A Traffic-aware Power Management Protocol for Wireless Ad Hoc Networks

Xiaobing Hou and David Tipper
Department of Information Science & Telecommunications
University of Pittsburgh
{xiaobing, dtipper}@mail.sis.pitt.edu

Shuju Wu
School of Polytechnic Studies
Southeast Missouri State University
swu@semo.edu

***Abstract*— Recently, we have proposed an energy saving *Gossip-based Sleep Protocol* (GSP) [1] for energy conservation in mobile ad hoc networks. A gossiping node employs a random variable (p) to decide whether or not to enter a sleep mode. Our observation is that in a well connected ad hoc network there are usually many paths existing between a source and a destination, so a percentage (p) of the nodes may be in an energy conserving sleep mode without losing network connectivity. However, putting nodes into a sleep mode may interrupt the ongoing traffic. This can result in a long packet delay and a large packet loss rate. In this paper, we extend GSP to consider the ongoing traffic before changing the radio mode of a node. We term it *Traffic-aware GSP* (T-GSP) and show its advantages through simulation studies.**

***Index Terms*— energy efficiency, ad hoc network, gossiping, sleep mode**

I. INTRODUCTION

In an ad hoc network, mobile nodes must cooperate to dynamically establish routes using multihop wireless links. There is no stationary infrastructure and each node acts as a router. A packet may have to be forwarded by a sequence of nodes to reach its destination. Many routing protocols have been proposed to solve the dynamic multihop routing problem in ad hoc networks. There are two general classes of routing protocols for ad hoc networks, proactive routing and reactive routing. Proactive routing protocols attempt to maintain consistent, up-to-date routing information from each node to every other node in the network. Such protocols are termed proactive because they store route information before it is needed. Proactive protocols suffer the disadvantage of additional control traffic that is needed to continually update stale routing entries. Some of the most popular proactive protocols are DSDV [2], WRP [3], OLSR [4] and FSR [5].

Reactive routing creates and maintains routes only when desired by the source node. Therefore, it's also known as on-demand, source-initiated, or demand-driven routing [6]. When a node requires a route to a destination, it initiates a route discovery process within the network, typically, by some form of flooding. This process is completed once a route is found or all possible route permutations have been determined. Once a route has been established, it is maintained by a route maintenance procedure until either the route is no longer desired or the destination becomes inaccessible. Compared to proactive routing, reactive routing consumes far less bandwidth for maintaining the routing tables at each node when only a small subset of all available routes is in use at any time, at the expense of high overhead and delay in setting up the route. Proposed reactive routing protocols include DSR [7], [8], AODV [9], and TORA [10]. A review of ad hoc routing protocols is given in [6].

Since most mobile hosts are not connected to a power supply and battery replacement is difficult, optimizing energy consumption in these networks has high priority. Conventional ad hoc routing protocols, as introduced above, require all nodes keep listening even if there is no traffic or neighbor nodes are totally redundant for each other. Obviously, this wastes energy and significantly reduces the lifetime of the nodes as well as the network's. In this paper, we propose a method for improving the energy efficiency of ad hoc networks by employing a statistical sleep mode while considering ongoing traffic. It can also be used in sensor networks, which can be seen as a special case of ad hoc networks with lower mobility and a tighter energy budget. Our design has been driven by the following three goals:

- Simplicity: mobile hosts may have limited computing capability and memory resources. Minimized operation and information maintenance are required.
- Scalability: an ad hoc network could be composed of a great number of nodes.
- Connectivity: network connectivity can keep the path

Based in part on "Traffic-aware Gossip-based Energy Conservation for Wireless Ad Hoc and Sensor Network Routing", by Xiaobing Hou, David Tipper, and Shuju Wu which appeared in the Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC) 2006, January 2006. © 2006 IEEE.

setup delay low.

With these goals in mind, we propose the Traffic-aware Gossip-based Sleep Protocol (T-GSP), which is based on the Gossip-based Sleep Protocol (GSP) proposed in [1]. It randomly places nodes in an energy saving sleep mode for a random interval based on a random variable (p), so that the network lifetime can be prolonged. The novelty of the T-GSP compared with the previous work (i.e., GSP) is that it adjusts the variable (p) based on the ongoing traffic to improve the network performance.

By introducing a sleep mode into the network, the total energy consumption of the network can be reduced and the network lifetime can be prolonged. However, the sleep mode may increase the length and the failure rate of a path. Therefore, simulation is conducted to study the effect of T-GSP based on the network lifetime, throughput and end-to-end delay.

The remainder of this paper is organized as follows. In section II, we present a brief review of current ad hoc and sensor network energy efficient routing protocols. Section III presents our Traffic-aware Gossip-based Sleep Protocol (T-GSP). Section IV presents the results of a simulation study. We conclude our work and point out some possible future work in section V.

II. RELATED WORK

Basically, there are two classes of energy efficient ad hoc and sensor network routing protocols employing a sleep mode in the literature, cluster-based and flat. Both of them achieve energy efficiency by employing different topology management techniques. This section presents a brief review of these two classes of routing to provide a better understanding of the current research issues in this area.

In cluster-based routing protocols, all nodes are organized into clusters with one node selected to be cluster-head for each cluster. This cluster-head receives data packets from its members, aggregates them and transmits to a data sink. In some cluster-based routing protocols, the cluster-head assigns TDMA slots to its members to schedule the communication and the sleep mode. Low-Energy Adaptive Clustering Hierarchy (LEACH) [11] is designed for proactive sensor networks, in which the nodes periodically switch on their sensors and transmitters, sense the environment and transmit the data. Nodes communicate with their cluster-heads directly and the randomized rotation of the cluster-heads is used to evenly distribute the energy load among the sensors. Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [12] is designed for reactive networks, where the nodes react immediately to sudden changes in the environment. Nodes sense the environment continuously, but send the data to cluster-heads only when some predefined thresholds are reached. Adaptive Periodic Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) protocol [13] combines the features of the above two protocols by modifying TEEN to make it send periodic data. The cluster-based routing protocols can arrange the sleep mode of each node to conserve energy. However, the high complexity and overhead are incurred.

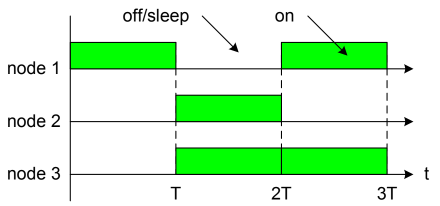
Flat schemes do not maintain the hierarchical structure. SPAN [14] forms a multi-hop forwarding backbone to preserve the original capacity of the network. Other nodes can go to sleep more often to conserve their energy. Backbone functionality is rotated among the nodes to balance the energy consumption. Geographic Adaptive Fidelity (GAF) [15] conserves energy by identifying nodes equivalent from a routing perspective and turning off unnecessary nodes. The network is divided into grids so that all nodes in the adjacent grids can communicate with each other directly. At each point in time, only one node in each grid is active. GPS or other positioning system is required to get the location information for grid formation. Sparse Topology and Energy Management (STEM) [16] exploits a separate paging channel to wake up nodes to trade off setup latency for energy savings. A low duty cycle radio is used to reduce the energy consumption of the paging channel. In the on-demand scheme from [17], connectivity is only maintained between pairs of senders and receivers and along the route of data communication and the nodes that do not carry any traffic can transit to power-save mode. The transition from power-save mode to active mode is triggered by communication events such as routing control messages or data packets. With the on-demand concept, more energy can be conserved, however, applications may face a long route setup delay.

Recently the concept of applying results of percolation theory to the development of sleep protocols has been advocated [1] [18]. In [1] we incorporate percolation theory or gossiping, into the sleep mode concept in a fashion such that it can be easily incorporated into conventional ad hoc and sensor network routing protocols and a simulation study is presented to evaluate it's effectiveness. A similar idea is used in [18] to study the network connectivity and coverage. In [19], we extended our work to consider the ongoing traffic before changing the radio mode of a node. In this paper, we provide a thorough development and evaluation of the use of traffic-aware gossiping in developing sleep modes and show how the proposed protocol allows one to better trade off network density for energy efficiency with local traffic information. Specifically, in addition to the previous work, we study the performance of the protocol in a partitioned network under different sleep probability and mobility level. Compared with other techniques, the one proposed here is very simple and scalable.

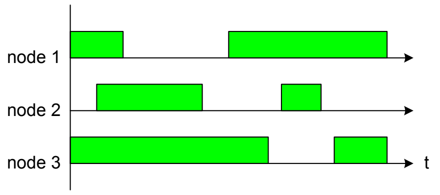
III. TRAFFIC-AWARE GOSSIP-BASED SLEEP PROTOCOL

A. Gossip-based Sleep Protocol (GSP)

As mentioned in section I, the current ad hoc network routing protocols require all the nodes to be awake and keep listening. This wastes a lot of energy. The energy efficient routing protocols and topology management schemes introduced in section II increase the computing complexity and require extra overhead or equipment.



(a) GSP1



(b) GSP2

Fig. 1. Nodes switch on and off in GSP (GSP1: synchronous GSP; GSP2: asynchronous GSP)

Therefore, we proposed a novel scheme termed Gossip-based Sleep Protocol (GSP) in [1]. The core idea of the GSP is to employ probabilistic based sleep modes - essentially, tossing a coin to decide whether or not a node should sleep for the next period. With a certain value of gossip sleep probability p (i.e., *gossip sleep probability*) and under certain topology density, the network is still connected, thus works properly.

The gossiping approach used here implements concepts from percolation theory [20] [21]. In a static infinite network (infinite nodes in an infinite space), if every link or node is open/active with probability p, the network will be grouped into clusters. We are interested in the size and the shape of the clusters as p varies from 0 to 1. Results from percolation theory show that there exists a critical value $p_c > 0$ such that in the *subcritical phase* (when $p < p_c$), nodes form finite clusters almost surely; in the *supercritical phase* (when $p > p_c$), however, there exist a unique infinite cluster almost surely. The fraction of nodes belonging to this infinite cluster determines the quality of the connectivity. To date, there is unfortunately no explicit expression of this fraction, nor of p_c. However, we can obtain approximations via simulation, as shown in [1].

Two versions of GSP have been proposed in [1], synchronous and asynchronous. With the synchronous GSP or GSP1, all nodes decide their next sleep state at the same time and repeat this periodically. With the asynchronous GSP or GSP2, every node stays in a sleep mode for a uniformly random time interval (i.e., *gossip interval*) and decides its next state independently. Fig. 1 shows the radio mode switching in GSP1 and GSP2. The asynchronous GSP generates more path failures, but it removes the synchronization requirement.

Unlike other protocols using a sleep mode (e.g., cluster-based schemes, SPAN and GAF), GSP is extremely simple and requires almost no information, even from immediate neighbors. The gossip sleep probability p is purely dependent on the network density and can be configured before the deployment of the network. GSP improves upon the energy consumption by schemes such as Span and GAF by not requiring nodes to transmit and receive additional network maintenance traffic. On the other hand, GSP is expected to provide less improvement of the network lifetime than other schemes due to the limited knowledge of the network, which contributes to the simplicity as we just mentioned. Therefore, GSP is more suitable to the large low-cost network, which desires low complexity to reduce the cost of every node as much as possible.

B. *Traffic-aware Gossip-based Sleep Protocol (T-GSP)*

In GSP, putting nodes into a sleep mode may interrupt the ongoing traffic. This may result in a longer packet delay and a higher packet loss rate, especially in a reactive routing protocol, where the setup delay of a new path is much longer than proactive protocols. Therefore, we propose the Traffic-aware Gossip-based Sleep Protocol (T-GSP) to reduce the probability of breaking an active communication when employing the gossiping technique to a node.

The goal of T-GSP is to base energy management decisions on traffic patterns in the network. By reacting to changes in these patterns, nodes that carry the traffic should stay awake with a higher probability while the other nodes can go to the sleep mode with the regular GSP sleep probability. We can see the key idea of T-GSP is that transitions from active mode (i.e., send, receive or idle mode) to sleep mode should be triggered by a lack of active communication in addition to the gossiping probability. The lack of active communication is determined by a *traffic timer*, which is refreshed by the transmission or reception of packets at a node. The expiration of the timer of a node indicates that there may be no traffic using this node.

The value of the traffic timer can be determined by the type of packet received by a node. Different types of packets indicate different time intervals before the incoming of data transmission. Therefore, integration with a routing protocol and understanding the semantics of the received packets can help to determine the timer values. The basic concepts in determining a timer value discussed in [17] can be employed in our scheme. The flooded control packets provide a poor hints of data transmission. On the contrary, data packets usually indicate the possible arrival of more of the same type of packets. Additionally, some special control packets, such as route reply packets in reactive routing protocols and query packets in sensor networks, provide a good hint that subsequent packets will follow this route. Therefore, the timer should be set to a value on the order of the data packet interarrival time, or the end-to-end delay between a source and a destination, whichever is greater. Since the packet interarrival time varies with traffic patterns, the timer value also varies with traffic patterns. Integrated with GSP, this scheme expects a gossip interval [1] larger than the traffic timer value. This can be easily achieved since a gossip interval smaller than

TABLE I
ENERGY CONSUMPTION MODEL FOR LUCENT IEEE 802.11 WAVELAN PC CARD WITH $2Mbps$

Radio mode	Energy Consumption (W)
Transmit	1.327
Receive	0.967
Idle	0.844
Sleep	0.066

TABLE II
SIMULATION PARAMETERS

Parameters	Values	Parameters	Values
Simulation time	$\geq 400sec$	Bandwidth	$2Mb/s$
Physical layer	IEEE 802.11b	Max. speed	$20m/s$
MAC layer	IEEE 802.11b	Radio range	$250m$
Traffic model	CBR/on-off	Gossip interval	$20sec$
Packet size	$512bytes$	Traffic timer	$1sec$
Network size	$100nodes$	CBR rate	$5/10$
Traffic nodes	10	on-off rate	$10/20$
Area size	$1500 \times 300m^2$	Pause time	$0/inf.$

data packet interarrival time and end-to-end delay is not a good choice for GSP.

Before its traffic timer expires, a node should stay awake with a higher probability to reduce the possibility of breaking an active communication which is using this node. The nodes carrying no traffic can switch to sleep mode with a regular GSP sleep probability [22]. We can see that a node requires no information from other nodes in T-GSP.

T-GSP has the same properties as GSP compared with other schemes (e.g., SPAN and GAF), i.e., they are both extremely simple and require no information from neighbors. Similarly, they are both expected to provide less improvement on the network lifetime than other schemes due to the limited knowledge of the network.

The major objective of T-GSP is to improve energy efficiency by putting some nodes in a sleep mode while considering ongoing traffic through a node to prevent breaking an active communication. In order to evaluate the performance of T-GSP, we have conducted a discrete event simulation based performance study.

IV. PERFORMANCE EVALUATION

A. Simulation model

We utilized the ns-2 network simulator [23], with CMU Monarch Project wireless and mobile ns-2 extensions, to study the characteristics of T-GSP. The distributed coordination function (DCF) of IEEE 802.11(b) for wireless LANs is used as the MAC layer. The radio model is similar to Lucent's WaveLAN, which is a shared media radio with a nominal bit rate of 2Mb/sec and a nominal radio range of 250 meters.

T-GSP can be integrated with a number of routing protocols. Here, we use Dynamic Source Routing (DSR) [7], [8] as an example to describe how T-GSP works. The traffic timer value is set as 1 second and a node will work for another 1 second with probability 1 after the gossip interval expires but the traffic timer doesn't. Other parameters of T-GSP are chosen to show the properties of GSP and they are not necessarily the optimal values. We use asynchronous GSP and uniformly distributed gossip intervals with an average of 20 seconds. The gossip sleep probability varies from 0.7 to 0.9. Each data point is an average of twenty runs with different random initial number seeds.

The simulation results presented in this paper are based on scenarios randomly generated by CMU ns-2 extensions. We use 100 transit nodes and nodes are randomly placed within a $1500m \times 300m$ area. The node mobility model is Random Waypoint [8], in which each node begins at a random position, picks a new random position to which to move, and moves there in a straight line at a random speed. Each node independently repeats this behavior and the average degree of mobility is varied by making each node remain stationary for a period called pause time before it moves to the next position. The smaller the pause time, the higher the average mobility. In our simulation, the maximum speed of the nodes is 20 m/s and the pause time is 0 second and infinity (i.e., static network). Besides the transit nodes, there are 10 traffic nodes, which are the source and the sink of the traffic. Long-lived CBR and exponential on-off traffic based on UDP are studied. Each packet carries 512 bytes of data payload, making the packet size 532 bytes including an IP header. The average packet rates are 5 and 10 packets/sec for long-lived CBR traffic and 10 and 20 packets/sec for on-off traffic. Both the busy and idle intervals of the on-off traffic follow an exponential distribution with a mean of 50 seconds.

Our energy consumption model is based on Feeney and Nilsson's measurements of an IEEE 802.11b Lucent WaveLAN wireless network interface operating in an ad hoc networking environment [24]. Their measurements are summarized in Table I, where we can see the sleep mode consumes only a tiny fraction of the energy of the other modes. Additional measurement values in the literature evaluating other 802.11b vendor equipments show similar energy consumption values.

To make sure the traffic does not stop before the network dies, we give traffic nodes infinite energy. The transit nodes have enough energy so that the DSR protocol can run for 400 seconds. Since all nodes in DSR keep listening even without traffic, they run out energy almost at the same time. Also, to mitigate the effects from traffic nodes, we make traffic nodes neither run T-GSP nor forward traffic in DSR. However, traffic nodes do follow the same mobility model as transit nodes and maintain their own connections as required by DSR. The above simulation parameters are summarized in Table II.

B. Simulation results

We evaluated three performance metrics defined as follows:

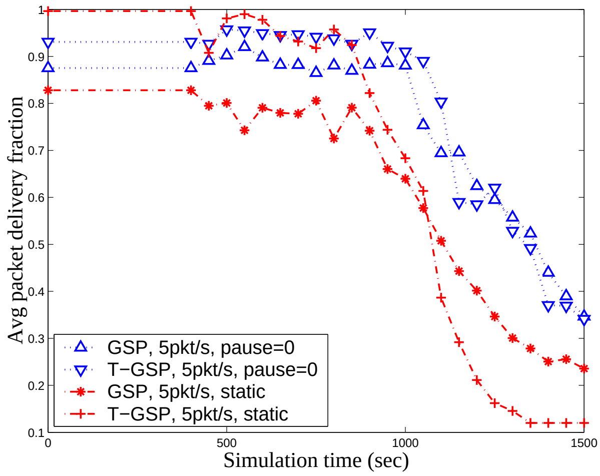


Fig. 2. Average packet delivery fraction of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

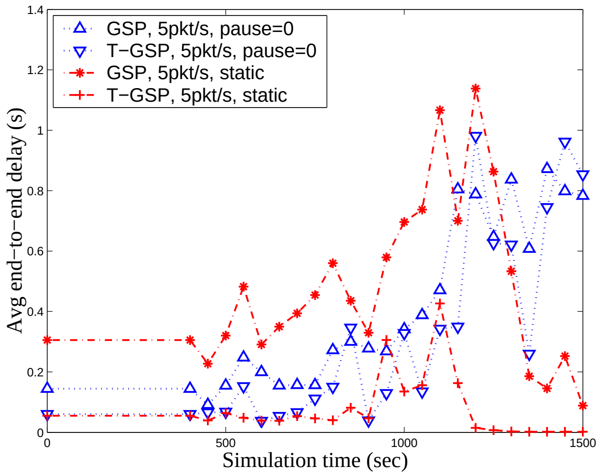


Fig. 3. Average end-to-end delay of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

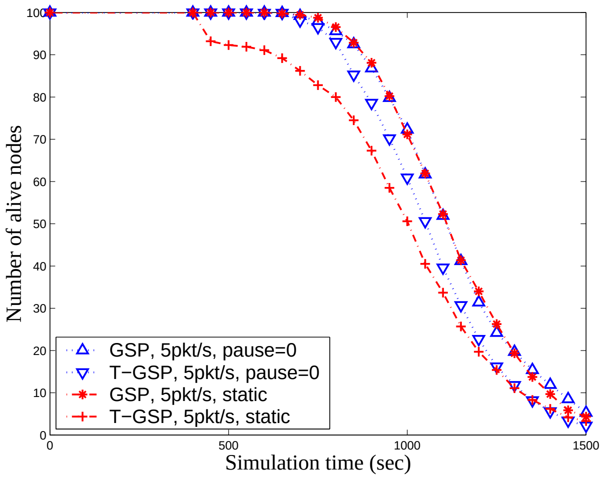


Fig. 4. Network lifetime of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

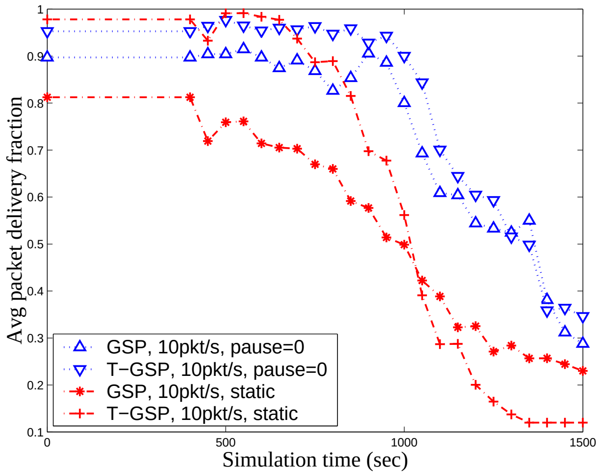


Fig. 5. Average packet delivery fraction of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

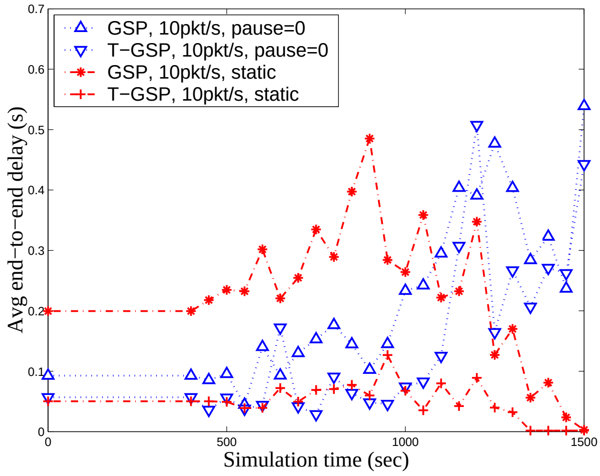


Fig. 6. Average end-to-end delay of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

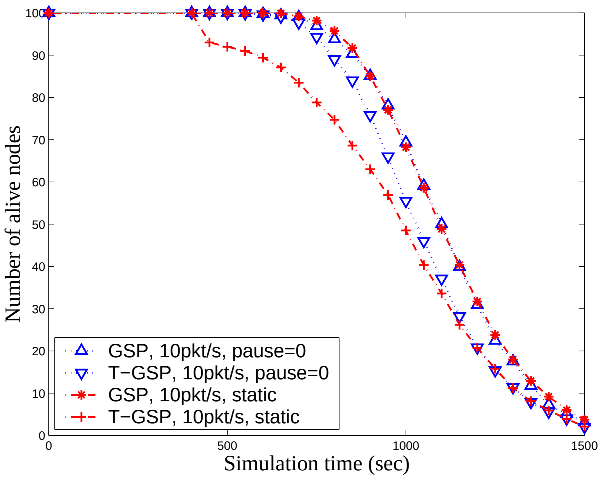


Fig. 7. Network lifetime of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.7$

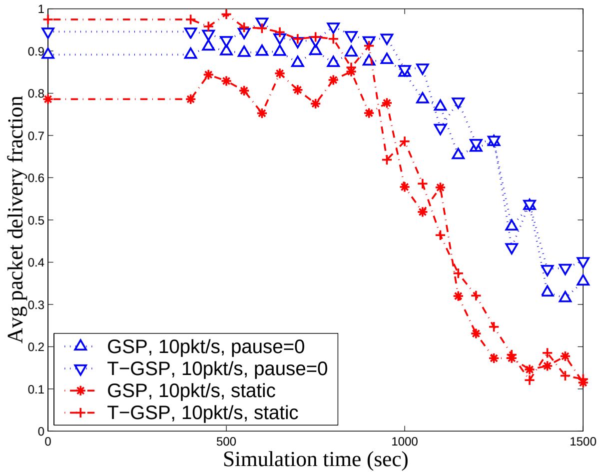


Fig. 8. Average packet delivery fraction of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

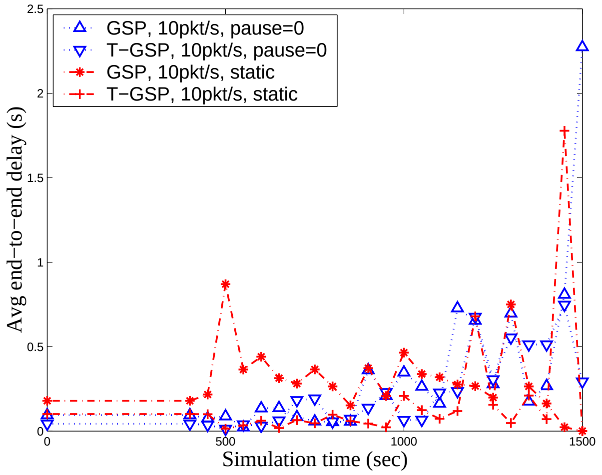


Fig. 9. Average end-to-end delay of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

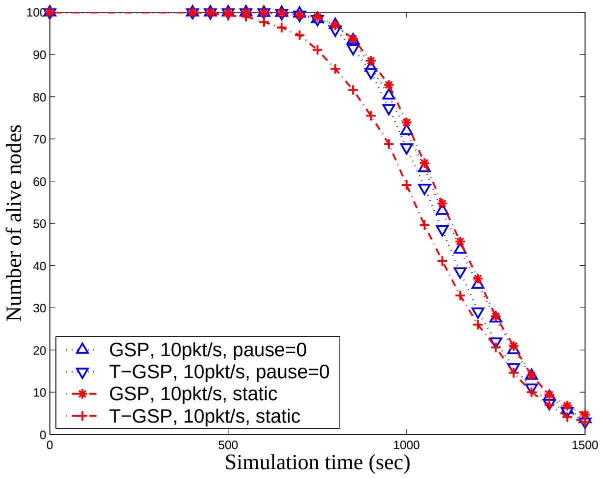


Fig. 10. Network lifetime of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

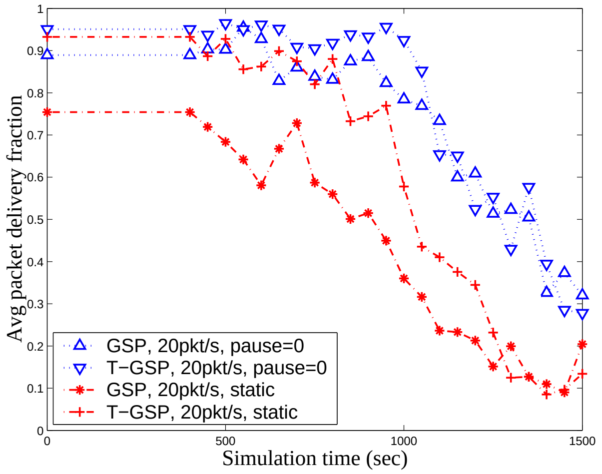


Fig. 11. Average packet delivery fraction of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

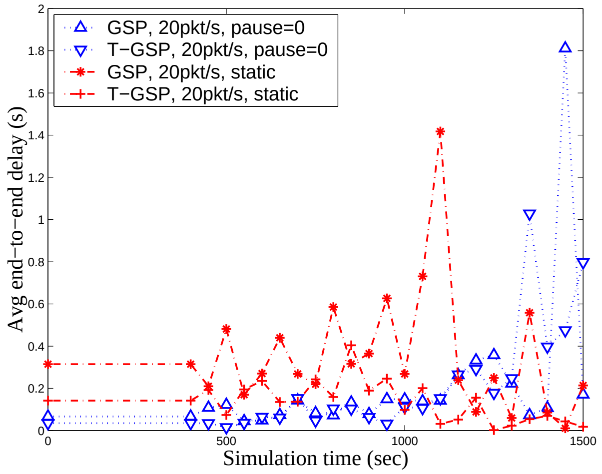


Fig. 12. Average end-to-end delay of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

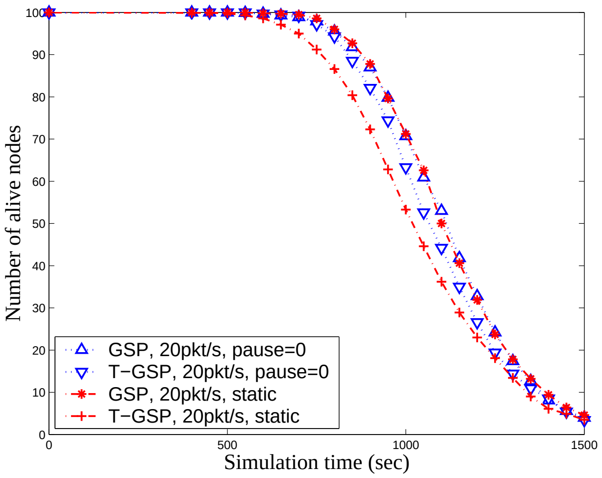


Fig. 13. Network lifetime of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.7$

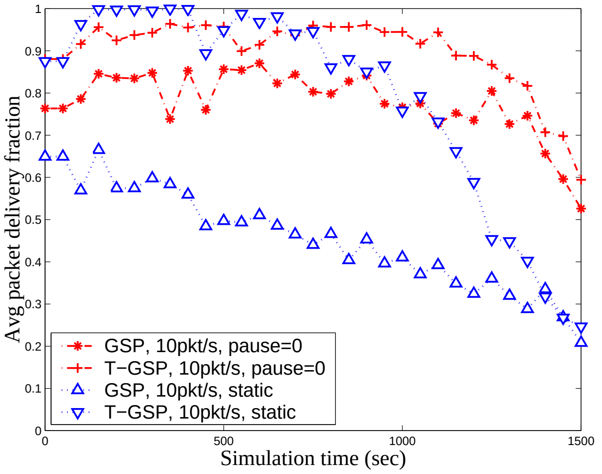


Fig. 14. Average packet delivery fraction of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.8$

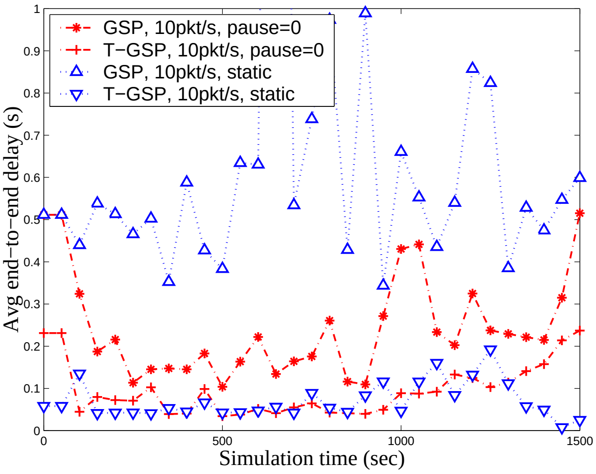


Fig. 15. Average end-to-end delay of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.8$

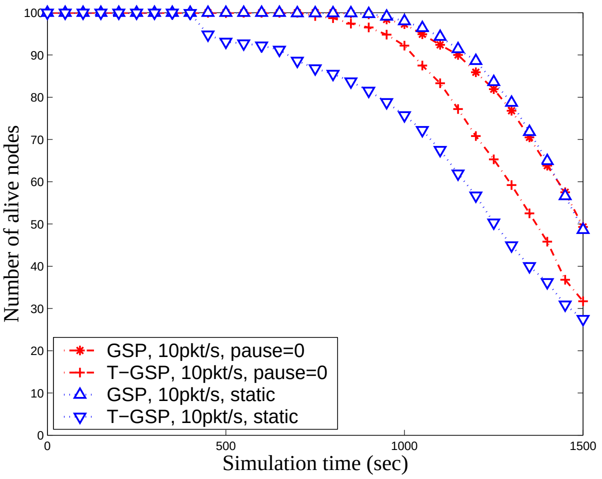


Fig. 16. Network lifetime of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.8$

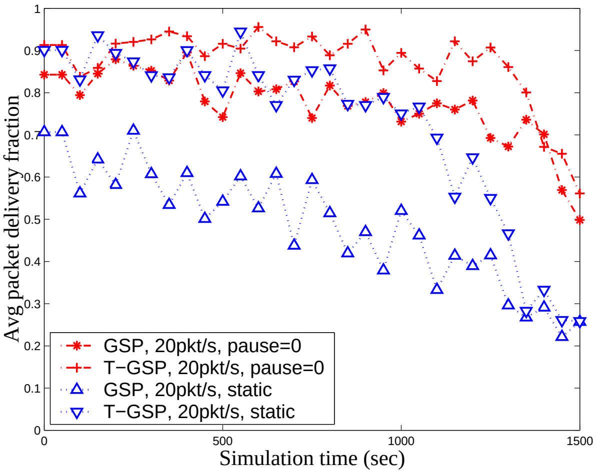


Fig. 17. Average packet delivery fraction of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.8$

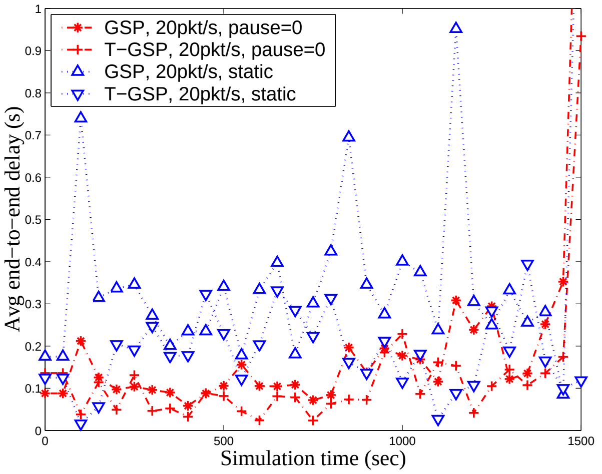


Fig. 18. Average end-to-end delay of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.8$

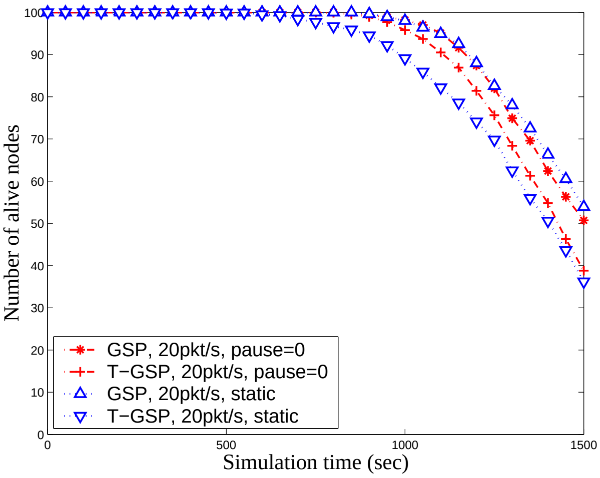


Fig. 19. Network lifetime of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.8$

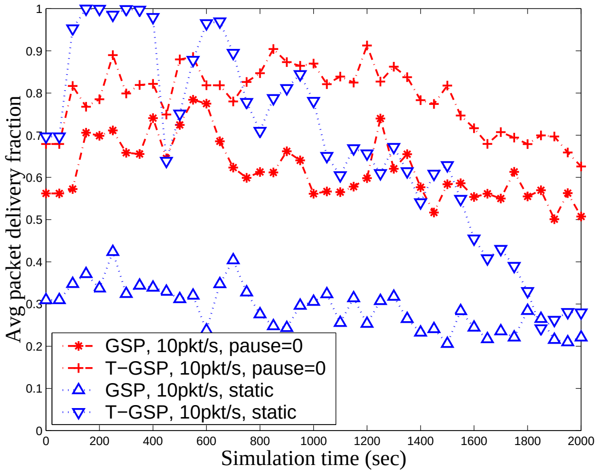


Fig. 20. Average packet delivery fraction of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.9$

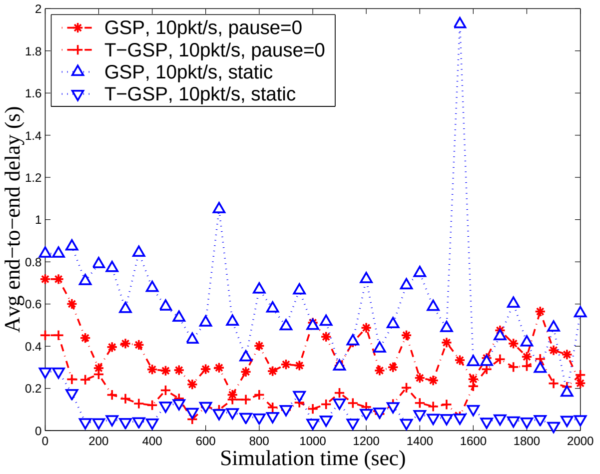


Fig. 21. Average end-to-end delay of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.9$

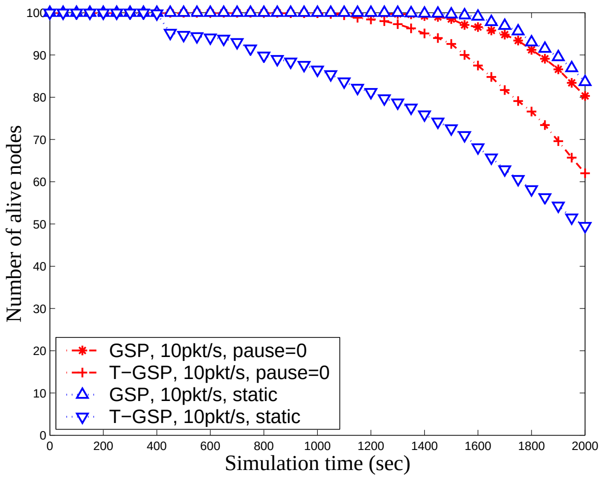


Fig. 22. Network lifetime of GSP and T-GSP with different level of mobility and long-lived CBR traffic, $p = 0.9$

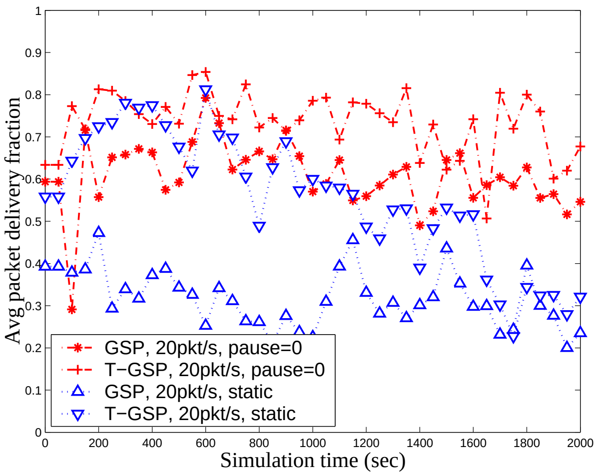


Fig. 23. Average packet delivery fraction of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.9$

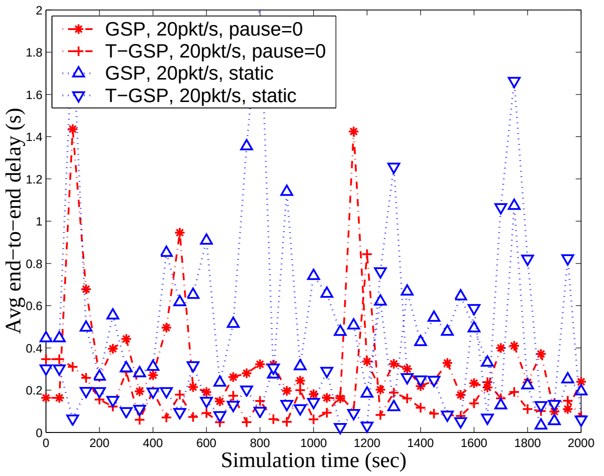


Fig. 24. Average end-to-end delay of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.9$

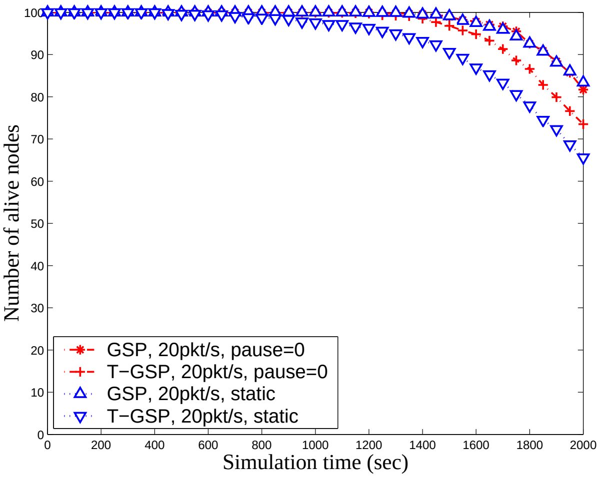


Fig. 25. Network lifetime of GSP and T-GSP with different level of mobility and exponential on-off traffic, $p = 0.9$

- Packet delivery fraction: the ratio of the number of packets delivered to the destination to those generated by the traffic sources;
- End-to-end delay: the delay experienced by each packet, including queuing delays, route discovery delays, retransmission delays at the MAC layer and the salvage process of DSR, and propagation delays;
- Network lifetime: the simulation time that a fraction of transit nodes are alive and the network maintains an acceptable packet delivery fraction and end-to-end delay.

First, let's see the results when the sleep probability is $p = 0.7$. In this case, the network connectivity can be maintained. Fig. 2, 3, 5, and 6 show the packet delivery fraction and end-to-end delay of GSP and T-GSP in case of long-lived CBR traffic with respect to the simulation time. We monitor the metrics for every 50 seconds in the figures. We can see that the packet delivery fraction and end-to-end delay can be improved in both scenarios, i.e., highly mobile network and static network. The improvement is more significant in case of a static network. The reason is that the capacity of a static network is lower than a mobile network with the same number of nodes [25] [26]. Therefore, a static network is more sensitive to the radio mode change introduced by GSP and T-GSP is able to more closely keep its original capacity.

Fig. 4 and 7 show that the network lifetime can be extended by both GSP and T-GSP. The cost of T-GSP is more energy consumption incurred by the nodes with traffic to forward. We can see that the nodes die more quickly with T-GSP than GSP. This cost is more significant in a static network. In a static network with T-GSP, a path is used until some of the nodes die. So, nodes may be overused in this case. On the other hand, mobility can break an active path and make the energy consumption more evenly distributed among all the nodes.

Fig. 8– 13 show the three performance metrics for GSP and T-GSP in the case of exponential on-off UDP based traffic. We can see that similar results are achieved. Of course, since the on-off traffic incurs more path setup, its packet delivery fraction and end-to-end delay performance is slightly lower than the one of long-lived CBR traffic. It's worth noting that the lifetime of a static network is relatively better than the one with long-lived CBR traffic. Apparently, on-off traffic can reduce the overuse of a node due to the switching of busy and idle intervals. The next busy interval may use a different path from the previous one.

The above results taken together show that by including local traffic information into the sleep mode protocol can improve the network performance. In addition, we expect that T-GSP is able to provide even greater improvements in some cases when the network is partitioned. As we know, with the basic GSP, a large sleep probability may result in a network partition breaking the communication in the network. However, the T-GSP can maintain the connectivity between an active source and destination pair while the rest of the network is partitioned. This connectivity will be maintained until the mobility or the failure of a node or link breaks it.

We increase the sleep probability to 0.8 and 0.9 and repeat the above simulation. The results of CBR traffic with rate of 10pkt/s and on-off traffic with rate of 20pkt/s are shown in Fig. 14– 25. With these high sleep probabilities, the network starts to get partitioned, especially when $p = 0.9$. T-GSP's feature of avoiding breaking an active communication can be easily seen in Fig. 14 and 20 when the network is static and the traffic is CBR. In this case, the communication will not be stopped by the mobility and idle intervals of the on-off traffic. Therefore, T-GSP can provide greater improvements and maintain high network performance. We can see that T-GSP needs some time to warm up at the beginning of the simulation since the network is partitioned and the initial delay is long. After about 400 seconds, the performance will drop down since some nodes die due to overuse. Soon the T-GSP warms up again and repeats this behavior with a higher frequency due to the lower remaining battery lives. The cycle of warmup, high performance, and drop in a static network with CBR traffic is the typical behavior of a sleep mode protocol if the network is partitioned, e.g., the on-demand scheme in [17]. Apparently, it's due to the fact that the paths in such scenarios will be used until some nodes die and the long delay to find a new path. With the mobility and the on-off process of the traffic in other scenarios, the performance of T-GSP will decrease, however, we still can easily see the significant improvements.

Compared with the on-demand scheme in [17], T-GSP has an extra parameter, i.e., the gossip sleep probability p, to adjust the network connectivity. We can easily make the network connected or partitioned based on the application requirements. A small p will maintain the network connectivity and achieve the higher performance without the long initial delay. A large p can prolong the network lifetime more and make the T-GSP behave similarly as the on-demand scheme. Unlike the on-demand scheme, T-GSP doesn't need any information from the neighbors. Therefore, T-GSP doesn't reply on any control messages at either routing layer or MAC layer.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we present an extension of the GSP [1] to consider the ongoing traffic before changing the radio mode of a node so that we can preserve the active communications in the network. In T-GSP, transitions from an active mode to a sleep mode are triggered by packets received besides the gossip interval in GSP. We implement a prototype of T-GSP in the ns-2 network simulator. Simulation results show that T-GSP can significantly improve the performance of GSP in terms of packet delivery fraction and end-to-end delay for different traffic patterns. The cost of more energy consumption compared to GSP is slight.

Further work is required to address various properties of T-GSP. Firstly, we expect to find an adaptive traffic timer. A fixed timer doesn't fit all traffic patterns. Secondly, an adaptive gossip sleep probability after the traffic timer expires may better balance the cost of energy consumption and performance improvement. Of course, this probability should also consider the traffic pattern in the network. Last, T-GSP pays relatively more cost on the lifetime in a static network as shown in our simulations. Load balancing by rotating the active paths may improve the situation.

REFERENCES

[1] X. Hou and D. Tipper, "Gossip-based sleep protocol (GSP) for energy efficient routing in wireless ad hoc networks," in *Proceeding of IEEE WCNC*, 2004.

[2] C. E. Perkins and P. Bhagwat, "Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers," in *Proceeding Of ACM SIGCOMM*, Oct. 1994, pp. 234–244.

[3] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *ACM Balzer Mobile Networks and Applications Journal*, 1996.

[4] T. Clausen, P. Jacquet, J. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, "Optimized link state routing protocol," *IETF Internet Draft*.

[5] B. A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T. W. Chen, "Scalable routing strategies for ad hoc wireless networks," *IEEE Journal on Selected Areas of Communications*, vol. 17, no. 8, Aug. 1999.

[6] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications*, Apr. 1999.

[7] D. Johnson, "Routing in ad hoc networks of mobile hosts," in *Proc. IEEE Workshop on Mobile Computing Systems and Applications*, Dec. 1995.

[8] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing, T. Imielinski and H. Korth, eds.*, 1996.

[9] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[10] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proceeding of IEEE INFOCOM*, Apr. 1997.

[11] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *IEEE Hawaii International Conference on System Sciences*, 2000.

[12] A. Manjeshwar and D. P. Agrawal, "TEEN: A routing protocol for enhanced efficiency in wireless sensor networks," in *IEEE International Parallel Distributed Processing Symposium*, 2001.

[13] A. Manjeshwar and D. P. Agrawal, "APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *IEEE International Parallel Distributed Processing Symposium*, 2002.

[14] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wireless Networks*, no. 8, pp. 481–494, 2002.

[15] Y. Xu, J. Heildemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceeding Of ACM SIGMOBILE*, 2001.

[16] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Topology management for sensor networks: Exploiting latency and density," in *Proceeding Of ACM MOBIHOC*, 2002.

[17] R. Zheng and R. Kravets, "On-demand power management for ad hoc networks," in *Proceeding of IEEE INFOCOM*, 2003.

[18] R. Iyengar, K. Kar, and S. Banerjee, "Low-coordination topologies for redundancy in sensor networks," in *Proceeding Of ACM MOBIHOC*, 2005.

[19] X. Hou, D. Tipper, and S. Wu, "Traffic-aware gossip-based energy conservation for wireless ad hoc and sensor network routing," in *Proceeding of IEEE Consumer Communications and Networking Conference (CCNC)*, 2006.

[20] R. Meester and R. Roy, *Continuum Percolation*. Cambridge University Press, 1996.

[21] G. Grimmett, *Percolation*. Springer-Verlag, 1989.

[22] X. Hou, *Energy Conservation for Wireless Ad Hoc Routing*. Ph.D. Dissertation, University of Pittsburgh, 2006.

[23] K. Fall and K. Varadhan, *The ns Manual*. http://www-mash.cs.berkeley.edu/ns/, 2002.

[24] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *Proceeding of IEEE INFOCOM*, 2001.

[25] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," in *Proceeding of IEEE INFOCOM*, 2001.

[26] T. Chu and I. Nikolaidis, "Node density and connectivity properties of the random waypoint model," *Computer Communications*, no. 27, pp. 914–922, 2004.

Xiaobing Hou (xiaobing@mail.sis.pitt.edu) received the Ph.D. degree in Telecommunications from the University of Pittsburgh in 2006. His current research interests are routing and energy efficiency in wireless networks, IP and overlay multicast, network design, modeling and simulation.

David Tipper (dtipper@mail.sis.pitt.edu) is an Associate Professor in the Telecommunications Program with a secondary appointment in the Electrical Engineering Department at the University of Pittsburgh. His current research interests are network design and traffic restoration procedures for survivable networks, network control techniques and performance analysis.

Shuju Wu (swu@semo.edu) is an Assistant Professor in the Industrial and Engineering Technology Department at Southeast Missouri State University. She received the Ph.D. degree in Telecommunications from the University of Pittsburgh in 2004. Her current research interests include IP and overlay multicast, multicast in wireless environment, wireless ad hoc networks.