sn(CH)$=$\{Ni \in N(CH) : d(N_i) \leq d(CH)\}$

In this section, we will describe how data packets are forwarded to the sink. Once the triggered CH, CH_t, performs the data aggregation and constructs the aggregated packet to send to the sink, it consults its routing table to make the routing decision (i.e., the choice of the next hop). It then chooses from the set $Sn(CH_t)$ the node that has the greatest amount of remaining energy as the next hop. This chosen node (whether it is a CH or not) uses the same metric, distance from sink and remaining energy, to forward the packet to the next hop. This repeats until the packet reaches the sink.

In order to avoid routing loops, the chosen path is updated at each hop, and stored in the forwarded packet so that intermediate nodes do not choose already visited nodes. Hence, our technique is loop free. This is because we always progress in the tree from one level to another that is closer to the sink.

It is important to mention that our routing scheme reduces overall energy via data aggregation and increases network lifetime via its tendency of choosing nodes that have higher remaining energy levels. Our routing scheme enables each node to keep track of energy levels. Each node periodically broadcasts an energy notification message to its neighbors, which is then used to update their routing tables so that they make better, energy-aware routing decisions when transmitting their future packets.

In summary, our proposed routing scheme saves overall energy consumption and prolongs network lifetime by: (i) reducing the number of transmissions through data aggregation, and (ii) balancing traffic loads and routing via nodes with higher levels of remaining energy.

V. PERFORMANCE EVALUATION

In this section, we will evaluate the performance of our routing scheme, and compare it with two existing schemes.

A. Simulated Schemes and Performance Metrics

We will compare our proposed scheme with two existing schemes: LEACH [3] and TEEN [4]. We now briefly describe/summarize each of these two schemes as well as our proposed scheme: WCDS-DCR.

1) Simulated Schemes:

LEACH [3]. It assumes that there exists a unique base station outside the sensor network, and that all sensor nodes can communicate with this base station directly. In order to save energy, LEACH chooses a fraction p among all sensor nodes to serve as CHs, where p is a design parameter that must be defined *a priori* before deployment. Each of the other sensor nodes joins the cluster whose CH provides the highest signal strength. In order to maintain equal energy consumption among all nodes, election of CHs is done on a per-round basis, where the set of CHs changes from one round to another. In each round, after cluster formation phase, CHs fuse the data received from their cluster members, and send the aggregated data to the base station in a single-hop communication, thus reducing the total number of transmitted packets to the base station.

LEACH is completely distributed and requires no global knowledge of the network. However, it uses single-hop routing where each source node can only transmit directly to its destination, whether that being a CH or the sink.

In our simulation, LEACH is deployed in a proactive sensor network, and uses single-hop routing (from a non-CH node to the CH, and from the CH to the sink). Periodically and each T_{Round}, new clusters are formed with LEACH through the random process of CH election followed by the exchange of Announcement/Join request packets between the CH and its neighbors. Once this is done (i.e., new clusters are formed), each CH wakes up once every check interval T_{CI} to sense the channel. Every T_{Round}, each CH receives data from the cluster members, aggregates this data (based on data fusion), and sends it to the sink.

TEEN [4]. TEEN is similar to LEACH in that both use hierarchical structures. But unlike LEACH, TEEN uses a data-centric mechanism to fuse information, and hence, reduces the number of transmitted data packets. In our simulation, the network architecture for TEEN is based on a hierarchical grouping, where closer nodes form a first level of clusters, and then closer cluster heads form a second level of clusters called Super-CHs. TEEN acts in a reactive way, and data is sent to the sink only whenever an event occurs. When this happens, packet transmissions are carried out by the triggered nodes in the event area only.

WCDS-DCR. Unlike LEACH and TEEN, WCDS-DCR uses data aggregation and supports multi-hop routing. The clusters as well as the tree structure are built once and for all. However, synchronization is maintained through announcement packets sent by the CHs (Reference/Follower role) every interval T_{AI}. Every check interval, each node wakes up to listen to its logical channel(s) (Sampling Role). Our scheme acts in a reactive way so we consider scenarios where data is reported to the sink only when an event occurs. Moreover, in order to make energy-aware decisions, in our simulations, energy tracking (or energy information update) is done every announcement interval T_{AI}. Like LEACH and TEEN, WCDS-DCR accounts for the energy costs incurred by both the MAC and the routing protocols.

2) Performance Metrics: The main purpose of this work is again to provide a distributed, energy-aware routing scheme to be built on top of an energy-efficient MAC with the two objectives of: (i) decreasing the amount of consumed communication energy, and (ii) increasing the network lifetime by balancing the communication traffic equally among all nodes. To this end, the following two metrics are used to analyze and compare the performance of our routing scheme with two existing ones.

- **Average consumed energy.** This is the average dissipated energy per node due to packet transmission, packet reception, node synchronization, channel sensing, and data aggregation.
- **Network lifetime extension.** This represents the percentage gain of network lifetime as a consequence of using WCDS-DCR when compared with the other two schemes. In this work, the network lifetime is defined to be the amount of time it takes the first node to die.

B. Simulation Setting and Method

The hardware radio characteristics shown in Table II and the energy dissipation model described in [13] are used in this simulation. This radio model has been adopted in several studies [4], [7], [13]–[15], and we next describe its main characteristics. Let d denote the distance between the sender and the receiver. Depending on d, one of the two propagation models, the free-space and the multi-path models, is used. Specifically, the free-space propagation model is considered when $d < d_0$, whereas, the multi-path model is considered when $d > d_0$, where d_0 represents the maximum range for which the free space model applies. The reason for this is that for short distances d, it is less likely to have obstacles between the sender and the receiver, and hence, it is safe to assume the free-space model. When the distance, on the other hand, is large, obstacles are likely to exist in between, and hence, the multi-path model is more appropriate.

Therefore, the energy $E_{Tx}(l,d)$ consumed to transmit an l-bit data packet can be expressed as follows:

$$E_{Tx}(l,d) = \begin{cases} lE_{elec} + ld^2 E_{Amp-fs} & \text{if } d < d_0 \\ lE_{elec} + ld^4 E_{Amp-mp} & \text{if } d \geq d_0 \end{cases}$$

where E_{elec} is the amount of energy to transmit or receive one bit, and E_{Amp-fs} and E_{Amp-mp} are the amounts of the per-bit amplified energy at the transmitter for long and short distances, respectively.

At the receiver side, the energy consumed for reception $E_{Rx}(l)$ can be written as:

$$E_{Rx}(l) = l \times E_{elec}$$

Recall that WCDS-DCR does not incur any cost when aggregating data, whereas both LEACH and TEEN each incurs a cost $E_{DA}(l)$, which can be expressed as

$$E_{DA}(l) = l \times E_{Data-fusion}$$

where $E_{Data-fusion}$ is the required energy per bit for data fusion. All protocol parameters used in this section are summarized in Table I, and all hardware parameters are summarized in Table II[3].

For evaluation purposes, we generated and simulated sensor networks, each consisting of 100 nodes and one sink. Unless otherwise stated, nodes are randomly placed in an area of size 100×100 units. We assume that different regions of the sensed area experience different temperatures, and that sensors sense and report temperatures to the sink. Unless otherwise mentioned, the

---

[3] Hardware parameters are those of [13].

TABLE I

PROTOCOL PARAMETERS USED IN SIMULATIONS

$T_{\text{CI}}: CheckInterval$	1 sec.
T_{Round}	20 sec.
$T_{\text{AI}}: AnnoucnementInterval$	20 min
Announcement Packet Length	10 B
Join-Req Packet Length	10 B
Data Packet Length	2000 bit
Energy Notification Packet Length	1 B
$T_{\text{Preamble}-\text{WCDS}}$	1 sec.
$T_{\text{Preamble}-\text{LEACH}-\text{TEEN}}$	1 sec.
Bandwidth	19.6 kbps

TABLE II

HARDWARE RADIO CHARACTERISTICS

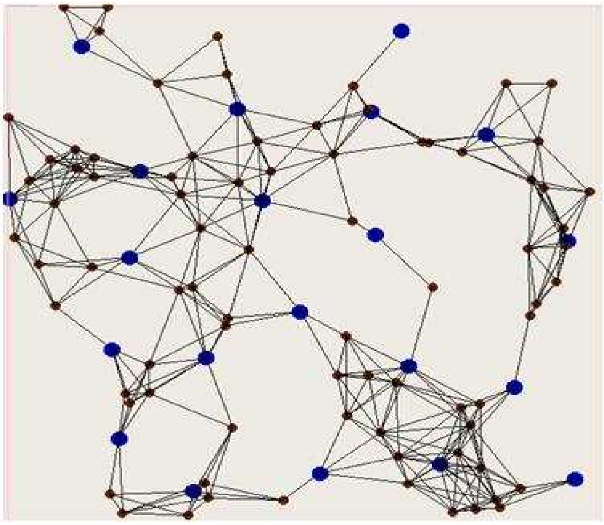
$Voltage$	3 Volt
$\theta(ClockDrift)$	$20 \cdot 10^{-6}$
$E_{\text{Tx}-\text{elec}}$	50 nJoule/bit
$E_{\text{Rx}-\text{elec}}$	50 nJoule/bit
$E_{\text{Amp}-\text{fs}}$	$10pJoule/bit/m^2$
$E_{\text{Amp}-\text{mp}}$	$13 \cdot 10^{-4}pJoule/bit/m^4$
$E_{\text{Data}-\text{fusion}}(LEACH)$	5 nJoule/bit/signal

used check interval (i.e., the duty cycle) value is set to 1 second. The average node degree is equal to 10 (i.e., on average, each node is surrounded by 10 neighbors). Fig. 4 illustrates the network organization for (i) WCDS-DCR (Fig. 4(a)), where CHs are connected to each other via non-CH nodes and cooperate to report the information to the sink in a multi-hop fashion, and (ii) LEACH (Fig. 4(b)), where CHs are directly connected to the sink.

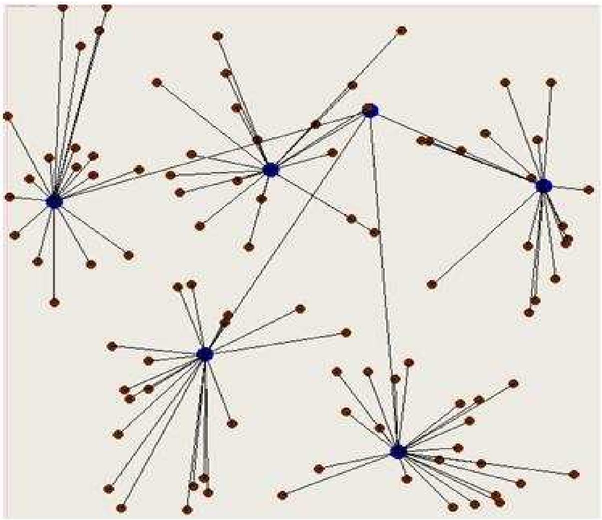
C. Simulation Results

The performance of WCDS-DCR is evaluated then compared to LEACH and TEEN. In our simulations, 20 events are generated randomly in space and time within the observation time (taken equal to 1 hour). These events represent temperature variations within randomly chosen areas. We executed 20 runs of the simulator for each protocol and for parameter setting. The readings from these 20 trials were then averaged and plotted.

1) Impact of the Check Interval: The check interval, commonly known as the duty cycle, is an important design parameter that represents the duration separating two successive channel sensing operations. This design parameter has an impact on energy consumption as well as network capacity. In TDMA-based solutions, when using one physical channel like in LEACH and TEEN, the smaller the check interval, the lesser the node throughput, especially in dense networks. Under WCDS-DCR, on the other hand, which uses multiple channels, a small check interval does not affect the capacity. Instead, it affects the energy consumption at the MAC layer, which increases as the check interval decreases.



(a) WCDS-clustered network



(b) LEACH-clustered network

Fig. 4. Illustration of cluster formation under WCDS-DCR and LEACH

Fig. 5 plots the average per-node, consumed energy as a function of the check interval under each of the three simulated schemes. First, observe that LEACH performs much worse than both TEEN and WCDS-DCR. In fact, our scheme consumes about 5 times less energy than LEACH

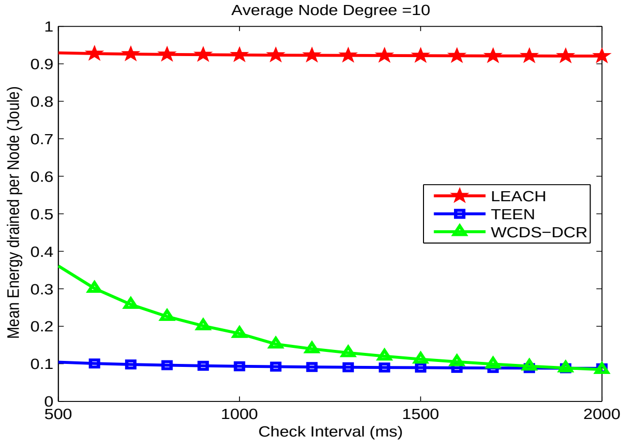


Fig. 5. Impact of the check interval on energy consumption

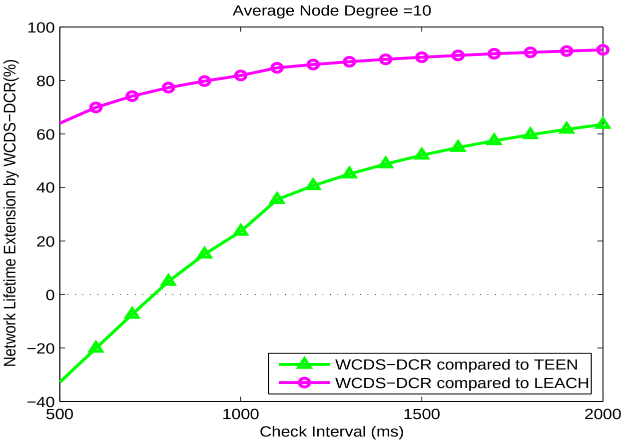


Fig. 6. Impact of the Check Interval on network lifetime

when the check interval T_{CI} is around 1 second, and up to 9 times less when $T_{CI} \geq 1.5$ seconds. Second, observe that for small values of check intervals, TEEN performs a little better than WCDS-DCR. However, as the check interval increases, WCDS-DCR starts consuming less energy, and it eventually achieves a similar performance to that of TEEN.

It is important to reiterate that WCDS-DCR is designed with the objective of increasing network lifetime while maintaining low overall energy consumption. Hence, even though our scheme performs almost as well as TEEN in terms of energy consumption, it performs much better than TEEN (and LEACH too) when it comes to network lifetime. To justify our claim, we plot in Fig. 6 the network lifetime gain under WCDS-DCR when compared with TEEN and LEACH as a function of the check interval. When compared with TEEN, the figure shows that our scheme prolongs the network lifetime by about 25% when T_{CI} is equal to one second, and by about 60% when T_{CI} is equal to two seconds. When compared with LEACH, our scheme performs even better: WCDS-DCR extends the network lifetime by more than 80% for check intervals of length equal to or greater than one second.

2) Impact of the Area size: In this experiment, we keep all parameters fixed while jointly varying the simulation area length and the per-node transmission range in order to keep the same average node degree, thus masking the effect of node degree. The per-node transmission range for our scheme is then varied when the simulation area is varied so as to maintain network connectivity with the same average node degree.

Fig. 7 shows the average, per-node consumed energy as a function of the simulation area size under each of the three routing schemes. Observe that WCDS-DCR outperforms LEACH substantially; in fact, as the simulation area grows, energy consumption increases exponentially under LEACH while remaining the same under our scheme. On the other hand, although TEEN incurs slightly lower energy consumption than ours for small sizes, as the size increases, the consumed energy starts increasing substantially under TEEN while remaining the same under WCDS-DCR. We conclude that for sparse networks, WCDS-DCR achieves better performances in terms of energy consumption and is more scalable than both TEEN and LEACH.

Fig. 8 plots the network lifetime extension as a result of using WCDS-DCR when compared with TEEN and LEACH. The figure shows that WCDS-DCR also outperforms both TEEN and LEACH in terms of network lifetime. When compared with LEACH, while WCDS-DCR can increase network lifetime by about 80% when the area length equals 80 units, it can even double it (i.e., a 100% increase) when the area length increases above 180 units. WCDS-DCR also performs better than TEEN. This is especially true in sparse networks, where network lifetime can also be doubled as a result of using WCDS-DCR.

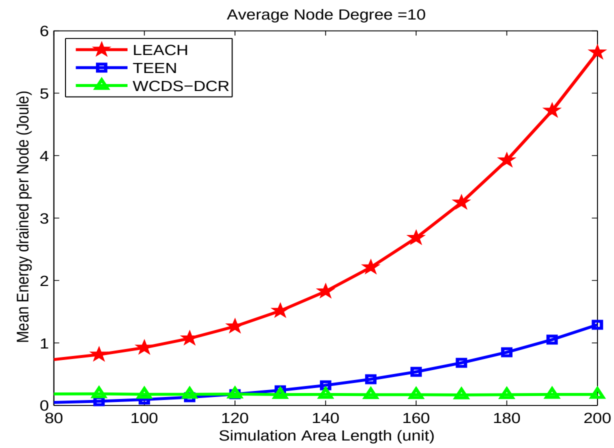


Fig. 7. Impact of the Simulation Area Length on energy consumption

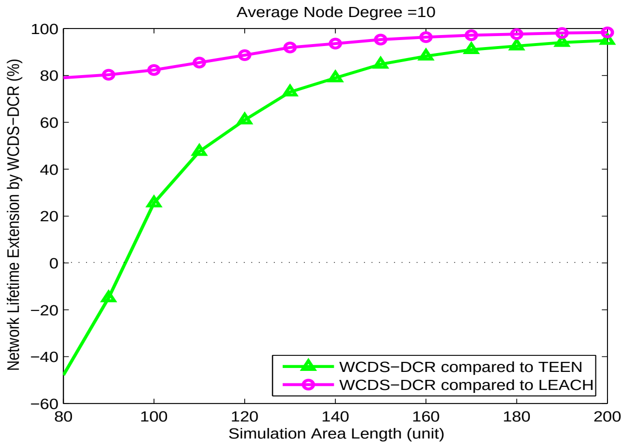


Fig. 8. Impact of the Simulation Area Length on network lifetime

VI. CONCLUSION

In this paper, we propose a new energy-aware routing scheme for data aggregation in wireless sensor networks. Our scheme is based on a WCDS-induced clustering structure, and uses the energy-efficient MAC that we recently proposed in [2] as its underlying MAC protocol. The clustering structure enabled efficient use of data aggregation by discarding redundant data packets, thus reducing network traffic load. In brief, our proposed routing scheme saves overall energy consumption and prolongs network lifetime by: (i) reducing the number of transmissions through data aggregation, and (ii) balancing traffic loads and routing via nodes with higher levels of remaining energy. Results show that the proposed scheme can prolong network lifetime substantially, and can even double the lifetime in the case of sparse networks. Results also show that the proposed scheme is more scalable than existing schemes.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” in *Computer Networks, published bu Elsevier Science*, 2002, pp. 392–422.

[2] N. Chakchouk, A. Bachir, M. Frikha, and A. Duda, “L’auto-organisation dans les rseaux de capteurs,” *Technical Report*, June 2008.

[3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless sensor networks,” in *the Hawaii International Conference System Sciences*, January 2000.

[4] A. Manjeshwar and D. P. Agrawal, “Teen: A routing protocol for enhanced efficiency in wireless sensor networks,” in *1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing*, April 2001.

[5] S. Lindsey and C.S. Raghavendra, “Pegasis: Power efficient gathering in sensor information systems,” in *the IEEE Aerospace Conference*, March 2002.

[6] 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.

[7] M. Ye, C.F. Li, G. Chen, and J. Wu, “Eecs: An energy efficient clustering scheme in wireless sensor networks,” *Intl Journal of Ad Hoc Sensor Networks*, vol. 3, pp. 99–119, 2007.

[8] N. Dankwa, “An evaluation of transmit power levels for node localization on the mica2 sensor node,” May 2004.

[9] ,” in *http://www.coronis-systems.com*.

[10] A. Bachir, D. Barthel, M. Heusse, and A. Duda, “Micro-frame preamble mac for multihop wireless sensor networks,” in *IEEE ICC*, June 2006.

[11] A.Ghose, J.Grossklags, and J.Chuang, “Resilient data-centric storage in wireless ad-hoc sensor networks,” in *4th International Conference on Mobile Data Management*, 2003.

[12] L.Guo, Y.Li, and J.Li, “Event query processing based on data-centric storage in wireles sensor networks,” in *Globecom*, 2006.

[13] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An application-specific protocol for wireless microsensor networks,” in *IEEE Transactions on Wireless Communications*, October 2002.

[14] S. Wang X. Zhou B. Gong, L. Li, “Multihop routing protocol with unequal clustering for wireless sensor networks,” in *ISECS International Colloquium on Computing, Communication, Control, and Management*, 2008, pp. 552–556.

[15] W. Huang, M. Yu, L. Xiong, and J. Wen, “Energy-efficient hierarchical routing protocol for wireless sensor networks,” in *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, 2008, pp. 640–644.