

Contribution Title^{*}

First Author¹[0000–1111–2222–3333], Second Author^{2,3}[1111–2222–3333–4444], and
Third Author³[2222–3333–4444–5555]

¹ Princeton University, Princeton NJ 08544, USA

² Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany
`lncs@springer.com`

<http://www.springer.com/gp/computer-science/lncs>

³ ABC Institute, Rupert-Karls-University Heidelberg, Heidelberg, Germany
`{abc,lncs}@uni-heidelberg.de`

Abstract. The abstract should briefly summarize the contents of the paper in 150–250 words.

Keywords: First keyword · Second keyword · Another keyword.

1 Introduction (DA SISTEMARE)

Graphs are used in a variety of sciences to model and to analyze complex relationships. In this framework, the search for interesting and relevant substructures is a standard procedure, and the detection of cliques and clique-like subgraphs is a fundamental tool in graph analysis. Such substructures have been applied in many different situations including community detection in social networks [7], [13], identification of real-time stories in the news [1] and graph visualization [17], [18]. In practice, due to noise in data, one is also interested in large “near-cliques”. While this is not a standard term, applications involve cliques that are missing a small sparse subgraph. For example, incomplete cliques have been used to predict missing pairwise interactions [17] and for identifying functional groups [4] in a protein interaction network. Also, they were exploited for community detection [19] and for detecting test collusion [2]. Recent works have used the fraction of near-cliques to k-cliques to define higher order variants of clustering coefficients [16].

In the present work, in order to quantitatively characterize these structures, we employ magnitude homology, a tool that comes from the field of algebraic topology. Magnitude is an isometric invariant of metric spaces, so-named for its web of connections to “size-like” quantities of significance in various corners of mathematics. Defined and first studied by Leinster [8], it is a special case of a general theory of magnitude of an enriched category, and has found applications in areas like biodiversity (e.g., Leinster and Cobbold [10]). As a finite graph naturally gives rise to a finite metric space, it is possible to associate magnitude with it. Magnitude homology has been invented by Hepworth and

^{*} Supported by organization x.

Willerton [6] as an enrichment of the magnitude of a graph which is equipped with a graph metric. The magnitude homology of graphs has been well studied in recent years and has proven to be a rich invariant [6] [3], [12]. Also, several tools for computing magnitude homology of a graph have been studied so far. For examples, Hepworth and Willerton proved a Mayer-Vietoris type exact sequence and a Kunneth type formula, and Gu [3] uses algebraic Morse theory for computation for some graphs. Although, in general, computation of magnitude homology remains a difficult problem.

TBC

In this work,...

The paper is organized as follows,...

2 Background

An undirected graph is a pair $G = (V, E)$ where V is a set of vertices and E is a set of edges (unordered pairs of vertices). A *trail* in G is an ordered sequence of vertices x_0, x_1, \dots, x_n such that there is an edge $\{x_i, x_{i+1}\}$ for all $0 \leq i < n$; a *path* is a trail with no repeated vertices. For the purposes of defining magnitude homology, we assume all graphs to have no self-loops and no multiedges [9]. We may view such a graph G as an extended metric space (i.e. a metric space with infinity allowed as a distance) whose points are the vertices of G by setting each edge to be of length one and defining an extended metric $d : V \times V \rightarrow [0, \infty]$ by declaring $d(u, v)$ to be equal to the length of a shortest path in G from u to v . By definition we let $d(u, v) = \infty$ if u and v lie in different components of G . We recall Hepworth and Willerton's construction [6] of the magnitude homology groups of a graph.

A k -path in a graph G is a $(k+1)$ -tuple (x_0, \dots, x_k) of vertices with $x_i \neq x_{i+1}$ and $d(x_i, x_{i+1}) < \infty$ for every $i \leq k-1$. The length of a k -path (x_0, \dots, x_k) in G is defined as the minimum length of a trail that visits x_0, x_1, \dots, x_k in this order, namely, $\ell(x_0, \dots, x_k) = d(x_0, x_1) + \dots + d(x_{k-1}, x_k)$. We define the (k, ℓ) -magnitude chain $MC_{k, \ell}(G)$ to be the free abelian group generated by the k -paths of length ℓ .

Definition 1. Let $(x_0, \dots, \hat{x}_i, \dots, x_k)$ denote the k -tuple obtained by removing the i -th vertex from the $(k+1)$ -tuple (x_0, \dots, x_k) . We define the differential

$$\partial_k : MC_{k, \ell}(G) \rightarrow MC_{k-1, \ell}(G)$$

as the sum $\partial_k = \sum_{i=1}^{k-1} \partial_{k, i}$ of the maps defined by

$$\partial_{k, i}(x_0, \dots, x_k) = a_i \cdot (x_0, \dots, \hat{x}_i, \dots, x_k),$$

where

$$a_i = \begin{cases} (-1)^{i+1}, & \text{if } \ell(x_0, \dots, \hat{x}_i, \dots, x_k) = \ell, \\ 0, & \text{otherwise.} \end{cases}$$

Definition 2 (Magnitude chain complex). We indicate as $MC_{*,\ell}(G)$ the following sequence of free abelian groups connected by differentials

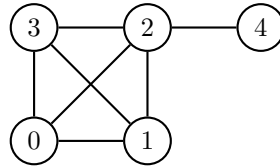
$$\cdots \rightarrow MC_{k+2,\ell}(G) \xrightarrow{\partial_{k+2}} MC_{k+1,\ell}(G) \xrightarrow{\partial_{k+1}} MC_{k,\ell}(G) \xrightarrow{\partial_k} MC_{k-1,\ell}(G) \rightarrow \cdots$$

It is shown in [6, Lemma 11] that the composition of two consecutive differentials $\partial_{k+1} \circ \partial_k$ vanishes, so that each chain $MC_{*,\ell}(G)$ is indeed a chain complex (as for the standard definition given in [5]) and it is thus possible to define its k -th homology group.

Definition 3. The k -magnitude homology group of the graph G is the abelian group defined by

$$MH_{k,l}(G) = H_k(MC_{*,l}(G)) = \frac{\ker(\partial_k)}{\text{imm}(\partial_{k+1})}.$$

Example 1. Consider the following graph G



We want to compute $MH_{2,2}(G)$. From Definition 3 we need to evaluate the quotient between the kernel of ∂_2 and the image of ∂_3 . So consider the map $\partial_2 : MC_{2,2}(G) \rightarrow MC_{1,2}(G)$. $MC_{2,2}(G)$ is generated by the 2-paths (i.e., triplets) in G of length 2. Therefore, $MC_{2,2}(G)$ is generated by the following 44 elements: $(0, 1, 0), (0, 1, 2), (0, 1, 3), (0, 2, 0), (0, 2, 1), (0, 2, 3), (0, 2, 4), (0, 3, 0), (0, 3, 1), (0, 3, 2), (1, 0, 1), (1, 0, 2), (1, 0, 3), (1, 2, 0), (1, 2, 1), (1, 2, 3), (1, 2, 4), (1, 3, 0), (1, 3, 1), (1, 3, 2), (2, 0, 1), (2, 0, 2), (2, 0, 3), (2, 1, 0), (2, 1, 2), (2, 1, 3), (2, 3, 0), (2, 3, 1), (2, 3, 2), (2, 4, 2), (3, 0, 1), (3, 0, 2), (3, 0, 3), (3, 1, 0), (3, 1, 2), (3, 1, 3), (3, 2, 0), (3, 2, 1), (3, 2, 3), (3, 2, 4), (4, 2, 0), (4, 2, 1), (4, 2, 3), (4, 2, 4)$.

Similarly, $MC_{1,2}(G)$ is generated by the pairs representing 1-paths in G of length 2: $(0, 4), (1, 4), (3, 4), (4, 0), (4, 1), (4, 3)$.

We thus have that the kernel of ∂_2 is generated by the 38 elements whose length diminishes when the middle vertex is removed. That is all elements in $MC_{2,2}(G)$ except $(0, 2, 4), (1, 2, 4), (3, 2, 4), (4, 2, 0), (4, 2, 1), (4, 2, 3)$.

Being ∂_2 a linear operator we can represent it in the following way: we construct the associated matrix indexing the rows and the columns with the elements of $MC_{1,2}(G)$ and $MC_{2,2}(G)$ respectively. We then evaluate the differential as stated in Definition 1 and fill the matrix with the coefficients appearing in the evaluation. So for example, $\partial_2(0, 1, 2) = 0 \cdot (0, \hat{1}, 2) = 0 \cdot (0, 2)$, since $\ell(0, 1) < 2$, and thus the entry in position $((0, 2), (0, 1, 2))$ will be zero. Similarly, $\partial_2(0, 2, 4) = 1 \cdot (0, \hat{2}, 4) = 1 \cdot (0, 4)$, since $\ell(0, 4) = 2$, and thus the entry in position $((0, 4), (0, 2, 4))$ will be one.

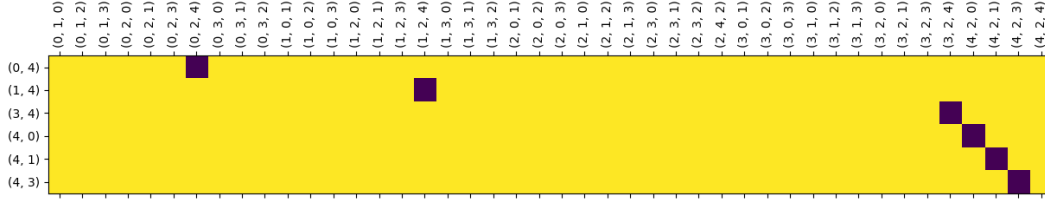


Fig. 1: Matrix representation of ∂_2 . The $+1$ entries are represented in purple, while the 0 entries are filled with yellow.

For what concerns the image of ∂_3 , we need to see which elements of $MC_{3,2}$ are sent to $MC_{2,2}$. But since $MC_{3,2}(G)$ is generated by the 4-tuples representing 3-paths in G of length 2, and since any path connecting 4 vertices must have length at least 3, $MC_{3,2}(G)$ is the trivial group $\langle 0 \rangle$. Therefore the image of ∂_3 is $\langle 0 \rangle$. In conclusion, we have that

$$MH_{2,2}(G) = \frac{\ker(\partial_2)}{\text{imm}(\partial_3)} = \ker(\partial_2),$$

and so $|MH_{2,2}(G)| = 38$.

Remark 1. We point out that if we represent the ranks of the magnitude homology groups of a graph in a (k, ℓ) -table as in Table 1, we will always have that the table is lower triangular. In other words, $MH_{k,\ell}(G) \neq 0$ implies that $k \leq \ell$. This is because if $MH_{k,\ell}(G) \neq 0$ then $MC_{k,\ell}(G) \neq 0$, and so there is at least a tuple (x_0, \dots, x_k) satisfying $\ell(x_0, \dots, x_k) = d(x_0, x_1) + \dots + d(x_{k-1}, x_k) = \ell$. Now, since consecutive vertices are distinct by construction, $d(x_i, x_{i+1})$ is at least 1 for every $0 \leq i \leq k-1$, which means k can be at most ℓ .

		k					
		0	1	2	3	4	5
l	0	5					
	1		10				
	2			10			
	3				10	10	
	4					30	10
	5						50

Table 1: Ranks of $MH_{k,l}(C_5)$ computed using Python **ADD GIT REF**

It is proven in [6] (Proposition 9) that given a graph G , if we indicate by V the set of vertices and by $2E$ the set of oriented edges, then $MH_{0,0}(G)$ is the free abelian group on V and $MH_{1,1}(G)$ is the free abelian group on $2E$.

This provides us with an interpretation of the first two groups on the diagonal, but leaves the question about the meaning of other magnitude homology groups open.

3 Interpretation of magnitude homology

The aim of this section is to establish an interpretation for magnitude homology groups on the first and second diagonal.

Our starting point was to reduce the size of the magnitude chain $MC_{k,l}(G)$. In fact, consider for example the group $MC_{2,2}(G)$ of example 1. The generator $(0, 1, 0)$ is the tuple corresponding to the path going twice across the edge $(0, 1)$, and so its presence does not provide any information about the graph and just “adds noise” to the information contained in both the magnitude chain and in the magnitude homology group. Therefore, we revise the definition of magnitude chain considering the “plain” subgroup of $MC_{k,l}(G)$, that is the subgroup of paths where a vertex is never required to be revisited.

Definition 4. (*Plain magnitude chain*) Let G be a graph. We define the plain (k, l) -magnitude chain $PMC_{k,l}(G)$ to be the free abelian group generated by tuples (x_0, \dots, x_k) of vertices of G such that (x_0, \dots, x_k) is a k -path of length l with $x_i \neq x_j$ for every $0 \leq i, j \leq k$.

Keeping the same differential as before we can construct the *plain magnitude chain complex* $PMC_{*,l}(G)$

$$\cdots \rightarrow PMC_{k+1,l}(G) \xrightarrow{\partial_{k+1}} PMC_{k,l}(G) \xrightarrow{\partial_k} PMC_{k-1,l}(G) \rightarrow \cdots$$

and subsequently define the *plain (k, l) -magnitude homology group*

$$PMH_{k,l}(G) = H_k(PMC_{*,l}(G)) = \frac{\ker(\partial_k)}{\text{imm}(\partial_{k+1})}.$$

Remark 2. We point out that all definitions and properties regarding magnitude homology proved in [6] and [11] continue to be valid for plain magnitude homology. This comes from the fact that, by construction, $PMC_{k,l}(G)$ is a subgroup of $MC_{k,l}(G)$, being generated by a subset of the generating set of the (k, l) -magnitude chain, and since we did not change the definition of the differential operator ∂_k it follows that the plain magnitude homology group $PMH_{k,l}(G)$ is a subgroup of the magnitude homology group $MH_{k,l}(G)$ for every $k, l \geq 0$.

In particular, with this new definition, $PMH_{0,0}(G)$ and $PMH_{1,1}(G)$ are still counting the number of vertices and edges in a graph respectively, since the generators of the groups $MC_{0,0}(G)$ and $MC_{1,1}(G)$ already satisfy the condition of not revisiting vertices.

3.1 First diagonal: counting triangles and squares

We are concerned in this section with the analysis of the information contained in the groups $PMH_{k,k}(G)$. Notice that Remark 1 implies that $PMC_{k+1,k}(G)$ (and more in general $MC_{k+1,k}(G)$) will always be the trivial group $\langle 0 \rangle$, hence the image of ∂_{k+1} will be $\langle 0 \rangle$, meaning the plain homology group $PMH_{k,k}(G)$ will be entirely determined by $\ker(\partial_k)$.

Consider now $PMH_{2,2}(G) = \ker(\partial_2)$. Take the matrix associated to ∂_2 constructed as described in example 1 and notice the following facts.

- The dimension of the kernel of our matrix is equal to the number of all-zero columns plus the number of columns that are “copies” of a previously written column.
- The number of all-zero columns is (modulo automorphisms) equal to the number of triangles contained in the graph. This is because if a 2-path (x_0, x_1, x_2) is sent to zero after removing the vertex x_1 , it means that the shortest path between x_0 and x_2 has length smaller than 2. So there exists an edge (x_0, x_2) and equivalently a triangle (x_0, x_1, x_2) .
- The number of repeated non-zero columns in the kernel indicates how many 2-paths (x_0, x, x_2) are sent to the same 1-path (x_0, x_2) , and this enables us to produce (at least) an estimate for the number of 4-cycles contained in the graph. Indeed, if $\ell(x_0, x, x_2)$ does not decrease after removing the middle vertex x , then this means that any shortest path from x_0 to x_2 is of length 2. In other words, two 2-paths with the same image signify the presence of a 4-cycle in the graph.

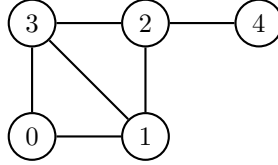
Given this, to obtain the number of 3-cycles contained in our graph we need to divide the number of all-zero columns by 6, i.e. by the cardinality of the automorphisms group of the triangle D_3 .

For what concerns the count of 4-cycles, notice that the number of non-zero columns will be $2N$ (since we are considering all possible orientations), so we first need to divide this number by two. At this point, depending on the complexity of the graph, two situations might arise:

- If we know the image of each 2-path we are able to compute the exact number of 4-cycles. Indeed if the same column is repeated n times this means there are n 2-paths $(x, x_1, y), (x, x_2, y), \dots, (x, x_n, y)$ sharing the same endpoints, and so they identify $\sum_{i=1}^{n-1} i = \frac{(n-1)n}{2}$ 4-cycles.
- In case we are not aware of the image of each 2-path, we can still produce upper and lower bounds for number of 4-cycles contained in our graph. Indeed, to produce a square we need at least two 2-paths $(x, x_1, y), (x, x_2, y)$ with the same image (x, y) , so the number of 4-cycles will be at least $\lfloor \frac{N}{2} \rfloor$. Also, the N non-zero columns might all be sharing the same endpoints, in which case the number of 4-cycles would increase to $\sum_{i=1}^{N-1} i = \frac{(N-1)N}{2}$. So, we are able to conclude that the number S of squares contained in our graph G is

$$\left\lfloor \frac{N}{2} \right\rfloor \leq S \leq \frac{(N-1)N}{2}.$$

Example 2. Consider the following graph G



We want to compute $PMH_{2,2}(G)$. The plain magnitude chain $PMC_{2,2}(G)$ is generated by $(0, 1, 2)$, $(0, 1, 3)$, $(0, 3, 1)$, $(0, 3, 2)$, $(1, 0, 3)$, $(1, 2, 3)$, $(1, 2, 4)$, $(1, 3, 0)$, $(1, 3, 2)$, $(2, 1, 0)$, $(2, 1, 3)$, $(2, 3, 0)$, $(2, 3, 1)$, $(3, 0, 1)$, $(3, 1, 0)$, $(3, 1, 2)$, $(3, 2, 1)$, $(3, 2, 4)$, $(4, 2, 1)$, $(4, 2, 3)$, while $PMC_{1,2}(G)$ is generated by $(0, 2)$, $(1, 4)$, $(2, 0)$, $(3, 4)$, $(4, 1)$, $(4, 3)$, and the matrix representing our differential is the one displayed in Figure 2.

We see that there are twelve all-zero columns, counting the triangles $(0, 1, 3, 0)$ and $(1, 2, 3, 1)$. We also have four non-zero columns, so modulo orientation they are just two and therefore they identify one 4-cycle. In this example they represent the square $(0, 1, 2, 3, 0)$.

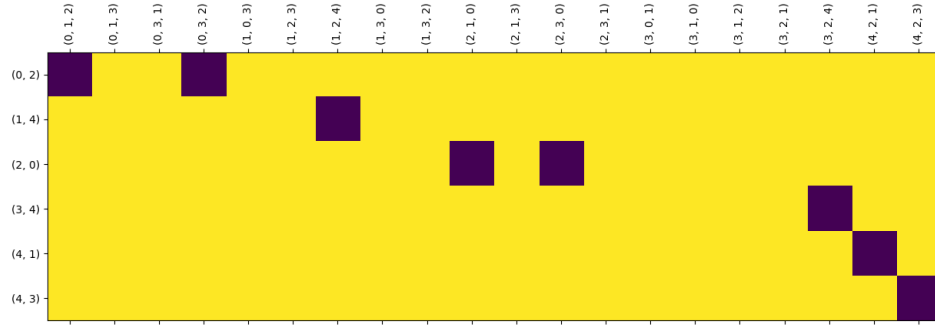


Fig. 2: Matrix representation of $\partial_2 : PMC_{2,2}(G) \rightarrow PMC_{1,2}(G)$. The $+1$ entries are represented in purple, while the 0 entries are filled with yellow.

An analysis on the higher degree magnitude homology groups on the diagonal reveals that the $PMH_{k,k}(G)$'s contain information about which k -tuples of vertices of G contain all possible triangles. Indeed, think about all-zero columns in a groups $PMH_{k,k}(G)$, $k \geq 3$. If a $(k-1)$ -tuple (x_0, \dots, x_k) is sent to zero, this means all the summand maps $\partial_{k,i}$ defining ∂_k have image zero, which in turns indicate that for every $1 \leq i \leq k-1$ $\ell(x_0, \dots, x_{i-1}, x_{i+1}, \dots, x_k) < l$. Therefore, for every $1 \leq i \leq k-1$ there exists a triangle (x_{i-1}, x_i, x_{i+1}) , and this provides a hint on where to look for a clique in our graph G .

3.2 Second diagonal: counting 4-cliques and 5,6-“near-cliques”.

We are concerned in this section with the analysis of the information contained in the groups $PMH_{k-1,k}(G)$.

While the question about a general interpretation for all magnitude homology groups $PMH_{k-1,k}(G)$ on the second diagonal remains open, we were able to understand the information contained in $PMH_{2,3}(G)$. Specifically, we believe the third plain magnitude homology group on the second diagonal provides us with information regarding the number of 4-cliques, 5-cliques and 6-cliques in the graph.

Consider the following chain

$$\begin{array}{ccccccc} \dots & \rightarrow & PMC_{3,3}(G) & \rightarrow & PMC_{2,3}(G) & \rightarrow & PMC_{1,3}(G) \rightarrow 0. \\ & & \downarrow & & \downarrow & & \downarrow \\ & & (x_0, x_1, x_2, x_3) & & (x_0, \hat{x}, x_2, x_3) & & (x_0, \hat{x}, \hat{y}, x_3) \end{array}$$

and assume $x_3 \neq \hat{x}$.

If $(x_0, \hat{x}_1, x_2, x_3) \in \ker(\partial_2)$ then one of the following is true:

- At least one other different 3-tuple in $PMC_{2,3}(G)$ is sent to the same 2-tuple in $PMC_{1,3}(G)$, which means there is either a 4-cycle or a 6-cycle.
- The considered tuple in $PMC_{2,3}(G)$ is sent to zero, which means there exists either a 4-cycle or a 5-cycle.

Now, when we quotient by the image of ∂_3 we are in fact disregarding the elements $(x_0, \hat{x}_1, x_2, x_3) \in PMC_{2,3}(G)$ such that $\ell(x_0, \hat{x}_1, x_2, x_3) = \ell(x_0, x_1, x_2, x_3)$. That is, we are disregarding the tuples that do not contain the triangle (x_0, x_1, x_2) , and that therefore cannot be part of a clique.

Summarizing, $PMH_{2,3}(G)$ is counting 4-cliques and candidates 5, 6-cliques.

Remark 3. We point out that the hypothesis “ $x_3 \neq \hat{x}$ ” is crucial to obtain this interpretation. Indeed, without this assumption it could happen that $x_3 = \hat{x}$, which would mean “revisiting an edge” and adding a lot of noise to $PMH_{2,3}(G)$.

We recall the definition of *diagonal graph* introduced by Hepworth and Willerton in [6].

Definition 5. A graph G is called *diagonal* if $MH_{k,l}(G) = 0$ whenever $k \neq l$.

The provided interpretation of $PMH_{2,3}(G)$ suggest the following fact.

Proposition 1. If a graph G is diagonal, then it is clique-free.

Proof. Suppose G is diagonal, then $MH_{k,l}(G) = 0$ whenever $k \neq l$ and by Remark 2 $PMH_{k,l}(G) = 0$ if $k \neq l$. In particular $PMH_{2,3}(G) = 0$, meaning the graph contains no 4-clique, and therefore no bigger clique.

4 Relation with clustering coefficients

In Graph Theory, a clustering coefficient is a structural feature that measures the degree to which nodes in a graph tend to cluster together. In other words, it tells how connected a vertex’s neighbors are to one another. There are two

existing versions of this measure. The *global*, which was designed by Wasserman and Faust in [14] to give an overall indication of the clustering in the network, and the *local*, first defined by Watts and Strogatz in [15] to give an indication about the tendency to cluster near a specific node.

In this section we provide a way to compute both clustering coefficients of a graph $G = (V, E)$ via $PMH_{2,2}(G)$, determining thus a close relation between these tools.

4.1 Local clustering coefficient

The local clustering coefficient C_i of a node x_i describes the likelihood that the neighbours of x_i are also connected. To compute C_i we consider the neighborhood N_i of x_i , where $N_i = \{x_j : (x_i, x_j) = e_{ij} \in E\}$ and compute the fraction of the number of links between the vertices within N_i divided by the number of links that could possibly exist between them. That is, we set

$$C_i = \frac{2\{e_{jk} : x_j, x_k \in N_i \text{ and } e_{jk} \in E\}}{d_i(d_i - 1)},$$

where $d_i = |N_i|$ is the degree of the vertex x_i .

In other words, we are dividing the number of triangles x_i is part of by the number of 2-paths of length 2 containing x_i .

Therefore, call $PMC_{2,2}^i(G)$ the subgroup of $PMC_{2,2}(G)$ such that x_i is the middle vertex of any 2-path, so $PMC_{2,2}^i(G) = \{(x, x_i, y) : \ell(x, x_i, y) = 2\}$. Then, by section 3.1, the number of triangles containing x_i is precisely the number of all-zero columns of $\ker(\partial_2(PMC_{2,2}^i(G)))$. Calling this number Z_i , we can write the local clustering coefficient as

$$C_i = \frac{2Z_i}{d_i(d_i - 1)}.$$

Remark 4. Given the connection just established between the local clustering coefficient and plain magnitude homology, one could think of using $PMH_{2,2}$ in a network analysis context as a *centrality measure*: if for a given vertex v_i the number Z_i defined above takes low values it means there are few connections between neighbors of x_i , meaning x_i has a lot of power over information flow.

4.2 Global clustering coefficient

The global clustering coefficient C is based on 3-tuples, i.e. on elements of $PMC_{2,2}$, and is computed as the number of closed 3-tuples (or $3 \times$ triangles) over the total number of 3-tuples (both open and closed). That is, calling Z the number of all-zero columns in $\ker(\partial_2(PMC_{2,2}(G)))$

$$C = \frac{Z}{|PMC_{2,2}(G)|}.$$

5 Algorithm complexity

6 Conclusions

References

1. Angel, A., Koudas, N., Sarkas, N., Srivastava, D., Svendsen, M., Tirthapura, S.: Dense subgraph maintenance under streaming edge weight updates for real-time story identification. *The VLDB journal* **23**(2), 175–199 (2014)
2. Belov, D.I., Wollack, J.A.: Graph theory approach to detect examinees involved in test collusion. *Applied Psychological Measurement* **45**(4), 253–267 (2021)
3. Gu, Y.: Graph magnitude homology via algebraic morse theory. *arXiv preprint arXiv:1809.07240* (2018)
4. Han, K., Cui, G., Chen, Y.: Identifying functional groups by finding cliques and near-cliques in protein interaction networks. In: 2007 *Frontiers in the Convergence of Bioscience and Information Technologies*. pp. 159–164. IEEE (2007)
5. Hatcher, A.: *Algebraic topology* (2005)
6. Hepworth, R., Willerton, S.: Categorifying the magnitude of a graph. *arXiv preprint arXiv:1505.04125* (2015)
7. Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Trawling the web for emerging cyber-communities. *Computer networks* **31**(11-16), 1481–1493 (1999)
8. Leinster, T.: The magnitude of metric spaces. *Documenta Mathematica* **18**, 857–905 (2013)
9. Leinster, T.: The magnitude of a graph. In: *Mathematical Proceedings of the Cambridge Philosophical Society*. vol. 166, pp. 247–264. Cambridge University Press (2019)
10. Leinster, T., Cobbold, C.A.: Measuring diversity: the importance of species similarity. *Ecology* **93**(3), 477–489 (2012)
11. Leinster, T., Shulman, M.: Magnitude homology of enriched categories and metric spaces. *Algebraic & Geometric Topology* **21**(5), 2175–2221 (2021)
12. Sazdanovic, R., Summers, V.: Torsion in the magnitude homology of graphs. *Journal of Homotopy and Related Structures* **16**(2), 275–296 (2021)
13. Sozio, M., Gionis, A.: The community-search problem and how to plan a successful cocktail party. In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. pp. 939–948 (2010)
14. Wasserman, S., Faust, K., et al.: *Social network analysis: Methods and applications* (1994)
15. Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘small-world’ networks. *nature* **393**(6684), 440–442 (1998)
16. Yin, H., Benson, A.R., Leskovec, J., Gleich, D.F.: Local higher-order graph clustering. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. pp. 555–564 (2017)
17. Zhang, Y., Parthasarathy, S.: Extracting analyzing and visualizing triangle k-core motifs within networks. In: 2012 *IEEE 28th international conference on data engineering*. pp. 1049–1060. IEEE (2012)
18. Zhao, F., Tung, A.K.: Large scale cohesive subgraphs discovery for social network visual analysis. *Proceedings of the VLDB Endowment* **6**(2), 85–96 (2012)
19. Zhu, J., Chen, B., Zeng, Y.: Community detection based on modularity and k-plexes. *Information Sciences* **513**, 127–142 (2020)