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

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 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.
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 RELATED WORK

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 kernel 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 \( t_1 \) to \( T \). A corresponding set of graph snapshots \( G \) is written as \( \{G_t\}_{t=1}^{T} \), where each \( G_t = (V_t, E_t) \) represents the static graph at timestamp \( t \). \( V_t \) 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 \( w \in \mathbb{R}^+ \) for weighted
graphs. We use an adjacency matrix \( A_t \in \mathbb{R}^{n \times n} \) to represent edges in
\( \mathcal{E}_t \) where \( n = |V_t| \). 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 \( G_t \) in \( G \). Given an anomaly scoring function \( f : \mathcal{G}_t \rightarrow \mathbb{R} \), find time steps \( t \) such that \( |f(G_t) - f(G_N)| > \delta \) or \( |f(G_t) - 
f(G_N)| > \epsilon \) where \( G_N \) is the normal behavior of the graph in the
global context, \( G_N \) is the short term behavior of the graph in recent
context window \( W \) and \( \delta, \epsilon \) 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 em-
beddings (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 sum-
marizing each graph snapshot into a low dimensional embedding.
The (unnormalized) Laplacian matrix \( L_t \) is defined as \( L_t = D_t - A_t \)
where \( D_t \) is the diagonal degree matrix and \( A_t \) is the adjacency
matrix of \( G_t \). In addition, we define the symmetric normalized
Laplacian \( L_{sym} \) as,

\[
L_{sym} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}} = I - D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \quad (1)
\]

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, \( L_2 \) normalization is per-
formed on the aggregated Laplacian spectrum seen so far \( \Sigma_0, \ldots, \Sigma_t \)
to obtain unit vectors. Next, a context matrix \( C \) is constructed:

\[
C = \begin{bmatrix} \Sigma_{t-w-1} & \Sigma_{t-w-2} & \ldots & \Sigma_{t-1} \end{bmatrix} \in \mathbb{R}^{w \times k} \quad (2)
\]

where \( k \) is length of the signature vector. The left singular vector
of \( C \) is computed to obtain the normal behavior vector \( \tilde{\Sigma}_t \). Then,
a short term sliding window of size \( w_1 \) and a long term sliding
window of size \( w_2 \) are used to detect both events and change points.

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

\[
Z = 1 - \frac{\Sigma_t^T \tilde{\Sigma}_t}{||\Sigma_t||_2 ||\tilde{\Sigma}_t||_2} = 1 - \Sigma_t^T \tilde{\Sigma}_t = 1 - \cos \theta \quad (3)
\]

where \( \cos \theta \) is the cosine similarity between vectors \( \Sigma_t \) and \( \tilde{\Sigma}_t \).

**Algorithm 1:** LADdos

**Input:** dynamic graph \( G \)

**Hyper-parameter:** sliding window sizes \( w_1, w_2 \), \( N_2 \) probe
vectors, \( N_m \) Chebyshev moments

**Output:** Final anomaly scores \( Z^* \)

1. `foreach graph snapshot \( G_t \in G \) do`
2. \( \text{Compute } L_{sym} \) (see Eq. (1));
3. \( \text{Compute } \text{the DOS approximation of } L_{sym} \) using \( N_2 \)
probing vectors and \( N_m \) Chebyshev moments;
4. \( \text{Obtain the } \text{frequency vector } \sigma_t \) for each singular value
interval from DOS ;
5. \( \text{Perform } L_2 \) normalization on \( \sigma_t \);
6. \( \text{Compute left singular vector } \sigma_{1}^{w_1} \) of context
\( C_{1}^{w_1} \in \mathbb{R}^{k \times w_1} \) (see Eq. (2));
7. \( \text{Compute left singular vector } \sigma_{1}^{w_2} \) of context
\( C_{1}^{w_2} \in \mathbb{R}^{k \times w_2} \) (see Eq. (2));
8. \( Z_{1}^{w_1} = 1 - \sigma_{1}^T \sigma_{1}^{w_1} \);
9. \( Z_{1}^{w_2} = 1 - \sigma_{1}^T \sigma_{1}^{w_2} \);

10. `end`

11. `foreach time step \( t \) do`
12. \( Z_{1}^{w_1,t} = \max(Z_{w_1,t} - Z_{w_1,t-1}, 0) \);
13. \( Z_{1}^{w_2,t} = \max(Z_{w_2,t} - Z_{w_2,t-1}, 0) \);
14. \( Z_{1}^{*} = \max(Z_{1}^{w_1,t} - Z_{1}^{w_2,t}) \);
15. `end`
16. `Return \( Z^* \)`.

To scale LAD to real world graphs with millions of nodes, we pro-
pose 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 $H \in \mathbb{R}^{N \times N}$ be any symmetric graph matrix with eigendecomposition $H = Q \Lambda Q^T$ where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_N)$ and $Q$ is an orthogonal matrix. The density of states or spectral density induced by $H$ is defined as:

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

where $\delta$ is the Kronecker delta. In this work, we focus on the DOS of the normalized Laplacian matrix $L_{norm}$ (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:

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

Next, let $T_m(L)$ be the $m$th Chebyshev polynomial of the Laplacian $L$ and $T_m(x) = w(x)T_m(x)$. Then the DOS $\mu(\lambda)$ can be expressed as:

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

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

$$\text{trace}(L) = \mathbb{E}[z^T L z] \approx \frac{1}{N_z} \sum_{j=1}^{N_z} z_j^T L z$$ (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 $M \in \mathbb{R}^{m \times n}$ where $m \leq n$. Thus the computational cost for LAD could be as high as $O(Tm^2 n)$.

<table>
<thead>
<tr>
<th>Time Point</th>
<th>Type</th>
<th>$N_z$</th>
<th>$P_{in}$</th>
<th>$P_{out}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>start point</td>
<td>4</td>
<td>0.030</td>
<td>0.005</td>
</tr>
<tr>
<td>16</td>
<td>event</td>
<td>4</td>
<td>0.030</td>
<td>0.015</td>
</tr>
<tr>
<td>31</td>
<td>change point</td>
<td>10</td>
<td>0.030</td>
<td>0.005</td>
</tr>
<tr>
<td>61</td>
<td>event</td>
<td>10</td>
<td>0.030</td>
<td>0.015</td>
</tr>
<tr>
<td>76</td>
<td>change point</td>
<td>2</td>
<td>0.030</td>
<td>0.005</td>
</tr>
<tr>
<td>91</td>
<td>event</td>
<td>2</td>
<td>0.030</td>
<td>0.015</td>
</tr>
<tr>
<td>106</td>
<td>change point</td>
<td>4</td>
<td>0.030</td>
<td>0.005</td>
</tr>
<tr>
<td>136</td>
<td>event</td>
<td>4</td>
<td>0.030</td>
<td>0.015</td>
</tr>
</tbody>
</table>

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
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.

### 7 DISCUSSION

Both LAD and LADdos models the changes 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.
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 arbitrarily 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