Topological Community Detection: A Sheaf-Theoretic Approach

Arne Wolf[1,2,†] and Anthea Monod[1,†]

1 Department of Mathematics, Imperial College London, UK
2 London School of Geometry and Number Theory, UK

† Corresponding e-mails: a.wolf22@imperial.ac.uk, a.monod@imperial.ac.uk

Abstract

We propose a model for network community detection using topological data analysis, a branch of modern data science that leverages theory from algebraic topology to statistical analysis and machine learning. Specifically, we use cellular sheaves, which relate local to global properties of various algebraic topological constructions, to propose three new algorithms for vertex clustering over networks to detect communities. We apply our algorithms to real social network data in numerical experiments and obtain near optimal results in terms of modularity. Our work is the first implementation of sheaves on real social network data and provides a solid proof-of-concept for future work using sheaves as tools to study complex systems captured by networks and simplicial complexes.

Keywords: Cellular sheaves; community detection; modularity; opinion dynamics; topological data analysis.

1 Introduction

Networks are used to describe, study, and understand complex systems in many scientific disciplines. One of the most important features in complex systems that networks are able to capture is the presence of *communities*. In networks, communities can be seen as partitioning a graph into clusters, which are subsets of vertices with many edges connecting the vertices within the subset, and comparatively fewer edges connecting to different subsets in the rest of the network. These clusters or communities can be considered as relatively independent components of a graph. The problem of *community detection* is to locate those clusters in networks which are more strongly connected than the whole network is, on average. Community detection is a challenging and active area of research in network science: a key aspect that makes the problem difficult is that there is no single, universally accepted definition of a community within a network and it is largely dependent on context or the specific system being studied.

Topological data analysis (TDA) is a recently-emerged approach to data science that uses principles from pure mathematics to extract meaningful information from large and complex datasets that may not possess a rigorous metric or vector space structure which is often required in classical data analysis. In our work, we focus on *sheaves*, which are a tool that relates local to global properties of various constructions in algebraic topology and have been used to reinterpret and generalize many significant results in classical geometry and algebraic topology. Sheaves allow for information to be assigned to subsets of topological spaces; when the topological space is a network and the information is vector space-valued, a computational framework for sheaves becomes available similar to that of *persistent homology*, which is a well-developed and the most widely-used tool in TDA; see Curry ([2014](#)) for complete details.

In this paper, we propose a topological approach to community detection based on sheaves. Specifically, we show that sheaves may be used to rigorously model the problem of community detection on a network and propose three novel, sheaf-based community detection algorithms. We test and compare their performance with numerical experiments on a real-world benchmarking dataset and show that we are able to attain near optimal community detection results.

1

Related Work. Particularly relevant to our work, sheaves have been previously used to model various dynamics of opinions over social networks, including bounded confidence models, stubbornness, and the formation of lies Hansen & Ghrist (2020). In a similar spirit, propagation of gossip has also been theoretically modeled using sheaves Riess & Ghrist (2022). However, it is important to note that no implementations nor applications to real data exist of these sheaf social network models. In a non-topological setting, the formation of opinion clusters has been investigated in the bounded confidence model Hegselmann & Krause (2019); another contrast to our work is that they use dynamics with discrete time steps and sharp confidence bounds. Very recently, the only other TDA approach to community detection that we are aware of was proposed Schindler & Barahona (2023), however, it is based on persistent homology, rather than sheaves, as in our work.

2   Background: Sheaves and Social Networks

In this section, we define sheaves and present sheaf-theoretic notions in the context of social networks that will be used in our proposed community detection algorithms.

2.1   Sheaves and Sheaf Cohomology

Sheaves assign data to open subsets of topological spaces X in a consistent manner. This compatibility is what enables them to relate *local* (i.e., the pure data) and *global* (i.e., having an assignment of compatible data) properties of X. *Cellular* sheaves are a special case where X is a cellular complex (which is a generalization of a simplicial complex) and the data are vector spaces. In this paper, we will further restrict to the case that the cellular complex is a graph $G = (V, E)$; we consider undirected graphs without loops or multiple edges. We write edges as ordered tuples $e = (v_1, v_2)$, ordered arbitrarily. Under these restrictions to graphs, we now outline how important concepts from graph theory generalize to the topological setting of sheaves which will be relevant to our work further on.

The graph G becomes a topological space when exploiting the fact that the *subface relation* allows G to be viewed as a partially ordered set and to be equipped with the Alexandrov topology. A sheaf on G turns out to be uniquely determined by the following data Curry (2014).

Definition 1. A *(cellular) sheaf* $\mathcal{F}$ assigns finite-dimensional real vector spaces, called *stalks*, $\mathcal{F}(v)$ to each vertex and $\mathcal{F}(e)$ to each edge of a graph, and a linear map $\mathcal{F}_{v \subset e} : \mathcal{F}(v) \to \mathcal{F}(e)$ (called a *restriction*) to each incidence $v \subset e$.
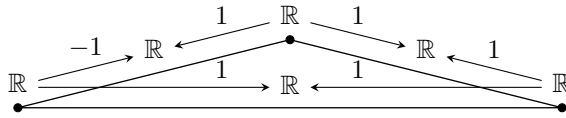

Figure 1: Example of a cellular sheaf

Figure 1 illustrates a cellular sheaf. Other examples that can be defined for any graph G are the *constant sheaves* $\mathbb{R}^n$ for $n \in \mathbb{N}$, where all stalks are $\mathbb{R}^n$ and all restrictions the identity map.

Algebraic topology is a field of pure mathematics that uses abstract algebra to study topological spaces; specifically, it defines algebraic ways of counting properties of topological spaces that are left unaltered under continuous deformations of the topological space, such as stretching or compressing. Such properties are referred to as *invariants*; *cohomology groups* are examples of such invariants. Sheaves over topological spaces give rise to *sheaf cohomology groups*. In our setting, these are obtained from a collection of vector spaces and maps between them, called *the cochain complex* (Shepard, 1985, Theorem 1.4.2)

$$0 \longrightarrow C^0(G, \mathcal{F}) \overset{\delta}{\longrightarrow} C^1(G, \mathcal{F}) \longrightarrow 0 \longrightarrow \cdots$$

with cochain groups

$$C^0(G, \mathcal{F}) := \bigoplus_{v \in V} \mathcal{F}(v), \qquad C^1(G, \mathcal{F}) := \bigoplus_{e \in E} \mathcal{F}(e).$$

Here, the linear *coboundary map* $\delta : C^0(G, \mathcal{F}) \to C^1(G, \mathcal{F})$ is defined by acting linearly on stalks. On $\mathcal{F}(v)$, it acts according to

$$x_v \mapsto \sum_{e=(v_k,v)} \mathcal{F}_{v \subset e}\, x_v - \sum_{e=(v,v_l)} \mathcal{F}_{v \subset e}\, x_v.$$

The sheaf cohomology groups of interest are then

$$H^0(G, \mathcal{F}) = \ker(\delta) \qquad H^1(G, \mathcal{F}) = C^1(G, \mathcal{F})/\mathrm{im}(\delta) = \mathrm{coker}(\delta).$$

Example 2. The graph in Figure 1 has trivial cohomology with respect to the illustrated sheaf, even though the classical (e.g., simplicial) cohomology is nontrivial. This shows that these two cohomology theories need not agree.

Sheaf cohomology is in fact a generalization of classical cohomology, because for the constant sheaf $\underline{\mathbb{R}}^1$, the sheaf cohomology yields precisely the cellular cohomology of G. In this case, $\delta = B^\top$ is the transpose of the *signed incidence matrix* B which is defined by

$$B_{v,e} = \begin{cases} 1 & \exists\, w \in V : e = (v,w), \\ -1 & \exists\, w \in V : e = (w,v), \\ 0 & \text{otherwise.} \end{cases}$$

Recall that the classical *graph Laplacian* can be obtained as $L = BB^\top$. This notion may be generalized to obtain the following definition.

Definition 3. For a cellular sheaf $\mathcal{F}$ over G, the *sheaf Laplacian* is $L_\mathcal{F} := \delta^\top \delta$.

It can be shown that neither sheaf cohomology nor the sheaf Laplacian depend on the initially chosen orientation.

2.2 Discourse Sheaves and Opinion Dynamics

By considering social networks G where persons are modeled by vertices and acquaintanceship is modeled by connections, the distributions of opinions may be expressed by a *discourse sheaf* $\mathcal{F}$ (Hansen & Ghrist, 2020); see Figure 2 for an illustration.
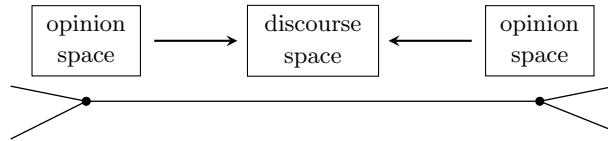

Figure 2: Sketch of a discourse sheaf

Each person v is assigned an *opinion space* $\mathbb{R}^{n_v} = \mathcal{F}(v)$ and *opinion* $x_v \in \mathcal{F}(v)$. A basis of $\mathcal{F}(v)$ can be seen as collection of basic topics that v cares about and the component of x_v in a basis direction expresses the opinion about that topic (e.g., how much v supports a certain politician).

Edges stand for discourse about topics in $\mathbb{R}^{n_e} = \mathcal{F}(e)$. There need not be a relation between bases of different stalks. However, each person v projects their opinion of the discussed topics on e via $\mathcal{F}_{v \subset e}$. There is *consensus* along $e = (u, v)$ if $\mathcal{F}_{v \subset e}\, x_v = \mathcal{F}_{u \subset e}\, x_u$.

In this framework, several models have been proposed to describe how opinions expressed by such a sheaf evolve over time Hansen & Ghrist (2020). The basic model assumes that everyone changes their opinion in order to minimize the difference to the average opinion of their friends, i.e.,

$$\frac{d}{dt} x_v(t) = \sum_{\substack{e \\ v \sim u}} \mathcal{F}_{v \subset e}^\top (\mathcal{F}_{u \subset e}\, x_u - \mathcal{F}_{v \subset e}\, x_v)$$

Here $v \overset{e}{\sim} u$ denotes that v and u share a common edge e. Combining all opinions to one vector x, this can be written as $\frac{d}{dt}x(t) = -L_{\mathcal{F}}\, x$. Solutions converge exponentially to consensus on all edges (Hansen & Ghrist (2020), Theorem 4.1).

The more realistic *bounded confidence model* assumes that the influence of a friend's opinion on v decreases if their opinions differ too much. The decay is expressed by a monotonically decreasing "bump" function $\phi : [0, \infty) \to [0, 1]$ that vanishes precisely whenever some threshold D is surpassed, as shown in Figure 6. The modified dynamics are described by

$$\frac{d}{dt}x_v(t) = \sum_{\substack{e \\ v \sim u}} \phi\left(\|\mathcal{F}_{u \subset e}\, x_u - \mathcal{F}_{v \subset e}\, x_v\|\right) \mathcal{F}_{v \subset e}^{\top}(\mathcal{F}_{u \subset e}\, x_u - \mathcal{F}_{v \subset e}\, x_v). \tag{1}$$

Configurations are stable if on each edge the difference is zero or at least D. In such a stable configuration, disregarding all edges without consensus and considering the connected components of the remaining graph provides a partition of the vertex set V (persons in the social network).

3 Methods and Experimental Design

In this section, we outline our proposed sheaf-theoretic algorithms used for community detection. We propose two algorithms that generate partitions of a given graph at random and one deterministic algorithm and evaluate their ability to detect communities. For the sake of comparability, we consider partitions of the full vertex set of a fixed graph (see Figure 3 for an example of such a partition) and evaluate how well they represent the community structure of the graph. The common evaluation measure that we will use in this work is *modularity*, which is computed in an assessment step to determine whether the division into communities is "relatively good."
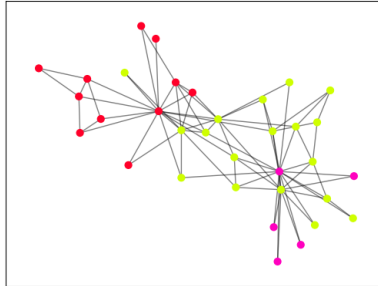

Figure 3: "Karate club" graph partitioned into three communities.

Definition 4. Let $G = (V, E)$ be a graph and V be subdivided into N subsets $\{V_c \mid c = 1, \ldots, N\}$, with E_c being the set of edges between nodes from V_c. Then the modularity Q is given by

$$Q := \sum_{c=1}^{N} \left[\frac{|E_c|}{|E|} - \left(\frac{\sum_{v \in V_c} \deg(v)}{2|E|} \right)^2 \right] \in [-1, 1]. \tag{2}$$

Intuitively, the quantity Q captures how many more edges than the average are within the subsets V_c, so high modularity indicates a good partitioning.

3.1 Detecting Communities with Constant Sheaves

Our first algorithm considers constant sheaves and models dynamics after the sheaf-theoretic bounded confidence model previously described in Subsection 2.2.

In Step 2 of Algorithm 1, we have included a stopping criterion for time efficiency: If the evolution does not converge after 1000 time units, the calculation is aborted. In Subsection 3.2 we explain why this is

Algorithm 1: Community Detection with Constant Sheaves $\mathbb{R}^n$

Input: dimension parameter n, parameter for diameter d

1 Initialize by picking a random opinion vector for each vertex stalk uniformly from $B(0, \frac{d}{2})$

2 Evolve the system using Eq. (1) until for no edge $e = (u, v)$, the difference $\|x_u - x_v\| \in (0.0033, 1)$; abort if that does not happen within 1000 time units

3 Obtain primary partition by grouping neighbors w, w' together if and only if $\|x_w - x_{w'}\| \le 0.0033$

4 if *we find a single-vertex community v* **then**

5 $\quad$ add the vertex to that adjacent community C, which has maximal
$\quad\quad 2|E|\, k_C - \deg(v) \cdot \sum_{w \in C} \deg(w)$, where k_C is the number of neighbors of v belonging to C

Output: The obtained partition

---

necessary. We keep track of the number of abortions and take them into account in our reported results below, when providing uncertainties.

We impose the positive value of 0.0033 for the computed difference in Steps 2 and 3 as another stopping criterion and trade-off between precision and time-efficiency.

Step 4 resolves single-vertex clusters by adding the vertex v to the cluster of that neighbor of v which is optimal in the sense of modularity. Proposition 5 justifies this choice by showing that each of these moves increases modularity.
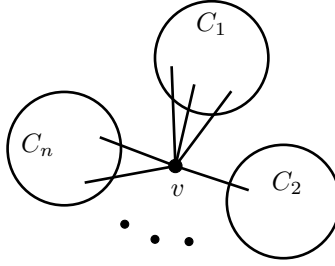

Figure 4: Sketch of a possible situation in the proof of Proposition 5 with $k_1 = 3$, $k_2 = 1$, $k_n = 2$

Proposition 5. *Single vertex clusters can be removed in a way that increases modularity. The removal of single vertex clusters implemented in Algorithm 1 is optimal in the sense of modularity.*

Proof. Consider a vertex v, the clusters $C_1, \ldots, C_n$ to which its neighbors belong, and let k_i be the number of neighbors of v that are in C_i, see Figure 4. Adding v to C_i increases the modularity of the given partition by

$$\Delta_i := \frac{k_i}{|E|} - \frac{2 \deg(v) \cdot \sum_{w \in C_i} \deg(w)}{4\,|E|^2}.$$

Notice that the value that we maximize in Step 5 is $2|E|\,\Delta_i$ (for the cluster C_i considered in Step 5). We must show that at least for one i, Δ_i is positive. Observe that

$$4|E|^2 \sum_{i=1}^n \Delta_i = \sum_{i=1}^n 4|E| k_i - 2 \deg(v) \sum_{i=1}^n \sum_{w \in C_i} \deg(w)$$

$$> 4|E| \deg(v) - 2 \deg(v) \cdot 2|E| > 0,$$

because $\sum_{i=1}^n \sum_{w \in C_i} \deg(w)$ is bounded from above by the sum of all degrees of vertices that are not v which is $2|E| - \deg(v)$. This completes the proof. $\qquad\square$

3.2 Convergence of Algorithm 1: Community Detection with Constant Sheaves

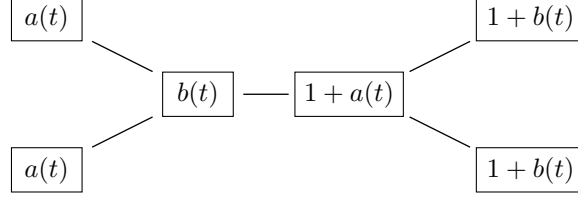The following example shows why we need the abortion criterion in Step 2 of Algorithm 1:

Figure 5: Example of a network in which Algorithm 1 is aborted in Step 2

Example 6. Consider the graph shown in Figure 5 and the constant sheaf $\mathbb{R}$ on that graph. Let

$$\phi(x) = \begin{cases} 1 - x & 0 \le x \le 1 \\ 0 & x \ge 1 \end{cases}.$$

Initialize with values a_0 and b_0 such that $1 + a_0 > b_0 > a_0$, making sure that none of the connections drawn in Figure 5 is ignored. Applying Eq. (1), we check that all vertices with value $a(t)$ and $1 + a(t)$ as well as all with $b(t)$ and $1 + b(t)$ follow the same derivative. Therefore, the opinions in the network will remain in the pattern shown in Figure 5 and the values $a(t)$ and $b(t)$ evolve according to

$$\frac{da(t)}{dt} = (b(t) - a(t))(1 + a(t) - b(t)) = -\frac{db(t)}{dt}.$$

That yields

$$a(t) = \frac{1}{2}\left(a_0 + b_0 - \frac{1}{e^{2t+c}+1}\right)$$

$$b(t) = \frac{1}{2}\left(a_0 + b_0 + \frac{1}{e^{2t+c}+1}\right)$$

with $c = \ln\left(\frac{1}{b_0 - a_0} - 1\right)$. In particular, $1 + a(t) - b(t) = 1 - \frac{1}{e^{2t+c}+1} < 1$, but it converges to one. Thus, the difference over the edge in the middle of Figure 5, between $b(t)$ and $1 + a(t)$ will always end up in the interval $(0.0033, 1)$ and Algorithm 1 will be aborted in Step 2.

Even though there are configurations without convergence, in our experiments we observed a convergence within 1000 time steps for more than 95% of starting configurations for any choice of parameters.

3.3 Detecting Communities with a Non-Constant Sheaf

We now consider a second community detection algorithm that uses the non-constant sheaf defined by setting $\mathcal{F}(v) = \mathbb{R}^{\deg(v)}$ (we think of it as one copy of $\mathbb{R}$ for every edge that uses v), $\mathcal{F}(e) = \mathbb{R}$, and restriction maps being the projections onto the corresponding edges.

At the vertices, there is no interaction between the directions corresponding to different edges, so each direction can be treated separately. If for an edge $e = (v, w)$, the difference between the initial components $x_{v,e}$ and $x_{w,e}$ is less than D, edge e will "survive" the evolution and otherwise not. Therefore, picking initial values at v uniformly from $[-\frac{d}{2}, \frac{d}{2}]^{\deg(v)}$ amounts to keeping each edge with the same probability p (which depends only on d) and ignoring it otherwise. We thus obtain Algorithm 2.

Algorithm 2: Community Detection with a Non-Constant Sheaf

    Input: probability parameter p
1 Obtain primary partition by grouping neighbors w, w' together with probability p
2 Remove single-vertex clusters as in Algorithm 1, Step 4
    Output: The obtained partition

3.4 Deterministic Sheaf Community Detection

A more general version of the bounded confidence model allows different functions ϕ_e for different edges. In the case of the non-constant sheaf, this means that the probability p can depend on the edge and the local structure of the graph.

With the idea in mind that we want to retain edges if the vertices belong to the same community, it should be more likely for $e = (u, v)$ to be retained if the number of common neighbors of u and v, denoted $N_{u,v}$, is not too small compared to the total number of neighbors of u and v. In an extreme case, this likelihood is either zero or one, and we arrive at Algorithm 3.

Algorithm 3: Deterministic Community Detection

Input: two parameters $a \in [0, 1]$ and $b \in \mathbb{R}$
1 Obtain primary partition by grouping the two ends u, v of an edge e together if
  $a \cdot (\deg(u) + \deg(v)) < b + N_{u,v}$
2 Remove single-vertex clusters as in Algorithm 1, Step 4
Output: The obtained partition

3.5 Experimental Setup

We tested the performance of Algorithm 1 by considering its dependence on d for $n = 1$ and the different bump functions ϕ shown in Figure 6, as well as for $n \in \{2, 3, 5, 10\}$ and ϕ_1. Note that for sake of comparability we fixed $D = 1$ to be the threshold for ignoring. On the interval $[0, 1)$, the ϕ_i are given by:

$$\phi_1(x) = 1 - x$$
$$\phi_2(x) = 1 - x^2$$
$$\phi_3(x) = (1 - x)^2$$
$$\phi_4(x) = 1 - x - \sin(2\pi x)/7$$

Algorithm 2 was run for various values of p and Algorithm 3 with different values of a and b.

All tests were performed on Zachary's karate club graph G, shown in Figure 3, which represents a social network of a karate club studied by sociologist Wayne Zachary from 1970 to 1972 (Zachary, 1977). The network captures 34 members of the karate club and includes links between pairs of members who interacted outside the club. The maximal possible modularity of a partition of G for this graph is $Q_{\max}(G) \approx 0.42$ Costa (2014). All experiments were implemented in Python and the code is freely and publicly available at https://github.com/ArneWolf/Bd_confidence_communities.
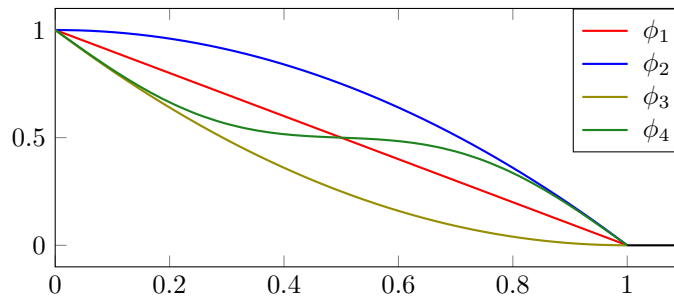

Figure 6: The different functions ϕ

4 Experimental Results

We report the results of our numerical experiments for our three proposed sheaf-theoretic community detection algorithms from Section 3. We compare the performance in terms of average number of clusters and

modularity (Definition 4).

Algorithm 1: The Constant Sheaves. To justify our stopping criterion of 0.0033 in Step 2 of Algorithm 1 and estimate the error caused, we compared the partitions to those obtained for running the evolution longer so that for every edge $\|x_u - x_v\| \notin (0.001, 1)$ instead. The more precise calculations took, on average, a factor of 100 longer. Out of 2930 runs we performed, the first stopping criterion applied 175 times and for the remaining 2755 runs, the two obtained partitions agreed. We thus concluded that 0.0033 is small enough to not cause a significant error.

Each parameter combination was simulated $N = 1000$ times. Figure 8 shows the average values of X (number of clusters, modularity) and an error bar with radius

$$X_{\text{error}} = \sigma_X + \frac{\#\,\text{early stoppings}}{N}\,\bar{X},$$

where $\bar{X}$ denotes the average and σ the standard deviation.

We immediately see that a larger width d of the distribution from which the initial opinions are picked results in more clusters. Neither the number of clusters nor the modularity depends significantly on the choice of ϕ.
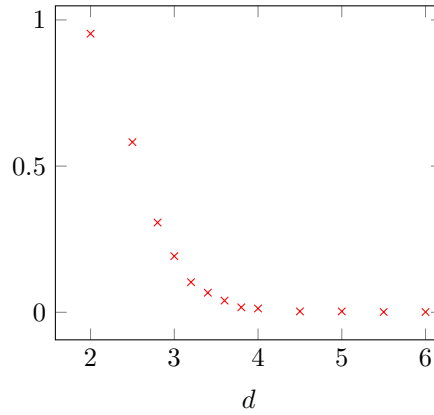

Figure 7: Likelihood of most likely partition for different diameters d of the initial distribution

In the case of smaller d, however, ϕ_3 tends to create more clusters than the others. Intuitively, this behavior can be explained by comparing ϕ_3 to linearly decaying bump functions: it is more similar to a bump function that decays quicker than ϕ_1, than to ϕ_1. Quicker decay means a smaller threshold $D < 1$. That in turn corresponds to having $D = 1$ and a bigger d. Thus, the properties of ϕ_3 tend to make d appear larger.

For large d, the influence of removing singleton clusters (compared to the primary partition) grows. This explains why the differences in the number of clusters as well as modularity for different choices of ϕ and d become smaller. Looking at $p = 0$ in Figure 10 shows that resolving singleton clusters from a collection of only singleton clusters yields a modularity of about 0.191. This is one explanation for having higher modularites for larger d. Another explanation is that small d are likely to result in one big cluster containing all the points. This trivial partition has modularity 0. Figure 7 shows the likelihood of the most likely partition for ϕ_1 and $n = 1$. For $d \leq 4$, whenever there is a partition that is significantly the most likely one, this is the trivial partition. The same qualitative behavior is observed for other ϕ and n.

If d is large enough, the number of clusters increases as n increases. This is significant for $n = 10$, but the tendency can be observed in Figure 9 for smaller n as well. This can be explained by larger n corresponding to more topics of discussion, which in turn increases the chances that persons can disagree. The modularity for small d seems to depend on n non-monotonously: $n = 10$ gives the highest values, followed by $n = 1$ and $n = 2$. If d is large, the differences become insignificant.

Algorithm 2: The Non-Constant Sheaf. For Algorithm 2, Figure 10 shows the expected decrease of the number of clusters with increasing p. The modularity has a maximum of about 0.26 near $p = 0.12$. For
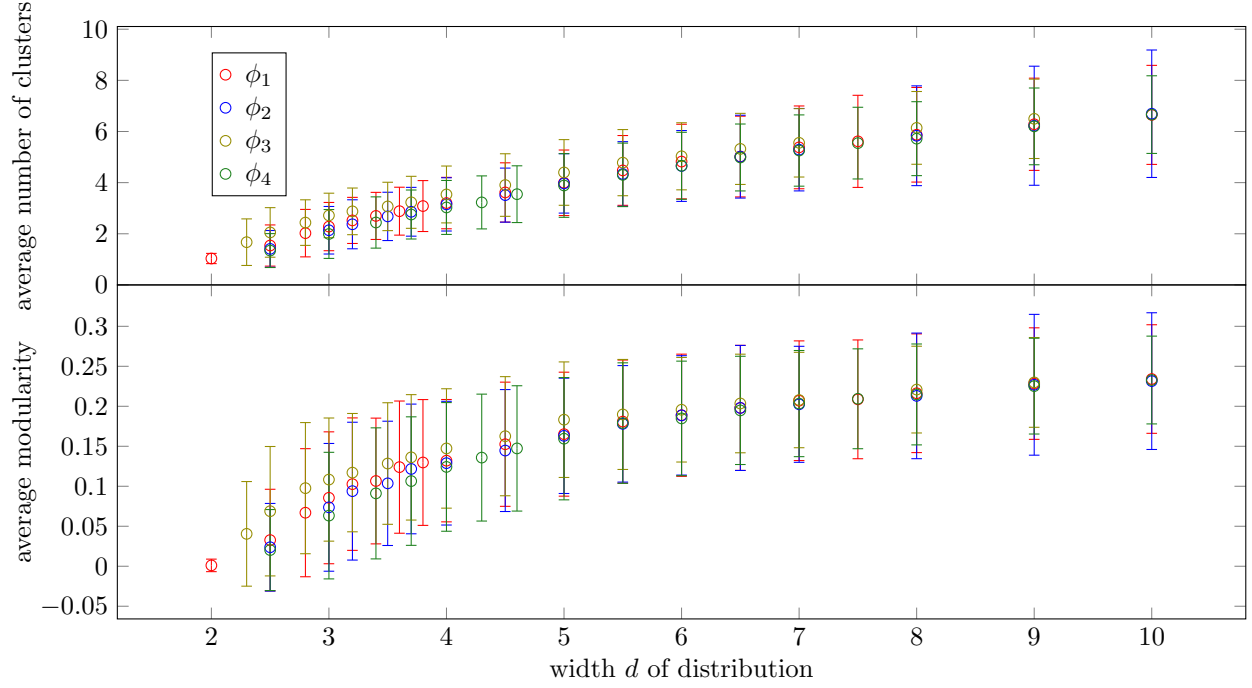
Figure 8: Average number of clusters and modularity for $n = 1$, different bump functions ϕ and different diameters of the initial opinion distributions

smaller $p \to 0$, the modularity becomes 0.191 and for large p it goes to zero due to the dominance of the trivial partition.

Algorithm 3: The Deterministic Algorithm. Due to the affine linear condition in Step 1 of Algorithm 3, the modularity and number of clusters change along affine lines. The deterministic algorithm reaches a maximal modularity of about 0.407, which comes close to $Q_{\max} = 0.42$. However, the portion of the parameter space that reaches this value is small—the other algorithms had broader ranges of maxima for their modularities. The partition yielding $Q = 0.407$ consists of four clusters, just as the partition obtaining $Q_{\max}$.

5 Discussion

In this work, we showed that sheaves are a viable algebraic topological tool to model the problem of community detection on networks. We proposed three different algorithms, two of which had random initializations and were based on constant sheaves and a non-constant sheaf, and a third deterministic version that allows for different bump functions for different edges of the graph. The deterministic sheaf-theoretic community detection algorithm, in particular, performed well in terms of modularity and achieved values near the maximal modularity value. Ours is the first work to computationally implement sheaves on real-world social network data and the first use of sheaves in the problem of community detection on networks. Moreover, our work provides a proof-of-concept for future work adapting the potential of cellular sheaves to studying complex systems captured by networks and simplicial complexes, in general.

Directions for future work involve combining various notions from the different algorithms to, for instance, allow bump functions in Algorithm 1 or the probability p in Algorithm 2 to depend on the edge, which may potentially improve performance. An optimization procedure may also be proposed to find optimal parameters, and thus, optimal partitions for a network.
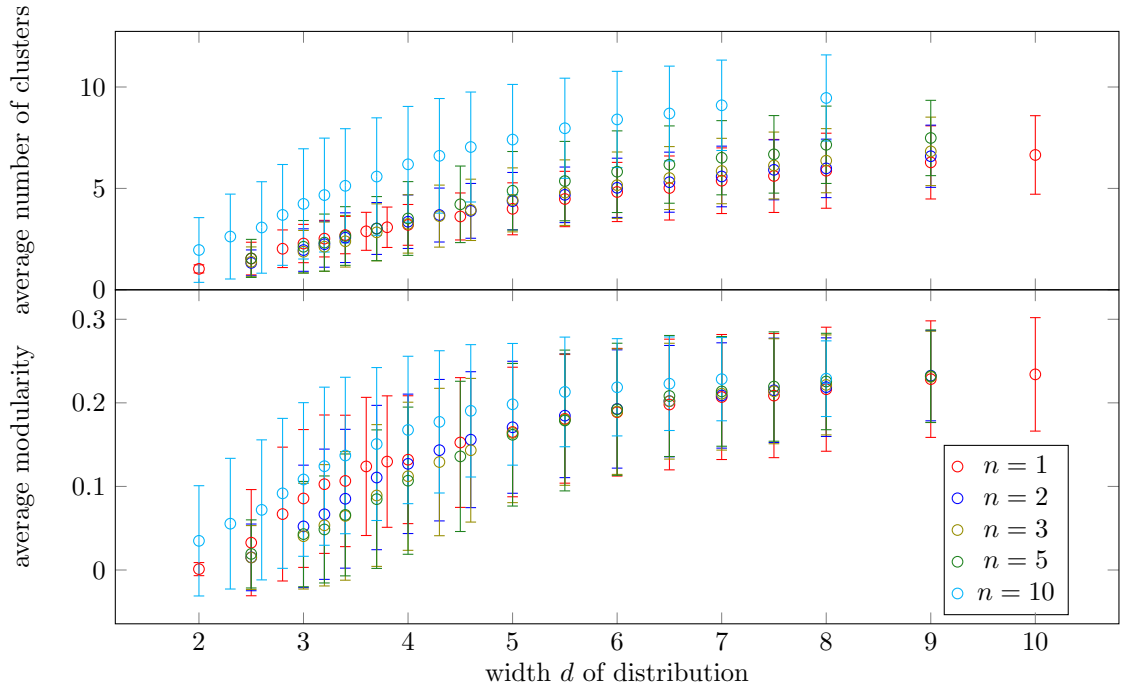
Figure 9: Average number of clusters and modularity for different dimensions n, bump function ϕ_1 and different diameters of the initial opinion distributions
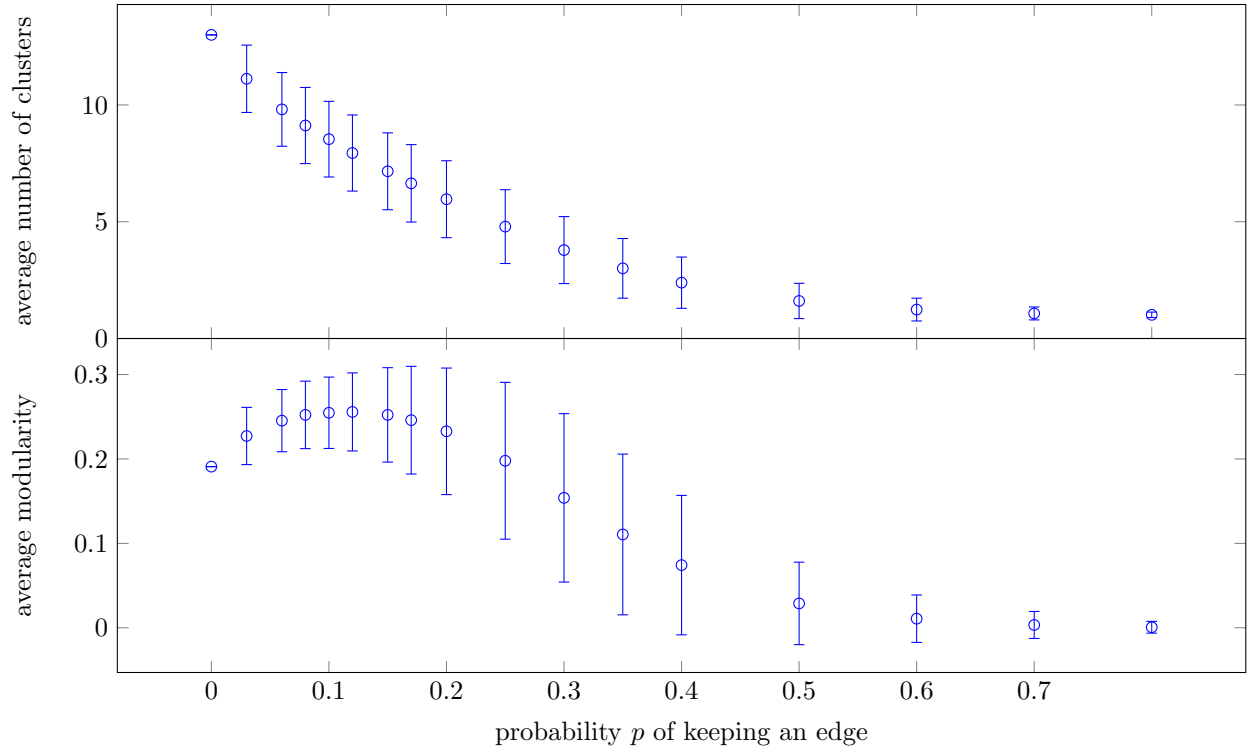

Figure 10: Average number of clusters and modularity for the non-constant sheaf algorithm for different probabilities p of keeping an edge
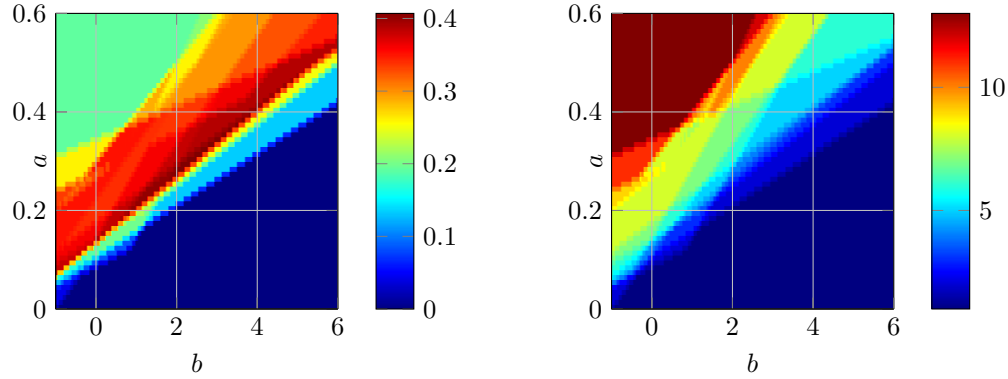
Figure 11: Modularities (left) and numbers of clusters (right) for several parameter combinations for the deterministic Algorithm 3

Acknowledgments

References

COSTA, ALBERTO. 2014. Some remarks on modularity density. *arXiv preprint arXiv:1409.4063*.

CURRY, JUSTIN MICHAEL. 2014. *Sheaves, cosheaves and applications*. University of Pennsylvania.

HANSEN, JAKOB, & GHRIST, ROBERT. 2020. *Opinion Dynamics on Discourse Sheaves*.

HEGSELMANN, RAINER, & KRAUSE, ULRICH. 2019. Consensus and Fragmentation of Opinions with a Focus on Bounded Confidence. *The American Mathematical Monthly*, **126**(8), 700–716.

RIESS, HANS, & GHRIST, ROBERT. 2022. *Diffusion of Information on Networked Lattices by Gossip*.

SCHINDLER, DOMINIK J, & BARAHONA, MAURICIO. 2023. Persistent Homology of the Multiscale Clustering Filtration. *arXiv preprint arXiv:2305.04281*.

SHEPARD, ALLEN DUDLEY. 1985. *A cellular description of the derived category of a stratified space*. Brown University.

ZACHARY, WAYNE W. 1977. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, **33**(4), 452–473.