2011 Ninth Annual Communication Networks and Services Research Conference

Performance Comparison of Opportunistic Routing Schemes in Wireless Sensor Networks

Petros Spachos, Liang Song, and Dimitrios Hatzinakos
Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada
E-mail: {petros,songl,dimitris}@comm.utoronto.ca

***Abstract*—Opportunistic routing is considered a promising direction for improving the performance of wireless sensor networks (WSN). In opportunistic routing, the intermediate nodes collaborate on packet forwarding in order to achieve high throughput in the face of lossy links. That makes the next node selection process crucial. In this paper, we are presenting an opportunistic routing protocol for wireless sensor networks. We also examine three extensions of that protocol, based on different next node selection criterion. We illustrate how each extension works and we evaluate and compare their performance in terms of energy consumption, delivery ratio and packet latency.**

I. INTRODUCTION

Wireless sensor networks have quickly gained popularity. Recent advancement in wireless communications and electronics has enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size. However, the unique characteristics of sensor nodes and their wireless communication can pose significant challenges.

Energy consumption is the major challenge in every sensor network. Usually, sensor networks are designed to operate unattended for long periods of time because battery replacement or rechargeability is sometimes infeasible or impossible. Therefore, the battery charge must be conserved to extend the life of each sensor and the entire sensor network. When applying a routing scheme, the impact that this scheme has on the lifespan of the network should always be considered. The energy required to transmit related routing data should always be considered when implementing a protocol within a sensor node. The communication range of sensor nodes is also limited in order to conserve energy. A great reduction in the transmission power can save sensor node energy. However, it reduces each sensor node detection probability and communication range.

On the other hand, wireless medium may lead to the packets damage or loss due to channel errors. Wireless communications are facing many unpredicted challenges such as air interference, channel fading, environmental changes etc. A reliable routing protocol should handle appropriately any lost or missing packets. Even if the channel is reliable enough, the broadcast nature of wireless communication causes one more problem. If multiple packets meet in the middle of transfer, conflicts will occur and the transfer will fail. In a large scale network with high traffic volume this can be a major problem.

Opportunistic routing tries to overcome the drawback of an unreliable wireless link by taking advantage of the broadcast nature of the wireless medium such that one transmission can be overheard by multiple nodes. A cluster of nodes serves as a relay candidate but only one node finally will forward the packet. An opportunistic routing protocol can use different nodes and follow multiple paths toward the destination for each packet transmission. The next node selection process is crucial and based on opportunistic rules.

The next node selection criterion of an opportunistic routing protocol, can be defined in order to enhance network security in terms of source-location privacy [1]. In this paper, we are examining the impact of the next node selection criterion on the network performance in terms of energy consumption and network transmission reliability.

The rest of this paper is organized as follows. In Section II, the related works are reviewed. The module design and implementation are presented in Section III while opportunistic routing extensions are described in Section IV. In Section V, performance analysis and simulation results are presented, followed by conclusions in Section VI.

II. RELATED WORK

During the last decade, a number of opportunistic protocols have been developed. The first opportunistic routing has been introduced in [2]. *Extremely Opportunistic Routing* (ExOR) selects the next relay node by a slotted ACK (acknowledge) mechanism. Having successfully received a data packet, the node calculates a priority level, which is inversely proportionate to the *expected transmission count metric* (ETX), [3] , which is based on the distance between the node and the destination. The shortest the distance, the highest the priority. The node with the highest priority will then be selected as the next relay node. The main drawback of ExOR is that it prevents spatial reuse because it needs global coordination among the candidate nodes. Candidate nodes transmit in order, only one node is allowed to transmit at any given time while all the other candidate nodes trying to overhear the transmission in order to learn which node will be the next relay node. Moreover, the simple priority criteria that it uses, ETX distance, may lead packets toward the destination through low-quality routes. To overcome this problem, *Opportunistic Any-Path Forwarding* (OAPF) [4] introduces an *expected any-path count* (EAX) metric. This can calculate the near-optimal candidate set at each potential relay node to reach the destination. However, it needs more state information about the network and it has high computational complexity.

ExOR ties the MAC with routing, imposing a strict schedule on routers access to the medium. The scheduler goes in rounds.

978-0-7695-4393-2/11 $26.00 © 2011 IEEE
DOI 10.1109/CNSR.2011.46

MAC-Independent Opportunistic Routing and Encoding Protocol (MORE) [5] tries to enhance ExOR. MORE uses the concept of innovative packets in order to avoid duplicate packets which might occur in ExOR.

In [6], [7] a *Geographic Random Forwarding* (GeRaF) technique was proposed. In GeRaF each packet carries the location of the sender and the destination and the prioritization of the candidates nodes is based on location information. This technique is simple to be implemented but it requires location information for all the nodes in the network. *Hybrid ARQ-Based Intercluster Geographic Relaying* (HARBINGER) [8] is a combination of GeRaF with hybrid automatic repeat request(ARQ). In GeRaF, when there is no forwarder within the range of the sender node, everything must start over again while in HARBINGER hybrid ARQ is used for a receiver to combine the information accumulated over multiple transmission from the same sender.

A number of other opportunistic routing protocols have been proposed [9]–[13]. *Coding-Aware Opportunistic Routing Mechanism* (CORE) [9] is an integration of localized interflow network coding and opportunistic routing. By integrating localized network coding and opportunistic forwarding, CORE enables a packet holder to forward a packet to the next hop that leads to the most coding changes among its forwarder set. *Opportunistic Routing in Dynamic Ad Hoc Networks* (OPRAH) [10] builds a braid multipath set between source and destination via on-demand routing to support opportunistic forwarding. For this purpose, OPRAH allows intermediate nodes to record more subpaths back to the source and also those subpaths downstream to the destination via received Route Request and Route Replies.

III. MODULE DESIGN AND IMPLEMENTATION

The highly dynamic and lossy nature of wireless medium causes frequent transmission failure which lead to retransmission and waste of the limited network resources. Opportunistic routing tries to take advantage of the broadcast nature of the medium and change the next node, increasing the number of the possible paths toward the destination in order to improve the performance comparting to the traditional best path routing. Hence, one of the most important tasks in every opportunistic routing protocol is the next node selection process.

In this section we are describing the main principles of the proposed opportunistic routing protocol.

A. Network address

Network address is related to the context and is subjected to a "cost of delivery" criteria. Given a node address n and the destination address d of a data packet, this "cost of delivery" $c_{n,d}$ should be locally obtained. This could indicate the average or the approximate cost of delivering a packet from the node n toward the destination d, independent of any dynamic change in the network. Usually, in large-scale wireless sensor networks $c_{n,d}$ is correlated with the distance between the two nodes.

Initially, the destination node broadcasts a number of identity advertisement packets and every nodes thereon flood the packet to the network. On the reception of a packet, a node can count the smallest number of hops from the destination and use it as "cost of delivery" criteria, $c_{n,d}$. Whenever a new node joins the network, it can estimate its logic address by acquiring the logic address of its neighbor nodes. If the destination node changes, the procedure should start from the beginning. When the source node changes there is no need to repeat the procedure. If a node leaves the network, it will not take part in the selection process.

B. Radio implementation

Cognitive radio was first introduced in [14] as an ideal-omnipotent radio for user centric communications because it takes into consideration all the available parameters. For large-scale wireless network, two propositions were further suggested in [15]:

1) In order to avoid collisions with other simultaneous ongoing transmissions, the radio can sense the spectrum resource, opportunistically, before any transmission.
2) The radio can extract useful information for local cooperation by opportunistically polling one or more proximity radios onto the selected spectrum.

With the above proportions, we can extend the concept of cognitive radio to the area of cognitive network, which implements both dynamic spectrum and radio access.

C. Collision avoidance

We consider a scheme that prevents collisions by making use of the cognitive radio. Specifically, the radio can have access to a group of data channels. Every channel in that group is associated with two different frequency tones, one for sensing and one for polling which are also distinctive from the data channel frequency. Therefore, the radio hardware should be composed of two transceivers, one for sensing/polling and one for data.

Initially, when a node s has to transmit a packet, it senses for an available channel and then broadcasts a polling tone. All the nodes which are in the range of node s can detect this polling tone. A neighbor node can decide to join the transmission based on its own autonomous availability. If a node decides to join the transmission it sends out a polling tone to its surrounding nodes. In this way, sensing and polling tones protect wireless link module from spectrum interference.

D. Transmission process

In every time slot $i \in \{i_0, ..., i_0 + T(i_0)\}$, the transmission strategy is decided by the transmission power at the sensor node. We assume that a number of packets can be transmitted in one time slot. Every damaged or lost packet will be retransmitted in the next assigned slot. If we use BPSK without channel coding, the Packet Error Rate, $PER(i)$, can be written as [16],

$$PER(i) = 1 - \Big(1 - Q\Big(\sqrt{\frac{2P_t(i)\cdot \hat{G}(i)}{\sigma_n^2}}\Big)\Big)^{F_d}, \quad (1)$$

where P_t is the transmission power, F_d is the length of the data, σ_n^2 is the noise power, $Q(x) = \frac{1}{\sqrt{\pi}} \cdot \int_{\frac{x}{\sqrt{2}}}^{\infty} e^{-t^2} dt$ and

$$\hat{G}(i) = A \cdot \hat{D}_s(i)^{-n}, \quad (2)$$

where A is a constant and $\hat{D}_s(i)$ is the distance between the sender node s and the next node i, and n is wireless channel path loss component.

Every packet transmission process is subjected to that PER.

Figure 1 shows the different values of PER for different transmission power.

E. Next node selection process

We are using four types of packets during the next node selection process: Request To Send (RTS), Confirm To Send (CTS), DATA and ACK. RTS/CTS are used during the handshake process between neighbor nodes while ACKs are used for verification of DATA delivery.

When a node s has to transmit a packet, first it broadcasts a RTS packet, in which it includes its own address and the destination address, d, and then node s keeps listening. All the surrounding nodes which are in the coverage area of s are able to hear this request, conforming a set of *candidate nodes* E_s. There is a subset $V_s \leq E_s$ conformed by any node $i \in E_s$ that satisfying the condition $c_{i,d} < c_{s,d}$ so,

$$V_s = \{i \in E_s | c_{i,d} < c_{s,d}\} \quad (3)$$

If a node i is in V_s subset and is available for receiving a packet, it should send a CTS packet back to the sender node s. In order to prioritize the nodes based on their distance from the destination, each node $i \in V_s$ initialize a timer, with timeout period T_i, which is inverse proportional to the difference $c_{s,d} - c_{i,d}$ and can be determined as follows:

$$T_i = \frac{C_0}{c_{s,d} - c_{i,d}} + SIFS, i \neq d \quad (4)$$

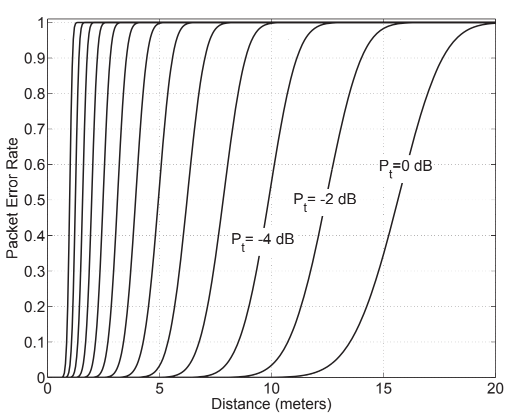


Fig. 1: Packet error probability curve for BPSK without channel coding.

where C_0 is a constant and SIFS is the smallest time interval composed of the module processing time and the transceiver RX/TX switch time. In our simulation, which will be presented in following section, T_i is also slotted according to the minimum carrier-sensing time.

In the next step, node i backs off for the period T_i. If the data channel is free after that period, node i sends a CTS to the sender node, otherwise it quits. After that procedure, the sender node s will receive the first CTS from the node which is closer to the destination, if there is no packet error or loss. This will be the next hop *relay node* and it will receive the DATA packet. When the next node receives the DATA packet it replies with an ACK to the sender and follows the same procedure until the DATA packet reaches the destination. Every transmission of the four types of packets is subjected to PER, leading to packets damage or loss.

In the case that the sender node receives more than one CTS packets simultaneously there are certain mechanisms in the sender node, such as cyclic redundancy check (CRC) that can detect this collision and differentiate the nodes.

When a node i sends a CTS it waits for time T_w to receive the DATA packet from the sender node s, otherwise it goes back to a sleep mode. T_w is the time needed for the sender to transmit the DATA packet to that node and can be defined as:

$$T_w = d(s,i) \cdot D_0 + SIFS \quad (5)$$

where $d(s,i)$ is the distance between the sender and the relay candidate and D_0 is a constant based on the size of the DATA packet.

In the same way, the sender node has to wait for T_c time to get a CTS packet before it broadcasts a RTS again. T_c is the time needed for a node which is located at the limit of the range R of the sender node, and can be defined as:

$$T_c = R \cdot C_0 + SIFS \quad (6)$$

where R is the range of the sensor and C_0 is a constant.

The time that a sender node will wait for an ACK before it retransmits the DATA can be also defined as:

$$T_A = d(s,i) \cdot A_0 + SIFS \quad (7)$$

where $d(s,i)$ is the distance between the sender node s and the candidate node i and A_0 is a constant based on the size of the ACK packet.

When a node receives a DATA packet, it stores it in its buffer. If the buffer of a node is full, this node does not reply to any RTS packet until it has space on its buffer. In this way, the size of the buffer is crucial. Small buffer size will lead to the use of different nodes and different paths toward the destination while large buffer size will lead to the use of almost the same nodes for every packet transmission, increasing the energy consumption of those nodes.

If the set V_s is empty, meaning that there is no available node or the candidate nodes have full buffers, the relay node is not updated. In the next time slot the sender tries again because there might be other nodes available (because node availability

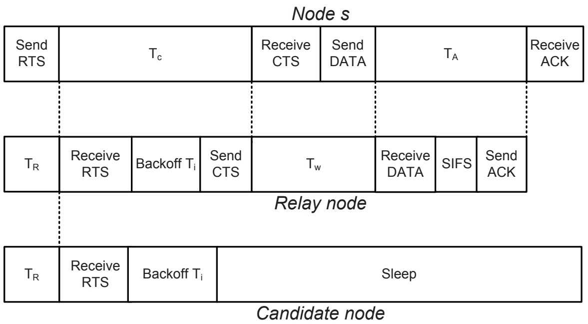


Fig. 2: Packet relaying mechanism.

is independently generated) that could provide advancement toward the destination.

The described packet mechanism is illustrated in Figure 2. We illustrate the algorithm for the source node s and any candidate node in Algorithm 1 and Algorithm 2, respectively.

IV. OPPORTUNISTIC ROUTING EXTENSIONS

In this section, we are introducing three more extensions of the proposed opportunistic routing protocol. Each extension has different selection criterion for the next relay node.

A. Non-repeating Opportunistic

Non-repeating opportunistic routing tries to avoid using the same nodes in sequential packet transmissions, with the use of memory in every node. That memory contains a flag that initiates if the node will participate in a packet transmission. When the flag is *true* the node can not participate in a packet transmission because it has participated in another packet transmission recently.

Initially, the flag in every node is *false*. When a node receives a packet, that flag becomes *true* in that node. The flag will remain *true* until the node receives a RTS packet. In that time slot, the node will not respond with a CTS packet but the flag will become *false* again and the node will be able to participate in later transmissions. The algorithm for every candidate node is illustrated in Algorithm 3.

B. Opportunistic routing with random delay

Opportunistic routing with random delay includes a random factor in the response time of each candidate node. As we have explained in section III, when a node receives a RTS packet, it will reply with a CTS packet after time T_i, equation 4. In this approach, we have add a random number in the range of $[0, 1)ms$ in that time interval. In this way, the backoff time for each node is not based only on the distance between the sender and the candidate node.

The only difference in the candidate node Algorithm 2 is in line 3 which indicates the backoff time. In the calculated T_i time, this approach adds a random delay in the range of $[0, 1)ms$. The new backoff time T_{delay} for that approach is:

$$T_{delay} = T_i + dblrand(0, 1) \tag{8}$$

Algorithm 1 Source Node.

```
if (isNewDataPacket(datapkt)) then
    BroadcastRTS()
    interval=T_R
    reason=Listen(interval)
    while (reason!=CTS) do
        BroadcastRTS()
        interval=T_R
        reason=Listen(interval)
    RelayNode= CTS.SenderNode
    SendPacket(datapkt,RelayNode)
    interval=T_A
    reason=Listen(interval)
    while (reason!=ACK) do
        SendPacket(datapkt,RelayNode)
        interval=T_A
        reason=Listen(interval)
    GoToSleepMode();
```

Algorithm 2 Candidate Node.

```
1: if (isRTS(rts)) then
2:     T_i= CalculateBackoff()
3:     wait(T_i)
4:     Channel=ChannelSensing()
5:     if (Channel==IDLE) then
6:         SendCTS(rts.SenderNode)
7:         interval=T_w
8:         reason=Listen(interval)
9:         if (reason==DATA) then
10:            SendACK(DATA.SenderNode)
11:        else
12:            GoToSleepMode()
13:    else
14:        GoToSleepMode()
```

where dblrand(0,1) is a *random number generator function* to generate a *double* number in the range of $[0, 1)$.

C. Opportunistic routing with random relay

Opportunistic routing with random relay tries to take advantages of the opportunistic routing, with random selection criteria in the next relay node process. In that approach, when a node has a packet to transmit, it will wait for almost all the nodes in its coverage area to reply with a CTS packet. Some of those nodes may not get a RTS packet or their CTS does not reach the sender node because of channel error, based on equation 1.

The time that the sender node has to wait have changed in that approach. The new $T_{R_{all}}$ should be sufficient for nodes located in the borders of the coverage area of the sender node, to reply with a CTS packet. This time is equal to:

$$T_{R_{all}} = R \cdot C_0 + SIFS, \tag{9}$$

where C_0 is a constant and R is the transmission range of the sensor, in meters.

V. PERFORMANCE ANALYSIS AND SIMULATION RESULTS

To evaluate the performance of the schemes proposed, we pursued simulations using OMNET++ [17], in terms of energy

Algorithm 3 Candidate Node- Non-repeating Opp.

```
if (isRTS(rts)) then
    if (flag!=TRUE) then
        T_i= CalculateBackoff()
        wait(T_i)
        Channel=ChannelSensing()
        if (Channel==IDLE) then
            SendCTS(rts.SenderNode)
            interval=T_w
            reason=Listen(interval)
            if (reason==DATA) then
                SendACK(DATA.SenderNode)
                flag==TRUE
            else
                GoToSleepMode()
        else
            GoToSleepMode()
    else
        flag==FALSE
        GoToSleepMode()
```

consumption, delivery ratio and packet latency.

In the simulations, 500 nodes with radio transmission range 14 meters, are uniformly randomly distributed over a square target area of $20,000(m^2)$. The communication parameters were chosen based on IEEE 802.15.4, as listed in Table I.

We examine 10 different pairs of source and destination node with distance between them from 261 to 283 meters. We also examine nodes with different buffer size.

Energy Consumption: Figure 3 shows the energy consumption of each approach for 200 packets and for buffer sizes=5, Figure 3(a), and buffer size=2, Figure 3(a).

Opportunistic routing consumes the less energy over all the other approaches. This is because the selection criterion of this approach is based on the distance between the transmitting and the receiving node. Transmitting node tries to to send the DATA packet to the neighbor node which is closer to the destination, leading to the shortest available path toward the destination at that time. Non-repeating opportunistic routing has similar energy consumption because it has similar selection criterion. However, non-repeating opportunistic routing does not follow the shortest available path in sequential packet transmission, leading to an increase in the energy consumption.

In opportunistic routing with random delay, the selection criterion is both the distance and a random delay leading to a more random selection of the next relay node. This creates more paths toward the destination, with more hops and as a consequence increases the energy consumption. The energy consumption in all these three approaches is increased as we increase the packet rate.

Opportunistic routing with random relay consumes the most energy over all the other approaches. In that approach, the next node selection criterion is completely random, leading to different paths toward the destination with different number of hops. Moreover, in that approach, every transmitting node has to remain active longer than in the other approaches, waiting for all the CTS packets. On the other hand, because the selection criterion is random, the increase of the packet rate does not affect this approach.

Parameter	Unit	Value
F_d	*bit*	128×8
n		2.5
A	dB	-31
σ_n^2	dBm	-92
P_t	dBm	-2

TABLE I: Communication Parameters Setup

When the buffer size decreased, all the four approaches consumes more energy. When the buffer of a node is full, this node does not reply to any RTS packet. Hence, the packet should follow different paths, with more hops, increasing the energy consumption. As the packet rate increased, the first three approaches tend to consume similar energy. This is mainly because they tend to use all the available paths toward the destination. Opportunistic routing with random relay is not affected from the buffer size.

Delivery Ratio: For the delivery ratio 1000 packets were sent from the source toward the destination, for all the 10 source-destination pairs. When a node sends 2 RTS for a packet but it does not get a CTS and its buffer is full, it drops that packet. Figure 4 shows the results.

Opportunistic routing and non-repeating opportunistic routing tends to follow almost the same nodes in packet transmission. As the packet arrival rate increased, the buffer at these node becomes full, increasing the probability to drop a packet. Opportunistic routing with random relay tends to use different nodes in each packet transmission, leading to a better delivery ratio. Opportunistic routing with random relay performs better over all in terms of delivery ratio. The probability for this approach to use the same node in sequential packet transmission is minor, because of the randomness in the next node selection process.

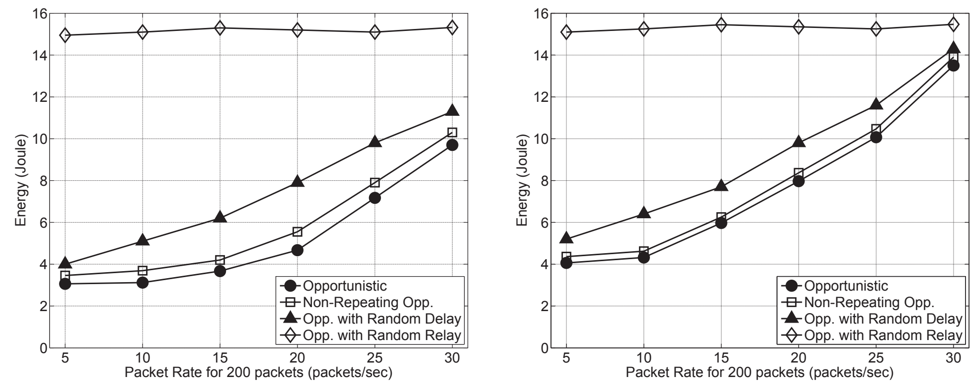
As the buffer size decreased, Figure 4(b), the first three approaches tends to drop more packets while opportunistic routing with random relay is not affected from the buffer size.

Packet Latency: Figure 5 shows the packet latency for 1000 packets for each approach. Opportunistic routing performs better over all the other approaches in terms of packet latency. This is because it tends to follow the shortest available path toward the destination in each time slot. The other approaches discover different paths toward the destination, with more number of hops.

As the buffer size decreased, Figure 5(b), and the packet arrival rate increased, all the four approaches tends to have the same performance in terms of packet latency.

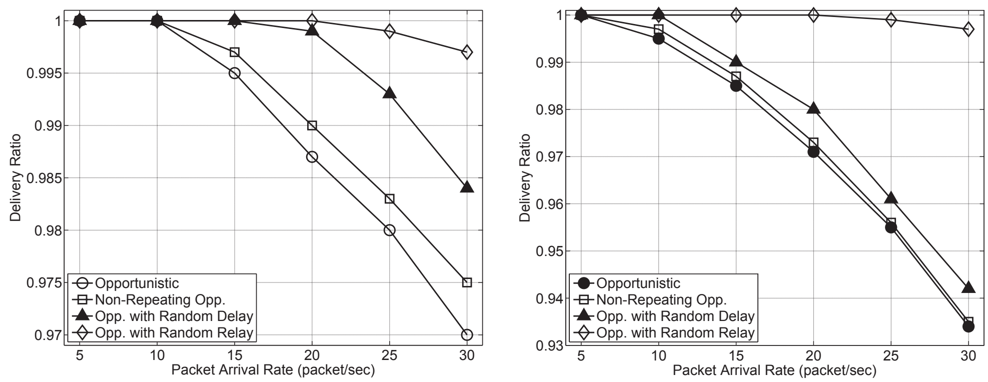
VI. CONCLUSION

Opportunistic routing tries to take advantage of the broadcast nature of wireless communication and increase network performance. Every opportunistic routing protocol is subjected to its next node selection criterion. In this paper, we presented an opportunistic routing protocol and we introduced three extensions of that protocol with different next node selection



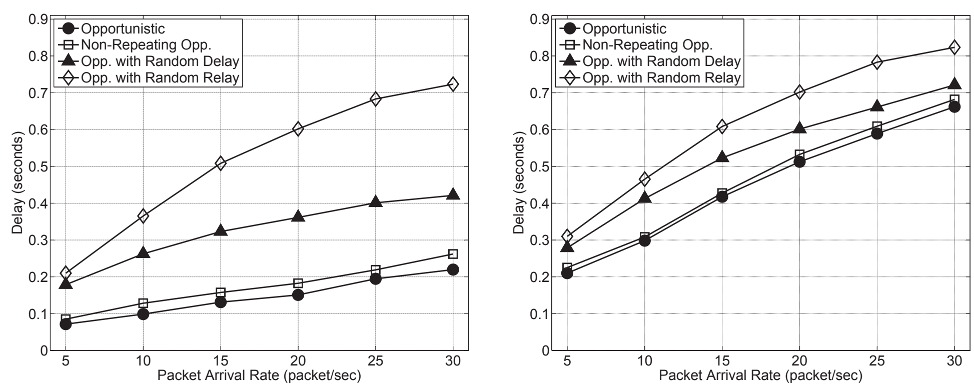
(a) Buffer size = 5 (b) Buffer size = 2

Fig. 3: Energy Consumption.



(a) Buffer size = 5 (b) Buffer size = 2

Fig. 4: Delivery Ratio.



(a) Buffer size = 5 (b) Buffer size = 2

Fig. 5: Packet Latency.

criterion. We examine how the selection criterion affects the performance of each extension. Simulation results shown that opportunistic routing performs better over the other alternatives, in terms of energy consumption and packet latency. However, opportunistic routing with random relay performs better in terms of delivery ratio and it is not affected of the buffer size of each node.

REFERENCES

[1] P. Spachos, Liang Song, and D. Hatzinakos, "Opportunistic routing for enhanced source-location privacy in wireless sensor networks," in *Communications (QBSC), 2010 25th Biennial Symposium on*, May 2010, pp. 315 –318.

[2] Sanjit Biswas and Robert Morris, "Opportunistic routing in multi-hop wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 1, pp. 69–74, 2004.

[3] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris, "A high-throughput path metric for multi-hop wireless routing," *Wirel. Netw.*, vol. 11, no. 4, pp. 419–434, 2005.

[4] Zifei Zhong, Junling Wang, Srihari Nelakuditi, and Guor-Huar Lu, "On selection of candidates for opportunistic anypath forwarding," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 10, no. 4, pp. 1–2, 2006.

[5] Szymon Chachulski, Michael Jennings, Sachin Katti, and Dina Katabi,

"Trading structure for randomness in wireless opportunistic routing," in *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, New York, NY, USA, 2007, pp. 169–180, ACM.

[6] Michele Zorzi and Ramesh R. Rao, "Geographic random forwarding (geraf) for ad hoc and sensor networks: Energy and latency performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, 2003.

[7] Michele Zorzi and Ramesh R. Rao, "Geographic random forwarding (geraf) for ad hoc and sensor networks: Multihop performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 337–348, 2003.

[8] Bin Zhao, R.I. Seshadri, and M.C. Valenti, "Geographic random forwarding with hybrid-arq for ad hoc networks with rapid sleep cycles," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, nov.-3 dec. 2004, vol. 5, pp. 3047 – 3052 Vol.5.

[9] Yan Yan, Baoxian Zhang, H.T. Mouftah, and Jian Ma, "Practical coding-aware mechanism for opportunistic routing in wireless mesh networks," in *Communications, 2008. ICC '08. IEEE International Conference on*, may 2008, pp. 2871 –2876.

[10] Cedric Westphal, "Opportunistic routing in dynamic ad hoc networks: the oprah protocol," in *Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on*, oct. 2006, pp. 570 –573.

[11] Yuan Yuan, Hao Yang, Starsky H. Y. Wong, Songwu Lu, and William Arbaugh, "Romer: Resilient opportunistic mesh routing for wireless mesh networks," in *in The 1st IEEE Workshop on Wireless Mesh Networks (WiMesh*, 2005.

[12] M.S. Nassr, Jangeun Jun, S.J. Eidenbenz, A.A. Hansson, and A.M. Mielke, "Scalable and reliable sensor network routing: Performance study from field deployment," in *INFOCOM 2007.*, may 2007, pp. 670 –678.

[13] S. Jain and S.R. Das, "Exploiting path diversity in the link layer in wireless ad hoc networks," in *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a*, june 2005, pp. 22 – 30.

[14] III Mitola, J. and Jr. Maguire, G.Q., "Cognitive radio: making software radios more personal," *Personal Communications, IEEE*, vol. 6, no. 4, pp. 13 –18, aug 1999.

[15] Liang Song and Dimitrios Hatzinakos, "Cognitive networking of large scale wireless systems," *Int. J. Commun. Netw. Distrib. Syst.*, vol. 2, no. 4, pp. 452–475, 2009.

[16] J.G. Proakis, *Digital Communication*, McGraw-Hill Inc., 1995.

[17] OMNeT++ discrete event simulator, "http://www.omnetpp.org," .