Scalable Change Point Detection for Dynamic Graphs

Shenyang Huang
shenyang.huang@mail.mcgill.ca
Mila, McGill University
Montreal, Canada

Guillaume Rabusseau
guillaume.rabusseau@umontreal.ca
Mila & DIRO, Université de Montréal
CIFAR AI chair
Montreal, Canada

Reihaneh Rabbany
rrabba@cs.mcgill.ca
Mila, McGill University
CIFAR AI chair
Montreal, Canada

ABSTRACT

Real world networks often evolve in complex ways over time. Understanding anomalies in dynamic networks is crucial for applications such as traffic accident detection, intrusion identification and detection of ecosystem disturbances. In this work, we focus on the problem of change point detection in dynamic graphs. The goal is to identify time steps where the graph structure deviates significantly from the norm. Despite empirical success of recent methods, building a change point detection method for real world dynamic graphs, which often scale to millions of nodes, remains an open question. To fill this gap, we propose LADdos, a scalable method for change point detection in dynamic graphs. LADdos brings together ideas from two recent works: an accurate change point detection method for graphs called LAD [10] which detects the changes in the full Laplacian spectrum of the graph in each timestamp, and the general framework of network density of states (DOS) [5] which models the distribution of the singular values through efficient approximation methods. In experiments with two common graph models –the Stochastic Block Model (SBM) and the Barabási-Albert (BA) model – we show that LADdos has equal performance to LAD, which is the current state-of-the-art, while being orders of magnitude faster. For instance, on a dynamic graph with total 21 million edges over 150 timestamps, LADdos achieves 100x speedup when compared to LAD.

CCS CONCEPTS

• **Computing methodologies** → *Spectral methods*; *Temporal reasoning*.

KEYWORDS

Anomaly Detection, Dynamic Graphs, Spectral Methods

ACM Reference Format:
Shenyang Huang, Guillaume Rabusseau, and Reihaneh Rabbany. 2021. Scalable Change Point Detection for Dynamic Graphs. In *ODD '21: 6th Outlier Detection and Description Workshop at KDD 2021, August 15, 2021, Singapore.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/1122445.1122456

ODD '21, August 15, 2021, Singapore

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Dynamic graphs have emerged as a principled way to model complex real world relations that evolves over time. Anomaly Detection in dynamic graphs has attracted attention due to its broad application in social media misinformation identification [20], fraud information detection and abnormal clinical information identification [4]. In essence, anomaly detection in dynamic graph aims to pinpoint different types of time-varying anomalies which significantly deviate from the "normal" behavior.

In this work, we focus on detecting change points which identifies time steps where the graph structure or component deviates significantly from the normal behavior. We also consider the closely related anomaly type known as events [11]. Following the definition from [10], a *change point* is a time point where there is a significant adjustment in the graph generation process and these changes persists from this point onward. In contrast, an *event* is a time point where there is a sudden change in the overall network behavior and the structure returns to normal afterwards. Similar to [10], our proposed method is designed to identify both change points and events based on the Laplacian spectrum of the graph in different timestamps.

Many real world dynamic graphs contain thousands or millions of nodes per snapshot. At this scale, computing Singular Value Decomposition (SVD) to get the Laplacian spectrum as required in LAD could become computationally expensive or even infeasible. To address this challenge, we propose LADdos which uses the distribution of the eigenvalues instead of the the actual sequence and integrates the general framework of network density of states (DOS) [5] to model the distribution of the singular values through efficient approximation methods. Figure 1 shows how the DOS of a Stochastic Block Model (SBM) evolves when ten equal sized communities are merged into two. In both the SBM and Barabási-Albert (BA) model, we observe drastic changes in the DOS when the generative model parameter change, which motivates the use of DOS for change point detection.

Summary of contributions:

- We propose a scalable extension of the state-of-the-art method LAD, named LADdos, which uses the distribution of eigenvalues instead of the full Laplacian spectrum and utilizes the general framework of density of states to efficiently model the distribution of singular values of the graph Laplacian matrix.
- We show that LADdos is capable of operating on large graphs with 100x speed up when compared to LAD and achieve equally strong performance with very little computational cost. LADdos is also able to operate on dynamic graphs with sizes that are infeasible for LAD.

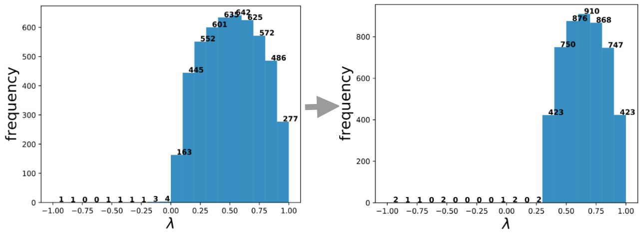

Figure 1: Changes in the density of states of the graph Laplacian reflects changes in the structure of the SBM model. This example plot shows the change when going from 10 equal sized communities (left) to 2 equal sized communities (right). Most notably, as the number of communities decreases, the intervals with high frequency of singular values λ changes from [0,1] to [0.25,1].

- We observe that the distribution of singular values on the graph Laplacian changes closely to the adjustment in the underlying generative model such as the SBM and the BA model. LADdos is able to captures these changes as effective as LAD for change point detection.

2 RELATED WORK

Anomalies in dynamic graphs can be broadly classified based on their role in the graph: anomalous nodes [21], edges [2, 22], subgraphs [6, 14, 18] and graphs [10, 11, 19]. There are also five categories of anomaly detection methods based on the survey by Ranshous et al. [16]: community based, compression based, decomposition based, distance based and probabilistic model based methods. The common strategy across all these methods is to extract a low dimensional representation from graph snapshots and then apply an anomaly scoring function to compare these representations.

2.1 Event Detection

Idé and Kashima [11] aim to find time points where the majority of the edge attributes in the network show significant deviation from the recent ones. The principal eigenvector corresponding to the maximum eigenvalue of the positive weighted adjacency matrix W is used as a low dimensional representation of the graph (called *activity vector*). Different from Idé and Kashima, we use the density of states of the graph Laplacian matrix to summarize graph structures and motifs. Koutra et al. [12] first formulate dynamic graphs as high order tensors and propose to use the PARAFAC decomposition [3, 7] to obtain vector representations for anomaly scoring.

2.2 Change Point Detection

Koutra et al. [13] proposed DeltaCon as a novel graph similarity function for anomalous graph snapshot detection. DeltaCon first computes pairwise node affinities in the first graph and then measures the difference in node affinity score of the two graphs. Therefore, DeltaCon is analogous to an anomaly scoring function. However, it is not straightforward how to extend DeltaCon to temporal reasoning for a sequence of graphs, as it is designed to operate between pairs of graphs.

Peel and Clauset [15] described the change point detection problem as identifying the times at which the large-scale patterns of interaction change fundamentally. Their proposed LetoChange method relies on an appropriate choice of a parametric family of probability distribution which describes the data. After which a Bayesian hypothesis test is used to accept or reject if a parameter change has occurred in the model.

Wang et al. [19] model network evolution as a first order Markov process thus designed their EdgeMonitoring method based on MCMC sampling theory. They assume some unknown underlying model that governs the generative process. Each graph snapshot is also assumed to be dependent on the current generative model as well as the previously observed snapshot (through resampling a portion of the edges at each step).

Recently, Huang et al. [10] proposed LAD which has two major differences with prior methods. First, LAD uses the singular values of the Laplacian matrix of each snapshot as the signature vector which closely relates to many structural properties in the graph and spectral graph theory. Second, LAD designs two sliding windows to model both the short term and long term temporal changes therefore it is able to detect both events (sudden changes) and change points (gradual changes). Building upon this work, in our proposed LADdos, first we use the distribution of the eigenvalues of the normalized Laplacian matrix as the signature vector instead of their actual values and achieve a similar performance as LAD. Second we compute these signature vectors using the recently proposed network density of states [5] which allows the LADdos to scale to graphs with millions of nodes.

2.3 Network Density of States

Dong et al. [5] demonstrate the importance to model the overall distribution of eigenvalues, the *spectral density* or *density of states (DOS)*, for common graph matrices such as the adjacency or the Laplacian matrix. They borrowed tools from condensed matter physics and added adaptation such as motif filtering to design an efficient approximation method for DOS in large scale real world networks. Through compelling visual fingerprints of graphs, Dong et al. showed that DOS facilitates the computation of many common centrality measures.

A recent work by Huang et al. [9] proposed a novel graph kernel which combines local and global density of states of the normalized adjacency matrix. They revealed that the global DOS captures the probability that a fixed length random walk returns to a random node in the graph. The local DOS then measures the probability that such random walk returns to a fixed node. By utilizing both global and local DOS, their proposed graph kernal is able to achieve strong performance on graph classification benchmarks. In this work, we consider the global DOS for the normalized Laplacian matrix which captures global structural information of a graph snapshot thus highly suitable for change point detection.

3 PROBLEM DEFINITION

3.1 Dynamic Graph

Let the interval of interest be from timestamp 1 to T. A corresponding set of graph snapshots G is written as $\{\mathcal{G}_t\}_{t=1}^{T}$, where each $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ represents the static graph at timestamp t. $\mathcal{V}_t$ and

$\mathcal{E}_t$ are the set of nodes and edges respectively. Define an edge $e = (i, j, w) \in \mathcal{E}_t$ as the connection between node i and node j at timestamp t in the dynamic graph with weight w. By convention, $w = 1$ for all edges in unweighted graphs and $w \in \mathsf{R}^+$ for weighted graphs. We use an adjacency matrix $\mathcal{A}_t \in \mathsf{R}^{n\times n}$ to represent edges in $\mathcal{E}_t$ where $n = |\mathcal{V}_t|$. It is often assumed that the number of nodes in the graph is constant for all time steps [11, 12, 21, 22]. Note that our proposed LADdos examines properties of each snapshot from a global view thus adding or removing a small number of nodes won't affect the performance of LADdos.

3.2 Change Point Detection

Based on the above formulation, the goal is to find anomalous graphs $\mathcal{G}_t$ in G. Given an anomaly scoring function $f : \mathcal{G}_t \rightarrow \mathsf{R}$, find time steps t such that $|f(\mathcal{G}_t) - f(\mathcal{G}_N)| > \delta$ or $|f(\mathcal{G}_t) - f(\mathcal{G}_W)| > \epsilon$ where $\mathcal{G}_N$ is the normal behavior of the graph in the global context, $\mathcal{G}_W$ is the short term behavior of the graph in recent context window W and δ, ϵ are thresholds. In general, the anomaly scoring function should clearly differentiate anomalous points from normal ones and assign higher anomaly scores to more anomalous points.

4 LAPLACIAN ANOMALY DETECTION

In this section, a brief overview of the LAD method is provided as background, more details can be found in [10]. The core idea of LAD is to detect high level graph changes from low dimensional embeddings (called signature vectors). Then the "typical" or "normal" behavior of the graph can be extracted from a stream of signature vectors based on both short term and long term dependencies. In this way, one can compare the deviation of current signature vector from the normal behavior.

4.1 Laplacian Spectrum

In LAD, the Laplacian spectrum is used as the signature vector summarizing each graph snapshot into a low dimensional embedding. The (unnormalized) Laplacian matrix L_t is defined as $\mathsf{L}_t = \mathsf{D}_t - \mathsf{A}_t$ where D_t is the diagonal degree matrix and A_t is the adjacency matrix of $\mathcal{G}_t$. In addition, we define the symmetric normalized Laplacian L_{sym} as,

$$\mathsf{L}_{sym} = \mathsf{D}^{-\frac{1}{2}}\mathsf{L}\mathsf{D}^{-\frac{1}{2}} = \mathsf{I} - \mathsf{D}^{-\frac{1}{2}}\mathsf{A}\mathsf{D}^{-\frac{1}{2}} \tag{1}$$

In this work, the symmetric normalized Laplacian L_{sym} is used for both LAD and LADdos. We find this results in better performance compared to the unnormalized version originally used in LAD. In this way, the Laplacian matrix is normalized by node degree while also being symmetric.

Characterize the Normal Behavior LAD computes a "typical" or "normal" behavior vector from the previous w signature vectors where w is the sliding window size. First, *L2* normalization is performed on the aggregated Laplacian spectrum seen so far $\Sigma_0, \ldots, \Sigma_t$ to obtain unit vectors. Next, a context matrix C is constructed:

$$\mathsf{C} = \begin{pmatrix} | & | & & | \\ \Sigma_{t-w-1} & \Sigma_{t-w-2} & \ldots & \Sigma_{t-1} \\ | & | & & | \end{pmatrix} \in \mathsf{R}^{k\times w} \tag{2}$$

Algorithm 1: LADdos

Input: dynamic graph G

Hyper-parameter: sliding window sizes w_s, w_l, N_z probe vectors, N_m Chebyshev moments

Output: Final anomaly scores Z^*

foreach *graph snapshot* $\mathsf{G}_t \in \mathsf{G}$ **do**
- Compute L_{sym} (see Eq. (1));
- Compute the DOS approximation of L_{sym} using N_z probing vectors and N_m Chebyshev moments;
- Obtain the frequency vector σ_t for each singular value interval from DOS ;
- Perform L2 normalization on σ_t;
- Compute left singular vector $\tilde{\sigma}_t^{w_s}$ of context $\mathsf{C}_t^{w_s} \in \mathsf{R}^{k\times w_s}$ (see Eq. (2));
- Compute left singular vector $\tilde{\sigma}_t^{w_l}$ of context $\mathsf{C}_t^{w_l} \in \mathsf{R}^{k\times w_l}$(see Eq. (2));
- $Z_t^{w_s} = 1 - \sigma_t^\top \tilde{\sigma}_t^{w_s}$;
- $Z_t^{w_l} = 1 - \sigma_t^\top \tilde{\sigma}_t^{w_l}$;

end

foreach *time step t* **do**
- $Z_{s,t}^* = max(Z_{w_s,t} - Z_{w_s,t-1}, 0)$;
- $Z_{l,t}^* = max(Z_{w_l,t} - Z_{w_l,t-1}, 0)$;
- $Z_t^* = max(Z_{w_s,t}^*, Z_{w_l,t}^*)$;

end

Return Z^*;

where k is length of the signature vector. The left singular vector of C is computed to obtain the normal behavior vector $\tilde{\Sigma}_t^w$. Then, a short term sliding window of size w_s and a long term sliding window of size w_l are used to detect both events and change points.

Infer the Anomaly Score The anomaly score is denoted by Z. Let $\tilde{\Sigma}_t^w$ be the normal behavior vector and $\boldsymbol{\Sigma}_t$ be the normalized Laplacian spectrum at current step. The Z score is computed as:

$$Z = 1 - \frac{\boldsymbol{\Sigma}_t^\top \tilde{\Sigma}_t^w}{\|\boldsymbol{\Sigma}_t\|_2 \|\tilde{\Sigma}_t^w\|_2} = 1 - \boldsymbol{\Sigma}_t^\top \tilde{\Sigma}_t^w = 1 - \cos\theta, \tag{3}$$

where $\cos\theta$ is the cosine similarity between vectors $\boldsymbol{\Sigma}_t$ and $\tilde{\Sigma}_t^w$. Essentially, the Z scores becomes closer to 1 when the current spectrum is very dissimilar to the normal, thus having a high likelihood of being an anomaly. The Z scores from different sliding windows are then aggregated by the max operation. Lastly, the jumps in anomaly score are emphasized by using $Z_t^* = min(Z_t - Z_{t-1}, 0)$. The points with the largest Z^* are then selected as anomalies.

5 LAD WITH DENSITY OF STATES

To scale LAD to real world graphs with millions of nodes, we propose LAD with Density of States (LADdos) which uses the density of states of the normalized Laplacian matrix as the signature vector instead of the whole Laplacian spectrum in LAD. When applied on large graphs, LADdos is *1).* fast to compute even for graph sizes infeasible with SVD and *2).* obtains a fixed dimension signature vector independent of the graph size. The procedures in LADdos is detailed in Algorithm 1.

5.1 Density of States

Let $\mathsf{H} \in \mathsf{R}^{N \times N}$ be any symmetric graph matrix with eigendecomposition $\mathsf{H} = \mathsf{Q}\Lambda\mathsf{Q}^T$ where $\Lambda = diag(\lambda_1, \dots, \lambda_N)$ and Q is an orthogonal matrix. The density of states or spectral density induced by H is defined as:

$$\mu(\lambda) = \frac{1}{N} \sum_{i=1}^{N} \delta(\lambda - \lambda_i) \tag{4}$$

where δ is the Kronecker delta. In this work, we focus on the DOS of the normalized Laplacian matrix L_{sym} (see Equation 1). Note that in directed and asymmetric networks, the eigenvalues in Equation 4 can be extended to singular values and our discussion is centered around the more general singular values.

When examining a dynamic graph with millions of nodes, it is natural to model the distribution of singular values rather than individual values. We propose to leverage the Kernel Polynomial Method (KPM) described in [5] to approximate the DOS through an expansion in the dual basis of the Chebyshev basis $\{T_m\}$. First, as required by Chebyshev approximation, the spectrum needs to be rescaled to the interval $[-1, 1]$ for numerical stability. This is achieved as follows:

$$\tilde{\mathsf{L}} = \frac{2\mathsf{L} - (\lambda_{max}(\mathsf{L}) + \lambda_{min}(\mathsf{L}))}{\lambda_{max}(\mathsf{L}) - \lambda_{min}(\mathsf{L})} \tag{5}$$

Next, let $T_m(\mathsf{L})$ be the mth Chebyshev polynomial of the Laplacian L and $T_m^*(x) = w(x)T_m(x)$. Then the DOS $\mu(\lambda)$ can be expressed as:

$$\mu(\lambda) = \sum_{m=0}^{\infty} d_m T_m^*(\lambda) \tag{6}$$

$$d_m = \int_{-1}^{1} T_m(\lambda)\mu(\lambda)d\lambda = \frac{1}{N}\sum_{i=1}^{N} T_m(\lambda_i) = \frac{1}{N} trace(T_m(\mathsf{L})) \tag{7}$$

In practice, we calculate a finite number of Chebyshev moments for the approximation. Now we want to efficiently extract the diagonal elements of the matrices $\{T_m(L)\}$ without explicitly forming the matrix. To achieve that, stochastic trace estimation can be used which utilizes N_z random probe vectors and estimates the trace:

$$trace(\mathsf{L}) = \mathsf{E}[\mathsf{z}^T \mathsf{L}\mathsf{z}] \simeq \frac{1}{N_z}\sum_{j=1}^{N_z} \mathsf{z}_j^T \mathsf{L}\mathsf{z} \tag{8}$$

5.2 Computational Complexity

Overall, the computational cost of estimating the DOS is $O(|E|N_z N_m)$ where $|E|$ is the number of edges in a graph snapshot, N_z is the number of probe vectors and N_m is the number of chebychev moments used for the approximation. Thus, LADdos has complexity $O(T|E|N_z N_m)$ for dynamic graphs with T steps. In comparison, the most expensive step in LAD is the computation of the singular values of the graph Laplacian matrix. The computational complexity for full SVD is $O(m^2 n)$, for a matrix $\mathsf{M} \in \mathsf{R}^{m \times n}$ where $m \leq n$. Thus the computational cost for LAD could be as high as $O(Tm^2 n)$.

Events & Change Points SBM Model parameters				
Time Point	Type	N_c	p_{in}	p_{out}
0	start point	4	0.030	0.005
16	event	4	0.030	**0.015**
31	change point	**10**	0.030	0.005
61	event	10	0.030	**0.015**
76	change point	**2**	0.030	0.005
91	event	2	0.030	**0.015**
106	change point	**4**	0.030	0.005
136	event	4	0.030	**0.015**

Table 1: The changes in generative models in section 6.1 where a combination of events and change points are observed.

6 EXPERIMENTS

In this section, to study the performance and efficiency of LADdos on large graphs, we conduct experiments in two controlled settings with the SBM and the BA model. For both experiments, we compare with the original LAD implementation and set the short term and the long term window to be 5 and 10 time steps respectively. For all experiments, we also set the number of probing vectors $N_z = 20$ and the number of Chebychev moments $N_m = 50$ as parameters for DOS.

Performance Metrics Similar to [10], we use Hits at n ($H@n$) metric which reports the number of correctly identified time points out of top n steps with the highest anomaly scores. We also report the execution time in seconds on a desktop with AMD Ryzen 5 1600 CPU and 16 GB memory. We use the ground truth labels planted in the synthetic generation process for evaluation. The experimental results are summarized in Table 3. Note that on SBM-7k and Ba-7k experiments, the computational time for LAD is too costly with the above resource set-up thus indicated as N/A.

6.1 SBM Experiments

The first set of experiment is conducted with the Stochastic Block Model (SBM) [8]. SBM is a widely used graph generation model with an emphasis on community structure [1]. The key parameters of the SBM model are: 1). the partitioning of communities, 2). the intra-community connectivity p_{in} and 3). the cross-community connectivity p_{out}. For simplicity, we assume equal sized communities and instead focus on changing the number of communities N_c. p_{in} and p_{out} determines the probability of an edge existing between nodes of the same community and different communities respectively. They also control the sparsity of the dynamic graph.

We consider both change points and events similar to the difficult Hybrid setting in [10]. We experiment with 1k nodes and 5k nodes per snapshot and focus our discussion on results with 5k nodes here. Change points corresponds to communities either merging or splitting and events corresponds to temporary boost in cross-community connectivity p_{out}. The details of these anomalies are shown in Table 1. Figure 2 shows that LADdos perfectly identifies all the events and change points on a dynamic SBM graph with 5000 nodes. We also visualize the signature vectors (the number of singular values in each interval) as a heatmap. The events (time point 16,61,91,136) corresponds to an energetic burst in the signature vector. Figure 1 shows the change in singular value distribution at

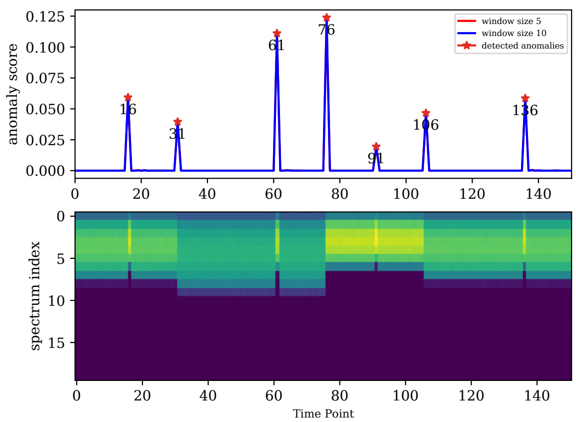

Figure 2: LADdos perfectly recovers all events and change points in the SBM model, experiment settings defined in Table 1 on a dynamic graph with 5k nodes.

BA Model parameters		
Time Point	Type	m
0	start point	1
16	change point	2
31	change point	3
61	change point	4
76	change point	5
91	change point	6
106	change point	7
136	change point	8

Table 2: The change points for BA model. m is the number of edges to attach from a new node to existing nodes. The increased color intensity in the table indicates the increased density of the network.

time point 76 where less communities results in a narrower distribution. We also observe the reverse behavior where as the number of community increases, the singular value distribution becomes broader. Overall, Table 3 shows that LADdos can achieve similar performance to LAD with a fraction of the computational time.

6.2 BA Experiments

We further investigate LADdos performance in the Barabási-Albert (BA) model to show that LADdos is effective beyond the SBM model. In this experiment, the change points correspond to the densification of the network (parameter m, increased number of edges attached from a new node to an existing node). The details are described in Table 2.

Figure 4 shows that LADdos is able to correctly detect all change points in the BA model. The signature vectors or the DOS are visualized as a heatmap. Interestingly, as graph gets increasingly dense over time, it is visually harder to see the changes in the signature vectors at anomalous points. In addition, the anomaly score assigned for the later change points are also smaller in magnitude. Table 3 shows that LADdos is again performing as well as LAD while being much lower in computational cost.

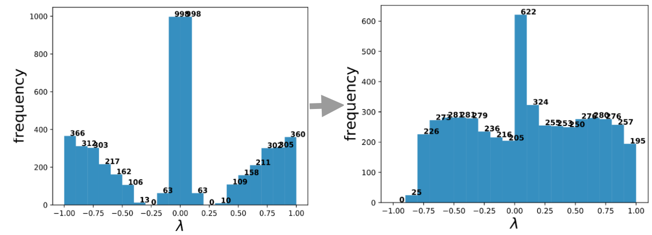

Figure 3: The change in DOS for BA model at time step 16. Left is BA model with $m = 1$ and right is BA model with $m = 2$.

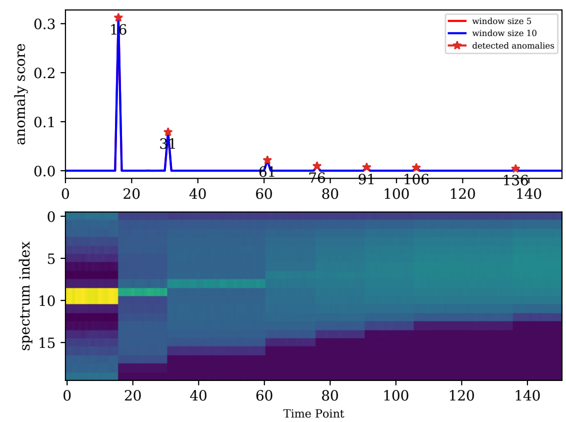

Figure 4: LADdos perfectly recovers all events and change points for the BA model, experiment setup explained in Table 2.

Dataset	SBM			BA		
# nodes	1k	5k	7k	1k	5k	7k
total edges	871k	21824k	42769k	687k	3451k	4833k
Metric	Hits @ 7					
LADdos	85.7%	100%	**100%**	100%	**100%**	**100%**
LAD [10]	**100%**	100%	N/A	100%	85.7%	N/A
Metric	Execution Time (sec)					
LADdos	**19.1**	**185.9**	**362.6**	**19.2**	**105.3**	**175.7**
LAD [10]	76.9	14934.8	N/A	78.1	15024.4	N/A

Table 3: LADdos can operate on large graphs while maintaining the same performance as LAD. Each dynamic graph has 151 time steps.

Figure 3 visualizes the change in the distribution of singular values of the Laplacian as approximated in DOS. For this particular change point, we observe drastic changes in the overall shape of DOS. Therefore, by reasoning with the changes in DOS, LADdos is able to correctly captures change points in the BA model experiments.

7 DISCUSSION

Both LAD and LADdos models the changes in the distribution of singular values of the Laplacian. As shown in Table 3, LADdos can correctly detect change points in large scale dynamic graphs and even outperforms LAD in BA-5k experiment. As LADdos is an approximation method, and it is still more accurate to compute the exact SVD for small graphs (i.e. number of nodes smaller than 1000). For large graphs, the DOS captures the distribution of singular values as a form of spectral signature [5]. When compared to LAD, LADdos achieves equal or better performance with only 1.0% and 0.7% of the computational time on SBM-5k and BA-5k experiment respectively.

Recently, Rubin [17] showed that the spectral embedding of existing random graph models including graphons and other latent position models should live close to a one-dimensional structure. This suggests that the spectrum of an arbitarily large graph sampled from a random graph model could be embedded into a low dimensional vector. As shown in Figure 1 and 3, the change in the distribution of singular values correspond closely to changes in the generative model. Therefore, the DOS could be a natural way to obtain the aforementioned low dimensional embedding. One promising future direction would be to examine how the distribution of singular values of the Laplacian matrix corresponds to parameter changes in the random graph model.

8 CONCLUSION

To address the scalability challenge of existing change point detection methods for dynamic graphs, we proposed LADdos, which integrates the general framework of density of states (DOS) [5] as a fast and efficient low dimensional embedding vector for each graph snapshot. In experiments with the Stochastic Block Model (SBM) and the Barabási-Albert (BA) model, we show that LADdos has equal performance to the state-of-the-art LAD while achieving 100x speedup.

ACKNOWLEDGMENTS

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canadian Institute for Advanced Research (CIFAR AI Chairs program).

REFERENCES

[1] Emmanuel Abbe. 2017. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research* 18, 1 (2017), 6446–6531.

[2] Siddharth Bhatia, Arjit Jain, Pan Li, Ritesh Kumar, and Bryan Hooi. 2020. MStream: Fast Anomaly Detection in Multi-Aspect Streams. *arXiv preprint arXiv:2009.08451* (2020).

[3] Rasmus Bro. 1997. PARAFAC. Tutorial and applications. *Chemometrics and intelligent laboratory systems* 38, 2 (1997), 149–171.

[4] Zhengzhang Chen, William Hendrix, and Nagiza F Samatova. 2012. Community-based anomaly detection in evolutionary networks. *Journal of Intelligent Information Systems* 39, 1 (2012), 59–85.

[5] Kun Dong, Austin R Benson, and David Bindel. 2019. Network density of states. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1152–1161.

[6] Dhivya Eswaran, Christos Faloutsos, Sudipto Guha, and Nina Mishra. 2018. Spotlight: Detecting anomalies in streaming graphs. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1378–1386.

[7] Richard A Harshman et al. 1970. Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multimodal factor analysis. (1970).

[8] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. 1983. Stochastic blockmodels: First steps. *Social networks* 5, 2 (1983), 109–137.

[9] Leo Huang, Andrew J Graven, and David Bindel. 2021. Density of States Graph Kernels. In *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*. SIAM, 289–297.

[10] Shenyang Huang, Yasmeen Hitti, Guillaume Rabusseau, and Reihaneh Rabbany. 2020. Laplacian Change Point Detection for Dynamic Graphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 349–358.

[11] Tsuyoshi Idé and Hisashi Kashima. 2004. Eigenspace-based anomaly detection in computer systems. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 440–449.

[12] Danai Koutra, Evangelos E Papalexakis, and Christos Faloutsos. 2012. Tensorsplat: Spotting latent anomalies in time. In *2012 16th Panhellenic Conference on Informatics*. IEEE, 144–149.

[13] Danai Koutra, Neil Shah, Joshua T Vogelstein, Brian Gallagher, and Christos Faloutsos. 2016. Deltacon: Principled massive-graph similarity function with attribution. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 10, 3 (2016), 1–43.

[14] Misael Mongiovi, Petko Bogdanov, Razvan Ranca, Evangelos E Papalexakis, Christos Faloutsos, and Ambuj K Singh. 2013. Netspot: Spotting significant anomalous regions on dynamic networks. In *Proceedings of the 2013 Siam international conference on data mining*. SIAM, 28–36.

[15] Leto Peel and Aaron Clauset. 2015. Detecting change points in the large-scale structure of evolving networks. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[16] Stephen Ranshous, Shitian Shen, Danai Koutra, Steve Harenberg, Christos Faloutsos, and Nagiza F Samatova. 2015. Anomaly detection in dynamic networks: a survey. *Wiley Interdisciplinary Reviews: Computational Statistics* 7, 3 (2015), 223–247.

[17] Patrick Rubin-Delanchy. 2020. Manifold structure in graph embeddings. *Advances in Neural Information Processing Systems* 33 (2020).

[18] Kijung Shin, Bryan Hooi, Jisu Kim, and Christos Faloutsos. 2017. Densealert: Incremental dense-subtensor detection in tensor streams. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1057–1066.

[19] Yu Wang, Aniket Chakrabarti, David Sivakoff, and Srinivasan Parthasarathy. 2017. Fast change point detection on dynamic social networks. *arXiv preprint arXiv:1705.07325* (2017).

[20] Rose Yu, Huida Qiu, Zhen Wen, ChingYung Lin, and Yan Liu. 2016. A survey on social media anomaly detection. *ACM SIGKDD Explorations Newsletter* 18, 1 (2016), 1–14.

[21] Wenchao Yu, Wei Cheng, Charu C Aggarwal, Kai Zhang, Haifeng Chen, and Wei Wang. 2018. Netwalk: A flexible deep embedding approach for anomaly detection in dynamic networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2672–2681.

[22] Li Zheng, Zhenpeng Li, Jian Li, Zhao Li, and Jun Gao. 2019. Addgraph: anomaly detection in dynamic graph using attention-based temporal GCN. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 4419–4425.