

# Improving QoS-based Routing by Limiting Interference in Lossy Wireless Sensor Networks

Shuang Li<sup>1</sup>, Alvin Lim<sup>2</sup> and Cong Liu<sup>3</sup>

<sup>1</sup>Department of Computer Science and Engineering, The Ohio State University,  
Columbus, Ohio, USA

[li.908@osu.edu](mailto:li.908@osu.edu)

<sup>2</sup>Department of Computer Science and Software Engineering, Auburn, Alabama, USA

[lim@eng.auburn.edu](mailto:lim@eng.auburn.edu)

<sup>3</sup>Department of Computer Science and Engineering, The University of North Carolina  
at Chapel Hill, Chapel Hill, NC, USA

[cong@email.unc.edu](mailto:cong@email.unc.edu)

## ABSTRACT

*Many new routing protocols have been proposed for wireless sensor networks to maximize throughput, minimize delay or improve other QoS metrics in order to solve the problems of resource-constrained sensor nodes in large networks. However, many of them are based on flooding or its variants. Many routing messages are propagated unnecessarily and may cause different interference characteristics during route discovery phase and in the actual application data transmission phase. As a result, incorrect routes may be selected. Epidemic algorithms have been used to limit flooding in the field of wireless sensor networks. Directed diffusion has been commonly used in wireless sensor networks because it is designed to improve energy efficiency and scalability. However, the intrinsic flooding scheme for interest subscriptions prevents it from achieving the maximal potential of these two goals. We propose a routing protocol that uses ID-free epidemic flooding to limit interference in conjunction with metrics for increasing throughput and reducing delay. Simulation results in ns2 show that there is an optimal number of neighbors to achieve the best throughput and delay performance. For a fixed topology of a certain size, there exists an optimal percentage of neighbors that forward the flooding message to achieve the best throughput and delay performance.*

## KEYWORDS

*Delay, epidemic flooding, interference, throughput, wireless sensor networks*

## 1. INTRODUCTION

Recent advances in low-power electronics design have enabled the rapid development of tiny, wireless unattended sensors. The field of wireless sensor networks has been under intensive research in the past ten years. Routing is one of the most important issues in protocol design. Most routing protocols proposed for wireless sensor networks, such as SPIN[1], directed diffusion[2], LEACH[3], GPSR[4], GAF[5] and other variants, target at maximizing throughput, minimizing delay or improving other QoS (Quality of Service) metrics since resource-constrained sensor nodes are usually deployed in large numbers.

Routing protocols that require location information, such as LAR [6], GPSR [4], and DREAM [7], do not need to flood routing requests. Others, such as DSR [8], AODV [9], ZRP [10], and TORA [11], suffer from the effects of flooding, even with some optimizations, since nodes do not know their locations. Flooding causes many routing messages to be propagated unnecessarily.

To reduce the number of routing messages sent and to guarantee reliable data dissemination, epidemic algorithms, which was first used in replicated databases [12], has begun to be used in the context of wireless sensor networks, such as GOSSIP [13] and Fireworks[14]. These protocols require the sender to have knowledge of the potential receivers, which is achieved by exchanging beacons. Anonymous Gossip (AG) [15] overcomes this problem by attempting to send a gossip message and waiting until the other node sends back a gossip reply, incurring higher overhead.

Directed diffusion, though regarded as an epidemic algorithm in [16] since it avoids broadcast storm, does not perform well with interest flooding. No matter what metrics are used in selecting a route (basic directed diffusion uses delay), the route which performs best during route discovery phase may not perform well during the actual data transmission phase due to differences in interference levels caused by the different traffic patterns. Interest flooding increases traffic in the network and causes maximum level of interference. Exploratory data are flooded to determine the best path, which follows gradients established in the interest propagation phase. Actual application traffic only flows through the reinforced path, which is not affected by inter-path traffic at all, assuming there is no other data transmission at that time. Every node has an interference range. Interference set [18] and conflict graph [19] are used to schedule network traffic or theoretically analyze the impact of interference on wireless networks. However, no routing protocol could have the prior knowledge about which path the actual data traffic will go through and what the traffic pattern will be like before the route is determined.

Our goal is to design solutions which make more accurate routing decisions by reducing the interference level during the route discovery phase and making it more similar to that during the actual data transmission phase. We limit the interest flooding in a modified directed diffusion, which uses a hybrid metric described in [17]. We measure the throughput and delay performance in different topologies and identify the optimal percentage of neighbors that propagate interests.

## 2. RELATED WORK

Most of the GPS-free routing protocols for ad hoc or sensor networks are based on flooding, such as DSR [8], AODV [9], and TORA [11]. In DSR, a source uses the complete hop-by-hop route to the destination determined by flooding the network with route request (RREQ) packets during route discovery. If the source is notified with a route error (RERR) packet, the source removes any route using this link from its cache and a new round of route discovery will be initiated. AODV and TORA, which share DSR's on-demand characteristics, also discover routes on an as needed basis through a similar route discovery phase.

Message redundancy caused by excessive flooding has been realized in recent years. The taxonomy of the major proposed solutions is described in [14]: Probabilistic-based schemes, area-based methods and neighbor knowledge methods. GOSSIP [13], Anonymous gossip [15] and Fireworks [14] fall under the first category. In GOSSIP, when a node first receives a route request, it broadcasts the request to its neighbors with probability  $p$  and it discards the request with probability  $1-p$ . In Fireworks, a node re-broadcasts the message to all its neighbors with probability  $p$  and it sends it to only  $c$  randomly selected neighbors with probability  $(1-p)$ . The Fireworks protocols results in higher reliability given the same number of links over which the broadcast packet is transmitted. Anonymous gossip (AG) does not require any member to know the other members of the multicast group because the sender sends a gossip message and it will gossip until the other node sends back a gossip reply. The other two categories require location awareness or two hop neighborhood knowledge, both of which incur more overhead than the first category.

The broadcast nature of wireless medium leads to interference, which is one of the most significant features of wireless networks. Network performance, such as throughput and delay, is affected by node interferences. Jain et. al. [19] theoretically analyzed the impact of interference on multi-hop wireless network performance and determined the maximum throughput that can be supported by the resulting network given a specific placement of wireless nodes in physical space and a specific traffic workload. Chaporkar et. al. [18] proposed maximal scheduling to maximize the network throughput under arbitrary topologies and interference models. Both assume prior knowledge of a specific topology.

### 3. PROBLEM STATEMENT

In this section, we first introduce the error and interference model we use to capture the lossy and interference natures of WSNs. We then discuss how to design an efficient routing algorithm for the problem.

#### 3.1. Error and Interference Model

Wireless links are affected by propagation loss, shadow fading, and multipath Rayleigh fading. *SNR* (Signal-to-Noise ratio) is a good indicator of link quality and can be determined from the hardware. Different *SNRs* cause different *BERs* (Bit error rate) [20]. Lee et. al. [21] derived the mathematical formula for calculating *BER*.

$$BER = 0.5 \times \text{erfc} \left( \sqrt{\frac{P_r \times W}{N \times f}} \right)$$

$P_r$  is the received power,  $W$  the channel bandwidth,  $N$  the noise power,  $f$  the transmission bit rate, and *erfc* the complementary error function. Most wireless cards typically measure:

$$SNR = 10 \log \frac{P_r}{N}$$

The above formula is the case when only one frame is received by the receiver. If other frames arrive at the receiver when it is receiving one frame, the *SNR* (or *SINR* to be more accurate, where *I* means interference) is,

$$SNR = 10 \log \frac{P_r}{N + \sum_{i=1}^n P_{r_i}}$$

$\sum_{i=1}^n P_{r_i}$  is the interference component and  $P_{r_i}$  is the signal strength of the other frames at the receiver.  $n$  is the number of other frames that arrive at the receiver simultaneously. Given the MAC frame size, *FER* (Frame Error Rate) could be calculated from *BER*.

#### 3.2. Overview of Directed Diffusion and Variants

Directed diffusion uses a publish/subscribe communication model in which a sink node floods interests as requests for a named data. As the interest is propagated through the network, each intermediate node sets up a gradient with its neighbors and enables data that match the interest to be pulled towards the sink. Sensor nodes with data that match the interest will forward exploratory data propagated by intermediate nodes through established gradients to the sink. The sink initiates a reinforcement message to the node that first forwarded the new data to it.

Other nodes use the same rule to reinforce the upstream neighbor. The source node continues to send data through the reinforced path after it received the reinforcement.

Based on the above rule, basic diffusion generally selects route with the lowest delay. In the past few years, researchers have proposed a variety of single or hybrid metrics with the purpose of improving the performance, including throughput, delay, jitter, and deadline-hit ratio, of wireless networks. Single metrics include RTT [22], PktPair [22], ETX [23], WCETT [24], and EDR [25]. Most of them can be implemented in directed diffusion. In order to consider both throughput and delay and take into account intra-path interference, a hybrid metric was proposed in [17].

### 3.3. Route Selection Problems

Directed diffusion and its variants select paths which perform best during the exploratory data phase. However, during this phase, the network has very high traffic since the exploratory data follow all the gradients set up during the interest propagation phase, which effectively results in flooding. Different topologies and traffic types lead to different interference levels for each node in the network. As a result, the performance of a candidate path is determined by how each link performs under high interference level due to exploratory data flooding. As shown in Figure 1, the interference range usually differs from the transmission range, typically by a factor of 2. Node 3 can send to Nodes 2, 4, 6, and 7 but not Nodes 1, 5, 8, and 9. However, Node 3's transmission will interfere with that of Nodes 1, 5, 8, and 9.

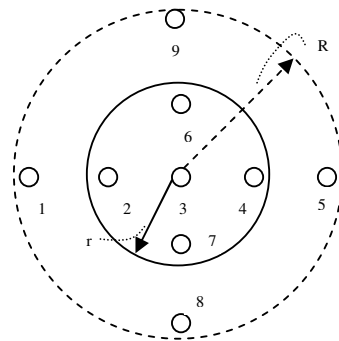


Figure 1. Node 3's transmission and interference ranges. The solid circle is the transmission range and the dashed circle is the interference range.  $R = 2r$ . We assume homogeneous nodes.

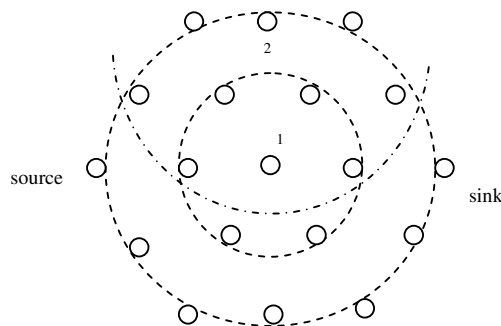


Figure 2. A simple topology of triangular tessellation with 19 nodes. Each node has the same distance to its nearest neighbors.

After reinforcement, one path is selected for application data transmission and it rarely suffers from interference from other paths used in route discovery since no data will be sent over them once a route is selected and reinforced. Different interference level in the two phases causes incorrect routes to be selected since a path that performs best under high interference environment may perform worst when transmitting data without other flooded paths. Border links are good examples (Figure 2). The inner circle is the transmission range and the outer circle is the interference range. Consider Nodes 1 and 2. Almost every node in the topology is in Node 1's interference range; thus all the links starting from or ending at Node 1 should have very bad performance when exploratory data are propagated. In contrast, Node 2 is on the border and only the upper half of the nodes is in its interference range. This is also true of other border nodes. However, when there is only one data path in the network, a central path (through Node 1) will perform better than a detour path composed of border links.

Interference causes route selection problems. Unpredictable traffic pattern and unknown geographic information aggravate this problem.

#### 4. EPIDEMIC ALGORITHM IN DIFFUSION

In this section, we present a set of localized epidemic algorithms (without using neighbor information) for a modified directed diffusion. The main purpose is to reduce interference during the exploratory data phase, make the interference level more similar to that during the actual data transmission phase and improve route selection. The key challenge for designing such a protocol is that probability-based schemes usually makes use of some basic understanding of the network topology to assign to a node a probability  $p$  to broadcast, which means the sender knows the IDs of potential receivers. However, directed diffusion is a localized ID-free routing protocol, where no node knows its neighbors before interest flooding. (We assume this original property of directed diffusion although in practice it may be possible for the sender to determine the MAC addresses of the receivers.) So we have two choices here: receivers probabilistically drop interest or senders probabilistically send exploratory data since neighbor information is known by the sender. In both ways, we are taking the risk of losing candidates. There is a trade-off between the degree of interference level match and the completeness of the candidate pool. (Our results below show the optimal probability that will maximize throughput or minimize delay for each network configuration.)

##### 4.1. Epidemic Interest Flooding

Interest flooding is the first phase of directed diffusion. To adapt it to directed diffusion, when a node receives an interest packet, it updates the gradient table with a probability of  $p$ , i.e. with a probability of  $1-p$ , the node will not update its gradient table (Table 1). (We assume the sender has no knowledge about its neighbors.) Strictly speaking, interests are still flooded through the network. However, exploratory data are propagated selectively, which satisfies our goal, since only parts of the gradients, with the percentage of  $p$ , are established in the earlier phase.

TABLE 1. Pseudocode segment for epidemic interest flooding

Receiver side:
If it is a new interest packet
Set $r = random()$
If $r > 1-p$
Update gradient table
Else
Drop interest

In Figure 3, interests from Nodes 2 and 4 are probabilistically dropped by Node 1; thus only the interest from Node 3 is updated in Node 1's gradient table. In the next phase, exploratory data flows from the other side to Node 1 and only Node 3 will be chosen as the downstream node. However, if Node 1 does not drop interests from Nodes 2 and 4, the exploratory data flowing from Node 1 to Node 3 will suffer from the interference coming from Node 2 and 4 since Node 1 is also sending exploratory data to them.

TABLE 2. Pseudocode segment for epidemic exploratory data propagation

```

Sender side:
1) Gossip version
If it is exploratory data
    Set  $r = random ()$ 
    If  $r \leq p$ 
        Send it to all nodes in gradient table
    Else
        Discard it
Else
    Execute original code

2) Fireworks version
If it is exploratory data
    Set  $r = random ()$ 
    If  $r \leq p$ 

```

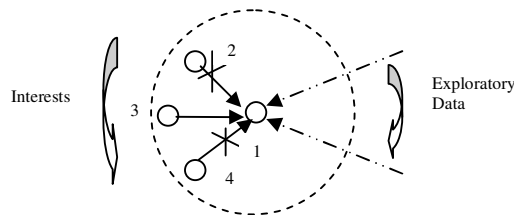


Figure 3. Probabilistic interest dropping reduces interference level during exploratory data phase. The dotted circle is the interference range of Node 1.

#### 4.2. Epidemic Exploratory Data Propagation

Another way to limit exploratory data flooding is to drop exploratory data probabilistically during the exploratory data phase. The sender selects gradients from the table probabilistically as the downstream links. This is the direct implementation to reduce flooding during the exploratory data phase. Both GOSSIP and Fireworks algorithms can be used.

Obviously, GOSSIP algorithm eliminates a variety of candidates, where the change in interference level may be very abrupt. In Figure 4, Node 1 discards the exploratory data completely while Node 1' randomly selects two neighbors ( $c = 2$ ). Although GOSSIP reduces interference in this case, it also eliminates all candidate links from Node 1. Nodes 2 and 2' send to all nodes in the gradient table. Links in the transmission range of Node 2 has the same interference range as those in the transmission range of Node 2'. The transmission of Node 1' rarely affects that of Node 2' if Node 1' and 2' are beyond the interference range. The two transmissions in the transmission range of 1' cause limited interference with each other

compared to full flooding. However, the advantage is that more candidate links are being considered.

The combination of (A) and (B) above is worth considering. Intuitively, there will be less interference and fewer candidates than either scheme. We will not discuss the details here.

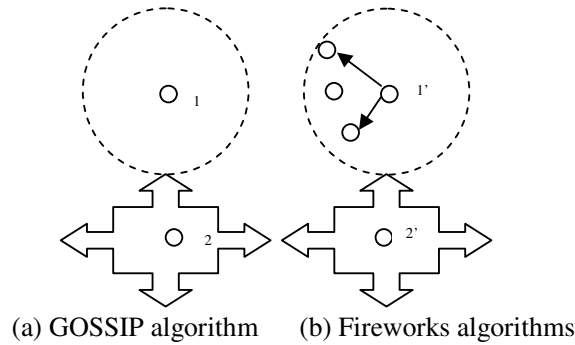


Figure 4. Comparison of two gossip-based algorithms.

## 5. IMPLEMENTATION

We conducted the simulation of epidemic interest flooding with a hybrid metric  $Cost_p$  [17] (we use  $delay * ETX^3$  in the simulation) for a modified directed diffusion in ns2. We measure the throughput and delay performance in different topologies with different network sizes and identify the optimal percentage of neighbors that drop interests.

### 5.1. Simulation Methodology

There is a source and a sink in each of the four topologies: grid, triangular tessellation, hexagon tessellation (Figure 5) and random. For the first three topologies, the source is at the bottom left corner and the sink is at the top right corner. Both are one hop inside the border. For the random topology, we manually select source-sink pairs whose distances are almost the same. We simulate the algorithm using a modification of directed diffusion release 3.2.0 in ns2.29 with the error and interference model introduced in the earlier section.

We use the IEEE 802.11 protocol for the MAC layer with a channel bandwidth of 2 Mbps. The transmission range is 250m and the interference range is 550m. In the first two topologies, the distance between the closest pair of nodes is 250m. We fix the size of random topology to 2000m by 2000m with different numbers of nodes. The simplest *CBR* traffic is generated at the rate of 10 packets per second.

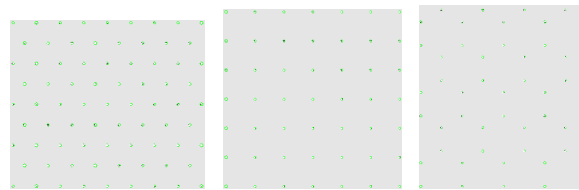


Figure 5. Topologies of triangular, grid and hexagon tessellation, from left to right. Every node has 6, 4, and 3 neighbors, respectively.

We plug in the *BER* formula (we use a much smaller constant 0.000005 instead of 0.5 because it matches the experimental data more accurately) into the interference model and calculate *FER* from *BER*. To make it simple, we set the packet size the same as frame size so that packet loss rate is *FER*. Although *ETX* can be computed by broadcasting probe packets before interest flooding, to simplify the simulation, we put the *FER* into the common header of ns2 and let the protocol read *FER* and calculate *ETX*. The noise levels we use in the error model are from the specification of the Orinoco cards. Ns2 has a global scheduler to record all the traffic so that the error model is able to capture the dynamic interference.

## 5.2. Performance Evaluation

We run each setting, with different topology and percentage of nodes that do not drop interests, for 50 times and take the average of the metrics. In our discussions below, let  $p$  be the fraction of nodes that do not drop interests. We compare the throughput, end-to-end delay and directed diffusion related metrics, such as, percentage of cases where no interest reaching the source, percentage of cases where no path was built (which includes the previous cases) and percentage of cases with zero throughput (which includes the previous two cases). Each run lasts for 300 seconds.

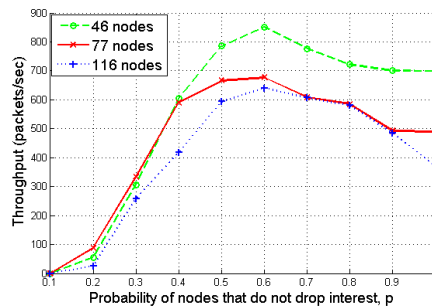


Figure 6. Throughput of triangular tessellation topology with different sizes.

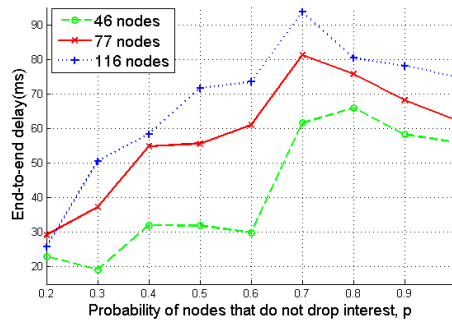


Figure 7. End-to-end delay of triangular tessellation topology with different sizes.

We first run the application for the triangular tessellation with different network sizes as shown in Figure 5 and measure the throughput (Figure 6) and end-to-end delay (Figure 7). The throughput increases rapidly when  $p$  increase from 0.1 to 0.5. It achieves the highest throughput when 60% of the nodes do not drop interests. Beyond 60%, the throughput drops because of the extra interference during exploratory data phase when more or all candidate gradients are considered (fewer or no node drop interests). Note that when  $p=1.0$ , it becomes the original directed diffusion algorithm and the throughput is lower than the peak, when  $p=0.6$ , which is 20% higher than the original directed diffusion. When many nodes drop interests, i.e.  $p$  is small, the throughput is lower because the number of candidates is very small. Network with smaller



sizes achieves higher throughput since it suffers less from cumulative packet loss. Figure 7 shows that end-to-end delay goes up to 80 ms when  $p=0.7$  or  $p=0.8$  and drops down after that. The higher delay is due to high traffic when the throughput peaks. Larger networks have longer delay since there are more hops between the source and the sink. Trade-off between throughput and delay is also shown in these two figures. Although we do not have the highest delay when  $p=0.7$  where throughput is maximum, we believe they reflect the trend that there should be a magic percentage, between 60% and 70% (or 80%), for this topology.

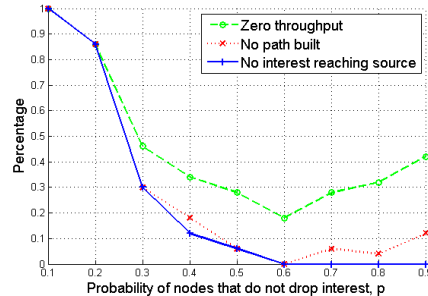


Figure 8. Percentage of zero throughput, no path built and no interest reaching source of triangular tessellation topology with 77 nodes.

One of the shortcomings of probabilistic forwarding is that in some cases no forwarding path for interest, exploratory data or application data, may be found. We analyze the above-mentioned probability phase by phase. In the first phase, interests may fail to reach the source due to the epidemic flooding. We measure the percentage of such instances in Figure 8. If more nodes are updating the gradient table, the percentage of interests reaching the source is monotonically higher. Also, we compare the percentage of cases where no path was built, which includes the cases where no interest reaches the source. Another reason for this might be that interference causes exploratory data to be dropped half way even if interests can reach the source successfully. As a result, it is no surprise to find that when  $p=0.6$ , more paths are built than when  $p=1$  (original directed diffusion). Percentage of cases with zero throughput (plotted in the same figure) includes the cases where no path is built. When  $p=0.6$  we not only achieves the highest throughput, but also achieve the highest percentage of successful delivery of application data.

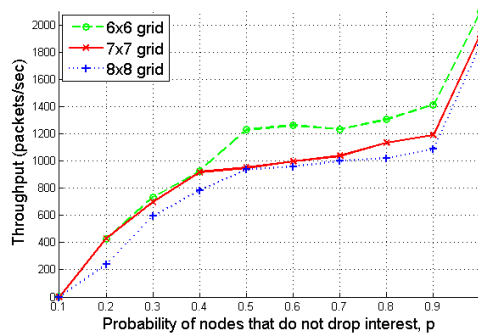


Figure 9. Throughput of grid topology with different sizes.

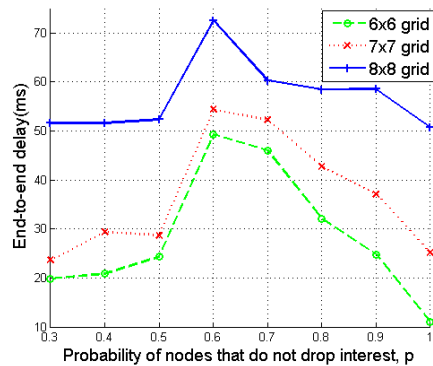


Figure 10. End-to-end delay of grid topology with different sizes.

Figure 9 and Figure 10 show the throughput and delay in grid topology with different network sizes. Delay follows almost the same trend as in the previous topology although the magic percentage is 0.6 here. Throughput looks quite different in grid topology and it keeps increasing till all nodes update their gradient tables. The reason is that every node has only 4 neighbors, which is small and hard to guarantee connectivity. Compared to the previous topology with 6 neighbors, the product of 6 and 0.6 is approximately 4, which is evidence that a magic number, instead of magic percentage, does exist for the number of neighbors with different topologies.

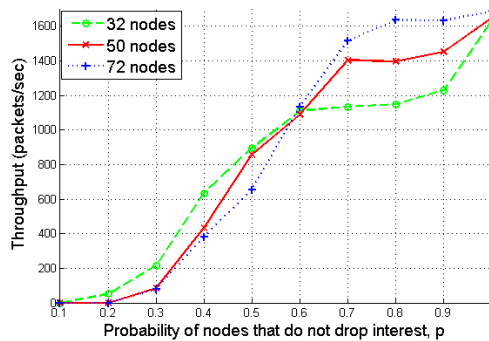


Figure 11. Throughput of hexagon tessellation with 32, 50, 72 nodes.

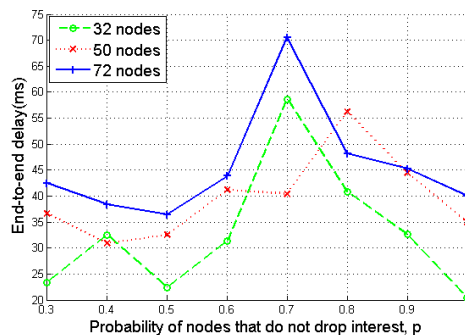


Figure 12. End-to-end delay of hexagon tessellation with 32, 50, 72 nodes.

Hexagon tessellations with different sizes are compared in Figure 11 and Figure 12 in terms of throughput and end-to-end delay. Throughput keeps increasing with higher percentages of nodes that do not drop interests since the topology has only 3 neighbors, even smaller than grid

topology. The peak of end-to-end delay is  $p=0.7$  or  $p=0.8$ , which is higher than grid topology at 0.6. The highest delay is due to high traffic level. For throughput,  $p=0.6$  is a breakpoint, beyond which smaller network sizes have larger delay than larger network sizes. We explain this phenomenon as: the probability of larger networks having connectivity increases much more rapidly than that of smaller networks. Figure 12 shows that in all network sizes, the highest end-to-end delay occurs at  $p=0.7$  or  $p=0.8$  due to the high traffic level.

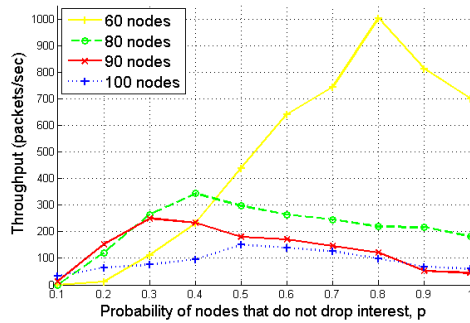


Figure 13. Throughput of random topology with 60, 80, 90, 100 nodes.

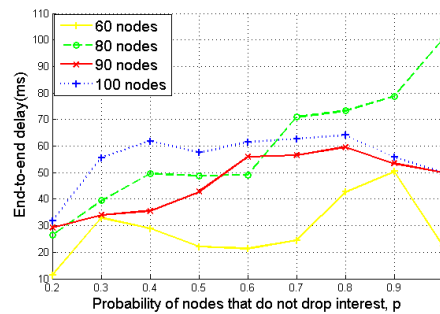


Figure 14. End-to-end delay of random topology with 60, 80, 90, 100 nodes.

We also test random topology with different network sizes (Figure 13 and Figure 14). Throughput always follows the “up and down” trend and the network with 60 nodes performs the best. The magic percentage varies between 0.3 and 0.8. Network size of 60 nodes achieves the highest throughput at a higher percentage than that of 80, 90, or 100 nodes because every node has fewer neighbors. The delay graph (Figure 14) also indicates the magic number. The delay of 80-node topology keeps increasing till  $p=1$ . Smaller networks achieve lower end-to-end delay than larger ones most of the time. Since nodes are uniformly distributed in this random topology, given the transmission range, area of the network, and number of nodes, we calculate the average number of neighbors is 3.9 (almost 4) in the 80-node network and 2.9 (almost 3) in the 60-node network. We may give the same explanation as in Figure 9 or Figure 10 although that is for throughput.

To show the magic number for different topologies, we put all four of them in the same figure (Figure 14 and Figure 15). We keep the hop count between the source and the sink the same for them. In Figure 14, hexagon tessellation performs the best and random topology the worst. Fewer neighbors cause less interference. Although the average number of neighbors of random topology is 4, the uneven deployment aggravates the side-effects of interference. An interesting thing is that hexagon tessellation throughput exceeds that of the 7 by 7 grid at  $p=0.6$ . The reason is that it is hard for the hexagonal tessellation topology to maintain connectivity with two few neighbors. Hexagon tessellation also performs best in end-to-end delay.

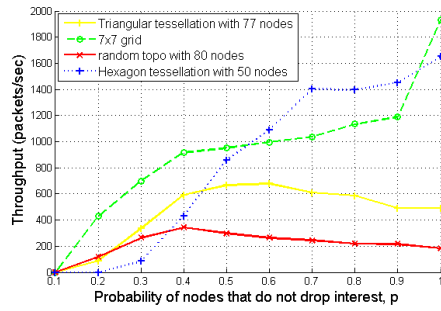


Figure 14. Throughput comparison of 4 topologies.

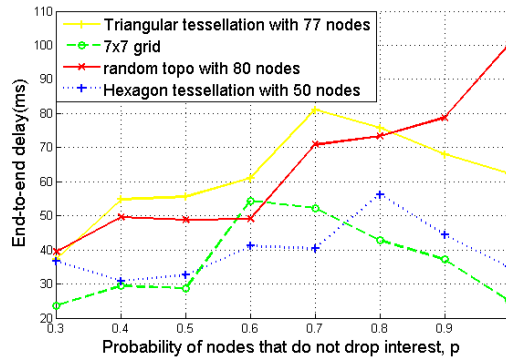


Figure 15. End-to-end delay comparison of 4 topologies.

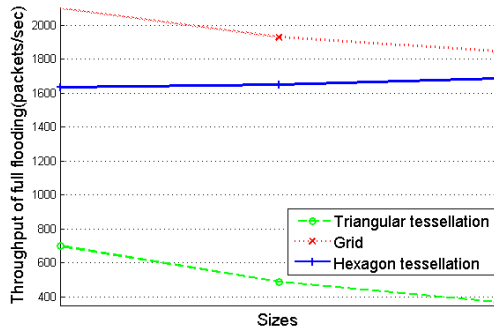


Figure 16. Throughput comparison of 3 topologies with full flooding.

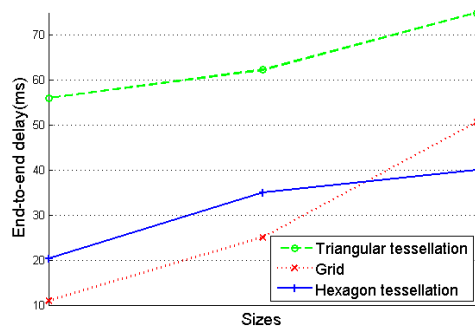


Figure 17. End-to-end delay comparison of 3 topologies with full flooding.

To further test the detrimental effect of full flooding, we put the throughput (end-to-end delay) of the 3 regular topologies with different sizes in the same figure (Figure 16 and Figure 17). We still make the hop count between the source and the sink the same for each point in the same column. Hexagon tessellation is affected less than others in both throughput and end-to-end delay since it has the smallest number of neighbors. The grid topology has the best overall throughput and delay performance since it has the magic number of neighbors.

## 6. CONCLUSION

In this paper, we propose a set of epidemic flooding algorithms for QoS-based directed diffusion, which makes the routing decision more accurate by reducing interference level during route discovery, as shown in our simulation results. We observe that different topologies have different magic percentages of neighbors to achieve optimal routing decision (which is different from the connectivity problem). We also identify the magic number of neighbors that can be applied to different topologies for optimizing throughput and delay.

For all the topologies we have tested, the optimal number of neighbors for achieving the best throughput is four. When the number of neighbors is greater than four, maximum throughput is achieved when some nodes drop interests, i.e. the percentage of nodes not dropping interest is less than 100%. In these cases, when more nodes drop interests, the level of interference is reduced and higher throughput is achieved in the resulting routes. As the number of neighbors decreases, the optimal percentage of nodes that do not drop interests for achieving maximum throughput moves toward 100%. In most cases, delay is highest when the percentage of nodes not dropping interest is optimal. Random topologies perform worse than topologies with regular tessellation no matter what density we choose because of the uneven distribution of nodes resulting in hotspots with high interference level and difficulty for the routing algorithm to estimate interference characteristics.

## REFERENCES

- [1] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," in *Proc. ACM MobiCom '99* Seattle, WA, 1999, pp. 174-185.
- [2] C. Intanagonwivat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proc. ACM MobiCom '00* Boston, MA, 2000, pp. 56-67.
- [3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. "Energy-Efficient Communication Protocols for Wireless Microsensor Networks". In *Proceedings of Hawaiian International Conference on System Science*, January 2000.
- [4] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," in *Proceedings of MOBICOM 2000*, Boston, MA, USA, 2000, pp. 243-254.
- [5] Y. Xu, J. Heidemann, D. Estrin, "Geography-informed energy conservation for ad hoc routing," *MobiCom 2001*, Rome, Italy, pp. 70-84, July 2001.
- [6] Y. B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Proc. Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 66-75, 1998.
- [7] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proc. Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 76-84, 1998

- [8] D. B. Johnson and D. A. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*. Kluwer Academic Publishers, 1996.
- [9] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90-100, February 1999.
- [10] Z. Haas and M. Pearlman. The performance of query control schemes for the zone routing protocol. In *Proc. ACM SIGCOMM*, pages 167-177, August 1998.
- [11] V. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proc. INFOCOM*, pages 1405-1413, April 1997.
- [12] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinchar, D. Terry, epidemic algorithms for replicated database maintenance, in: *Proceedings of the sixth ACM Symposium on Principles of Distributed Computing*, Vancouver, Canada, August 1987.
- [13] Zygmunt Haas, Joseph Halpern, and Li Li. Gossip based ad hoc routing. In *INFOCOM*, New York, USA, June 2002.
- [14] L. Orecchia, A. Panconesi, C. Petrioli, A. Vitaletti, Localized techniques for broadcasting in wireless sensor networks, *Proceedings of the 2004 joint workshop on Foundations of mobile computing*, October 01-01, 2004, Philadelphia, PA, USA.
- [15] Anonymous Gossip: Improving Multicast Reliability in Mobile Ad-Hoc Networks, *Proceedings of the The 21st International Conference on Distributed Computing Systems*, p.275, April 16-19, 2001.
- [16] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, S. Wicker, An empirical study of epidemic algorithms in large scale multihop wireless networks, *Tech. Rep. IRB-TR-02-003*, Intel Research, March 2002.
- [17] S. Li, A. Lim, S. Kulkarni, C. Liu, "EDGE: A Routing Algorithm for Maximizing Throughput and Minimizing Delay in Wireless Sensor Networks", *Proceedings of the 26th Military Communications Conference (MILCOM'07)*, October 2007.
- [18] P. Chaporkar, K. Kar, and S. Sarkar. Throughput guarantees through maximal scheduling in wireless networks. In *Proc. 43rd Allerton Conf. on Commun., Control, and Comp.*, September 2005.
- [19] Kamal Jain , Jitendra Padhye , Venkata N. Padmanabhan , Lili Qiu, Impact of interference on multi-hop wireless network performance, *Proceedings of the 9th annual international conference on Mobile computing and networking, September 14-19, 2003, San Diego, CA, USA.*
- [20] J. J. Lemmon, "Wireless Link Statistical Bit Error Model," NTIA Report 02-394, June 2002.
- [21] S. Lee, B. Bhattacharjee, and S. Banerjee. Efficient Geographic Routing in Multihop Wireless Networks. In *ACM Mobihoc'05*, May 2005.
- [22] R. Draves, J. Padhye, and B. Zill, Comparison of Routing Metrics for Static Multi-Hop Wireless Networks. *Proc. of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pages 133-144, ACM Press, 2004.
- [23] D. D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. *Proc. of MobiCom*, 2003.
- [24] R. Draves, J. Padhye, and B. Zill, Routing in multi-radio, multi-hop wireless mesh networks. *Proc. of the 10th Annual International Conference on Mobile Computing and Networking*, 2004.
- [25] J. Park and S. Kasera, Expected data rate: an accurate high-throughput path metric for multi-hop wireless routing. *Proc. of second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, 2005.

## Authors

Shuang Li is a PhD student in the department of Computer Science and Engineering at the Ohio State University. She received her M.S. degree from Auburn University, Auburn, Alabama in 2008. Her research interests are routing metrics in wireless sensor networks, power control and resource allocation in MANETs and VANETs.



Alvin Lim is currently an associate professor of computer science and software engineering at Auburn University. He received his Ph.D. degree in computer science from University of Wisconsin at Madison in 1993. His research interests include self-organizing sensor networks, mobile and pervasive computing, network security, wireless networks, reliable and dynamically reconfigurable distributed systems, complex distributed systems, mobile and distributed databases, distributed operating systems, and performance measurement and analysis. He has published widely in journals and conferences in these networking and distributed systems areas. He is a subject area editor of the International Journal of Distributed Sensor Networks. His work had been supported by the National Science Foundation, the DARPA SensIT program, U.S. Air Force Research Lab and the U.S. Army.



Cong Liu is a PhD student in the department of Computer Science at the University of North Carolina – Chapel Hill. He received his Master's degree from the department of Computer Science and Software Engineering at Auburn University in Dec. 2007. His current research interests are real-time systems and scheduling theory on real-time multicore platforms.

