Quality of Service (QoS) Sensitivity for the OSPF Protocol in the Airborne Networking Environment

Diane Kiwior, Elizabeth G. Idhaw, and Steven V. Pizzi

The MITRE Corporation
202 Burlington Road
Bedford, MA 01730

ABSTRACT

In this paper we examine the use of the Open Shortest Path First (OSPF) Protocol for use within the future Airborne Network. OSPF is one of the standards-based routing protocols that could be used within the Airborne Network. However, given the fading characteristics of airborne line-of-sight (LOS) channels, OSPF may not be as effective as it is in the commercial terrestrial Internet. Therefore, some modification of OSPF timer settings, most notably the "Hello" protocol timers, may be necessary to enable OSPF to be useful for airborne networking.

Our experiments, which examined the effects of OSPF settings in an Airborne Network environment, demonstrate that configuring the settings to provide faster convergence can reduce the traffic loss by up to 80%. Average packet latency and average packet jitter were not significantly affected by the changes to the OSPF timers. It will be important for the Airborne Network to consider connectivity interruptions when configuring a routing protocol to limit data losses.

I INTRODUCTION

An IP-based Airborne Network for the Air Force has been motivated by the Department of Defense's emphasis on network-centric operations. The Airborne Network would need to support the communication transport needs of multiple platforms, including widebodies such as AWACS, Joint Stars, or E-10A, as well as strike assets such as fighters and bombers. The platforms have varying capabilities and the connections between them may be of different types. Interoperability among these airborne platforms is a key goal for the information infrastructure.

Therefore, the on-board network architecture must be adaptable to mission needs, and each platform must act as a node within the larger airborne network. Standards-based architecture and protocols must be leveraged to provide this flexible and interoperable capability.

In this paper we examine the performance of quality of service (QoS) techniques with an underlying routing protocol in an Airborne Network. We consider the Open Shortest Path First (OSPF) Protocol, which is one of the standards-based routing protocols that could be used within the Airborne Network. Given the fading characteristics of airborne line-of-sight (LOS) channels, OSPF may not be as effective as it is in the terrestrial Internet. Therefore, we expect some modification of OSPF parameters, most notably the "Hello" protocol timers, will be necessary to enable OSPF to be useful for Airborne Networks.

Quality of Service(QoS)

Implementing QoS has proven to be challenging for the wired commercial world. Most commercial approaches solve the QoS issue in traditional wired networks by overprovisioning (i.e. allocating more than enough bandwidth to satisfy needs)[1]. Other QoS methods, such as Integrated Services (IntServ) [2], and Differentiated Services (DiffServ) [3]-[5], have been proposed but have not been widely deployed. In addition, since these solutions have been based on an environment where connections are stable and always available, the performance of the underlying routing protocol has not been an issue.

Implementing QoS in an airborne network is an even more difficult problem. The variable link quality of (wireless) airborne links presents additional challenges to any QoS mechanism. Overprovisioning is not a viable option for tactical networks that are constrained by limited bandwidth resources. For both IntServ and DiffServ, the question of whether QoS can be delivered successfully is related to the routing protocol's ability to react to transient link characteristics. Routing protocols in use today have not needed to deal with link outages on a regular basis. They have not been tested in situations where link variability is the norm.

OSPF

OSPF is the de facto standard for medium to large size networks and is deployed in most IP-based enterprise networks [6]. OSPF was designed for the relatively fixed structure of the Internet, has worked well and is well-known. Because of its widespread use, modifications to OSPF have been proposed to solve a number of problems. In addition to starting with a well-known basic protocol, modifying a commercially proven routing protocol allows backward compatibility with legacy OSPF routers as well as easier integration with a wired OSPF network.

Link outages occur occasionally in wired networks but are an inherent characteristic of airborne networks. Routing protocol performance in an airborne network must be evaluated in terms of fast recovery from topology changes.

Since OSPF was designed for wired networks, it has minimal provisions to deal with link outages. The default OSPF settings relating to recognizing and recovering from link outages result in convergence times on the order of tens of seconds. This delay is primarily due to the Hello protocol that discovers link failures. Recently, faster failure detection in wired networks has been the focus of several research papers in an attempt to increase network reliability and enable multi-service traffic. For example, [7] suggests reducing the OSPF timer settings to decrease the number of packets lost during link failures in a wired network. Both [8] and [9] recommend reducing the granularity of link failure timers to milliseconds rather than seconds to improve the stability of wired networks. In response to the interest of improving convergence times, Cisco now provides a method of setting sub-second Hello Intervals [10] as well as a method to schedule route table recalculations in millisecond intervals [11].

Other research into the modification of OSPF has focused on reducing protocol overhead messages to meet scalability for wireless networks. Baker [12] suggests extensions to OSPF v3 to support movement among different OSPF area routers by a Mobile Ad-hoc Network (MANET). Reference [13] adapts OSPF to a wireless interface by limiting the flooding of Link State Update messages, thereby reducing the routing overhead. While [13] has explored the scalability of their own modifications, they have not investigated their performance in the presence of link outages.

In this paper, we present our evaluation of OSPF timer modifications and QoS performance in an emulated airborne network. Our experiments investigate reduced OSPF timer intervals as a means of reducing packet loss during topology changes. We are interested in controlling the overhead required by OSPF timer modifications and also in the effectiveness of OSPF timer modifications on routing convergence in the presence of frequent link outages.

II APPROACH

In this section we describe our method to evaluate OSPF responsiveness and the performance of DiffServ QoS mechanisms with an underlying OSPF routing protocol in an emulated airborne network. Since DiffServ assigns per-hop behavior to different classes of traffic, the underlying routing protocol plays a crucial role. It is the responsibility of the routing protocol to recognize link outages and reroute traffic appropriately. The OSPF parameters and the role they play in reacting to link failures are listed in Table 1 along with the Cisco Default Ethernet Settings [16]. The Hello and Dead Intervals are directly related to the time it takes for a topology change to be detected while the SPF Delay and SPF Holdtime determine the delay between detection of a topology change and the calculation of new routes. The Minimum Link State (LS) Interval limits the time between origination of Link State Advertisements, which are used to notify the network of topology changes.

Table 1. OSPF Configurable Timer Intervals.

OSPF Timer Settings	Description	Cisco Default Settings (sec)
Hello Interval	Time delay between successive Hello packets	10
Dead Interval	Time delay since the last Hello before a link is declared down. Usually 4 times the Hello Interval.	40
SPF Delay	Time delay between a link update notification and the ShortestPath First (SPF) calculation	5
SPF Holdtime	Minimum time delay between successive Shortest Path First (SPF) calculations	10
Minimum LS Interval	Minimum time between distinct originations of any particular Link State Advertisement (LSA).	5

Experimental Set-Up

We developed an experimental scenario to emulate an airborne node-to-airborne node configuration shown in Figure 1. Details of the laboratory implementation of the airborne network architecture are available in [17]. The airborne nodes can communicate via either a wideband, low reliability link or a narrowband, high reliability link. The wideband link is typical of a line-of-sight (LOS) link which provides a high bandwidth but is subject to variable signal strength. The narrowband link is typical of a beyond-line-of-sight (BLOS) link with high reliability but low bandwidth, as in a satellite link.

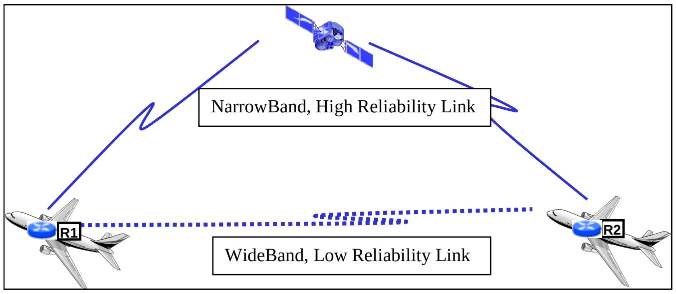

Figure 1. Experiment Scenario.

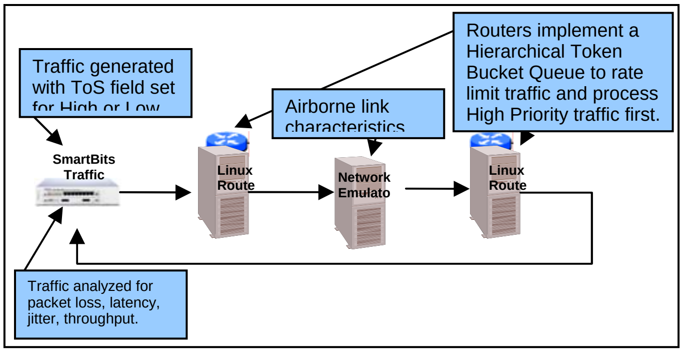

Figure 2. QoS Model for MITRE Experiment.

The QoS model as implemented in our lab is shown in Figure 2. The Linux routers implemented priority queueing via a Linux HTB (Hierarchical Token Bucket) queue. HTB provides priority queueing as well as bandwidth shaping. HTB allows several classes to share a fixed amount of bandwidth and prioritizes how the bandwidth will be shared. Each class is allocated a specified amount of bandwidth. If a particular class uses less than the amount allocated to it, the remainder can be shared among the other classes, according to their priority. In our testbed, we set the HTB to 150 kb/s bandwidth for the narrowband link and specified that High Priority (HP) traffic could access this bandwidth first, followed by Low Priority (LP) traffic, and any remaining bandwidth would be used for unclassified traffic. Classes were identified by the Type of Service (ToS) field in the IP header. The SmartBits traffic generated HP traffic with a ToS setting of 8 and LP traffic with a ToS setting of 4.

HTB also addressed the problem of regulating the rate of packet transmission from the Linux routers to implement the narrowband link. Without a transmission rate regulator, packets dequeued from the Linux routers are sent out at the network speed of the lab. In an airborne network, the network speed of BLOS radios is significantly less than wired Ethernet speeds. To ensure that our testbed emulated a narrowband radio channel, the Linux HTB mechanism was set to 150 kb/s.

Traffic Generation and Channel Modeling

In the lab experiments, the MITRE Network Emulator [18] was used to model the communications links. The Network Emulator allows a radio system to be modeled to reflect real-world system performance by specifying throughput, delay and bit error rate (BER). In our experiments, the link quality or BER was presented as a binary quantity, either on (i.e., no errors/perfect transmission) or off (i.e., no signal), to avoid complications due to variable signal strength. Performance of variable strength signals will be the subject of future experiments. Traffic was generated by a SmartBits 600 chassis. The Linux routers were configured to run OSPF routing, implemented with GNU zebra/OSPFv2 code with HTB queues as described above. By assigning link costs appropriately, the OSPF routing protocol was configured to prefer the wideband (WB) link; but upon recognition of an outage of the WB link, OSPF rerouted traffic to the narrowband link.

A variety of experiments were conducted to assess the performance of OSPF and DiffServ QoS. Experiment 1 investigated the effects of modifying OSPF timers on routing protocol responsiveness to link changes and routing protocol overhead. Experiment 2 investigated the effects of link outages on packet loss, latency and jitter for prioritized traffic.

III RESULTS

Experiment 1

Experiment 1 addressed the effects of modifying OSPF timer intervals on convergence time (i.e. the time needed to recognize an outage or appearance of a new link and to reroute traffic appropriately). There are multiple steps and multiple timers involved in determining a routing protocol's convergence time, including detection of a topology change, propagation of the change to other routers and calculation of new routes. Detection time of a topology change is directly related to the Hello Interval and Dead Interval timers. If a Hello Packet is not received by the Dead Interval time, a link outage is recognized. The appearance of a new link is recognized by the arrival of a Hello Packet on the interface to the new link. Once a router has detected a topology change, this information is flooded throughout the network in a Link State Update (LSU) packet. In our simple two-node test bed, the time to propagate the flooded LSUs is minimal but actual release of the LSU message is restricted by the Minimum LS Interval. The calculation of new routes is triggered by the receipt of an LSU message, but the SPF Delay and SPF Holdtime timers restrict the start of the calculation. In our 2-node test bed, the SPF calculation time is minimal but the SPF timers still delay calculation of new routes. Decreasing the timer intervals listed in Table 1 increases OSPF responsiveness to link failures but also increases the protocol overhead and may not scale in larger topologies; the advantages of quick recovery must be balanced against the need for scalable Airborne Networks.

The OSPF timers were varied according to the settings listed in Table 2 with baseline OSPF Default values set as per Cisco router default Ethernet settings [16]. For each of the settings in Table 2, a 180 second test was run with one failure and one recovery of the wideband link. The traffic load and link characteristics for this experiment are shown in Figure 3.

Results of Experiment 1 shown in Figure 4 clearly indicate that reducing the timer intervals from the Cisco Ethernet default values improves responsiveness to link failures. In addition, comparing results for Settings B through G indicate that reducing the timer intervals improves response time without increasing routing overhead. OSPF's responsiveness to topology changes was measured by calculating the link failure detection time and link recovery response time from packet captures on the routers' interfaces. Link failure detection time is measured as the time difference between an expected Hello message and the cessation of traffic over the interface to the failed link. Link recovery response time is measured as the time difference between the notification of a new route (i.e. the receipt of a new Hello packet) and the receipt of the first data packet over the interface to the new link. It was expected that the Link Failure Detection Time should be in the same range as the Dead Interval. This was the case for the Default settings, as well as for Setting A. However the Link Failure Detection time for Setting B did not correlate to the Dead Interval of 4 sec. Here the SPF Holdtime adds additional delay. Decreasing the SPF Delay and SPF Holdtime improved the Link Failure Response results seen for Settings C through G. It appears that link Failure Response is controlled by the maximum (Dead Interval, SPF Holdtime). The Link Recovery Response Time is clearly limited by the Minimum LS Interval. When the Minimum LS Interval is set to 1 second (i.e. Settings D, E, and G), the Link Recovery Response falls in the range of 4 seconds. The Minimum LS Interval setting of 5 seconds (i.e. Settings Default, A, B, C, and F) results in Link Recovery Response that ranges from 7-9 seconds.

OSPF Timer Settings	**Hello Interval (sec)**	**Dead Interval (sec)**	**SPF Delay (sec)**	**SPF Holdtime (sec)**	**Min LS Interval (sec)**
OSPF Defaults	10	40	5	10	5
A	5	20	5	10	5
B	1	4	5	10	5
C	1	4	1	4	5
D	1	4	1	4	1
E	1	4	0	0	1
F	1	2	0	0	5
G	1	2	0	0	1

Table 2. OSPF Timer Settings Used in Experiment 1.

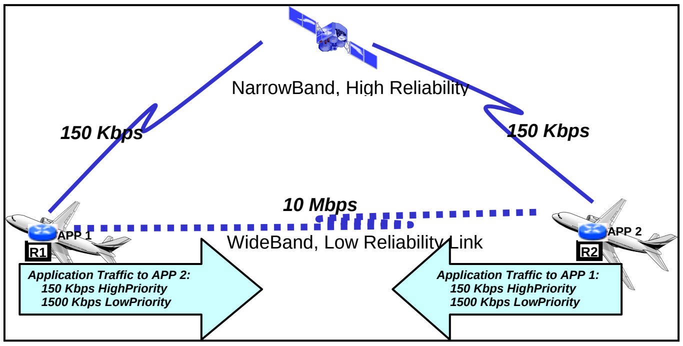

Figure 3. Settings for Experiments 1 and 2.

The routing overhead reported in Figure 4 corresponds to the traffic generated by Hello packets from the two nodes in our test bed. The Hello overhead is a baseline of minimum overhead required by the routing protocol that cannot be reduced. When there are topology changes, OSPF generates additional messages, whose sizes depend on the number of nodes in the network. In our 2-node configuration, a single topology change incurred a cost of 1154 Bytes of traffic, consisting of Database Description, Link State Update, and Link State Acknowledgement packets. The size of the Hello, Database Description and Link State Update packets would increase with the number of nodes in the network.

Since this experiment did not represent the varying signal strength of RF links, there may be additional complications related to fast timers when link availability fluctuates. Further study is needed to determine the effects of varying OSPF settings in these conditions. It is important to emphasize that these results are based on a limited two-node topology. As the number of nodes increases, convergence time will increase, due to the increased time to calculate and propagate new routes to all routers.

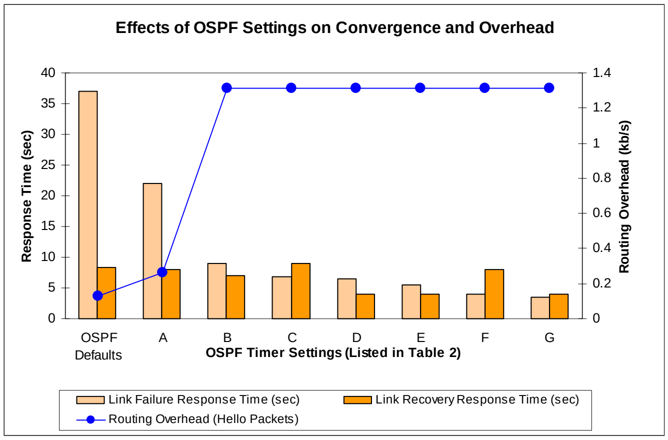

Figure 4. Effects of Varying OSPF Timer Settings on Convergence Time and Routing Overhead.

Experiment 2

Experiment 2 investigated the effects of link outages on the packet loss, latency, and jitter of prioritized traffic. The traffic was marked as either High Priority (HP) or Low Priority (LP) by setting the Type of Service (ToS) field of each IP packet at the data source. The number of link outages was varied for each test, from one failure every 10 minutes to one failure per minute. The experiment was run with four different OSPF settings with traffic load and link characteristics as shown in Figure 3.

Figure 5 demonstrates that minimized OSPF timer intervals (i.e., Hello Interval: 1 sec, Dead Interval: 4 sec, SPF Delay: 1 sec, SPF Holdtime: 4 sec and Minimum LS Interval: 5 sec) can result in 80% reduction of packet loss for the High Priority (HP) traffic when compared to the HP loss under OSPF Defaults. All traffic was set to prefer the wideband link but upon link failure, traffic was directed over the narrowband link, where the HP traffic was given preferential treatment in the HTB queue.

Figure 6 shows that minimized OSPF timer values have no benefits for LP traffic; a result that reflects the experiment's link bandwidth. In each experiment, the wideband link was available 50% of the time and the LP traffic was forced to contend with the HP traffic for the narrowband link. Since the HP traffic was set at a data rate of 150 kb/s, there was minimal opportunity for LP traffic to access the 150 kb/s narrowband link. If unlimited bandwidth had been available on the failover link and no priority treatment had been implemented, both HP and LP traffic would experience identical losses, due to the time taken to switch the traffic over to the high-reliability link.

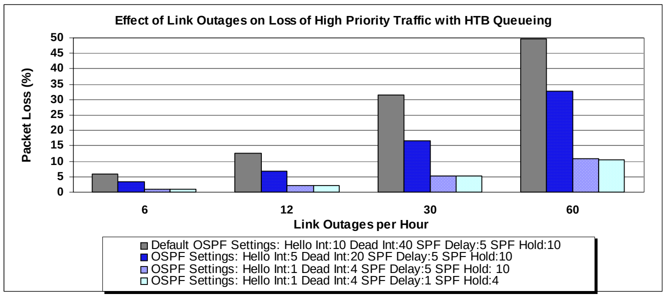

Figure 5. Effects of Link Failures on Loss of High Priority Traffic for Different OSPF Settings.

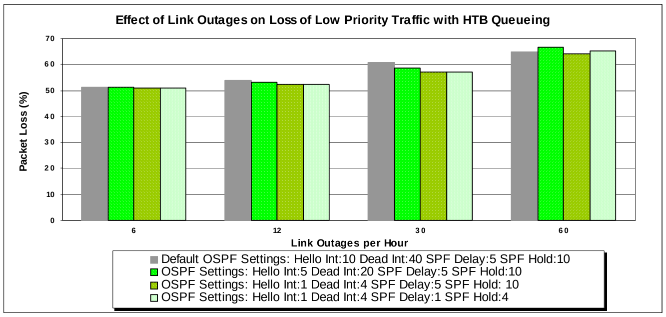

Figure 6. Effects of Link Failures on Loss of Low Priority Traffic for Different OSPF Settings.

Figure 7 displays the average packet latencies of HP and LP traffic under different link outages and OSPF settings. While OSPF settings and higher frequencies of link outages do not significantly affect HP latency, there is some variation in the HP average packet latency measurements. Minimized OSPF settings and high frequencies of link outages affect the latency of LP traffic. At high rates of link failures (30 and 60 outages per hour), the LP latency increases as the OSPF settings are minimized.

Figure 8 displays the average jitter of HP and LP traffic under different link outages and OSPF settings. There is no clear relationship between rates of link failures, minimized OSPF settings and jitter.

It is clear that HP traffic experiences decreased average latency than LP traffic, regardless of frequency of link failures, and OSPF settings. This is to be expected as a direct result of priority queueing at the routers. It is unclear why jitter is higher for the HP traffic relative to that of the LP traffic.

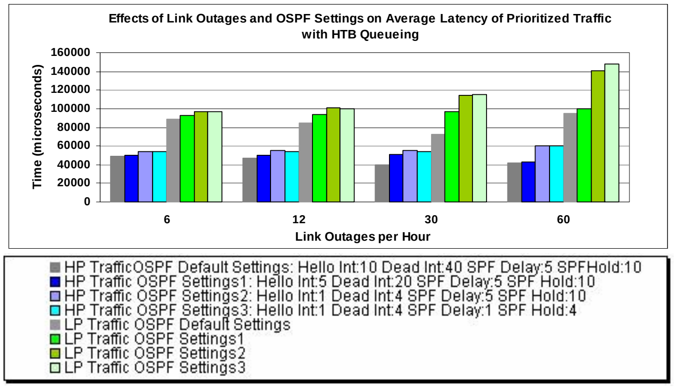

Figure 7. Effects of Link Failures on Packet Latency of Prioritized Traffic with Different OSPF Settings.

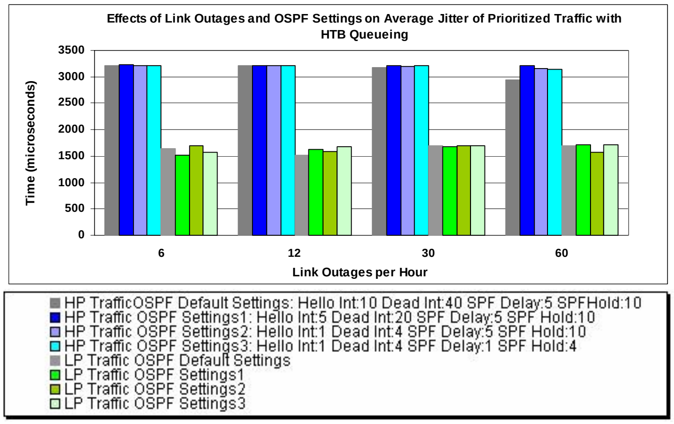

Figure 8. Effects of Link Failures on Jitter of Prioritized Traffic with Different OSPF Settings.

III CONCLUSIONS

Our experiments studying the effects of OSPF settings in an Airborne Network environment demonstrated that configuring the settings to provide faster convergence reduced the loss of High Priority traffic by up to 80% with no significant effects on latency and jitter. These results demonstrate the importance of modifying the timers of a link state routing protocol, such as OSPF, in an Airborne Network. Rather than relying on default settings, the timer settings of an airborne routing protocol need to address the intermittent behavior of the links. Precise optimization of the timer settings will involve additional research.

Since the work presented here involved a simple 2- node configuration, further studies with larger topologies are needed to address scalability issues with respect to router overhead, network-wide stability, and complexity of routing protocol update calculations. Future work should also include examination of issues related to a varying bit-error rate rather than the binary “on-off” link availability used in this set of experiments. A varying bit-error rate reflects actual real-world conditions and will expose more issues related to route stability versus routing protocol reaction to topology changes. Use of MITRE’s Network Emulator and the Satellite Tool Kit (STK) with an airborne traffic pattern will provide realistic representations of airborne RF communication links. Also, our current work has focused on QoS for UDP traffic only. It is important to understand the effects of intermittent link availability on TCP traffic as well. Future work should include both UDP and TCP traffic streams.

IV ACKNOWLEDGEMENT

The authors would like to thank Lt. Col. Andrew Beaudoin of the USAF and Charles Arouchon, Claudia Ward, James Apicco, William Wilder, and Thomas Wheeler, of the MITRE Corporation for encouraging us in this effort.

V REFERENCES

[1] G. J. Armitage, “Revisiting IP QoS: Why do we care, what have we learned?” ACM SIGCOMM 2003 RIPOS Workshop Report, ACM SIGCOMM Computer Communications Review, Vol. 33, No. 5, October 2003.

[2] R. Braden, D. Clark, and S. Shenker, “Integrated Services in the Internet Architecture: An Overview,” RFC 1633, June 1994.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Want, and W. Weiss, “An Architecture for Differentiated Service,” RFC 2475, December 1998.

[4] Cisco Systems, “DiffServ – The Scalable End-to-End QoS Model,” Cisco Systems White Paper available at http://www.cisco.com/en/US/tech/tk543/tk766/technologies_white_paper09186a00800a3e2f.shtml, 2001.

[5] B. Davie, “Deployment Experience with Differentiated Services,” Proceedings of the ACM SIGCOMM 2003 workshops, August 2003.

[6] J. Moy, “OSPF Version 2,” RFC 2328, April 1998.

[7] M. Goyal, K. K. Ramakrishnan, and W. Feng, “Achieving Faster Failure Detection in OSPF Networks,” Proceedings of the IEEE International Conference on Communications (ICC 2003), Alaska, May 2003.

[8] A. Basu and J. G. Riecke, “Stability Issues in OSPF Routing,” SIGCOMM 01, August 2001.

[9] H. Yu, C. Alaettinoglu, and V. Jacobson, “Towards Milli-Second IGP Convergence,” IETF Internet Draft: draft-alaettinoglu-ISIS-convergence-00, November 2000.

[10] Cisco Systems, “OSPF Support for Fast Hello Packets,” Cisco IOS Releases 12.0(23)S, 12.2(15)T, and 12.2(19)S Technical Documentation available at http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/12os/120s23/fasthelo.htm, 2003.

[11] Cisco Systems, “OSPF Shortest Path First Throttling,” Cisco IOS Software Release 12.2(14)S Technical Documentation Feature Guide available at

http://www.cisco.com/en/US/products/sw/iosswrel/ps1838/products_feature_guide09186a0080134ad8.html.

[12] F. Baker, "An outsider's view of MANET," Internet-Draft: draft-baker-manet-review-01.txt (expired), March 2002.

[13] T. R. Henderson, P. A. Spagnolo, and J. H. Kim, "A Wireless Interface Type for OSPF," Proceedings of 2003 IEEE MILCOM Conference, October 2003.

[14] Advanced Networks and Applications Group, MIT Lincoln Laboratory, "Link Outage Detection and Recovery Experiment Report," (FOUO), AN-SIG Report, December 2003.

[15] Air Force Experimentation Office, Joint Expeditionary Force Experiment (JEFX) 2002 available at http://afeo.langley.af.mil.

[16] Cisco Systems, "Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2," available at http://www.cisco.com/application/pdf/en/us/guest/products/ps1835/c1051/ccmigration_09186a008011dfe9.pdf, 2001.

[17] E. Idhaw and D. Kiwior, "MC2A Network Emulation Facility," MITRE Working Note WN 04B0000011, February 2004.

[18] K. Grace, "Emulator STK Interface Manual," Version 1.0, The MITRE Corporation, January 2004.