A non-overlapping community detection approach based on α-structural similarity

Motaz Ben Hassine, Saïd Jabbour, Mourad Kmimech, Badran Raddaoui, Mohamed Graiet

A Non-overlapping Community Detection Approach based on α-Structural Similarity

Motaz Ben Hassine[1,2], Saïd Jabbour[1], Mourad Kmimech[3], Badran Raddaoui[4,5], and Mohamed Graiet[6]

[1] CRIL, University of Artois & CNRS, Lens, France
{benhassine,jabbour}@cril.fr
[2] University of Monastir, UR-OASIS-ENIT, Monastir, Tunisia
motaz.benhassine@fsm.u-monastir.tn
[3] ESILV, Courbevoie, France
mourad.kmimech@devinci.fr
[4] SAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, France
[5] Institute for Philosophy II, Ruhr University Bochum, Germany
badran.raddaoui@telecom-sudparis.eu
[6] LS2N Nantes, Nantes, France
mohamed.graiet@imt-atlantique.fr

Abstract. Community detection in social networks is a widely studied topic in Artificial Intelligence and graph analysis. It can be useful to discover hidden relations between users, the target audience in digital marketing, and the recommender system, amongst others. In this context, some of the existing proposals for finding communities in networks are agglomerative methods. These methods used similarities or link prediction between nodes to discover the communities in graphs. The different similarity metrics used in these proposals focused mainly on common neighbors between similar nodes. However, such definitions are missing in the sense that they do not take into account the connection between common neighbors. In this paper, we propose a new similarity measure, named α-Structural Similarity, that focuses not only on common neighbors of nodes but also on their connections. Afterwards, in the light of α-Structural Similarity, we extend the Hierarchical Clustering algorithm to identify disjoint communities in networks. Finally, we conduct extensive experiments on synthetic networks and various well-known real-world networks to confirm the efficiency of our approach.

Keywords: Local similarity · Community detection · Social network · Agglomerative approaches.

1 Introduction

Over the years, graphs have been widely used to model various real-world applications where vertices represent objects and edges represent relationships between these objects. Social network analysis is one such application where the automatic discovery of communities has been a major challenge in recent years

[7]. Community detection involves identifying a set of nodes that are strongly connected within but weakly connected outside a community [15]. Community detection algorithms can be categorized into *overlapping* and *non-overlapping* approaches, with the latter being of particular interest in this paper. Long ago, various non-overlapping approaches have been studied. More precisely, Enright et al. [5] introduced a novel approach called TRIBE-MCL. The authors used the sequence protein similarity to detect the sequence protein families. Furthermore, the well-known approach, coined LPA (Label propagation algorithm), was proposed by Raghavan et al. [16] in which the nodes having the same label form the same community. In addition, Rodrigo et al. [1] introduced a novel optimized measure for detecting communities called surprise. In addition, Traag et al. [21] introduced a novel method based on a new measure called significance for detecting clusters. Moreover, Traag et al. [22] proposed a novel approach that improved the Louvain algorithm [2]. The authors found that 25% of communities are poorly connected, and then they presented a novel algorithm named LEIDEN to overcome this issue. Moreover, they enhanced the running time. Despite exhibiting strong performance, these existing proposals are still limited in terms of community quality, due to the wide variety of the real-world social network structures. Notice also that other non-overlapping approaches exist, which are based on local similarities and modularity maximization. Precisely, Yi-CHENG CHEN et al. [4] developed an approach named Hierarchical Clustering (HC, for short) used in the context of the influence maximization problem. Their algorithm starts by considering each vertex as an initial community. Then, the authors merged each pair of communities having the highest similarity values whose merging gives the greatest increase in modularity. Afterwards, they applied a local Structural Similarity (in short 2S) having the same definition of the Salton similarity [19] but utilizing differently defined neighborhood sets. Despite the good results in terms of community quality demonstrated by the use of the 2S in the HC approach, there are cases where the definition of the 2S may not be sufficient. This raises the question of whether the interaction between common neighbors enriched with the definition of the 2S would ultimately improve the quality of the detected communities. In this context, we propose a new similarity measure, named α-Structural Similarity (α-2S), that focuses not only on common neighbors of nodes but also on their connections. Ultimately, considering α-2S, we extended the HC algorithm to identify disjoint communities in networks. In this paper, we introduce some formal notations in Section 2. Then, in Section 3, we deal with a HC based on α-2S which we call α-HC. Section 4 presents our experiments on both synthetic and real-world datasets. Finally, Section 5 concludes the paper with hints for future work.

2 Preliminaries

In this paper, we consider a simple undirected graph $G = (V, E)$, where V is the set of vertices, and E is the set of edges. The set of **neighbors** of a node $u \in V$ is defined as $N(u) = \{v \mid (u, v) \in E\}$. The **degree** of $u \in V$

is then $|N(u)|$. For a given node $u \in V$, we write $adj(u)$ for the set of neighbors of u including u itself, i.e., $adj(u) = N(u) \cup \{u\}$. Given a set of nodes $X \subseteq V$, a **subgraph** induced by X, denoted as $G_X = (X, E_X)$, is a graph over X s.t. $E_X = \{(u, v) \in E \mid u, v \in X\}$. Further, let $w : E \to \mathbb{R}_{>0}$ be a function that maps each edge from E to a non negative real value which $\in]0..1]$. We write $E^w_{\max}$ for the set of edges sets with a maximum weight w, i.e., $E^w_{\max} = \{\{u, v\}\ s.t.\ (u, v) \in E \mid \nexists\ (u', v') \in E\ s.t.\ w(u', v') > w(u, v)\}$. A graph $G = (V, E)$ is called a **clique** iff. $\forall\ u \in V, |adj(u)| = |V|$. A graph $G = (V, E)$ can be splitted into numerous subgroups called **communities**, denoted as $C_G = \{c_1, c_2, \ldots, c_m\}$. Let $P(V) = 2^V$ the power set of V. The **Merge** function is defined as $Merge : 2^{P(V)} \times 2^{P(V)} \to 2^{P(V)}; (C_G, E^w_{max}) \mapsto Merge(C_G, E^w_{max})$ returns a merged set of subsets i.e., $Merge(C_G, E^w_{max}) = \{\ c_i \cup \{u, v\}, \{u, v\} \in E^w_{max},\ c_i \in C_G,\ \mid \forall\ 1 \leq i \leq m\ s.t.\ c_i \cap \{u, v\} \neq \emptyset\}$. Let $d_{\max}$ be the **maximum degree** of G, i.e., $d_{\max} = \max_{|N(u)|}\{u \in V\}$. Besides, let d_{av} be the **average number of neighbors** of all the vertices in X, i.e., $d_{av} = \frac{1}{|X|} \sum_{u \in X} |N(u)|$.

The similarity measure $2S$ [4] is a local function that uses the immediate neighborhood between vertices as defined below.

Definition 1 (Structural Similarity). *Let $G = (V, E)$ be an undirected graph and $(u, v) \in E$, then the 2S of u and v, denoted by $s_2(u, v)$, is defined as:*

$$s_2(u, v) = \frac{|adj(u) \cap adj(v)|}{\sqrt{|adj(u)| \times |adj(v)|}} \tag{1}$$

For the quality of the set of communities, the modularity is formally defined as follows:

Definition 2 (Modularity [4]). *Let $G = (V, E)$ be an undirected graph with $C_G = \{c_1, c_2, \ldots, c_m\}$ is the set of communities of G and s be a general similarity measure. The modularity is defined as:*

$$Q(C_G) = \sum_{i=1}^{m} \left[\frac{IS_i}{TS} - \left(\frac{DS_i}{TS} \right)^2 \right] \tag{2}$$

where $IS_i = \sum_{u,\, v \in c_i} s(u, v)$, $DS_i = \sum_{u \in c_i,\, v \in V} s(u, v)$, *and* $TS = \sum_{u,\, v \in V} s(u, v)$.

3 A Hierarchical Clustering Approach based on α-Structural Similarity

We propose an extended version of the 2S. Indeed, we added another term in Equation 1, which will denote the rate of interaction between common neighbors. we added the concept of connection between common neighbors. Therefore, the identification of communities will be more significant. To be more precise, let

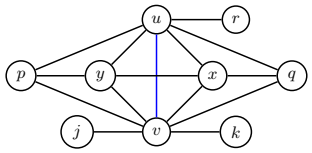
$G = (V, E)$ be a graph and given $(u, v) \in E$. We consider the ratio between the number of connections between common neighbors (i.e., $adj(u) \cap adj(v)$) and the minimum between the number of edges of the subgraph induced by $adj(u)$ and the one induced by $adj(v)$. Then, the new version of the similarity metric, which will be called α-2S is formally defined as follows:

$$s_2^{\alpha}(u, v) = (1 - \alpha)\frac{|adj(u) \cap adj(v)|}{\sqrt{|adj(u)| \times |adj(v)|}} + \alpha\frac{|E_{adj(u) \cap adj(v)}|}{\min(|E_{adj(u)}|, |E_{adj(v)}|)} \quad (3)$$

where α is a parameter in [0..1]. It should be noted that $s_2^{\alpha}(u, v) \in]0..1]$.

The parameter α in Equation 3 ensures the trade-off between the notion of common neighborhood and the rate of their interactions. Thus, it is interesting to determine the value of α. It should be noted that when $\alpha = 0$, the α-2S is identical to the 2S. We illustrate the behaviour of our α-2S through the following example. We set $\alpha = 0.8$.

Example 1. Let us consider the undirected graph depicted in Figure 1.



Then, we have that:

$$\begin{aligned} s_2^{0.8}(u, v) &= (1 - 0.8)\frac{|adj(u) \cap adj(v)|}{\sqrt{|adj(u)| \times |adj(v)|}} + 0.8\frac{|E_{adj(u) \cap adj(v)}|}{min(|E_{adj(u)}|, |E_{adj(v)}|)} \\ &= (1 - 0.8) \times \frac{6}{\sqrt{7 \times 8}} + 0.8 \times \frac{12}{min(13, 14)} \\ &= (1 - 0.8) \times \frac{6}{\sqrt{56}} + 0.8 \times \frac{12}{13} = 0.89 \end{aligned}$$

Fig. 1. A simple undirected graph with $\alpha = 0.8$.

In what follows, to show the effectiveness of our new similarity metric α-2S, let us consider the case of two disjoint cliques C_1 and C_2. A set of links is then added over C_1 and C_2 to form a new clique C_3 overlapping with C_1 and C_2. We will show how the two communities formed by the initial cliques C_1 and C_2 remain identifiable when varying the set of links between C_1 and C_2. The obtained graph will be coined k-linked-cliques graph.

Definition 3. *Let $G = (V, E)$ be a graph and k, n two integers s.t. $1 \leq k \leq n$ and $n \geq 4$. Then, G is called a* **k-*linked-cliques graph*** *iff. G is formed by three cliques $C_1 = (V_1, E_1)$, $C_2 = (V_2, E_2)$, and $C_3 = (V_3, E_3)$ where:*

- $V = V_1 \uplus V_2$ *s.t.* $|V_1| = |V_2| = n$, *and* $E_1 \cap E_2 = \emptyset$
- $V_3 \subseteq V_1 \cup V_2$, $|V_1 \cap V_3| = k$, *and* $|V_2 \cap V_3| = 2$

In the sequel, we consider $V_2 \cap V_3 = \{u_0, v_0\}$. Below we are interested in computing the values of k making the similarity inside C_1 and C_2 higher than the one of the edges of C_3 using both $2S$ and α-2S metrics.

Proposition 1. *Let G be a k-linked-cliques graph. $\forall\ (u_1, v_1) \in (E_1 \cup E_2)$ and $(u_2, v_2) \in E_3 \setminus (E_1 \cup E_2)$, we have $s_2(u_2, v_2) < s_2(u_1, v_1)$ iff. $k < n - 1$.*

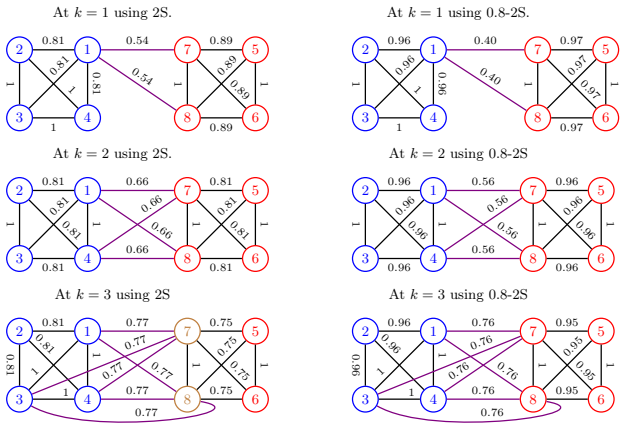
Proof. (Refer to Appendix A)

Proposition 2. *Let G be a k-linked-cliques graph. $\forall\ (u_1, v_1) \in (E_1 \cup E_2)$ and $(u_2, v_2) \in E_3 \setminus (E_1 \cup E_2)$, we have $s_2^\alpha(u_2, v_2) < s_2^\alpha(u_1, v_1)$ iff. $k \leq n - 1$ and $\alpha > 0.06$.*

Proof. (Refer to Appendix B)

Proposition 1 states that for a k-linked-cliques graph, the 2S of edges linking C_1 and C_2 is lower than the ones of edges within the two cliques C_1 and C_2 for 1 until $n-2$. While in the proposition 2, α-2S is from 1 until $n-1$ which makes it better than the 2S on a k-linked-cliques graph.

Example 2. Let's consider the k-linked-cliques graph depicted in Figure 2. This figure illustrates an example of k-linked-clique graph for $k = 1$ to $k = 4$. As shown, the similarity values of the linking edges (colored in purple) are lower than the rest of the edges in the graph from $k = 1$ to $k = 2$ while according to 0.8-2S, it is from $k = 1$ to $k = 3$.



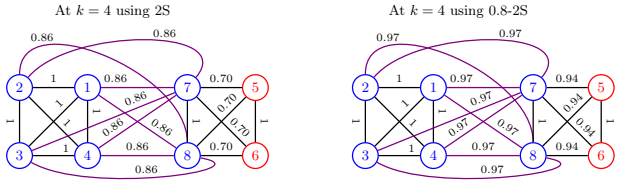


Fig. 2. An example of k-linked-cliques with $n = 4$ and $\alpha = 0.8$.

Based on α-2S, our approach follows the one HC where 2S is substituted with α-2S. Algorithm 1 named α-HC describe our approach. First, we computed the similarity for each edge in G. Second, we initialized the set of communities, where each vertex is considered as community. Then, we calculated the corresponding modularity. Third, at each iteration, we merged each pair of nodes having the strongest similarity. Fourth, the modularity is recalculated on the current merged set. If we have a modularity gain, then the process continue. Otherwise, the previous result is considered as the best clustering set.

Algorithm 1: α-HC

Input: $G(V, E), \alpha$
Output : C_G
begin
  for $(u, v) \in E$ **do**
    $w(u, v) \leftarrow s_2^{\alpha}(u, v)$
  end
  $C_G \leftarrow \emptyset$
  for $u \in V$ **do**
    $C_G \leftarrow C_G \cup \{\{u\}\}$
  end
  $PreviousModularity \leftarrow Q(C_G)$
  $CurrentModularity \leftarrow PreviousModularity$
  $C \leftarrow C_G$
  while $CurrentModularity \geq PreviousModularity$ **do**
    $C_G \leftarrow C$
    $C \leftarrow Merge(C, E^w_{max})$
    $PreviousModularity \leftarrow CurrentModularity$
    $CurrentModularity \leftarrow Q(C)$
  end
  return C_G
end

Computational complexity . Usually, clustering algorithms based on modularity maximization require $O((|E|+|V|)|V|)$ [12]. Similarity computation should be considered. Indeed, the 2S requires $O(|V|d^3_{max})$ [14]. The extraction of edges of an induced subgraph requires $O(|X|d_{av})$ [13]. Then, α-2S requires $O(|V|d^3_{max}+$

$4|E||X|d_{av}$) and therefore α-HC requires $O(|V|d^3_{max}+4|E||X|d_{av}+(|E|+|V|)|V|)$. The complexity is polynomial.

4 Experiments

To validate our proposal, we propose an implementation [23]. We performed two kinds of experiments. First, α-HC and HC were tested on artificial networks called LFR networks [9] by changing a parameter called mixing parameter μ from 0.1 to 0.9. The parameter μ allows to control the mixture between communities. When μ is growing the identification of the communities becomes harder. Our goal is to identify how community quality is correlated to μ. In our setting, for each value of μ, the value of α is varied from 0.1 to 1 to identify the best value of α providing the best quality. The quality of the founded communities were measured using the well-known F1-score [18] and NMI [6] metrics. In our second experiment, α is fixed, and then we compare our approach with other disjoint community detection approaches on 8 well-known real-world datasets. We illustrated all the datasets in tables 1 and 2. We denote by `AD` the average degree, `GT` the ground truth communities, and `MinCS` the minimum community size.

Table 1. Real-world datasets

Datasets	**Nodes/Edges**	**GT**	**Source**
Karate	34/78	2	[24]
Dolphin	62/159	2	[11]
Books	105/441	3	[8]
Citeseer	3264/4536	6	[3]
Email-Eu-Core	1005/25571	42	[10]
Cora	23166/89157	70	[20]
Amazon	334863/925872	75149	[10]
YouTube	1134890/2987624	8385	[10]

Table 2. LFR networks

μ	**AD**	**MinCS**	**Nodes**	**Source**
[0.1..0.9]	5	50	1000	[17]
[0.1..0.9]	5	50	5000	[17]
[0.1..0.9]	5	50	10000	[17]
[0.1..0.9]	5	50	50000	[17]

In the first phase of the experiment, α-HC is compared to HC according to NMI and F1-score by considering various LFR networks. The figures 3, 4, 5, 6, 7, 8, 9 and 10 illustrates the obtained results. The histograms reveal that, for $\mu \geq 0.3$, there is always at least an $\alpha \neq 0$ (more precisely, $\alpha \geq 0.6$) for which α-HC outperforms HC in terms of NMI and F1-score. These findings suggest that α-HC is more reliable than HC for detecting mixed communities, as confirmed by the results of propositions 1 and 2.

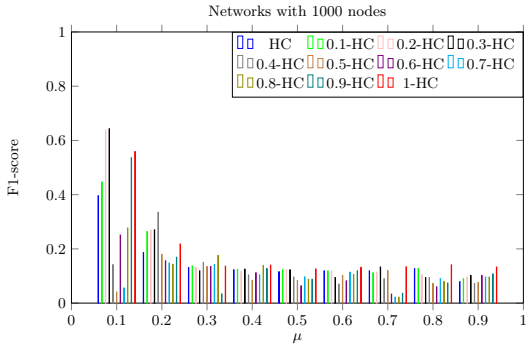


Fig. 3. HC vs. α-HC on LFR networks with 1000 nodes based on F1-score.

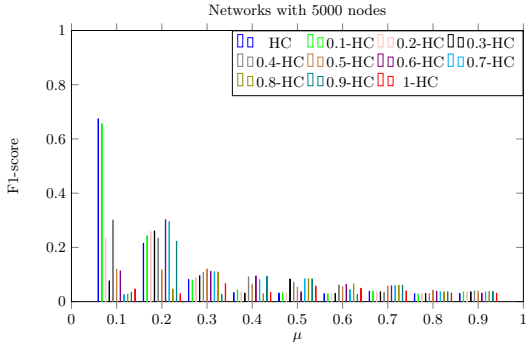


Fig. 4. HC vs. α-HC on LFR networks with 5000 nodes based on F1-score.

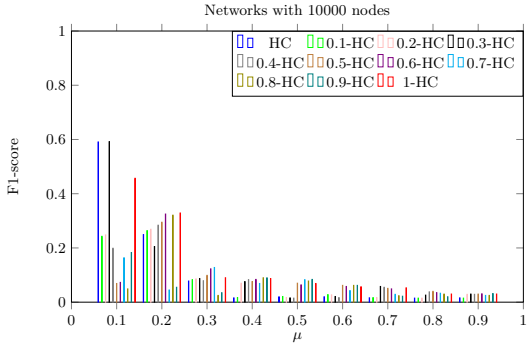


Fig. 5. HC vs. α-HC on LFR networks with 10000 nodes based on F1-score.

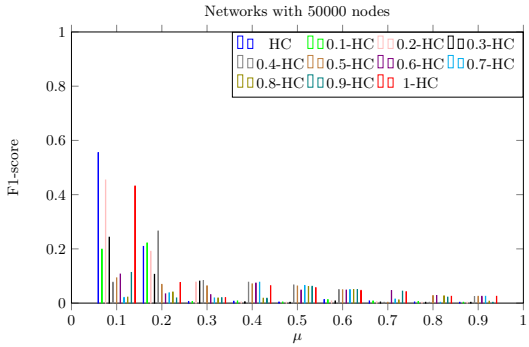


Fig. 6. HC vs. α-HC on LFR networks with 50000 nodes based on F1-score.

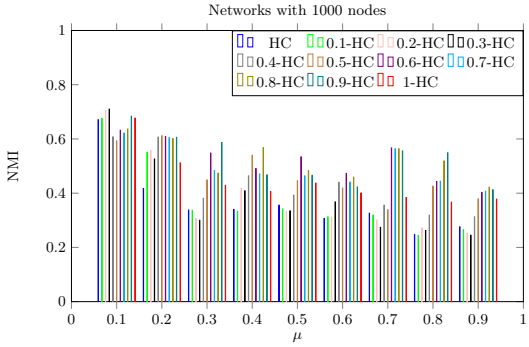


Fig. 7. HC vs. α-HC on LFR networks with 1000 nodes based on NMI.

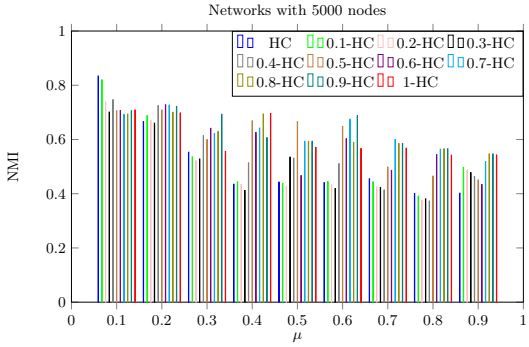


Fig. 8. HC vs. α-HC on LFR networks with 5000 nodes based on NMI.

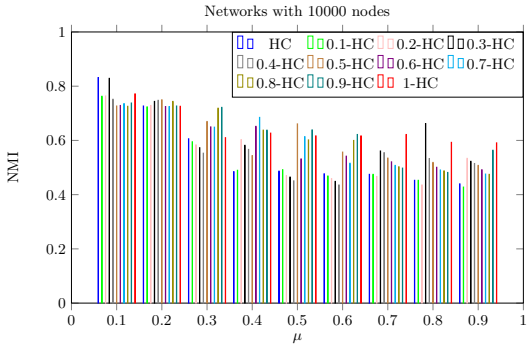


Fig. 9. HC vs. α-HC on LFR networks with 10000 nodes based on NMI.

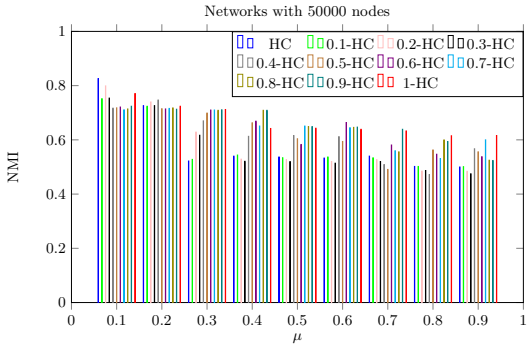


Fig. 10. HC vs. α-HC on LFR networks with 50000 nodes based on NMI.

In the rest of the experiment, α is fixed to 1. We made two comparisons. First, 1-HC with HC, then 1-HC with other some non-overlapping approaches mentioned in section 1 like (Label Propagation Algorithm (LPA) [16], LEIDEN algorithm [22], Surprise Communities (SC) [1], Significance Communities Approach (SCA) [21] and Markov Clustering algorithm (MC) [5]) on 8 well-known real-world datasets. Tables 3, 4, 5, and 6 illustrates the results.

Tables 3 and 4 show that 1-HC performs better than HC. Indeed, 1-HC outperforms HC allowing a gain of 4% according to the NMI. For the F1-score, an improvement of 4% is also obtained. The results shown in table 5 prove that 1-HC overpasses other state-of-the-art approaches considered in this paper. Indeed, 1-HC exceeds on average 13%, 14%, 14%, 9%, and 23% LPA, LEIDEN, SC, SCA, and MC respectively according to NMI. In table 6, 1-HC overpasses the above mentioned approaches. In fact, 1-HC exceeds on average 4%, 6%,

10%, 8%, and 13% LPA, LEIDEN, SC, SCA and MC respectively according to F1-score. Therefore, 1-HC shows good results for communities compared to the above-mentioned algorithms.

Despite the potentially fruitful results observed with 1-HC, there exist some datasets where its performance is limited. This can be explained by the variety of the structures of the networks. Furthermore, $\alpha = 1$ is not universally applicable. In fact it may be not the best optimal result. Then, it is important to search the most appropriate α value that aligns with the structure of the considered network.

Table 3. 1-HC vs. HC based on NMI

Comparison based on NMI		
Datasets	**HC**	**1-HC**
Karate	0.579	**0.777**
Email-Eu-Core	**0.705**	0.660
Citeseer	**0.098**	0.069
Dolphin	0.429	**0.509**
Books	**0.49**	0.474
Amazon	0.635	**0.752**
Cora	**0.623**	0.613
YouTube	0.127	**0.199**
Average	0.46	**0.50**

Table 4. 1-HC vs. HC based on F1

Comparison based on F1-score		
Datasets	**HC**	**1-HC**
Karate	0.669	**0.919**
Email-Eu-Core	0.164	**0.280**
Citeseer	**0.147**	0.099
Dolphin	0.470	**0.660**
Books	**0.606**	0.345
Amazon	**0.425**	0.415
Cora	0.085	**0.102**
YouTube	0.178	**0.287**
Average	0.34	**0.38**

Table 5. 1-HC vs. others based on NMI

Comparison based on NMI						
Datasets	**1-HC**	**LPA**	**LEIDEN**	**SC**	**SCA**	**MC**
Karate	**0.777**	0.207	0.202	0.220	0.462	0.164
Email-Eu-Core	**0.660**	0.180	0.593	0.648	0.671	0.428
Citeseer	0.069	0.087	**0.128**	0.102	0.093	0.080
Dolphin	**0.509**	0.436	0.098	0.120	0.164	0.090
Books	0.474	0.534	**0.573**	0.441	0.441	0.526
Amazon	**0.752**	0.579	0.206	0.539	0.586	0.601
Cora	**0.613**	0.551	0.472	0.552	0.584	0.337
YouTube	0.199	0.393	**0.616**	0.307	0.286	0.008
Average	**0.50**	0.37	0.36	0.36	0.41	0.27

Table 6. 1-HC vs. others based on F1

Comparison based on F1-score						
Datasets	**1-HC**	**LPA**	**LEIDEN**	**SC**	**SCA**	**MC**
Karate	**0.919**	0.630	0.560	0.490	0.490	0.735
Email-Eu-Core	0.280	0.065	0.217	0.075	**0.387**	0.159
Citeseer	0.099	0.074	**0.286**	0.081	0.073	0.046
Dolphin	**0.660**	0.585	0.480	0.360	0.256	0.320
Books	0.345	0.656	**0.776**	0.590	0.435	0.696
Amazon	**0.415**	0.397	0.028	0.318	0.370	0.084
Cora	0.102	0.221	0.238	**0.257**	0.255	0.032
YouTube	**0.287**	0.112	0.005	0.087	0.164	0.007
Average	**0.38**	0.34	0.32	0.28	0.30	0.25

5 Conclusion and future work

In this paper, we extended the HC method called α-HC based on α-2S to find disjoint communities. While using 2S in HC takes only into account the neighborhood, our approach improves such formula by taking into account the number of interactions between common neighbors. We proved theoretically that for a k-linked-cliques graph, identifying the two cliques using α-2S is better than using 2S. Experimentation evaluation showed that our approach surpasses the above-mentioned methods. In a future work, we plan to use our approach in the context of the Influence Maximization problem to find the seed nodes. Another direction

for future work is to develop an adaptive approach can predict the value of α which provides the best quality based on machine learning solutions.

References

1. Aldecoa, R., Marín, I.: Deciphering network community structure by surprise. PloS one (2011)
2. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment (2008)
3. Bollacker, K.D., Lawrence, S., Giles, C.L.: Citeseer: An autonomous web agent for automatic retrieval and identification of interesting publications. In: Proceedings of the second international conference on Autonomous agents. pp. 116–123 (1998)
4. Chen, Y.C., Zhu, W.Y., Peng, W.C., Lee, W.C., Lee, S.Y.: Cim: community-based influence maximization in social networks. ACM Transactions on Intelligent Systems and Technology (TIST) **5**(2), 1–31 (2014)
5. Enright, A.J., Van Dongen, S., Ouzounis, C.A.: An efficient algorithm for large-scale detection of protein families. Nucleic acids research **30**(7), 1575–1584 (2002)
6. Fortunato, S., Lancichinetti, A.: Community detection algorithms: a comparative analysis: invited presentation, extended abstract. In: 4th International ICST Conference on Performance Evaluation Methodologies and Tools (2010)
7. Ganley, D., Lampe, C.: The ties that bind: Social network principles in online communities. Decision support systems **47**(3), 266–274 (2009)
8. Krebs, V.: Books about us politics. unpublished, http://www. orgnet. com (2004)
9. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Physical Review E **78**(4) (oct 2008). `https://doi.org/10.1103/physreve.78.046110`
10. Leskovec, J., Krevl, A.: Snap datasets: Stanford large network dataset collection at http://snap. stanford. edu/data (2014)
11. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology **54**(4) (2003)
12. Luís Rita: Towards data science: Modularity maximization greedy algorithm (30 May 2020), `https://towardsdatascience.com/modularity-maximization-5cfa6495b286`
13. Martin Jambon: Theoretical computer science: Fast extraction of the edges of an induced subgraph (26 December 2015), `https://cstheory.stackexchange.com/questions/33440/fast-extraction-of-the-edges-of-an-induced-subgraph`
14. Martínez, V., Berzal, F., Cubero, J.C.: A survey of link prediction in complex networks. ACM Comput. Surv. **49**(4) (dec 2016), `https://doi.org/10.1145/3012704`
15. Newman, M.E.: Fast algorithm for detecting community structure in networks. Physical review E **69**(6), 066133 (2004)
16. Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. Physical review E (2007)
17. Rossetti, G., Milli, L., Cazabet, R.: Cdlib: a python library to extract, compare and evaluate communities from complex networks. Applied Network Science (2019)
18. Rossetti, G., Pappalardo, L., Rinzivillo, S.: A novel approach to evaluate community detection algorithms on ground truth. In: Complex networks. Springer (2016)
19. Salton, G., McGill, M.J.: Introduction to modern information retrieval (1983)

20. Šubelj, L., Bajec, M.: Model of complex networks based on citation dynamics. In: Proceedings of the 22nd international conference on World Wide Web (2013)
21. Traag, V.A., Krings, G., Van Dooren, P.: Significant scales in community structure. Scientific reports **3**(1), 1–10 (2013)
22. Traag, V.A., Waltman, L., Van Eck, N.J.: From louvain to leiden: guaranteeing well-connected communities. Scientific reports **9**(1), 1–12 (2019)
23. unknown: Cdp (2022), `https://github.com/2x254/CDP`
24. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of anthropological research **33**(4) (1977)

A Appendix A

We have $s_2(u_2, v_2) = \frac{k+2}{\sqrt{(n+2)(n+k)}}$
if $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_2}$, we can distinguish two cases:

- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_0}, \mathbf{v_0}\}| \neq \mathbf{1}$. In this case, we have $s_2(u_1, v_1) = 1$. Then, $\frac{k+2}{\sqrt{(n+2)(n+k)}} < 1 \Leftrightarrow (k+2)^2 < (n+2)(n+k) \Leftrightarrow k < (n-2) + \sqrt{(2-n)^2 + 4(n^2+2n-4)}$. k.t. $(n-2) + \sqrt{(2-n)^2 + 4(n^2+2n-4)} > n \implies s_2(u_2, v_2) < s_2(u_1, v_1)\ \forall\ 1 \leq k \leq n$.
- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_0}, \mathbf{v_0}\}| = \mathbf{1}$. In this case, we have $s_2(u_1, v_1) = \frac{n}{\sqrt{n(n+k)}}$. Then, $\frac{k+2}{\sqrt{(n+2)(n+k)}} < \frac{n}{\sqrt{n(n+k)}} \Leftrightarrow \frac{(k+2)^2}{(n+2)(n+k)} < \frac{n}{(n+k)} \Leftrightarrow (k+2)^2 < n(n+2) \Leftrightarrow k+2 < \sqrt{n(n+2)} \Leftrightarrow k < \sqrt{n(n+2)} - 2$. k.t. $\sqrt{n(n+2)} - 2 < n-1\ \forall\ n \geq 4 \implies \mathbf{s_2(u_2, v_2) < s_2(u_1, v_1)}$ **iff.** $\mathbf{k < n-1}$.

if $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_1}$, there are also two cases:

- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_2}, \mathbf{v_2}\}| = \mathbf{0}$ **or** $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_1} \cap \mathbf{E_3}$. In this case, $s_2(u_1, v_1) = 1$. Then, $s_2(u_2, v_2) < s_2(u_1, v_1)\ \forall\ 1 \leq k \leq n$. (proved).
- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_2}, \mathbf{v_2}\}| = \mathbf{1}$. In this case, $s_2(u_1, v_1) = \frac{n}{\sqrt{n(n+2)}}$. Then, $\frac{k+2}{\sqrt{(n+2)(n+k)}} < \frac{n}{\sqrt{n(n+2)}} \Leftrightarrow (k+2)^2 < n(n+k) \Leftrightarrow k < \frac{(n-4)+\sqrt{n(5n-8)}}{2} \Leftrightarrow k \leq n-1 < \frac{(n-4)+\sqrt{n(5n-8)}}{2} \Leftrightarrow s_2(u_2, v_2) < s_2(u_1, v_1)$ iff. $k \leq n-1$.

B Appendix B

$s_2^\alpha(u_2, v_2) = (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} + \alpha\frac{(k+2)(k+1)}{min(n(n-1)+4k+2,\ n(n-1)+(k+2)(k+1)-2)}$
It should be noted that $min(n(n-1)+6,\ n(n-1)+4) = n(n-1)+4$ iff $k = 1$ and $min(n(n-1)+4k+2,\ n(n-1)+(k+2)(k+1)-2) = n(n-1)+4k+2$ iff $k \geq 2$

if $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_2}$, we have the same cases as mentioned in the proof of 2S:

- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_0}, \mathbf{v_0}\}| \neq \mathbf{1}$. In this case, we have $s_2^\alpha(u_1, v_1) = 1$. Then,

- **if $\mathbf{k = 1}$:** k.t. $\frac{3}{\sqrt{(n+2)(n+1)}} < 1 \ \forall\ n \geq 4 \Leftrightarrow (1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} \leq (1-\alpha)\ \forall\ \alpha \in [0..1]$ and k.t. $\frac{6}{n(n-1)+4} < 1 \ \forall\ n \geq 4 \Leftrightarrow \alpha\frac{6}{n(n-1)+4} < \alpha \ \forall\ \alpha \in]0..1] \Leftrightarrow (1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} + \alpha\frac{6}{n(n-1)+4} < 1 \ \ \forall\ \alpha \in]0..1]$

- **if $\mathbf{k \geq 2}$:** k.t. $\frac{k+2}{\sqrt{(n+2)(n+k)}} < 1 \ \forall\ 1 \leq k \leq n$ (proved) $\Leftrightarrow (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} \leq (1-\alpha) \quad \forall\ 1 \leq k \leq n,\ \forall\ \alpha \in [0..1]$ and wtp $\frac{(k+2)(k+1)}{n(n-1)+4k+2} < 1$ iff. $k < n$. $\frac{(k+2)(k+1)}{n(n-1)+4k+2} < 1 \Leftrightarrow (k+2)(k+1) < n(n-1)+4k+2 \Leftrightarrow k^2+3k+2-4k < n^2-n+2 \Leftrightarrow k^2-k < n^2-n \Leftrightarrow k < n$. $\Leftrightarrow \alpha\frac{(k+2)(k+1)}{n(n-1)+4k+2} < \alpha$ iff. $k < n,\ \ \forall\ \alpha \in]0..1] \Leftrightarrow (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} + \alpha\frac{(k+2)(k+1)}{n(n-1)+4k+2} < 1$ iff $k < n,\ \ \forall\ \alpha \in]0..1]$.

– $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_0}, \mathbf{v_0}\}| = \mathbf{1}$. We have $s_2^{\alpha}(u_1, v_1) = (1-\alpha)\frac{n}{\sqrt{n(n+k)}} + \alpha$. Then,

- **if $\mathbf{k = 1}$:** k.t. $(1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} \leq (1-\alpha)\frac{n}{\sqrt{n(n+1)}} \ \ \forall\ \ n \geq 4,\ \forall\ \alpha \in [0..1]$ and $\alpha\frac{6}{n(n-1)+4} < \alpha \ \ \forall\ \ n \geq 4,\ \forall\ \alpha \in]0..1] \Leftrightarrow (1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} + \alpha\frac{6}{n(n-1)+4} < (1-\alpha)\frac{n}{\sqrt{n(n+1)}} + \alpha \quad \forall\ \alpha \in]0..1]$.

- **if $\mathbf{2 \leq k < n-1}$:** k.t. $\frac{k+2}{\sqrt{(n+2)(n+k)}} - \frac{n}{\sqrt{n(n+k)}} < 0$ iff. $k < n-1$ (proved) and $\frac{(k+2)(k+1)}{n(n-1)+4k+2} - 1 < 0$ iff. $k < n$. (proved) $\Leftrightarrow (1-\alpha)[\frac{k+2}{\sqrt{(n+2)(n+k)}} - \frac{n}{\sqrt{n(n+k)}}] \leq 0$ iff. $k < n-1,\ \forall\ \alpha \in [0..1]$ and $\alpha[\frac{(k+2)(k+1)}{n(n-1)+4k+2} - 1] < 0$ iff. $k < n,\ \forall\ \alpha \in]0..1] \Leftrightarrow (1-\alpha)[\frac{k+2}{\sqrt{(n+2)(n+k)}} - \frac{n}{\sqrt{n(n+k)}}] + \alpha[\frac{(k+2)(k+1)}{n(n-1)+4k+2} - 1] < 0$ iff. $k < n-1,\ \forall\ \alpha \in]0..1] \Leftrightarrow (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} + \alpha\frac{(k+2)(k+1)}{n(n-1)+4k+2} < (1-\alpha)\frac{n}{\sqrt{n(n+k)}} + \alpha$ iff. $k < n-1,\ \forall\ \alpha \in]0..1]$.

- **if $\mathbf{k = n-1}$:** let be $a = (1-\alpha)\frac{n+1}{\sqrt{(n+2)(2n-1)}} + \alpha\frac{(n+1)n}{(n+4)(n-1)+2}$ and $b = (1-\alpha)\frac{n}{\sqrt{n(2n-1)}} + \alpha \Leftrightarrow a - b = (1-\alpha)\frac{n+1}{\sqrt{(n+2)(2n-1)}} + \alpha\frac{(n+1)n}{(n+4)(n-1)+2} - (1-\alpha)\frac{n}{\sqrt{n(2n-1)}} - \alpha = [\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}}] - \alpha[\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}} + 1 - \frac{(n+1)n}{(n+4)(n-1)+2}]$
  $\Leftrightarrow a - b < 0$ iff. $\alpha > \frac{\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}}}{\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}} + 1 - \frac{(n+1)n}{(n+4)(n-1)+2}}$. Let's consider $f(n) = \frac{\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}}}{\frac{n+1}{\sqrt{(n+2)(2n-1)}} - \frac{n}{\sqrt{n(2n-1)}} + 1 - \frac{(n+1)n}{(n+4)(n-1)+2}}$ a continuous decreasing function on $[4, +\infty[$. Calculating the limits : $\lim\limits_{n \to +\infty} f(n) = 0$ and $\lim\limits_{n \to 4} f(n) \approx 0.06 \Leftrightarrow 0 \leq f(n) \leq 0.06$. Then, $a < b$ iff. $\alpha > 0.06$.

- **if $\mathbf{k = n}$:** let be $a = (1-\alpha)\sqrt{\frac{n+2}{2n}} + \alpha$ and $b = (1-\alpha)\frac{1}{\sqrt{2}} + \alpha$. Let's consider $g(n) = \sqrt{\frac{n+2}{2n}}$ a continuous and strictly positive decreasing function on $[4, +\infty[$. Calculating the limits : $\lim\limits_{n\to+\infty} g(n) = \dfrac{1}{\sqrt{2}}$ and $\lim\limits_{n\to 4} g(n) = \sqrt{\dfrac{3}{4}} \Leftrightarrow \dfrac{1}{\sqrt{2}} \leq g(n) \leq \sqrt{\dfrac{3}{4}} \Leftrightarrow g(n) \geq \dfrac{1}{\sqrt{2}} \Leftrightarrow (1-\alpha)\ g(n) + \alpha \geq (1-\alpha)\dfrac{1}{\sqrt{2}} + \alpha \Leftrightarrow a \geq b \Leftrightarrow$ if $k = n, s_2^\alpha(u_2, v_2) \geq s_2^\alpha(u_1, v_1)\ \forall\, \alpha \in [0..1]$.

$\implies \mathbf{s_2^\alpha(u_2, v_2) < s_2^\alpha(u_1, v_1)}$ **iff.** $\mathbf{k \leq n-1}$ **and** $\alpha > \mathbf{0.06}$.

if $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_1}$, there are also two cases, which are the same in the proof of 2S:

- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_2}, \mathbf{v_2}\}| = \mathbf{0}$ **or** $(\mathbf{u_1}, \mathbf{v_1}) \in \mathbf{E_1} \cap \mathbf{E_3}$. In this case, $s_2^\alpha(u_1, v_1) = 1$. Then, $s_2^\alpha(u_2, v_2) < s_2^\alpha(u_1, v_1)$ iff $k < n$ (proved).
- $|\{\mathbf{u_1}, \mathbf{v_1}\} \cap \{\mathbf{u_2}, \mathbf{v_2}\}| = \mathbf{1}$. In this case, $s_2^\alpha(u_1, v_1) = (1-\alpha)\frac{n}{\sqrt{n(n+2)}} + \alpha$. Then,
  - **if $\mathbf{k = 1}$:** k.t. $\frac{3}{\sqrt{(n+2)(n+1)}} < \frac{n}{\sqrt{n(n+2)}} \Leftrightarrow (1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} \leq (1-\alpha)\frac{n}{\sqrt{n(n+2)}}\ \forall\ \alpha \in [0..1],\ \forall\ n \geq 4$. k.t. $\frac{6}{n(n-1)+4} < 1 \Leftrightarrow \alpha\frac{6}{n(n-1)+4} < \alpha\ \forall\ \alpha \in\ ]0..1] \Leftrightarrow (1-\alpha)\frac{3}{\sqrt{(n+2)(n+1)}} + \alpha\frac{6}{n(n-1)+4} < (1-\alpha)\frac{n}{\sqrt{n(n+2)}} + \alpha\ \forall\ \alpha \in\ ]0..1],\ \forall\ n \geq 4$.
  - **if $\mathbf{k \geq 2}$:** k.t. $\frac{k+2}{\sqrt{(n+2)(n+k)}} < \frac{n}{\sqrt{n(n+2)}}$ iff $k \leq n-1$ (proved) $\Leftrightarrow (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} \leq (1-\alpha)\frac{n}{\sqrt{n(n+2)}}$ iff $k \leq n-1,\ \forall\ \alpha \in [0..1]$. k.t. $\frac{(k+2)(k+1)}{n(n-1)+4k+2} < 1$ iff $k < n$. (proved) $\Leftrightarrow \alpha\frac{(k+2)(k+1)}{n(n-1)+4k+2} < \alpha\ \forall\ \alpha \in\ ]0..1] \Leftrightarrow (1-\alpha)\frac{k+2}{\sqrt{(n+2)(n+k)}} + \alpha\frac{(k+2)(k+1)}{n(n-1)+4k+2} < (1-\alpha)\frac{n}{\sqrt{n(n+2)}} + \alpha$ iff $k \leq n-1\ \forall\ \alpha \in\ ]0..1] \Leftrightarrow s_2^\alpha(u_2, v_2) < s_2^\alpha(u_1, v_1)$ iff. $k \leq n-1$.