Opportunistic Overlay Multicast in Wireless Networks

Tan Le and Yong Liu

Department of Electrical and Computer Engineering

Polytechnic Institute of NYU

Brooklyn, New York, 11201, USA

Email: tle04@students.poly.edu, yongliu@poly.edu

Abstract—**Opportunistic Routing (OR) has recently been proposed to improve the efficiency of unicast in multi-hop wireless networks. OR exploits the broadcast nature of wireless transmission medium and opportunistically selects a relay path to deliver a packet to its receiver. To adopt OR in wireless multicast, the main challenge is to efficiently share opportunistic relay paths between multiple receivers. In this paper, we propose an opportunistic overlay multicast design for wireless networks, named Minimum Steiner Tree with Opportunistic Routing (MSTOR). In MSTOR, the source and receivers are connected by an overlay Steiner tree. The source multicasts packets along the overlay links of the Steiner tree to reach all receivers. The transmission of packets on each overlay link is controlled by unicast OR. We first propose an overlay construction algorithm based on the optimal "OR distance" between nodes. We then design the MSTOR protocol and implement it in OPNET by customizing the IEEE 802.15.4 modules. Through OPNET simulations, we study the performance improvement of MSTOR over several existing unicast and multicast routing schemes. Our results demonstrate that MSTOR can achieve a much higher multicast efficiency than the original unicast OR and the traditional minimum multicast-tree based schemes. MSTOR can be easily deployed for multicast in multi-hop wireless networks.**

I. INTRODUCTION

The increasing popularity of wireless devices and new wireless applications makes it important to deliver multicast services efficiently over multi-hop wireless networks. It is well-known that the general minimum-cost multicast routing problem is NP-hard. Wireless multicast additionally has to deal with interferences and collisions on volatile wireless links. While some attempts have been made to achieve a high efficiency in wireless multicast, the research field is still largely open. In parallel, Opportunistic Routing (OR) [1], [2], [3], [4], [5] has recently been proposed to improve the efficiency of unicast in multi-hop wireless networks. OR exploits the broadcast nature of wireless transmission medium and opportunistically selects a relay path to deliver a packet to its destination. By exploiting opportunistic packet receptions, OR can significantly reduce the number of transmissions necessary to deliver a packet. It is therefore tempting to adopt OR to improve the efficiency of wireless multicast. However, the adoption of OR in multicast is not straightforward. The main challenge is how to efficiently share opportunistic relay paths between multiple receivers.

Some limited efforts have been made to adopt OR in multicast. In MORE [5], a source node first calculates for each receiver the forwarder set and the expected number of transmissions of each forwarder based on the link ETX metric [6]. Then the forwarder sets for multiple receivers are merged and the transmission credits of each forwarder are updated. By

deploying random linear network coding, coded packets are pushed from the source through the merged forwarder sets to reach all receivers. The integration of network coding with opportunistic routing simplifies the coordination between wireless nodes. However, the merge of forwarder sets for different receivers is not necessarily efficient. As a result, MORE incurs high data transmission redundancy on source and relay nodes in multicast. In addition, network coding incurs considerable computation overhead. Packet encoding and decoding increase the packet latency at receivers. A more recent work of adopting OR in wireless multicast is Pacifier [7]. The source first calculates the shortest path tree to reach all receivers based on link ETX metric. To exploit the OR gain, a node not only receives packets from its ancestor nodes, but also can overhear packets from its sibling nodes. Similar to MORE, Pacifier also uses random linear network coding to improve multicast efficiency and simplify node coordination. In their experiments, Pacifier increases the average throughput over MORE by 171%. However, it still suffers very high source redundancy. Their experiments showed that for Pacifier, the source transmits on average 5.84 times the original data size while in MORE the source transmits on average 17 times the data size. In Pacifier, the construction of multicast tree does not explicitly take into account the opportunistic packet reception between sibling nodes. The constructed tree is therefore sub-optimal under OR.

In this paper, we employ a different approach, Overlay Multicast, to adopt OR in wireless multicast. Overlay networks have been widely employed to deliver multicast services on the Internet [8], [9]. Overlay multicast does not require native network multicast support, and can be easily deployed based on unicast primitives. For wireless multicast, we build an overlay Steiner tree to connect the source with all receivers. At the overlay level, packets are multicast to all receivers along the Steiner tree. Neighbors in the overlay tree are not necessarily neighbors in the underlying wireless network. Packet transmission on each overlay link is realized by a multi-hop unicast transmission in the underlying wireless network. To take advantage of opportunistic packet receiving, we employ unicast OR to transmit packets between adjacent overlay nodes. Figure 1 illustrates an example of opportunistic overlay multicast, with one source S, two receivers R_1, R_2, and one overlay Steiner node T. The rest of nodes in the figure are OR forwarders on overlay links. The thick arrows in Figure 1 are the overlay links of the overlay Steiner tree. The thin arrows in Figure 1 are the potential OR transmissions in the underlying wireless network along overlay links.
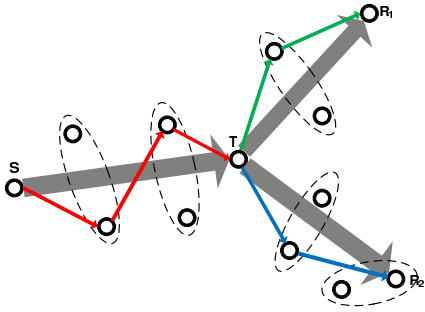
Fig. 1. Conceptual View of Opportunistic Overlay Multicast.

To minimize the total transmission cost to reach all receivers, we develop an algorithm to build the *Minimum Steiner Tree with Opportunistic Routing (MSTOR)*. More specifically, in a connected wireless network, we first calculate the shortest *OR distance* [10] between each pair of nodes. The OR distance between nodes i and j is the expected total number of transmissions to send a packet from i to j using unicast OR. By allowing any pair of nodes in the network communicate with each other using unicast OR, we construct a fully connected overlay graph out of the underlying wireless network. In the overlay graph, the source node calculates a minimum overlay Steiner tree with OR distances as overlay link weights. The overlay Steiner tree acts as a virtual backbone to deploy unicast OR from the source to receivers. The aggregate OR distance cost of the tree is the expected total number of OR transmissions to send one packet from the source to all receivers. Different from MORE and Pacifier, MSTOR constructs a multicast tree at the overlay level and exploits the gain of opportunistic receiving through unicast OR in underlying wireless network. MSTOR explicitly takes into account the cost of OR transmissions between nodes when constructing the overlay multicast tree. It can more effectively merge the OR relay paths to different receivers. To avoid computation and packet latency overheads, the current MSTOR does not employ network coding. Meanwhile, when such overheads are not major concerns, MSTOR can be augmented with network coding in a way similar to MORE and Pacifier. We leave it as a future research topic.

The rest of the paper is organized as follows. We briefly review the related work on unicast opportunistic routing in Section II. The MSTOR scheme is presented in Section III. The MSTOR protocol design and implementation in OPNET is presented in Section IV. The performance of MSTOR is evaluated through OPNET simulations in Section V. The paper is concluded in Section VI.

II. RELATED WORK: UNICAST OPPORTUNISTIC ROUTING

The original Opportunistic Routing algorithm, called ExOR was first proposed by S. Biswas and R. Morris in [1]. Instead of pre-selecting a multi-hop path, ExOR exploits the broadcast nature of wireless transmission and employs a dynamic sequence of forwarders to deliver a packet to its destination. In [2], the authors introduced a robust distributed opportunistic routing scheme base on ETX metric that can find the optimal

OR path from a source to a receiver. The work in [3] proposed a method to calculate the maximum throughput between two end nodes with OR in ad-hoc networks. The recent work from [4] studies the capacity of hybrid wireless networks under OR, which exploits high speed data transmissions in infrastructure network through base stations to improve the routing performance.

III. MINIMUM STEINER TREE OPPORTUNISTIC ROUTING

A. Network Model and Assumptions

We consider a network of N static wireless nodes, including one source node S, a set of $K < N$ receivers $\mathcal{R} = \{R_1, R_2, ..., R_K\}$, and $N - K - 1$ relay nodes. All nodes are equipped with radio interfaces and can communicate with neighbor nodes within their effective radio transmission ranges. Wireless links between neighbor nodes are not reliable. The success probability of packet transmission on a link is given by the Packet Reception Ratio (PRR). The PRR p_{ij} of link $\langle i, j \rangle$ theoretically depends on the distance between nodes i and j, node density and traffic around i and j, and the MAC scheduling scheme. As commonly assumed [11], packet losses on different links are independent.

B. Unicast OR Distance between Nodes

Given a wireless network, the length of a unicast OR path from node i to node j is the expected number of packet transmissions to send a packet from i to j along the OR path. The OR distance from i to j is defined as the length of the shortest OR path from i to j. In the recent work from Laufer et al. [10], they proposed the Shortest Anypath First (SAF) and Shortest Multi-rate Anypath First (SMAF) algorithms to calculate the optimal OR paths from every node in a network to one receiver with single and multiple transmission rates. These schemes were shown to have the same complexity as the Dijkstra's shortest path algorithm. For the single-rate case, given a graph $G = (V, E)$, the Algorithm calculates the shortest OR paths from every node in the network to a single receiver d. For every node $i \in V$, D_i is the distance of the shortest OR path to d. It consists of two parts. The first part d_{iJ} is the Anycast Link Cost from a transmitter node i to a neighbor set J. That is the expected number of transmissions from i to ensure at least one neighbor in J get the packet. The remaining path cost D_J is the expected number of transmissions to send one packet, originated from i, to get from J to the receiver d. The recursive process calculate the minimum value of D_i for every node $i \in V$ in a polynomial time. The complexity of the algorithm is $O(V log V + E)$. Unicast OR distance between nodes will be used to construct overlay Steiner tree in Section III.

C. Minimum Steiner Tree algorithm

We assume the wireless network is connected and any pair of nodes can reach each other using unicast OR. Given p_{ij} on all links, every node i calculates the shortest OR distance L_{ij} to any other node j based on the SAF algorithm presented in Section III-B. We construct a fully connected overlay network

Algorithm 1 Minimum Steiner Tree Algorithm

1: $X \Leftarrow Y$
2: $C = MST(X)$
3: **repeat**
4:     $X_{temp} \Leftarrow X$
5:     $i^* = \infty$
6:     **for** $i \in V \setminus X$ **do**
7:         $C_{temp} = MST(X \cup i)$
8:         **if** $C_{temp} < C$ **then**
9:             $C = C_{temp}$
10:            $i^* = i$
11:         **end if**
12:     **end for**
13:     **if** $i^* \neq \infty$ **then**
14:         $X \Leftarrow X \cup i^*$
15:     **end if**
16: **until** $(X \neq X_{temp})$

$G^o = (V, E^o)$, with V consisting of all nodes in the network and $E^o = V \times V$. The cost of the virtual link between i and j is the OR distance L_{ij} in the underlying wireless network. In the overlay graph, we construct the minimum Steiner tree connecting the source node S with the set of receivers $\mathcal{R}$. The most popular algorithm to construct the minimum Steiner tree was proposed by Dreyfus and Wagner [12] based on dynamic programming. The complexity of the algorithm is $O^*(3^k)$, where k is the number of terminal nodes to be connected. However, the dynamic nature of wireless ad-hoc networks requires a simpler minimum Steiner tree algorithm. In the following, we propose a simple heuristic algorithm to calculate a Steiner tree for our purpose.

Let $Y = S \cup \mathcal{R}$ be the set of terminal nodes. The other nodes on the tree are called Steiner nodes. The idea for the heuristic algorithm is to first construct the Minimum Spanning Tree among all terminal nodes, then grow the spanning tree into a Steiner tree by incorporating Steiner nodes step by step. At each step, the algorithm will find a node i^*, that when added to the tree could maximize the reduction of the total cost of the current tree. In Algorithm 1, Set X stands for the set of nodes that have been admitted into the Steiner tree. $MST(X)$ is the cost of the Minimum Spanning Tree of the set X. The fastest minimum spanning tree algorithm to date can calculate $MST(X)$ in close to linear time [13]. Variable C stores the cost of the minimum Steiner tree. The algorithm completes when the cost could not be further reduced. This heuristic algorithm takes $O(V^2 E)$ time to complete.

The obtained Steiner tree at the overlay level will be mapped to a set of relay paths connecting the source with all receivers in the underlying wireless network. Each overlay link between two nodes is mapped to a unicast OR relay path between the two nodes in the underlying wireless network. A packet will be transmitted using unicast OR along all the relay paths until it reaches all the receivers. We design the OR protocol similar to the ExOR protocol [1]. In the following Section, we briefly describe the OR protocol design and implementation in OPNET.

## IV. MSTOR DESIGN AND IMPLEMENTATION

To test the performance of MSTOR, we design the MSTOR protocol and implement it in OPNET Modeler Version 14.0 by customizing the IEEE 802.15.4 wireless sensor network simulation modules. We inherited the implementation of the physical layer and a portion of the data link layer from the open source Open-ZB library [14]. We customized the network layer and data link layer. Specifically, the packet relay function on overlay Steiner nodes is implemented at the network layer. After the initial overlay Steiner tree construction, each node knows its children in the tree. Whenever a node receives a packet from its parent, it will forward the packet to each of its overlay children using unicast OR. Unicast OR along overlay link is implemented by customizing the network layer and MAC layer of IEEE 802.15.4. The main challenge is to coordinate the packet transmissions on forwarders in the same candidate forwarding set. When a node sends a packet to its downstream node in the overlay Steiner tree, we temporarily refer the node as the sender and the downstream node as the receiver for this unicast OR. The sender knows the receiver's MAC address and the OR distances from its neighbors to the receiver. It constructs a candidate forwarding set (CFS) consisting of neighbors that have shorter OR distances toward the receiver than itself. To limit the coordination overhead between forwarders, we set the maximum CFS size to be three in our experiments. Forwarders in a CFS are prioritized based on their OR distances to the receiver, with higher priority given to forwarders closer to the receiver. The priority order and MAC addresses of forwarders in CFS are embedded in the header of the packet sent out from the source. The sender sends a packet to data link layer which follows the CSMA/CA protocol of IEEE 802.15.4 to transmit it out. The contention window is set equal to the CFS size. It helps to avoid the collision with ACK packets sent from other nodes in the network. After broadcasting a packet, the sender will wait for the ACKs sent back from forwarders in its CFS. The number of time slots that the sender waits for the ACKs equals to the number of nodes in its CFS. If at least one ACK is received from a forwarder, data link layer will inform upper layer to move on to the next packet. If no ACK is received before the timeout, data link layer will employ CSMA/CA again to retransmit the packet. The upper bound of the backoff time in CSMA/CA will be doubled for each packet retransmission. The source node will drop the packet after five unsuccessful attempts. Meanwhile, whenever receiving a packet, a forwarder analyzes the packet header. It will drop the packet if its own MAC address is not included in the CFS field. Otherwise, follow the priority order, the forwarder will send out the ACK packet at the time slot dedicated for it. It also keeps listening to the channel in time slots dedicated to other forwarders in the CFS. By analyzing the ACKs sent out from other forwarders, it will update the information about who is the highest priority node that successfully received the packet. That information will be embedded in the ACK packet that it sends back to the sender. If a forwarder has the highest priority among all forwarders that received the packet, and the packet
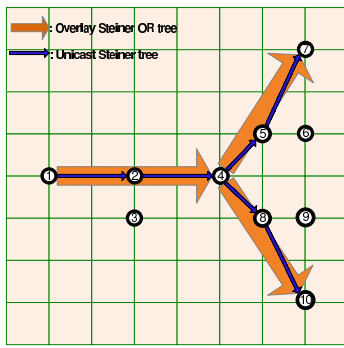
(a) Network topology
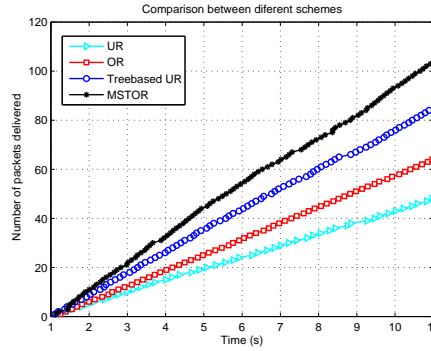
(b) Number of packets delivered at loss rate of 10%
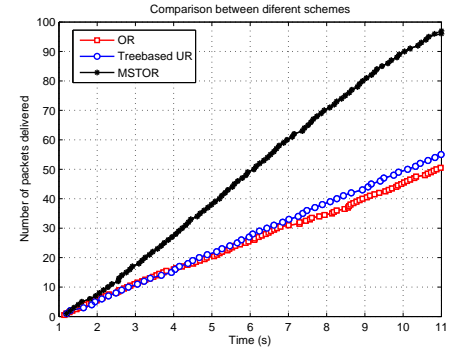
(c) Number of packets delivered at loss rate of 15%

Fig. 2. Simulation Scenario 1

is not a duplicate one, it will send the data packet down to data link layer for transmission to the next relay hop. Otherwise, the forwarder will just drop the packet. The process continues hop-by-hop until the packet is delivered to the receiver.

V. PERFORMANCE EVALUATION

A. Simulation Setup

To compare the performance of MSTOR with other wireless multicast algorithms, we also implemented the shortest path unicast routing algorithm based on ETX metric (UR), unicast opportunistic routing (OR), and tree-based unicast routing (tree-based UR). In UR, the source sends packets to each receiver one by one by following the shortest ETX path. In OR, the source sends packets to each receiver one by one by following the optimal unicast OR path. In tree-based UR, packets are sent to receivers along the minimum Steiner tree with the unicast distance as link weight. The sending rate at physical layer is set to be 250Kb/s. We configure the transmission power with the effective radio coverage of 200m. At the data link layer, we made a small change by disabling the super frame structure and beacon messages. Time is slotted with 4ms time slot duration. There is no RTS/CTS messages in the CSMA/CA mechanism.

B. Simulation Results and Analysis

We first setup a simple network topology as in Figure 2(a). The source is node 1, with two receivers node 7 and 10. The grid size is 70m. The average packet loss rate on each wireless link is around 10%, which is determined by the distance between nodes, physical layer setting and data link layer scheduling scheme. The source sends multicast traffic to receivers with four different routing schemes as referred in Section V-A. For the tree-based UR, the minimum Steiner tree based on ETX metric is plotted as the thin arrows in Figure 2(a). For MSTOR, the overlay Steiner tree based on OR distance is plotted as the thick arrows.

Figure 2(b) presents the comparison of the average number of packets delivered at all receivers as the simulations progress. For this simple topology, MSTOR outperforms other routing schemes. The improvement of MSTOR over tree-based UR is around 20%. In this experiment, the Steiner trees of MSTOR

and tree-based UR have significant overlap. The performance difference comes from that MSTOR employs unicast OR while tree-based UR employs unicast routing to transfer packets along the tree. Tree-based schemes also outperform the other two unicast routing schemes. In tree-based schemes, two receivers share the common path from node 1 to node 4, while in unicast OR and UR, each packet needs two redundant transmissions on this path to reach two receivers. The gain of OR over UR is smaller than the gain obtained from the Steiner trees. We then change the setting to increase the average packet loss rate on wireless links to approximately 15% - Figure 2(c). The gain from OR gets bigger. As a result, MSTOR throughput outperforms the throughput of unicast OR and tree based UR by a factor of 2. Unicast OR also got approximately the same performance as tree based UR. This demonstrates that MSTOR can improve the multicast efficiency in wireless networks with high link loss rate. With the same meaning as [7], source redundancy is defined as the total number of data transmissions made by source divided by the file size (in packets) source need to send out toward receivers. For MSTOR, the source redundancy is 1.25 which is much lower than MORE and Pacifier. MSTOR source redundancy coming from the lossy links out of the source. For each data transmission, sender only need to retransmit the packet when no ACK gets back to it successfully.

We then study a more general network with 30 static nodes randomly located in an area of $1000m \times 1000m$. We randomly choose one source and five receivers. In Figure 3(a), source is the star node and receivers are the red nodes. The average packet loss rate on wireless links is around 15%. The "solid-line tree" is the minimum overlay Steiner tree for MSTOR. The "dot-line tree" is the minimum Steiner tree based on ETX metric. Routes from source to receivers ranged from 5 to 8 hops. Figure 3(b) shows the average number of packets delivered at each receiver as the simulations progress. From the result, the average number of packets delivered at each receiver node with MSTOR is almost three times of unicast OR, and two times of tree-based UR. These results are quite similar to the gain of MORE [5] over ExOR in the wireless multicast case. Compare to scenario 1, the big improvement of MSTOR over other routing schemes is originated from the
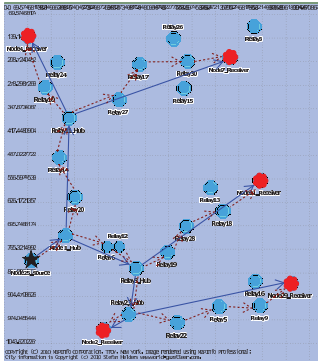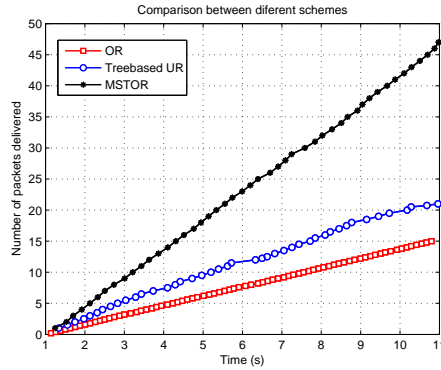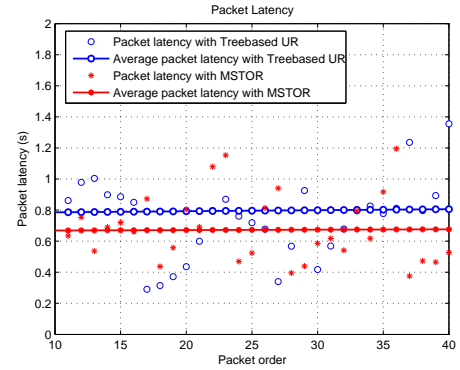
| (a) Network topology | (b) Number of packets delivered | (c) Packet latency comparison |

Fig. 3. Simulation Scenario 2

gains of the minimum Steiner tree based on OR distance, higher packet loss rates on wireless links, and higher wireless node density. The more hierarchical the Steiner tree is, the bigger advantage of MSTOR over other routing schemes. Also since the average packet loss rate on wireless links is high, OR gains more advantage over UR. The higher the density of nodes, the more candidate forwarders for each overlay link in the Steiner tree. It also increases the OR gain over UR. We also measure the packet latency under different routing schemes. Packet latency is defined as the time lag from the source sends out the first copy of a packet until all receivers receive the packet successfully. Figure 3(c) plots the per-packet and average latency comparisons between MSTOR and tree-based UR. MSTOR gets lower average packet latency over the simulation time. The average packet latency in MSTOR is 0.6759 second, while in tree-based UR the average latency is 0.8037 second. This is because of the advantages of dynamic routes of OR over the static routes of UR in a lossy wireless network. The source redundancy for MSTOR is only 1.18.

VI. Conclusion

In this paper, we studied opportunistic overlay multicast for wireless networks. We proposed a routing scheme MSTOR to simultaneously take the advantages of OR and the overlay multicast. Using the shortest OR distances between nodes as overlay link weights, we construct minimum overlay Steiner tree and map it into unicast OR relay paths connecting the source with all receivers. We implemented MSTOR in OPNET by customizing the IEEE 802.15.4 Open-ZB library. Through simulations in OPNET, we showed that MSTOR outperforms unicast OR and existing tree based algorithms. MSTOR significantly improves multicast throughput and achieves low packet latency in wireless networks with lossy links.

Our main focus in this work is to demonstrate the idea of integrating OR into wireless multicast through overlay multicast. MSTOR is a simple protocol, and can be easily deployed in multi-hop wireless networks to achieve high multicast efficiency. The current MSTOR does not employ network coding. MSTOR can be augmented with network coding in a way similar to MORE and Pacifier. We leave it as a future research topic. The current MSTOR implementation

is based on the low speed and simple wireless sensor network standard IEEE 802.15.4 in order to help us clearly understand the advantages of the proposed routing mechanism. As another future work, we will evaluate the performance of MSTOR in wireless networks with higher speed, such as the IEEE 802.11 standard. In addition, current MSTOR design does not exploit opportunistic packet receiving cross neighboring overlay links. As the next step, we will explore other multicast OR schemes that directly extends the OR concept to multicast to maximally exploit the gain of both routing paradigms.

References

[1] S. Biswas and R. Morris, "Opportunistic routing in multi-hop wireless networks," in *In Proceedings of the Second Workshop on Hot Topics in Networks (HotNets-II)*, Cambridge, MA, Nov. 2003.

[2] H. Dubois-Ferrire, M.Grossglauser, and M. Vetterli, "Least-cost opportunistic routing," in *2007 Allerton Conference on Communication, Control, and Computing*, Monticello IL, September 2007.

[3] K. Zeng, W. Lou, and H. Zhai, "On End-to-end Throughput of Opportunistic Routing in Multirate and Multihop Wireless Networks," in *IEEE Infocom 2008*, Phoenix, AZ, April 15-17, 2008.

[4] T.Le and Y.Liu, "On the Capacity of Hybrid Wireless Networks with Opportunistic Routing," in *WASA'09*, Boston, USA, August 2009.

[5] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *In Proc. ACM Sigcomm*, 2007.

[6] D. S. J. D. Couto, D. S. J. De, C. Daniel, R. Morris, D. Aguayo, and J. Bicket, "A high-throughput path metric for multi-hop wireless routing," in *Proc. of ACM MOBICOM*, 2003.

[7] D. Koutsonikolas, Y. C. Hu, and C.-C. Wang, "Pacifier: High-throughput, reliable multicast without "crying babies" in wireless mesh networks." in *IEEE Infocom 2009*, Rio de Janeiro, Brazil, April 19-25, 2009.

[8] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O'Toole, Jr., "Overcast: Reliable multicasting with an overlay network," in *Proceedings of Operating Systems Design and Implementation*, 2000, pp. 197–212.

[9] Y. Chu, S. Rao, and H. Zhang, "A case for end system multicast," in *Proceedings of ACM SIGMETRICS*, 2000.

[10] R. Laufer, H. Dubois-Ferrire, and L. Kleinrock, "Multirate Anypath Routing in Wireless Mesh Networks," in *IEEE Infocom 2009*, Rio de Janeiro, Brazil, April 2009.

[11] C. Reis, R. Mahajan, D. Wetherall, and J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," in *in SIGCOMM Computer and Communications Review*, 2006.

[12] S. E. Dreyfus and R. A. Wagner, "The Steiner problem in graphs," in *Networks*, 1972, pp. 195–207.

[13] B. Chazelle, "A minimum spanning tree algorithm with inverse-ackermann type complexity," *J. ACM*, vol. 47, no. 6, pp. 1028–1047, 2000.

[14] "Open source toolset for ieee 802.15.4 and zigbee." [Online]. Available: http://www.open-zb.net