Comparison of two wireless ad hoc routing protocols on a hardware test-bed

Shweta Bhandare, Sheetalkumar Doshi, Timothy X Brown, Sagar Sanghani
University of Colorado, Boulder, 80309-0530
{bhandare, doshi, timxb, sanghani}@colorado.edu

Abstract— **In this paper, we compare the Dynamic Source Routing (DSR) protocol and our Energy Aware Dynamic Source Routing protocol (EADSR). The implementation of the routing protocols is carried out using the Click modular router infrastructure on laptops with wireless ethernet cards running Linux. We demonstrate the working of both ad hoc routing protocols through our experiments and highlight the energy efficient behavior of EADSR as compared to DSR.**

I. Introduction

An ad hoc network is a collection of wireless mobile nodes that form a temporary network. Communication is established among the nodes without the use of centralized infrastructure or administration. In an ad hoc network, each node acts as both an end-host and as a router due to limited propagation range of each node's wireless transmission. These nodes are usually PDAs or laptops that are battery operated and are equipped with wireless ethernet cards. To maximize the lifetime of the node, conservation of battery power is critical. Research is being carried out to reduce energy consumption at various levels e.g. at the operating system level, physical level, and MAC level. Energy can be conserved at the routing level by designing cross-layered protocols and deploying low-power routing algorithms that use the power cost of the route as a metric for routing packets. Prior work has investigated energy aware ad hoc routing protocols. These protocols are tested predominately using software simulation. Very few implementations have demonstrated the effectiveness of the protocols on a hardware test-bed: one example is the work by Gomez et al. [1].

In a prior paper [2] we described the general features required to enable energy aware routing and a modified version of the Dynamic Source Routing (DSR) protocol that incorporates these features which are (i) Energy based link cost with transmit power control, (ii) Energy based power route selection, (iii) Energy based route discovery, and (iv) Energy based route maintenance.

We denote this protocol the Energy Aware DSR (EADSR). In this paper we conduct experiments to demonstrate these energy aware features of the EADSR protocol implementation and compare its performance to DSR on a hardware test-bed. Earlier work has considered energy-aware routing in terms of fairness and network lifetime in [3] and [4]. To simplify the exposition in this paper, we equate Energy Aware with choosing routes with minimum total transmit power.

We describe our implementation using the Click modular router [5] infrastructure on a hardware test-bed consisting of laptops with 802.11b wireless ethernet cards in Section II. In Section III, we discuss the hardware setup required to test the two ad hoc routing protocols. The results of the protocol testing along with a comparison of the two protocols are discussed in Section IV.

II. DSR and EADSR Click implementation

We first describe the DSR and EADSR protocols that we are comparing as well as the routing infrastructure they are built with.

A. Routing Infrastructure

We chose the Click modular router as the routing software infrastructure to implement the protocol on the test-bed. The Click-based router can be configured to run at user level which allows the packets to be manipulated in the user-space and also allows for the re-insertion of the packets into the kernel stack.

Click builds routers from a collection of modules called elements. Every element controls a specific aspect of the router. They handle the functions of communicating with the network devices, packet modifications, and packet scheduling. These elements are inter-connected to create the router configuration file. The Click program reads the configuration file, and sets up the router accordingly. One instance of a successful Click implementation of an ad hoc routing protocol is the AODV implementation done by Tornquist et. al. [6].

Our Click design of the DSR and EADSR protocol implementations have been described in detail in [2]; we highlight the main points in this paper.

B. Dynamic Source Routing protocol implementation

This section describes our implementation of DSR using Click which provides a basis for describing our energy aware modifications to the protocol in the next section. Our design conforms to the DSR protocol defined in the IETF draft [7]. Readers unfamiliar with DSR are referred to this document. Fig. 1 enumerates the packet structure. The main Click elements for the DSR router are as follows

- The RouteTable element choses routes in the DSR protocol. It includes a link cache for saving the routes obtained during route discovery. Since routing cost is the hop count, each link has cost one. The minimum cost route is selected using Dijkstra's algorithm.

IP Header	DSR Fixed Header	DSR Source Route header	DSR Source Route Addr1 Addr2… AddrN	DSR ACK Request	Data

(a) Data Packet format in DSR

IP Header	DSR Fixed Header	DSR Route Request Header	Request Addresses Addr1 Addr2 ...AddrN

(b) DSR Route Request Packet format

IP Header	DSR Fixed Header	DSR Route Reply Header	Reply Addresses Addr Src Addr1..AddrN Addr dst	DSR Source Route Header	DSR Source Route Addr1…AddrN

(c) DSR Route Reply Packet format

IP Header	DSR Fixed Header	DSR ACK Header

(d) DSR ACK Packet format

IP Header	DSR Fixed Header	DSR Route Error Header	Unreachable Node Address	DSR Source Route Header	DSR Source Route Addr1…AddrN

(e) DSR Route Error Packet format

Fig. 1. Packet formats in the Click DSR implementation

The Send Buffer saves packets with an unknown destination route. For each saved packet it initiates a route request every second until a route is found and the packet is forwarded or a timer expires and the packet is discarded. It also processes incoming data packets, route replies, and route errors, and sends out ACK replies.

- The ACKTable element is responsible for link maintenance. It stores a clone of every transmitted data packet. The clone is discarded when an ACK reply is received. If an ACK reply is not received within a timeout period, the link is declared as invalid and the packet is forwarded to the RouteTable which removes that particular link from the link cache.
- The RequestTable element sends out route request packets when no route is available. Route requests are repeated according to a back off algorithm until it receives a route reply or reaches the maximum number of requests after which the entry in the RequestTable is removed.
- The ARPTable element looks up the MAC Address for a given IP Address. On a successful lookup, the element adds an ethernet header to the incoming packet and forwards it on to an output port.

The elements are connected together in a configuration file as shown in Fig. 2 to make up the DSR router. This router configuration runs on all the nodes to form the DSR ad hoc network.

C. Energy Aware DSR router implementation

In this section we enumerate the changes made to the DSR router to make the protocol energy aware. The link

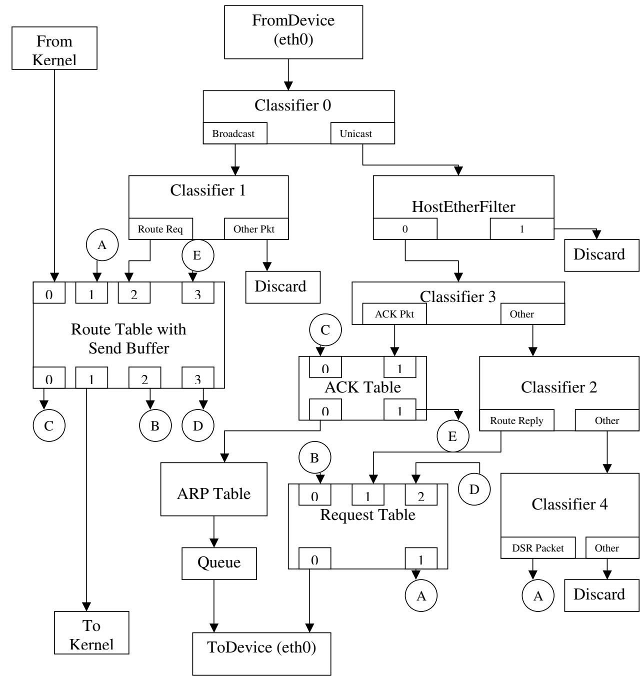

Fig. 2. The Click DSR router

IP Header	DSR Fixed Header	DSR Source header	DSR Source Route Addr1 Addr2..AddrN	**DSR Source Powers ToPwr1… ToPwrN**	DSR ACK Request	Data

(a) Data Packet format in Energy Aware DSR

IP Header	DSR Fixed Header	DSR Route Request Header	Request Addresses Addr1 Addr2..AddrN	**Power Values ToPwr1…ToPwrN**

(b) Energy Aware DSR Route Request Packet format

IP Header	DSR Fixed Header	DSR Route Reply Header	Reply Addresses Addr Src Addr1.. AddrN Addr Dst	**Reply Powers ToPwr Src ToPwr1 … ToPwrN ToPwr Dst**	DSR Source Route Header	DSR Source Route Addr1… AddrN	**DSR Source Route Powers ToPwr1.. To PwrN**

(c) Energy Aware DSR Route Reply Packet format

IP Header	DSR Fixed Header	DSR ACK Header

(d) Energy Aware DSR ACK Packet format

IP Header	DSR Fixed Header	DSR Route Error Header	Unreachable Node Address	DSR Source Route Header	DSR Source Route Addr1…AddrN	**DSR Source Route Powers ToPwr1..ToPwrN**

(e) Energy Aware DSR Route Error Packet format

Fig. 3. Packet formats for Energy Aware DSR implementation: Added fields are highlighted

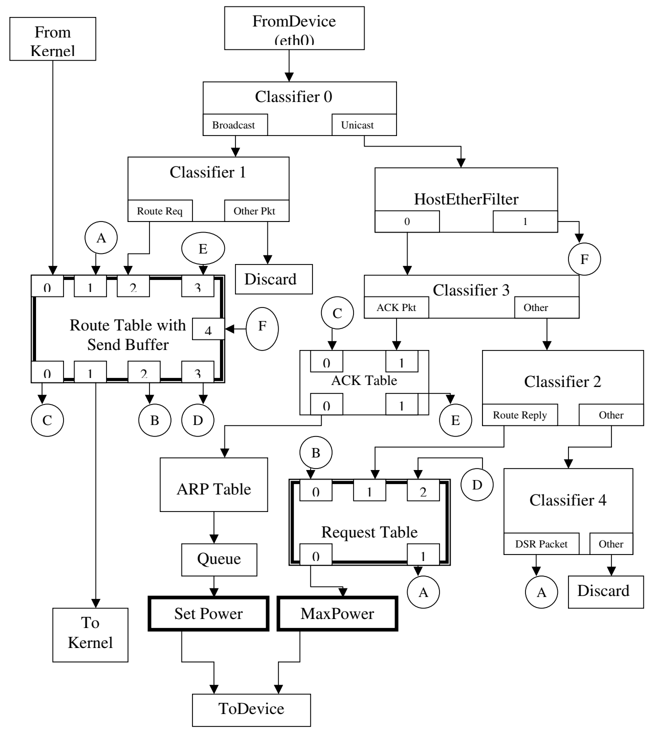

Fig. 4. Energy Aware DSR implementation in Click: New and modified elements compared to DSR are highlighted

metric is the minimum transmit power required for successful communication for a given link. These power values need to be advertised during route discovery and stored in the link cache as link costs. This is achieved by adding a power header to the original DSR packet structure. The header is a list of power values that are the minimum transmit powers for each link. The packet formats for energy aware DSR are shown in Fig. 3.

The Minimum Transmit Power is computed in dB using

$$P_{TXmin} = P_{TX} + P_{Thresh} - P_{RX} + M_D \qquad (1)$$

where P_{TXmin} is the minimum transmit power, P_{TX} is the power at which the packet was transmitted, P_{RX} is the received signal strength of the packet, P_{Thresh} is the sensitivity threshold of the node and M_D is a margin added to ensure successful communication under channel variability.

Two existing elements were changed to enable the minimum transmit power metric.

The RouteTable finds the received signal strength in dBm through ioctl calls to the card driver, which maintains a received signal strength table indexed by the MAC Address. The minimum transmit power is found using (1). This power value for every link is propagated by each node by adding a field to the header with the value. This uses the optional header feature of DSR. The RouteTable adds these headers called power headers. It also extracts these values from the headers and adds the information to the link cache. An extra input port that processes snooped packets and discovers new minimum energy routes is added. When new routes are found it sends out a gratuitous route reply to the source of the packet. This process also involves constantly monitoring the power cost of each link in the route. The DSR source header is modified to include an additional flag that indicates a change in the power costs of the links in the route. The flag is set when the difference in the power values is greater than M_Δ dB. The destination node checks this flag and sends out a gratuitous reply to the source informing it about the new costs of the route.

The RequestTable element is also modified to implement minimum transmit power computation to calculate the minimum transmit power value of the link. This value is then added to the power list in the route request packet when it has to be rebroadcast. The RequestTable element now propagates the new energy metric through the network.

Two additional elements were added to the implementation of Click DSR Router to carry out transmit power control.

- The SetPower element sets the packet transmit power to P_{TXmin}. It is used to transmit ACK, data, route reply and route error packets at controlled power.
- The MaxPower element sets the transmit power to the maximum value. It is used only for route request packets.

III. Experimental Setup

This section describes the experimental setup we used to carry out the testing of the EADSR protocol. The hardware test-bed setup is described in [8] and consists of laptops equipped with Cisco Aironet 350 series wireless ethernet cards. Each node is comprised of a packet protocol development environment based on Click [5] running Linux Red Hat 7.2. An external antenna with attenuation of 20 dB was added to each node and they were arranged on tables having a total area of 16 feet that comprised the test-bed. The attenuators shrunk the range of a node to 1.5 meters, reduced the range to a controlled environment within a room, and ensured uniformity in the test procedure. To vary the path loss between the antennas the distance between them was changed and was equivalent to moving the nodes. The antenna diversity of the Cisco 350 series wireless ethernet cards was set to left. This ensured that the packets were received and transmitted using the same antenna and there was no other communication path present. Traffic was generated at the source node to the destination node using ping packets. The 802.11b RTS/CTS signaling was disabled. The ACK packets were transmitted at full power. The value of the margins M_D and M_Δ discussed in Section II-C were set to 6 dB and 4 dB respectively.[1] The data for the results is based on 10 runs of an experiment. In each run, the nodes are reset, 110 packets are sent, and the data recorded on the routes of the last 10 packets. The median values for these 100 packets is plotted with error bars. The protocol requires a measure of the Received Signal Strength (RSSI) for the received packets. The value of the RSSI varied

[1]Setting these values is outside the scope of this paper. They were set conservatively for this paper based on initial experiments

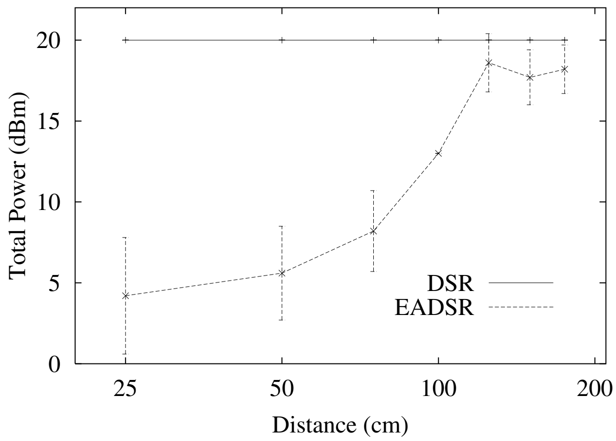

Fig. 5. Test 1: Link Power Adjustment:Power chosen by EADSR and DSR to transmit vs distance d. Error bars (dBm)

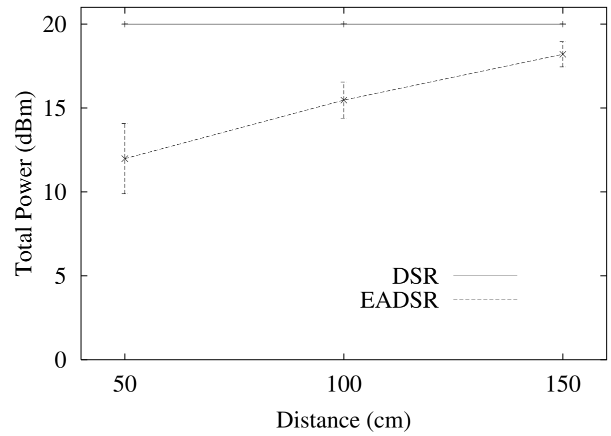

Fig. 6. Test 2: Minimum Power Route Choice: Comparison of the routes selected by DSR and EADSR vs distance d. Error bars (dBm)

from packet to packet. To smooth the readings we maintained a window of previous five packets received from each node. We selected the median value from the window as the value of the RSSI for that packet. A different techniques of selecting the best value of RSSI is discussed in [9] where the authors distinguish true mobility events from spurious multipath noise.

IV. Protocol Testing and Results

In this section, we discuss the four experiments conducted to demonstrate a comparison of the protocols and highlight the energy efficient features of the EADSR protocol.

A. *Test 1: Link Power Adjustment*

- Objective: To demonstrate the ability of EADSR to determine and adjust the link power due to varying path loss (i.e. distance).
- Test procedure: We placed both the source and destination node along a straight line. We varied the distance d between them by moving the antennas along the line. For each d, the source node initiated ping packets and we observed the behavior of the routing protocols.
- Result: EADSR estimates the minimum transmit power required for that particular distance and uses the power value to transmit the packets to the destination. As shown in Fig. 5, the value of the transmit power increases as the distance d between the nodes increases.
  When the distance d is 25 cms, EADSR estimates the minimum transmit power required for the source node to communicate successfully with the destination to be 4 dBm and achieves successful communication between the source and destination node. As the distance between the nodes is increased, EADSR realises that the nodes can no longer communicate at a low power value and it estimates the value of the minimum transmit power required for successful communication. It does this for every value of d. As seen in the Fig. 5, the DSR protocol always transmits at 20 dBm for all distances thus expending more energy compared to the EADSR protocol.

B. *Test 2: Minimum Power Route Choice*

- Objective: To demonstrate that EADSR selects the minimum power route among the available routes.
- Test procedure: In this experiment, we used source node a, intermediate node b, and destination node c. These nodes were placed along a straight line. The position of node b was fixed and was placed equi-distant from the source and destination node. We varied the positions of nodes a and c along this line such that they were each distance d, from the intermediate node b. At each position, ping packets were initiated from the source node to the destination node. We observed the behavior of the DSR and EADSR protocol.
- Result: We observed that both protocols discover the routes a-c and a-b-c. DSR uses the minimum hop route a-c to route the packets and transmits them at 20 dBm. EADSR discovers that the route a-b-c is the minimum power path and uses this route to transmit packets. As seen in Fig. 6, as d increases the total transmit power used by EADSR to route packets increases. When d equals 50 cms, EADSR uses a total transmit power of around 11 dBm to transmit packets from the source node a to destination node c via intermediate node b. However, the DSR protocol uses 20 dBm to carry out the same task using the direct hop route a-c. Similarly for the other two values of the distances, EADSR consumes less transmit power compared to DSR.

C. *Test 3: Route Power Maintenance*

- Objective: To demonstrate the ability of EADSR to discover newly formed lower power routes and maintain them.
- Test procedure: In this experiment, we used source node a, intermediate node b, and destination node c. The

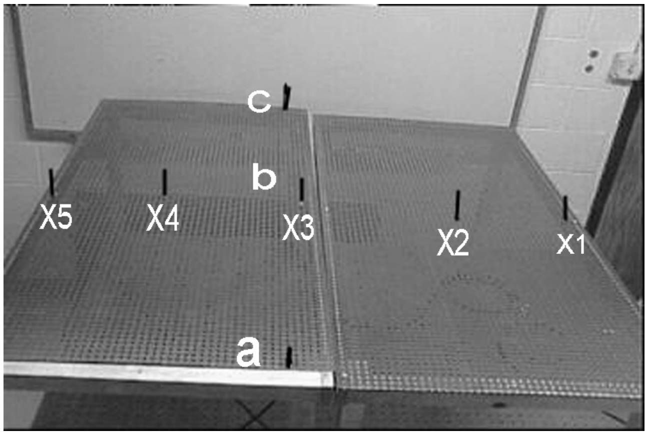

Fig. 7. Test 3: Route Power Maintenance Experimental Setup

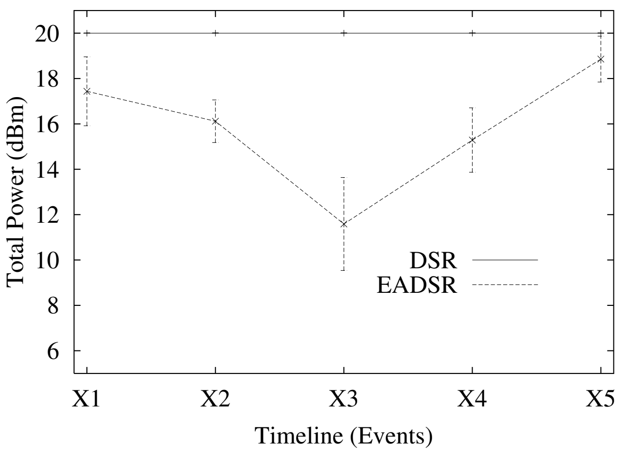

Fig. 8. Test 3: Route Power Maintenance: Comparison of the routes selected by EADSR and DSR in the presence of the intermediate node. Error bars(dBm)

distance between the nodes a and c were fixed during the course of the experiment and the nodes were placed along a straight line. As shown in Fig. 7, the intermediate node b was placed along a line perpendicular to the line joining nodes a and c. Initially the router on node b was turned off and hence was out of transmission range from nodes a and c. Ping packets were initiated from node a to node c. Node b was turned on and was moved from positions X1 to X5 as shown in the figure. We observed the behavior of the protocols at each position of node b.

- Result: We observed that the DSR protocol used the direct route a-c to transmit packets at full power of 20 dBm from source to destination throughout the experiment.

  Initially when node b is not within transmission range, EADSR uses the direct route a-c at 20 dBm transmit power. When node b is placed at location X1 and is turned on, it snoops on the Data/ACK packets that are exchanged by nodes a and c.

  Using the energy aware route maintenance feature, it discovers that it lies on a lower power route a-b-c and sends a gratuitous reply to node a informing it about this route. Node a now uses this route to transmit packets at total transmit power of around 17 dBm to the destination. Fig. 8 shows the value of the total transmit power at different positions of the intermediate node b. At each position DSR always transmits using the direct route a-b at 20 dBm.

D. Test 4: Minimum Power Route Discovery

- Objective: To demonstrate the ability of EADSR to discover the low power routes not found through the initial route discovery.
- Test procedure: We used source node a, destination node d, and intermediate nodes b,c. The nodes were placed along a straight line at intervals of 31 cms from each other.

  Node a could not communicate to node d directly but could communicate via intermediate nodes b or c. Ping packets were initiated from source node a to destination node d.
- Result: We observed that the DSR protocol discovers routes a-b-d and a-c-d and uses the route a-b-d to transmit packets at a total transmit power of 200 mW.

  EADSR also discovers routes a-b-d and a-c-d with total transmit powers 200mW and 120 mW respectively. It selects the lower power route a-c-d. Node a transmits data packets using packet exchange between nodes a and c. It finds that it lies on a lower power route a-b-c-d with a total transmit power of 30 mW and sends a gratuitous reply to node a. Node a caches this reply and uses this route to transmit packets to node d. Thus EADSR discovers the lowest power route to the destination in addition to the other routes. Fig. 9 demonstrates the timeline of packet exchange in this minimum power route discovery procedure.

V. Discussion

The experiments described in Test 2, Test 3, and Test 4 demonstrate that EADSR uses routes that consume as much as 6-10 times lower power than the route used by DSR and the experiment described in Test 1 demonstrates 30 times lower power than DSR. This occurs at the cost of additional control packets. Table I shows the power budget for the transmission of a single data packet for the experiment with the greatest overhead, Test 4. We conservatively assume equal packet lengths for all packet types in this computation. As seen from the table, despite the additional route discovery overhead, EADSR still consumes lower power than DSR starting with the first packet. Subsequent data packets transmitted using EADSR will result in further power savings as shown in Table I. These power savings achieved at the routing layer can be converted to energy savings in transmission, by considering actual packet lengths.

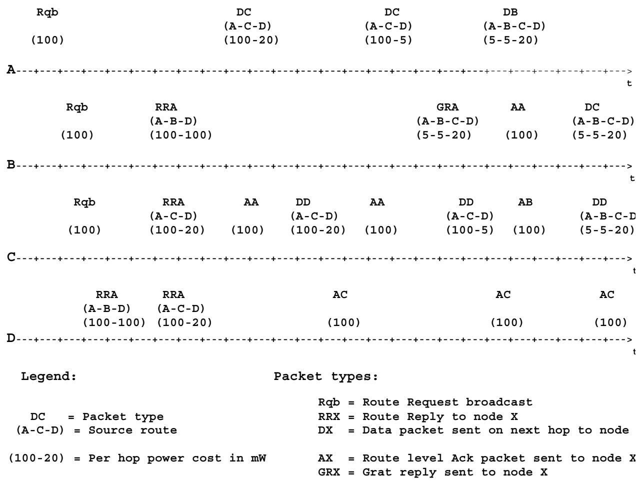

Fig. 9. Test 4: Minimum Power Route Discovery in EADSR: Timeline of the sequence of events by EADSR to select the minimum power route

The ACK packets were transmitted at full power to ensure robust behavior of the protocol. However, if they are transmitted by adding a margin M_A that equals 9 dB[2], the power per subsequent packet as shown in Table I will effectively reduce to 100 mW which is four times lower power than DSR.

At the MAC layer, the 802.11b cards are not the best choice in terms achieving energy efficiency as observed in [10]: they have high receive and idle powers which are comparable to the power required for transmission. We use them for our protocol development due to their widespread support and availability. To translate power savings to overall energy savings would require a regime where transmit power dominates energy expenditures; for instance high power longer range scenarios or a more energy efficient MAC layer protocol.

Another observation was that using multi-hop routes led to lower total transmit power values, but increased the round trip time (RTT) of the ping packets. This indicates that real time traffic requirements may limit power savings.

The variations in the minimum transmit power measurements occured despite a static setup and are displayed as error bars in Fig.5, Fig.6, and Fig.8. The routing protocol was robust and was able to maintain connectivity even with this variability.

VI. Conclusion

We have successfully implemented and compared the DSR and EADSR protocols on a hardware test-bed. The Click implementation of these protocols makes it easy to incorporate additional features to the existing protocols.

Experimental results demonstrate that DSR transmits packets at full power using minimum hop routes. This makes the implementation energy-inefficient. EADSR uses power cost of the links as a metric to route packets. It discovers and selects

[2]Setting this value is outside the scope of this paper. This is based on a conservative assumption.

TABLE I

Power Budget of DSR vs EADSR

Packet Type / Protocol Type	DSR (Number of packets * Power transmitted at (mW))	EADSR (Number of packets * Power transmitted at (mW))
Route Request	3 * 100	3 * 100
Route Reply	2* (100+100)	1*(100+20) + 1*(100+100)
Gratuitous Reply	0	1* 5
Data Packet 1	1*(100+100)	1*(120)
ACK packet	100+100	100+100+100
Total Power First packet (mW)	1100	1045
Power per Subsequent packet (mW)	4*100 = 400	(3*100 + (20+5+5)) =330

the lowest power route using minimum transmit power to route the packets. This makes the EADSR protocol more power efficient as compared to DSR. Assuming an energy efficient MAC layer, the power savings obtained using EADSR can be translated to energy savings.

Future work involves measuring and quantifying the energy savings obtained using EADSR. We also plan to increase the number of nodes in the network and carry out more extensive testing of the protocols.

VII. Acknowledgment

The work in this paper is supported by NSF Grant ANI-0082998. We would like to thank Jean Tourrilhes, Javier Achirica, Peter K. Lee for their help in modifying the device driver for the Cisco 350 Aironet Series card and the Wireless Tools used. We would also like to thank Micheal Neufeld for his help at various stages in the project.

References

[1] J. G. Andrew, "Conserving transmission power in wireless ad hoc networks." [Online]. Available: citeseer.nj.nec.com/478321.html

[2] S. Doshi, S. Bhandare, and T. X. Brown, "An on-demand minimum energy routing protocol for a wireless ad hoc network," *Mobile Computing and Communications Review*, vol. 6, pp. 50–66, 2002.

[3] T. Brown, Q. Zhang, and H. Gabow, "Maximum flow life curve for a wireless ad hoc network," 2001. [Online]. Available: citeseer.nj.nec.com/brown01maximum.html

[4] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *INFOCOM (1)*, 2000, pp. 22–31. [Online]. Available: citeseer.nj.nec.com/chang00energy.html

[5] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The click modular router," *ACM Transactions on Computer Systems*, vol. 18, no. 3, pp. 263–297, August 2000.

[6] A. Tornquist, "Modular ad hoc routing algorithm using the click modular router," Master's thesis, University of Colorado at Boulder, 2001.

[7] D. Johnson, D. Maltz, Y.-C. Hu, and J. G. Jetcheva, "Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)ting," IETF MANET group," INTERNET DRAFT, November 2001.

[8] S. Sanghani, T. X. Brown, S. Bhandare, and S. Doshi, "EWANT: Emulated wireless ad hoc network test-bed," *IEEE WCNC*, March 2003.

[9] A. Sheth and R. Han, "A mobility-aware adaptive power control algorithm for wireless lans, a short paper," *IEEE CAS Low Power Workshop*, August 2002.

[10] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *IEEE INFOCOM*, 2001. [Online]. Available: citeseer.nj.nec.com/feeney01investigating.html