Approximating Simplet Frequency Distribution for Simplicial Complexes*

Hamid Beigy[1], Mohammad Mahini[2], Salman Qadami[3], and Morteza Saghafian[4]

1 **Sharif University of Technology**
beigy@sharif.edu
2 **Sharif University of Technology**
m_mahini@ce.sharif.edu
3 **Amirkabir University of Technology**
salmanqadami@gmail.com
4 **Institute of Science and Technology Austria**
morteza.saghafian@ist.ac.at

Abstract

Simplets, constituting elementary units within simplicial complexes (SCs), serve as foundational elements for the structural analysis of SCs. Previous efforts have focused on the exact count or approximation of simplet count rather than their frequencies, with the latter being more practical in large-scale SCs. This paper enables simplet frequency analysis of SCs by introducing the Simplet Frequency Distribution (SFD) vector. In addition, we present a bound on the sample complexity required for accurately approximating the SFD vector by any uniform sampling-based algorithm. We also present a simple algorithm for this purpose and justify the theoretical bounds with experiments on some random simplicial complexes.

1 Introduction

In a range of disciplines, including biology, geology, and social science, the application of simplicial complexes is frequently employed to extract essential structural insights. *Simplicial Complexes (SCs)* are defined as networks of higher-order that possess the property of downward closure, which makes them suitable for representing higher-order relationships within network-like structures and their geometrical aspects [6, 11, 13]. In particular, SCs are used to study the geometric and combinatorial structure of protein interaction networks [12], epidemic spreading [17], co-authorship relations [26], analyze email communications [15], and investigate the functional and structural organization of the brain [18].

Analyzing network behavior using small network building blocks, commonly known as *motifs*, is common in numerous fields, including biological [1] and social networks [25]. Graphs are great examples where researchers use small building blocks called *graphlets* to understand how networks behave based on local structures [23]. Graphlet analysis has many applications in biological networks [10, 30], and social networks [2, 3]. By considering *simplets* as fundamental elements within simplicial complexes, analogous to graphlets in the context of SCs, we can examine the specific patterns formed by the simplices associated with different

* Work by the second and fourth authors is partially supported by the European Research Council (ERC), grant no. 788183, by the Wittgenstein Prize, Austrian Science Fund (FWF), grant no. Z 342-N31, and by the DFG Collaborative Research Center TRR 109, Austrian Science Fund (FWF), grant no. I 02979-N35.

40th European Workshop on Computational Geometry, Ioannina, Greece, March 13–15, 2024.
This is an extended abstract of a presentation given at EuroCG'24. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

sets of nodes [22]. This approach offers a straightforward way of analyzing complex networks' structural characteristics and their constituent parts.

Approximating Graphlet Count and Distribution. Numerous investigations have delved into the precise enumeration of graphlet types or approximating their frequencies. Several studies, like the ESU and RAGE algorithms, count the precise number of graphlets [29, 20]. Meanwhile, various algorithms like GRAFT, CC have employed sampling techniques to estimate the frequency of graphlets [5, 7, 8, 9]. For instance, Bressan in [7] introduced a random walk based method that preprocesses k-vertex graphlets, and gives a random graphlet in the time complexity of $k^{O(k)} \cdot \log \Delta$, where Δ is the maximum degree in the given graph.

Approximating Simplet Count and Distribution. Preti et al. introduced the concept of simplets, and the FRESCO algorithm that indirectly estimates the quantity of each simplet by utilizing a proxy metric referred to as *support* [21, 22]. B-Exact precisely enumerates up to 4-node configurations through combinatorial techniques [4]. Importantly, each simplet can correspond to zero, one, or more than one configuration. Kim et al. presented SC3, a sampling-based algorithm for approximating simplet count, that utilizes color coding techniques [14]. Thus far, a limited number of dedicated algorithms designed for counting simplets either precisely or approximately. The concept of simplets is relatively novel, and numerous opportunities remain untapped for their application in various contexts.

Contribution. Calculating the exact quantity of each graphlet type or simplet type is frequently prohibitively expensive, and for numerous practical purposes, obtaining an estimated count of various graphlet types and simplet types or approximating their frequency distribution is usually adequate. This paper studies the concept of the *Simplet Frequency Distribution (SFD)* for the first time (to the best of the authors' knowledge), which can be more practical in analyze of large-scale SCs. Alongside this new concept, we present an algorithm to approximating the SFD vector based on uniform sampling of simplets.

More importantly, we present an upper-bound on the sample complexity (number of samples needed) of any approximation algorithm based on a uniform sampling method. By doing this, we aim to enhance our comprehension and analysis of simplicial complexes, mapping them to vector spaces and using this vector for machine learning applications such as classification. In overview, we present the following contributions.

- Defining the concept of the Simplet Frequency Distribution (SFD) vector
- Studying an upper bound on the number of samples we need for every sampling based algorithm for approximating the SFD vector
- Proposing an algorithm for approximating the SFD vector by uniform sampling of simplets

2 Preliminaries

Within this section, we lay out the foundational concepts employed in this paper.

Simplex. A n-simplex is the convex hull of $n+1$ distinct points in n-dimensional space. A face of an n-simplex σ is the convex hull of any non-empty subset S of its vertices.

Simplicial Complex. A simplicial complex $\mathcal{K}$ is a set of simplices that is closed under taking faces, and the non-empty intersection of any two simplices $\sigma, \tau \in \mathcal{K}$ is a face of both σ, τ.

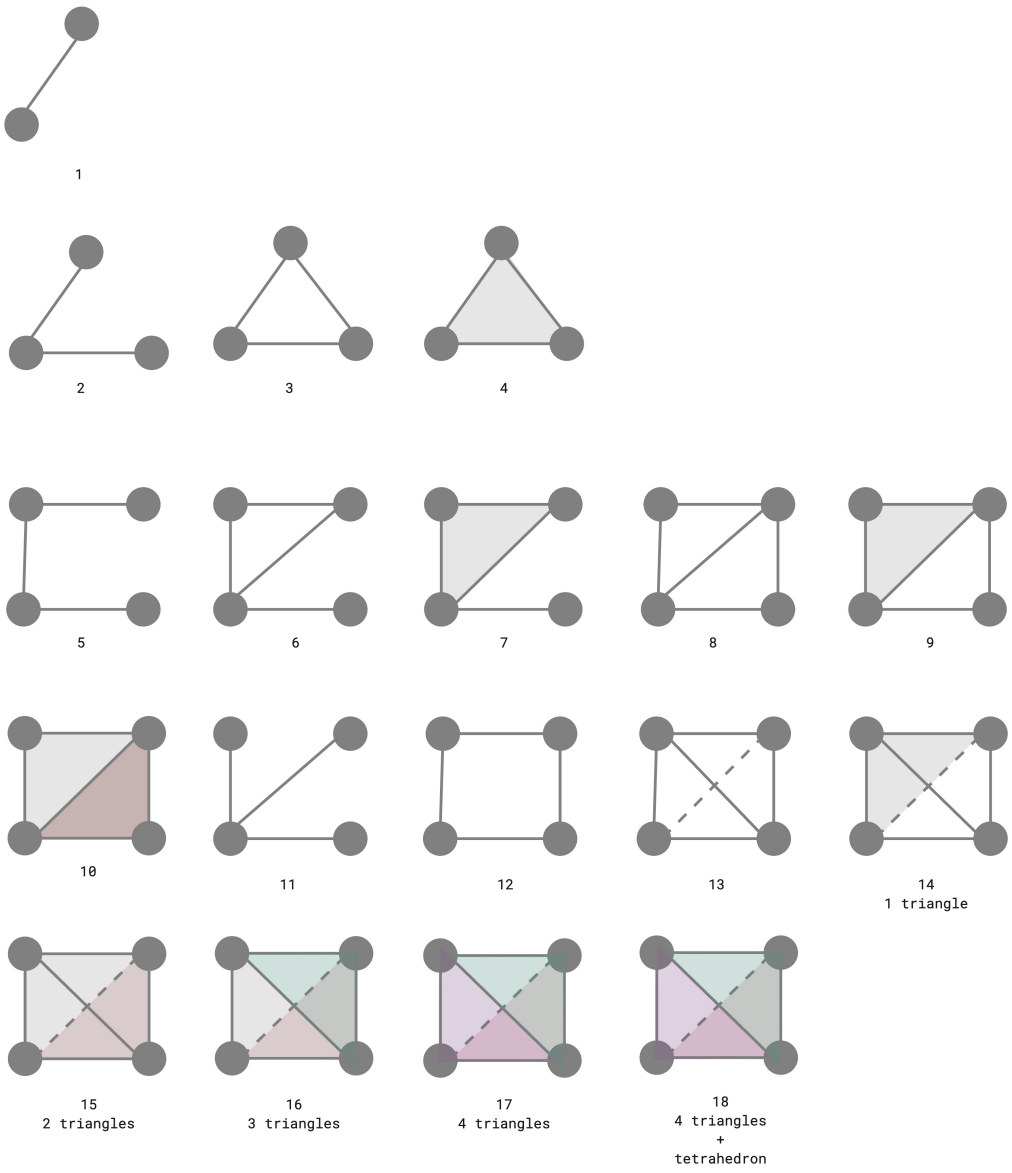


Figure 1 The set of all 18 simplet types with at least two and at most four vertices.

Simplicial complexes provide a combinatorial and topological framework for studying the structure of spaces through simplices, capturing both geometric and connectivity information.

Simplet. *Simplets* are small induced connected sub-complexes of a massive complex that appear at any frequency. A complex H is an induced sub-complex of $\mathcal{K}$ if and only if, for any simplex S in $\mathcal{K}$ whose vertices are a subset of $V(H)$, S should also be in H. So, every simplet can be identified by its vertices, typically regarded as being at least two. A *simplet set* is a set of simplets of a simplicial complex. *Simplet types* are isomorphic classes of simplets. We denote $\mathcal{S}_{\mathcal{K}}(i)$ as a set of all simplets of type i in $\mathcal{K}$, where $1 \leq i \leq N_m$, and N_m is the number of simplet types with at most m vertices. Also, we denote $\mathcal{S}_{\mathcal{K}}^m$ as the set of all simplets in $\mathcal{K}$ with at most m vertices. We assume that m is a constant small number.

Simplet Frequency Distribution. The SFD vector of complex $\mathcal{K}$ characterizes the relative frequencies of various simplets in $\mathcal{K}$. By definition, $|\mathcal{S}_{\mathcal{K}}(i)|$ is the number of simplets of type i in $\mathcal{K}$, where $i \in \{1, \ldots, N_m\}$. The frequency, denoted by $\phi_{\mathcal{K}}(i)$, is obtained by dividing $|\mathcal{S}_{\mathcal{K}}(i)|$ by $\sum_{j=1}^{N_m} |\mathcal{S}_{\mathcal{K}}(j)|$. The vector $(\phi_{\mathcal{K}}(1), \ldots, \phi_{\mathcal{K}}(N_m))$ is called the SFD vector of the $\mathcal{K}$. In Figure 2, we show an SFD vector for two sample SCs.

3 Approximating the SFD Vector

In this section, we focus on showing that if we have a method for sampling simplets uniformly from an SC, we can have an (ϵ, δ)-approximation of the SFD vector. After that, we study an algorithm for simple uniform sampling that is better than a trivial brute-force sampling method. Consider a collection of independent samples $X^k = X_1, \ldots, X_k$ drawn from a distribution ϕ over a domain D. Here, $\phi(A)$ signifies the probability of selecting an element from the set $A \subseteq D$. The empirical estimation of $\phi(A)$ based on the samples X^k is:

$$\hat{\phi}^X(A) = \frac{1}{k} \sum_{j=1}^{k} 1_A(X_j),$$

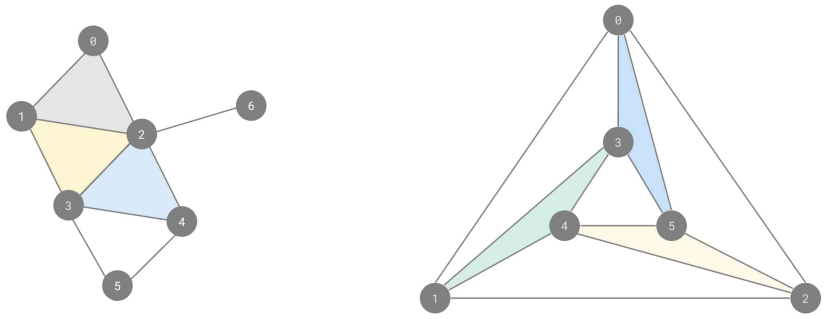
In this equation, $1_A(X_j)$ is an indicator function that equals 1 when X_j belongs to A and equals 0 otherwise. Additionally, let $\mathcal{R}$ be a family of subsets of D.

(ϵ, δ)-approximation. For any given $\epsilon, \delta \in (0, 1)$, we say $X \subseteq D$ is an (ϵ, δ)-*approximation* of $(\mathcal{R}, \phi)$, if with a probability of at least $(1 - \delta)$, it satisfies $sup_{A \in \mathcal{R}} |\phi(A) - \hat{\phi}^X(A)| \leq \epsilon$.

3.1 Sample Complexity of Approximating the SFD Vector

We utilize the concept of Vapnik-Chervonenkis dimension (VC dimension), introduced in [27]. In short, for a domain D and a collection $\mathcal{R}$ of subsets of D, the VC dimension $VC(D, \mathcal{R})$, represents the maximum size of a set $X \subseteq D$ that can be shattered by $\mathcal{R}$, which means $\{r \cap X | \forall r \in R\} = 2^{|X|}$. We use VC dimension to determine the sample complexity for approximating the SFD vector through simplet sampling models. Theorem 3.1 establishes the VC dimension of the collection of simplet sets.

▶ **Theorem 3.1** (VC Dimension of Simplets). *Let $\mathcal{R} = \{\mathcal{S}_i \mid 1 \leq i \leq N_m\}$ be a family of all simplet sets where N_m is the number of simplet types with at most m vertices, and $D = \mathcal{S}_{\mathcal{K}}^m$. Then, we have $VC(D, \mathcal{R}) = 1$.*



Simplet Type	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\lvert\mathcal{S}_{\mathcal{K}_1}\rvert$	**10**	**12**	**1**	**3**	**7**	**1**	**6**	0	**1**	**2**	**3**	0	0	0	0	0	0	0
$\lvert\phi_{\mathcal{K}_1}\rvert$	**0.22**	**0.26**	**0.02**	**0.07**	**0.15**	**0.02**	**0.13**	0	**0.02**	**0.04**	**0.07**	0	0	0	0	0	0	0
$\lvert\mathcal{S}_{\mathcal{K}_2}\rvert$	**12**	**12**	**5**	**3**	0	0	0	**3**	**9**	0	0	**3**	0	0	0	0	0	0
$\lvert\phi_{\mathcal{K}_2}\rvert$	**0.26**	**0.26**	**0.11**	**0.06**	0	0	0	**0.06**	**0.19**	0	0	**0.06**	0	0	0	0	0	0

Figure 2 The SFD vector and the number of at most 4-vertices simplets for two sample SCs. The simplet types in the table refer to the types in Figure 1.

Proof. We show that a set X with $|X| > 1$ can not be shattered with $(D, \mathcal{R})$. Let X be a set of simplets shattered with $(D, \mathcal{R})$, and assume that $|X| > 1$. Let s_1 and s_2 be two distinct elements of X. There are two possibilities. If elements s_1 and s_2 belong to the same simplet type, then, set $\{s_1\}$ can not be shattered because there is no set $\mathcal{S}_i$, including only s_1. Otherwise, elements s_1 and s_2 belong to different simplet types, but then $\{s_1, s_2\}$ can not be shattered because no set $\mathcal{S}_i$ contains both. Clearly every singleton set can be shattered by one of the $\mathcal{S}_i$s, hence $VC(D, \mathcal{R}) = 1$. ◀

The subsequent theorem from [24] illustrates the relationship between the upper bound on the sample complexity of sampling-based (ϵ, δ)-approximations and VC dimension.

▶ **Theorem 3.2.** *Let D be a domain and $\mathcal{R}$ be a family of subsets of D, with $VC(D, \mathcal{R}) \leq d$ and ϕ be a distribution on D. For every $\epsilon, \delta \in (0, 1)$, every set X of independent samples drawn from D using ϕ that satisfies*

$$|X| \geq \frac{c}{\epsilon^2}\left(d + \ln\frac{1}{\delta}\right),$$

is an (ϵ, δ)-approximation of $(\mathcal{R}, \phi)$ for some positive constant c.

Combining Theorem 3.1 and Theorem 3.2 we conclude our main result.

▶ **Proposition 3.3.** *Let X be a set of at least $\frac{c}{\epsilon^2}(1 + \ln\frac{1}{\delta})$ simplets sampled uniformly from simplicial complex $\mathcal{K}$. Then, X obtains an (ϵ, δ)-approximation on the SFD vector of $\mathcal{K}$.*

Proposition 3.3 shows that we can approximate the SFD vector using sampling-based algorithms, and the sample complexity of these approximations are independent of the simplicial complex size. This property suggests the usage of approximation algorithms for various simplicial complex sizes with the same sample complexity.

3.2 Simplet Uniform Sampling Algorithm

In this section, we propose a uniform sampling algorithm for simplets in a connected simplicial complex $\mathcal{K}$ that is better than a trivial brute-force method. The algorithm we present is a Monte-Carlo Markov-Chain algorithm [28], that samples sufficiently many simplets uniformly at random. We assume that $\mathcal{K}$ is connected with at least three vertices, and $m \geq 3$.

For the sampling part, we perform a random walk on a directed graph $\mathcal{P}^m_{\mathcal{K}}$ whose vertex set (states) is a set of all simplets in complex $\mathcal{K}$ with at most m vertices. Out-neighbors of every state s can be created by adding one vertex to s, removing one vertex from s, or replacing one vertex in s with another vertex out of s.

The transition probability matrix T for the random walk is such that every cell $T(i,j)$ defines the transition probability from state i to j. If i and j are not neighbors, we set $T(i,j) = 0$. Otherwise, we set $T(i,j) = min(\frac{1}{d(i)}, \frac{1}{d(j)})$ where $d(i)$ specifies the number of out-neighbors of state i. Also, for every i, if the sum of transitions from i is not equal to 1, we allocate the remaining probability to a self-loop for i. Observe that since $\mathcal{K}$ is finite, $\mathcal{P}^m_{\mathcal{K}}$ is finite and since $\mathcal{K}$ is connected, the random walk is irreducible. Indeed, since $\mathcal{K}$ is connected, there is a vertex u in $\mathcal{K}$ that is connected to at least two other vertices v, w. So the three simplets on $\{u,v,w\}$, on $\{u,v\}$, and on $\{u,w\}$ form a triangle in $\mathcal{P}^m_{\mathcal{K}}$ with positive probabilities on the edges, this means the random walk is aperiodic. Also T is symmetric, meaning $T = T^T$. This ensures that the random walk on $\mathcal{P}^m_{\mathcal{K}}$ converges to the uniform stationary distribution $(\frac{1}{|\mathcal{S}^m_{\mathcal{K}}|}, \ldots, \frac{1}{|\mathcal{S}^m_{\mathcal{K}}|})$. So, using this random walk on $\mathcal{P}^m_{\mathcal{K}}$, we can select a simplet from the input complex $\mathcal{K}$ with uniform distribution.

3.3 The SFD Vector Approximation Algorithm

Now, we propose the (ϵ, δ)-approximation algorithm on the SFD vector of $\mathcal{K}$. For input $\epsilon, \delta \in (0,1)$ and simplicial complex $\mathcal{K}$, first the algorithm calculates the number ℓ of samples needed, according to Proposition 3.3. After that it executes ℓ times the sampling algorithm, presented above, to find the set X of ℓ simplets that are chosen uniformly at random. Based on X, it computes $\hat{\phi}^X_{\mathcal{K}}(i)$, that is a (ϵ, δ)-approximation for $\phi_{\mathcal{K}}(i)$, for $1 \leq i \leq N_m$. The vector $(\hat{\phi}^X_{\mathcal{K}}(1), \ldots, \hat{\phi}^X_{\mathcal{K}}(N_m))$ is therefore a (ϵ, δ)-approximation for the SFD vector of $\mathcal{K}$.

Time Complexity of the SFD vector Approximation Algorithm The time complexity of the (ϵ, δ)-approximation algorithm, consists of two components: the number of samples and the time complexity for sample identification. Having established that $O(\frac{1}{\epsilon^2} \cdot (1 + \ln \frac{1}{\delta}))$ samples are sufficient for (ϵ, δ)-approximation, our focus shifts to analyzing the time complexity of the MCMC sampling algorithm. The mixing time t^G_{mix} in a random walk on graph G is the number of steps needed to be close to its stationary state with high probability. Lemma 3.4 limits the maximum degree of $\mathcal{P}^m_{\mathcal{K}}$, and then Lemma 3.5 shows an upper bound on $t^{\mathcal{P}^m_{\mathcal{K}}}_{mix}$ in terms of the number of vertices n, the maximum degree Δ in $\mathcal{K}$, and the diameter $diam(\mathcal{K})$, which is the length of maximum shortest path between any pair of vertices in $\mathcal{K}$.

▶ **Lemma 3.4.** *The maximum degree of $\mathcal{P}^m_{\mathcal{K}}$ satisfies $\Delta(\mathcal{P}^m_{\mathcal{K}}) \in O(m^2 \cdot \Delta)$.*

Proof. We can create neighbors of every state in $\mathcal{P}^m_{\mathcal{K}}$ by adding a new vertex, removing a vertex, or replacing two vertices. The number of neighbors by adding a new vertex is at most $m \cdot \Delta$, by removing a vertex is at most m, and by replacing two vertices is at most $m \cdot (m-1) \cdot \Delta$. Therefore, the maximum degree of every state in $\mathcal{P}^m_{\mathcal{K}}$ is in $O(m^2 \cdot \Delta)$. ◀

▶ **Lemma 3.5.** *The mixing time of the markov chain on $\mathcal{P}^m_{\mathcal{K}}$ is in $O(\log(n) \cdot \Delta \cdot diam(\mathcal{K})^2)$.*

Proof. Theorems (12.4) and (13.26) in [16] imply that the mixing time t^G_{mix} of a random walk on graph G with n vertices is in $O(\log(n)\cdot\Delta(G)\cdot diam(G)^2)$. For $\mathcal{P}^m_{\mathcal{K}}$ we can reach from every state i to every other state j with $diam(\mathcal{K})+m$ steps as follows: Assume $v\in V(i), u\in V(j)$ and assume a shortest path from v to u in $\mathcal{K}$. Starting from i, in every step, we replace one vertex from the current state with the unused closest vertex to v in the shortest path from v to u, until we reach u. After that we replace vertices that are not in j with vertices in j, starting from neighbors of u. We make sure that after each step the simplet remains connected. So, $diam(\mathcal{P}^m_{\mathcal{K}}) = diam(\mathcal{K})+m$ and therefore in the markov chain $\mathcal{P}^m_{\mathcal{K}}$ we have

$$t^{\mathcal{P}^m_{\mathcal{K}}}_{mix} \in O(log(n(\mathcal{P}^m_{\mathcal{K}}))\cdot\Delta(\mathcal{P}^m_{\mathcal{K}})\cdot diam(\mathcal{P}^m_{\mathcal{K}})^2) \in O(log(n)\cdot\Delta\cdot diam(\mathcal{K})^2).$$

◀

▶ **Corollary 3.6** (Time Complexity of (ϵ,δ)-approximation of SFD vector)**.** *Let $\mathcal{K}$ be a simplicial complex with the number of vertices n, maximum degree Δ and diameter $diam(\mathcal{K})$. The time complexity of (ϵ,δ)-approximation of SFD vector of $\mathcal{K}$ is $O(\frac{1}{\epsilon^2}\cdot(1+\ln\frac{1}{\delta})\cdot\log(n)\cdot\Delta\cdot diam(\mathcal{K})^2)$.*

In practice, for a large sparse simplicial complex $\mathcal{K}$, since Δ and $diam(\mathcal{K})$ are bounded, the above bound is sublinear in the size of $\mathcal{K}$ (i.e. the number of vertices or $\mathcal{K}$).

Implementation and Experiments. We implement an algorithm for counting the exact number of simplets of different types and another algorithm for approximating the frequencies based on uniform simplet sampling, with their source code accessible on GitHub [19]. This experimental outcome demonstrates that the confidence in the (ϵ,δ)-approximation is unrelated to the size of the input complex.

4 Conclusion

This paper introduced the Simplet Frequency Distribution (SFD) vector and a method for approximating it with simplet sampling algorithms. Also, we studied the sample complexity of approximating the SFD vector for SCs and showed that the obtained bounds are independent of SC size. We also showed that we can approximate the SFD vector with a specific error and confidence, and the time complexity depends only on the time complexity of sampling algorithm for finding a sample, that is sublinear in the algorithm we presented. It would be beneficial to have such algorithms with time complexity that is independent of the SC size.

Combining these approaches with filtrations of simplicial complexes and exploring them within alpha complexes would be interesting. Additionally, defining the vertex-specific SFD vectors for each vertex in the complex could offer valuable insights into their potential to convey more information about the global structure of the complex.

References

1 Uri Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007.

2 James R Ashford, Liam D Turner, Roger M Whitaker, Alun Preece, and Diane Felmlee. Understanding the characteristics of covid-19 misinformation communities through graphlet analysis. *Online Social Networks and Media*, 27:100178, 2022.

3 Andrew Baas, Frances Hung, Hao Sha, Mohammad Al Hasan, and George Mohler. Predicting virality on networks using local graphlet frequency distribution. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2475–2482. IEEE, 2018.

4 Austin R Benson, Rediet Abebe, Michael T Schaub, Ali Jadbabaie, and Jon Kleinberg. Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*, 115(48):E11221–E11230, 2018.

5 Mansurul A Bhuiyan, Mahmudur Rahman, Mahmuda Rahman, and Mohammad Al Hasan. Guise: Uniform sampling of graphlets for large graph analysis. In *2012 IEEE 12th International Conference on Data Mining*, pages 91–100. IEEE, 2012.

6 Ginestra Bianconi. *Higher-order networks*. Cambridge University Press, 2021.

7 Marco Bressan. Efficient and near-optimal algorithms for sampling small connected subgraphs. *ACM Transactions on Algorithms*, 19(3):1–40, 2023.

8 Marco Bressan, Flavio Chierichetti, Ravi Kumar, Stefano Leucci, and Alessandro Panconesi. Counting graphlets: Space vs time. In *Proceedings of ACM International Conference on Web Search and Data Mining*, pages 557–566, 2017.

9 Marco Bressan, Flavio Chierichetti, Ravi Kumar, Stefano Leucci, and Alessandro Panconesi. Motif counting beyond five nodes. *ACM Transactions on Knowledge Discovery from Data*, 12(4):1–25, 2018.

10 Alexander Douglas et al. *Exploring how graphlet analysis can be used to identify highly-connected cancer driving genes*. PhD thesis, University of Northern British Columbia, 2022.

11 Herbert Edelsbrunner. *A short course in computational geometry and topology*. Springer, 2014.

12 Ernesto Estrada and Grant J Ross. Centralities in simplicial complexes. applications to protein interaction networks. *Journal of theoretical biology*, 438:46–60, 2018.

13 Jakob Jonsson. *Simplicial complexes of graphs*, volume 3. Springer, 2008.

14 Hyunju Kim, Jihoon Ko, Fanchen Bu, and Kijung Shin. Characterization of simplicial complexes by counting simplets beyond four nodes. In *Proceedings of the ACM Web Conference 2023*, pages 317–327, 2023.

15 Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2–es, 2007.

16 David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

17 Zhaoqing Li, Zhenghong Deng, Zhen Han, Karin Alfaro-Bittner, Baruch Barzel, and Stefano Boccaletti. Contagion in simplicial complexes. *Chaos, Solitons & Fractals*, 152:111307, 2021.

18 Louis-David Lord, Paul Expert, Henrique M Fernandes, Giovanni Petri, Tim J Van Hartevelt, Francesco Vaccarino, Gustavo Deco, Federico Turkheimer, and Morten L Kringelbach. Insights into brain architectures from the homological scaffolds of functional connectivity networks. *Frontiers in systems neuroscience*, 10:85, 2016.

19 Mohammad Mahini and Salman Qadami. Network Signature. URL: `https://github.com/mmahini/graphlet-analysis`.

20 Dror Marcus and Yuval Shavitt. Rage–a rapid graphlet enumerator for large networks. *Computer Networks*, 56(2):810–819, 2012.

21 Giulia Preti, Gianmarco De Francisci Morales, and Francesco Bonchi. Strud: Truss decomposition of simplicial complexes. In *Proceedings of the Web Conference 2021*, pages 3408–3418, 2021.

22 Giulia Preti, Gianmarco De Francisci Morales, and Francesco Bonchi. Fresco: Mining frequent patterns in simplicial complexes. In *Proceedings of the ACM Web Conference 2022*, pages 1444–1454, 2022.

23 Pedro Ribeiro, Pedro Paredes, Miguel EP Silva, David Aparicio, and Fernando Silva. A survey on subgraph counting: concepts, algorithms, and applications to network motifs and graphlets. *ACM Computing Surveys (CSUR)*, 54(2):1–36, 2021.

24 Matteo Riondato and Evgenios M Kornaropoulos. Fast approximation of betweenness centrality through sampling. *Data Mining and Knowledge Discovery*, 30(2):438–475, 2016.

25 Rahmtin Rotabi, Krishna Kamath, Jon Kleinberg, and Aneesh Sharma. Detecting strong ties using network motifs. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 983–992, 2017.

26 Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, pages 243–246, 2015.

27 Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity: festschrift for alexey chervonenkis*, pages 11–30. Springer, 2015.

28 Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

29 Sebastian Wernicke. Efficient detection of network motifs. *IEEE/ACM transactions on computational biology and bioinformatics*, 3(4):347–359, 2006.

30 Sam Freddy Ludwien Windels. *Graphlet-adjacencies provide complementary views on the functional organisation of the cell and cancer mechanisms*. PhD thesis, UCL (University College London), 2022.