Experiences With Monitoring OSPF on a Regional Service Provider Network

David Watson
Farnam Jahanian
University of Michigan
Department of Electrical Engineering and Computer Science
Ann Arbor, Michigan 48109-2122
{*dwatson,farnam*}*@eecs.umich.edu*

Craig Labovitz
Arbor Networks
625 E. Liberty Street
Ann Arbor, Michigan 48104
labovit@arbornetworks.com

Abstract

This paper presents the results from a detailed, experimental study of OSPF, an intra-domain routing protocol, running on a mid-size regional Internet service provider. Using multiple, distributed probes running custom monitoring tools, we collected continuous protocol information for a full year. We use this data to analyze the health of the network including the amount, source, duration and periodicity of routing instability. We found that information from external routing protocols produces significant levels of instability within OSPF. We also examine the evolution of the routing topology over time, showing that short term changes are incremental and that the long term trend shows constant change. Finally, we present a set of detailed investigations into several large scale anomalies. These anomalies demonstrate the significant impact external routing protocols have on OSPF. In addition, they highlight the need for new network management tools that can incorporate information from routing protocols.

1 Introduction

Routing protocols are a critical component of the Internet. Their goal is to ensure that normal traffic is able to efficiently make it from source to destination. While routing protocols are based on simple theories, designing a protocol that functions correctly and efficiently under real loads is difficult. In addition, testing the protocols under realistic scenarios is often impossible. Simulations of large scale routing are either too complex to perform or too simplistic to accurately predict the challenges encountered on real networks. Large scale testbeds don't exist, and monitoring the protocol on production systems is also difficult. This paper presents the results of a detailed study into just this kind of real-world system.

This study collected routing protocol data from MichNet, a mid-sized regional Internet service provider covering the state of Michigan. Data was collected from four geographically distributed probe machines running custom monitoring software. This year-long effort collected information continuously from MichNet's intra-domain routing protocol, OSPF. We used this data to analyze the performance of OSPF on a production network and discovered surprising anomalies. There is a long history of studies looking for these kinds of problems in routing protocols. Detailed studies of routing on the Internet have uncovered significant problems with the performance of the Border Gateway Protocol (BGP) [9; 15; 13; 14; 16]. BGP, an inter-domain routing protocol, interconnects the autonomous systems that comprise the Internet. As such, any significant problems with BGP can affect routing on the Internet as a whole. Unlike BGP, the work on intra-domain routing protocols such as OSPF and IS-IS has mainly focused on their theoretical behavior. Distinguished by the fact that they are run within an autonomous system, intra-domain routing protocols have a significant impact on the performance of the local network. In addition, because inter-domain and intra-domain routing protocols are often interdependent, problems with one can affect the performance of the other. While these studies based on simulations and small scale testbeds have identified important issues, they still fall short of identifying the issues that affect these protocols on production networks. Our work, however, has focused on understanding the performance of intra-domain protocols under real-world conditions. We attempt to highlight several anomalies of OSPF that are not seen under controlled conditions.

Our work also complements the few existing studies that have looked at the behavior of intra-domain routing protocols on production networks. Our work extends these studies by providing year-long analysis, corroborating

others' results and introducing previously unseen results. A study of Qwest's network focused on issues affecting the convergence time of IS-IS [1]. This study identified single, flapping links as the predominant source of instability. It also made the theoretical case for using an incremental shortest path algorithm. Our work corroborates this by identifying the lopsided distribution of problem links and providing practical measurements supporting the argument for an incremental shortest path algorithm. Our work also complements two studies concurrent to ours. The first paper analyzes link failures on Sprint's IS-IS backbone [10]. It examines the temporal locality and duration of link failures in addition to studying the effects of induced link failures. The second paper provides a more comprehensive study of OSPF on an enterprise network [23]. It also provides temporal and frequency views of instability in the network, and makes a distinction between internal OSPF links and external links. Finally, the paper looks at issues with redundant LSAs caused by OSPF's reliable forwarding mechanism. Our work presents several complimentary results to these two papers. First, we present an analysis of the amount, duration, and frequency of updates seen on MichNet. We also analyze these updates both by source and by the type of change they represent. By collecting a year's worth of routing updates we are also able to demonstrate that these results are not singular anomalies but rather common issues across time and different networks.

In addition to providing complementary results to these existing and concurrent studies, our work introduces new issues with intra-domain routing. Our analysis found significant periods of localized instability. The predominant source of this strange routing behavior was from customer networks. These routes are injected into OSPF from other routing protocols such as RIP. Like most ISPs, MichNet is constructed from around 50 core, backbone routers connecting a much larger number of customer networks. Often overlooked in routing analysis, this additional layer of hierarchy is a significant source of instability. Unlike the simplified view that would consider MichNet an autonomous system, far more of the routers within MichNet are not actually under the operators' control. These customer networks often have less monitoring, lower levels of redundancy, and often use older routing protocols to maintain their connection to MichNet. These factors all contribute to the increased level of instability we see with these routes. Some of the anomalies we see from these injected routes appear to be caused by well known failure behavior of the originating routing protocol.

We also expand the existing body of knowledge by providing detailed analysis of the source and behavior of several specific anomalies. These anomalies exhibit very surprising behavior that produces notable effects throughout our analysis. In addition to highlighting previously unobserved behavior these anomalies demonstrate the significant impact that information injected into OSPF from external routing protocols can have on the network. These specific anomalies account for the most prominent period of instability, the first and second largest source of instability by router, and the largest source of instability by an individual link. In addition, these anomalies demonstrate the need for new network management tools that understand routing protocols. All of these anomalies appear to go unnoticed by the network operators for significant periods of time. Better monitoring and management of the underlying routing protocols would improve the reliability and performance of the network.

The main contributions of this work are:

- A detailed, year-long analysis of OSPF running on a production network. We examine the overall traffic levels as well as the amount, source, and duration of instability on the network. In addition, we examine the changes in routing topology over time. We also make an important distinction between the core OSPF network, and the edge network comprising the customer connections, leading to new insights about the behavior of OSPF. We corroborate the results seen in previous work, and discuss previously unseen issues.

- The identification of customer networks as a major source of instability injected into OSPF. While it is well known that customer networks are less stable, we show that this external instability causes increased instability in the core OSPF network.

- The identification of individually flapping links as the predominant source of instability. These high frequency oscillations are confined to a single physical link and do not induce failures in other parts of the network. In addition, several routers contribute disproportionately to this instability. We also found that 80% of the individual periods of flapping last less than five minutes.

- The discovery that the routing topology is constantly evolving although the individual changes are incremental. Rather than reverting back to a stable configuration, we found that no single topology of the network lasted longer than 28 days. In addition, we show that a significant majority of the changes to the topology require

only slight modifications to the shortest path tree. This provides a strong argument for the use of an incremental shortest path calculation [1].

- An examination of specific anomalies that highlight impact of external routes on OSPF and the need for advanced network management tools. These anomalies lasted for significant periods of time without being resolved. They demonstrate the significant negative impact that customer routes can have on OSPF. They also demonstrate the need for advanced network management tools that are able to collect information from routing protocols and alert operators of these problems.

2 Background

Since this paper discusses some of the finer points of the OSPF protocol, this section presents a short overview of its operation. Readers who are unfamiliar with the protocol and wish to fully understand the fine details are encouraged to read a more detailed source such as [18].

The Internet is divided into a large number of different regions of administrative control commonly called *autonomous systems.* These autonomous systems (AS) usually have distinct routing policies and connect to one or more remote autonomous systems at private or public *exchange points.* Autonomous systems are traditionally composed of network service providers or large organizational units like college campuses and corporate networks. At the boundary of each autonomous system, peer border routers exchange reachability information to destination IP address blocks, or *prefixes,* for both transit networks and networks originating in that routing domain. Most autonomous systems exchange routing information through the Border Gateway Protocol (BGP).

OSPF is one of the more popular *intra-domain* routing protocols. Unlike inter-domain routing protocols such as BGP, intra-domain routing protocols are used to configure routing within an autonomous system. In general this distinction is not arbitrary as the functional requirements between the two require distinct implementation differences [18]. In general, inter-domain routing protocols are based on distance vector algorithms while intra-domain routing protocols use link-state algorithms. These two algorithms are both distributed, but they differ in how the information and calculations are distributed. Distance vector algorithms distribute the computation so each router doesn't need a global view of the entire network. Link state algorithms however, distribute the data allowing each router a global view of the network and allowing them to perform their own calculations.

With a link-state protocol each router contributes one small piece to a larger distributed database. This information consists of the individual connections that each router has with other routers as well as with networks containing end hosts. This information from each router is distributed to every router participating in the protocol. Once a router has a copy of the entire database, it can calculate the best path for routing packets. This is accomplished by using the database of link information to construct a graph of the network. From there each router runs a shortest-path calculation to each destination. Finally, the router uses this information to construct a routing table describing the next hop for each destination address.

OSPF is a link state protocol, so the fundamental unit of information is the set of links or connections from each router. These links describe the set of connections a router has. These connections can be point-to-point links to other routers, connections to a shared network segment such as an Ethernet LAN, or more complex connections such as a virtual link over an ATM mesh. Each router constructs a Link State Advertisement (LSA) containing this set of links and broadcasts it to its immediate neighbors. These neighbors in turn distribute these LSAs to their neighbors until the information has been distributed to every router participating in the OSPF protocol. Once each LSA has made it to every router, each router has a complete map of the entire network and can then use Dijkstra's shortest path algorithm to compute an optimal routing path.

These LSAs describing the connections from each router are referred to as type 1, or *router* LSAs. There are four other LSA types, two of which we see on MichNet. Type 2, or *network* LSAs are used to describe the routers connected to a broadcast network. Rather than having each router on the network announce a connection to every other router on the network, OSPF uses a type 2 LSA to list all the routers connected to the network. The broadcast network essentially becomes another node in the topology graph. Types 3 and 4 are known as *summary* LSAs. These LSAs are used when OSPF is divided into several areas. Areas are used to improve the scalability of OSPF to larger networks by limiting the amount of information each router needs to keep. The OSPF topology on MichNet is small enough to require only one area, so we never see type 3 or 4 LSAs on the network. The final LSA type, labeled type 5, is used to inject routes learned from other routing protocols into OSPF. This is most commonly used to announce

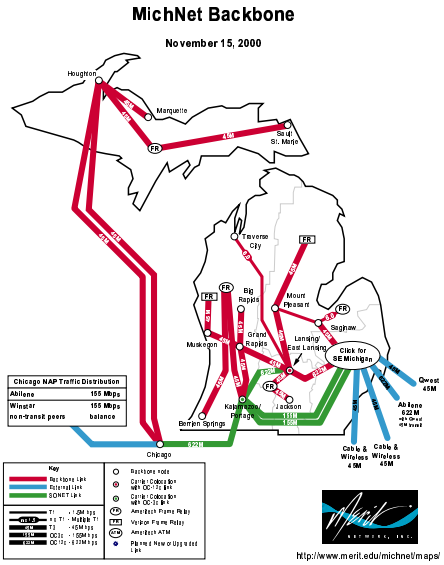

Figure 1: MichNet Topology

connections to customer networks. Each type 5, or *external* LSA specifies the network prefix along with a metric. These routes are not considered when performing the shortest path algorithm. If there are multiple internal routers announcing a connection to the same prefix, whichever router is closer after performing the shortest path calculation is used to route to the external network.

LSAs are announced by a single router and reliably broadcast to every OSPF router. Each router stores every LSA seen in the network in a local database. Each LSA has an age associated with it in order to detect stale information. LSAs are refreshed after 30 minutes by the originating router. If an LSA exists for 60 minutes (MAXAGE) without being refreshed, it is deleted from the database on every router. In order to ensure that the databases stay consistent, once a router times an LSA out of its database, it announces the LSA itself with an age of 60 minutes, to ensure that the LSA is deleted from every router's database.

3 Testing methodology

In order to understand the behavior of OSPF on a production network, we connected several probe machines to MichNet routers. MichNet is a statewide network composed of hundreds of routers covering the state of Michigan (Figure 1). These routers connect over 500 physical locations to the Internet through several outside connections. 50 of these routers form the core OSPF network. The remainder are used to connect for customer connections. In addition to external connections to Cable and Wireless and Qwest, MichNet has a high speed connection to Internet2.

To collect OSPF data from MichNet we deployed three probe machines and used data from an existing IPMA [11] probe machine.[1] The bulk of the raw data used in our analysis comes from the IPMA machine which uses a custom OSPF implementation designed to dump the raw LSAs to disk without any analysis. Three additional machines were deployed running a customized version of Zebra [26]. These customizations allowed us to prevent Zebra from announcing LSAs, to run shortest path and forwarding table calculations from the perspective of each router in the network, and to use Zebra as a simulator based on raw OSPF packet data. Unlike previous studies that use custom packet collection tools, Zebra is a full OSPF implementation. This allows us to monitor the higher level actions of the protocol such as shortest path calculations and the resulting routing tables. In addition, by using a full OSPF implementation we are able to switch from more passive monitoring to actively injecting routes into the network. This can be used to actively introduce faults into the network without disturbing active links. The IPMA machine and one of the Zebra machines were deployed in Ann Arbor, the logical center of the network. The other two machines were

[1]The Internet Performance Measurement and Analysis (IPMA) project develops tools and performs analysis of networks and networking protocols in local and wide-area networks. The goal of the project is to promote the stability and rational growth of the Internet.

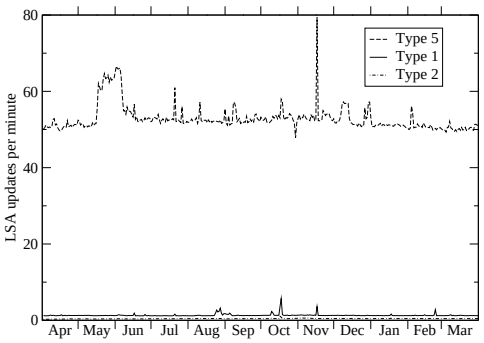

Figure 2: Overall OSPF traffic.

deployed in Houghton and Berrien Springs, both closer to the edge of the network. High level examination shows that the data collected from all four probe machines is consistent.

One of the major issues with collecting data from production networks is the impact that the measurement infrastructure will have on the network itself. We don't want our measurement to affect the analysis, but more importantly we don't want to negatively affect the performance of the network. We have specifically designed our measurement software to not announce any LSAs. However, because we form a full peering relationship with the neighboring router, our probe does affect the list of links between the peering router and the probe machine. The probe machines are connected using a private Ethernet segment, so a change in adjacency status causes the link to switch between a broadcast network and a stub network. For the initial connection, this provides minimal disturbance to the network. We do make an effort, however, to prevent the peering relationship from going down in order to minimize this source of disturbance.

4 Experimental Results & Analysis

With our measurement infrastructure in place, we attempted to answer several questions about the behavior of OSPF on a production network. First, we wanted to examine the overall level of OSPF traffic. This gives us two valuable insights. First is an understanding of how much OSPF traffic we can expect on a production network. Second is an idea of how much instability we see on the network. In addition to measuring the amount of instability, we also wanted to identify periods of instability that could be avoided by changing the routing protocol or the router configuration. Other periods of instability caused by external sources, such as power outages, are out of our control. On the other hand, we are interested in external causes that produce an abnormal amount of protocol traffic. For example, a misbehaving router interface that is rapidly switching between the up and down states can cause the routing protocol to constantly try to adapt to the changing state. Since this interface is not reliably routing packets, it would be better for the routing protocol to assume that the interface is down until the instability has settled. For simplicity, we will refer to these periods of instability that can be fixed through protocol or configuration changes as *undesirable*. Finally, we wanted to understand what impact these undesirable problems have on the network. We wanted to determine both how long these periods of instability last, and what impact they have on the functionality of the network. The main result is that the majority of the undesirable behavior of OSPF is due to the routes injected from customer networks. In addition, both the internal and external instability is localized to a single link with a significant percentage lasting under five minutes. The exceptions, however, demonstrate the extreme impact external routing protocols can have on OSPF and the need for better network monitoring tools.

4.1 Overall traffic

The first thing we wanted to understand about MichNet is the amount of OSPF information produced by the network. Figure 2 shows the overall level of OSPF LSA announcements over a year. Each point on the graph measures the number of LSA updates per second averaged over the period of a day. Note that this is a measure of the number of LSAs seen, not the number of OSPF packets on the network. The most noticeable observation is that there are significantly more type 5 (external) LSA announcements than type 1 (router) or type 2 (network) LSA announcements. As with

most ISPs, MichNet consists of a relatively small number of core backbone routers that connect their customers to the Internet. Since the number of customer routes is much larger than the number of backbone routers, we see many more external LSAs than router LSAs.

While understanding the overall level of traffic is important, our underlying goal is to differentiate between traffic that conforms to the design of the protocol and traffic that is caused by instability. On a typical day, we see about 1700 active external routes. Each one of these routes is represented by an LSA that is supposed to be refreshed every 30 minutes, and in stable state, these are the only updates we would expect to see. 1700 updates per 30 minutes corresponds to 56.7 announcements per minute. As we'll see later, most of the routers on MichNet wait 4 minutes before announcing LSA refreshes. Therefore, we really expect to see 1700 announcements per 34 minutes or 50 announcements per minute which is very close to the baseline level in Figure 2. Similarly, there are about 45 type 1 LSAs on a given day, or about 1.3 announcements per minute, which is also very close to the observed baseline level of traffic.

Another observation from Figure 2 is that there are obvious periods of increased activity. While it is usually easy to determine that a single router or link is producing the extra announcements, it's difficult to explain why that router or link was causing problems. We will return to this issue later and explore some of the more predominant spikes. For now we will focus on the general observations. Also, we should point out that we are going to minimize our discussion of type 2 LSA announcements for two reasons. First, the number of type 2 announcements is so few that little changes produce big anomalies in the statistical analysis. Second, the results we do see in the type 2 announcements mirror the results from the other two LSA types.

While the baseline level of traffic we see matches our expectations, we want to understand what kind of announcements cause the levels to increase beyond the baseline. To better characterize the source of LSA announcements, we label each LSA with the type of update it represents.

New: An LSA that has not been seen before. These are mostly seen as our monitoring starts, but are also seen as new routers or links are added to the network. This is different from the definition used in RFC 2328 [17] where new is used to denote a new *instance* of an LSA.

Refresh: An LSA that contains no changes. These are used by the protocol to ensure consistency across all the routers.

Down: An LSA that is explicitly being removed from the database. The age of the LSA is set to MAXAGE and announced to all routers, ensuring that this path is quickly removed from the database. It usually represents a failed link.

Up: An LSA that was seen previously but was not currently active. This usually represents a link coming back up after a failure.

Modified: An LSA that has changed a parameter not related to route availability. This usually represents something simple like a metric change.

Timeout: A marker in the data representing an LSA that has timed out in our local database. This LSA is not ever sent on the network, instead it is used to recognize LSAs that timed out before ever being explicitly marked down.

These labels are also used to describe changes in individual links in type 1 LSAs and the set of attached routers in type 2 LSAs. It's important to note that there are two sources of updates that we add into the data. Since LSAs are removed from the database after 60 minutes, we add an explicit update with a *timeout* label into the data. Since normal routers will broadcast the fact that they've timed out an LSA, this update is usually followed by the *down* update produced by another router. In addition, since type 1 link deletions and type 2 router membership deletions are represented by re-announcing the LSA with the removed link or router omitted, we add an explicit entry into the LSA labeled *down* to keep track of these implicit announcements.

Figures 3 and 4 show the year-long announcements broken down by update type. We have omitted most of the update types because they don't produce enough data to show up on the graph. In addition, the type 5 up announcements mirror the type 5 down announcements, so they have been removed for clarity. It is important to first point out that an individual link changing state results in changes for type 5 LSAs at the LSA level, while type 1 LSAs have more than one link per LSA. This is why the type 1 data is dominated by modified announcements with very few down/up

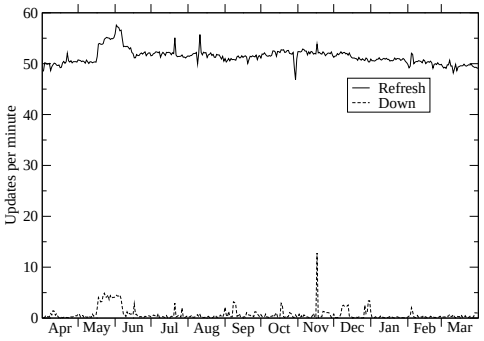

Figure 3: External (Type 5) traffic.

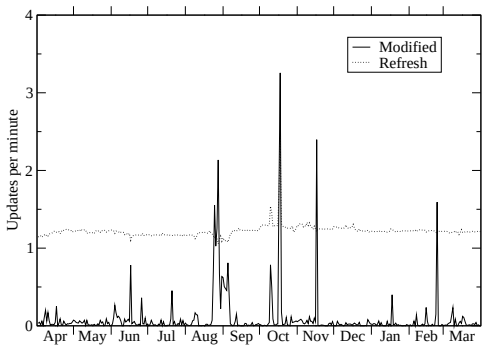

Figure 4: Internal (Type 1) traffic.

announcements. The modified label in this case represents something changing in the LSA; in particular one of the links changing state. The up and down labels on the other hand represent the entire router going up or down.

As we saw before, the baseline amount of refresh traffic represents the stable operation of the network. For the most part, this is the predominant source of announcements. However, for the type 1 announcements, the level of modified events exceeds the level of refreshes on several occasions. For both LSA types however, the predominant periods of instability can be traced back to a single link. Due to the topology of the network there are far more type 5 links than type 1 links. This means that a single type 1 link flapping produces a much higher relative level of traffic. Several of these spikes represent significant periods of instability. For example, during the end of May into the beginning of June we see about ten updates per minute (five updates/minute for both down and up events) from a single customer connection. Not only does the amount of up/down announcements increase during this period, but the refresh announcements are elevated too. As we will show later, this is due to the level of aggregation changing, resulting in more announced routes. Not only is the instability producing more load on the network from down/up events, it also introduces more announced routes, further aggravating the problem.

	Count			
Status	Type 1	Type 1 Link	Type 2	Type 5
New	56	421	38	4763
Down	264	54935	2355	378987
Up	292	55261	2216	377491
Modified	50501	229	1972	20363
Timeout	180	1166	58	7878
Refresh	641328	6643041	219689	26993828

Table 1: Status counts for all events.

While these graphs give a good understanding of the characteristics of the updates over time, we need more detail to better understand the traffic breakdown. In particular, we want to look at some of the numbers that are significant but don't show up on the graphs. Table 1 shows the distribution of LSA updates by type and by status. There are several interesting points in this table that we couldn't see in the graphs. First the number of new events gives an approximate count for the number of entries for each type. For example, there were 56 unique routers on MichNet over the course of the year. Not all were necessarily active at one time. This count does under-represent the number of type 1 links due to the method we use to classify new links. Links that arrive with a new type 1 LSA are classified as new, while never before seen links that are from a known router as classified as up. While we classified 421 links as new, we saw a total of 750 announced links.

Another interesting result is the high level of type 5 modified events. Unlike type 1 LSAs, a modified type 5 LSA doesn't correspond to a state change. Instead it indicates that the parameters for the given prefix have changed. In this case, about 16,000 of the modified events are due to the route tag changing and about 4,000 are due to changes in the metric type bit. The route tag is a 32-bit field used to pass meta data through the OSPF cloud. For example, a router connected to a customer network using BGP might use this field to record the originating AS number. This

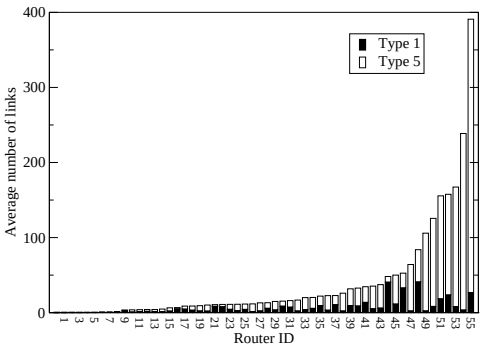

Figure 5: Number of links per router.

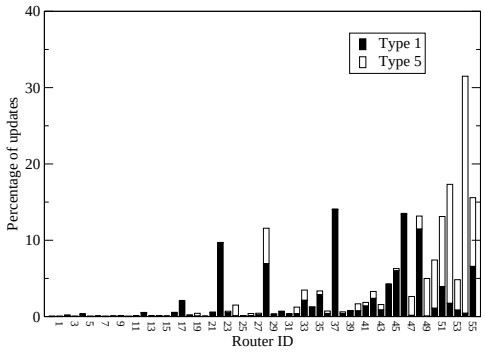

Figure 6: Normalized updates by router.

would allow another properly configured router to decide whether or not to inject the OSPF advertisement back into BGP. We know that this field is used on MichNet to classify specific prefixes, however the large number of changes is surprising. We discuss this issue further in section 4.4. The metric type bit is used to distinguish between external metrics that are comparable to OSPF metrics and external metrics that are considered much larger than OSPF metrics. Since these 4,000 updates occurred without the metric itself changing, the cause for these updates is unclear.

4.2 Sources of instability

The previous section has identified some major instability in the network. This next section attempts to characterize these periods of instability both by their symptoms and by possible causes.

4.2.1 Breakdown by router

One possible source of undesirable behavior is a software or hardware problem in a single router flooding the network with LSA updates. To examine this, we've split updates by router. Figure 5 shows the average number (by day) of type 1 (router) links and type 5 (external) prefixes by advertising router. For simplicity, the routers are ordered by total number of links, and given a unique id. As the graph shows, the distribution of routes is concentrated in a small subset of the routers.

In comparison, figure 6 shows the percentage of updates by type produced by each router, using the same ordering as in figure 5. These numbers only include down, up, new, and modified events, excluding the more benign refresh and timeout events. Also, the results are normalized by type to prevent the type 5 updates from overwhelming the graph. As with the link density, the updates are dominated by a small subset of routers. This skewed distribution is more prominent with the type 5 (external) LSAs than with the type 1 (router) LSAs. For the type 5 LSAs, this skew is somewhat correlated with the number of underlying links. Other than router 28, the top seven routers produce most of the type 5 updates. The type 1 updates are much less correlated with the number of underlying links.

Looking at the top eight sources of external LSA instability we see some interesting trends as well as some interesting anomalies. The most prominent source of this instability is router number 54. This is the source of the large period of instability seen in the latter part of May. As we discussed before, this instability is due to flapping on a single customer link. The second largest source of instability, router number 52, is also the source of the largest source of instability. This single prefix contributes a significant amount of instability even though it is only announced for about a week in December, and has no relation to MichNet. A simple typing mistake caused a prefix for France Telecom to be included in MichNet's configuration. In addition to this simple mistake, router 52 announces other random prefixes. It appears that configuration mistakes and frame relay anomalies cause this router to announce routes to random destinations across MichNet as well as the rest of the Internet. The third router in our list, router number 51, has two prominent sources of instability. The first is a list of 98 dialup addresses that go up and down every time a user connects and disconnects. The other source of instability is a localized spike at the end of December. A small but related set of prefixes experienced a large period of instability for two days and a couple of weeks later completely disappeared from the router. We suspect that this common problem is caused by moving a customer connection to a new physical link. The link exhibits a short period of instability followed by a return to normal behavior. A few

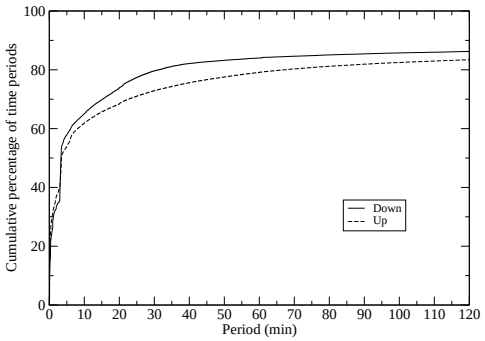

Figure 7: Length of type 5 up and down periods.

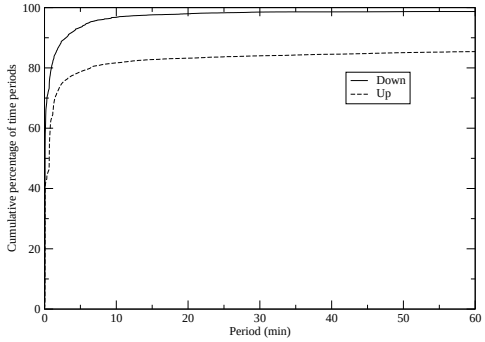

Figure 8: Length of type 1 link up and down periods.

days later, the prefix disappears from the original router and shows up on the new router. The fourth largest source of instability is router number 55. While this router does exhibit two localized periods of increased instability from individual links, the main contributor appears to be a generalized level of instability. There is no apparent correlation between these sources of instability, leading us to believe that the relative level of instability is mainly due to the large number of external prefixes this router injects into OSPF. The fifth router, number 50, shows a long (1.5 months) period of instability for a single customer from May to June. The sixth router, number 49, exhibits a general level of instability from many sources similar to number 55. However, for this router the prefixes concerned appear to be correlated - they're all frame relay links through a single provider. This group of links exhibits short but frequent periods of instability for the first eight months of our analysis. The last four months these links were moved to another router. The seventh largest source of instability, router number 28, is the odd ball in the group. This router exhibits almost no instability on a daily basis. However, for nine hours in the middle of November, every prefix announced by the router experienced an extremely large amount of instability. This is the cause of the large spike in both internal and external LSAs seen in figure 2. This anomaly was detected by the existing network monitoring software, and labeled as a misconfiguration by the operators. Finally, the eighth router, number 53, exhibits one significant localized period of instability with a few other smaller localized periods.

Analysis of these top routers shows that the leading cause of instability is localized failures. Failures due to a single link last anywhere from a couple of hours to months. Complete router failures however seem to be noticed quite quickly and are usually resolved in hours. Moreover, these failures seem to be caused either by network upgrades or by configuration errors. As it turns out, the top two routers are a major source of undesirable behavior, and will continue to come up in our analysis.

4.2.2 Length of instability

We now know that the instability we see on the network is usually due to a single source. The next issue we want to address is the timing of these periods of instability. First we examine how long a particular link stays down or up; then we investigate how long the periods of instability themselves last.

To understand how long links spent in the down and up states, we measured the amount of time each link spent in each state. We then plotted the cumulative distribution function by number of periods over the length of these periods. Figures 7 and 8 show the graphs for the type 5 LSAs and type 1 link announcements respectively. On an ideal network, we would hope that the links would stay down for relatively short periods of time, and up for significantly longer. These graphs show that there are many periods when the prefix or link was only up or down for a very short period of time: 53% of the type 5 periods lasted less than five minutes, while 75% of the type 1 link periods lasted less than 2.5 minutes. While the low periods for the down events might suggest stability, this concentration of very short periods for both down and up events indicates the links are flapping, or oscillating quickly between the down and up states. This behavior is definitely undesirable. On the other hand, the distribution is heavy tailed. About 1%, or 3790, of the type 5 up periods and 2%, or 1098, of the type 1 up periods lasted longer than thirty days. On the other hand, about 3020 or 0.8% of the type five down periods and 211 or 0.5% of the type 1 down periods lasted longer than 30 days. One important consideration is that we are looking at the distribution by period count, not by length. In other

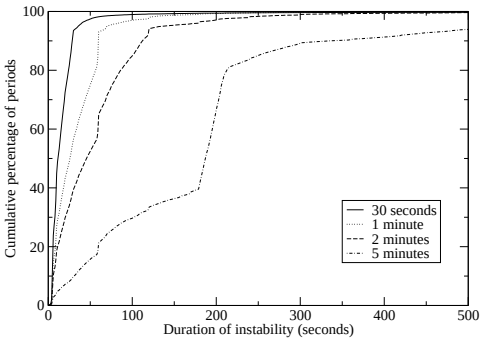

Figure 9: Type 5 periods of instability.

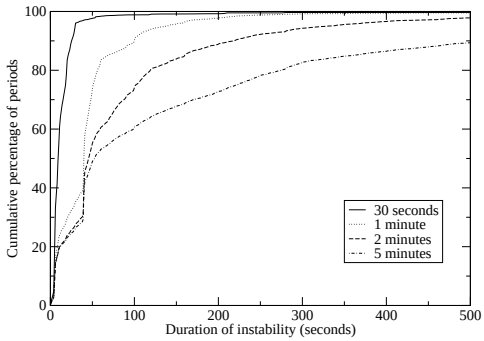

Figure 10: Type 1 Link periods of instability.

words, a three month period is counted the same as a five minute period. So, while this data shows significant amounts of traffic in the network, individual links are predominantly stable.

Looking at the amount of time each link stayed up or down gives us a idea of how much flapping is seen on the network. We also want to understand how long these periods of instability last. The next set of graphs show the length of these periods of instability. We first collect all periods less than a given threshold. We then string adjacent periods together to determine how long the instability lasted. Figure 9 shows the cumulative distribution of the periods of instability for different thresholds. We strung together sequences of up/down events that were 30 seconds, 1, 2, and 5 minutes apart. The result is that most of the periods of instability are relatively short; 80% of the five minute sequences last for less than two minutes. More significant are the top few that last up to 5.5 days. The most prominent of these is the France Telecom link we discussed earlier which continuously changes state at least once every five minutes for 5.5 days. In fact, this is the largest single source of instability on the network. As we'll discuss later, this is indicative of a larger problem caused by mistyped prefixes causing single routers to inject random prefixes into OSPF for short periods of time. The second longest period of instability is a collection of routes that flap every five minutes for 2.5 days. Unlike the previous period, this one appears to be composed of about ten legitimate customer prefixes. They do however, exhibit the behavior we discussed before of a change in aggregation level. During the instability we see more prefixes announced for the same address ranges than we see during stable periods, most likely due to errors in BGP.

The data for the type 1 link information, shown in figure 10 is less disturbing. Once again, we see about half of the periods of instability last for a very short period, about one minute. The rest tail off up to about 5.8 hours. If we extend the periods we monitor to 20 minutes, the longest period goes up to 26 hours. This is still a significant period of time for a single link to flap. Since these links were constantly changing state rather than staying down over these periods, we expect them to appear more like congested links. This makes them much harder to detect using conventional tools. On the other hand, these types of faults are the easiest to detect using routing protocol analysis since they produce an abnormal number of updates in a short period of time.

4.2.3 Frequency components

Previous work with other routing protocols found temporal effects such as increased routing protocol traffic during the middle of the local day due to the increased traffic on the network at that time [15]. Other than one specific case shown in section 4.4, our results mirror previous studies that haven't found any significant correlation between the time of day, day of week, month of year and the amount of OSPF updates observed [14]. We did however notice some interesting trends in the inter-arrival time between updates.

The most expected source of frequency components in the data comes from refresh updates. In stable state, each LSA should be re-announced every thirty minutes. So, we would expect a significant percentage of the refresh events to arrive 30 minutes after the previous announcement. While figure 11 does show about 10% of the refresh events right at 30 minutes, the rest arrive much closer to 34 minutes. This is due to a four minute hold time used by Cisco routers to consolidate refreshes. There is a small amount of jitter added to this hold time, producing an even distribution around 34 minutes. As we saw with the instability graphs, we see a significant number of up and down events with a relatively low inter-arrival time. We also notice that a majority of the modified events have a short inter-arrival time. The predominant source of these modified events are the route_tag flapping of a single external route which we discuss

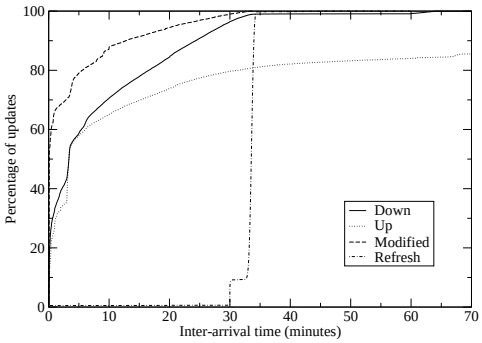

Figure 11: Type 5 LSA inter-arrival time CDF.

in section 4.4. Other than the refresh traffic, we don't observe any temporal clustering in the OSPF data either from time-of-day effects or from synchronization.

4.3 Routing Topology

While previous sections have tried to characterize the type and distribution of the LSA updates, this section attempts to characterize the impact these periods of instability have on the network. The most obvious impact on the network is changes in the topology of the network. This can cause packets to be lost or delivered out of order. Also, changes in the routing topology correspond to each router running the shortest path algorithm. This is a relatively expensive operation, producing extra load on the router and causing a delay in the convergence of the network. In addition to collecting the raw OSPF data, we also simulated the shortest path calculation performed by each router. This analysis uncovered two important results. First, the routing topology was constantly changing over time, with specific configurations lasting shorter than 28 days. Additionally, the changes in the topology were incremental, only requiring slight changes in the shortest path tree.

We first studied how the routing topology changes over time. To do this, we assigned an identifier to each unique shortest path tree that we saw. The results are shown in figure 12. For each shortest path calculation, we plotted the unique id at the corresponding time. So, for example, if a single link is flapping, we would expect to see the points on the graph oscillate between two different values over the period of instability. Because the shortest path trees are labeled increasingly over time, the y-axis also serves as a rough time measurement. Shortest path trees with lower valued ids were first seen earlier.

Looking at this data we see some surprising results. First, the overall trend is constantly increasing. In other words, the topology of the network is always changing and doesn't appear to focus close to a single configuration for long periods of time. We expect the network to constantly evolve as customers are added and links are upgraded. However, this is a surprising amount of change in the network topology, especially considering that we only looked at links between the core OSPF routers. This excludes any changes due to links to customer networks. We saw 1,852 unique shortest path trees from the perspective of a single router over the course of a year. This is an average of about five unique configurations per day. Close examination of the graph shows that there are periods when the topology oscillates between two different configurations. This reflects the instability produced by a single link flapping. However, the topology continues to change over time, with the longest duration of a single configuration lasting just 27.25 days.

We also produced similar graphs from the perspective of several other routers with very similar results (Figures 14 and 16). These routers, which are further from the topological center of the network did exhibit several instances of reverting back to a much earlier tree. However, these trees represented failure conditions and consisted of either the router by itself, or a cluster of routers connected by a single link to the rest of the network. Router number 45 on the other hand, has many physical links to other parts of the network making it much less likely for it to experience these kinds of partitions.

With a better understanding of the changes in the topology over time, we wanted to investigate the incremental changes in topology. Figure 13 shows a graph of the difference between two adjacent shortest path trees. The metric used to calculate this difference is a simple count of the number of changed links. This was done by subtracting the

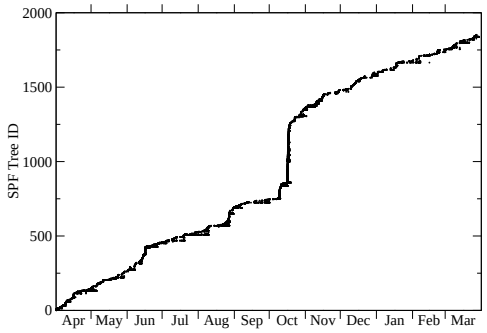

Figure 12: Configuration changes in the shortest path tree from router 45.

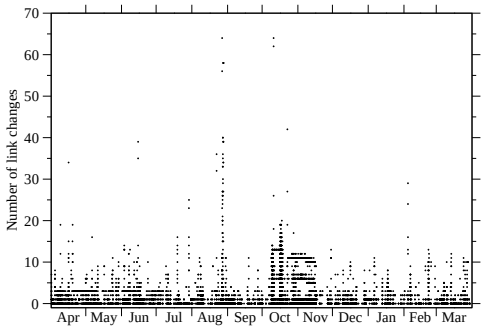

Figure 13: Shortest path tree changes from router 45.

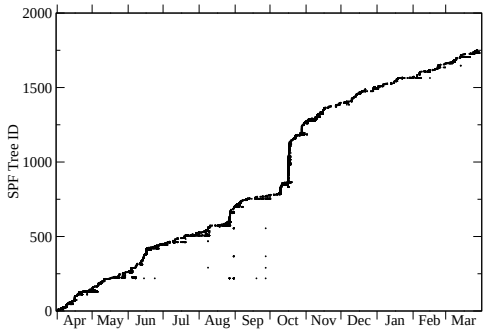

Figure 14: Configuration changes in the shortest path tree from router 51.

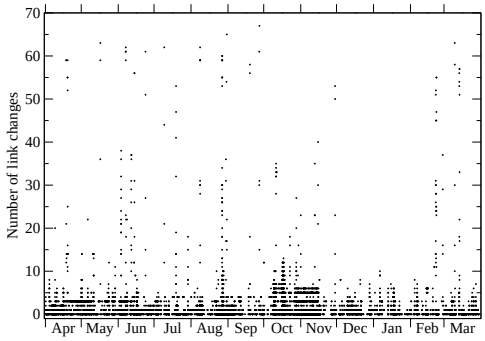

Figure 15: Shortest path tree changes from router 51.

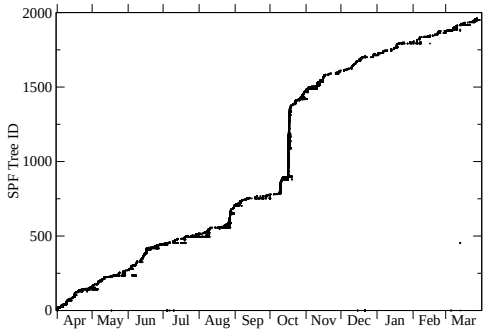

Figure 16: Configuration changes in the shortest path tree from router 24.

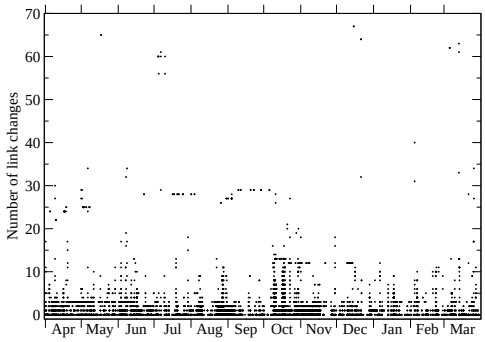

Figure 17: Shortest path tree changes from router 24.

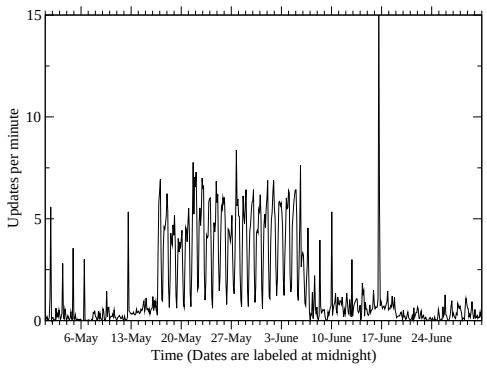

Figure 18: Instability from a single customer connection.

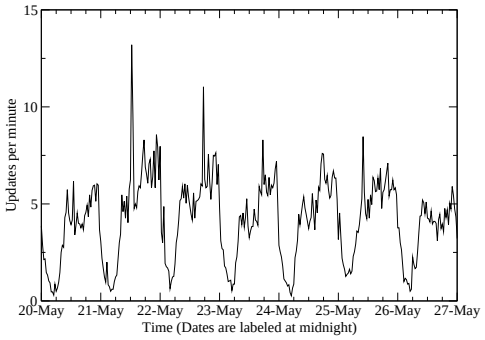

Figure 19: Periodic instability.

adjacency matrices and taking the sum of the resulting values in the matrix.[2] This should accurately reflect small changes in the network that cause a subset of the shortest path tree to shift. The most significant observation from this graph is that the overwhelming majority of the changes to the tree are very small. In fact, 65% of the shortest path calculations produce the same tree as the previous calculation. Note, in order to highlight this, the zero line has been shifted above the baseline. This is a strong argument for the use of an incremental shortest path first algorithm [2; 1].

Figures 15 and 17 show similar graphs for routers 51 and 24. As before, the results are quite similar to those we saw from router 45. The main difference is the presence of more outliers in the middle to upper range. This is due to the partitioning problem described before. When these routers are partitioned into small clusters separate from the main network, returning to the rest of the network causes a new tree with many link changes.

These results highlight our observations from examining the LSA data. For the most part the instability in the network is confined to a small number of links. These links produce a large number of updates, but cause few actual changes in the topology of the network. In addition, the network exhibits a steady change over time due to physical changes in the topology.

4.4 Detailed analysis

The previous sections explored the aggregate, coarse view of the network over the period of a year. From this perspective, we can see that there are periods of increased instability, but we have little understanding of the specifics of these anomalies. In addition, the aggregate view can completely miss significant anomalies that play an important role in the day to day performance of the network. This section focuses on three such anomalies. The first highlights a significant period of flapping caused by a single physical link. The second anomaly involves a single router sporadically injecting random routes into OSPF. Finally, the third anomaly highlights very low frequency flapping that lasted an entire year. These specific incidents highlight the general lack of comprehensive network monitoring tools available to network operators as well as the amount of instability external routes can introduce into OSPF.

4.4.1 Periodic Instability

The most noticeable period of instability can be seen in figure 3 during the end of May and beginning of June. Looking closer at the instability in figure 18 we can see that there are significantly more updates during this period than normal. In addition, the updates appear to be periodic. As you can see in figure 19, this instability corresponds with the normal human waking cycles for the local time zone. While this kind of instability has been seen in BGP behavior before [15], this is the only instance we've found in the OSPF data. Talking to the network operations staff, the source of this failure was indeed BGP. During this time period, the particular customer was upgrading their physical connection to MichNet. The prefixes for this customer are maintained using a BGP connection and are then injected into OSPF. However, there are no filters placed on this BGP data, so when BGP failed, all of the instability was directly injected into OSPF.

Despite the fact that this data appears to be caused by a single link, we see a large number of prefixes related to this instability, all allocated to the same customer. Figure 20 shows the number of prefixes announced by the MichNet

[2] While it is possible to have redundant links between two routers, we ignore these extra links and construct a boolean adjacency matrix

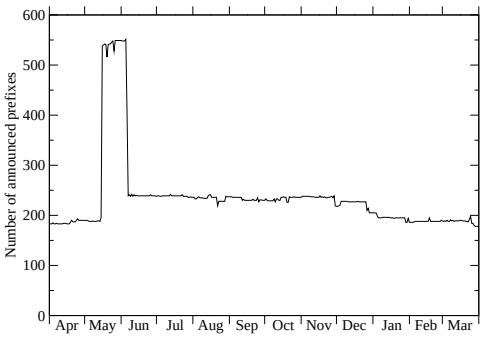

Figure 20: Number of routes announced by router 54.

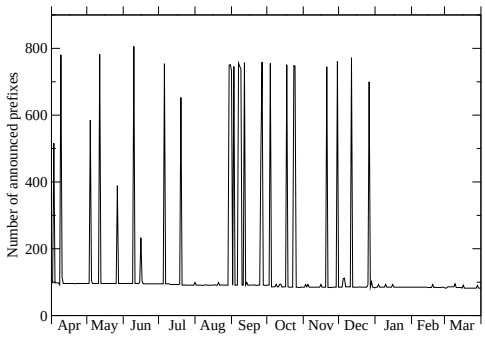

Figure 21: Number of routes announced by router 53.

router connected to this customer over the entire year. Surprisingly the number of prefixes triples during this period of instability. It appears that these extra prefixes are due to a change in the level of aggregation. Rather than aggregating several prefixes into one shorter prefix before injecting them into OSPF, the router was announcing the longer prefixes directly into OSPF. This has serious implications about the stability of the network. Not only can the number of updates increase during periods of instability, but the number of LSAs can increase due to aggregation failure. As with the periodicity, this instability is most likely caused by BGP failures being directly injected into OSPF without any filtering.

While we are unable to give a detailed explanation for the cause of this instability, it does raise several important points. First is the need for different network monitoring tools that are capable of monitoring routing protocols. While we know that existing network monitoring tools don't provide complete coverage of network outages, we were extremely surprised to find this much instability in the network lasting this long. Second, this incident highlights how much impact routes injected from other protocols can have on OSPF.

4.4.2 Aggregation Breakdown

The previous anomaly leads to another interesting observation. The router producing the data from figure 20 is router number 54 in figure 5. As we saw in figure 20, the number of routes announced by router 54 is normally under 250, except during the previous period of instability. If we counted the total number of announced prefixes, we would include a large number of incorrect routes. Router 53 from figure 5 shows even more variability as seen in figure 21. Unlike the previous case, the increased number of routes for this router never persists for longer than three days. In addition, the increased levels are not localized to a single period. Looking at the *trouble ticket system* for MichNet, these increased routes correspond to recorded problems with this router. In addition, rather than adding subsets of existing routes, this router announces additional prefixes from different parts of the network. For example, announcing prefixes assigned to a customer from the western side of the state when the router is on the eastern side. In addition, as is the case with the France Telecom route, some of these routes appear to be completely unrelated to MichNet. Talking to the operations staff, it appears that all of these additional prefixes are caused by configuration errors. The routes from across the state are topologically neighbors to the announcing routers due to ATM links that span the state. In addition, the France Telecom link has prefix very similar to ones assigned to MichNet. It appears that a simple typing mistake caused this prefix to be injected into OSPF without any real connection to an external protocol. While it is entirely possible that these extra routes would cause some of the problems seen in the trouble tickets, we're not able to definitively determine if the extra routes are the root cause or merely indicators of general configuration errors. It is clear however, that monitoring the routing protocols produces much faster detection of these types of configuration errors.

4.4.3 Route Tag Flapping

The third anomaly that we found was again caused by a single link. Figure 22 shows the number of updates caused by the route tag field changing on a single type 5 LSA. The route tag is an extra field in type 5 LSAs used to transport arbitrary data across the OSPF network. One possible use is to record the originating AS number of the BGP source. In fact, this particular LSA switches between a route tag field that is empty, and the number 237, which is MichNet's

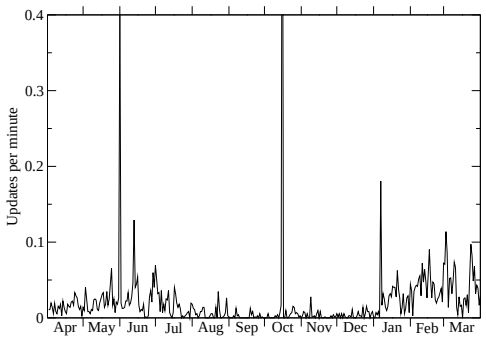

Figure 22: Route tag flap.

AS number. Unlike the first anomaly, this one lasted the entire year. We know that the MichNet operators use this field to tag routes for specific actions when exporting them using BGP. However, the frequency at which this particular LSA changes leads us to suspect that this anomaly represents a more fundamental flaw in the interaction between the external routing protocol and OSPF. One possible explanation is that this route is being learned from two different sources, such as BGP and a statically configured route. Based on some fluctuations in one or both of these external sources, one is chosen over the other producing only a change in the route tag field.

5 Related Work

Our work on understanding the performance of OSPF on a production network is part of a larger body of work that has attempted to understand the Internet infrastructure from its infancy. Most of the study of production networks has focused on the inter-domain behavior of the Internet. For example, it was found that the choice of TCP for the transport layer of BGP KeepAlive messages leads to route storms under heavy load [15]. In addition, further research showed that convergence times for Internet routes can be in the tens of minutes [13; 16]. Another study combined routing analysis and end-to-end traffic characterization to understand how inter-domain routing affects end users [20]. Previous work has also focused on understanding the origin of network instability both in the Internet and within an ISP [14].

Like this previous work, our work has attempted to provide a comprehensive view of routing protocol behavior on a production network. Our work differs in that we focus on the intra-domain routing of a regional ISP. Lots of work has been done to understand the theoretical behavior of intra-domain routing protocols both through simulation and experimentation on small testbeds. Some of these studies have focused on understanding the factors affecting convergence times in an effort to improve the performance of the network [2; 22]. Other research has simulated these protocols attempting to identify issues with stability [24; 3]. This work is important to understanding the behavior of routing protocols, but study of these protocols on production networks is needed to completely understand their behavior.

Several studies have attempted to extend this theoretical work by studying these protocols on production networks. An analysis of Qwest's production IS-IS network showed that a small number of individual links produce the majority of the instability. In addition, this work makes the case for using an incremental shortest path algorithm to decrease the load on the routers and to improve convergence times. Our work has confirmed the lopsided source of instability over a longer term analysis. In addition we have produced operational measurements that confirm the case for an incremental shortest path algorithm. An analysis of Sprint's IS-IS backbone focused on understanding the behavior of link failures. In addition, they introduced failures into the network to study convergence times. Finally, a third, soon to be published, study analyzed the behavior of OSPF on an enterprise network [23]. This work studied the temporal behavior and source of link state advertisements. Like our work, they analyze the behavior from different perspectives and find similar results. Our work extends this collection of past and concurrent work by providing a more comprehensive analysis. We analyzed a year's worth of OSPF data both by looking at behavior of the LSAs in the network and by looking at the changes these updates have on the routing topology. In addition, we performed detailed analysis of several of the anomalies we observed. This analysis showed a new perspective on the source of these anomalies, showing previously undiscovered behavior.

Our work has also shown the need for better network management tools. For the most part these tools ignore information available from routing protocols, although newer tools are emerging to leverage this information. Researchers at AT&T have begun to develop NetScope, a set of tools for managing large IP networks [5]. To develop these tools they have worked on methods for determining traffic demands [6], determining the network configuration [7], and optimizing OSPF weights [8]. They have worked with the group running AT&T's central backbone to develop better tools for controlling routing on the network as well as developing automation to perform some of these tasks. Their work started with the large goal of automating control of routing policies. They quickly discovered that the resources needed to perform these tasks were unavailable.

Commercial tools are also starting to use information from routing protocols to provide a better view of the network. Tools such as Arbor Network's Peakflow Traffic [19] and Ixia's IxTraffic [12] combine other network monitoring tools with BGP monitoring to provide a more comprehensive view of the network. Packet Design's Route Explorer [4] collects OSPF or IS-IS data to provide visualization of problems in real time, logging to replay events for analysis, and modeling to allow operators to analyze the effect of various changes on the network.

6 Conclusion

This paper provides a comprehensive analysis of an operational OSPF network over the period of a year. We looked at the overall traffic levels, characterized the instability in the network, and analyzed the changes in routing topology over time. In addition we provided detailed analysis of the largest periods of instability we saw on the network. We found that routes injected from external routing protocols were the predominant source of instability within OSPF. In addition we found that these periods of instability often went undiagnosed due to the lack of routing protocol information available to network operators. Finally, we showed that the routing topology is constantly changing over time. The individual changes however, are localized necessitating only minor changes in the shortest path tree at each router.

One theme that continues to arise in the process of investigating the performance of production networks, is that the tools available to network managers are extremely limited. We have highlighted three sources of strange behavior that we suspect could be easily fixed with configuration changes. Network operators rarely have tools that present this information in a usable manner. Additionally, configuration interfaces are often presented at such a low level that operators are reluctant to make minor tweaks for fear of producing disastrous results. Some of our observations from OSPF were compared to two existing tools currently used to manage MichNet. The first tool, Rover [21], is configured to ping the network interfaces of specific routers. The second tool, Spectrum [25], uses SNMP queries to the routers to determine the status of the interfaces. Unfortunately, the combination of these tools still misses link failures that prompt customers to complain about problems with their Internet connection. While we were able to correlate some of the OSPF events with the existing tools, we were surprised to find that OSPF did not detect all the faults found by the other tools. The predominant source of instability in MichNet is related to customer networks. However, the information from customer networks is often aggregated before it is injected in OSPF. This can cause a failed connection to one customer to not show up in OSPF. We also found a lot of faults using OSPF that didn't show up in the other tools. For the most part, the core OSPF network is stable, and what individual link failures we do see often have alternate paths that provide backup routing. Finally, OSPF sees very high resolution information due to the quick propagation of detected failures throughout the network. Tools such and Rover and Spectrum rely on periodic polling which can miss some high frequency changes. While this high resolution data makes it difficult to use raw OSPF information in an operations center, we think that the anomalies we highlight demonstrate that OSPF data would make a valuable addition to the set of tools available to network operators.

We also plan to do further research into the interaction between OSPF and other routing protocols. These routing protocols often have undesirable characteristics that can negatively impact the performance of OSPF. Limiting their impact would decrease the load on the network, providing better performance for the network as a whole. Unfortunately, the links to customer networks are one of the largest sources of administrative headaches. Using dynamic routing protocols greatly simplifies the configuration of both the ISP and customer routers. While it might be possible to minimize the impact of these injected routes, administrative issues make current solutions undesirable.

We are also interested in exploring the impact that OSPF has on BGP announcements made to the rest of the Internet. While most of the information injected into BGP from OSPF is aggregated into shorter prefixes, preliminary analysis has shown that instability in OSPF can transition into BGP. This has significant implications as BGP routing updates have a broad impact on the global Internet.

7 Acknowledgements

We wish to thank Michael Bailey, Eric Sobocinski, Jeff Ogden, Russell Dwarshuis, and Brian Cashman, as well as Fred Rowe, Bert Rossi, and the the rest of the MichNet operations staff for their support and helpful insight. We also thank the anonymous referees that reviewed the conference version of this paper for their feedback and constructive criticism.

References

[1] Cengiz Alaettinoglu and Steve Casner. A detailed analysis of ISIS routing protocol behavior. NANOG Presentation, February 2002.

[2] Cengiz Alaettinoglu, Van Jacobson, and Haobo Yu. Towards millisecond IP convergence. NANOG Presentation, October 2000.

[3] Anindya Basu and Jon Riecke. Stability issues in OSPF routing. In *Proceedings of the ACM SIGCOMM 2001 Conference (SIGCOMM-01)*, New York, August 27–31 2001.

[4] Packet Design CNS. Route explorer. http://www.route-explorer.com/.

[5] Anja Feldmann, Albert Greenberg, Carsten Lund, Nick Reingold, and Jennifer Rexford. NetScope: Traffic engineering for IP networks. *IEEE Network Magazine, special issue on Internet traffic engineering*, pages 11–19, March/April 2000.

[6] Anja Feldmann, Albert Greenberg, Carsten Lund, Nick Reingold, Jennifer Rexford, and Fred True. Deriving traffic demands for operational IP networks: Methodology and experience. In *Proceedings of the ACM SIGCOMM*, August/September 2000.

[7] Anja Feldmann and Jennifer Rexford. IP network configuration for traffic engineering. Technical Report 000526-02, AT&T Research, May 2000.

[8] Bernard Fortz and Mikkel Thorup. Internet traffic engineering by optimizing OSPF weights. In *IEEE INFOCOM*, March 2000.

[9] Ramesh Govindan and Anoop Reddy. An analysis of inter-domain topology and route stability. In *Proceedings of the IEEE INFOCOM '97*, Kobe, Japan, April 1997.

[10] Gianluca Iannaccone, C. Chuah, Richard Mortier, Supratik Bhattacharyya, and Christophe Diot. Analysis of link failures in an IP backbone. In *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, November 2002.

[11] Internet Performance Measurement and Analysis (IPMA) project. http://www.merit.edu/ipma/.

[12] Ixia. Ixtraffic. http://www.ixiacom.com.

[13] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed internet routing convergence. In *Proceedings of ACM SIGCOMM*, 2000.

[14] Craig Labovitz, Abha Ahuja, and Farnam Jahanian. Experimental study of internet stability and wide-area network failures. In *Proceedings of FTCS99*, June 1999.

[15] Craig Labovitz, G. Robert Malan, and Farnam Jahanian. Internet routing instability. *IEEE/ACM Transactions on Networking*, 6(5):515–528, October 1998.

[16] Craig Labovitz, Roger Wattenhofer, Srinivasan Venkatachary, and Abha Ahuja. The impact of internet policy and topology on delayed routing convergence. Technical Report MSR-TR-2000-74, Microsoft Research, 2000.

[17] J. Moy. OSPF version 2. RFC 2328, IETF, April 1998.

[18] John T. Moy. *OSPF: Anatomy of an Internet Routing Protocol*. Addison Wesley, 1998.

[19] Arbor Networks. Peakflow traffic. http://arbornetworks.com.

[20] Vern Paxson. End-to-end routing behavior in the Internet. *IEEE/ACM Transactions on Networking*, 5(5):601–615, October 1997.

[21] Internet Rover. http://www.merit.edu/merit/archive/rover/.

[22] Aman Shaikh and Albert Greenberg. Experience in Black-Box OSPF measurement. In *Proceedings of the First ACM SIGCOMM Internet Measurement Workshop (IMW-01)*, pages 113–126, New York, November 1–2 2001. ACM Press.

[23] Aman Shaikh, Chris Isett, Albert Greenberg, Matthew Roughan, and Joel Gottlieb. A case study in OSPF behavior in a large enterprise network. In *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, November 2002.

[24] Aman Shaikh, Lampros Kalampoukas, Rohit Dube, and Anujan Varma. Routing stability in congested networks: Experimentation and analysis. In *Proceedings of the ACM SIGCOMM 2000 Conference*, pages 163–174, August 2000.

[25] Aprisma Management Technologies. Spectrum. http://www.aprisma.com/spectrum/.

[26] Zebra. http://www.zebra.org/.