

HLSAD: Hodge Laplacian-based Simplicial Anomaly Detection

Florian Frantzen
 RWTH Aachen University
 Department of Computer Science
 Aachen, Germany
 florian.frantzen@cs.rwth-aachen.de

Michael T. Schaub
 RWTH Aachen University
 Department of Computer Science
 Aachen, Germany
 schaub@cs.rwth-aachen.de

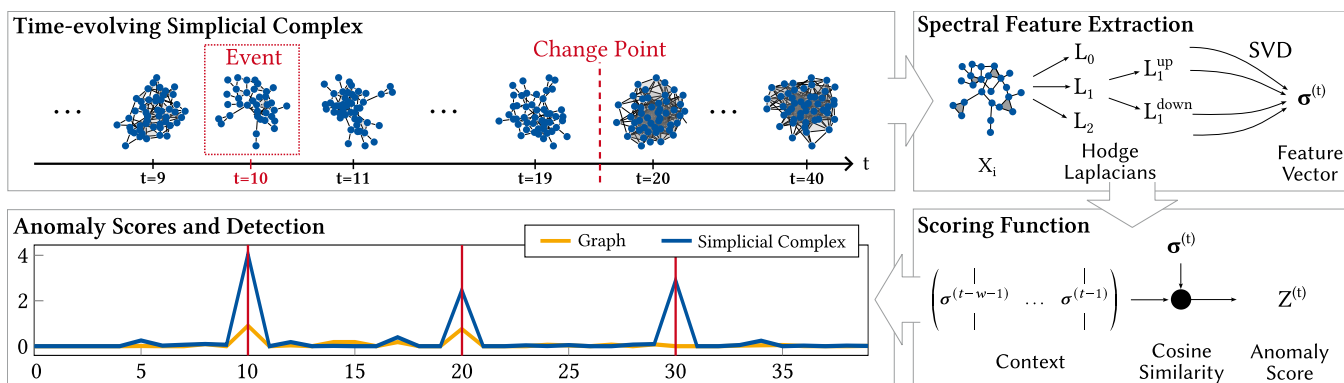


Figure 1: Illustrative process of anomaly detection on a toy-example simplicial complex sequence. Starting from a time-evolving simplicial complex, we compute the Hodge Laplacians for each snapshot and extract spectral features for each. We employ a sliding window approach to identify anomalies by evaluating the deviation of current spectral features from established temporal patterns. In this illustrative example, HLSAD successfully detects all three artificially introduced anomalies, whereas traditional graph-based analysis fails to identify the final anomaly that manifests in higher-order interactions.

Abstract

In this paper, we propose HLSAD, a novel method for detecting anomalies in time-evolving simplicial complexes. While traditional graph anomaly detection techniques have been extensively studied, they often fail to capture changes in higher-order interactions that are crucial for identifying complex structural anomalies. These higher-order interactions can arise either directly from the underlying data itself or through graph lifting techniques. Our approach leverages the spectral properties of Hodge Laplacians of simplicial complexes to effectively model multi-way interactions among data points. By incorporating higher-dimensional simplicial structures into our method, our method enhances both detection accuracy and computational efficiency. Through comprehensive experiments on both synthetic and real-world datasets, we demonstrate that our approach outperforms existing graph methods in detecting both events and change points.

CCS Concepts

• **Mathematics of computing** → *Graph theory; Topology*; • **Computing methodologies** → **Anomaly detection**; *Temporal reasoning; Spectral methods*; • **Theory of computation** → *Dynamic graph algorithms*.



This work is licensed under a Creative Commons Attribution 4.0 International License. *KDD '25, Toronto, ON, Canada*

© 2025 Copyright held by the owner/author(s).
 ACM ISBN 979-8-4007-1454-2/2025/08
<https://doi.org/10.1145/3711896.3736998>

Keywords

anomaly detection, change point detection, event detection, simplicial complex, spectral methods

ACM Reference Format:

Florian Frantzen and Michael T. Schaub. 2025. HLSAD: Hodge Laplacian-based Simplicial Anomaly Detection. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25)*, August 3–7, 2025, Toronto, ON, Canada. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3711896.3736998>

KDD Availability Link:

The source code of this paper has been made publicly available at <https://doi.org/10.5281/zenodo.15511585> and at <https://git.rwth-aachen.de/netsci/hlsad>.

1 Introduction

Anomaly detection within time-evolving networks has widespread applications across diverse domains, including network security for detecting intrusions and system faults [6, 22, 31], social network analysis for identifying collective behavioral changes induced by external factors [27, 34], and ecological monitoring for detecting ecosystem disturbances [5]. Accordingly, many graph-based techniques have been developed for identifying events and change points in dynamic networks [2, 29, 35]. Despite their utility, graph-based approaches are however limited by their focus on dyadic relationships, and fail to capture higher-order interactions that can

be essential for identifying complex structural anomalies in the underlying system. Such higher-order data can manifest itself through different mechanisms:

Higher-Order Data Many real-world datasets inherently contain multi-way interactions that extend beyond dyadic relationships. Traditional graph-based methods are fundamentally limited in their ability to capture and analyze these higher-order structures. For instance, during the COVID-19 pandemic, social distancing measures disproportionately affected large group gatherings compared to pairwise interactions, creating structural changes that are more readily detectable through higher-order methods [28].

Lifting Even when data only contains pairwise relations between entities, we can employ graph lifting techniques to construct a simplicial complex representation. A common approach identifies cliques in the graph with equivalent simplices. The effectiveness of lifting has been demonstrated in various contexts; notably, the Weisfeiler-Lehman isomorphism test exhibits strictly greater discriminative power when applied to clique complexes compared to their graph counterparts [3].

Building upon the graph anomaly detection frameworks developed by Akoglu and Faloutsos [1] and Huang et al. [21], we propose a novel method that leverages simplicial complexes to capture and analyze multi-way interactions in temporal data sequences. It is important to distinguish our focus on temporal anomaly detection in time-evolving networks from the related but distinct problem of identifying anomalous substructures within static networks, as these two problems address fundamentally different analytical objectives despite their similar names.

The incorporation of higher-order interactions via simplices in our method provides significant advantages over traditional graph-based approaches: First, it enhances detection accuracy through the explicit modeling of higher-order interactions that are inherently absent in graph representations. These higher-order structures are particularly relevant in domains such as epidemic modeling, where large group gatherings may exhibit distinct response patterns to intervention measures compared to dyadic interactions. Second, even if we only have pairwise interaction data, the application of graph lifting techniques results in improved anomaly detection capabilities. Our experimental results indicate that our method achieves superior accuracy when applied to lifted graph datasets compared to conventional spectral approaches on graph skeletons and can even offer enhanced computational efficiency.

HLSAD operates through the following systematic process: For each snapshot within a temporal sequence of simplicial complexes, we compute the corresponding Hodge Laplacians up to a user-specified order. Subsequently, we extract the principal eigenvalues from both up and down Hodge Laplacians and concatenate these spectral features into a comprehensive feature vector for each temporal instance. Employing a sliding window mechanism, we derive a characteristic feature vector by computing a weighted average across the window's temporal span. When a new feature vector exhibits significant deviation from this established characteristic pattern, we designate the corresponding temporal instance as an

anomaly. We illustrate this systematic process in Figure 1 and highlight a case where our higher-order approach successfully identifies an anomaly that is overlooked by traditional graph-based methods.

Contribution. To the best of our knowledge, this paper presents the first methodology for anomaly detection in temporal sequences of simplicial complexes. Our experimental results demonstrate that leveraging simplicial complex representations yields superior performance compared to traditional graph-based approaches in two key scenarios: First, when analyzing datasets containing inherent higher-order interactions, and second, when applying graph lifting techniques to enrich the topological structure.

Outline. The remainder of this paper is organized as follows: In Section 2, we review the existing literature on graph anomaly detection methods. In Section 3, we introduce the mathematical framework and notation used throughout this work. Section 4 formally defines the anomaly detection problem on simplicial complexes that we focus on. In Section 5, we present our spectral approach for detecting anomalies using Hodge Laplacians. Section 6 evaluates our method on synthetic and real-world datasets, demonstrating its effectiveness for event and change point detection. Finally, Section 7 summarizes our findings and discusses future research directions.

2 Related Work

Following the taxonomy established by Ranshous et al. [29], existing graph anomaly detection algorithms can be categorized into five principal categories: (1) community-based, (2) compression-based, (3) decomposition-based, (4) similarity/distance-based, and (5) probabilistic model-based approaches. Many methods can be subsumed under the following common framework: Initially, the algorithm extracts a low-dimensional feature representation from each temporal snapshot. Subsequently, these feature vectors are analyzed to establish baseline dissimilarity metrics that characterize normal temporal fluctuations of the data in the absence of an anomaly. When a snapshot exhibits dissimilarity exceeding an established threshold, it is identified as an anomaly. Events and change points can be detected by whether the shift in dissimilarity is localized or persists over the following snapshots.

Further, anomaly detection algorithms can be characterized by several essential properties, with existing methods exhibiting varying combinations of these attributes [21, 33]: First, the approach should maintain generality without imposing assumptions about the underlying data distribution, such as conformity to a specific graph model. Second, algorithms should be applicable to both event and change point detection scenarios. Third, computational scalability is crucial, particularly for large-scale networks with dynamic node populations. Within this context, we can also consider offline and online-style algorithms, where the former analyzes the entire time-series at once, while the latter flags anomalies in real-time as new data arrives. Fourth, node permutation invariance is essential to accommodate datasets where consistent node ordering across temporal snapshots cannot be guaranteed. Fifth, the method should provide quantitative confidence measures for anomaly predictions, enhancing interpretability and decision-making capabilities.

Model-based detection algorithms are a prominent approach in literature, wherein probabilistic graph models are fitted to observed

data and anomalies are identified through parameter changes. These approaches offer significant advantages in terms of interpretability, as detected parameter shifts directly correspond to identifiable large-scale structural changes in the network. Peel and Clauset [27] developed an online learning algorithm for change point detection in dynamic networks, learning probabilistic distributions over graphs and detecting when these distributions shift due to external events. Their method utilizes a generalized hierarchical random graph model for temporal network representation, though the framework remains model-agnostic. Notably, they identified a list of different types of change points, including community formation and merge events. In a similar direction, Wang et al. [33] proposed EdgeMonitoring, which derives snapshot features through joint edge probability estimation. Their method quantifies temporal dissimilarity using Kolmogorov-Smirnov statistics and Kullback-Leibler divergence measures. Gahrooei and Paynabar [12] developed a framework that fits generalized linear models to sequential network snapshots, employing extended Kalman filtering for continuous parameter estimation. They identify structural changes through significant deviations in the estimated model parameters.

Idé and Kashima [22] (Activity Vector), Akoglu and Faloutsos [1] and Huang et al. [21] (LAD) proposed different spectral-based methods for anomaly detection in dynamic graphs, which all rely on identifying changes in the graph spectrum as proxy for structural changes. These methods have been shown to be effective in detecting both event and change points in dynamic networks. In particular, LAD was later extended to attributed [20] and multi-view graphs [19].

Complementing these approaches, Koutra et al. [23] developed TENSORSPLAT, a decomposition-based method that leverages tensor factorization to identify structural changes in dynamic networks. Koutra et al. [24] introduced DELTACON, which utilizes direct similarity-based comparisons between consecutive graph snapshots rather than utilizing intermediate feature representations. While this method offers an alternative perspective on structural change detection, it imposes significant constraints: the framework requires both a static node set and consistent node identification across temporal snapshots.

With the increased interest in deep learning approaches in recent years, several methods have emerged that leverage neural network architectures for anomaly detection in dynamic graphs. Notably, Gong et al. [13] conceptualized change point detection as a temporal prediction framework, employing a latent evolution model to forecast subsequent graph snapshots. Their approach identifies anomalies when significant discrepancies arise between predicted and observed network states. Sulem et al. [30] proposed a method based on siamese graph neural networks to learn a similarity function between graph snapshots and detect change points based on deviations from this learned similarity. To the best of our knowledge, however, again no deep learning approach explores higher-order data in a simplicial complex setting.

For an in-depth review of existing anomaly detection methods on dynamic graphs, readers are directed to the recent surveys by Zhou et al. [35] and by Ho et al. [18]. Additional foundational perspectives can be found in the older surveys by Akoglu et al. [2] and Ranshous et al. [29].

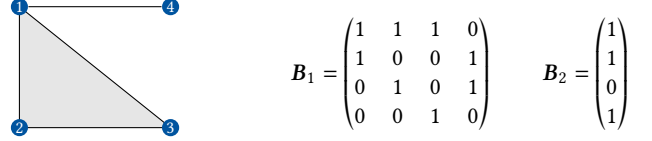


Figure 2: Illustration of a simplicial complex with associated boundary matrices. The simplicial complex contains a 2-simplex as shaded area. The rows and columns of the boundary matrices are ordered lexicographically.

3 Preliminaries

Notation. We denote vectors with boldface lowercase letters \boldsymbol{v} and matrices with boldface capital letters \boldsymbol{F} throughout this work. The temporal index of a variable at time step t is indicated by the superscript notation $(\cdot)^{(t)}$.

Simplicial Complexes. Given a set of points \mathcal{V} , a k -simplex \mathcal{S}^k is a subset of $k + 1$ points. An (abstract) simplicial complex (SC) \mathcal{X} comprises a collection of simplices that satisfies the following simplicial closure property: for any k -simplex $\mathcal{S}^k \in \mathcal{X}$, all of its proper subsets $\mathcal{S}^{k'} \subsetneq \mathcal{S}^k$ with $0 < k' < k$ are also elements of \mathcal{X} . We denote the set of k -simplices in \mathcal{X} by \mathcal{X}_k . The rank k_{\max} of a simplicial complex is defined as the maximum dimension among all its simplices, i.e., $k_{\max} = \max\{|\mathcal{S}^k| \mid \mathcal{S}^k \in \mathcal{X}\}$. Following standard terminology for graphs, we refer to 0-simplices as vertices, 1-simplices as edges, and 2-simplices as triangles.

The algebraic structure of a simplicial complex can be encoded through boundary matrices \boldsymbol{B}_k , which formalize the incidence relationships between simplices of adjacent dimensions [4, 17]. Specifically, the rows of \boldsymbol{B}_k correspond to $(k - 1)$ -simplices and columns correspond to k -simplices. The entry (i, j) takes value 1 if the i -th $(k - 1)$ -simplex is incident to the j -th k -simplex, and 0 otherwise. In particular, \boldsymbol{B}_1 represents the (unsigned) vertex-to-edge incidence matrix, \boldsymbol{B}_2 captures the edge-to-triangle incidence relations, and so on. For notational convenience, we define $\boldsymbol{B}_0 = \boldsymbol{B}_{k_{\max}+1} = \mathbf{0}$. We illustrate these constructs in Figure 2.

Utilizing these boundary matrices, we define the k -th combinatorial Hodge Laplacian \boldsymbol{L}_k of the simplicial complex \mathcal{X} as follows:

$$\boldsymbol{L}_k = \boldsymbol{B}_k^\top \boldsymbol{B}_k + \boldsymbol{B}_{k+1} \boldsymbol{B}_{k+1}^\top. \quad (1)$$

Notably, when $k = 0$, we recover the classical graph Laplacian $\boldsymbol{L}_0 = \boldsymbol{B}_{k+1} \boldsymbol{B}_{k+1}^\top$. To facilitate our analysis, we decompose the Hodge Laplacian into its constituent components: the down-Laplacian $\boldsymbol{L}_k^{\text{down}} = \boldsymbol{B}_k^\top \boldsymbol{B}_k$ and the up-Laplacian $\boldsymbol{L}_k^{\text{up}} = \boldsymbol{B}_{k+1} \boldsymbol{B}_{k+1}^\top$, which capture distinct aspects of the complex's structure.

In practical applications, it is often advantageous to assign non-negative weights to simplices to quantify their relative importance in the complex. Let $w_k : \mathcal{X}_k \rightarrow \mathbb{R}_{\geq 0}$ denote the weight function for k -simplices and \boldsymbol{W}_k the corresponding diagonal weight matrix. The boundary operators can be generalized to incorporate these weights by defining the weighted boundary matrix $\tilde{\boldsymbol{B}}_k = \boldsymbol{W}_{k-1}^{-1} \boldsymbol{B}_k \boldsymbol{W}_k$ [14]. Consequently, the weighted k -th Hodge Laplacian takes the form:

$$\tilde{\boldsymbol{L}}_k = \boldsymbol{W}_k \boldsymbol{B}_k^\top \boldsymbol{W}_{k-1}^{-2} \boldsymbol{B}_k \boldsymbol{W}_k + \boldsymbol{W}_k^{-1} \boldsymbol{B}_{k+1} \boldsymbol{W}_{k+1}^2 \boldsymbol{B}_{k+1} \boldsymbol{W}_k^{-1}. \quad (2)$$

Synthetic Model and Continuity Rate. Consider a temporal sequence of graphs and an underlying generative model that governs

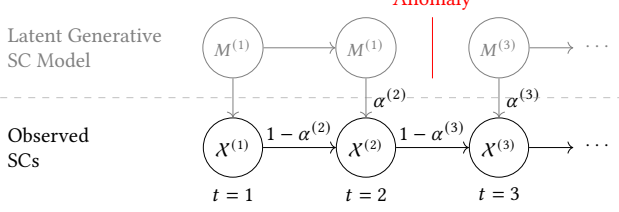
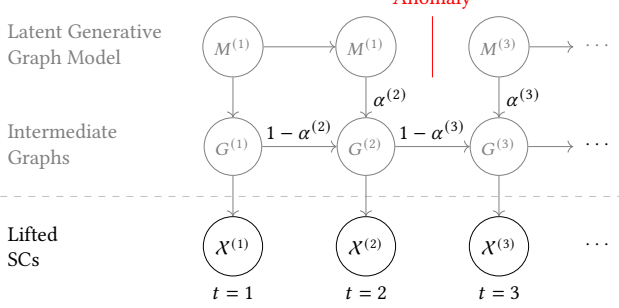
A: Simplicial Data**B: Graph Lifting**

Figure 3: Illustration of the continuity rate $\alpha^{(t)}$ governing the temporal evolution of a simplicial complex in a generative model setting. The latent variables $M^{(t)}$ denote the underlying generative model, while $X^{(t)}$ represents the observed simplicial complex snapshots. In (A), we show the case of underlying higher-order data, while (B) illustrates the scenario of graph lifting. The continuity rate $\alpha^{(t)}$ determines the proportion of simplices or edges that persist versus those resampled between consecutive temporal instances. A change in the model parameters at time step 3 illustrates an anomalous event that the detection algorithm aims to identify.

the evolution of the graph structure over time. The temporal evolution between consecutive graph snapshots is regulated by the continuity rate $\alpha^{(t)}$ [33], which determines the balance between preserved and resampled dyadic relationships. Specifically, for each dyad e , the probability $1 - \alpha^{(t)}$ governs whether its connection status persists from time step $t - 1$, while with probability $\alpha^{(t)}$ its status is resampled according to the current generation model. Thus, the structural similarity between adjacent temporal snapshots exhibits an inverse relationship with $\alpha^{(t)}$, where lower values yield higher topological preservation.

The extension of continuity rates to simplicial complexes presents significant challenges due to the inherent dependencies between simplices of different ranks: The removal of an edge that forms part of a triangle’s boundary necessitates the removal of the triangle, regardless of the continuity rate acting upon it. Conversely, when a resampled edge completes a triangle’s boundary that was previously incomplete, it creates the potential for a new triangle. We address these dependencies through two distinct approaches, determined by the origin of the higher-order interactions:

For graph lifting scenarios, we apply the continuity rate exclusively to the graph skeleton. Higher-order simplices are derived through clique lifting, independent of the continuity rate.

For data-informed higher-order simplices, we restrict the application of the continuity rate to simplex candidates that maintain validity across consecutive snapshots. Invalid simplices are systematically removed, while new simplex candidates are always sampled from the generation model. This approach facilitates the emergence of new higher-order simplices in subsequent snapshots, a practical consideration for evolving complex structures.

Figure 3 illustrates the continuity rate $\alpha^{(t)}$ governing the temporal evolution of a sequence of simplicial complexes in a generative model setting.

4 Problem Statement

Let \mathcal{X} denote a dynamic simplicial complex represented as a sequence of snapshots $\{\mathcal{X}^{(t)}\}_{t=1}^T$. Each simplex exists at specific timestamps t and may dynamically appear or disappear throughout the time series. Furthermore, we do not require identification of simplices across temporal instances, permitting arbitrary vertex and simplex orderings within each snapshot. Unlike some graph-based methods that require a fixed number of vertices [23, 33], our formulation does not impose such constraints. Specifically, we allow for a varying number of nodes across timestamps and do not require knowledge of the total number of vertices in the system.

Based on this formulation, our objective is to identify anomalous simplicial complexes $\mathcal{X}^{(t)}$ within the time series \mathcal{X} . Specifically, given an anomaly scoring function $f: \mathcal{X} \rightarrow \mathbb{R}$, we aim to detect time steps t exhibiting anomalous behavior. Following the taxonomy proposed by Ranshous et al. [29], we differentiate between two distinct types of anomalies:

- *Event points* represent temporal instances where a snapshot exhibits significant deviation from the typical behavior of the system. These deviations may arise from various sources including sensor errors, data corruption, or external perturbations. More formally, we characterize an event point t by the condition $|f(\mathcal{X}^{(t)}) - f(\mathcal{X}(W))| > \epsilon$, where $\mathcal{X}(W)$ denotes the short-term behavior within a recent context window W and ϵ represents a significance threshold.
- *Change points* indicate temporal locations where fundamental shifts occur in the underlying data distribution. Such transitions may result from external events altering the environment or modifications in the intrinsic data generation process. Formally, a change point t satisfies $|f(\mathcal{X}^{(t)}) - f(\mathcal{X}_{\text{norm}})| > \delta$, where $\mathcal{X}_{\text{norm}}$ captures the normal behavior in the global context and δ denotes a threshold parameter.

The anomaly scoring function should provide clear discrimination between anomalous and normal points while maintaining monotonicity with respect to the degree of anomalousness. We note that in the literature, event points are sometimes referred to as anomaly points, and a unified terminology for both types of anomalies is not always consistently employed.

5 Method

In this section, we present our method HLSAD for detecting event and change points in temporal sequences of simplicial complexes. This essentially implements the anomaly scoring function f introduced in the previous section, which assigns an anomaly score to each snapshot by analyzing its deviation from both short-term and

long-term temporal contexts, defined by window sizes w_s and w_ℓ respectively.

On a high level, HLSAD consists of the following steps: First, given an input time series $\{X^{(t)}\}_{t=1}^T$ of simplicial complexes, we compute the combinatorial Hodge Laplacians L_k up to a user-specified maximum rank K for each temporal snapshot. These Laplacians capture the topological information at different dimensional levels of the simplicial complex. Second, we extract spectral features by computing the ℓ largest singular values from each Hodge Laplacian, concatenating them into a feature vector. This step effectively compresses the high-dimensional structural information into a lower-dimensional representation while preserving the most significant topological characteristics. Finally, we employ a sliding window approach to detect anomalies by comparing the current feature vector against both short-term and long-term temporal contexts. Significant deviations from these temporal contexts are flagged as potential anomalies.

5.1 Spectrum

HLSAD leverages spectral features obtained through singular value decomposition (SVD) of the Hodge Laplacians for each temporal snapshot, extending established graph-based techniques to the simplicial domain. The choice of singular values as discriminative features is motivated by several theoretical considerations:

First, the Hodge Laplacian spectrum captures fundamental structural and topological properties of the simplicial complex: for instance, the dimension of the kernel space of the k -th Hodge Laplacian equals the k -th Betti number of the complex [9]. This provides information about its topological features such as connected components, cycles, and holes.

Second, by considering both up and down Laplacians at each dimension, we capture distinct aspects of the simplicial structure. The up-Laplacian L_k^{up} encodes how k -simplices combine to form $(k+1)$ -simplices, while the down-Laplacian L_k^{down} describes how k -simplices decompose into $(k-1)$ -simplices. This bidirectional analysis provides a comprehensive view of the complex's structure that would be impossible to achieve through graph-based methods.

Third, truncated SVD provides optimal low-rank approximations of matrices with respect to both the Frobenius and 2-norm [8]. The $(k+1)$ -th singular value σ_{k+1} quantifies the reconstruction error of the best rank- k approximation in the 2-norm. Thus, the ordered singular spectrum $\sigma_1, \sigma_2, \dots, \sigma_r$ encodes the information loss that would occur at different levels of approximation. Significant changes in these values indicate structural modifications that affect the complex's topology.

Fourth, our method maintains permutation invariance with respect to the ordering of simplices. This property is crucial for practical applications where consistent ordering cannot be guaranteed across temporal snapshots. Row or column permutations of the Hodge Laplacian matrices do not affect their singular values, enabling our method to handle arbitrary simplex orderings.

Finally, the computational efficiency of sparse SVD algorithms makes our approach practical for large-scale datasets. Real-world simplicial complexes often exhibit sparsity, particularly at higher dimensions where the number of simplices typically decreases. By computing only the top k singular values through truncated SVD,

we can achieve significant computational savings compared to full decomposition while maintaining high detection accuracy.

The selection of spectral features from Hodge Laplacians presents multiple strategic approaches, each capturing distinct topological characteristics of the simplicial complex while offering different computational trade-offs. These methodological choices significantly influence both the structural information preserved and the algorithmic efficiency of the detection process and can be tuned on a per-dataset basis. We discuss this decision matrix further in the next section. For simplicity, here we use the first K Hodge Laplacians and extract the first ℓ singular values from each, resulting in a feature vector of length $K \cdot \ell$ for each temporal snapshot. If a simplicial complex has a rank lower than K , or if the number of singular values is less than ℓ , we pad the feature vector with zeros.

5.2 Selection of Singular Values

The selection of singular values exhibits multiple dimensions of choice: First, we can compute singular values for different ranks k of Hodge Laplacians, enabling analysis of distinct structural properties within the simplicial complex. In our implementation, we consider the first K Hodge Laplacians, which is motivated by two key observations: For one, dyadic interactions serve as fundamental indicators for anomaly detection, and second, these lower-order interactions inherently influence higher-order connections through the simplicial closure property. From a practical perspective, simplicial complexes typically exhibit increasing sparsity at higher dimensions, allowing for the selection of a relatively small value of K without significant loss of structural information.

Second, the selection of ℓ singular values per Hodge Laplacian determines the dimensionality of the spectral representation. This parameter establishes a fundamental trade-off between computational efficiency and structural information retention: A larger value of ℓ enables us to capture more nuanced topological features at the cost of increased computational complexity of the anomaly detection algorithm. Retaining only a relatively small number of singular values usually sufficiently characterizes the dominant structural properties while maintaining computational tractability. While one could theoretically optimize this parameter independently for each order, our code and experiments maintain a uniform selection across ranks to reduce the parameter space complexity.

Third, in contrast to the graph Laplacian, the Hodge Laplacian is composed of two distinct components: the down-Laplacian L_k^{down} and the up-Laplacian L_k^{up} (with the graph Laplacian consisting solely of the latter component). This decomposition introduces an additional dimension for spectral feature optimization, enabling selective focus on the spectral properties of either component individually or a combination of both. This way, depending on the dataset, one can fine-tune the algorithm to emphasize different structural aspects of the simplicial complex.

5.3 Context Matrix and Typical Behavior

Following established practices in anomaly detection literature, we evaluate anomalies relative to a temporal context window of size w containing previous observations. To that end, we maintain a context matrix $C^{(\ell)}$ comprising the ℓ_2 -normalized spectra from the

last w snapshots:

$$C^{(t)} = \begin{pmatrix} \sigma^{(t-w-1)} & \sigma^{(t-w)} & \dots & \sigma^{(t-1)} \\ | & | & & | \\ | & | & & | \end{pmatrix} \in \mathbb{R}^{n \times w}. \quad (3)$$

This context matrix enables derivation of a “typical” spectrum $\tilde{\sigma}^{(t)}$ over the preceding w snapshots, which serves as a baseline for comparison with the current snapshot $\sigma^{(t)}$. Akoglu and Faloutsos [1] investigated two methodologies for obtaining such a typical feature vector: First, they used

$$\tilde{\sigma}^{(t)} = \frac{1}{w} \sum_{i=1}^w C_{:,i} = \frac{1}{w} \sum_{i=1}^w \sigma^{(t-i-1)} \quad (4)$$

as the arithmetic average of the last w feature vectors. Second, they computed the left singular vector of the context matrix $C^{(t)}$ using SVD decomposition, which can be interpreted as a weighted average of the preceding feature vectors. Both their empirical analysis and our experiments demonstrated superior performance of the SVD-based approach, leading us to adopt the SVD-based averaging.

5.4 Scoring Function

Let $\tilde{\sigma}_s^{(t)}$ and $\tilde{\sigma}_\ell^{(t)}$ denote the characteristic spectral features derived according to the previous section within the short-term and long-term context windows w_s and w_ℓ , respectively. Our goal now is to quantify the dissimilarity between the current spectral features $\sigma^{(t)}$ and these characteristic behaviors.

To quantify the structural deviation, we employ the angular distance between the current feature vectors and the established typical spectra:

$$Z_s^{(t)} = 1 - \left(\sigma_s^{(t)} \right)^\top \tilde{\sigma}_s^{(t)} \quad (5)$$

$$Z_\ell^{(t)} = 1 - \left(\sigma_\ell^{(t)} \right)^\top \tilde{\sigma}_\ell^{(t)}. \quad (6)$$

This normalized dissimilarity measure yields values in the interval $[0, 1]$, where $Z = 0$ corresponds to perfect structural alignment between current and characteristic features, while $Z = 1$ indicates maximal topological deviation manifested through orthogonality of the feature vectors. We define the final anomaly score

$$Z^{(t)} = \max\{Z_s^{(t)}, Z_\ell^{(t)}\} \quad (7)$$

as the maximum deviation across both short-term and long-term contexts.

Having established the anomaly scoring function, we can employ multiple strategies to identify events and change points: The first approach establishes a fixed threshold τ and classifies temporal instances t where $Z^{(t)} > \tau$ as anomalies. This threshold can be calibrated using an initial training period, during which we assume normal system behavior, by analyzing the distribution of $Z^{(t)}$ scores for a fixed “initialization window” at the beginning of the time series. Alternatively, we can adopt a rank-based approach that identifies the k snapshots with the highest anomaly scores. This latter method proves particularly effective when focusing on the most significant structural deviations, as it automatically adapts to the scale of the anomaly scores without requiring explicit threshold calibration.

Having defined our anomaly scoring function $Z^{(t)}$, we can differentiate between event and change points by analyzing the persistence of high scores across consecutive time steps: Event points typically manifest as isolated spikes in the anomaly score, where $Z^{(t)}$ exhibits a sharp increase followed by an immediate return to baseline levels. In contrast, change points are characterized by sustained elevations in $Z^{(t)}$ across multiple consecutive time steps, reflecting a persistent shift in the underlying network structure. Formally, we classify a time step t as a change point if both $Z^{(t)} > \tau$ and the moving average of $Z^{(t)}$ over the next w time steps exceeds $\tau/2$, where τ is our anomaly threshold and w is a user-defined window size. Time steps that exceed the threshold τ but do not meet the persistence criterion are classified as event points.

5.5 Performance Considerations

The computational complexity of HLSAD is dominated by the truncated singular value decomposition. For a matrix $A \in \mathbb{R}^{n \times n}$, computing a rank- ℓ decomposition requires $\mathcal{O}(n^2 \ell)$ operations. This complexity can be significantly reduced to $\mathcal{O}(n^2 \log \ell)$ using a randomized SVD algorithm [16], which provides efficient approximations while maintaining high accuracy. In practical applications, we can leverage the inherent sparsity structure of the Hodge Laplacians to achieve substantial computational efficiency gains.

Let n_k denote the maximum number of k -simplices across all temporal instances of the simplicial complex sequence $\{\mathcal{X}^{(t)}\}_{t=1}^T$. The computational complexity of HLSAD is then given by

$$\mathcal{O} \left(T \cdot \sum_{k=0}^K n_k^2 \log \ell \right), \quad (8)$$

where the summation over k accounts for the spectral computations required for each Hodge Laplacian up to order K .

6 Experiments

In this section, we present a comprehensive evaluation of HLSAD utilizing both synthetic and real-world datasets. Through these experiments, we demonstrate the effectiveness of our approach in detecting both event and change points both for data-informed higher-order simplices and graph liftings. Statistics for all datasets can be found in Table 5 in the appendix.

To quantitatively assess the performance of our anomaly detection method, we employ the Hits@ N metric, which measures the proportion of correctly identified anomalies among the top N most anomalous points detected by our algorithm. Specifically, a Hits@7 score of 0.5 indicates that 50% of the seven points identified as most anomalous correspond to actual anomalies according to the ground truth labels. If N equals the number of ground truth anomalies, then the Hits@ N score is equal to the recall. For synthetic experiments, we utilize the ground truth labels inherent in the generation process for evaluation purposes. In real-world datasets, we leverage known significant external events as ground truth, while acknowledging the limitation that additional unidentified external factors may have influenced the temporal evolution of the data.

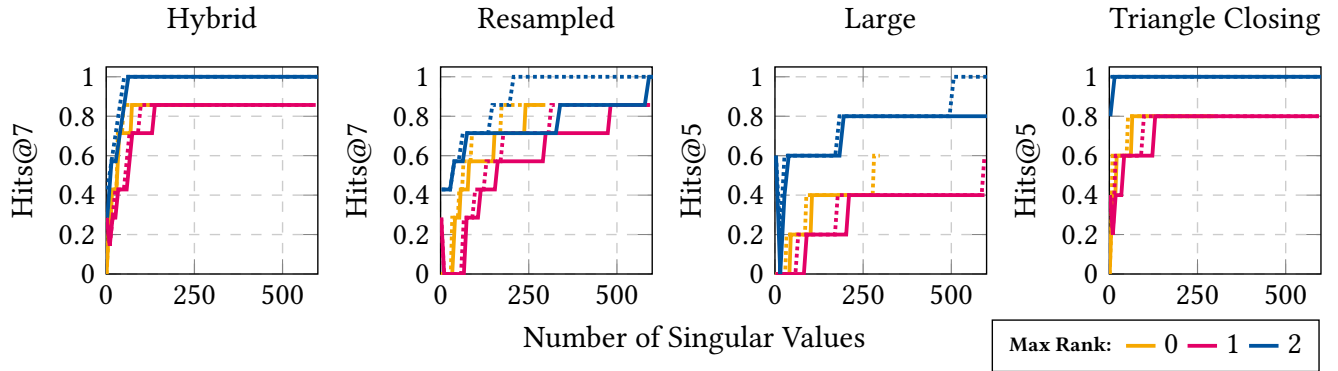


Figure 4: Performance comparison on synthetic data. We compare the Hits@ N scores as a function of the total number of singular values used for anomaly detection. The solid line gives the Hits@ N score for N the number of true anomalies in the data generation process. The dashed line gives the Hits@ $N + 2$ score, i.e., the accuracy with up to 2 false positives.

6.1 Synthetic Experiments

To evaluate HLSAD’s effectiveness and demonstrate its advantages over graph-based approaches, we conduct four synthetic experiments: In the first two, we replicate the *hybrid* and *resampled* synthetic experiments by Huang et al. [21]. The time series originates from graphs sampled from a stochastic block model (SBM), which are lifted to simplicial complexes using a clique lifting. In a third *large* experiment, we evaluate scalability by using the same setting with 10 000 nodes. In a fourth experiment, we introduce a *triangle closing* dataset where, instead of using clique lifting, triangles are closed with a given probability. This setting reflects scenarios where external events have a distinct influence on multi-way interactions in the network, which may differ from their impact on dyadic relationships. In all settings, we change the model parameters at pre-selected time steps to simulate events and change points in the data. The exact sampling parameters for all experiments are given in Section B. We use a short term and a long term window of $w_s = 5$ and $w_l = 10$ snapshots, respectively.

The Hits@ N scores for each dataset as a function on the total number of singular values computed are shown in Figure 4. For the higher-order cases, the plotted number of singular values refers to the total number of singular values computed over all Hodge Laplacians, i.e., for a value of 300 on the x-axis and maximum rank $K = 3$, we computed 100 singular values for each Hodge Laplacian. In other words, all points on the same x-axis correspond to the same total number of singular values, though in the higher-order cases, these are distributed over multiple Hodge Laplacians.

The experimental results demonstrate the superior performance of HLSAD in leveraging higher-order structural information compared to the graph-based LAD baseline across both the *hybrid*, *large*, and *triangle closing* experiments. HLSAD consistently achieves higher Hits@ N scores while requiring significantly fewer singular values to attain comparable detection accuracy, indicating enhanced computational efficiency. Specifically, in the *hybrid* setting, HLSAD successfully identifies all anomalies using only 40 singular values, whereas LAD requires more than 300 singular values to achieve similar performance. The advantage of HLSAD is particularly evident in the *triangle closing* experiment, where it detects all anomalies

Table 1: Prediction scores on the UCI and Senate real-world datasets. We compare HLSAD against several other state-of-the-art anomaly detection techniques.

Dataset	UCI	Senate
Metric	Hits@10	Hits@2
HLSAD (ours)	1.0	1.0
LAD	0.5	1.0
EdgeMonitoring	0.0	1.0
Activity Vector	0.5	0.5
TENSORPLAT	0.0	0.0

with merely 10 singular values, while LAD fails to identify anomalies that manifest exclusively in higher-order interactions due to its inherent limitation to dyadic relationships.

The interpretation of the *resampled* experiment yields more nuanced results: While the general trends observed in other experiments persist, namely superior overall performance and enhanced efficiency with fewer singular values, there exist specific instances where LAD marginally outperforms HLSAD. Specifically, LAD achieves a Hits@7 score of 0.8 with 250 singular values, whereas HLSAD requires approximately 320 singular values to attain equivalent performance. However, examination of the Hits@9 metric reveals that HLSAD consistently and significantly outperforms LAD across all singular value quantities in a minor relaxed metric. This suggests that while HLSAD may occasionally identify false positives, it demonstrates superior capability in detecting the complete set of true anomalies within the data.

6.2 UCI Online Message Dataset

We further replicate the experiment on the UCI Online Message dataset [26], which captures communication patterns within an online student community at the University of California, Irvine. The dataset comprises a weighted network where nodes represent users and edges represent message exchanges between them. Edge weights correspond to message lengths, quantifying interaction

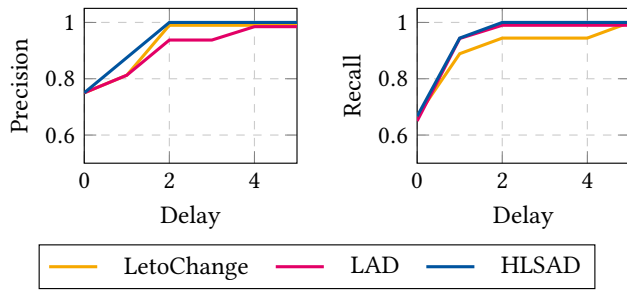


Figure 5: Precision and recall as a function of the detection delay on the MIT Reality Mining dataset. We compare our method HLSAD against the spectral-based LAD approach and the model-based LetoChange framework.

intensity. The dataset encompasses 1899 users and records 59 835 messages exchanged between April and October 2004. On June 19, the spring term ended and on September 20, the fall term began, marking significant external events that influenced the communication patterns. We construct daily network snapshots for our temporal analysis and use a clique lifting to transform the dyadic interactions into higher-order simplices. Following the arguments outlined in the previous works, we employ a short-term context window of 7 days, with a corresponding long-term window of 14 days to capture broader temporal patterns.

Our analysis, summarized in Table 1, demonstrate the superior performance of HLSAD in detecting temporal anomalies. The method achieves a perfect Hits@10 score, accurately identifying all ground truth anomalies in the dataset. LAD correctly detected the end of the spring term, but missed the beginning of the fall term by a single timestep. Similarly, the Activity Vector method only successfully identified the end of the spring term. Both EdgeMonitoring and TENSORPLAT prove ineffective, failing to identify any anomalies in the dataset. Notably, all methods that demonstrate meaningful detection capabilities—HLSAD, LAD, and Activity Vector—utilize spectral-based approaches, underlining the effectiveness of spectral features in temporal anomaly detection tasks.

6.3 Senate Network

We analyze the Senate co-sponsorship network [10, 11], which represents legislative collaboration patterns through higher-order interactions: when multiple senators co-sponsor a bill, they form a k -simplex in the complex. The dataset comprises biannual snapshots spanning multiple congressional sessions.

Prior research has identified two notable anomalies in collaboration patterns: the 100th Congress exhibited exceptionally high levels of bipartisan cooperation, while the 104th Congress marked a historical low point in cross-party collaboration [33]. Our experimental results demonstrate that HLSAD successfully identifies both these significant structural anomalies using only the top 6 singular values of the Hodge Laplacians, as shown in Table 1.

6.4 MIT Reality Mining

For our third real-world experiment, we analyze the MIT Reality Mining dataset [7], which documents proximal interactions between students through continuous Bluetooth device monitoring. We construct weekly temporal networks from the raw proximity data, where edges indicate detected device co-location between participants. Higher-order simplices are derived through clique lifting. To ensure data quality, we exclude participants with incomplete or corrupted records, particularly when data was corrupted due to power failures. For ground truth validation, we leverage the 16 external events identified by Peel and Clauset [27] that potentially influenced interaction patterns throughout the study period.

In analogy to previous work [27], we compute the precision and recall of our method as a function of the detection delay between the ground truth events $\{t_j\}$ and the detected anomaly $\{\hat{t}_i\}$. More formally, the precision and recall are defined as

$$\text{Precision}(s) = \frac{1}{n_{\text{pred}}} \sum_i \delta \left(\inf_j (\hat{t}_i - t_j) \leq s \right) \quad (9)$$

$$\text{Recall}(s) = \frac{1}{n_{\text{true}}} \sum_j \delta \left(\inf_i (\hat{t}_i - t_j) \leq s \right), \quad (10)$$

where s is the allowed detection delay, n_{pred} is the number of predicted anomalies, n_{true} is the number of true anomalies, and $\delta(\cdot)$ is the indicator function.

In our experimental evaluation, we compare HLSAD against two established baselines: the spectral-based LAD approach [21] and the model-based LetoChange framework [27]. The comparative analysis, presented in Figure 5, demonstrates that HLSAD achieves comparable precision to LetoChange for instantaneous anomaly detection ($s = 0$), while exhibiting superior detection capabilities when allowing for some detection delays. Furthermore, HLSAD consistently outperforms LetoChange in terms of recall across all detection delays, indicating superior sensitivity in identifying true anomalies within the dataset. Although the spectral-based LAD method exhibits recall performance comparable to HLSAD, it generates additional false positives in temporal proximity to ground truth events, resulting in diminished precision metrics. Overall, our results highlight the effectiveness of HLSAD in detecting temporal anomalies in real-world social interaction networks.

7 Conclusion

In this paper, we introduced a novel methodology for detecting anomalies in time-evolving simplicial complexes. Our approach leverages the spectral properties of Hodge Laplacians to capture higher-order structural changes that traditional graph-based methods fail to detect. Through comprehensive experiments on both synthetic and real-world datasets, we demonstrated that our method successfully identifies both event and change points in temporal sequences of simplicial complexes.

The key advantages of HLSAD are threefold: First, it naturally handles higher-order interactions that are either inherent in the data or arise through graph lifting techniques. Second, it maintains computational efficiency through the use of truncated SVD computations on sparse Hodge Laplacians. Third, it is invariant to node permutations and growing simplicial complexes, making it

applicable when consistent node ordering cannot be guaranteed or a priori knowledge of the total number of nodes is unavailable.

Our experimental results show that our method achieves superior detection accuracy compared to traditional graph-based approaches, particularly when analyzing datasets with significant higher-order structure. Furthermore, when applied to lifted graph data, our method requires fewer eigenvalues to achieve comparable accuracy, resulting in improved computational efficiency.

Future work could explore adaptive window sizes for the temporal context, automated parameter selection techniques, and extensions to weighted or directed simplicial complexes. Additionally, investigating the relationship between specific types of structural changes and their manifestation in the Hodge spectrum could provide deeper insights into the nature of detected anomalies as well as possibilities to fine-tune to specific anomaly kinds.

In many real-world applications, the presence of attributed data or node features can provide additional context for anomaly detection beyond the structural information captured by the Hodge Laplacian. However, our current implementation focuses solely on the topological structure of the simplicial complex. In future work, we plan to extend our method to incorporate attributed data or node features, allowing for a more comprehensive analysis of anomalies in complex systems.

Acknowledgments

The authors acknowledge funding by the Ministry of Culture and Science of the German State of North Rhine-Westphalia (“NRW Rückkehrprogramm”) and the European Union (ERC HIGH-HOPeS, 101039827). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

References

- [1] Leman Akoglu and Christos Faloutsos. 2010. Event Detection in Time Series of Mobile Communication Graphs. In *Army Science Conference*, Vol. 1. 141.
- [2] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2015. Graph Based Anomaly Detection and Description: A Survey. *Data Mining and Knowledge Discovery* 29, 3 (May 2015), 626–688. doi:10.1007/s10618-014-0365-y
- [3] Cristian Bodnar, Fabrizio Frasca, Yuguang Wang, Nina Otter, Guido F Montufar, Pietro Lió, and Michael Bronstein. 2021. Weisfeiler and Lehman Go Topological: Message Passing Simplicial Networks. In *Proceedings of the 38th International Conference on Machine Learning*. 1026–1037.
- [4] Glen E. Bredon. 1993. *Topology and Geometry*. Springer-Verlag, New York.
- [5] Haibin Cheng, Pang-Ning Tan, Christopher Potter, and Steven Klooster. 2008. A Robust Graph-Based Algorithm for Detection and Characterization of Anomalies in Noisy Multivariate Time Series. In *2008 IEEE International Conference on Data Mining Workshops*. IEEE, Pisa, Italy, 349–358. doi:10.1109/ICDMW.2008.48
- [6] Qi Ding, Natallia Katenka, Paul Barford, Eric Kolaczyk, and Mark Crovella. 2012. Intrusion as (a)Anti)Social Communication: Characterization and Detection. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Beijing China, 886–894. doi:10.1145/2339530.2339670
- [7] Nathan Eagle and Alex (Sandy) Pentland. 2006. Reality Mining: Sensing Complex Social Systems. *Personal and Ubiquitous Computing* 10, 4 (May 2006), 255–268. doi:10.1007/s00779-005-0046-3
- [8] Carl Eckart and Gale Young. 1936. The Approximation of One Matrix by Another of Lower Rank. *Psychometrika* 1, 3 (Sept. 1936), 211–218. doi:10.1007/BF02288367
- [9] Beno Eckmann. 1944. Harmonische Funktionen und Randwertaufgaben in einem Komplex. *Commentarii Mathematici Helvetici* 17, 1 (Dec. 1944), 240–255. doi:10.1007/BF02566245
- [10] James H. Fowler. 2006. Connecting the Congress: A Study of Cosponsorship Networks. *Political Analysis* 14, 4 (2006), 456–487. doi:10.1093/pan/14.4
- [11] James H. Fowler. 2006. Legislative Cosponsorship Networks in the US House and Senate. *Social Networks* 28, 4 (Oct. 2006), 454–465. doi:10.1016/j.socnet.2005.11.003
- [12] Mostafa Reisi Gahrooei and Kamran Paynabar. 2018. Change Detection in a Dynamic Stream of Attributed Networks. *Journal of Quality Technology* 50, 4 (Oct. 2018), 418–430. doi:10.1080/00224065.2018.1507558
- [13] Yongshun Gong, Xue Dong, Jian Zhang, and Meng Chen. 2023. Latent Evolution Model for Change Point Detection in Time-Varying Networks. *Information Sciences* 646 (Oct. 2023), 119376. doi:10.1016/j.ins.2023.119376
- [14] Nicola Guglielmi, Anton Savostianov, and Francesco Tudisco. 2023. Quantifying the Structural Stability of Simplicial Homology. *Journal of Scientific Computing* 97, 1 (Oct. 2023), 2. doi:10.1007/s10915-023-02314-2
- [15] Mustafa Hajji, Mathilde Papillon, Florian Frantzen, Jens Agerberg, Ibrahim Al-Jabea, Rubén Ballester, Claudio Battiloro, Guillermo Bernárdez, Tolga Birdal, Aiden Brent, Peter Chin, Sergio Escalera, Simone Fiorellino, Odin Hoff Gardaa, Gurusankar Gopalakrishnan, Devendra Govil, Josef Hoppe, Maneel Reddy Karri, Jude Khouja, Manuel Lecha, Neal Livesay, Jan Meißner, Soham Mukherjee, Alexander Nikitin, Theodore Papamarkou, Jaro Prilepok, Karthikeyan Natesan Ramamurthy, Paul Rosen, Aldo Guzmán-Sáenz, Alessandro Salatiello, Shreyas N. Samaga, Simone Scardapane, Michael T. Schaub, Luca Scofano, Indro Spinelli, Lev Telyatnikov, Quang Truong, Robin Walters, Maosheng Yang, Olga Zaghen, Ghada Zamzmi, Ali Zia, and Nina Miolane. 2024. TopoX: A Suite of Python Packages for Machine Learning on Topological Domains. *Journal of Machine Learning Research* 25, 374 (2024), 1–8.
- [16] N. Halko, P. G. Martinsson, and J. A. Tropp. 2011. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Rev.* 53, 2 (Jan. 2011), 217–288. doi:10.1137/09071806
- [17] Allen Hatcher. 2002. *Algebraic Topology*. Cambridge University Press.
- [18] Thi Kieu Khanh Ho, Ali Karami, and Narges Armanfard. 2025. Graph Anomaly Detection in Time Series: A Survey. doi:10.48550/arXiv.2302.00058 arXiv:2302.00058 [cs]
- [19] Shenyang Huang, Samy Coulombe, Yasmeen Hitti, Reihaneh Rabbany, and Guillaume Rabusseau. 2024. Laplacian Change Point Detection for Single and Multi-view Dynamic Graphs. *ACM Transactions on Knowledge Discovery from Data* 18, 3 (April 2024), 1–32. doi:10.1145/3631609
- [20] Shenyang Huang, Jacob Danovitch, Guillaume Rabusseau, and Reihaneh Rabbany. 2023. Fast and Attributed Change Detection on Dynamic Graphs with Density of States. In *Advances in Knowledge Discovery and Data Mining*, Hisashi Kashima, Tsuyoshi Idé, and Wen-Chih Peng (Eds.). Vol. 13935. Springer Nature Switzerland, Cham, 15–26. doi:10.1007/978-3-031-33374-3_2
- [21] 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*. ACM, Virtual Event CA USA, 349–358. doi:10.1145/3394486.3403077
- [22] 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, Seattle WA USA, 440–449. doi:10.1145/1014052.1014102
- [23] Danai Koutra, Evangelos E. Papalexakis, and Christos Faloutsos. 2012. Tensor-Splat: Spotting Latent Anomalies in Time. In *2012 16th Panhellenic Conference on Informatics*. IEEE, Piraeus, Greece, 144–149. doi:10.1109/PCI.2012.60
- [24] 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* 10, 3 (Feb. 2016), 1–43. doi:10.1145/2824443
- [25] Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B. Hall, Christopher H. Tomkins-Tinch, Vanessa Sochat, Jan Forster, Soohyun Lee, Sven O. Twardziok, Alexander Kanitz, Andreas Wilm, Manuel Holtgrewe, Sven Rahmann, Sven Nahnsen, and Johannes Köster. 2021. Sustainable Data Analysis with Snakemake. *F1000Research* 10 (April 2021), 33. doi:10.12688/f1000research.29032.2
- [26] Pietro Panzarasa, Tore Opsahl, and Kathleen M. Carley. 2009. Patterns and Dynamics of Users’ Behavior and Interaction: Network Analysis of an Online Community. *Journal of the American Society for Information Science and Technology* 60, 5 (May 2009), 911–932. doi:10.1002/asi.21015
- [27] Leto Peel and Aaron Clauset. 2015. Detecting Change Points in the Large-Scale Structure of Evolving Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29. doi:10.1609/aaai.v29i1.9574
- [28] Caspar A. S. Pouw, Federico Toschi, Frank Van Schadewijk, and Alessandro Corbetta. 2020. Monitoring Physical Distancing for Crowd Management: Real-time Trajectory and Group Analysis. *PLOS ONE* 15, 10 (Oct. 2020), e0240963. doi:10.1371/journal.pone.0240963
- [29] Stephen Ranshous, Shitian Shen, Danai Koutra, Steve Harenberg, Christos Faloutsos, and Nagiza F. Samatova. 2015. Anomaly Detection in Dynamic Networks: A Survey. *WIREs Computational Statistics* 7, 3 (May 2015), 223–247. doi:10.1002/wics.1347
- [30] Déborah Sulem, Henry Kenlay, Mihai Cucuringu, and Xiaowen Dong. 2024. Graph Similarity Learning for Change-Point Detection in Dynamic Networks. *Machine Learning* 113, 1 (Jan. 2024), 1–44. doi:10.1007/s10994-023-06405-x

- [31] Jimeng Sun, Yinglian Xie, Hui Zhang, and Christos Faloutsos. 2008. Less Is More: Sparse Graph Mining with Compact Matrix Decomposition. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 1, 1 (Feb. 2008), 6–22. doi:10.1002/sam.102
- [32] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. Van Der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul Van Mulbregt, SciPy 1.0 Contributors, Aditya Vijaykumar, Alessandro Pietro Bardelli, Alex Rothberg, Andreas Hilboll, Andreas Kloeckner, Anthony Scopatz, Antony Lee, Ariel Rokem, C. Nathan Woods, Chad Fulton, Charles Masson, Christian Häggström, Clark Fitzgerald, David A. Nicholson, David R. Hagen, Dmitrii V. Pasechnik, Emanuele Olivetti, Eric Martin, Eric Wieser, Fabrice Silva, Felix Lenders, Florian Wilhelm, G. Young, Gavin A. Price, Gert-Ludwig Ingold, Gregory E. Allen, Gregory R. Lee, Hervé Audren, Irvin Probst, Jörg P. Dietrich, Jacob Silterra, James T Webber, Janko Slavić, Joel Nothman, Johannes Buchner, Johannes Kulick, Johannes L. Schönberger, José Vinicius De Miranda Cardoso, Joscha Reimer, Joseph Harrington, Juan Luis Cano Rodríguez, Juan Nunez-Iglesias, Justin Kuczynski, Kevin Tritz, Martin Thoma, Matthew Newville, Matthias Kümmerer, Maximilian Bolingbroke, Michael Tartre, Mikhail Pak, Nathaniel J. Smith, Nikolai Nowaczyk, Nikolay Shebanov, Oleksandr Pavlyk, Per A. Brodtkorb, Perry Lee, Robert T. McGibbon, Roman Feldbauer, Sam Lewis, Sam Tygier, Scott Sievert, Sebastiano Vigna, Stefan Peterson, Surhud More, Tadeusz Pudlik, Takuya Oshima, Thomas J. Pingel, Thomas P. Robitaille, Thomas Spura, Thouis R. Jones, Tim Cera, Tim Leslie, Tiziano Zito, Tom Krauss, Utkarsh Upadhyay, Yaroslav O. Halchenko, and Yoshiki Vázquez-Baeza. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17, 3 (March 2020), 261–272. doi:10.1038/s41592-019-0686-2
- [33] Yu Wang, Aniket Chakrabarti, David Sivakoff, and Srinivasan Parthasarathy. 2017. Fast Change Point Detection on Dynamic Social Networks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, Melbourne, Australia, 2992–2998. doi:10.24963/ijcai.2017/417
- [34] 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 (Aug. 2016), 1–14. doi:10.1145/2980765.2980767
- [35] Yuxuan Zhou, Shang Gao, Dandan Guo, Xiaohui Wei, Jon Rokne, and Hui Wang. 2024. A Survey of Change Point Detection in Dynamic Graphs. *IEEE Transactions on Knowledge and Data Engineering* 37, 3 (Dec. 2024), 1030–1048. doi:10.1109/TKDE.2024.3523857

A Teaser Figure

Table 2 lists the exact parameters for the time series illustrated in Figure 1. Graph skeletons are sampled with 30 nodes from a SBM using uniform community sizes. Triangles are closed (identified with a 2-simplex) with probability p_{Δ} . The continuity rate $\alpha^{(t)}$ is set to 1.0 for anomaly points and $\alpha^{(t)} = 0.005$ everywhere else.

Table 2: SBM parameters and triangle closing probability for the time series used in Figure 1. Parameter changes are marked in bold.

Step	Type	Communities		Intra- p_{in}	Inter- p_{ex}	Triangle p_{Δ}
		N_c				
1–9	-	3	0.25	0.05	0.09	
10	Event	3	0.15	0.05	0.09	
11–19	-	3	0.25	0.05	0.09	
20–29	Change	3	0.25	0.15	0.09	
30–39	Change	3	0.25	0.15	0.03	

B Experimental Setup

Table 3 lists the exact SBM parameters for the *hybrid* and *resampled* synthetic experiments, which are derived from the work of Huang

et al. [21]. We excluded their *pure* experiment, as anomalous points in this setting can be perfectly detected in the first singular values. Table 4 and 6 lists the parameters for the *large* and *triangle closing* synthetic experiment, respectively.

All graphs except for the large dataset are sampled with 500 nodes and uniform community sizes. For the large dataset, the sampled graphs have 10 000 nodes. For the hybrid and large setting, the continuity rate was set to $\alpha^{(t)} = 1.0$ for anomaly points and $\alpha^{(t)} = 0.1$ everywhere else. For the resampled setting, the continuity rate was set to $\alpha^{(t)} = 1.0$ everywhere, i.e., a fresh graph was resampled at each time step. For the first three experiments, we used a clique lifting to obtain the simplicial complexes from the graph skeletons sampled from the SBMs. For the *triangle closing* experiment, triangles are closed with a given probability p_{Δ} , simulating a data-informed higher-order structure where external events have a distinct influence on multiway interactions in the network.

Table 3: SBM parameters for the *hybrid* and *resampled* synthetic experiments. Parameter changes are marked in bold.

Step	Type	Communities	Intra-Prob.	Inter-Prob.
		N_c	p_{in}	p_{ex}
1–16	-	4	0.25	0.05
17	Event	4	0.25	0.15
18–31	-	4	0.25	0.05
32–61	Change	10	0.25	0.05
62	Event	10	0.25	0.15
63–76	-	10	0.25	0.05
77–91	Change	2	0.5	0.05
92	Event	2	0.5	0.15
93–106	-	2	0.5	0.05
107–136	Change	4	0.25	0.05
137	Event	4	0.25	0.15
138–150	-	4	0.25	0.05

Table 4: SBM parameters for the *large* synthetic experiment. Parameter changes are marked in bold.

Step	Type	Communities	Intra-Prob.	Inter-Prob.
		N_c	p_{in}	p_{ex}
1–16	-	4	0.0125	0.0025
17	Event	4	0.0125	0.0075
18–31	-	4	0.0125	0.0025
32–61	Change	10	0.25	0.0025
62	Event	10	0.25	0.0075
63–76	-	10	0.25	0.0025
77–91	Change	2	0.025	0.0025
92	Event	2	0.025	0.00755
93–100	-	2	0.025	0.0025

Table 5 lists the statistics for the synthetic and real-world datasets used in our experiments. We implemented HLSAD in Python, leveraging the TopoX library [15] for topological computations and

Table 5: Statistics for the datasets used in our experiments. We give the mean and standard deviation of the number of nodes, edges, and triangles in the dynamic simplicial complexes.

Dataset	T	avg $ \mathcal{X}_0^{(t)} $	avg $ \mathcal{X}_1^{(t)} $	avg $ \mathcal{X}_2^{(t)} $
Hybrid	150	500	19 266 ± 10 346	190 476 ± 284 845
Resampled	150	500	19 237 ± 10 254	188 018 ± 279 797
Triangle Closing	60	500	16 898 ± 6652	22 092 ± 31 509
Large	100	10 000	394 171 ± 215 818	211 516 ± 294 775
UCI	193	117 ± 121	134 ± 180	5 ± 11
Senate	12	101.0 ± 0.8	5033 ± 88	165 770 ± 4366
Reality Mining	32	94 ± 17	603 ± 302	2937 ± 2214

Table 6: SBM parameters and triangle closing probability for the triangle closing synthetic experiment. Changes in the parameters are marked in bold.

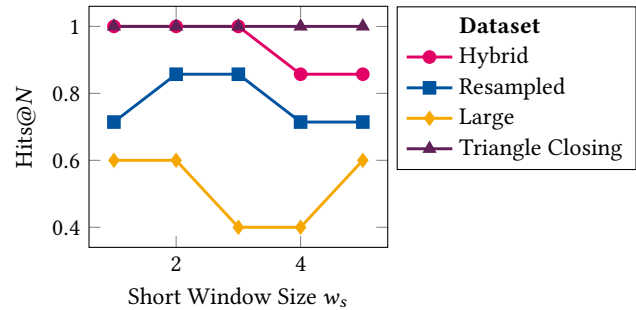
Step	Type	Communities	N_c	Intra	Inter	Triangle
				p_{in}	p_{ex}	p_Δ
1–9	-	4	0.25	0.05	0.8	
10	Event	4	0.25	0.15	0.8	
11–19	-	4	0.25	0.05	0.8	
20–29	Change	4	0.25	0.05	0.7	
30	Event	4	0.25	0.15	0.7	
31–39	-	4	0.25	0.05	0.7	
40–49	Change	4	0.5	0.05	0.5	
50–59	Change	4	0.5	0.05	0.5	

SciPy [32] for efficient algebraic operations. HLSAD has been implemented as a Snakemake pipeline [25]. We took inspiration from the implementation of LAD by Huang et al. [21]. For comparative analysis, we utilized reference implementations of baseline methods, except LAD, which we realized as a special case of HLSAD with appropriate parameter settings. All source code and experimental data will be made publicly available with the camera-ready version to ensure reproducibility.

C Ablation Study: Influence of the Context Window

In this section, we conduct an ablation study to investigate the influence of the context window sizes on the prediction performance of HLSAD. The context window sizes have only small influence on the computation time of HLSAD, as the computational complexity is dominated by the SVD computation, which is independent of the context window sizes, see Section 5.5.

To that end, Figure 6 shows the Hits@N scores for different context window sizes on the synthetic datasets used in Section 6.1. In this experiment, we set the number of singular values to 75 and the maximal simplex rank to 2. We can see that the Hits@N scores drop for large context windows. This is likely due to the fact that large contexts smooth out any changes in the network structure, i.e., $\sigma^{(t)}$ and $\sigma^{(t+1)}$ are more similar with larger context even if the network structure changes. On the other hand, short windows can also be problematic, see the resampled setting, as the inferred

**Figure 6: Ablation study on the influence of the context window on the performance of HLSAD. We show the Hits@N scores for different context window sizes on the synthetic datasets. The long window w_ℓ is set to double the short window w_s .**

typical spectra tend to be noisy and lead to more false positives. Interestingly, the *large* dataset shows a counterintuitive behavior, where the accuracy drops in the middle and increases at its tails. We don't have an intuitive explanation for this behavior.

The results show that the context window sizes can have significant influence on the performance of HLSAD, which is in line with other algorithms for temporal graph anomaly detection employing context windows. Moreover, the different behaviors even on the relatively similar synthetic datasets highlight the importance to fine-tune the context window size specifically for the dataset at hand.