Transfer Operators : Directed Graph Clustering. [4]

James Zoryk
Computational Applied Mathematics MSc Programme - University of Edinburgh

Abstract

We illustrate how transfer operators can be used to detect clusters or community structures in directed graphs. Our goal is to define transfer operators on graphs and to apply data-driven methods for the approximation of these operators.

Introduction.

Transfer operators are commonly used in the analysis of dynamical systems due to the valuable information embedded in their eigenvalues and eigenfunctions, which reveals crucial insights into the overall characteristics of the underlying processes. The associated eigenfunction can be utilised for the identification of metastable sets.

The spectral analysis of transfer operators such as the **Perron–Frobenius** or **Koopman** operator is by now a well established technique in molecular conformation analysis [2].

Although transfer operators have been known for a long time, they recently gained renewed attention due to the availability of large data sets and powerful machine learning methods to estimate these operators and their spectral properties from simulation or measurement data.

Coherent Sets

We define **Coherent sets** which can be regarded as time dependent metastable sets: Let s^τ be the flow associated with a dynamical system and τ be a fixed lag time. Two sets $\mathbb{A}$ and $\mathbb{B}$ form a coherent pair if $s^\tau(\mathbb{A}) \approx \mathbb{B}$ and $s^{-\tau}(\mathbb{B}) = \mathbb{A}$. In other words the set $\mathbb{A}$ is almost invariant under the forward-back dynamics.

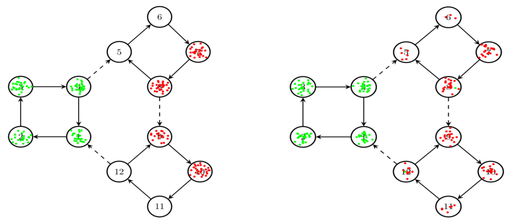

Figure 1. (a) Two distinct groups of random walkers (depicted in green) (depicted in red). Solid edges carry a weight of 1, dashed edges weight of 0.01. (b) Positions after one step forward and one step backward. The green set exhibits coherence.[4]

Transition probability matrix

Here we define the **row-stochastic transition probability** matrix $S = D_\theta^{-1}A$, with A being the **weighted adjacency** matrix and D_θ is the matrix of **out-degree**:

$$D_\phi = \mathrm{Diag}\,(\theta(v_1), \ldots, \theta(v_n)) \quad \text{and} \quad \theta(v_i) = \sum_{j=1}^{n} a_{i,j}.$$

That is, $s_{i,j}$ is the probability that a random walker starting at the point v_i will go to v_j in one step.

Transfer Operators: Definitions

For a function $\rho \in \mathbb{U}$, the **Perron-Frobenius** operator $\mathcal{P} : \mathbb{U} \to \mathbb{U}$, and similarly for a for a function $f \in \mathbb{U}$, the **Koopman** operator $\mathcal{K} : \mathbb{U} \to \mathbb{U}$, hence we define them as:

$$\mathcal{P}\rho(v_i) = \sum_{j=1}^{n} s_{j,i}\rho(v_i), \qquad \mathcal{K}f(v_i) = \sum_{j=1}^{n} s_{i,j}f(v_i).$$

Spectral Clustering

The random-walk normilized Laplacain is defined [3] as

$$L_\mathcal{K} = I - D_\theta^{-1}A = I - K,$$

where K is the matrix representation of the Koopman operator. For directed graphs we compute the eigenvectors associated with the largest eigenvalues of the forward-backward or backward-forard operators in order to detect coherent sets. It has been shown that these operators are real-valued [4, 3].

Algorithm 1 Cluster Assignment Algorithm

Require: Matrix F, Number of clusters k
Ensure: Cluster assignments

1: **Step 1:** Compute the k largest eigenvalues λ_ℓ and associated eigenvectors ϕ_ℓ of F
2: **Step 2:** Define $\Phi = [\phi_1, \ldots, \phi_k] \in \mathbb{R}^{n\times k}$ and let r_i denote the ith row of Φ
3: **Step 3:** Cluster the points $\{r_i\}_{i=1}^n$ using, e.g., k-means

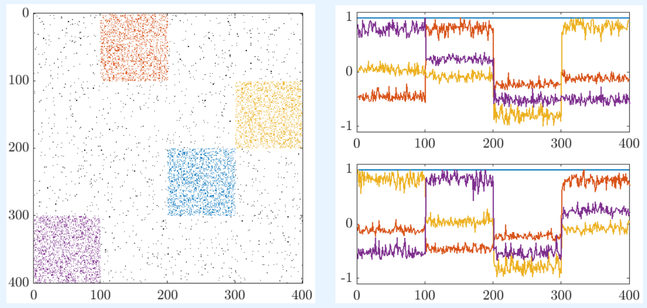

Figure 2. a) Adjacency matrix of a directed graph comprising four clusters. (b) Four dominant eigenfunctions φ_l(top) and ψ_l (bottom). [4]

Numerical Approximation

The transfer operators can be estimated from a random walk data. Given m random walkers $x^{(i)}$ sampled from the distribution μ, each random walker takes a single step forward according to the transition probability matrix S, with the final position of the random walker being denoted $y^{(i)}$.

Can define the data matrices $\mathbf{\Phi}_x, \mathbf{\Phi}_y \in \mathbb{R}^{r\times m}$ by

$$\mathbf{\Phi}_x = \left[\phi(x^{(1)}) \ldots \phi(x^{(m)})\right] \quad \text{and} \quad \mathbf{\Phi}_y = \left[\phi(y^{(1)}) \ldots \phi(y^{(m)})\right]$$

Furthermore, consider the numerical **Galerkin Approximation** $\hat{G}_{xx}, \hat{G}_{yy}, \hat{G}_{xy} \in \mathbb{R}^{r\times r}$ which can be defined as:

$$\hat{G}_{xx} = \frac{1}{m}\mathbf{\Phi}_x\mathbf{\Phi}_x^T, \quad \hat{G}_{yy} = \frac{1}{m}\mathbf{\Phi}_y\mathbf{\Phi}_y^T, \quad \hat{G}_{xy} = \frac{1}{m}\mathbf{\Phi}_x\mathbf{\Phi}_y^T,$$

Then it can be shown that as $m \to \infty$ we can approximate the transfer operators as:

- $\hat{G}_{xx}^{-1}\hat{G}_{xy}$ is a numerical Galerkin approximation of the **Koopman** operator $\mathcal{K}$.
- $\hat{G}_{xx}^{-1}\hat{G}_{yx}$ is a numerical Galerkin approximation of the **Perron-Frobenius** operator $\mathcal{P}$.

Numerical Results

In this study, 300 weighted graphs with heterogeneous node-degree distributions and community sizes were generated using [5]. All graphs consisted of 500 vertices and were characterized by four to six clusters. Three distinct test cases were considered: (1) exclusively dense blocks on the diagonal, (2) exclusively off-diagonal dense blocks, and (3) a combination of both.

Evaluating the **degree-discounted bibliometric symmetrization** (DDBS) [6], **Hermitian clustering approach** (Herm-RW) [1], and Algorithm 1 involves comparing their performance. For each method and test case, we calculate the average adjusted Rand index (ARI) and the average percentage of misclassified vertices (NMV).

Table 1. Comparison of DDBS, Herm-RW, and Algorithm 1

	DDBS		Herm-RW		Algorithm 1	
	ARI	NMV (%)	ARI	NMV (%)	ARI	NMV (%)
diagonal blocks	0.954	3.938	0.059	68.262	0.993	0.558
off-diagonal blocks	0.953	4.341	0.927	6.681	0.993	0.629
mixed	0.958	3.742	0.325	49.933	0.994	0.531

The clustering method based on the transfer operator demonstrates effective performance across all graph types under consideration, yielding slightly superior results.

32-bit adder

The circuit exhibits a relatively straightforward structure, and the spectral clustering algorithm identifies 32 blocks of nearly equal size. When we rearrange the adjacency matrix according to the assigned cluster numbers, we achieve a block matrix characterized by minimal nonzero entries in the off-diagonal blocks, as illustrated in the following.

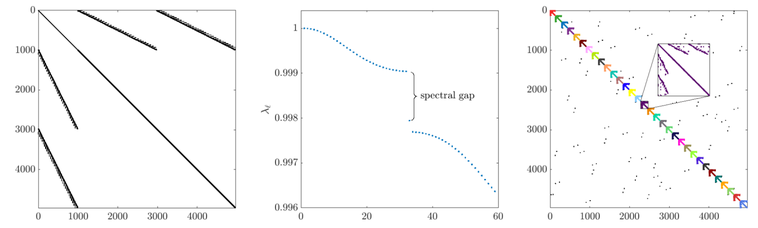

Figure 3. (a) Configuration of the original adjacency matrix for the 32-bit adder. (b) Principal eigenvalues of the forward–backward operator reveal a spectral gap between the 32nd and 33rd eigenvalue.(c) Adjacency matrix after permutation, wherein the vertices are rearranged based on the identified clusters. [4]

References

[1] Mihai Cucuringu, Huan Li, He Sun, and Luca Zanetti. Hermitian matrices for clustering directed graphs: insights and applications. In *International Conference on Artificial Intelligence and Statistics*, pages 983–992. PMLR, 2020.

[2] Stefan Klus, Andreas Bittracher, Ingmar Schuster, and Christof Schütte. A kernel-based approach to molecular conformation analysis. *The Journal of Chemical Physics*, 149(24), 2018.

[3] Stefan Klus and Nataša Djurdjevac Conrad. Koopman-based spectral clustering of directed and time-evolving graphs. *Journal of Nonlinear Science*, 33(1):8, 2023.

[4] Stefan Klus and Maia Trower. Transfer operators on graphs: Spectral clustering and beyond. *arXiv preprint arXiv:2305.11766*, 2023.

[5] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80:016118, Jul 2009.

[6] Venu Satuluri and Srinivasan Parthasarathy. Symmetrizations for clustering directed graphs. Association for Computing Machinery, 2011.