Analyzing BGP ASPATH Behavior in the Internet

Dionysus Blazakis
Institute for Systems Research
University of Maryland
College Park, Maryland 20742
Email: dblaze@isr.umd.edu

Manish Karir
Networking Research and Development
Merit Network Inc.
Ann Arbor, Michigan 48104
Email: mkarir@merit.edu

John S. Baras
Institute for Systems Research
University of Maryland
College Park, Maryland 20742
Email: baras@isr.umd.edu

***Abstract*— In this paper we introduce a new metric for analyzing the behavior of ASPATH values in the Border Gateway Protocol (BGP) routing protocol. We base our metric on the edit distance algorithm, an algorithm used for approximate string matching. We modify this basic algorithm by adding features that embed BGP domain knowledge. This allows us to perform meaningful comparisons of ASPATH values contained in BGP update messages. We call our modified metric ASPATH Edit Distance(AED). We illustrate the application of this metric to characterize ASPATH changes at a global scale using the example of a major Internet routing anomaly. At the other end of the spectrum we illustrate how this metric can be used to quantify and model the behavior of ASPATH values for individual Autonomous Systems. AED provides us with an important measure with which we can study the behavior of ASPATHS in the Internet. With sufficient refinement, AED can be suitably adapted and used alongside other metrics in BGP routing anomaly detection algorithms and tools.**

I. Introduction

The BGP protocol lies at the heart of the Internet. It is responsible for interconnecting individual network entities called Autonomous Systems (AS) into the global Internet. The Internet is now an integral part of everyday life, and it is important to study its resilience, stability, and performance in order to better understand current limitations and its future growth. One of the primary functions of BGP is to propagate reachability information regarding various networks. This information is carried in updates messages that are exchanged between BGP peers. One of the key pieces of information is the ASPATH field which describes the complete set of AS numbers that must be passed in order to reach any particular network.

Several studies have attempted to study the dynamics of BGP routing in the Internet. However, most of these have focused on studying the *temporal* behavior of BGP update messages. The primary metric used in most similar studies has been either the number of messages over various intervals of time or time for BGP to adapt to changes and converge. Below, we provide a brief summary of the most directly relevant previous work in this field as well as how the AED metric can add yet another complementary component those earlier contributions.

The seminal work by Labovitz et. al. [1][2] addressed the issue of instability in the BGP routing protocol. They presented various measurements to illustrate this instability and attempted to determine its root causes. In an effort to quantify instability they described several ASPATH transition events that they observed in their measurements. For example a BGP update message that re-announces a previously announced prefix but with a different ASPATH is considered to have implicitly withdrawn the earlier path. This contributes to the instability of that prefix. In [3] Rexford et. al. use a modified metric that attempts to collapse update messages that occur close together in time into a single event in order to study the relative stability of popular prefixes. They argue that based on their analysis, despite the large number of update messages, popular prefixes tend to have stable BGP paths for days or weeks at a time. [4] also takes a similar approach in attempting to reduce the overwhelmingly large amount of BGP update information into a manageable set. In [5] the authors analyze the surge in BGP the rate of update messages during the Slammer Worm event of January 2003.

Our contribution in this paper is to introduce a new metric that is complementary to these studies. AED can be used to analyze changes or variations in ASPATH values in BGP update messages. This can in turn enhance our understanding of stability and help in the task of identifying important features from large amounts of BGP routing information. AED can allow us to make *quantitative* statements regarding observed BGP ASPATH's, i.e. how much does an ASPATH change.

This can in turn be used in automated anomalous route detection algorithms and tools. Using AED we can study ASPATH variations which provides us with addition insight into BGP.

In this paper we first describe the basic concept of edit-distance upon which we construct a new metric called ASPATH Edit Distance (AED) for studying the dynamics of the ASPATH values represented in BGP update messages. We then describe some features of AED using simple examples to illustrate how this metric behaves at both the coarse global level, as well as at the level of individual prefixes.

The rest of this paper is organized as follows: In section II we define and construct the ASPATH metric; Section III provides some results based on BGP update message measurements that illustrate how the AED metric can be used to make both global as well as specific ASPATH comparisons; Section IV provides a discussion regarding how AED can be further enhanced beyond the preliminary work described in this paper and how it can be used in BGP anomaly detection algorithms and tools. Section V presents our conclusions and outlines some future work.

II. AED: An ASPATH Change Metric

A. Edit Distance/Levenshtein Distance

Levenshtein Distance [6][7] is a measure of the similarity or difference between two strings. Levenshtein distance between two strings can be computed by determining the number of operations (change, add, or delete) that are needed to convert one string into another. The larger the number the more different the strings are. For example if we consider 2 strings:

```
s="He hit the ball with a bat"
t="He hit the ball with a cat"
```

The distance between strings *s*, and *t* is 1. In order to convert string *s* into string *t* we need 1 operation, i.e. to change the ’b’ character of ’bat’ to a ’c’. The Levenshtein distance metric has been used in various applications such as speech recognition, DNA analysis, as well as several others. Levenshtein distance is also referred to by the term *edit distance*, and for the rest of this paper we will use that term.

B. ASPATH Edit Distance - AED

The ASPATH contained within BGP update messages is a list of AS numbers which describes that path that must be followed across multiple Autonomous Systems in order to reach a given network or prefix. Changing network conditions can cause an ASPATH to change. In order to understand instability in the Internet it is important to not only understand the number of ASPATH changes that occur, but able be able to quantify the relative magnitude of the changes.

As Edit Distance is able to quantify the similarity or the difference between 2 strings, it represents a good starting point to study ASPATH changes. However, we need to modify the algorithm slightly for our specific application domain. In particular ASPATHs are composed of strings of AS numbers and therefore cannot be directly treated as character strings. For example if we consider ASPATH strings *s*, and *t*:

```
s = "7018 174 237"
t = "7018 212 237"
```

A direct application of the Edit Distance algorithm would give us:

```
ED(s,t) = 3
```

The characters ”174” in *s* are replaced by ”212” to form string *t*. However, this is not an accurate representation of our intuitive understanding of the difference between *s* and *t*. From the BGP perspective, the difference between ASPATH *s*, and *t* is that AS number 174 is replaced by AS number 212, a single change.

In order to embed BGP domain knowledge into the concept of Edit Distance, we make the following two changes to the basic algorithm. First, we modify the Edit Distance comparison algorithm to compare AS numbers within the ASPATH strings as entities instead of as characters. The second modification involves the concept of Origin AS. The last AS number in the ASPATH represents the Origin AS number. If the same prefix is seen advertised in BGP update messages with two different Origin AS number, it represents a significant event(ignoring multiple origin AS advertisements for now, we address this further in the discussion in section IV). This scenario is illustrated in the following example:

```
s="7018 174 237"
t="7018 174 65005"
```

A simple Edit Distance algorithm would give us a metric of 1. However, in order to completely capture the importance of this event in our metric, we assign a large value(MAX_EDIT_DISTANCE) when a change in the Origin AS number is seen between two ASPATHs being compared. We refer to our modified Edit Distance algorithm as ASPATH Edit Distance(AED). In the following section we describe some simple experiments that

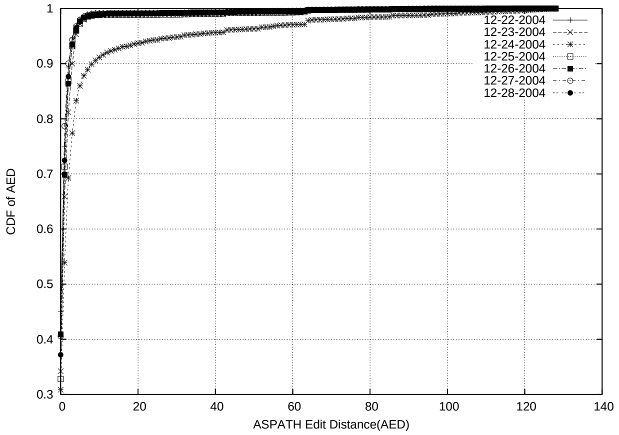

Fig. 1. Cumulative Distribution Function of average AED

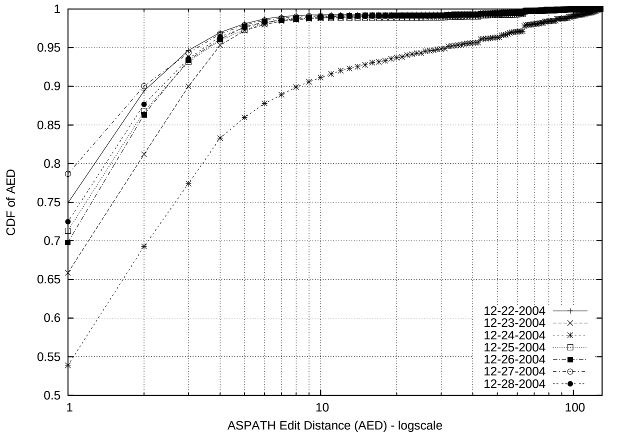

Fig. 2. Cumulative Distribution Function of average AED (logscale)

illustrate the behavior of AED in a variety of different situations.

III. Experimental Results and Analysis

A. Characterizing Global AED

In our first experiment we attempt to characterize how AED behaves across all prefixes that are advertised in BGP update messages in the Internet. What we aim to determine with this experiment is to obtain an estimate of typical values of AED that occur in the Internet across over 160K prefixes in the global BGP routing table. For this experiment we compute the average AED for each prefix over a time period of one day and then study the distribution of these values across several days.

The input dataset we use is the 1 week time period from December 21, 2004 through December 28, 2004. The reason for choosing this time period is that this dataset includes data from the December 24, 2004 route leakage incident [8]. Our dataset is obtained from the Routeviews [9] BGP data repository, We limit ourselves only to examining data as seen from the routeviews peering session with AT&T. Not only does this experiment illustrate what the distribution of typical values of AED are at a coarse global level, but also shows whether the effects of a significant route leakage incident can be captured by this metric.

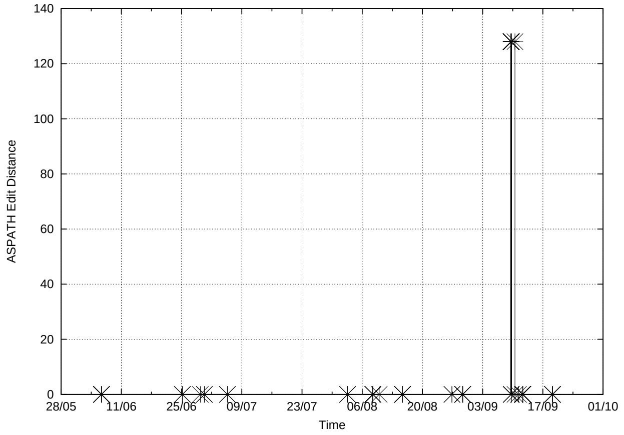

Fig. 3. ASPATH Edit Distance for 12/8 prefix from June-Sept 2005

Figure 1 shows the cumulative distribution function (cdf) of the number of prefixes with a certain average daily AED value. Each line depicts the cdf computed for one day over the one week dataset. The figure shows that the average AED metric across all prefixes is fairly consistent on most days. The exception to the above is the data for December 24th. Here we see a dramatic variation when compared with measurements on the other days. This illustrates the utility of the AED metric as a way to quantify the impact of global events on BGP routing stability. Figure 2 shows the same data as Figure 1 but uses logscale on the x-axis to amplify the differences between the different days. This figure clearly shows that on days other than December 24 2004, roughly 95 percent of all prefixes in the Internet, exhibit an average AED of less than or equal to 4. On December 24 2004, however, only 83 percent of all prefixes exhibit this behavior.

B. Origin AS Anomalies

An origin AS anomaly can be defined as an event where a prefix which has been advertised previously with a certain Origin AS is suddenly advertised with a different Origin AS. Using the AED metric this event would result in a sudden change in the AED value and would therefore be easily identified. Figure 3 and Figure 4 show two recent examples of such events.

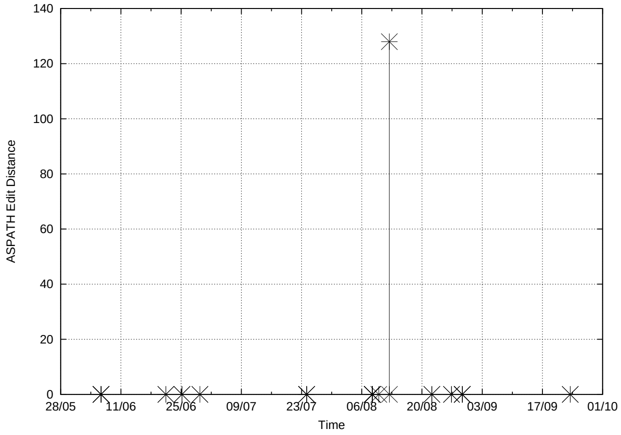

Fig. 4. ASPATH Edit Distance for 4/8 prefix from June-Sept 2005

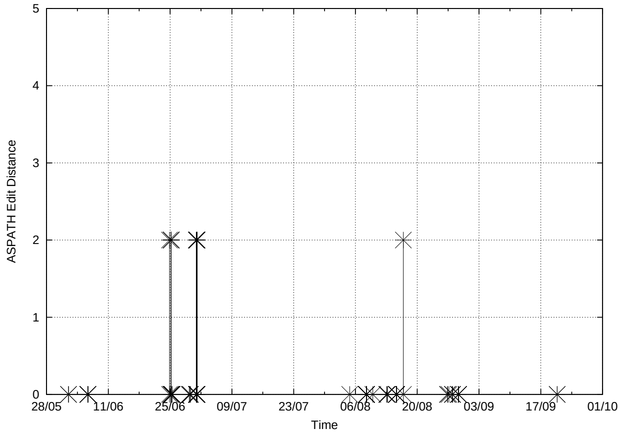

Fig. 5. ASPATH Edit Distance for 35/8 prefix from June-Sept 2005

Figure 3 shows the ASPATH Edit Distance values for the 12/8 prefix. We analyzed a Routeviews dataset the four month period from June till September 2005. We limited ourselves to only examining the subset of data as observed via the Sprint-Routeviews peering session. The first occurrence of the ASPATH was recorded as the ”base” value, and then for each subsequent update message that announced the 12/8 prefix, we compute the AED. This value is then plotted versus the time at which it occurred. As expected, the 12/8 prefix is extremely stable and the computed AED is zero most of the time. The commonly seen ASPATH for this prefix is ”1239 7018”. However, on September 9th 2005, AS 26210 is seen to originate this prefix instead of 7018. The new observed ASPATH is ”1239 12956 26210”. This change results in a sudden change in the observed AED value. This is easily visible in Figure 3 as the sharp spikes. This shows that AED is correctly able to identify critical routing anomaly events at the individual prefix level.

Figure 4 shows another example where a change in the origination of the 4/8 prefix is easily identified using AED. The figure displays data from the same time four month time period as the previous graph, however, in this case the anomaly is observed on August 12th, 2005. Once again, the stable ASPATH that is normally associated with the 4/8 prefix, is seen to change from ”1239 3356” to ”1239 8764 8764 8764 8764 8764 8764”, thereby resulting in a sudden change in AED.

C. ASPATH Anomalies

While AED makes it simple to identify origin AS related anomalies, it is perhaps best suited to identifying more subtle anomalies that might occur in the ASPATH. A stable prefix would consistently exhibit an AED value of zero. In fact, the results described in the the previous section, illustrate that a large portion of prefixes in the Internet do indeed exhibit a zero AED. In this section, we describe two very simple scenarios that serve to illustrate situations where a non-zero AED value serves to indicate a potential routing anomaly. Figure 5 shows the measured AED for the prefix 35/8 over the four month time period from June-September 2005. Once again we are limiting ourselves to the data as observed via the Sprint-Routeviews peering session. Once again we notice that the 35/8 prefix is relatively stable, and therefore most of the time exhibits an AED value of zero. However, we do notice a few non-zero values of AED. Closer examination of the data reveals that on those occasions, the observed ASPATH for this prefix changed from ”1239 3561 237” to ”1239 2914 174 237” resulting in an AED value of 2. This example shows how AED can be used to identify ASPATH changes which might be of interest even though the origin AS of that prefix does not change. It should be noted that a non-zero AED does not necessarily imply a routing anomaly; however it can be a useful tool to guide further analysis of observed ASPATHS for a given prefix.

While the previous example illustrates the use of AED for a simple scenario where the prefix being observed is relatively stable, it does not describe how well it might work for some more complex prefixes. In order to study this we focused our attention on the prefix 81.212.149.0/24. This prefix is originated by AS 9121, and frequently shows up as one of the top 20 most active prefixes(in terms of the number of times it is announced). Figure 6 shows the value of AED over time that this prefix exhibits. As expected there is significantly more activity for this prefix; however, what is interesting

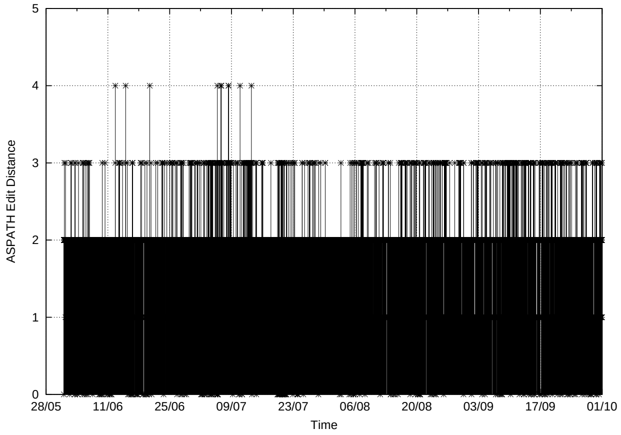

Fig. 6. ASPATH Edit Distance for 81.212.149.0/24 from June-Sept 2005

is that the AED values for this prefix seem vary from 0-4, but an AED value of 4 is only seen during June-July. Closer examination of the data reveals that this prefix is observed with 3-4 different ASPATHS such as ”1239 6762 9121”, ”1239 701 6762 6762 9121”, and ”1239 1299 12713 9121”. While these may simply be advertisements of alternate paths, it is still activity that is pointing at some inherent instability in that prefix.

IV. Discussion

So far in this paper we have described a very simple AED metric. However, there are several enhancements that can be made that can improve the performance and results obtained by using AED. In the subsections below, we briefly outline a few such refinements that we would like to investigate.

A. Building a Prefix Model

In order to accurately use AED for anomaly detection, it is important to first build a model of how a given prefix behaves in terms of ASPATH values observed over time. We need to first employ a base-lining phase where we make a determination of what a good base ASPATH for that prefix is. We can then compute AED from that base path. This could be as simple as computing a histogram containing each unique ASPATH observed in the training phase and then selecting the most frequently advertised path as the base. Prefixes advertised by multiple origin ASes (MOAS) can be handled if a base path is selected for each origin AS. In this scenario, for each new ASPATH value observed after the training phase, a comparison would be made not with a single ASPATH but with the *relevant* ASPATH from the same origin. Despite any care taken to train on a *clean* (absent of anomalies) dataset, there does not exist any reliable way to make that guarantee. Fortunately, determining a base path based upon announce frequency will discard most small anomalies, while larger anomalies can be accounted for through human intervention.

B. Parameter Setting

An important parameter that we have used in this paper is MAX_EDIT_DISTANCE. In the experiments described in this paper we set this value to be roughly 4 times the maximum AED value observed in the datasets being studied. The goal of this parameter is to emulate an impulse function that will magnify the particular event where there is a change in the origin AS. There are, however, other ways in which the appropriate value for this parameter could be determined. One possibility is to track the largest value of AED observed during the training phase for each prefix and then have a per-prefix MAX_EDIT_DISTANCE which is several times larger than that value. Another approach is to compute this parameter for each (prefix,unique-origin) pair similar to the method for finding a base path.

C. Clustering

Clustering is an extremely useful technique for reducing the complexity of a dataset. For example in [10] Zhang et. al. propose a BGP anomaly detection system based on instance learning. They represent BGP update message dynamics via wavelet analysis, and then use clustering techniques in order to identify anomalous behavior. We are looking into using a similar clustering technique based on AED. [10] clusters prefixes which are advertised close together in time, while our clustering would use AED as a similarity metric. ASPATHs for a single prefix would be clustered to describe sets of similar paths. Using such a technique would enable us to build a more complete model regarding the behavior of a prefix. It would allow us to correctly model complex observed behaviors, such as multiple advertised paths and multiple origin AS while, at the same time, allow the model to incorporate tolerances for deviations from the trained model. For example, a new observed path might only be a slightly different from a normal ASPATH. In this case, the new path would fall within one of the clusters constructed during training. At the same time significant variations from previously observed ASPATHs would not map to any existing clusters and this would be an indicator of a significant anomaly.

V. Conclusion

In this paper we have described a new metric called ASPATH Edit Distance (AED) which can be used to study the behavior and performance of the BGP protocol. AED is based on the concept of edit distance which allows us the ability to *quantitatively* determine the difference between two strings. AED allows us to make similar comparisons of the ASPATH values contained in BGP update messages. We describe some key features of AED and also provide some discussion regarding how the preliminary work described in this paper can be extended.

Our future work in this area is focusing on building accurate and detailed models of prefix behavior that captures observed ASPATH information. These include incorporating both the base-lining approach as well as the clustering approach we outlined briefly here. In particular our goal is to be able to use such models to develop and improve BGP routing anomaly detection techniques.

Acknowledgments

This work was supported in part by the Department of Homeland Security(DHS) under contract number NBCHC040146, and by a corporate gift from Cisco Corporation.

References

[1] C. Labovitz, R.G. Malan, and F. Jahanian. Internet routing instability. *Proceedings of the ACM/IEEE Transactions on Networking*, pages 515–528, October 1998.

[2] C. Labovitz, R.G. Malan, and F. Jahanian. Origins of internet routing instability. *Proceedings of IEEE INFOCOM*, March 1999.

[3] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang. Bgp routing stability of popular destinations. *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, Nov 2002.

[4] J. Wu, Z.M. Mao, J. Rexford, and J. Wang. Finding a needle in a haystack: Pinpointing significant bgp routing changes in an ip network. *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation(NSDI)*, May 2005.

[5] M. Lad, X. Zhao, B. Zhang, D. Massey, and L. Zhang. Analysis of bgp update surge during slammer worm attack. *Proceedings of the 5th International Workshop on Distributed Computing (IWDC)*, Dec 2003.

[6] V.I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady 10(8)*, pages 707–710, Feb 1966.

[7] M. Gilleland. Levenshtein distance, in three flavors. *Merriam Park Software: http://www.merriampark.com/ld.htm*.

[8] D. Blazakis, M. Karir, and J.S. Baras. Bgp-inspect: Extracting information from raw bgp data. *Proceedings of the IEEE/IFIP Network Operations and Management Symposium*, Apr 2006.

[9] The University of Oregon. The routeviews project. *http://www.routeviews.org*.

[10] J. Zhang, J. Rexford, and J. Feigenbaum. Learning-based anomaly detection in bgp updates. *Yale University Department of Computer Science Tech Report - YALEU/DCS/TR-1318*, April 2005.