AMSI SUMMER RESEARCH SCHOLARSHIPS 2025–26

Get a taste for Research this Summer

Graph Fourier Analysis for Community Detection in Covert Networks

James Murray Streitberg

Supervised by Prof. Asha Rao

RMIT University

Contents

Preface

0.1 Abstract

Covert transaction networks pose a fundamental challenge for community detection: labels are scarce or unreliable, and node attributes may be missing or unusable, so methods must rely primarily on graph topology. This report investigates whether the Laplacian geometry of such networks is sufficient to reveal meaningful substructures without seeded supervision. We compare two topology-only pipelines applied to the Elliptic Bitcoin transaction dataset. The first is classical spectral clustering. The second is a diffusion-driven method motivated by graph Fourier analysis: Laplacian-derived probe signals are smoothed by a heat-kernel filter and implemented at scale via Chebyshev polynomial approximation to avoid eigendecomposition; vertices are then embedded by filtered responses and clustered in the induced diffusion geometry. Illicit/licit labels are used strictly post hoc for evaluation. Performance is assessed through community quality metrics (modularity and conductance) and a covert-relevance criterion (illicit lift). Across snapshots, spectral clustering consistently yields lower conductance and often higher modularity than the heat-filter pipeline, while the diffusion method produces more variable cut quality but can recover operationally interpretable motif-like structures; in at least one representative timestep both methods identify the same star-shaped high-lift cluster, a motif commonly associated with suspicious transaction behaviour. These results support the premise that Laplacian-based structure contains exploitable information in covert settings, while highlighting that topology-only diffusion embeddings are sensitive to signal design and embedding dimension and may capture motifs not well scored by standard cut objectives. Future work is directed toward better matched diffusion feature maps and clustering objectives, and evaluation criteria that complement cut metrics with motif- and stability-oriented measures.

0.2 Acknowledgement

I would like to give thanks to my supervisor Professor Asha Rao for her support, guidance and wisdom throughout this project, and for the many hours she has spent meeting with me. I would also like to thank E. Sule Yazici for her insights and great questions, and for taking time out of her few weeks in Melbourne to meet with me. Finally, I thank all my loved ones for believing in me.

0.3 Statement of Authorship

All mathematical theorems, definitions and descriptions are derived from textbooks, papers and journals, and other scholarly sources, and are cited appropriately. Sections of the methodology and implementation that are influenced or motivated by existing literature are likewise referenced accordingly. The datasets used in this project were obtained from publicly available sources, and are cited appropriately. The experimental design, remaining computational pipeline, results obtained, and their interpretation and analysis are my own original work, conducted under the guidance of my supervisor.

Introduction

Illicit financial activities increasingly manifest as complex transaction networks. Detecting relevant substructures in such networks is a unique challenge since organised behaviour often appears as communities, motifs, or diffusion patterns rather than as isolated nodes. In practice, however, the most operationally relevant settings are commonly covert; only a small fraction of entities are labelled, and these labels may be noisy, delayed, or obscured (Dey and Medya (2019)). This lack of reliable observation then motivates the pursuit of community-detection methods that can operate at a network-topology level; leveraging structural regularities in the graph even when node features or annotations are unavailable or unusable.

Spectral methods provide a principled approach to this problem. In spectral clustering, community structure is sought through the eigenspaces of the graph Laplacian, whose low-frequency harmonics encode slowly-varying structure over the graph and are connected to cluster metrics such as conductance (Luxburg (2007)). More broadly, graph signal processing adapts the Laplacian as an operator that generates a Fourier basis on the vertex set, enabling the construction of filters that attenuate or amplify structural variation over graph frequencies (Shuman et al. (2013)). These perspectives suggest a recurrent hypothesis: if illicit activity is organised in structurally coherent groups, then Laplacian geometry alone may contain exploitable information about that organisation, even in the absence of node attributes.

This report investigates this hypothesis on the Elliptic Bitcoin transaction network (Elmougy and Liu 2023): a temporal network comprising 49 snapshots with over 2×10^5 nodes and 2×10^5 edges in total. Nodes represent accounts, and edges represent transactions. Labels (licit/illicit/unknown) are applied strictly post hoc rather than as signal inputs to clustering. We compare two topology-only community discovery pipelines. The first is spectral clustering following the classical framework of Luxburg (2007). The second is a diffusion-driven method designed from graph Fourier principles. We construct Laplacian-derived probe signals, smooth them using a heat-kernel filter $g_\tau(\lambda) = e^{-\tau\lambda}$, approximate this filter at scale via Chebyshev polynomials to avoid eigendecomposition, embed vertices using the filtered responses, and then cluster in the induced diffusion geometry (Shuman et al. (2013) and Defferrard, Bresson, and Vandergheynst (2016)), to test whether diffusion of Laplacian-informed signals can act as a topology-only lens for extracting structures such as communities.

The contributions of this work are as follows. First, we provide a self-contained formulation of the graph Fourier viewpoint to motivate topology-only filtering and clustering on graphs. Second, we implement a heat-filter pipeline based on Chebyshev approximation and apply it across all Elliptic snapshots. Finally, we evaluate both the baseline and the proposed method using community-quality measures like modularity and conductance, alongside a covert-relevance criterion (lift with respect to illicit labels).

The remainder of the report is organised as follows. Section 2 summarises the background required to define the Laplacian eigenbasis and functional calculus, and the graph Fourier transform. Section 3 develops intuition on toy graphs and canonical signals, motivating the use of heat filtering as a diffusion operator. Section 4 describes the Elliptic dataset and the two pipelines in detail, including temporal snapshot construction, signal design, Chebyshev approximation, and evaluation metrics. Section 4 also reports quantitative results over time and examines the representative snapshots. Section 5 concludes with the key findings and outlines methodological and evaluative directions for future work.

1 Background

In this section, we build the foundational concepts required for this project. These definitions follow the classical formulation of Harary (1969).

Definition 1.1 (Graph)**.** *A graph G is an ordered pair $G = (V, E)$, where $V = \{v_1, \ldots, v_n\}$ is a finite set of nodes, and $E \subseteq V \times V$ the set of edges between nodes.*

Definition 1.2 (Undirected Graph)**.** *If $(u, v) \in E$ implies $(v, u) \in E$, the graph is said to be undirected.*

In undirected graphs, an edge between nodes i and j will carry the same meaning regardless of whether it is viewed as connecting i to j, or j to i since the relationship is mutual by construction.

Definition 1.3 (Adjacency Matrix)**.** *Let $G = (V, E)$ be a graph with $|V| = n$. The adjacency matrix of G is the matrix $A \in \mathbb{R}^{n\times n}$ defined by*

$$A_{ij} = \begin{cases} 1, & \text{if } (v_i, v_j) \in E, \\ 0, & \text{otherwise.} \end{cases}$$

Each row and column corresponds to a node, and the presence or absence of an edge is encoded as a binary entry. In this way, the connectivity of the entire network is captured in a single matrix, allowing tools from linear algebra to be applied directly to the study of graphs.

Definition 1.4 (Degree Matrix)**.** *The degree matrix of a graph G is the diagonal matrix $D \in \mathbb{R}^{n\times n}$ whose diagonal entries are given by*

$$D_{ii} = \sum_{j=1}^{n} A_{ij},$$

with all off-diagonal entries equal to zero.

The degree matrix records the connections of each node, quantifying node participation in the network.

Definition 1.5 (Graph Laplacian)**.** *The (combinatorial) graph Laplacian is defined as $L = D - A$, where A is the adjacency matrix and D is the degree matrix of G.*

Thus, the entries of the Laplacian are described as:

$$L_{ij} = \begin{cases} d_i & i = j, \\ -A_{ij} & i \neq j \text{ and } (v_i, v_j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The Laplacian matrix measures the extent to which a graph differs at one node from its values at nearby nodes, acting as a discrete analogue of the continuous Laplace operator.

Many fundamental properties of a graph such as connectivity, diffusion behaviour, random walks, and community structure are reflected in the eigenvalues and eigenvectors of L, which form the mathematical foundation of spectral graph theory, and the Graph Fourier Transform, used throughout this project.

The set of (ordered) eigenvalues, and the set of corresponding eigenvectors, forms the *Laplacian spectrum* of a graph.

Definition 1.6 (Laplacian Spectrum of a Graph (Peter and François (2020)))**.** *Let G be a graph with Laplacian matrix L. The Laplacian spectrum of G is the set of eigenvalues $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ of L, together with their corresponding (column) eigenvectors $\{\vec{u}_k\}_{k=1}^{n}$ satisfying $L\vec{u}_k = \lambda_k \vec{u}_k$.*

For undirected graphs, the Laplacian matrix is symmetric with all eigenvalues real and non-negative, and the eigenvectors form an orthonormal basis of $\mathbb{R}^n$ (Friedman and JP (2004)). This basis plays an analogous role to the Fourier basis in classical signal processing and constitutes the foundation of the Graph Fourier Transform (Furutani et al. (2020)).

Intuitively, the Laplacian spectrum captures how information, influence, or variation propagates across the network. Small eigenvalues correspond to smooth, slowly-varying structures over the graph, while larger eigenvalues encode increasingly oscillatory patterns (Singh, Chakraborty, and Manoj (2016)). Additionally, the spectrum reveals global structural properties such as connectivity, bottlenecks, and community organisation, hence its pivotal role in spectral clustering and graph signal processing.

1.1 The Laplacian as a Linear Operator

While the Laplacian matrix L has thus far been introduced as an algebraic object, it is equally important to view it as a linear operator acting on functions defined over the nodes of the graph.

Definition 1.7 (Graph Signal)**.** *A graph signal is a function $f : V \to \mathbb{R}$ assigning a real value to each node of the graph. Equivalently, f may be represented as a vector $\vec{f} \in \mathbb{R}^n$, where the i-th entry corresponds to the value of the signal at node v_i.*

Under this interpretation, the Laplacian defines a linear mapping $L : \mathbb{R}^n \to \mathbb{R}^n$, sending a signal $\vec{f}$ to a new signal $L\vec{f}$.

The action of L on a signal measures how the value at each node deviates from the values at neighbouring nodes. This mirrors the role of the continuous Laplace operator in classical analysis, where it measures local curvature or variation of a function.

From an operator-theoretic perspective, the eigenvalue equation $L\vec{u}_k = \lambda_k \vec{u}_k$ states that each eigenvector $\vec{u}_k$ is a special signal whose shape is preserved under the action of the Laplacian, up to a scalar factor.

This interpretation provides the conceptual foundation for the Graph Fourier Transform: the Laplacian eigenvectors form a basis of elementary "graph harmonics", and arbitrary graph signals may be expanded as linear combinations of these fundamental modes.

Definition 1.8 (Graph Harmonics (Singh, Chakraborty, and Manoj (2016))). *Let G be a graph with Laplacian matrix L and eigenpairs $\{(\lambda_k, \vec{u}_k)\}_{k=1}^n$. The eigenvectors $\vec{u}_k$ of L are called the graph harmonics of G.*

Analogous to classical Fourier analysis, where sinusoidal functions arise as eigenfunctions of the continuous Laplace operator, graph harmonics play the role of elementary oscillatory components on discrete network domains (Shuman et al. (2013)).

1.2 Graph Fourier Transform

Definition 1.9 (Graph Fourier Transform). *Let $f \in \mathbb{R}^n$ be a graph signal and let $\{\vec{u}_k\}_{k=1}^n$ denote the graph harmonics, i.e., the orthonormal eigenvectors of the Laplacian matrix L. The Graph Fourier Transform (GFT) of f is defined as*

$$\hat{f}(k) = \vec{u}_k^\top \vec{f} = \sum_{i=1}^n f(i) u_k(i).$$

The coefficients $\hat{f}(k)$ quantify the contribution of the kth graph harmonic to the signal f, and therefore represent the spectral content of f in the graph frequency domain.

Definition 1.10 (Inverse Graph Fourier Transform). *Given spectral coefficients $\{\hat{f}(k)\}_{k=1}^n$, the original signal f may be reconstructed as*

$$f = \sum_{k=1}^n \hat{f}(k)\, \vec{u}_k.$$

Equivalently, if $U = [\vec{u}_1\ \vec{u}_2\ \cdots\ \vec{u}_n]$ denotes the matrix of Laplacian eigenvectors and $\hat{f}$ the vector of Fourier coefficients, then: $\hat{f} = U^\top f$, and: $f = U\hat{f}$.

This pair of transformations establishes a direct analogue of the classical Fourier transform and its inverse, with the Laplacian eigenbasis replacing the complex exponential basis on Euclidean domains (Hammond, Vandergheynst, and Gribonval (2011)).

2 Graph Signal Processing Operators

Definition 2.1 (Graph Filter (Shuman et al. (2013))). *A graph filter is a linear operator $H : \mathbb{R}^n \to \mathbb{R}^n$ acting on graph signals, which can be expressed as a function of the graph Laplacian: $H = g(L)$. Here, $g : \mathbb{R} \to \mathbb{R}$ is called the filter transfer function.*

In classical signal processing, time-invariant linear filters are diagonalised by the Fourier basis. Analogously, graph filters defined as functions of the Laplacian, $H = g(L)$, share the Laplacian eigenbasis and therefore are diagonalised by U, thus filtering is a method of independently scaling the spectral components.

Operationally, filtering *mixes* the signal values across the graph: for $f_{\text{out}} = Hf$, each output entry satisfies $(f_{\text{out}})_i = \sum_{j=1}^n H_{ij} f_j$, so the value at node i becomes a weighted combination of values from other nodes (typically nearby nodes when $H = g(L)$).

Definition 2.2 (Spectral Graph Filter (Shuman et al. (2013))). *Let $L = U\Lambda U^\top$ be the eigendecomposition of the Laplacian. A spectral graph filter with transfer function g acts on a signal f as $Hf = g(L)f = Ug(\Lambda)U^\top f$, where $g(\Lambda) = \mathrm{diag}(g(\lambda_1), \ldots, g(\lambda_n))$, and $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of L.*

Spectral filtering modifies each graph Fourier coefficient $\hat{f}(k)$ by the factor $g(\lambda_k)$ and then transforms back to the vertex domain.

Definition 2.3 (Graph (spectral) Convolution (Shuman et al. (2013))). *Let $g : \mathbb{R} \to \mathbb{R}$ be a filter transfer function. The (spectral) convolution of f with g is $f * g := g(L)f = Ug(\Lambda)U^\top f$.*

Remark 2.1 (Spectral multiplication property). Let $\hat{f} = U^\top f$ denote the graph Fourier transform of f. Then $U^\top(g(L)f) = g(\Lambda)\hat{f}$. Thus, graph convolution corresponds to elementwise multiplication of Fourier coefficients by the transfer function $g(\lambda_k)$ (Perraudin and Vandergheynst (2017)). Additionally, the word *filter* is used interchangeably within this report to refer to the operator $H = g(L)$ and its associated transfer function $g(\lambda)$.

Definition 2.4 (Polynomial Graph Filter (Defferrard, Bresson, and Vandergheynst (2016))). *A polynomial graph filter of degree K is defined as*

$$H = \sum_{k=0}^{K} a_k L^k.$$

This avoids eigendecomposition and has complexity proportional to the number of edges (Defferrard, Bresson, and Vandergheynst (2016)). Sufficiently smooth spectral filters $g(\lambda)$ can then be approximated by Chebyshev polynomials:

$$g(L) \approx \sum_{k=0}^{K} c_k T_k(\tilde{L}),$$

where T_k are Chebyshev polynomials and $\tilde{L}$ is the rescaled Laplacian (Defferrard, Bresson, and Vandergheynst (2016)).

Using direct spectral filtering requires the eigendecomposition of L, which is computationally prohibitive for large networks. Polynomial approximations allow filtering to be performed using only sparse matrix–vector multiplications, making the approach scalable to very large graphs, or networks.

3 Toy Graphs and Canonical Signals

To develop intuition for graph Fourier analysis and spectral filtering, we first consider a collection of small, "toy" graphs, and simple, canonical signals defined on their nodes.

For each graph $G = (V, E)$ with $|V| = n$, we construct its Laplacian $L = D - A$, compute its eigendecomposition $L = U\Lambda U^\top$, and treat the resultant eigenvectors $\{\vec{u}_k\}_{k=1}^{n}$ as the graph Fourier basis. A graph signal is then represented as a vector $f \in \mathbb{R}^n$, whose i-th entry corresponds to the value assigned to the node v_i.

3.1 Toy Graphs

We focus on three graph families that directly mirror the structural motifs present in covert transaction networks:

Path graph P_n. The path graph provides the simplest setting where low-frequency modes vary gradually along the chain, while high-frequency modes oscillate rapidly between adjacent nodes (Shuman et al. (2013)).

Cycle graph C_n (ring). The cycle graph restores translation symmetry (no endpoints) and behaves most like a periodic 1D signal domain. Many canonical signals (sinusoids on node index, impulses, steps) display clean spectral signatures on C_n, making it a useful reference model for interpreting frequency responses.

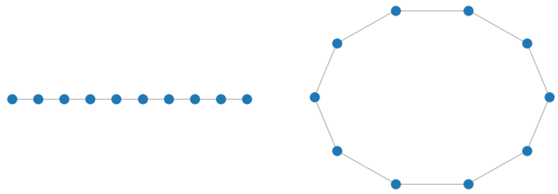

Figure 1: A simple 10 node path graph (left) and a 10 node ring graph (right).

Stochastic block model (SBM) with two blocks. The SBM is the toy model most directly aligned with community detection: it contains two densely connected groups of nodes (blocks) with sparse connections between them. In this setting, the first nontrivial eigenvectors concentrate on block membership, and low-frequency filtering tends to preserve blockwise-constant patterns (Abbe (2018)). This makes the SBM the most relevant toy graph for the project's central premise as community structure is primarily a low-frequency phenomenon.

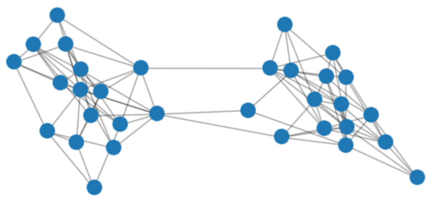

Figure 2: A 36 node stochastic block model graph, generated with Python's NetworkX library using $p = 0.35$ and $q = 0.02$

3.2 Canonical Toy Signals

On each toy graph $G = (V, E)$ with $|V| = n$, we define a small set of canonical signals $f : V \to \mathbb{R}$. Each signal is chosen because it isolates a particular spectral behaviour of interest.

(i) Step / community-indicator signal.

Definition 3.1 (Community-indicator signal)**.** *Let $G = (V, E)$ be a graph whose nodes are partitioned into two communities $V = V_1 \cup V_2$, $V_1 \cap V_2 = \varnothing$. The community-indicator signal $f_{\text{comm}} : V \to \mathbb{R}$ is defined by*

$$f_{\text{comm}}(v_i) = \begin{cases} +1, & v_i \in V_1, \\ -1, & v_i \in V_2. \end{cases}$$

A blockwise-constant signal is the idealised model of a community label: it is nearly constant inside each community and changes sharply across the cut (Luxburg (2007)). On SBM graphs, this signal aligns closely with low Laplacian eigenvectors and is therefore concentrated at low graph frequencies (Shuman et al. (2013)).

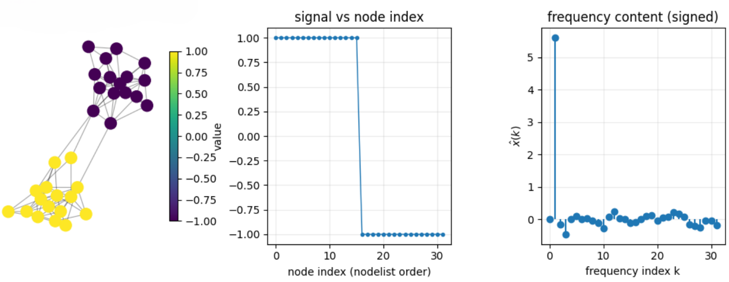

Figure 3: The SBM with node colours corresponding to signal values (left). The signal values against the node index (middle). Plot of the frequency domain of the SBM with the community-indicator signal applied (right).

(ii) Ramp signal.

Definition 3.2 (Ramp signal)**.** *Let $G = (V, E)$ be a graph with $|V| = n$, and fix an ordering $V = \{v_1, v_2, \ldots, v_n\}$. The ramp signal $f_{\text{ramp}} : V \to \mathbb{R}$ is defined by*

$$f_{\text{ramp}}(v_i) = \frac{2i}{n-1} - 1, \qquad i = 0, 1, \ldots, n.$$

A slowly varying signal over the node ordering (especially on P_n or C_n) concentrates strongly in the lowest eigenvectors. It is a canonical example of a low-frequency signal. Also note that f_{ramp} depends on the chosen node ordering; on graphs without a natural linear order, it should be interpreted as a generic low-to-high trend along the chosen indexing of V.

(iii) Sinusoid signal.

Definition 3.3 (Sinusoid signal)**.** *Let $G = (V, E)$ be a cycle graph with $|V| = n$, and fix an ordering $V = \{v_1, v_2, \ldots, v_n\}$. For a fixed integer $k \in \{1, \ldots, \lfloor n/2 \rfloor\}$, the sinusoid signal $f_{\text{sine}} : V \to \mathbb{R}$ is defined by*

$$f_{\text{sine}}(v_i) = \sin\left(\frac{2\pi k i}{n}\right), \qquad i = 1, \ldots, n.$$

Because C_n is a discrete periodic domain, its Laplacian eigenvectors closely resemble discrete Fourier modes. For each nonzero frequency, the corresponding Laplacian eigenvalue has multiplicity two, with sine- and cosine-like eigenvectors spanning the associated eigenspace. As a result, a sinusoid defined over the node index does

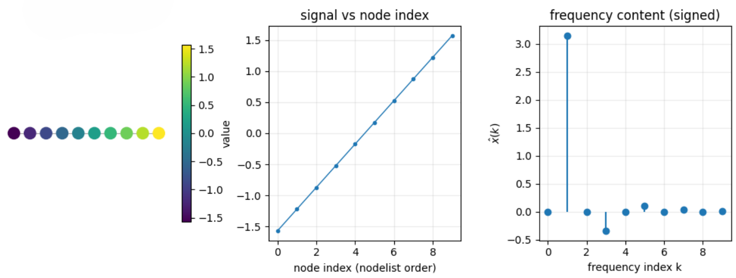

Figure 4: The path graph with node colours corresponding to signal values (left). The signal values against the node index (middle). Plot of the frequency domain of the path graph with the ramp signal applied (right).

not align with a single ordered eigenvector, but rather with a low-dimensional eigenspace corresponding to one graph frequency band.

Consequently, the graph Fourier transform of f_{sine} concentrates sharply on the pair of eigenvalues associated with that frequency, which may appear at indices $(\lambda_{2k}, \lambda_{2k+1})$ depending on the eigenvalue ordering. This behaviour illustrates that graph frequencies are fundamentally associated with eigenspaces rather than individual eigenvectors, and confirms that spectral filters operate on frequency bands rather than single modes.

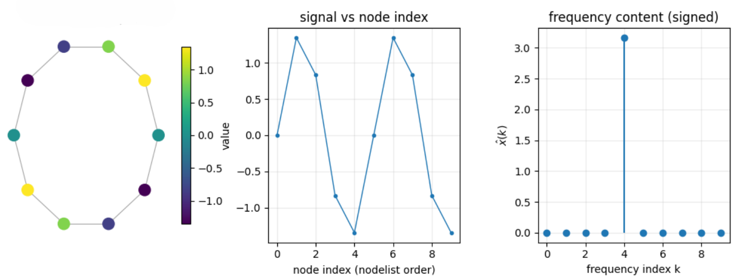

Figure 5: The cycle graph with node colours corresponding to signal values (left). The signal values against the node index (middle). Plot of the frequency domain of the cycle graph with the sinusoid signal applied, using k = 2 (right).

3.3 Toy Filters and What They Demonstrate

Definition 3.4 (Heat-kernel low-pass filter (Shuman et al. (2013)).). *The heat-kernel is a soft low-pass filter,* $g_\tau(\lambda) = e^{-\tau\lambda}, H_\tau = g_\tau(L)$.

Here, larger τ increases smoothing by attenuating higher frequencies more aggressively. The heat kernel is particularly important because it is spectrally interpretable, stable, and efficiently approximable by Chebyshev polynomials without eigendecomposition (Defferrard, Bresson, and Vandergheynst (2016)).

4 Elliptic Bitcoin Dataset

Studied in this project is the Elliptic Bitcoin dataset (Elmougy and Liu 2023). This network consists of 49 timesteps, and totals over 2×10^5 nodes and more than 2×10^5 edges. Nodes represent accounts, and edges represent bitcoin transactions or the flow of bitcoin. Nodes have been labelled as illicit, licit, or unknown, depending on the respective nature of the transactions. All illicit labels are used solely for post hoc evaluation and interpretation, and are not incorporated into the clustering process. In all analyses, the dataset is treated as undirected and unweighted.

The majority of the nodeset is labelled unknown, which categorises the Elliptic Bitcoin dataset as a covert network (Dey and Medya (2019)), and enforces a reliance on methods that leverage the structural properties of the network and do not depend on node attributes. In light of these constraints, we use spectral clustering as a baseline community detection algorithm (following the formulation of Luxburg (2007)). Then, we design a novel heat filter method in which Laplacian-based signals are subjected to spectral filtering via Chebyshev approximation, followed by dimensional embedding and clustering.

4.1 Temporal Snapshot Construction

For each time step t, a snapshot graph $G_t = (V_t, E_t)$ is constructed by selecting nodes and edges present at that time. To ensure spectral stability, analysis is restricted to the largest connected component of each snapshot. Let $n_t = |V_t|$ denote the number of nodes in this component. Additionally, let $\vec{y}_t \in \{-1, 0, 1\}^{n_t}$ denote node labels such that -1 denotes unknown nodes, 0 denotes licit nodes, and 1 denotes illicit nodes.

For each snapshot the adjacency matrix A_t, degree matrix D_t and Laplacian matrix L_t are constructed. The normalised Laplacian is defined as $L_t = I - D_t^{-1/2} A_t D_t^{-1/2}$, which serves as the primary operator utilised in the methodology. The eigendecomposition is then defined as usual: $L_t = U_t \Lambda_t U_t^\top$.

Remark 4.1. The main difference between the Laplacian and the normalised Laplacian is the normalised Laplacian restrains degree-bias, where the unnormalised Laplacian does not. Due to the scale of the Elliptic Bitcoin dataset, it is not acceptable to use the unnormalised Laplacian since it leaves the matrix unbalanced and ill-conditioned (Chung (1996)).

4.2 Signal Construction

Two Laplacian-driven probe signals are defined to examine structural behaviour.

Adjacency connectivity signal: $\vec{x}_A = A_t \mathbf{1}$

This signal aggregates neighbour connectivity and is equivalent to node degree in unweighted networks. It provides the typical degree-centrality; measuring local connectivity intensity.

Laplacian Signal: $\vec{x}_L = L_t \vec{x}_A$

This signal probes the discrete Laplacian response of connectivity, highlighting regions of heterogenic connectivity structures. Consequently, $\vec{x}_L$ emphasises potential community boundaries and structural transitions.

4.3 Spectral Clustering

As a benchmark method, spectral clustering is applied following the framework of Luxburg (2007). The first k non-trivial Laplacian eigenvectors define an embedding of nodes into a lower dimensional spectral space. The number of clusters is selected via a robust eigengap heuristic, and k-means clustering is performed on the embedding, optimising cluster separability. This algorithm is utilised as a baseline for community detection.

4.4 Heat Filtering Graph Signal Processing Pipeline

This project's main method employs the heat kernel defined in Definition 3.4, simulating the diffusion of signals across the network, and attenuating high-frequency topological variation.

Since direct eigendecomposition is computationally prohibitive at the scale of the datset, the filter is approximated using Chebyshev polynomials,

$$g(L_t) \approx \sum_{k=0}^{K} c_k T_k(\tilde{L}_t),$$

where $\tilde{L}_t$ is the rescaled Laplacian and T_k denotes Chebyshev polynomials. This approximation enables filtering without explicit spectral decomposition.

Filtered probe signals are computed as $\vec{x}_A^{(f)} = g(L_t)\vec{x}_A$, $\vec{x}_L^{(f)} = g(L_t)\vec{x}_L$.

Nodes are then embedded into a diffusion geometry defined by filtered probe signals, $Z_t = [\vec{x}_A^{(f)}, \vec{x}_L^{(f)}]$, where proximity reflects similarity of diffusion behaviour. Clustering is performed via k-means on this embedding, yielding diffusion-driven partitions of the network.

To interpret cluster relevance to covert behaviour, clusters are evaluated using illicit enrichment metrics. For each cluster C with known labels, illicit density is defined as

$$\rho_C = \frac{|\{v \in C : y_v = 1\}|}{|\{v \in C : y_v \neq -1\}|}.$$

Lift is computed as $\text{lift}(C) = \frac{\rho_C}{\rho_{\text{global}}}$, where ρ_{global} denotes illicit density across the snapshot.

Clusters with maximal enrichment are selected for visual analysis. Induced subgraphs are extracted and pruned using connectivity-preserving strategies to ensure interpretability while retaining illicit nodes.

4.5 Evaluation Metrics

Three metrics are employed to evaluate clustering behaviour.

The first is modularity, defined $Q = \sum_c \frac{E_c}{m} - \gamma(\frac{k_c}{2m})^2$, where E_c is the number of edges within community c, k_c is the sum of degrees of nodes in community c, m is the total edge count for the network, and γ is a resolution parameter. This metric measures the effectiveness of community divisions by comparing the actual edge density within clusters against the expected edge density seen in random networks (Clauset, Newman, and Moore (2004)). Community divisions that yield high modularity indicate dense clusters with few edges between them.

The second cluster evaluation metric is conductance, defined $\phi(c) = \frac{E_{c,\bar{c}}}{\min(E_c, E_{\bar{c}})}$. Here, E_c is again the number of edges in community c, $E_{\bar{c}}$ the number of edges *not* in c (or its complement), and then $E_{c,\bar{c}}$ the number of edges between c and its complement. Conductance is a measure of graph connectivity with respect to a partition of interest. That is, low conductance will imply low connectivity between two node sets, and if we define these node sets as communities, we get a measure of edge sparsity between communities.

Extending this, we can evaluate a network's global conductivity by selecting the minimum conductance over all communities $\Phi = \min_{c \in V} \phi(c)$. Finally, we assess the covert relevance of detected communities via the lift metric defined in the section above.

4.6 Results and Discussion

Across the 49 snapshot graphs, spectral clustering consistently produced partitions with low conductance and, in many timesteps, higher modularity than the heat-filter pipeline. In contrast, the heat-filter method yielded relatively stable but generally small modularity scores and higher conductance, indicating that the diffusion embedding, as currently instantiated, does not reliably recover sharply separated edge cuts. Neither method has any consistent performance in this metric, however there is notably less variation yielded by the heat filter method.

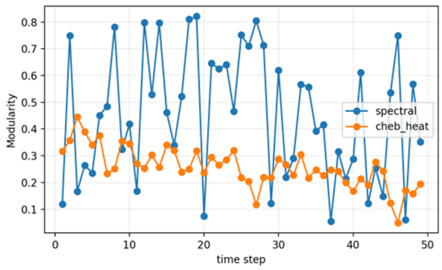

Figure 6: Plot of average cluster modularity, spectral clustering in blue and the heat filter method in orange.

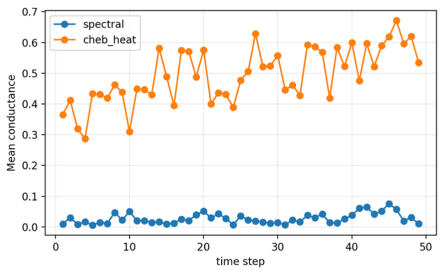

Figure 7: Plot of cluster conductance, spectral clustering in blue and the heat filter method in orange.

Due to pagelimit restrictions, only a small selection of snapshot outputs are examined in this report.

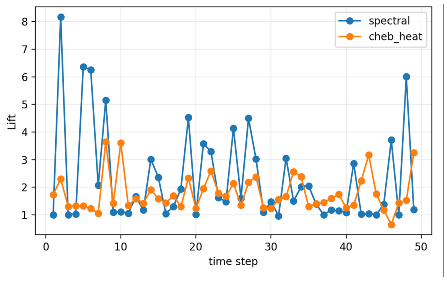

Figure 8: Plot of cluster conductance, spectral clustering in blue and the heat filter method in orange.

Timestep 14

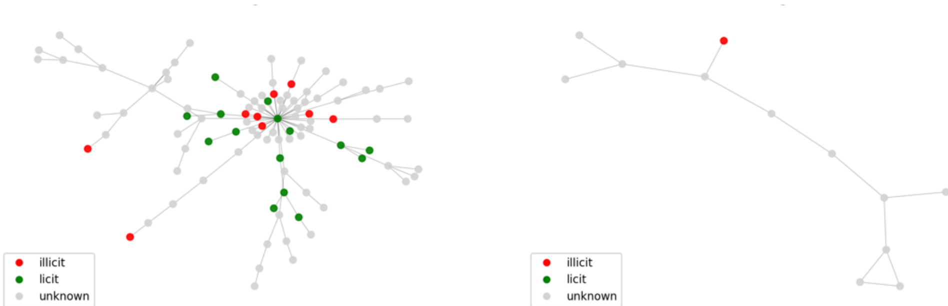

Figure 9: Spectral clustering highest lift cluster (left) contains 97 nodes and 98 edges. The heat filter method (right) yields a highest lift cluster of 12 nodes and 12 edges.

Timestep 42

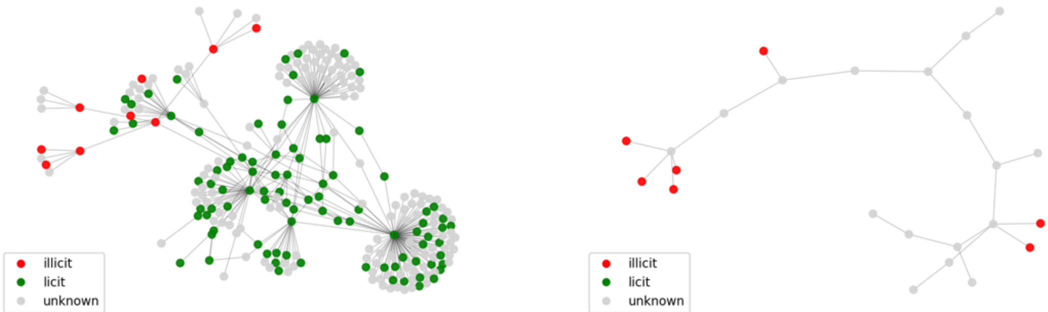

Figure 10: Spectral clustering highest lift cluster (left) contains 250 nodes and 276 edges. The heat filter method (right) yields a highest lift cluster of 24 nodes and 23 edges.

Nevertheless, in several snapshots the diffusion method extracted structures that were qualitatively distinct from those produced by spectral clustering, and in at least one representative case (Timestep 13) both methods converged to the same star-shaped cluster: a motif commonly associated with transaction patterns in financial crime networks.

Timestep 13

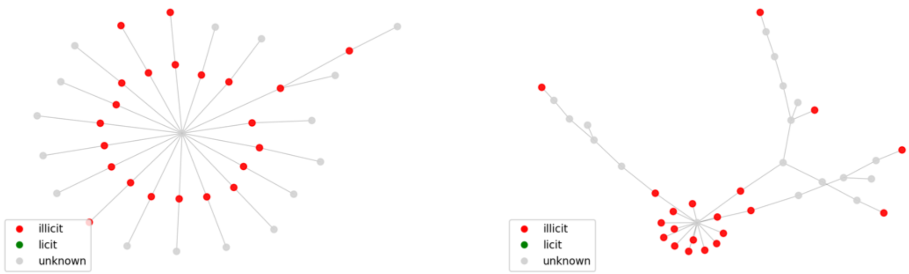

Figure 11: Spectral clustering highest lift cluster (left) contains 39 nodes and 38 edges. The heat filter method (right) yields a highest lift cluster of 38 nodes and 37 edges.

This suggests that while classical spectral partitioning is better aligned with cut-based notions of community in this setting, diffusion-style filtering can still expose recurrent meso-scale motifs - when the topology strongly supports them.

Two key observations follow. First, "good" communities in covert transaction graphs are not always well captured by a single cut-based objective such as modularity or conductance; motif-like structures (e.g. star/hub patterns) may be operationally meaningful while scoring poorly under standard community quality metrics. Second, the effectiveness of topology-only diffusion embeddings is highly sensitive to modelling choices, especially the probe signals and the embedding dimension.

5 Conclusion and Future Work

This project investigated whether the Laplacian geometry of a covert transaction network alone is sufficient in revealing meaningful community structures, without using node features or seeded labels. Studying the Elliptic Bitcoin transaction graph, we compared a classical spectral clustering baseline against a topology-only diffusion method based on Laplacian-driven probe signals passed through a heat-kernel filter approximated by Chebyshev polynomials.

Naturally, future work could focus on methodological refinement and evaluation refinement. On the methodological side, the diffusion pipeline can be primarily strengthened by replacing the two-dimensional embedding $[\vec{x}_A^{(f)}, \vec{x}_L^{(f)}]$ with a higher-dimensional diffusion feature map, and secondly, exploring clustering objectives that better match the geometry induced by diffusion (e.g. density-based clustering or regularised k-means). On the evaluation side, the analysis could incorporate motif- and anomaly-oriented criteria in addition to cut metrics, and could report temporal stability across snapshots (such as structure persistence, or recurrence over time).

Overall, the results support the core premise that Laplacian-based geometry contains exploitable information about covert network structure, but they also show that extracting useful communities requires aligning the signal model and filter design with the notion of "structure" one seeks to detect: edge cuts, diffusion similarity, or criminally-relevant motifs.

References

Abbe, Emmanuel (2018). "Community Detection and Stochastic Block Models: Recent Developments". In: *Journal of Machine Learning Research* 18.177, pp. 1–86. URL: http://jmlr.org/papers/v18/16-480.html.

Chung, Fan R. K. (1996). "Spectral Graph Theory". In: URL: https://api.semanticscholar.org/CorpusID:60624922.

Clauset, Aaron, M. E. J. Newman, and Cristopher Moore (Dec. 2004). "Finding community structure in very large networks". In: *Physical Review E* 70.6. ISSN: 1550-2376. DOI: 10.1103/physreve.70.066111. URL: http://dx.doi.org/10.1103/PhysRevE.70.066111.

Defferrard, Michaël, Xavier Bresson, and Pierre Vandergheynst (2016). "Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering". In: *Neural Information Processing Systems*. URL: https://api.semanticscholar.org/CorpusID:3016223.

Dey, Palash and Sourav Medya (2019). "Covert Networks: How Hard is It to Hide?" In: *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. AAMAS '19. Montreal QC, Canada: International Foundation for Autonomous Agents and Multiagent Systems, pp. 628–637. ISBN: 9781450363099.

Elmougy, Youssef and Ling Liu (2023). "Demystifying Fraudulent Transactions and Illicit Nodes in the Bitcoin Network for Financial Forensics". In: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. KDD '23. Long Beach, CA, USA: Association for Computing Machinery, pp. 3979–3990. ISBN: 9798400701030. DOI: 10.1145/3580305.3599803. URL: https://doi.org/10.1145/3580305.3599803.

Friedman, J and Tillich JP (2004). "Wave equations for graphs and the edge-based Laplacian". In: *Pacific Journal of Mathematics* 216.2, pp. 229–266. DOI: http://dx.doi.org/10.2140/pjm.2004.216.229. URL: https://msp.org/pjm/2004/216-2/p03.xhtml.

Furutani, Satoshi et al. (2020). "Graph Signal Processing for Directed Graphs Based on the Hermitian Laplacian". In: *Machine Learning and Knowledge Discovery in Databases*. Cham: Springer International Publishing, pp. 447–463. ISBN: 978-3-030-46150-8. DOI: 10.1007/978-3-030-46150-8_27.

Hammond, David K., Pierre Vandergheynst, and Rémi Gribonval (2011). "Wavelets on graphs via spectral graph theory". In: *Applied and Computational Harmonic Analysis* 30.2, pp. 129–150. ISSN: 1063-5203. DOI: https://doi.org/10.1016/j.acha.2010.04.005. URL: https://www.sciencedirect.com/science/article/pii/S1063520310000552.

Harary, F. (1969). *Graph Theory*. Massachusetts, USA: Addison-Wesley.

Luxburg, Ulrike von (2007). "A Tutorial on Spectral Clustering". In: *Statistics and Computing* 17.4, pp. 395–416.

Perraudin, Nathanaël and Pierre Vandergheynst (2017). "Stationary Signal Processing on Graphs". In: *IEEE Transactions on Signal Processing* 65.13, pp. 3462–3477. DOI: 10.1109/TSP.2017.2690388.

Peter, Wills and G. Meyer François (Feb. 2020). "Metrics for graph comparison: A practitioner's guide". In: *PLOS ONE* 15.2, pp. 1–54. DOI: 10.1371/journal.pone.0228728. URL: https://doi.org/10.1371/journal.pone.0228728.

Shuman, David I et al. (2013). "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains". In: *IEEE Signal Processing Magazine* 30.3, pp. 83–98. DOI: 10.1109/MSP.2012.2235192.

Singh, Rahul, Abhishek Chakraborty, and B. S. Manoj (2016). "Graph Fourier transform based on directed Laplacian". In: *2016 International Conference on Signal Processing and Communications (SPCOM)*, pp. 1–5. DOI: 10.1109/SPCOM.2016.7746675.