Mesh networking optimized for robotic teleoperation

Abraham Hart, Narek Pezeshkian, Hoa Nguyen
Unmanned Systems Branch, Code 71713
Space and Naval Warfare Systems Center San Diego, CA 92152

ABSTRACT

Mesh networks for robot teleoperation pose different challenges than those associated with traditional mesh networks. Unmanned ground vehicles (UGVs) are mobile and operate in constantly changing and uncontrollable environments. Building a mesh network to work well under these harsh conditions presents a unique challenge. The Manually Deployed Communication Relay (MDCR) mesh networking system extends the range of and provides non-line-of-sight (NLOS) communications for tactical and explosive ordnance disposal (EOD) robots currently in theater. It supports multiple mesh nodes, robots acting as nodes, and works with all Internet Protocol (IP)-based robotic systems. Under MDCR, the performance of different routing protocols and route selection metrics were compared resulting in a modified version of the Babel mesh networking protocol. This paper discusses this and other topics encountered during development and testing of the MDCR system.

Keywords: mesh, networking, Babel, robot, teleoperation, VPN, communication, relay, ad hoc, routing protocol.

1. BACKGROUND

The Manually Deployed Communication Relay (MDCR) project started with a need to field a system that extends robotic teleoperation range in non-line-of-sight (NLOS) operating environments. Previous work with the Automatically Deployed Communication Relay (ADCR)[1] system dealt with relay-deployment techniques and highlighted the amount of work still needed to have a reliable, quality mesh network for robotic teleoperation.

Research in mobile mesh networks is still a relatively young field. Yang, Wang, and Kravets present an analysis of different mesh protocols and associated metrics[2]. In their research, they present four requirements for protocols with good mesh network performance: route stability, good performance for minimum-weight paths, efficient algorithms to calculate minimum-weight paths, and loop-free routing. Three promising open-source mesh implementations that meet these requirements were identified: the Optimized Link-State Routing Protocol (OLSR)[3], the Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.)[4], and Babel[5]. Abolhasan, Hagelstein, and Wang performed tests pitting OLSR, B.A.T.M.A.N., and Babel against each other in a controlled environment[6]. They conclude that in small mesh networks, Babel has higher throughput but, due to the slow convergence times of all three tested networks, none may be suitable for mobile meshes. Murray, Dixon, and Koziniec suggest that B.A.T.M.A.N. and OLSR have similar performance characteristics and state that Babel has higher throughput in smaller mesh networks[7]. Because of this similarity, and an incomplete implementation at the time of testing, B.A.T.M.A.N. was not considered for testing. The purpose for conducting the following assessment was to quantify the performance between OLSR and Babel on a fielded robotic system. Due to time constraints Babel and OLSR, which had existing implementations (*babeld* and *OLSRD)* pre-packaged for the development platform (*OpenWRT*), were used.

Preliminary testing of *OLSRD* showed severely degraded video quality and intermittent control between the remote controlled vehicle (RCV) and operator control unit (OCU) if the routing path more than two radios. As a result, teleoperation was practically unusable in all situations. The test with *babeld* fared slightly better. Good video quality and control was observed over a single hop when given a few minutes for route stabilization and convergence to take place. Over multiple hops and route changing events, however, video quality degraded and control became intermittent. The

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **APR 2012**	2. REPORT TYPE	3. DATES COVERED **00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE **Mesh networking optimized for robotic teleoperation**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Space and Naval Warfare Systems Center Pacific,Unmanned Systems Branch, Code 71713,San Diego,CA,92152**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Mesh networks for robot teleoperation pose different challenges than those associated with traditional mesh networks. Unmanned ground vehicles (UGVs) are mobile and operate in constantly changing and uncontrollable environments. Building a mesh network to work well under these harsh conditions presents a unique challenge. The Manually Deployed Communication Relay (MDCR) mesh networking system extends the range of and provides non-line-of-sight (NLOS) communications for tactical and explosive ordnance disposal (EOD) robots currently in theater. It supports multiple mesh nodes, robots acting as nodes, and works with all Internet Protocol (IP)-based robotic systems. Under MDCR, the performance of different routing protocols and route selection metrics were compared resulting in a modified version of the Babel mesh networking protocol. This paper discusses this and other topics encountered during development and testing of the MDCR system.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**7**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

testing showed that modifications to the route selection scheme, the culprit for the degraded performance, was required for both *babeld* and *OLSRD*. Because *babeld* employed a code-base that facilitated implementation of various route selection schemes, and because it performed better than *OLSRD*, it was chosen as the network protocol for MDCR.

This paper consists of two main parts. The first part (sections 2 and 3) describes the requirements and the tests used to identify weak areas of the mesh network. The second part (sections 4 and 5) describes the modifications to *babeld*.

2. REQUIREMENTS AND TESTING METHODOLOGY

The video stream is the primary means of feedback to the operator and potentially the most important aspect of robotic teleoperation. The purpose of this project was to make modifications to the Babel routing protocol to minimize video glitches and artifacts with some tolerance for minor interruptions for short periods of time. Access to quantitative data was not readily available with the proprietary RCV platforms used for testing, so a qualitative method to evaluate the networks performance was devised. The operator performed several test runs observing RCV video and control while noting signs of degradation. Signs include, in order of increasing severity, pixelated video, smearing video, choppy video, intermittent and delayed control of the vehicle. Pixelated video is the appearance of artifacts localized to a small section of video output. Smearing video is the appearance of a smeared ghosting image that that affects the entirety of video output. Choppy video is when the video stream stalls for a few seconds, normally followed by smearing video. Intermittent control of the RCV is when the video and vehicle movement stutter. Delayed vehicle control is when RCV movement occurs several seconds after a command has been given.

3. TESTING

Several test scenarios were constructed to test modifications made to *babeld* and qualitatively judge the effect of these modifications on the performance of the network. Test 1 consisted of up to six mesh nodes setting on a table. Test 2 through 7, illustrated in Figures 1 through 6, were conducted using an OCU and RCV. Dotted lines show possible routes between mesh nodes. The dashed lines show the path the RCV takes relative to the mesh nodes. The solid vertical lines separate the test area into regions where the RCV can only communicate with mesh nodes within the same or adjacent regions.

Route selection and multi-hop routes were determined to cause degraded video quality, therefore the test scenarios must force these conditions upon the network to allow proper comparison between various modifications made to *babeld*. Two different methods were used to create route paths. For line-of-sight (LOS) tests the maximum effective communications range from the RCV to the OCU was determined, then a mesh node was placed in a way that driving the RCV beyond that point would ensure the creation of a multi-hop route. For the NLOS tests, building corners and other large obstacles were used to create the desired multi-hop route. In most cases, both methods showed similar route selection performance for the same testing situation. The method chosen was based on each method's feasibility given the environment.

Test 1 consisted of between two to eight mesh nodes within close proximity of each other. Firewall rules were used to block communication between two or more mesh nodes, thus simulating a route-changing event. Two mesh nodes were chosen to be the “end-point” nodes, simulating the RCV and OCU. If N is the total number of mesh nodes along the route, the number of hops equals N-1. Route convergence was determined by watching for changes in the mesh node routing tables at an interval of 1 second. All radios were within interference range of each other. A throughput measuring program was used to gather information on network quality under different routing conditions.

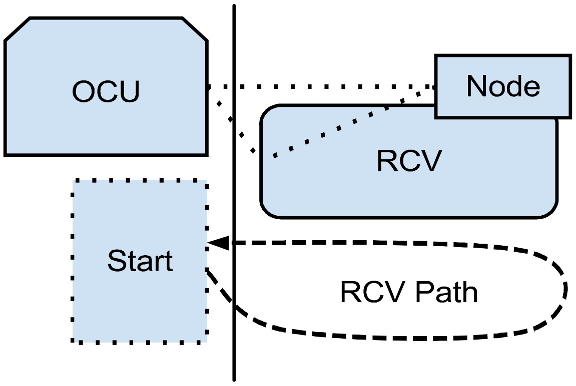

Figure 1. Test 2: a single-node mounted on robot. This tests that a node carried by the vehicle does not deteriorate teleoperation performance.

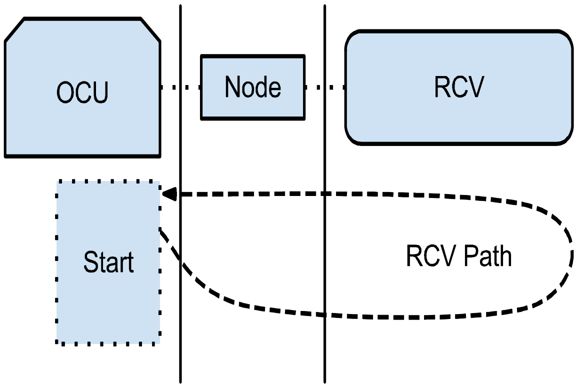

Figure 2. Test 3: a single hop. This tests the quality of a single route change event.

Test 2, diagrammed in Figure 1, consisted of a single mesh node mounted on the RCV. The position where communication began to weaken was then compared to a baseline test run without a mesh node. This test ensured that a mesh node carried by the RCV did not deteriorate teleoperation performance.

Test 3, diagrammed in Figure 2, consisted of a single mesh node placed using the NLOS method. The goal was to observe the quality of a route change from a direct link to a single hop.

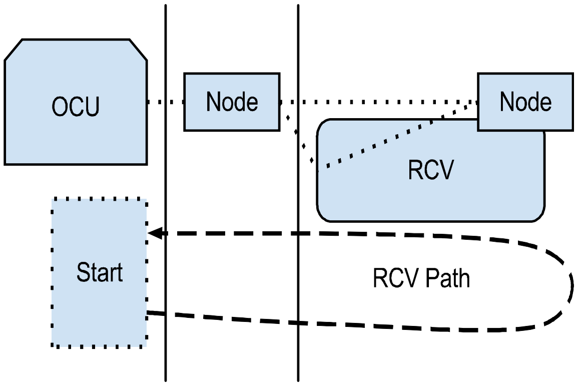

Figure 3. Test 4: two nodes, one mounted on RCV. This tests ensures that a node, carried by the vehicle, does not deteriorate performance on a route change event.

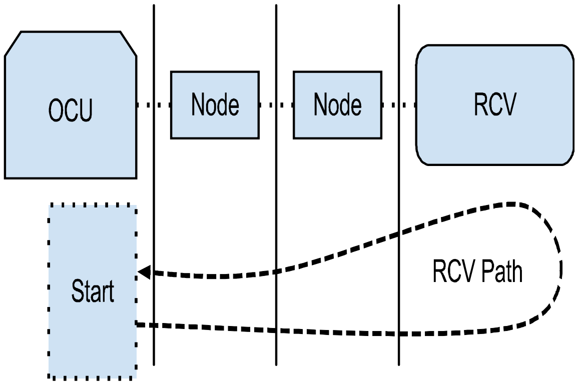

Figure 4. Test 5: two nodes in-line. This tests route changes from a direct link to a single hop to a double hop.

Test 4, diagrammed in Figure 3, consisted of a two mesh nodes, one placed using the NLOS method and the other mesh node mounted on the RCV. The goal was to observe the quality of a route change from a direct link to a single hop with a mesh node mounted on the RCV.

Test 5, diagrammed in Figure 4, consisted of two mesh nodes placed in-line. The first mesh node was placed using the NLOS method and the second was placed using the LOS method. The goal was to observe the quality of a routing change from a direct link to a single hop, then to a double hop.

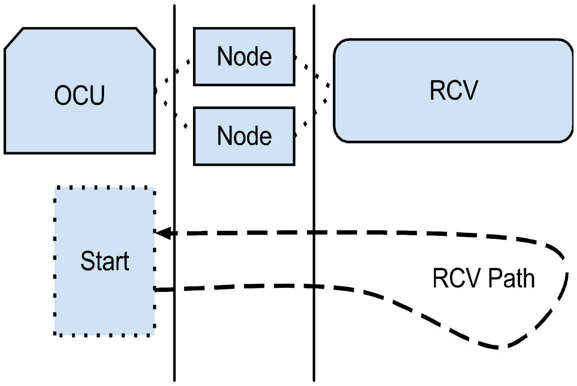

Figure 5. Test 6: route flapping. This tests a common case where route flapping is observed.

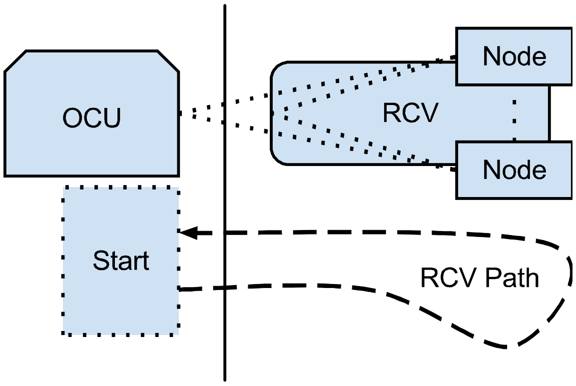

Figure 6. Test 7: two nodes mounted to RCV. This tests that two nodes, mounted to the vehicle, do not deteriorate teleoperation performance.

Test 6, diagrammed in Figure 5, consisted of two mesh nodes placed within 2 meters of each other. The goal was to observe the quality of a route change in a situation where route flapping was observed.

Test 7, diagrammed in Figure 6, consisted of a two mesh nodes mounted on the RCV. The position where communication began to weaken was compared to a baseline test run without mesh nodes. This test ensured that two mesh nodes carried by the RCV did not deteriorate performance.

4. RESULTS

The results of Test 1 are illustrated in Figures 7 and 8. Figure 7 illustrates latency in milliseconds versus number of hops while Figure 8 shows throughput in megabits per second versus number of hops. In all cases *babeld* converged in under 1 second.

Figure 7. Test 1 result: approximate latency.

Figure 8. Test 1 result: approximate throughput.

Comparing throughput with the RCV's data needs determined that a mesh of up to four mesh nodes could easily support teleoperation network traffic. Previous experience with robotic teleoperation had shown that an additional 20 ms of RCV control and video latency could be tolerated. Therefore, the 20 ms of added latency with six mesh nodes would not pose a problem with operation of the RCV.

Tests 2, 3, 5, and 6 were conducted to gain a better understanding of the performance of *babeld* and to identify any weak points of operation. Figure 9 shows a summary of what was observed for each test. Test 2 showed signs of route flapping at the fringes of the mesh connection for the RCV and the onboard mesh node. Video and control was intermittent, with periods of up to a few minutes of complete communication loss. Test 3 showed signs of route flapping when the RCV was near the mesh node. Video and control was intermittent, with periods of up to a few minutes of complete communications loss. When the RCV was moved to an area where it could no longer connect to the OCU directly and was given a minute for the route to settle, video and control operated without any major problems. Test 5 and 6 were attempted but RCV video and control could not be maintained long enough to perform any useful experiments. The OCU, RCV, and mesh nodes showed signs of route flapping and slow network convergence at all mesh nodes.

	Test 2	Test 3	Test 5	Test 6
Pixelation	O	O	O	O
Smearing	O	O	O	O
Choppy	O	O	O	O
Intermittent	O	O	O	O
Delayed				

Figure 9. Results of Tests 2, 3, 5, and 6. An 'O' means that the effect was observed. An empty space means the effect was not observed.

The tests showed that when mesh nodes were introduced into the route, the main data path would flap between two or more possible links. The route without data flowing through it appeared more reliable than the route with data flow. It was likely that the data flow from the RCV affected the Estimated Transmission Cost (ETX)[8] calculation of *babeld*. This was probably due to the RCV data stream nearly saturating available bandwidth, causing some of the packets used to calculate ETX to be lost. The ETX would then increase on the selected route, causing *babeld* to choose an alternate mesh node as a better route and switch to it. This process would repeat itself, causing the route to alternate between two or more mesh nodes. It is likely that the effects of this were exacerbated by relatively slow route convergence time.

5. SOLUTION

The solution described in this paper modifies *babeld* to form a mesh network that provides uninterrupted vehicle teleoperation. Multiple ETX algorithms were evaluated and the following was selected empirically by observing the network's behavior. Only the implemented method will be discussed.

Solving the route flapping problem required two steps. It was observed that ETX values showed a sensitivity to teleoperation traffic which probably caused routes to flap between mesh nodes. To counteract this, it was determined that as long as a link was good enough to carry the required network traffic, the precise ETX value was unimportant. Therefore, all ETX values below a certain threshold were classified as perfect. All values above the threshold were doubled to more heavily penalize a poor link and discourage any routes through that mesh node. Next, hysteresis at the threshold level was added. With these changes, decent performance was observed under tests 2-7. However, whenever a link would need to switch to a new route, a 10-30 second period of intermittent video and control occurred, very likely due to slow convergence time.

The easiest way to decrease convergence time is to increase Babel's “hello” interval. The "hello" interval is the time between "hello" packets, used by Babel to calculate ETX. While this approach increased mesh overhead and ETX sensitivity, overall the effects were beneficial. Experimenting with a few different “hello” rates found a rate that was able to decrease the convergence time while still providing a good teleoperation link. Route convergence went from approximately 10-30 seconds to under 1 second. With the decreased convergence time, most remaining cases of route flapping converged quickly enough that no interruptions in video and control were observed.

Running Tests 2 through 7 on the modified *babeld* resulted in excellent performance. Figure 10 shows what was observed with the modified *babeld*. Adding a new mesh node to the network no longer created an unreliable link. Most often route changes resulted in smooth transitions that were undetectable to the user. The few rarely encountered issues lasted for a short period of time, normally less than 2 seconds.

	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7
Pixelation		O	O	O	O	
Smearing				O		
Choppy						
Intermittent						
Delayed						

Figure 10. Results of Tests 2 through 7. An 'O' means observed. An empty space means the effect was not observed.

6. NETWORK TOPOLOGY

Because *babeld* is a routing protocol, OCU and RCV network traffic need to route through the mesh. For much of the initial testing, the OCU and RCV were reconfigured to natively route through the mesh network. This configuration process proved impractical for a fielded product due to the many variations in RCV and OCU configurations. To overcome these configuration issues a virtual private network (VPN) would be used. O*penVPN* was configured to use UDP and with encryption and any redundancy mechanisms (such as retries and verification) disabled. This allowed the VPN to act as a UDP wrapper for physical Ethernet packets from the RCV and OCU, as if connected by a virtual cable. Because this happens at the hardware layer, any Ethernet network can be connected to any other network without any special configuration. The Wireless Area Network (WAN) was then optimized to accommodate the overhead generated by the VPN by increasing the maximum transmission unit (MTU) of the WAN to the size of the physical networks packets plus the VPN header overhead. Figures 11 and 12 show how the VPN affects mesh throughput and latency in Test 1. While slightly raising latency and lowering throughput, the effects of the VPN were negligible for the operation of the RCV.

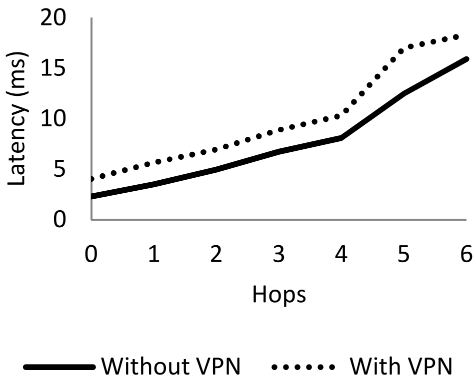

Figure 11. Approximate latency comparison with and without a VPN.

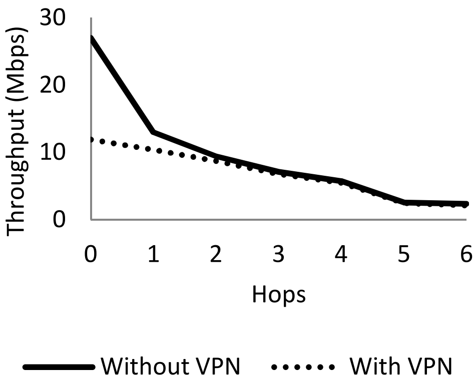

Figure 12. Approximate throughput comparison with and without a VPN.

7. SUMMARY

Building mesh networks for robot teleoperation is challenging due to the mobility of the mesh nodes, the changing and uncontrolled operating environments, and the requirement for near-zero network interruption. Research conducted on various mesh network protocols led to three potential solutions (OLSR, B.A.T.M.A.N., and Babel) that were considered for use. Two protocols (OLSR and Babel) were tested for performance, and Babel was selected for further optimization for robotic teleoperation. To prevent route-flapping (a commonly encountered problem) an ETX threshold with hysteresis was implemented. Additionally, network convergence time was significantly decreased by increasing the "hello" packet rate. Finally, to produce a plug-and-play system requiring no modification to the OCU and RCV software, a tuned VPN was used. These modifications resulted in the development of a robust mesh network that is integrated into the MDCR system, which will be fielded for use with tactical and explosive ordnance robots currently in theater.

8. REFERENCES

[1] Pezeshkian, N., Nguyen, H.G., Burmeister, A., “Unmanned Ground Vehicle Radio Relay Deployment System for Non-line-of-sight Operations,” Proc. 13th IASTED Int. Conf. on Robotics and Applications, Wuerzburg, Germany, August 29-31 (2007).

[2] Yang, Y., Wang, J., Kravets, R., “Designing routing metrics for mesh networks,” Proc. of WiMesh, IEEE Press (2005).

[3] Clausen, T., Jacquet, P., “Optimized Link State Routing Protocol,” IETF, RFC 3626 (October 2003).

[4] Neumann, A., Aichele, C., Linder, M., Wunderlich, S., “Better Approach To Mobile Ad-hoc Networking (B.A.T.M.A.N.),” IETF Internet-Draft (2008).

[5] Chroboczek, J., “The Babel Routing Protocol,” IETF, RFC 6126 (April 2011).

[6] Abolhasan, M., Hagelstein, B., Wang, J. C.-P., “Real-world performance of current proactive multi-hop mesh protocols,” IEEE APCC, Shanghai China (October 2009).

[7] Murray, D., Dixon, M., Koziniec T., “An Experimental Comparison of Routing Protocols in Multi Hop Ad Hoc Networks,” ATNAC (2010).

[8] De Couto, D. S. J., Aguayo, D., Bicket, J., Morris, R., "A High-throughput Path Metric for Multi-hop Wireless Routing," Wireless Networks 11 (4), 419-34 (2005).