

Wireless Mesh Networking Protocol for Sustained Throughput in Edge Computing

Josiah Burchard*, Dmitrii Chemodanov[†], John Gillis[†], and Prasad Calyam[†]

*Department of Computer Science, Southeast Missouri State University, Email: josiah@jburchard.com

[†]University of Missouri, Email: {calyam, gillisj}@missouri.edu, dycbt4@mail.missouri.edu

Abstract—It is critical to provide sustained data throughput in edge computing, where several sensor devices generate information that needs to be fused and used for decision making in e.g., disaster incident scenes. To this end, we compare the effectiveness of two protocols, Hybrid Wireless Mesh Protocol (HWMP) and Greedy Perimeter Stateless Routing (GPSR), based upon their ability to stream data in a mesh network. We model the two protocols using three topologies consisting of a sender, receiver and multiple Mesh Points to relay the data. We perform experiments varying the density, scale and failure rates of the topology. Finally, we evaluate the effectiveness of both protocols by comparing the total throughput from sender to receiver in each experiment. We show that geographic routing algorithms such as GPSR, given their potential for statelessness, can be a more effective protocol than the 802.11s standard HWMP in high failure and large scale topologies.

Index Terms—Mesh network; hybrid wireless mesh protocol; geographic routing; greedy perimeter stateless routing; edge computing; sustained throughput; internet of things

I. INTRODUCTION

Mesh networking, consisting of a sender, receiver and multiple mesh points has recently become an important area of study with the advent of the Internet of Things (IoT), which requires an edge computing paradigm. IoT requires high, sustained throughput across large areas that may not have wireless or even wired connectivity. For example, disaster triaging involving first responders combing through a disaster scene with IoT devices to stream data to an incident commander for analysis. A mesh network can quickly provide a wireless connection over a large area to facilitate the data flow between the edge, gateway and the cloud as shown in Fig. 1.

In a mesh network where a sender is out of range of the receiver, the sender will route packets through intermediate mesh points until it reaches its destination. Routing decisions throughout these mesh points are gravely important when considering the efficiency and viability of a mesh network. High, sustained throughput is necessary for demanding applications such as live video streaming and mission critical data analysis.

The industry standard for mesh networks is defined by 802.11s as the Hybrid Wireless Mesh Protocol (HWMP)[9].

This material is based upon work supported by the National Science Foundation under Award Number: CNS-1359125 and Coulter Foundation. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or Coulter Foundation.

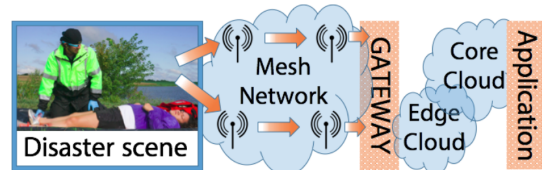


Fig. 1. Edge to Gateway to Cloud to App

HWMP learns the network topology in real time to make routing decisions. While this allows for an easily configured network, it requires overhead to learn the current state of the network. This will congest the network and degrade the overall throughput. This effect is compounded each time the network topology changes, and is magnified as the network scales. Additionally, mesh networks often have a high rate of failure as devices are added/removed, moved out of range or malfunctioning. Hence, topology learning is not an efficient solution to make routing decisions in a mesh network.

Geographic routing is a technique that uses the knowledge of the mesh points' locations for routing decisions. Geographic routing has the potential to implement a 'Stateless' protocol that eliminates overhead by avoiding learning the network topology[10]. Stateless in this context is meant to include only the minimum necessary state required to make any forwarding decisions, which only includes immediate one-hop neighbors of a mesh point. There are many geographic routing protocols suggested in literature ([3], [13], [4]), including Greedy Perimeter Stateless Routing [10] (GPSR). GPSR is a well known implementation of geographic routing and has been selected for study in this work based upon its simplicity and establishment as a classical example in mesh networking literature. GPSR provides a higher, sustained throughput than HWMP because of its fault tolerance and low overhead provided by stateless routing.

In this paper, we compare the effectiveness of HWMP and GPSR protocols and compare their ability to stream data in a mesh network at sustained throughput levels in edge computing scenarios involving e.g., disaster incidents. We model the two protocols using three topologies consisting of a sender, receiver and multiple Mesh Points to relay the data. We perform experiments varying the density, scale and failure rates of the topology. Finally, we evaluate the effectiveness of both protocols by comparing the total throughput from sender to receiver in each experiment.

The remainder of the paper is organized as follows: Section II provides an overview of the HWMP and GPSR protocols. Section III lists related works that have compared these two protocols. Section IV describes our experiment setup. Section V describes our experiment results along with a detailed discussion of the findings. Section VI concludes the paper.

II. GEOGRAPHICAL ROUTING PROTOCOLS

A. HWMP

HWMP uses a blend of reactive and proactive methods to perform path selection with updated metrics on the current topology. A proactive tree-based routing scheme is implemented as well as an adaptation of Ad hoc On Demand Distance Vector (AODV) using up to date metrics on the network [9], [2].

The proactive portion of HWMP uses control messages to send throughout the network to build a tree-based location table. These control messages all have a time to live (TTL) field, a destination sequence number (DSN) and a metric field. The metric is specifically implemented as Airtime Link and gathers information about the quality of the route to later use in path selection.

To construct the location table, a Path Request (PREQ) message containing a destination MAC address is first broadcasted through the network to each available mesh point. As each mesh point receives the PREQ, it checks the DSN and metric fields and updates its reverse path information if the path of the PREQ has a greater DSN or if it has a better metric. Then it will construct and append new metric data into the PREQ considering the link between the previous mesh point and itself and rebroadcast the message for other mesh points to receive. When the destination receives the PREQ, it will unicast a Path Reply (PREP) message back towards the sender using the path provided by the PREQ metrics. Each intermediate mesh point will update its forward path (between the mesh point and the sender of the PREP) with newly created metrics and forward the message along. When the PREP message reaches the destination, there is now a valid route with up to date metrics from the source to the destination. This process can be repeated until there are multiple paths between the source and destination.

The reactive portion of HWMP uses Radio Metric Ad hoc On Demand Distance Vector (RM-AODV). This is an extension of AODV which makes use of Airtime Link metric for path selection. When the sender begins to send a message, it will check its location table for the possible paths to the destination then use RM-AODV to select the best path to the destination. The path selection always happens at transmission time to make use of the most up-to-date metrics.

B. GPSR

GPSR implements a greedy forwarding algorithm in which a mesh point will always forward packets to the node that is closest to the destination and within range of the forwarding mesh point. This technique provides a straightforward method of efficient path guessing from source to destination. By

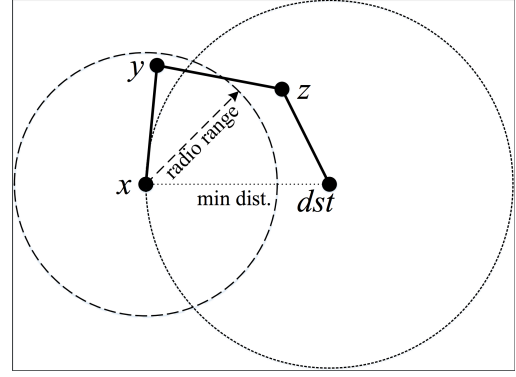


Fig. 2. Local Minimum Problem. X is a local minimum with distance to D, however a valid path exists through w or y.

utilizing greedy forwarding, each mesh point only needs to know the state of its current one-hop neighbors. Additionally, each mesh point only requires this information at transmission time. This implies no overhead in the network since control messages do not need to be transmitted throughout the network. By using a purely reactive method of path selection, forwarding information is always fresh. This makes greedy forwarding extremely fault tolerant since a failure in any mesh point does not increase any overhead on the network [10].

There are classic examples of where greedy forwarding fails. The Local Minimum problem shown in Fig. 2 describes a situation where a path exists from a mesh point to a destination, but the next mesh point in the path sequence is farther away than the current mesh point. When there is no possible path in greedy forwarding mode, GPSR switches to recovery mode which uses perimeter routing.

Perimeter routing instructs packet forwarding to route around the void following the right hand rule. Perimeter mode continues operation until the packet is closer to the destination than when perimeter mode had been started. At this point greedy forwarding mode continues.

III. RELATED WORK

Many recent works have studied performance comparisons between popular routing protocols such as HWMP [1][12][15][7][17], GPSR [14][16][18] and many others. These studies generally focus on end-to-end throughput and latency as their performance metrics.

Many applications such as live streaming and real time applications depend on sustained throughput. A large variance in throughput, even when resulting in overall higher throughput, is undesirable in these applications. This work focuses on two mesh network protocols, GPSR and HWMP, in their ability to support high, sustained throughput given high scale and failures[10].

IV. EXPERIMENT SETUP

A. Simulation Parameters

Our simulation was implemented using the industry standard Network Simulator 3 (NS-3). HWMP implementation



Fig. 3. Joplin Area Mobility Scene

was provided by the core NS-3 library and our GPSR implementation was based upon [6]. Each experiment contained a sender, receiver and multiple mesh points. The sender streamed 5 Mbps of data to the receiver from a one second starting delay to the end of the experiment. Each node has an approximate range of 250 m, but the number of neighbors to each node depends on the trial. A full list of the simulation parameters is shown in Table I.

TABLE I
GLOBAL SIMULATION PARAMETERS

Data rate	5 Mbps
Packet Size	512 b
Frequency	2.4 Ghz
Radio Type	802.11g
Receiver Moving Speed	2.8 m/s
Simulation Time	400 s

1) *Mobility Scenes*: Three mobility scenes were selected for experimentation. First is a full grid, with $N \times M$ nodes spaced evenly apart. The sender was positioned within the grid in the top left corner. The receiver originated outside the right edge of the grid, moved at a slanted angle into the grid, stopped for 60 seconds, then continued through the bottom edge outside the grid and remained there until the end of the simulation. In this scene, there are several paths to route data through the mesh and the shortest path consequently changes throughout the experiment.

The second scene is entitled Joplin, MO disaster area as shown in Fig. 3, modeled after a city block that had been destroyed by a tornado in 2011 and consequently searched for survivors. This scene contains mesh points lining the perimeter of a near square city block. The sender and receiver move in a similar fashion to the full grid scene.

The third scene is a straight line. The sender is positioned on the left, then each mesh point consecutively and ending with the receiver on the right end. This experiment was included to compare throughput when path selection and topology changes are minimal.

2) *Scale and Failure Rate Levels*: Six scale and failure experiments were performed with each protocol and mobility scene set for a total of $2 * 3 * 2 * 6 = 72$ experiments. The scale experiments had a zero percent failure rate and an increasing number of mesh points from four to twenty. The failure experiments had nine mesh points and increased in the probability of failure from 10-60%. To implement failure in the experiment, every 30 seconds $N\%$ of mesh points are moved

to an isolated area very far away from the experiment and are returned after 30 seconds.

Each experiment described above was performed 50 times with varying topology densities up to twice as dense. For example, the spacing between mesh points in the full grid mobility scene is varied between 50 and 100 m.

B. Case Study: Panacea's Cloud

A direct application of this research is to select a mesh protocol for high, sustained throughput for the Panacea's Cloud platform [8]. Panacea's Cloud is a disaster triage management system consisting of various IoT edge devices, an offline mesh network and a server with intelligent edge computing. Mesh devices will move about the scene while collecting and transmitting data which will change the network topology at a human timescale. The mesh network will have to accommodate these changes in topology and provide high, sustained throughput. The data transmitted from the edge devices often consist of priority data regarding the disaster scene that requires immediate processing by the server. Another application could simply consist of a live video stream that should be viewed smoothly by another edge device or the server application itself.

C. Selected Mobility Scenes

The Full Grid and Joplin Area mobility scenes were selected because of their relevance to the Panacea's Cloud platform. These mobility scenes represent common topologies that would be used by Panacea's Cloud in disaster triaging. The Straight Line mobility scene was selected purely for the experimental purposes of comparing sustainable throughput between protocols as this is not a practical topology for disaster triage management.

V. SIMULATION RESULTS

A. Results

In Full Grid and Joplin Area mobility scenes, GPSR presented significantly higher throughput in all experiments as shown in Fig. 4. As the scale increased, GPSR throughput slowly declined while HWMP throughput rapidly decreased. In all mobility scenes, GPSR shows sustained throughput with low standard deviation while HWMP presents highly fluctuating throughput as shown through standard deviation plots.

Additionally as shown in Fig. 5, as the mesh points scale beyond 16 in Joplin Area and 20 in Full Grid, HWMP throughput degrades to near zero Mbps. Further, as shown in Fig. 6, GPSR scales to this size with only minimal degradation.

B. Discussion

While these results plainly show that GPSR presents a better solution for mesh routing in simulation, it is not necessarily a practical candidate in real environments. GPSR relies on the unit graph assumption, detailing that each mesh point is always connected to every other mesh point within its fixed radio range. In practice, even in static environments, radios frequently violate this rule as small conditions change in the

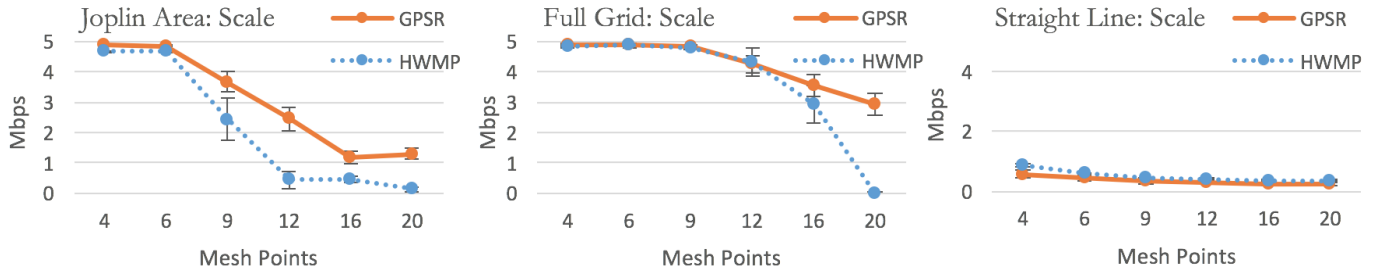


Fig. 4. Mean throughput results over increasing number of mesh points. In Joplin Area and Full Grid where there are multiple possible routing decisions, GPSR shows higher throughput

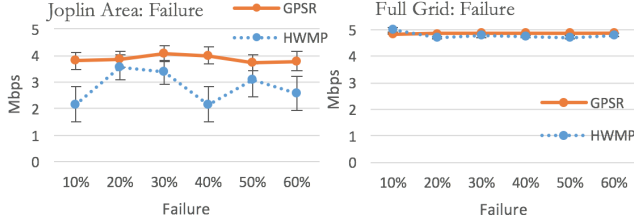


Fig. 5. In Joplin Area and Full Grid, we see that GPSR performs better or near equivalent to HWMP

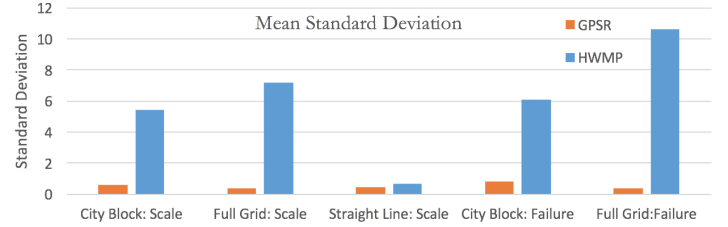


Fig. 6. In every experiment, GPSR presents more sustained throughput than HWMP as shown by a smaller mean standard deviation

environment [11]. There are suggestions to combat the reality of dynamic radio range such as Cross-Link Detection Protocol in [11], however this will add small additional overhead to GPSR that was not accounted for in this simulation.

A larger burden in the physical implementation of geographic routing is that each mesh point relies on accurate location information to make forwarding decisions. This location could at best increase the price per mesh point and at worse be completely unavailable. In environments where GPS is not viable (e.g. under ground, inside large buildings) and no other location information can be obtained, GPSR or any other geographic routing solution is not available. While GPSR only shows marginally higher throughput at small scale (below nine mesh points), it presents a more sustained throughput in every experiment. However, note that this is not important for all applications. While this is a priority for many real-time applications such as Panacea's Cloud, it is not required for others such as file transfers and low-priority web requests.

VI. CONCLUSION

We have shown GPSR with the benefit of stateless routing can achieve higher and more sustainable throughput than HWMP. We have also shown that GPSR scales with less degradation than HWMP with topologies greater than about 16 mesh points.

In the case of Panacea's Cloud, sustained throughput is very important to the application, location information provided by IoT devices give location information to the mesh points at little cost and the network scale must increase far beyond 16 mesh points. Therefore these results show that GPSR is a practical, optimal protocol for high, sustained throughput in edge computing.

A part of our future work, we want to verify our simulation results through implementation of GPSR on physical routers and repeat these experiments in the field. OpenWRT is an open source Linux distribution designed as a router operating system [5]. A good starting place would be to add geographic capabilities to a router running OpenWRT and implement GPSR as a new routing package on the device.

REFERENCES

- [1] SMS Bari, Farhat Anwar, and MH Masud. Performance study of hybrid wireless mesh protocol (hwmp) for ieee 802.11 s wlan mesh networks. In *Computer and Communication Engineering (ICCCE), 2012 International Conference on*, pages 712–716. IEEE, 2012.
- [2] Jalel Ben-Othman, Lynda Mokdad, and Mohamed Ould Cheikh. Q-hwmp: improving end-to-end qos for 802.11 s based mesh networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6. IEEE, 2010.
- [3] Prosenjit Bose, Pat Morin, Ivan Stojmenović, and Jorge Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless networks*, 7(6):609–616, 2001.
- [4] Andrej Cvetkovski and Mark Crovella. Hyperbolic embedding and routing for dynamic graphs. In *INFOCOM 2009, IEEE*, pages 1647–1655. IEEE, 2009.
- [5] Florian Fainelli. The openwrt embedded development framework. In *Proceedings of the Free and Open Source Software Developers European Meeting*, 2008.
- [6] António Fonseca, André Camões, and Teresa Vazão. Geographical routing implementation in ns3. In *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques*, pages 353–358. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2012.
- [7] Sana Ghannay, Sonia Mettali Gammar, and Farouk Kamoun. Comparison of proposed path selection protocols for ieee 802.11 s wlan mesh networks. In *Wireless and Mobile Networking*, pages 17–28. Springer, 2008.
- [8] John Gillis, Prasad Calyam, Ashley Bartels, Mihai Popescu, Stephen Barnes, Jennifer Doty, Dena Higbee, and Salman Ahmad. Panacea's glass: Mobile cloud framework for communication in mass casualty disaster triage. In *Mobile Cloud Computing, Services, and Engineering*

(MobileCloud), 2015 3rd IEEE International Conference on, pages 128–134. IEEE, 2015.

- [9] Guido R Hiertz, Dee Denteneer, Sebastian Max, Rakesh Taori, Javier Cardona, Lars Berlemann, and Bernhard Walke. Ieee 802.11 s: the wlan mesh standard. *IEEE Wireless Communications*, 17(1):104–111, 2010.
- [10] Brad Karp and Hsiang-Tsung Kung. Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM, 2000.
- [11] Young-Jin Kim, Ramesh Govindan, Brad Karp, and Scott Shenker. Geographic routing made practical. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, pages 217–230. USENIX Association, 2005.
- [12] Adlen Ksentini and Obaid Abassi. A comparison of voip performance over three routing protocols for ieee 802.11 s-based wireless mesh networks (wlan mesh). In *Proceedings of the 6th ACM international symposium on Mobility management and wireless access*, pages 147–150. ACM, 2008.
- [13] Martin Mauve, Holger Füßler, Jörg Widmer, and Thomas Lang. Position-based multicast routing for mobile ad-hoc networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3):53–55, 2003.
- [14] Valery Naumov, Rainer Baumann, and Thomas Gross. An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pages 108–119. ACM, 2006.
- [15] Jerry Chun-Ping Wang, Brett Hagelstein, and Mehran Abolhasan. Experimental evaluation of ieee 802.11 s path selection protocols in a mesh testbed. In *Signal Processing and Communication Systems (ICSPCS), 2010 4th International Conference on*, pages 1–3. IEEE, 2010.
- [16] Wenjing Wang, Fei Xie, and Mainak Chatterjee. An integrated study on mobility models and scalable routing protocols in vanets. In *2007 Mobile Networking for Vehicular Environments*, pages 97–102. IEEE, 2007.
- [17] Azila Zakaria, Hafizal Mohamad, Nordin Ramli, and Mahamod Ismail. Performance evaluation of routing protocols in wireless mesh network. In *Advanced Communication Technology (ICACT), 2013 15th International Conference on*, pages 1111–1115. IEEE, 2013.
- [18] Mohamed Adnene Zayene, Nabil Tabbane, and Refaat Elidoudi. Performance evaluation of greedy perimeter stateless routing protocol in ad hoc networks. In *Computer Sciences and Convergence Information Technology, 2009. ICCIT'09. Fourth International Conference on*, pages 907–912. IEEE, 2009.