A Novel Graph Spectra Approach to Anomaly Detection in Border Gateway Protocol

Leigh Metcalf[1], Will Casey[2], Timur Snoke[3], Heeralal Janwa[4], Shirshendu Chatterjee[5], and Ernest Battifarano[6]

[1] Carnegie Mellon University, Pittsburgh PA 15213, USA
lmetcalf@andrew.cmu.edu
[2] US Naval Academy Annapolis, MD, USA
wcasey@usna.edu
[3] Carnegie Mellon University, Pittsburgh PA 15213, USA
emailtsnoke@andrew.cmu.edu
[4] University of Puerto Rico San Juan, USA
heeralal.janwa@upr.edu
[5] City University of New York, Graduate Center and City College New York, NY, USA
shirchat1@gmail.com
[6] Retired at New York, NY, US
ernest.battifarano@gmail.com

Abstract. We carry out an effective clustering of distributed time series graphs. The work uses singular value decomposition (SVD) of spectral gaps of graphs derived from BGP data found at the RouteViews Project at the University of Oregon. We detect extremal events and examine anomalous events that cause disruptions in Internet routing. We use spectral gaps to examine how redundant and sparse the graph is. We base the anomaly detection method in BGP networks on the fact that a drop in the value would signify a lack of this redundancy, which in the distributed redundant Internet would be an anomalous event.

Keywords: BGP, graph, spectral-gap, expanders, time series, time series clustering, spectral clustering, SVD, power-law

1 Introduction

The underlying process that sends traffic through the Internet is called BGP, Border Gateway Protocol. BGP lies below e-mail, web traffic, TCP, UDP, and other protocols that are often thought of when considering traffic. Instead, it is a core protocol that allows the top protocols to do their job. A TCP connection only manages the traffic between two computers, it does not determine how the traffic arrives at its destination.

A major disruption of BGP can bring the Internet to a halt. As we rely on the Internet for almost everything in modern life, studying the basic underlying system to analyze how these disruptions occur is important. We not only want

to know how the disruptions occur, we want to know how widespread they need to be before the protocol is affected.

BGP consists of routers located around the Internet. Each router in the BGP fabric shares its routes with every router to which it connects. This is not a single event; the routes are shared repeatedly between the two connections as a keep-alive method. In addition, the routes are shared with every other router to which the two routers are connected. After some time, every router connected to the BGP system has a route to all networks that use the system. BGP creates a very distributed system, with no single router responsible for everything.

If we consider a single router, then we can examine a time series of the routes that it has received during a given time period. Multiple routers yield multiple time series. In addition, the Internet is often modeled using graphs, in particular, graphs derived from the connections made in routing. Using the routes collected by a router, we can create time series of the graphs for each collector.

Disruptions on the Internet can yield disruptions in the time series. The question becomes how big of a disruption in the Internet is needed before the disruption is apparent. How extremal should the event be before it affects the entire Internet. We tackle this and other questions in our analysis of multiple time series in this paper. We want to cluster the time series of graphs to find anomalies in their behavior and to determine how known disruptions in the Internet affected their behavior. BGP network graphs for routing have been shown in [10] as scale-free expander graphs (i.e., they have large spectral gaps). In this paper, we expand the methods for analyzing and studying the evolution of those graphs.

We make two fundamental contributions. First, we leverage spectral gap and expansion to convert the graph behavior (BGP routing) into a time series via its time-dependent spectral gap. Second, we use SVD clustering, which helps us identify similar behaviors as clusters (of the time series) as well as anomalous outliers, such as router problems, as we confirmed in this case study. As a result, we detect an anomaly and identify the aberrant router.

This paper is organized as follows. We begin with a background, that is, previous methods used to cluster time series of graphs. Time series clustering has been studied by various authors, including the authors of this paper in [6]. We then move on to a discussion of the spectral gap, followed by how the BGP protocol works in general. Next, we consider the method used in this paper for clustering time series of graphs. Next, we apply the method to data collected from various routers that share their data and examine the results. Finally, we end with a discussion of the results and plans for future work.

2 Related Work

Research has strongly suggested that the BGP graph follows the preferential attachment model, is scale-free and follows the power law [7].

The power-law graphs are robust; that is, the structure does not change by deleting nodes randomly. However, they are fragile in the sense that if one removes some of the hub or authority nodes, then the power law is disrupted [5].

Some network spectral gap analysis has been performed [13]. For BGP graphs, an analysis can be found in [10].

Substantial work has been done on dynamic graph evolution and clustering; the work here is closest to [6].

In [14] a time series clustering algorithm is performed using K-means for financial data. Their algorithm resembles a traditional machine learning algorithm in that a loss function, the point-wise difference between time series and the conjectured centroid, is minimized. This algorithm is close to the second phase of our method. However, we minimize compressed time sequences and provide a graphical representation. In [20] non-linear encoders, Deep Temporal Clustering Representation (DTCR) was used prior to the K-means clustering to improve aspects of seq2seq classification quality. Our method, in contrast, derives metrics of connectivity quality from dynamic graphs, forms representation with SVD decomposition, and then applies K-means to coefficients of a data-centered basis.

3 Spectral Gap Introduction

An expander graph is a highly connected "sparse" graph (see, for example, [24]). Expander graphs have numerous applications including those in communication science, computer science (especially complexity theory), network design, cryptography, combinatorics, and pure mathematics.

Expander graphs have played a prominent role in recent developments in coding theory (LDPC codes, expander codes, linear-time encodable and decodable codes, codes attaining the Zyablov bound with low complexity of decoding) (see [21], [19], [26], [30], [28], [32], [18], [9], [15], [16], [4], [11], and others).

We shall consider graphs $\mathcal{X} = (V, E)$, where V is the set of vertices and E is the set of edges of $\mathcal{X}$. We will assume that the graph is undirected and connected, and we shall only consider finite graphs. For $F \subset V$, the *boundary* ∂F is the set of edges connecting F to $V \setminus F$. The *expanding constant*, or *isoperimetric constant* of $\mathcal{X}$ is defined as,

$$h(\mathcal{X}) = \min_{\emptyset \neq F \subset V} \frac{|\partial F|}{\min\{|F|, |V \setminus F|\}}.$$

Moreover if $\mathcal{X}$ is viewed as the graph of a communication network, then $h(\mathcal{X})$ measures the "quality" of the network as a transmission network. In all applications, the larger the $h(\mathcal{X})$ the better, so we seek graphs (or families of graphs) with $h(\mathcal{X})$ as large as possible with some fixed parameters.

In [21] M. Tanner introduced another notation for the expansion coefficient. Let, as before $\mathcal{X} = (V, E)$, be a graph where V is the set of vertices and E is the set of edges of $\mathcal{X}$. Let $X \subseteq V$ with $|X| \leq \alpha|V|$, then

$$c(\alpha) = \min_{\emptyset \neq X \subset V} \frac{|\partial X|}{\min\{|X|, |V \setminus X|\}}.$$

It is well known that the expansion properties of a graph are closely related to the eigenvalues of *adjacency matrix* A of the graph $\mathcal{X} = (V, E)$; it is indexed by pairs of vertices x, y of $\mathcal{X}$ and A_{xy} is the number of edges between x and y. When $\mathcal{X}$ has n vertices, A has n real eigenvalues, repeated according to multiplicities that we list in decreasing order.

$$\mu_0 \geq \mu_1 \geq \ldots \geq \mu_{n-1}.$$

It is also known that if $\mathcal{X}$ is D regular, i.e. all vertices have degree D, then $\mu_0 = D$ and if, moreover, the graph is connected $\mu_1 < D$. Also, $\mathcal{X}$ is bipartite if and only if $-\mu_0$ is an eigenvalue of A. We recall the following.

Theorem 1. *[8] Let $\mathcal{X}$ be a finite, connected , D-regular graph then*

$$(D - \mu_1)/2 \leq h(\mathcal{X}) \leq \sqrt{2D(D - \mu_1)}.$$

And

Theorem 2. *[8] Let $(\mathcal{X}_m)_{m\geq 1}$ be a family of finite connected, D-regular graphs with $|V_m| \to +\infty$ as $m \to \infty$. Then*

$$\liminf_{N\to\infty} \mu_1(\mathcal{X}_m) \geq 2\sqrt{D-1}.$$

This leads to the following

Definition 1. *A finite connected, D-regular graph $\mathcal{X}$ is* Ramanujan *if, for every eigenvalue μ of A other than $\pm D$, one has*

$$\mu \leq 2\sqrt{D-1}.$$

We will also need

Definition 2 (Bipartite Ramanujan Graphs). *Let $\mathcal{X}$ be a (c, d)-regular bipartite graph. Then $\mathcal{X}$ is called a Ramanujan graph if*

$$\mu_1(\mathcal{X}) \leq \sqrt{(c-1)} + \sqrt{(d-1)}.$$

It is known that computing the expansion coefficient of arbitrary graphs is an NP-complete problem. Thanks to the work of Tanner, and Alon and Millman, one can derive bounds on the expansion coefficient in terms of μ_1. The complexity of determining μ_1 is still difficult if the number of vertices is large (for example of the order 10^6 for LDPC codes). For new bounds on the spectral gaps of irregular graphs, we refer to Hoeholdt and Janwa [13].

4 Border Gateway Protocol Introduction

The process of sending data through the Internet is called routing. This process was relatively easy in the early days of the Internet, as it was a small size. As the Internet grew, a new process was needed to handle the size, and the Border

Gateway Protocol (BGP) was created. BGP is a protocol designed to route traffic between organizations. Each organization has a network or networks that it would like to be available online. This collection is known as an Autonomous System. Their local Regional Internet Registry (RIR) assigns an Autonomous System Number (ASN) for use in BGP.

The ASNs are logical collections of networks; there is no guarantee that the networks are located in the same building or even the same country. That means that BGP is a logical routing protocol, not one tied to any geographic area. It is also important to know that there is no one-to-one relationship between organizations and ASNs. An organization may have one or more ASNs, depending on their requirements.

BGP is a peering network, which means that ASNs have relationships with each other to exchange information, that is, routes. They want traffic to be routed to their networks, so they *announce* to their peers that they have a network. Their peers then share that information with their peers. For example, suppose that an organization has ASN 65512 and the network 192.168.50.0/24. They peer with ASN 64656, so they announce to that network

```
65512 192.168.50.0/24
```

That is, the BGP method of saying 'This network resides here.' Then ASN 64656 announces to its peers:

```
64656 65512 192.168.50.0/24
```

Then any peer of 64656 that wishes to access the network 192.168.50.0/24, would send its traffic to that ASN. That ASN would then pass it to 65512, the owner of the network.

The actual traffic passing is done by what are called *default routes*. The organization that runs the ASN 65512 has a router with an interface on it that is assigned an IP address. That IP address is then given a direct route to the interface on the router that runs ASN 64656. The IP address assigned to 65512 is the source of the route, and the IP address assigned to 64656 is the destination.

The routes are propagated through the Internet until everyone has that route. Each router that receives a route prepends its own ASN to the list and sends it on to its peers. Another important fact about BGP is that it constantly sends update messages, usually on a configured time interval, to ensure the route is available. We would see the previous example repeatedly in our data. If we don't see it, it is usually because of an interruption at some point in the path.

If ASN 65512 is no longer the owner of 192.168.50.0/24, then that ASN announces a *withdrawal* of that network to its peers. That withdrawal is propagated through the collection of ASNs until everyone has withdrawn from the network. Announcements contain the full path, but withdrawals do not. Analysts then have to infer the path in question from the message.

The RouteViews Project [22] is run by the University of Oregon and shares routing tables, that is, the results of its peers sharing routes with it. The routers also share the continuous announcements and withdrawals they see for a fuller view of Internet routing. They collect data from at least 46 routers located around the world. Information about collectors can be found here [1]. They have

two types of collectors: exchanges and multi-hops. [2] Exchanges are created by entering into an agreement with RouteViews to provide data to their system. The multi-hops are routers they maintain for anyone to peer with and share information. The downside of the multi-hops is that they are then subject to the very issues that the RouteViews Project was designed to discover in the Internet as a whole, but the upside is we get a wider section of the Internet from them.

RouteViews is freely available and allows for Internet-wide analysis of routes. We do know that it isn't a full and complete view of the Internet. For example, China uses what is known as the *Great Firewall of China*, which restricts what is allowed to pass beyond its borders.

We want to analyze the differences in behavior for each router in the RouteViews collection. Due to the nature of paths propagating through the Internet via the protocol, we would expect that the would have similar graphs and that large events that affect routing in the Internet would affect them in the same manner.

The data stored by RouteViews contain the announcements and withdrawals of routes. We do not see the keepalives or any other communications used in the protocol. This does mean that we often see the same route announced repeatedly, as mentioned before. It does not mean that the route is bad or new, just that the system is ensuring that we have all of the possible routes.

BGP is an underlying protocol; it is the process that allows traffic to flow. It is agnostic, as a rule, about the traffic it sends, but at the same point, it can be disrupted by the traffic. For example, a DDoS attack can exhaust the resources of a BGP router, knocking it off the net. [3] A widespread DDoS attack could interrupt the connections for multiple routers, causing interruptions in the fabric of the Internet.

5 Data

We begin by binning the data into hourly bins for each router. Within that hour, we look at the announcements seen by the router and consider the paths that were shared with the router during that time. From these paths, we create a directed graph.

Recall that the paths are seen in this format: `AS1 AS2 AS3 AS4 192.168.50.0/24` Where `AS4` is the originating Autonomous System for the route to the network 192.168.50.0/24. Therefore, the edges in this graph are given by `AS4 AS3`, `AS3 AS2`, `AS2 AS1`, as the announcement of 192.168.50.0/24 is given by `AS4` to `AS3` and so forth. We only analyze the announcements, as mentioned before; withdrawals do not directly contain the paths.

There are millions of records per router per day, so looking at the hourly records is more tenable. Also, by looking at just the paths every hour, we reduce the amount of data even further.

6 Method

Domain and behaviors: Letting $\mathcal{U}$ be the domain, a discrete set of routers. Let the time domain be represented by counter k corresponding with $t_0 + kh$, where t_0 is the start time, and h is a discrete-time difference. Behaviors are qualified by characteristic or salient features of functions over the domains.

For example, a significant dip in the spectral gap from hour to hour would imply that the highly connected nature of the Internet was altered, perhaps by an event such as a DDoS.

6.1 Phase 1: behavior characterization for local networks.

In stage 1, we develop a low-dimensional proxy measure for more complex behaviors. We focus on graph connectivity quality measures and derive a method to develop those measures from our source data. For each time step k, for each router ν, our data analysis yields the graph's adjacency matrix M_k^ν at time step k. We then compute Common out Neighbor degree Matrix (for time step k) as $A_k^\nu = M_k^\nu \left(M_k^\nu\right)^T$, which is non-negative definite. Using A_k^ν we determine an eigen-value decomposition:

$$A_k^\nu = U \begin{bmatrix} \mu_0^{(\nu,k)} & 0 & \cdots & 0 \\ 0 & \mu_1^{(\nu,k)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_{n-1}^{(\nu,k)} \end{bmatrix} U^*$$

.

The spectral gap, denoted as σ_k^ν, is defined to be:

$$\sigma_k^\nu := \mu_0^{(\nu,k)} - \mu_1^{(\nu,k)}$$

$\sigma(A_k^\nu)$. The spectral-gap defined as the difference of the leading two eigenvalues associated to the graph, is often interpretated as a measure of graph connectivity quality. For directed graphs, we look at the spectral gap of AA^T (i.e., the singular value gap). A large spectral gap implies that the graph is highly connected yet sparse, which is preferred in the Internet. [29] [27]. For example, a large dip in the spectral gap from hour to hour would imply that the highly connected nature of the Internet was altered, perhaps by an event such as a DDoS.

The derived data $\sigma(A_k^\nu)$, will be used to cluster routers in stage 2.

The end result of phase 1, is a time series for each router:

$$f_\nu : \mathbb{N} \to \mathbb{R} : k \to f_\nu(k) = \sigma_k^\nu$$

, the time series is composed of dynamic graph quality measures.

An example of this can be found in Figures 1 and 2.

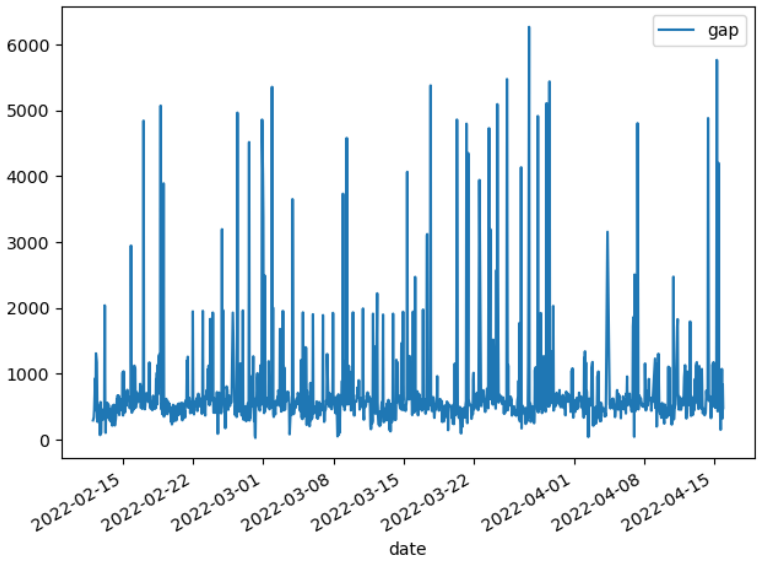

Fig. 1. Spectral Gap Time Series for route-views.sydney

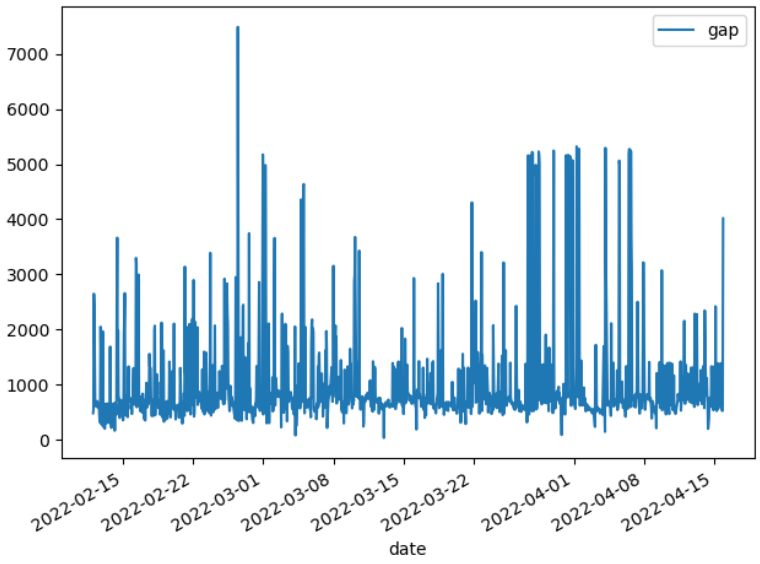

Fig. 2. Spectral Gap Time Series for route-views5

6.2 Phase 2: clustering coexpression.

We next classify router behavior. Formally our problem is to partition the set $\mathcal{U}$ by salient behavioral aspects detectable within the set of functions:

$$\mathcal{F} = \{f_\zeta | \zeta \in \mathcal{U}\}.$$

Here the goal is to create clusters whose members express similar evolving trends with regard to the graph connectivity quality measure.

To cluster functions, we apply the methods of [6], by implementing the following steps.

Step 1: Convert the time series set $\mathcal{F}$ into a data matrix. This step is not required, but allows for a more straightforward application of the methods cited. Let $X_{ij} = f_i(j)$. So, each row of X refers to a router, and each column is a time sampling from all routers.

Step 2: Compute the Singular Value Decomposition (SVD) of $X_{m\times n} = U_{m\times m}S_{m\times n}(V^T)_{n\times n}$. Characteristic time series are associated with rows of V^T.

Step 3: Use the elbow method (using the *kneed method [25]*) to determine the set S of significant singular values. The size of S determines the number of clusters, number of cluster centroids, and number of behavior classes for the clustering objective. (The kneed method finds the elbow, denoted by η, which determines the best low-rank approximation. One pays a cost for increasing the rank- suffering the consequence of the law of diminishing returns. The elbow index, η, represents the best trade-off. We denote these dominant singular values by $S := [s_1, s_2, \cdots, s_\eta]$.)

Step 4: Set the submatrices $\tilde{U}$ and $\tilde{V}$ to be the first η columns from U and V, respectively.

Step 5: Let $W_{m\times n} = \tilde{U}_{m\times\eta}(\text{diag}(s))_{\eta\times\eta}(\tilde{V}^T)_{\eta\times n}$. W is the best rank η approximation of X (see the Eckart-Young Theorem).

Step 6: Reformulate the data, using the new basis of the characteristic time series with $Y_{m\times\eta} = \tilde{U}_{m\times\eta}(\text{diag}(s))_{\eta\times\eta}$. The rows represent coefficients of linear approximation over characteristic time series (best fitting all data in the L_2 norm).

Step 7: Cluster using the K-means clustering (with η clusters) method on the rows of Y. The rows of the matrix Y represent the coefficients of linear approximation over the time trends. The clustering results in an assignment:

$$X : \mathcal{U} \to [1, \ldots, k] : u \to X(u)$$

Two routers a, b such that $X(a) = X(b)$ are assigned to the same group. This can occur when they have similar loadings with respect to the fundamental time series basis exposed by the SVD. Thus their grouping similarity reflects the similar expression of those characteristic time series toward overall minimized L_2 approximation.

Step 8: Validate the clustering, by testing the *sum of squared errors* (SSE) for K-means. Validation testing can be considered by letting k (number of clusters) range from 1 to η, determine the SSE. Treating the SSE as a function of

cluster size k, To select the best k, another round of diminishing returns can be employed by using the elbow finding algorithm. Additionally, similar SSE testing can be re-performed over by relaxing the cost associated with selecting η in Step 3.

6.3 Phase 3: visualization.

View 1: cluster centroids: Cluster centroids represent all members of a cluster class. The centroid is a time series that minimizes the variance, or L_2 distance from all other cluster members. As such, we can also recover and plot the time series of the centroid. A cluster member should look more alike to its centroid than any other centroid.

View 2: Centroid Weightings: Given all members of a cluster, The centroid's weightings are merely average load weightings. That is letting cluster i include $C_i\{j : X(j) = i\}$:

$$w_i = \frac{1}{|C_i|} \sum_{j \in C_i} Y_j$$

by salient features that are responsive enough to identify a class of cybersecurity problems such as DoS, DDoS, Sinkholeing, disruptions, BGP interruptions, connectivity fluctuations, and so forth.

We repeat this graph creation; this is done for each hour and each router in our dataset from 20220212 through 20220415. From there, we can compute temporal clusters of the graphs using the method created in [6].

7 Results

The first anomaly found in our method is illustrated in Figure 3. The top cluster, Cluster 0, clearly has very anomalous spectral gaps compared to the other clusters. The cluster contains only one router, the router labeled route-views5 by the RouteViews Project. Following through the data for further analysis, we see that in Figure 4 the behavior continued until the next day and, in fact, did not stop until noon on 20220328.

There were no Internet-wide behaviors that could have caused it, so we turned to the logs that the RouteViews Project makes available. Analyzing those logs, one of which is found here `https://archive.routeviews.org/bgpstatus//2022-02/18-routeviews-bgpd-status.txt`, we see that that router was constantly losing and re-gaining connections with more than one of its peers. There is no single cause for such an event. It could have been a hardware issue, such as when the hardware starts to go bad. This often causes flakiness in the router's connection. The connection between the router and the Internet could have been hampered or there could have been other reasons. These reasons are generally unknown to the public, but we do see their effects.

In this case, the single behavior of that router is not repeated by the Internet at large, but we still see the results in our method.

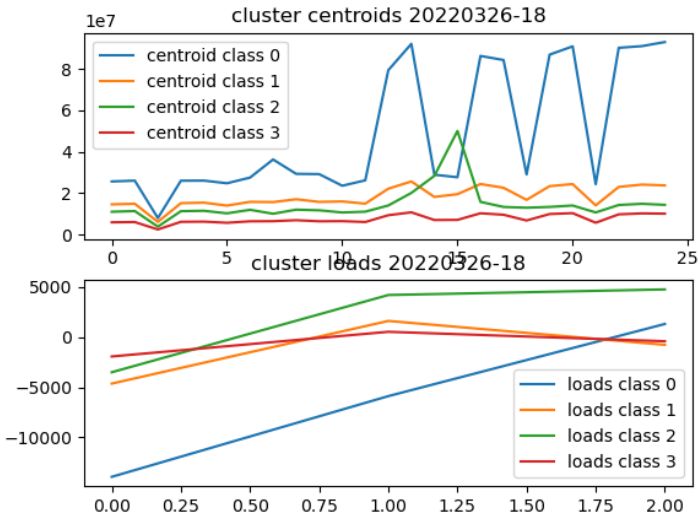

Fig. 3. Cluster Loads for 20220326-18 through 20220327-17

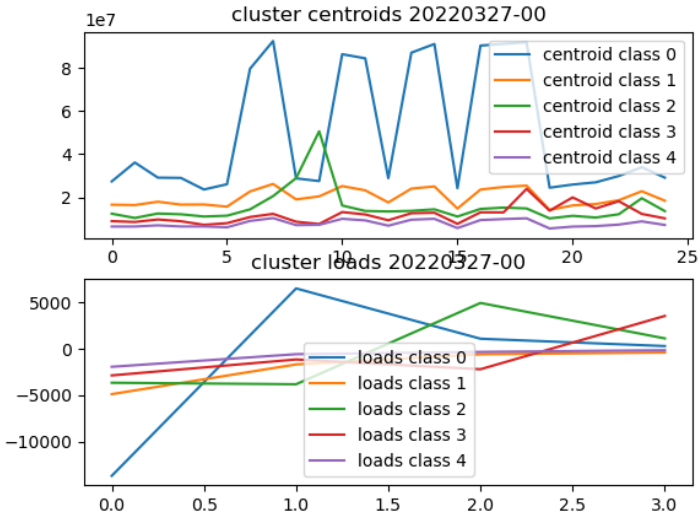

Fig. 4. Cluster Loads for 20220327-00 through 20220327-23

For another example, consider Figure 5. The drop in spectral gap data was recorded at 20220218-09. Interestingly, we know that a DDoS was recorded on this date; however, we do not know the exact time of the event [31] as it was not shared. A DDoS can cause interruptions in the sharing of BGP routes by interrupting the connection between routers. For example, a DDoS would interrupt the connection, causing the connection between the two routers to fail. This would stop the DDoS from saturating that connection, as it would lose its connection to its target through those routers. Once saturation stops, the BGP connection is re-established, and the DDoS often resumes its attack. This is sometimes called *flapping*.

The DDoS was caused by an amplification attack. An amplification attack [17] uses various methods to increase the amount of traffic seen in the attack. A common amplification attack [23] uses the Network Time Protocol, or NTP. In this case, a small request to an NTP server would return a very large response. The attack on February 18, 2022 was first found on that date, without a given time. It was caused by a vulnerability in a phone system, which reiterates how dependent we are on the Internet for everything.

The attack may be correlated with the drop and the immediate recovery of the spectral gap, as the traffic was interrupted, resumed, and then interrupted again repeatedly.

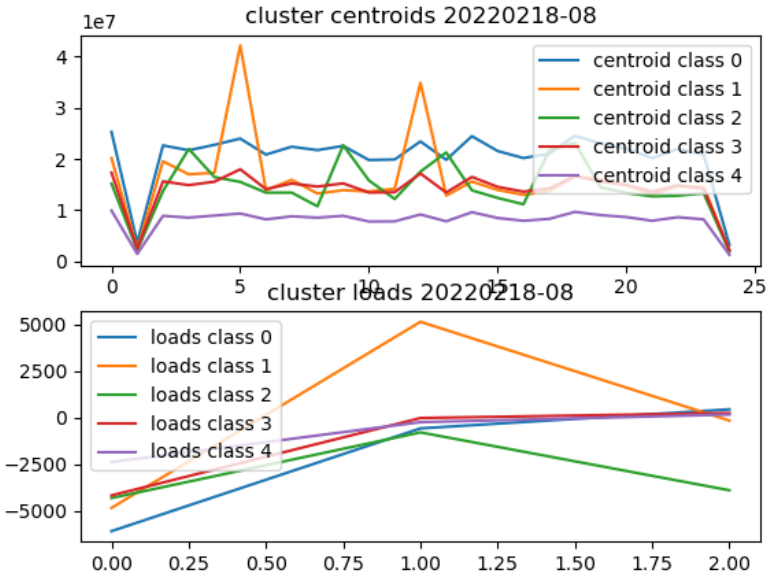

Fig. 5. Cluster Loads for 20220218-08 through 20220219-07

For a final example, we considered the event caused by Rogers Communications in July 2022. [12] This event started at approximately 1600 hours on July 8, 2022 and was caused by the company's massive withdrawal of routes. Rogers Communications is located in Canada and, at the time of the event, Internet connectivity greatly affected the BGP system in Canada.

Figure 6 illustrates the effects of this massive withdrawal on the routing table as a whole. Clearly it caused instability throughout the Internet, as seen in the drop of the spectral gap.

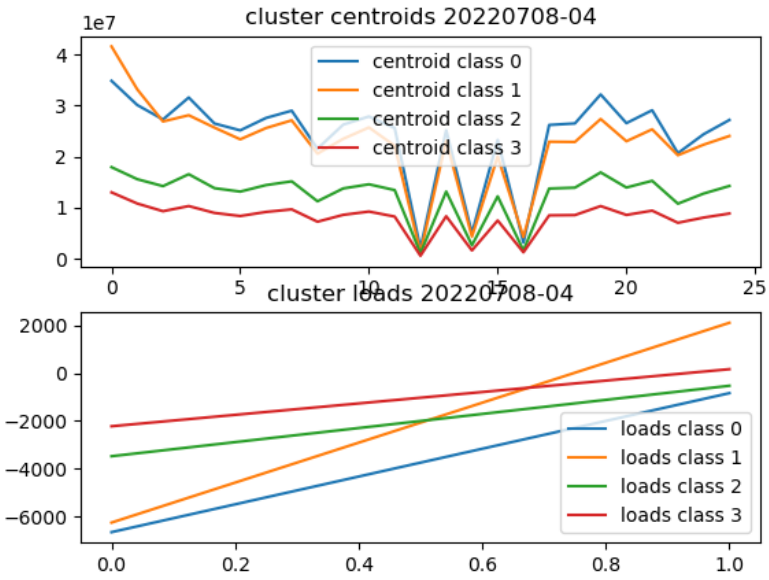

Fig. 6. Cluster Loads for 20220708-04 through 20220709-04

Routeviews publishes a map of collectors [1]. Using information derived from this map, we can illustrate the cluster locations as seen in Figure 7. This method highlights the widespread effects of Internet issues from the Rogers event, as well as illustrates how routers located across the world from each other can still have similar behavior.

Cluster Quality: Various measures of cluster quality such as Silhouette Score or Davies-Bouldin index can be deployed to test and validate clustering. In figure 8 we plot the Silhouette Score for clustering as a function of time. The over all average Silhouette Score of 0.498 suggests an overall level of quality for the method. Note also the oscillation rage between 0.3 to 0.7, suggests that cluster quality score maybe an interesting signal of its own and relate to changes

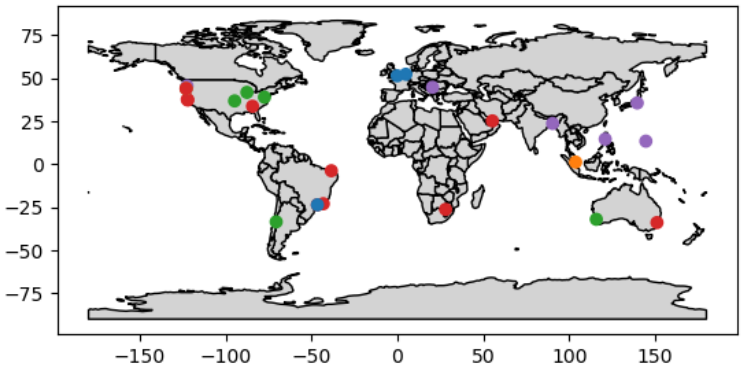

Fig. 7. Mapping the Location of Routers in RouteViews, dots indicate the location of routers, color indicates cluster identity for those routers.

such introduction of anomalous behavior. We plan to investigate this possibility further in future work.

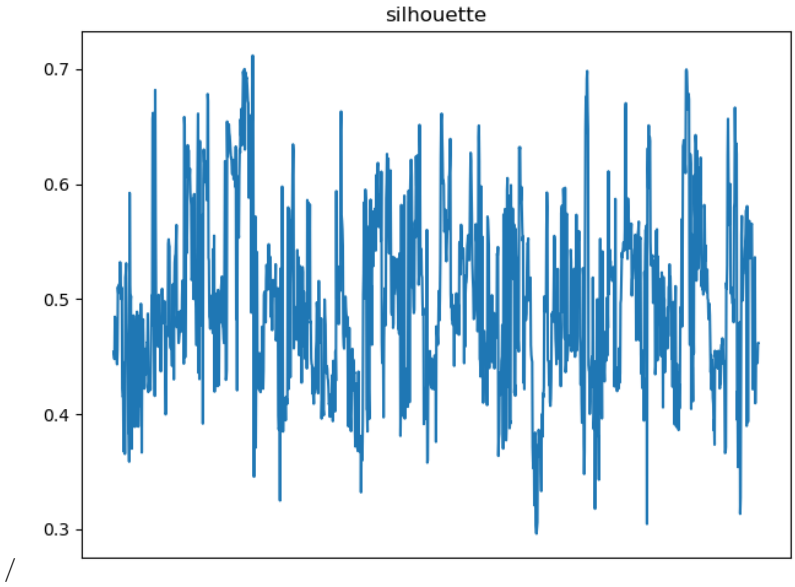

Fig. 8. Cluster Quality can be measured using Silhouette Score as a function of time. After developing the clustering for each time point (hourly) in the date range 20220211-19 to 20220414-19 by spectral gap statistics, we reassess the cluster quality using Silhouette Score for each sample. The cluster quality offers another interesting signal for anomaly detection.

8 Conclusions and Future Work

In this paper, we examined a time series of graphs created at route collectors of BGP data. Using the spectral graph of each graph created at each time, we were able to cluster the collectors and look for patterns in the results that corresponded to known Internet events.

We saw how a single router can be affected by events that affected the router, even if we are not aware of the causes. These events only affected a single router, and we saw how they did not affect other routers, which is to be expected in clustering.

We then saw the results of a DDoS and how that disruption can affect the entire fabric of route sharing. A DDoS may have a single target, but its effects are widely distributed across the Internet. A DDoS can use thousands of hosts on the Internet, located all over the world. It stands to reason that their behavior, in particular in an amplification attack, would be beyond the target of the attack.

Finally, we looked at an outage caused by a single company misannouncing a large number of routes, and saw that that single company could affect the routing graphs worldwide.

In the future, we should look at the results on a much larger scale, examining stability beyond the three examples used in this paper. We also want to create an hourly stability report for network engineers.

Finally, we want to use this method on other graphs connected to Internet data, such as network traffic.

9 Acknowledgments*

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM25-0164

The Authors would like to thank the reviewers who offered careful review and suggestions to improve the paper. This research of SC has been partially supported by NSF DMS grant #2154564. The research of Heeralal Janwa was supported by the NASA grant 80NSSC22M0248 and also by the PR NASA Grant Consortium and its Director, Dr. Gerardo Morell.

References

1. Routeviews collectors. `https://www.routeviews.org/routeviews/collectors/`

2. Routeviews: The bgp collector for today's network operator & researcher. https://bgp4all.com/pfs/_media/conferences/mnnog1_routeviews_usecases.pdf
3. Awasthi, K.: Defending against ddos attacks: Everything you need to know. https://fidelissecurity.com/cybersecurity-101/threat-intelligence/ddos-attack/
4. Barg, A., Zémor, G.: Error exponents of expander codes. IEEE Transactions on Information Theory **48**(6), 1725–1729 (2002)
5. Birge-Lee, H., Apostolaki, M., Rexford, J.: Global bgp attacks that evade route monitoring. arXiv preprint arXiv:2408.09622 (2024)
6. Casey, W., Metcalf, L., Janwa, H., Chatterjee, S., Battifarano, E.: A novel ml method for temporal evolution of geographic clusters of disease spread patterns. In: International Symposium on Intelligent Computing and Networking. pp. 149–164. Springer (2024)
7. Chen, Q., Chang, H., Govindan, R., Jamin, S.: The origin of power laws in internet topologies revisited. In: Proceedings. twenty-first annual joint conference of the ieee computer and communications societies. vol. 2, pp. 608–617. IEEE (2002)
8. Davidoff, G.P., Sarnak, P., Valette, A.: Elementary number theory, group theory, and Ramanujan graphs, vol. 55. Cambridge university press Cambridge (2003)
9. Forney, G.D.: Codes on graphs: recent progress. Physica A: Statistical Mechanics and its Applications **302**(1-4), 1–13 (2001)
10. Gkantsidis, C., Mihail, M., Saberi, A.: Conductance and congestion in power law graphs. In: Proceedings of the 2003 ACM SIGMETRICS International Conference on Measurement and modeling of computer systems. pp. 148–159 (2003)
11. Guruswami, V., Indyk, P.: Expander-based constructions of efficiently decodable codes. In: Proceedings 42nd IEEE Symposium on Foundations of Computer Science. pp. 658–667. IEEE (2001)
12. Heinle, A.: The canada wide rogers outage on july 8, 2022: What exactly happened & how can it be prevented? urlhttps://www.coguard.io/post/canada-rogers-outage-root-cause-analysis
13. Høholdt, T., Janwa, H.: Eigenvalues and expansion of bipartite graphs. Designs, Codes and Cryptography **65**, 259–273 (2012)
14. Huang, X., Ye, Y., Xiong, L., Lau, R.Y., Jiang, N., Wang, S.: Time series k-means: A new k-means type smooth subspace clustering for time series data. Information Sciences **367**, 1–13 (2016)
15. Janwa, H., Lal, A.: On tanner codes: parameters and decoding. Applicable Algebra in Engineering, Communication and Computing **13**, 335–347 (2003)
16. Janwa, H., Lal, A.K.: On expander graphs: parameters and applications. preprint, arXivcs/0406048 (2004)
17. Kopp, D., Dietzel, C., Hohlfeld, O.: Ddos never dies? an ixp perspective on ddos amplification attacks. In: International Conference on Passive and Active Network Measurement. pp. 284–301. Springer (2021)
18. Kou, Y., Lin, S., Fossorier, M.P.: Low-density parity-check codes based on finite geometries: a rediscovery and new results. IEEE Transactions on Information theory **47**(7), 2711–2736 (2001)
19. M. Sipser, D.S.: Expander codes. codes and complexity. IEEE Trans. Inform. Theory **42**(6), 1710–1722. (1996)
20. Ma, Q., Zheng, J., Li, S., Cottrell, G.W.: Learning representations for time series clustering. Advances in neural information processing systems **32** (2019)
21. MICHAEL, R.: Tanner: Explicit concentrators from generalized n-gons. SIAM Journal on Algebraic Discrete Methods **5**(3), 287–293 (1984)

22. RouteViews, O.: University of oregon routeviews project. Eugene, OR.[Online]. Available: http://www. routeviews. org (2013)
23. Rudman, L., Irwin, B.: Characterization and analysis of ntp amplification based ddos attacks. In: 2015 Information Security for South Africa (ISSA). pp. 1–5. IEEE (2015)
24. Sarnak, P.: What is . . . an expander? Notices of the American Mathematical Society **51**(7), 762–763 (Aug 2004)
25. Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B.: Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In: 2011 31st international conference on distributed computing systems workshops. pp. 166–171. IEEE (2011)
26. Spielman, D.A.: Linear-time encodable and decodable error-correcting codes. In: Proceedings of the twenty-seventh annual ACM symposium on Theory of computing. pp. 388–397 (1995)
27. Stanić, Z.: Graphs with small spectral gap. The Electronic Journal of Linear Algebra **26**, 417–432 (2013)
28. Tanner, R.M.: Minimum-distance bounds by graph analysis. IEEE Transactions on Information Theory **47**(2), 808–821 (2001)
29. Tran, W.: Spectral theory of graphons (2021)
30. Vontobel, P.: Constructions of ldpc codes using ramanujan graphs and ideas from margulis. In: Slides, CSL seminar series, UIUC (2001)
31. Yoachimik, O., Forster, A.: Cve-2022-26143: A zero-day vulnerability for launching udp amplification ddos attacks. `https://blog.cloudflare.com/cve-2022-26143-amplification-attack/`
32. Zémor, G.: On expander codes. IEEE Transactions on Information Theory **47**(2), 835–837 (2001)