Abstract: Wireless Sensor Networks (WSNs) is a class of wireless ad hoc networks in which sensor nodes collect, process, and communicate data acquired from the physical environment to an external Base-Station (BS). But the fundamental challenge in the design of Wireless Sensor Networks (WSNs) is to maximize their lifetimes especially when they have a limited and non-replenishable energy supply. To increase the lifetime of the sensor network, power management and energy-efficient communication techniques at all layers become necessary. In this paper, we present solutions for the data gathering and routing problem with in-network aggregation in WSNs. Our objective is to maximize the network lifetime by utilizing data aggregation and in-network processing techniques.

1. INTRODUCTION

A Wireless Sensor Network (WSN) consists of spatially distributed autonomous devices that cooperatively sense physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants at different locations [1], [2].

Wireless Sensor Networks (WSNs) is a class of wireless ad hoc networks in which sensor nodes collect, process, and communicate data acquired from the physical environment to an external Base-Station (BS) [1]. Future WSNs are envisioned to revolutionize maintenance free and fault tolerant platform for collecting and processing information in diverse environments. A major technical challenge for WSNs, however, lies in the node energy constraint and its limited computing resources, which may pose a fundamental limit on the network lifetime [1]. Therefore, innovative techniques to eliminate energy inefficiencies that would otherwise shorten the lifetime of the network.

Sensor nodes are energy-constrained devices and the energy consumption is generally associated with the amount of gathered data, since communication is often the most expensive activity in terms of energy. For that reason, algorithms and protocols designed for WSNs should consider the energy consumption in their conception. Moreover, WSNs are data-driven networks that usually produce a large amount of information that needs to be routed, often in a multihop fashion, toward a sink node, which works as a gateway to a monitoring center. Given this scenario, routing plays an important role in the data gathering process.

The main idea of the data centric routing or data aggregation and in-network processing approaches is to combine the data arriving from different sources (sensor nodes) at certain aggregation points (or simply aggregators) and route, eliminate redundancies by performing simple processing at the aggregation points, and minimize the total amount of data transmission before forwarding data to the external Base station. Removing redundancies results in transmitting fewer numbers of bits, and hence reduces energy consumption and increases the sensor nodes’ lifetimes.

A possible strategy to optimize the routing task is to use the available processing capacity provided by the intermediate sensor nodes along the routing paths. This is known as data-centric routing or in-network data aggregation. For more efficient and effective data gathering with a minimum use of the limited resources, sensor nodes should be configured to smartly report data by making local decisions. For this, data aggregation is an effective technique for saving energy in WSNs. Due to the inherent redundancy in raw data gathered by the sensor nodes, in-networking aggregation can often be used to decrease the communication cost by eliminating redundancy and forwarding only smaller aggregated information. Since minimal communication leads directly to energy savings, which extends the network lifetime, in-network data aggregation is a key technology to be supported by WSNs. In this paper, the terms information fusion and data aggregation are used as synonyms. In this context, the use of information fusion is twofold: (i) to take advantage of data redundancy and increase data accuracy, and (ii) to reduce communication load and save energy.

2. APPLICATIONS OF SENSOR NETWORKS
In the recent past, wireless sensor networks have found their way into a wide variety of applications and systems with vastly varying requirements and characteristics [6][8]. The sensor networks are used in surveillance systems like Disaster Relief, Emergency Rescue operation, Military, Habitat Monitoring, Health Care, Environmental monitoring, Home networks, detecting chemical, biological, radiological, nuclear, and explosive material etc. How the sensor network is used in these areas are listed in table 1 below.

<table>
<thead>
<tr>
<th>Area</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Military</td>
<td>Military situation awareness[6].</td>
</tr>
<tr>
<td></td>
<td>Sensing intruders on basis.</td>
</tr>
<tr>
<td></td>
<td>Detection of enemy unit movements on land and sea [4].</td>
</tr>
<tr>
<td></td>
<td>Battle field surveillances [5].</td>
</tr>
<tr>
<td>Emergency situations</td>
<td>Disaster management [9].</td>
</tr>
<tr>
<td></td>
<td>Fire/water detectors [3].</td>
</tr>
<tr>
<td></td>
<td>Hazardous chemical level and fires [4].</td>
</tr>
<tr>
<td>Physical world</td>
<td>Environmental monitoring of water and soil [7].</td>
</tr>
<tr>
<td></td>
<td>Habitual monitoring [7].</td>
</tr>
<tr>
<td></td>
<td>Observation of biological and artificial systems [7].</td>
</tr>
<tr>
<td>Medical and health</td>
<td>Sensors for blood flow, respiratory rate, ECG(electrocardiogram), pulse oxymeter, blood pressure and oxygen measurement [10].</td>
</tr>
<tr>
<td></td>
<td>Monitoring people’s location and health condition [5].</td>
</tr>
<tr>
<td>Industrial</td>
<td>Factory process control and industrial automation [6].</td>
</tr>
<tr>
<td></td>
<td>Monitoring and control of industrial equipment [3].</td>
</tr>
<tr>
<td>Home networks</td>
<td>Home appliances, location awareness (blue tooth [3]).</td>
</tr>
<tr>
<td></td>
<td>Person locator [10].</td>
</tr>
<tr>
<td>Automotive</td>
<td>Tire pressure monitoring [3][4].</td>
</tr>
<tr>
<td></td>
<td>Active mobility [8].</td>
</tr>
<tr>
<td></td>
<td>Coordinated vehicle tracking [6].</td>
</tr>
</tbody>
</table>

In many applications of sensor networks, it is not feasible to assign global identifiers to each node due to the sheer number of nodes deployed. Such lack of global identification along with random deployment of sensor nodes makes it difficult to select a specific set of sensor nodes to be queried. Therefore, data is usually transmitted from every sensor node within the deployment region with significant copies of the data which is called as redundancy. Since this is very inefficient in the aspect of energy consumption, routing protocols that will be able to select a set of sensor nodes and utilize data aggregation during the transmission of data have been considered. This consideration has led to data-centric routing, which is different from traditional address-based routing where routes are created between addressable nodes and no aggregation will happen the address-centric/address-based routing which is managed in the network layer of the communication stack.

In data-centric routing, the sink sends queries to certain regions which might be done using clusters or may be a certain nodes in the routing path and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data. The first data-centric routing is done by using a SPIN protocol, which considers data negotiation between nodes in order to eliminate redundant data and save energy. Later, Directed Diffusion [11] has been developed and has become a breakthrough in data-centric routing. Then, many other methods have been proposed will describe some of those methods in detail and highlight the some key points about those methods.

3.1 Flooding and gossiping

Flooding and gossiping [14] are two classical mechanisms to relay data in sensor networks without the need for any routing algorithms and topology maintenance. In flooding, each sensors in the network receives a data packet broadcasts it to all of its neighbors and this process continues until the packet arrives at the destination or the maximum number of hops for the packet is reached but this may lead to the packet dropping by wasting the energy of the sensors in the network. On the other hand, gossiping is a slightly enhanced version of flooding where the receiving node sends the packet to a randomly selected neighbor, which picks another random neighbor to forward the packet to and so on.

![Figure 1](image_url)

**Figure 1** The implosion problem. Node A starts by flooding its data to all of its neighbors. D gets two same copies of data eventually, which is not necessary.
Gossiping is among the early work in which a data-centric routing mechanism is done for the first. The idea behind SPIN is to name the data using high-level descriptors or meta-data. Before transmission, metadata are exchanged among sensors via a data advertisement mechanism, which is the key feature of SPIN. Each node upon receiving new data, advertises it to its neighbors and interested neighbors, i.e. those who do not have the data, retrieve the data by sending a request message. SPIN’s meta-data negotiation solves the classic problems of flooding such as redundant information passing, overlapping of sensing areas and resource blindness thus, achieving a lot of energy efficiency. But there is no standard meta-data format and it is assumed to be application specific, e.g. using an application level framing. There are three messages defined in SPIN to exchange data between nodes. These are: ADV message to allow a sensor to advertise a particular meta-data to their neighboring sensors, REQ message contains a request message for the specific data and DATA message that carry the actual data from the neighboring nodes to the node which sends the ADV message. Figure 3, summarizes the steps of the SPIN method. Later the node B will send the ADV message to the all its neighboring nodes and the process will repeat as done for the node A. One of the advantages of SPIN is that topological changes are localized since each node needs to know only its single-hop neighbors. SPIN gives a factor of 3.5 less than flooding in terms of energy dissipation and meta-data negotiation almost halves the redundant data. However, SPIN’s data advertisement mechanism cannot guarantee the delivery of data. For instance, if the nodes that are interested in the data are far away from the source node and the nodes between source and destination are not interested in that data, such data will not be delivered to the destination at all. Therefore, SPIN is not a good choice for applications such as intrusion detection, which require reliable delivery of data packets over regular intervals.

3.3 Directed Diffusion

Directed Diffusion [11, 12] is an important milestone in the data-centric routing research of sensor networks. The idea aims at diffusing data through sensor nodes by using a naming scheme for the data. The main reason behind using such a scheme is to get rid of unnecessary operations of network layer routing in order to save energy. Direct Diffusion suggests the use of attribute-value pairs for the data and queries the sensors in an on demand basis by using those pairs. In order to create a query, an interest is defined using a list of attribute-value pairs such as name of objects, interval, duration, geographical area, etc. The interest is broadcast by a sink through its neighbors. One of the advantage here is each node receiving the interest can do caching for later use. The interests in the caches are then used to compare the received data with the values in the interests. The interest entry also contains several gradient fields. A gradient is nothing but a reply link to a neighbor from which the interest was received. It is characterized by the data rate, duration and expiration time derived from the received interest’s fields. Hence, by utilizing interest and gradients, paths are established between sink and sources. Several paths can be established so that one of them is selected by reinforcement. The sink resends the original interest message through the selected path with a smaller interval hence reinforces the source node on that path to send data more frequently. Figure 4, summarizes the Directed Diffusion method.

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

However, Directed Diffusion cannot be applied to all sensor network applications since it is based on a query-driven data delivery model. The applications that require
continuous data delivery to the sink will not work efficiently with a query-driven on demand data model. Therefore, Directed Diffusion is not a good choice as a routing protocol for the applications such as environmental monitoring. In addition, the naming schemes used in Directed Diffusion are application dependent and each time should be defined a priori. Moreover, the matching process for data and queries might require some extra overhead at the sensors.

The method DRINA can be divided into three phases. In Phase 1, the hop tree from the sensor nodes to the sink node is built. In this phase, the sink node starts building the hop tree that will be used by Coordinators for data forwarding purposes. Phase 2 consists of cluster formation and clusterhead election among the nodes that detected the occurrence of a new event in the network. Finally, Phase 3 is responsible for both setting up a new route for the reliable delivering of packets and updating the hop tree.

3.4.1 Phase 1: Building the Hop Tree

In this phase, the distance from the sink to each node is computed in hops. This phase is started by the sink node sending, by means of a flooding, the Hop Configuration Message (HCM) to all network nodes. The HCM message contains two fields: ID and HopToTree, where ID is node identifier that started or retransmitted the HCM message and HopToTree is the distance, in hops, by which an HCM message has passed.

The HopToTree value is started with value 1 at the sink, which forwards it to its neighbors (at the beginning, all nodes set the HopToTree as infinity). Each node, upon receiving the message HCM, verifies if the value of HopToTree in the HCM message is less than the value of HopToTree that it has stored and if the value of FirstSending is true, then the node updates the value of the NextHop variable with the value of the field ID of message HCM, as well as the value of the HopToTree variable, and the values in the fields ID and HopToTree of the HCM message. The node also relays the HCM message. Otherwise, if that condition is false, which means that the node already received the HCM by a shorter distance, then the node discards the received HCM message. The steps described above occur repeatedly until the whole network is configured.

Before the first event takes place, there is no established route and the HopToTree variable stores the smallest distance to the sink. On the first event occurrence, HopToTree will still be the smallest distance; however, a new route will be established. After the first event, the HopToTree stores the smaller of two values: the distance to the sink or the distance to the closest already established route.

3.4.2 Phase 2: Cluster Formation

When an event is detected by one or more nodes, the leader election algorithm starts and sensing nodes will be running for leadership (group coordinator); this process is described here. For this election, all sensing nodes are eligible. If this is the first event, the leader node will be the one that is closest to the sink node. Otherwise, the leader will be the node that is closest to an already established route. In the case of a tie, i.e., two or more concurrent nodes have the same distance in hops to the sink (or to an established route), the node with the smallest ID maintains eligibility. Another possibility is to use the energy level as a tiebreak criterion.

At the end of the election only one node in the group will be declared as the leader (Coordinator). The remaining nodes that detected the same event will be the...
Collaborators. The Coordinator gathers the information collected by the Collaborators and sends them to the sink. A key advantage of this cluster formation is that all of the information gathered by the nodes sensing the same event will be aggregated at a single node (the Coordinator), which is more efficient than other aggregation mechanisms.

### 3.4.3 Phase 3: Routing Formation and Hop Tree Updates

The elected group leader, as described in previous stage, will lead to establishing the new route for the event dissemination. This process is described in this phase. For that, the Coordinator sends a route establishment message to its NextHop node. When the NextHop node receives a route establishment message, it re-transmits the message to its NextHop and starts the hop tree updating process. These steps are repeated until either the sink is reached or a node that is part of an already established route is found. The routes are created by choosing the best neighbor at each hop. The choices for the best neighbor are twofold: (i) when the first event occurs, the node that leads to the shortest path to the sink is chosen and (ii) after the occurrence of subsequent events, the best neighbor is the one that leads to the closest node that is already part of an established route. This process tends to increase the aggregation points, ensuring that they occur as close as possible to the events.

The resulting route is a tree that connects the Coordinator nodes to the sink. When the route is established, the hop tree updating phase is started. The main goal of this phase is to update the HopToTree value of all nodes so they can take into consideration the newly established route. This is done by the new relay nodes that are part of an established route. These nodes send an HCM message (by means of a controlled flooding) for the hop updating. The whole cost of this process is the same of a flooding, i.e., each node will send only one packet.

### 3.4.4 Route Repair Mechanism

In the method DRINA the route repair mechanism is also considered because if any of the nodes get dropped or weak by its energy level the aggregated message will get dropped so it has been included here. The route created to send the data toward the sink node is unique and efficient since it maximizes the points of aggregation and, consequently, the information fusion. However, because this route is unique, any failure in one of its nodes will cause disruption, preventing the delivery of several gathered event data. Possible causes of failure include low energy, physical destruction, and communication blockage. Some fault tolerant algorithms for WSNs are used here. Some are based on periodic flooding mechanisms [15], [16], and rooted at the sink, to repair broken paths and to discover new routes to forward traffic around faulty nodes. This mechanism is not satisfactory in terms of energy saving because it wastes a lot of energy with repairing messages. So DRINA offers a piggybacked, ACK based, route repair mechanism, which consists of two parts: failure detection at the NextHop node, and selection of a new NextHop. It creates a new route when the sensor node is down in its behavior.

### 4. Conclusions

In this paper, we studied the maximum lifetime data gathering and routing problem in WSNs. We showed that cluster-based algorithms along with data aggregation and in-network processing can achieve significant energy savings in WSNs. In a particular scheme for WSNs that combines the ideas of fixed cluster-based routing together with application-specific data aggregation in order to enhance the wireless sensor network performance in terms of extending the network lifetime, while incurring acceptable levels of latency under data aggregation.

### References


AUTHOR