A Novel Routing Protocol that Guarantees Swiftness and Reliability for Wireless Sensor Network

D. Usha
Assistant Professor/CSE Dept
National Engineering College
K.R.Nagar, Kovilpatti
Tamil Nadu, India

B. Paramasivan, Ph.D
Professor and Head/CSE Dept
National Engineering College
K.R.Nagar, Kovilpatti
Tamil Nadu, India

ABSTRACT
Wireless sensor network has been used in many real-time applications that require reliable and timely delivery of data. This paper presents a real-time communication protocol with data recovery at each hop level to guarantee reliability and also uses the latency of link to determine the shortest path and promises speedy delivery for wireless sensor networks. The proposed protocol provides services such as route discovery based on the reaching speed of the node, multipath forwarding using required reliability and data recovery in single hop. Simulation result shows that the proposed protocol significantly improves the effective capacity of a sensor network in terms of reliability and quickness. Moreover the protocol is highly responsive to the various error conditions experienced and adaptive to large-scale dynamic sensor networks.

Keywords
Wireless Sensor Networks, Reliability, Quickness, Qos, Reaching Speed, Multipath forwarding, hop level recovery.

1. INTRODUCTION
Multihop wireless mesh networks are becoming a new attractive communication paradigm owing to their low cost and ease of deployment. Hence the routing protocols have a crisis over performance and reliability in wireless mesh networks. Traditional routing protocols send traffic along predetermined paths and face difficulties in coping with unreliable and unpredictable wireless medium. Different application may have different reliability requirements. Significant work is being done to identify reliable links using metrics such as received signal strength, link quality index which is based on "errors," and packet delivery ratio. Significant empirical evidence indicates that packet delivery ratio is the best metric.

For example, in temperature monitoring application the information can reach the destination tolerating certain percentage of loss. But a sudden fire, should be delivered with very high probability. Some sensory data may vary dynamically within short time limit and needs to be delivered within that time limit. Depending upon the importance and urgency, the data is set with different deadline. For example, a moving target has shorter deadline than that of a still target. In order to develop a network to support real-world applications, all of the characteristics such as quickness and reliability must be taken into account and systematically optimized to work together in harmony.

The traditional protocols are based on the end-to-end path discovery, resource reservation along the discovered path and path recovery in case of topological changes. Such approaches does not support sensor network because of the dynamic changes of the topology. Also the path discovery latency is not tolerable for time significant applications and it creates a huge overhead in large scale sensor networks.

The adhoc network is formed without any predetermined topology or shape and it is the responsiveness of the nodes to quickly adapt itself to changes in topology. To achieve high responsiveness, an adhoc network should issue and exchange more control packets, which will naturally result in less scalability and less reliability. The problem of achieving reliable transmission between remote nodes over multiple hops despite channel errors, collisions or congestion is very critical. To our knowledge there has been little or no work on the design of reliable transport protocols for sensor networks.

Hence the proposed protocol improves performance in two quality domains namely, quickness and reliability. The proposed protocol namely MMR provides speed options based on localized geographic forwarding using only immediate neighbor information. Here the path selection decisions are made at hop level based on the distance and the delay of the link. This increases the scalability and self adaptability of the network. As a result, packets can meet their end-to-end requirements with a high probability even if packet delivery decisions are made locally.

Existing protocols fails to recovery the lost data immediately and a longer delay is experienced. In the reliability domain, the protocol takes a different approach and supports a simple, robust and scalable transport that supports error recovery in the hop level itself rather cumulating it still the end. This simple approach with minimum requirements on the signaling reduces the communication cost for data reliability, and finally, responsive to high error rates allowing successful operation even under highly error-prone conditions.

Our goal is to provide guaranteed packet delivery services in both quickness and reliability domains while preserving the benefits of localized geographical routing.

2. RELATED WORK
In literature, several QoS provisioning protocols have been proposed for wireless ad hoc networks, which can be used for many mission critical applications. In these applications, reliable and timely delivery of sensory data plays a crucial role in the success of the mission. Specifically, the sensor network applications share the following characteristics such as diverse real-time requirements and diverse reliability requirements. Providing acceptable QoS for the traffics with the above characteristics is a challenging problem due to dynamic topology changes due to node mobility, failure and unreliable nature of wireless links.

To activate the nearest sensor node in a network and to find the position of the destination node, a data dissemination protocol called GEAR [1] (Geographical and Energy Aware Routing) uses a geographical and energy aware neighbor selection heuristic to route the packet towards the target region. When the packet arrives in the target region, its time to disseminate the packet within the region is calculated using the geographic forwarding algorithm. Ding et al [2] proposed a protocol for knowing the position of the nodes or relative position among them, especially to detect, monitor, and check on working attendance.

C.Lu et al [3] proposed a Real-Time Communication Architecture that provides service differentiation in the quickness domain by velocity-monotonic classification of packets. Based on packet's deadline and destination, its required velocity is calculated and its priority is determined in the velocity-monotonic order so that a high velocity packet can be delivered earlier than a low velocity one. It ensures a uniformly guaranteed network wide speed to meet end-to-end deadline of packet delivery. However, it is best-effort service differentiation without any guarantee in the end-to-end sense. M. Caccamo et al [4] proposed an Implicit Prioritized Access Protocol that provides hard real-time guarantee based on decentralized packet scheduling. However, it works only when most traffic is periodic and all periods are known a priori, which is not the case for many sensor network applications. Also, it is not adaptive to dynamics of sensor networks.

In [5] Pagani and Rossi suggested a protocol for reliable broadcast delivery in adhoc networks. Although the protocol is mainly suitable to broadcast almost to multicast group, it becomes inadequate for sparse groups. Moreover the protocol requires an underlying clustering protocol and hence it is inoperable in non-clustered networks. Furthermore the protocol is not scalable because it may switch to flooding and it uses Ack messages which are known to be expensive in wireless networks. In B. Deb et al [6], multiple paths from source to sink is used in diffusion routing framework to quickly recover from path failure. The multiple paths provided by such protocols could be used for sending the multiple copies of each packet. However it incurs extra overhead of multiple path formation and maintenance of path state in each node and is not adaptive to channel errors.

Q. Huang et al [7] aims at reliable and just-in-time delivery of alert packets to all sensor nodes in the moving delivery zone. This service is useful for waking up sensors ahead in the target trajectory being tracked. However, it assumes reliable and time-bounded transmission between every pair of sensor nodes and uses all nodes in a quite large forwarding zone to forward packets.

Directed diffusion [11] is one of a representative class of data dissemination mechanisms, specifically designed for a general class of applications in sensor networks. Directed diffusion provides robust dissemination through the use of multi-path data forwarding, but the correct reception of all data messages is not assured. It is necessary to provide a uniform delivery speed across the sensor network to meet the requirement of real-time applications such as disaster and emergency surveillance in sensor networks. Another necessary requirement of wireless sensor network is to deliver data reliably due to the dynamic and lossy nature of wireless transmissions. RMST (Reliable Multi-Segment Transport) [12], a new transport layer protocol provides guaranteed delivery and fragmentation / reassembly for applications that require them. RMST is a selective NACK-based protocol that can be configured for in-network caching and repair.

The proposed protocol uses the shortest path for routing by adopting the speedy link and for a less fast link, it takes multiple paths to meet out the reliability.

In case of error, the loss is recovered from the immediate neighbor itself quickly without any delay. As a result, packets can meet their end-to-end requirements with a high probability even if packet delivery decision is made locally. Also it uses multiple paths for sending the multiple copies of each packet. Our goal is to provide guaranteed packet delivery with quickness using the benefits of localized geographic routing.

3. PROPOSED PROTOCOL

In wireless sensor network, for mission critical applications reliable and timely delivery of sensory data plays a crucial role. The proposed protocol guarantees quickness with localized packet delivery, to a node that has required reaching speed without global topology information. It also provides reliable data transfer thereby minimizing the number of retransmissions for recovery with minimal signaling for data delivery. Hence the network achieves scalability to a very large extent, no setup or recovery latency and finally self adaptation to network dynamics.

For the localized packet routing, the protocol adopt Global Positioning System (GPS) [13] based on location awareness. The location of the neighbor nodes are determined using the control packets exchanged between immediate neighbors. This neighbor node information is used to make the routing decision, such that packets reach geographically towards their final destinations. Then each node relays the packet to a neighbor which is closer to the destination area.

3.1 Discrimination of Quickness

The challenging task for quickness is to deliver the sensed data within a time deadline [3]. The deadline for a packet is the maximum time within which the packet should reach its destination. The deadline of the sensed data depends upon the importance and dynamics of data. For example fire, flood, volcano or earthquake has shorter deadline than the data such as climatic conditions, monsoon etc. So whenever a data is sensed a time deadline is fixed for that data depending upon the drastic changes in environment and its effect over the environment. Therefore the minimum required speed level for a packet to reach its destination is calculated to meet the end-to-end deadline. The required speed is calculated using the following formula.

$$\mathrm{Re}\,q.Speed(x) = \frac{dis_{s,d}(x)}{deadline(x)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

The distance is from source to destination ($Dis_{s,d}(x)$.The Deadline is the maximum time limit for the packet to reach the destination. The node selects the proper neighbor with appropriate reaching speed along the virtual direct line from source to destination, for the packet to reach its destination as depicted in the Fig.1. The reaching speed of a node towards its destination is the geographical distance from the source to the destination divided by the delay between two neighbor nodes (queuing, processing, Mac collision resolution, packet loss percentage etc). The reaching speed is calculated using the following formula

$$\mathrm{Re}\,q.Speed_{i,j}^{k} = \frac{dis_{i,k} - dis_{j,k}}{delay_{i,j}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

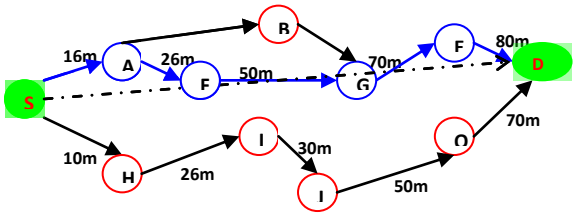

Fig.1: Path with appropriate reaching speed

A node with higher reaching speed than the required speed level is selected to forward the packet until the packet reaches its final destination. But in some congested area the node may not be able to find any node with higher reaching speed and that node are made to drop the packet to probabilistically reduce the workload. Also the node issues the "reverse packet" control message to reduce the incoming packet traffic in that node.

3.2 Updating of Required Speed

Suppose the intermediate nodes, due to congestion, face a longer delay and if the deadline is less, then a new required speed is calculated. This is done by comparing the expected latency and the remaining time to deadline. If the time needed to reach the destination is larger than the remaining time, then the required speed should be booted up to achieve the deadline. To handle this problem the elapsed time is calculated whenever the node receives the last bit of the packet and also when the node transmit the first bit to physical link. Using this the remaining time to deadline is also calculated. The elapsed time is calculated by using the following formula.

$$t^{elapsed} = t^{departure} + t^{transdelay} - t^{arrival} \quad \ldots\ldots\ldots\ldots\ldots (3)$$

$t^{departure}$ - time when the node transmits the first bit of a packet to the physical link

$t^{transDelay}$ - transmission delay of the packets

$t^{arrival}$ - the time when the node receives the last bit of the packet

Hence the remaining time to dead line is calculated using the elapsed time by the following formula

$$\mathrm{Re}\, m.time(x) = deadline(x) - t^{elapsed} - t^{propdelay} \quad \ldots\ldots\ldots\ldots\ldots (4)$$

Where $t^{propDelay}$ - the propagation delay between two nodes.

If any node notices that the current speed is insufficient to meet the remaining time to deadline then the speed level is boosted by the following formula

$$\mathrm{Re}\, q.Speed(x) = \frac{dis_{m,d}(x)}{\mathrm{Re}\, m.deadline(x)} \quad \ldots\ldots\ldots\ldots\ldots (5)$$

Where $dist_{m,d}(x)$ - distance between the intermediate node and the final destination node.

3.3 Discrimination in Reliability

The protocol also assures a certain level of reaching probability using another mechanism in the reliability domain. A non-shortest path is chosen as long as the packet reaches within deadline. This is accomplished by forwarding the packets to multiple paths based on dynamic reimbursement [9]. If a node forwards a packet to more than one neighbor node, then the combined reaching probability is determined and compared with the reaching speed. The probability that the packet reaches its final destination grows as the number of paths used to deliver the packet increases although there are some packet drops or failures.

The sensed data are given different reaching probability requirements based on the importance of data. Assume that the source s detects an event that needs to be reported to the destination d needs a reaching probability requirement of P^{reach} = 75%. Each node locally determines the multiple forwarding nodes based on an average packet loss percentage, packet drops or errors on the link. Using this packet loss probability the node estimates the local reaching probability to the next hop node. Local reaching probability is calculated by using the following formula

$$RP_{i,j}^{d} = (1 - e_{i,j})(1 - e_{i,j})^{[dis(j,d)/dis(i,j)]} \quad \ldots\ldots\ldots\ldots\ldots (6)$$

where $e_{i,j}$ - Packet loss percentage.

dist(j,d) / dist (i,j) - Hop count estimation from node j to final destination.

Let the source s find two immediate neighbor A and B with local reaching probability 60% and 65%. If more than one neighbor is chosen to forward the packet then the Total Reaching Probability (TRP) for the two paths can be calculated by using the following formula

$$TRP = 1 - (1 - TRP)(1 - RP_{i,j}^{d}) \quad \ldots\ldots\ldots\ldots\ldots (7)$$

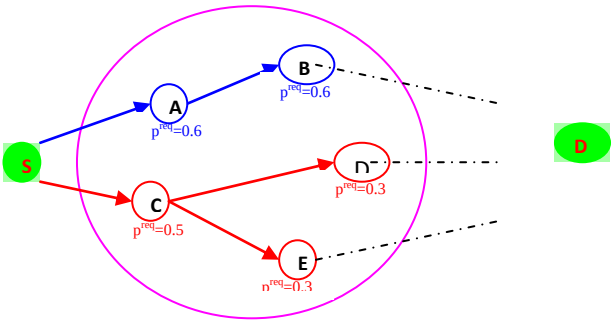

Fig.2: Multipath with required reaching probability

From the Fig.2 the source node has a TRP=0.88, which is higher than the reaching probability requirement. Therefore the packets are transferred to the two nodes A and B and the required reaching probability is split between these nodes as 0.6, 0.5 respectively so that the TRP is 0.8. Again these nodes A and B make local forwarding decisions such that A forwards to C with local reaching probability 0.9 which is higher than the required reaching probability, whereas B forwards to D and E with local reaching probability divided as 0.3 and 0.3 respectively for the two paths so that the TRP is 0.51 greater than the required reaching probability. In this way the node dynamically reimburse the previous wrong decisions as the packet travels to the final destination. If the receiving node cannot find a neighbor node to satisfy the reaching probability, it employs "reverse packet" backpressure mechanisms to reduce the reliability expectations of the previous nodes. With this probabilistic multipath forwarding, different reliable packets can be differentiated according to their reaching probability requirement and forwarded among different paths.

3.4 Error Recovery at Hop Level

Further to ensure the reliability and also to reduce the delay in recovering data, the MMR protocol proposes a hop by hop error recovery in which the intermediate nodes also take responsibility for loss detection and recovery. The protocol localizes loss to avoid the propagation of loss to downstream nodes and also ensures the timely dissemination of code segments. To enable this local loss recovery and in-sequence data delivery, an inject message is associated with all transmitted data. The inject message gives information such

as file identification, file length, sequence number and TTL (Time to Live). For the timely dissemination of the code segments, the packets are broadcasted by the nodes after every T_{min} until there are no packets for the node to send. The neighbor nodes that receive the packets check against their local data cache discarding any duplicates. Otherwise buffer the packet and decrease the TTL field in the header by 1. If there is no gap in the sequence number, then the packet is forwarded after every T_{min} to its neighbors. If there is any out of sequence in the packets, it switches to the demand mode where the node request for retransmission of the specific lost packet. The appeal is made quickly among the immediate neighboring nodes before next segment comes to it.

For a node with the packet error rate p, the chances of exchanging the message successfully across the single hop are (1-p). To maximize the probability of successful delivery of a packet, the "Controllable Time Frame"(CRT) enables the retransmission of a packet before the next packet arrives [10]. The probability of a successful delivery of a packet between two nodes with n retransmission is calculated using the following formula

$$\Omega(n) = (1-p) * p * \Omega(n) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (8)$$

The redundancy in dense deployed network is avoided by counting the number of times the same broadcast message is heard. If a message has been heard more than four times within T_{min} period, then the transmission is canceled for that node.

3.5 Single packet loss

The loss is made aware by using a NACK message which contains the loss window. The loss window represents the pair of sequence numbers denoting the left and right edge of lost packets. Each neighbor that receives the NACK message verifies its catch with the loss window to check for the missing data. If found the node relays the data immediately within a random time between 0 to T_r. ($T_r < T_{max}$). All the nodes that deliver packets maintain the data cache to retransmit in the event of data loss. Hence the probability that all neighboring node do not have the missing segments is very low and hence all data's are recovered at one hop itself. Very rarely the NACK message is propagated towards the source.

3.6 Busty Loss

Some time data loss is usually in bunches due to some channel impairments. Here all the bunch of data's may not be acquired from only one neighbor. Hence this lost window is broadcasted to all neighbors and each neighbor may send some missing data to the requested node. Consequently different segments of loss window are obtained from different neighbors at a random time period to avoid redundancy. If only a partial set of missing packets are recovered within $T_r < T_{max}$, the node resend NACK for every T_r period until packets are received.

3.7 Last packet Loss

The loss of packet is determined using only the out of sequence number. But this cannot be identified, if the last packet is lost and is not recovered. To avoid this problem, for every T_{pro} the node sends a NACK message with a loss window whose left edge is equal to (S_{last} + 1) and right edge is equal to S_{max}, when no new packet is received within that time. Time period T_{pro} is proportional to the difference between last highest sequence number (S_{last}) and largest sequence number (S_{max}). The T_{pro} can be calculated by using the following formula

$$T_{pro} = a * (S_{\max} - S_{last}) * T_{\max} (a \geq 1) \quad \ldots\ldots\ldots\ldots (9)$$

Where α - Scaling factor to adjust the delay.

T_{pro} guarantee a node to wait long enough until all upstream nodes receives all segments before moving to demand mode.

4. SIMULATION AND RESULTS

This section presents the performance results of the designed MMR protocol compared with various other existing protocols such as RMST, and PSFQ, obtained through NS2 simulation. The results are measured in terms of delivery delay and routing overhead. To evaluate the performance of the MMR protocol in a realistic scenario, the sensor nodes are deployed randomly. Nodes use radios with 2 mbps bandwidth with a nominal radio range of 20 m. The channel access is the simple Carrier Senses Multiple Access/Collision Avoidance (CSMA/CA), and a uniformly distributed channel error model is used. A user node attempts to inject a program image file of a size equal to 2.5 kb to another node. The packet size is 512 bytes. Packets are generated from the user node and transmitted at the rate of one packet every 50 ms.

Fig.3 shows the average delivery delay of the protocols used for comparison with the MMR. The proposed MMR protocol has lower delay compared to other protocols in various network sizes. When the network size is below 200 nodes, then the delivery delay of MMR is low and when the network is above 350, it is found to be constant.

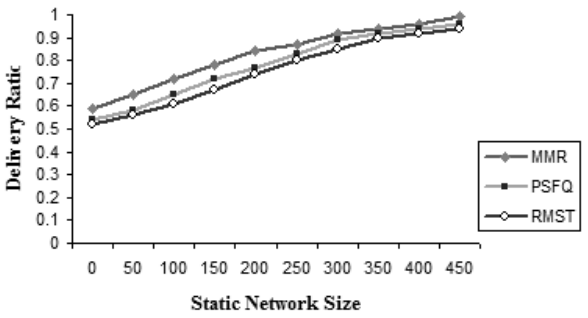

Fig.3: Delivery delay of various protocols

For different number of active connections, the overheads of the MMR protocol are shown in Fig.6. As can been seen, RMST and PSFQ exhibits the highest routing overhead. This is an indication that maintenance operation is very expensive. The overhead to RTS, CTS, DATA and Ack for 802.11 is 20, 14, 34 and 14 whereas in MMR it is 26, 14, 34 and 16. Hence generates smallest routing overhead compared to other protocols.

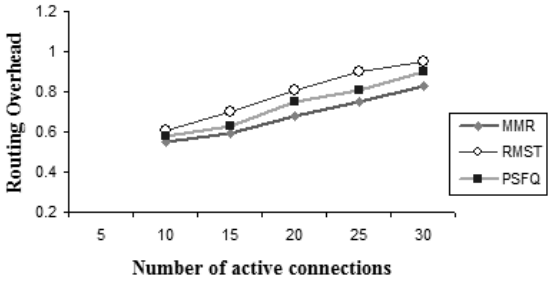

Fig. 4: Routing overhead

5. CONCLUSION

This paper presents a real time communication protocol namely MMR for wireless sensor networks to provide quickness and reliability for the network. For the quickness domain, the protocol provides a required speed level for the nodes at individual hop level depending upon the importance of the data. Hence important data reaches the destination within short time span than other normal data's. For the reliability domain, it uses proactive multipath forwarding to control the number of packet delivery paths depending on the required end-to-end reaching probability. All these methods are implemented in the hop level and thus increase the scalability of the network. Also the proposed protocol focuses on error recovery which identifies packet loss at each hop level and rectifies it so that the protocol also operates under high error rate conditions. The results of the simulations are used to verify the robustness and effectiveness of this routing protocol and it looks very promising in an actual wireless test bed. The proposed work is executed in a homogenous network. In future, the work can be extended to heterogeneous network where every node has different transmission radius. Also the mobility of the node can be considered.

6. REFERENCES

[1] Yan Yu, Ramesh Govindan, and Deborah Estrin "GEAR: Geographical and Energy Aware Routing: A recursive data dissemination protocol for wireless sensor networks". 2001.

[2] M.Ding, D.Chen, K.Xing and X.Cheng, "localized Fault-Tolerant Event Boundary Detection in Sensor Network" in Proceedings of IEEE INFOCOM, 2005.

[3] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks," Proc. IEEE Int'l Conf. Distributed Computing Systems, pp. 46-55, 2003.

[4] M. Caccamo, L. Zhang, L. Sha, and G. Buttazzo, "An Implicit Prioritized Access Protocol for Wireless Sensor Networks," Proc. IEEE Real-Time Systems Symp. (RTSS '02), pp. 39-48, 2002.

[5] E.Pagani and G.P.Rossi ,"Reliable Broadcast in Mobile Multihop Packet Networks" Proceedings of ACM/IEEE MOBICOM97, Budapset, Hungary, Sept.1997, pp34-42.

[6] B. Deb, S. Bhatnagar, and B. Nath, "ReInForm: Reliable Information Forwarding Using Multiple Paths in Sensor Networks," Proc. IEEE Int'l Conf. Local Computer Networks, pp. 406-415, 2003.

[7] B. Hughes and V. Cahill, "Achieving Real-Time Guarantees in Mobile Ad Hoc Wireless Networks," Proc. Work-in-Reaching Session 24th IEEE Real-Time Systems Symp., Dec. 2003.

[8] S.-J. Lee, M. Gerla, C.-C. Chiang, "On-demand multicast routing protocol", Proc. IEEE Wireless Communications and Networking Conf., pp. 1298-1304, Sept. 21-25, 1999.

[9] Emad Felemban, Chang-Gun Lee, and Eylem Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Quickness in Wireless Sensor Networks," Proc. IEEE Transactions on mobile computing, Vol.5, No. 6, June 2006.

[10] Chieh-Yih Wan, Andrew T. Campbell, and Lakshman Krishnamurthy, "PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks", September 28, 2002, Atlanta, Georgia, USA. Copyright 2002 ACM 1-58113-589-0/02/0009.

[11] C. Intanagonwiwat, RC. Govindan, D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", Proc. of the Sixth Annual ACM International Conference on Mobile Computing and Networking, pp. 56-67, Aug. 2000.

[12] F. Stann and J. Heidemann, "RMST: Reliable Data Transport in Sensor Networks," Proceedings of IEEE International Workshop on Sensor Network Protocols and Applications, pp. 102–112, 2003.

[13] B. Karp and H. Kung, "Greedy Perimeter Stateless Routing for Wireless Networks," Proceedings of IEEE/ACM International Conference on Mobile Computing and Networking, pp. 243–254, 2000.