DSRP: Distributed SensorWeb Routing Protocol

Abhinav Valada, David Kohanbash, George Kantor
The Robotics Institute
Carnegie Mellon University, Pittsburgh, PA 15213
Email: {avalada, dkohanba, kantor}@cmu.edu

***Abstract*—We propose a new multi-hop routing protocol for wireless sensor networks, suited for monitoring and control applications. The aim of this research is to adapt flat and hierarchical architectures to create a new hybrid that draws on current protocol theories. The protocol uses a hybrid network structure to achieve scalability and is source initiated along with time driven reporting to reduce the number of packet transmissions. The protocol incorporates a link quality estimation algorithm, which enables only the nodes with high quality symmetric links to be chosen for routing. Route selection is calculated using both hop count and link quality as routing metrics. The protocol is also designed such that it is computationally simple, reliable, energy aware, does not impose any special hardware prerequisites and most importantly credible. Its credibility was verified by performing a series of field tests in a real world operating environment.**

I. Introduction

Wireless Sensor Networks (WSNs) has attracted much attention in recent years. They are used for collecting, storing and sharing sensed data. WSNs have been used for various applications including habitat monitoring, agriculture, nuclear reactor control, security and tactical surveillance. The architecture of a wireless sensor network system comprises of a set of sensor nodes and a base station that communicate with each other and gather local information to make global decisions about the physical environment.

Routing protocols in WSNs can be classified according to the network structure as flat, hierarchical and location-based routing [1]. In flat routing protocols every node has the same function and participates equally in the routing task. Hierarchical routing also known as cluster-based routing subdivides the network into small groups and routing is performed hierarchically. In location-based routing protocols, the sensor nodes are addressed according to their location. They often require special hardware like GPS and RFID to identify their location.

Network lifetime and communication integrity are the two primary factors that were considered while designing the DSRP. There are many other factors related to the inherent characteristics of WSNs that have to be considered. Some of the factors are reliability, scalability, energy consumption, simplicity, data reporting method, hardware independence, connectivity, and quality of service [1]. The paper is organized as follows. In section II we discuss some of the relevant multi-hop routing protocols and how the DSRP is better than these. Section III presents the design choices made and the effect of them on the performance of the protocol. The protocol operation is described in section IV and in section V we evaluate the protocol via results from the field experiments. Finally, section VI concludes the paper.

II. Related Work

Various routing protocols have been proposed for WSNs to address the different concerns. Flooding is the simplest routing approach in WSNs. There are several disadvantages to flooding which makes it unreliable and inefficient for practical applications. Low-Energy Adaptive Clustering Hierarchy (LEACH) is the most popular hierarchical cluster based routing protocol. The operation of LEACH partitions the network into clusters and a special node known as the cluster head aggregates the data packets and transmits it to the base station [2]. Direct Diffusion is a data centric routing protocol where the data is named by attribute-value pairs [3]. The base station requests for data by broadcasting interests which propagates throughout the network. Gradients are then setup from the receiving node to the base station. When the interests fit gradients, paths of information are formed and then the best path is reinforced. Minimum Cost Forwarding Algorithm (MCFA) is a simple protocol in which data is transmitted to the base station in a multi-hop path that has the lowest cost metric [4]. Power Efficient Gathering in Sensor Information Systems (PEGASIS) is another clustering based routing protocol in which nodes are organized into a chain using a greedy algorithm so that each node transmits to and receives only from one of its neighbors.

In [5], the authors use a packet delivery rate routing metric which chooses long low loss paths but ignores shorter paths which may give a better performance. In the Dynamic Destination-Sequenced Distance-Vector (DSDV) protocol, the authors use a modified ETX (Expected number of transmissions) routing metric. Although this protocol gives good average throughput, it results in high end-to-end delays. In [6], the authors use the RTQ metric for the route selection process but it is very similar to the AODV (Ad-hoc On Demand Distance Vector) protocol and it does not make any improvements. The authors of [7] propose a bidirectional AODV protocol for sensor networks. The results depict that it has a high message overhead. Most importantly it requires frequent updation of the routing table and on node failure, it again initiates the route discovery which causes a lot of power consumption. In this paper we propose a protocol which uses a modified PRR (Packet reception ratio) and link quality as routing metrics. Our protocol shows an increase in the average throughput compared to the above mentioned protocols and also results in low end-to-end delays. The creation of hotspots are prevented in the DSRP by only using the nodes having energies greater than a certain critical level for routing purposes. The link quality estimation algorithm takes into account the normalized remaining energy in the nodes so that the low energy nodes are avoided in the routing tasks.

Even though there are several WSN routing protocols, there is still a need for new protocols that address the application specific design goals and can be implemented easily keeping in mind the current technological limitations. In this work, we propose a new hybrid protocol called Distributed SensorWeb Routing Protocol (DSRP), which is designed for monitoring and control applications. Some of the features of this protocol which makes it unique from the others are symmetric link quality estimation, two way communication for configuring node parameters, dual routing metrics, confirmed delivery data transmission, fault tolerance, topology and special hardware independent, hotspots prevention and routing path reconfiguration upon node failure.

III. Distributed SensorWeb Routing Protocol

A. Protocol Design Choices

Reliability is one of the most important design criteria of WSNs as the nodes are always susceptible to failure. We introduce reliability into the design by implementing confirmed delivery transmission in which the node transmits the packet up to n times until an acknowledgment is received from the destination node. Symmetric link quality estimation makes sure that only the nodes with strong stable links are chosen for routing. Scalability can be defined as the ability of a network to adapt to an increase in network size. The number of nodes in a WSN can range from a few nodes to thousands of nodes. The DSRP does not have any constraint to the maximum number of nodes that can join the network. An important factor that influences the scalability of a routing protocol is the network structure. To have high scalability, a flat network structure was chosen. The use of a flat network structure implies that every node in the network has the same function and will be able to participate equally in the routing task. The protocol is source initiated, hence the base station does not have to query the nodes each time for data.

We can improve the throughput of the network by adopting a routing metric which includes both hop count and link quality. Most protocols use only hop count as the routing metric which might lead to the selection of a longer distance link. As the data rates of the radio are controlled according to the wireless link quality, selection of long distance links will lead to a low signal to noise ratio and also increase the packet loss. Energy consumption in WSN nodes occurs due to computational processing and communication. Communication energy is conserved by limiting the packet sizes and the number of packets routed through the network. Computational energy is also be conserved by limiting the number of tasks that the node has to perform.

A time driven reporting approach is adopted where the data packets are transmitted periodically whenever a new sensor measurement is taken. The DSRP does not require the node to have special hardware capabilities like GPS, high power transmitters and RFID. This facilitates the ability to operate on different platforms. Two way communication is required in networks used in control applications. The base station should be able to transmit command packets to the sensor nodes to modify or update the control parameters. The DSRP supports the feature by providing a configuration packet. Fault tolerance influences the reliability of a WSN to a great extent. Nodes may fail due to several reasons such as environmental interference, low power or physical damage. In the case of node failure the source node transmitting the data should be able to reroute the packet through another neighbor.

B. Protocol Design and Operation

In the Distributed SensorWeb Routing Protocol each node in the network maintains a neighborhood table containing an entry for all nodes within transmission distance. The node parameters in the table are assigned during the setup stage and the routing decisions are made depending on them. The neighborhood table is updated whenever a change in any of the parameters is detected. The protocol operation can be described in five steps.

- STEP 1: Link Quality Estimation
- STEP 2: Network Setup
- STEP 3: Transmitting and Forwarding Data
- STEP 4: Network Maintenance and Neighborhood Table Management
- STEP 5: Configuring the Node Settings

1) Link Quality Estimation: The neighbors of the nodes are discovered in this step and the link quality between them is calculated. Packet delivery ratio is used to estimate link quality. Each node in the network determines the link quality by broadcasting N link estimate packets. A random delay τ is introduced between successive link estimate packets to avoid packet collisions. The nodes that receive these packets transmit an acknowledgement. The source node calculates the link quality as

$$PRR_m = \frac{\textit{No of ack packets received}}{\textit{No of packets sent}} \tag{1}$$

$$batt_N = \frac{(batt_t - batt_{critial})}{batt_{max}} \tag{2}$$

$$L = batt_N \times PRR_m \tag{3}$$

Where, PRR_m is the modified packet reception ratio, L is the link quality estimator, $batt_N$ is the normalized remaining battery voltage, $batt_t$ is the nodes battery voltage at time t, $batt_{critial}$ is the nodes critical low battery voltage and $batt_{max}$ is the maximum battery voltage. A threshold T is introduced so that the nodes with poor link quality are not added to the neighborhood table. The nodes may transmit corrupted packets when they have low power and this causes the network to malfunction. To prevent such failures, the normalized value of the nodes remaining battery voltage is also considered for the link quality estimation. This also helps in choosing nodes with higher power for routing, as the nodes with low battery voltages will fail sooner. The link estimate packet also serves as an initializer for the nodes that are added to the network after the setup phase. The new node initializes its hop count and time from the link estimate packets. This enables it to dynamically join the network without the need of resetting the entire network.

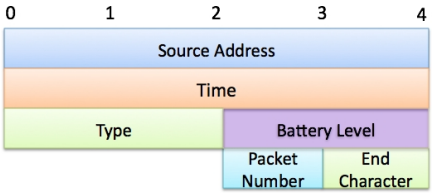


Fig. 1. Link quality estimate packet

Once the node is deployed in the operating environment, it begins the link quality estimation. The contents of the link estimate packet are shown in figure 1. When a node receives a link estimate packet, it adds the address, hop count and battery voltage of the source node into its neighborhood table and transmits the link quality acknowledgement packet to that node. The contents of the link quality acknowledgement packet are shown in figure 2. In case the network has already been setup when a node is powered on, the node initializes its hop-count and current time from the contents of the link quality acknowledgement packets that it receives from its neighboring node.

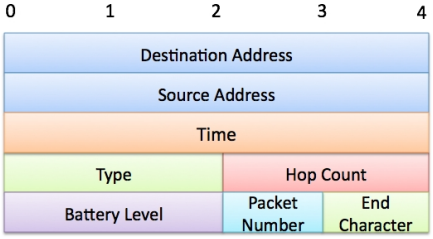


Fig. 2. Link quality acknowledgement packet

2) Network Setup: The base station transmits the setup packet soon after the link quality estimation stage. It is first transmitted to the neighbors of the base station, which then propagates through out the network. The contents of the setup packet are shown in figure 3. When a node receives the setup packet, it first compares the destination address on the packet to its own address to see if they match. If the addresses are a successful match, then the node checks to see if the sequence number in the setup packet is greater than the sequence number that is stored in the node. Each time the base station sends a new setup packet it increments the sequence number. The nodes store the sequence number in the setup packets locally for future comparisons. This prevents the flooding of setup packets throughout the network. The receiving node then compares the hop count on the setup packet to the hop count stored in the node. If the hop count in the setup packet is lower than the hop count stored in the node, it increments the hop count in the setup packet and initializes it as its own hop count. Each time the node receives a setup packet it also updates the hop count of the source node in its neighborhood table. If the address of the source node is not found in its neighborhood table then the node resends the link estimate packet to add the new neighbor to the table. The node forwards the setup packet each time it updates its hop count. Before forwarding the setup packet, the node changes the source address to its own address and the hop count to the nodes updated hop count. This way all the nodes maintain an updated neighborhood table.

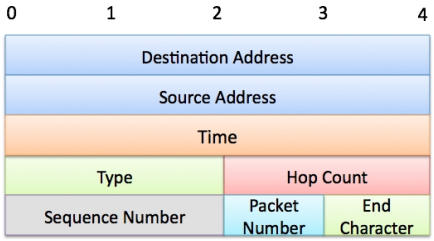


Fig. 3. Network setup packet

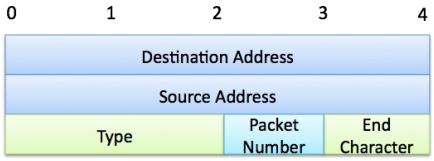


Fig. 4. Confirmed delivery acknowledgement packet

An illustration of the neighborhood table is shown in figure 5. The neighborhood table contains a column for a parameter called as the history variable. The history variable is used to keep track of the successful data transmissions to the neighboring nodes. The history variable has a value of *1111* initially. The functioning of the history variable is described in the following section. The setup flag in the neighborhood table is used to keep track of the acknowledgement for the setup packet. The contents of the confirmed delivery acknowledgement packet are shown in figure 4. By incorporating confirmed delivery transmission to the network setup, we make sure that the nodes which are active in the network receive the setup packet. The maximum number of retries for confirmed delivery is set to ten by default.

1 5 6 7 8 9

Address	Hop Count	Link Quality	History Variable	Setup Flag
88802C86	1	0.9658772	1111	1
34932ECB	2	0.6109181	1011	1
88802B61	2	0.0647221	1111	1
819347BC	3	0.7688097	0100	1

Fig. 5. Neighborhood table

3) Transmitting and Forwarding Data: The nodes transmit the data packets whenever a new sensor measurement is taken. The contents of the data packet are shown in figure 6. A node bases its routing decisions on two metrics, namely hop count and link quality. Using hop count as the routing metric ensures that the packet is always sent in the direction of the base station. The procedure for choosing the next hop neighbor is as follows. First the source node selects all the nodes having hop count lesser than itself from its neighborhood table. It then chooses a node having the highest link quality among them and transmits the data packet to it. In case there is more than one node with the same link quality, it chooses a node which has the largest history variable value. Each time the node transmits the data packet to a next hop neighbor, it modifies the history variable of that neighboring node. If the data packet is sent successfully, it shifts a *1* left from the MSB to the LSB of the

history variable. A *0* is shifted left from the MSB to the LSB of the history variable if the transmission was unsuccessful. A transmission is considered unsuccessful only if all the ten confirm delivery packets fail to receive an acknowledgement. Nodes with a history variable value of zero are deleted from the neighborhood table because there is a high probability that the node has failed. In case the source node does not find a suitable neighbor or if all the nodes in the neighborhood table have exhausted its maximum limit of confirmed delivery, then the source node transmits the link estimate packets to rediscover its neighbors.

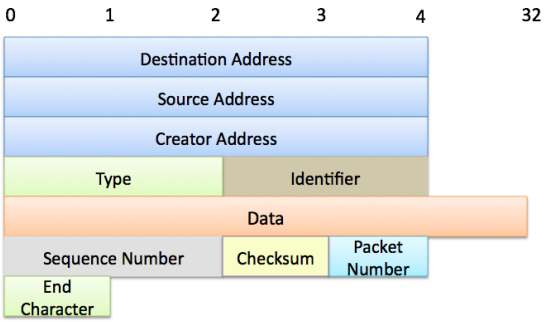


Fig. 6. Data packet

The node forwards the data packet if the destination address does not match the current node address. The same procedure is followed to find a next hop neighbor, except for a minor difference. While selecting the next hop neighbor, the node does not choose the source node or the creator node as the next hop destination. A creator node is a node from whom the data pack was initially originated from. This ensures that there are no routing loops in the network.

4) Network Maintenance and Neighborhood Table Management: Network maintenance plays a very crucial role in improving the network lifetime and reliability of the system. The nodes dynamically remove the failed nodes from their neighborhood table by keeping track of the history variable. The routing protocol might not work efficiently if the nodes are physically moved after the network is setup. Therefore the base station sends a new setup packet periodically to refresh the network and to facilitate any changes made to the physical position of the nodes. As the battery voltage of the node is used for calculating the link quality, it is important that the nodes maintain an updated neighborhood table. The battery value is updated from the link estimate packet and the data packet.

5) Configuring the Node Settings: In many sensor network applications, the base station needs to be able to transmit packets to the sensor nodes to configure parameters such as current time, sample time, sensor measurement settings, relay settings and etc. This functionality is not supported in many well known sensor network protocols. The DSRP supports the feature by providing a configuration packet. The structure of this packet is similar to the data packet. The base station broadcasts the configuration packet to all its neighbors. When a node receives this packet, it checks the packet for a new sequence number and broadcasts it if it is a new packet. The check for the sequence number prevents flooding of these packets and allows for a centralized broadcast.

IV. Experimental Evaluation

The DSRP was written in ANSI C and implemented on the CMU SensorWeb platform which uses an ATMEGA1281 as a processor and a XBee Pro XSC as a radio. Tests were conducted to asses its performance. The setup consisted of five sensor nodes and a base station. A XBee Pro XSC development board was used to monitor all transmissions. The test site was similar to the real world operating environment with both natural and man made obstacles on the field. The snapshots of the network layout in the following figures were taken at an eye altitude of 1.02kms. The node markers in the snapshots show the hop-count of the nodes from the base station. A brief description of the tests and the results obtained are discussed in the following section.

A. Three Node Multi-Hop Test

This test verifies the basic multi-hop capability of the DSRP. Node1 was deployed at an arbitrary distance from the base station. Node2 and Node3 were deployed such that they were within the transmission range of Node1 but not within the transmission range of the base station. Node4 was placed such that it was only able to communicate with nodes 2 and 3. Figure 7 shows the different stages of the routing protocol.

Node4 is three hops away from the base station. To choose the next hop neighbor it has two options, Node2 and Node3. It chooses Node2 as the next hop because it has a higher link quality than Node3. For the second hop, Node2 again has two options to choose from, either Node1 or Node3. It does not consider Node4 again because the source node and the creator node are excluded from being the next hop neighbor for forwarding packets. Node2 chooses Node1 as the next hop destination because it has a lesser hop count than Node3. Node1 then forwards the packet to the base station. Node1 does not send the packet to Node3 because it has a higher hop count than the base station. Similarly the data packet from Node3 is first forwarded to Node1 and then the base station. Similar to the three node multi-hop test, we also performed a four and five node multi-hop test, which verified the protocols capability to handle networks with large hop counts. The latency for each hop was less than 0.6 seconds.

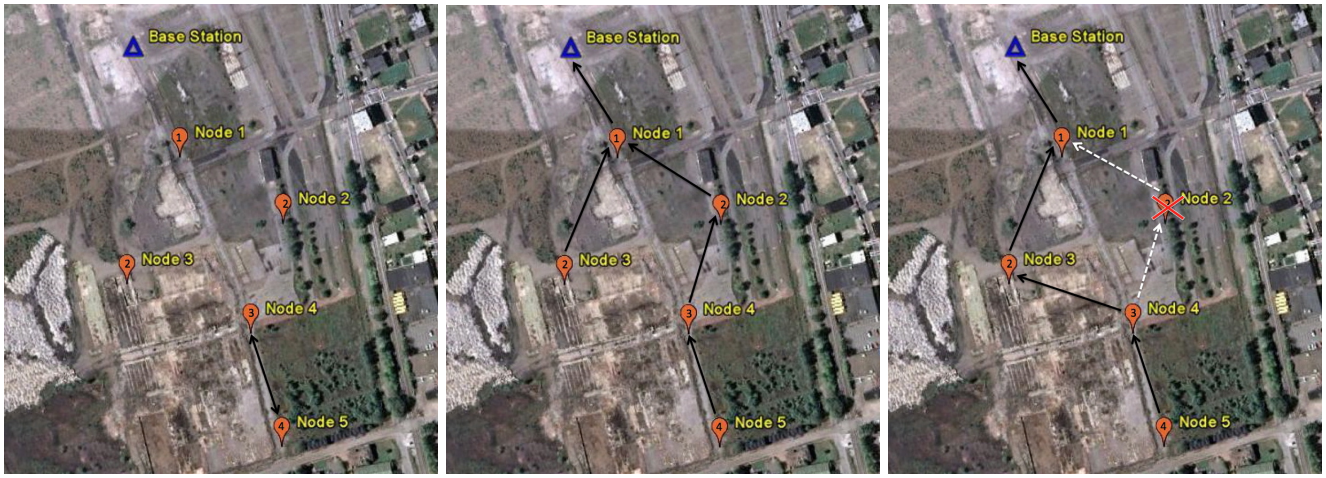
B. New Nodes Joining the Network after Network Setup

One of the biggest challenges in multi-hop routing is how a new node can join the network without resetting the entire network. To test this feature, a new node (Node5) was powered on after the network setup. It was placed such that it could communicate only with Node4. It initializes its hop count as one more than the lowest hop count of its neighbors. In this case as it has only one neighbor, it initializes its hop count as Node4's hop count plus one. The new node now has a hop count of four and it is four hops away from the base station. The node also initializes the time from the link estimate packet that it received from Node4. It then starts transmitting data through the path, Node5 - Node4 - Node2 - Node1 - Base Station. Figure 8(b) shows the multi-hop paths.

One potential disadvantage of initializing the new nodes time from that of its neighbors is the clock drift factor. As the

(a) Link quality estimation. (b) Network setup. (c) Multi-hop routing.

Fig. 7. Three node multi-hop test.



(a) (b) (c)

Fig. 8. (a)Link quality estimation of the new node; (b)Multi-hop routing with the new node; (c)Rerouting the packets upon node failure

clocks in the nodes drift, the new node gets initialized with the neighbors time plus the clock drift. However the base station transmits a time update packet if the drift becomes very large.

C. Rerouting Packets upon Node Failure

Another feature of this protocol is the dynamic rerouting of packets incase of node failure. Figure 8(c) illustrates this feature. To depict node failure, Node2 was turned off suddenly. Node4 detects the failure of Node2 when it does not receive an acknowledgement for the transmitted data packets. After it has transmitted ten data packets and exhausted its confirmed delivery count, it updates the history variable of Node2. A zero is shifted left from the LSB to the MSB in the history variable of Node2. Node4 then recalculates the next hop destination and chooses Node3. In the next cycle Node4 again tries to transmit the data packet to Node2 and if it fails it recalculates the address. Node4 will remove Node2 from the neighborhood table when the history variable of Node2 becomes zero.

D. Network Setup Time

Figure 10 shows the change in the network setup time for the variation in the number of nodes in the network. It can be

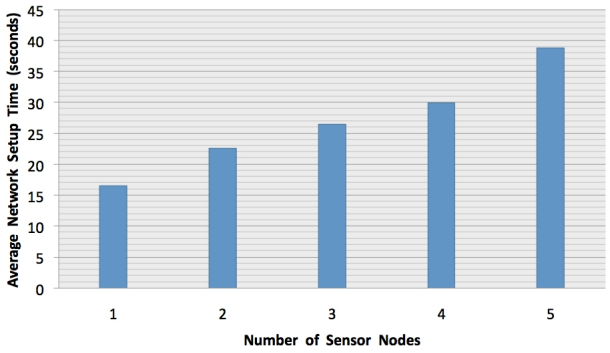


Fig. 9. Network setup time

seen from the graph that as the number of nodes in the network increase, the time for the network setup also increases. The network setup time here can be defined as the time taken for both link quality estimation and hop count initialization. This also includes the AT command response time, which was about 2.5 seconds. The average network setup time was only 36.48 seconds for a network with five nodes.

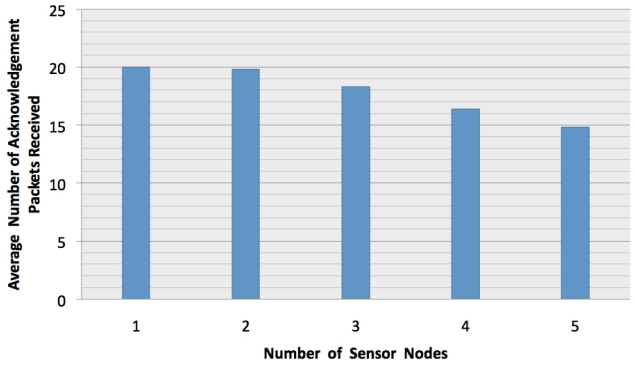


Fig. 10. Number of acknowledgement packets received during link quality estimation

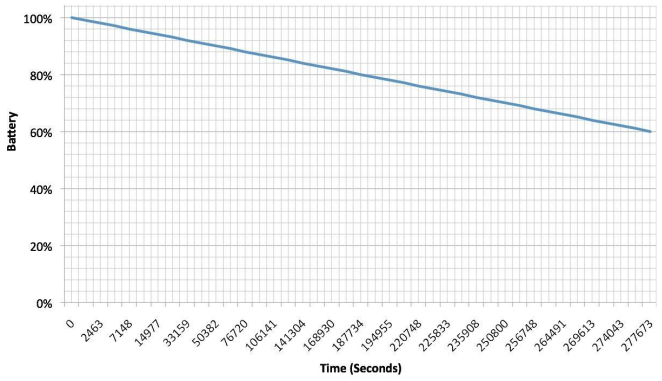


Fig. 11. Battery lifetime (with an abnormally high data rate)

E. Number of Acknowledgements Received During Link Quality Estimation

For this experiment all the nodes were within the transmission range of the base station so they all start transmitting the link quality estimation packets at the same time. As the density of the nodes in the network increases the number of packet collisions also increase. Hence the link quality estimation packets does not reach all the nodes. It was found that if the value of τ is increased then the number of collisions in the network decreases, but this increases the network setup time in turn. For a five node network there are a total of 100 packets exchanged in less than 30 seconds. The network was tuned and the value of τ was set to 50ms. It was then found that the average number of packet collisions during the link quality estimation stage was about two for each node.

F. Battery Lifetime

The battery lifetime graph is very important for analyzing the network performance. The energy source for the sensor nodes were two 1.5 volt ZnCl D cell batteries. A CMU node was attached with five analog sensors and the sample time was set to 5 seconds. The graph in figure 11 shows the battery profile for the testing period. The sensor nodes began to fail when its battery reached to about 60 percent. However it should be kept in mind that this was obtained with a very high sampling rate. In most applications the normal sampling rate is 5 minutes, under this condition the battery would last for more than 6 months.

TABLE I
COMPARISON OF DSRP WITH EXISTING WSN ROUTING PROTOCOLS.

	Scala-bility	Multi-Path	Compl-exity	Power Usage
Direct Diffusion	Limited	Yes	Low	Low
LEACH	Good	Yes	Moderate	High
MCFA	Good	No	Low	Moderate
PEGASIS	Good	No	Moderate	High
DSDV	Good	Yes	Moderate	High
AODV	Good	Yes	Moderate	Moderate
DSRP	Good	Yes	Low	Moderate

V. CONCLUSION

An application specific hybrid routing protocol known as Distributed SensorWeb Routing Protocol (DSRP) was designed and implemented on the CMU SensorWeb platform. Its validity was analyzed by performing field tests and the results were quantified. The protocol is simple and is portable enough to be implemented on existing WSN platforms. In the DSRP, the data is transmitted to the next hop neighbor having the lowest hop count along with the strongest link quality and highest history variable value. Some of the important features of the DSRP include symmetric link quality estimation, fault tolerance, hotspots prevention, dual routing metrics and two way communication. Future work may include embedding a sleep scheduling algorithm to improve the network lifetime and energy efficiency. As the protocol requires the nodes to maintain a neighborhood table and since the on chip memory is limited, the reliability can be improved tremendously by using the external memory on the nodes for maintaining the table.

REFERENCES

[1] J. N. Al-Karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey", *IEEE Wireless Communications*, vol. 11, no. 6, Dec. 2004, pp. 6 28.

[2] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", *Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS)*, 31 January - 4 February 2005, Hawaii, USA, pp. 1 - 10.

[3] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", *Proceedings of the 6th International Conference on Mobile Computing and Networking (Mobicom)*, Boston, USA, August 2000, pp. 56 - 67.

[4] F. Ye , A. Chen, S. Lu and L. Zhang, "A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks", *Proceedings of the IEEE International Conference on Computer Communication and Networks (ICCCN)*, 15 - 17 October 2001, Phoenix, USA, pp. 304 - 309.

[5] M. D. Yarvis, W. S. Conner, et al, "Real-world experiences with an interactive ad hoc sensor network", *Proc. of the International Conference on Parallel Processing Workshops*, August 2002, pp. 143 - 151.

[6] Hsin Mu Tsai, Nawapom Wisitpongphan and Ozan K. Tonguz, "Link-Quality Aware Ad-Hoc On-Demand Distance Vector Routing Protocol", *Wireless Pervasive Computing 2006 1st International Symposium on 16 - 18*, January 2006.

[7] Z. Sun, X. Guang Zhang, H. Li and A. Li, "The Application of TinyOS Beaconing WSN Routing Protocol in Mine Safety Monitoring", *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications 2008*, October 2008, pp. 415 - 419.

[8] Abhinav Valada, David Kohanbash, George Kantor, "Design and Development of a Wireless Sensor Network System for Precision Agriculture", *tech.report CMU-RI-TR-10-21, The Robotics Institute, Carnegie Mellon University*, Pittsburgh, PA, 2010.