nature portfolio

Peer Review File

Local dominance unveils clusters in networks

This manuscript has been previously reviewed at another Nature Portfolio journal. This document only contains reviewer comments and rebuttal letters for versions considered at Communications Physics.

Reviewers' comments:

Reviewer #1 (Remarks to the Author):

The authors propose a community detection algorithm in networks, named Local Search (LS), that uses the notion of local dominance to identify community centers and local hierarchies based on local information, where lower degree nodes are assigned to the basin of influence of higher degree ones. The soundness of the method is demonstrated on synthetic benchmarks and on empirical networks with community labels. The numerical evaluation also includes networks derived from vector data, which is meaningful since the method is based on ideas of distance borrowed from clustering algorithms on vector data. Finally, LS is fast and claimed to be resistant to noise.

The problem of detecting communities in networks is important and without unique solution, and the relation between hierarchies, communities, community labels, and underlying metric spaces is interesting.

Response: We thank the reviewer for these positive comments.

However, it is not clear how the work is related with previous literature, how well it performs as compared with alternative strategies, and which are its limitations. The analysis is poor and needs more elaboration. I have some specific questions and comments.

My main concern is that it is not clear how the proposed method performs as compared with other strategies, especially as compared with gradient network strategies for community detection reminiscent of the current proposal. For instance, in IEEE Transactions on Knowledge and Data Engineering 29, 92-935 (2017) the authors propose a mapping by representing each network node with its geodesic distances from all other nodes and the space spanned by the geodesic distance vectors is the geodesic space of that network. In this space, a continuous density field is induced by each node that is drifted in the direction of positive density gradient till a local maximum using an iterative algorithm. The nodes that drift to the same region of space belong to the same communities in the original network. The analogies/differences of the proposed algorithm with this method need to be discussed in the manuscript, and quantitative comparisons of the two methods need to be performed.

Response: We thank the reviewer for this great suggestion that further spans our knowledge of community detection. After carefully reading the above paper, we realize that the GDG algorithm [1] is different from most other methods mentioned in our manuscript. It first embeds the network into a high dimensional vector space based on shortest path length between nodes, and then applies an iterative clustering algorithm, which is similar to mean-shift algorithm, to obtain partitions. It overcomes the drawbacks of applying hierarchical clustering to distance matrix [2] and obtains pretty good performances on networks. However, we also find that the computation cost of this algorithm is relatively high, for example, it requires calculating distances between all node pairs, which is costly in computation and infeasible for large scale networks both when considering time and storage space cost (e.g., the DBLP network).

We added some related discussions in the main text: “For example, the GDG (geodesic density gradient) algorithm [3] first embeds the network into vector space based on shortest path length between nodes and then applies an iterative clustering algorithm, which is similar to mean-shift algorithm, both of which are costly in computation, to obtain partitions of communities. The GDG algorithm achieves the best performance in two out of seven networks, and has an obvious advantage over other methods on Citeseer (see Table 2 and Supplementary Table 2 for more details).”,

We also added more discussions in the Supplementary Materials: “It is worth pointing out the analogies and differences between the GDG and LS algorithms: Most community detection

algorithms directly work on network data, while the GDG algorithm eventually works in a vector space, which is continuous and has metric properties (e.g., mutual distance, triangle inequality), and implicitly relies on the density of data points [3] in the embedded space; the LS algorithm works on network data, which is discretised and may easily violates metric properties (i.e., asymmetric interaction between nodes, and $l_{ik}+l_{kj}$ might be smaller than l_{ij}), with a focus on the hidden leader-follower relation revealed by local dominance. When applying the LS algorithm to clustering vector data, we establish a connection between the concept of density of data points in the vector space and degree of nodes in the network. This connection enhances our understanding of both clustering analysis and community detection, which have been long regarded as similar tasks, and better captures the underlying structure within the data."

	N	E	N_c	GDG			LS			Δt (ms)
				F1	N_c	t(ms)	F1	N_c	t(ms)	
Karate	34	78	2	**0.91**	3	40	0.83	**2**	**6**	34
Football	115	613	10	**0.60**	12	276	0.35	**6**	**20**	256
Polbooks	105	441	3	0.75	**3**	118	**0.80**	2	**8**	110
Polblogs	1,490	19,090	2	0.66	3	72,939	**0.69**	3	**212**	72,727
Cora	2,708	5,429	7	0.29	5	800,015	**0.33**	**7**	**139**	799,876
Citeseers	3,264	9,072	6	**0.63**	5	907,446	0.45	7	**131**	907,315
PubMed	19,717	44,327	3	0.19	529	87,238,940	**0.46**	**8**	**2,298**	87,236,642

Supplementary Table 2. Comparisons between the GDG (geodesic density gradient) algorithm [27] and LS on networks with ground truth community labels. As GDG requires calculating the shortest path between all node pairs to embed the network in a vector space, whose dimension equals the number of nodes, for large scale networks, a PCA is applied in GDG to reduce the dimension [27], and we select the first one thousand components as the embedding. GDG achieves the best performance in two out of seven networks when compared with a broad range of other popular algorithms (see Table 2 in the main text). LS is much more efficient in implementation, and has a better performance in four out of seven networks when compared with GDG.

	Karate	Football[49]	Polbooks	Polblogs	Cora	Citeseers	Pubmed	Time Complexity
Spin glass[54]	0.61	<u>0.92</u>	0.62	0.88	<u>0.33</u>	0.22	0.21	NA
GN[8]	0.59	0.84	**0.80**	0.74	0.32	0.23	0.18	$O(N^3)$
GDG[55]	**0.91**	0.60	0.75	0.66	0.29	**0.63**	0.19	$O(dtN^2)$
Walktrap[56]	0.51	0.88	<u>0.79</u>	0.88	0.29	0.14	0.16	$O(N^2 \log N)$
Spectral[23]	0.62	0.54	0.70	<u>0.89</u>	0.32	0.25	**0.46**	$O(N^2 \log N)$
Inferential[57,58]	0.65	0.87	0.77	0.32	0.31	0.40	0.07	$O(N^2 \log N)$
Fastgreedy[25]	0.75	0.56	0.78	<u>0.89</u>	**0.39**	0.28	<u>0.32</u>	$O(N \log N)$
Infomap[26]	0.76	**0.96**	0.69	0.80	0.07	0.04	0.01	$O(N \log N)$
LPA[30]	<u>0.88</u>	0.79	0.69	**0.91**	0.22	0.11	0.18	$\sim O(E)$
Louvain[9]	0.63	0.87	0.70	0.85	0.32	0.27	0.20	$\sim O(E)$
LS	0.83	0.35	**0.80**	0.69	<u>0.33</u>	<u>0.45</u>	**0.46**	$O(E)$

Table 2. Comparisons with classical community detection algorithms on real networks with ground-truth community labels. The algorithm with the highest F_1 score is highlighted in bold, and the second highest one is highlighted by underline. Overall, our LS algorithm have a pretty good performance, it is ranked first or second for five out of seven networks when compared to other popular algorithms, some of which are slower but more accurate ones. And our algorithm is the fastest one. For the GDG algorithm, d is the dimension of embedding space, and t is the number of iterations.

References:

[1] IEEE Transactions on Knowledge and Data Engineering, 29: 92-935 (2017)
[2] Finding and evaluating community structure in networks. Physical Review E, 69: 026113 (2014)
[3] IEEE Transactions on pattern analysis and machine intelligence 24, 1273–1280 (2002)

Also, the authors claim that LS outperforms the currently most widely used community detection method and, in Table 1, LS is compared with the Louvain algorithm. LS seems better overall. But, why only discussing the Louvain modularity optimization algorithm in the main text? Supp Tab 2 is more interesting and informative. From results there, it is clear that there is no a single detection method that beats the others in all real networks. In fact, LS is the best in only 3 out of 8 networks reported. In addition, the authors claim a good performance of the LS algorithm on vector data but this is not evident from the results reported in Table 2.

Response: We thank the reviewer for this great suggestion, indeed no method is expected to dominate all others given implicit dependence of the perfomance of algorithms on the properties of the data. What we can do, however, is to show that when LS is adapted, it clearly outperforms existing methods, and when it isn't in the top spot, it still produces very good results. We have moved the Table 2 in the main text, and added comparisons with GDG and an inferential method, and we also added more descriptions on the comparison results in the main text. "We chose to compare with the Louvain algorithm in Table 1 because both algorithms are linear, making them well-suited for large-scale networks and facilitating a more meaningful comparison. … In Table 2, we extend our comparison of the LS algorithm to include other algorithms with greater accuracy albeit slower in implementation. The LS algorithm takes the lead in two out of seven networks and secures second place in an additional two. GDG claims the top spot in two networks, while Fastgreedy leads in one network and secures second place in two others. Notably, other algorithms typically only attain the first position in one out of seven networks. This suggests that achieving optimal performance across all scenarios is highly improbable, aligning with the *No Free Lunch theorem*."

Overall, our algorithm works pretty well on heterogeneous networks, and we intended to put Football network as an illustration that on relatively homogeneous networks, nodes are too similar to each other, the detection of local leader is ineffective, which leads to underperformance. In the previous version, we put two Football networks in comparison, as we previously want to show that ground-truth label can affect the evaluation of classifications by other methods, in the revised version, we only keep one of the football network; so in the revised version, there are eventually seven networks, and our algorithm rank first or second in four out of seven.

As for clustering high-dimensional data, we changed the phrase of the first sentence of the subsection as "We now show the advantage of combining the ϵ-ball discretisation and community detection methods on clustering high-dimensional data sets."

We previously want to address the observation that the performance of our method have a great advantage over the DDB method [4], which is a state-of-the-art clustering algorithm, on high-dimensional datasets (such as MNIST handwritten figures and Olivetti face images) or equivalent (such as Wine), only in the other example (Irish flowers), the performance of the LS algorithm is lower than DDB. On most 2D benchmark data, we achieve a similar performance as of DDB, while on challenging 2D benchmark data (Fig. 6 E-G in the main text), the DDB method doesn't have the correct partition (see Fig. 6H) as of ours (see Fig. 6F). One improvement of DDB method via a costly learning strategy was published in Science Advances [5], and our LS algorithm can achieve the same performance as of the one proposed in Ref.[5], and the implementation efficiency of the LS algorithm is much higher.

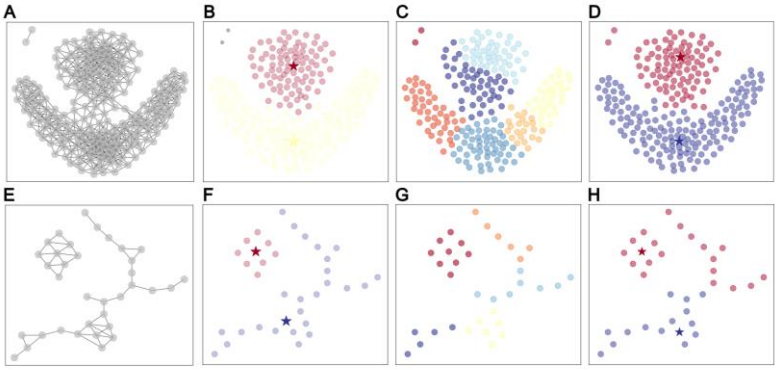
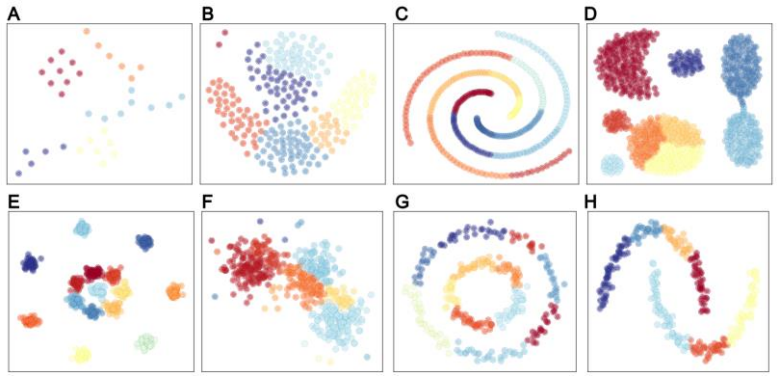


Figure 6. Comparisons on the clustering performance between the LS, Louvain, and DDB algorithms for two dimensional benchmark vector data. (**A**) and (**E**) represent networks constructed from vector data using the ε-ball method (see Supplementary Note 3.1 and Supplementary Fig. 10 for details on the network constructions). (**B**) and (**F**) show the result of the LS method that correctly identify clusters that align with common consensus (see Supplementary Fig. 11 for more cases). In addition, LS can detect noisy points (marked in grey) that are of low degrees but long l_i. (**C**) and (**E**) show the partitions obtained from the Louvain method which are more fragmented than the LS results (see Supplementary Fig. 13 for more cases). (**D** and **H**) show the results obtained from the DDB method which provides correct partitions to most benchmark data, see (**D**) and Ref.[16] for other cases, but fails in the test case in (**E**), where both a low density manifold and a high density cluster exist, due to its local association rule[84] being affected by a mixture of local and global metrics. LS and Louvain methods are performed on the constructed networks shown in **A** and **E**, and the DDB algorithm is performed on the original vector data.



Supplementary Figure 13. Clustering results obtained by the Louvain algorithm on networks constructed from vector data. The Louvain algorithm works on the same network constructed from vector data as of the LS method. Node color indicates clustering partitions.

Although the Louvain algorithm has the highest F1 score on Olivetti and MNIST datasets, it cannot yield good partitions in two dimensional benchmark data (see Fig. 6 E-G in the main text and comparisons between Supplementary Fig. 11 for LS algorithm and Supplementary Fig. 13 for the Louvain algorithm, which are applied on the same constructed networks as of the LS algorithm). So, overall, the LS algorithm still has a good performance on clustering data clouds.

In addition, the better performance of the Louvain algorithm on MNIST and Olivetti lies in some subtle differences from clustering results obtained by the LS method (see comparisons between Supplementary Fig. 19A and Supplementary Fig. 19B for the Olivetti dataset with 100 images. The Louvain algorithm detects each image of the first person as a separate cluster, but the LS method classifies them as noisy data and doesn't give a partition label).

In the Discussion section, we also elaborate “For low-dimensional data, we find LS algorithm provides the expected clusters and outperforms Louvain modularity optimisation algorithm ran on the discretised data, which generally yields too many communities and performs poorly.” “LS also outperforms DDB, a state-of-the-art unsupervised clustering method, on some challenging cases in the presence of low-density manifolds.” “For high-dimensional data, LS still outperforms DDB, but not Louvain, although on closer inspection, Louvain obtains a better F_1-score, but suffers again from providing too many communities, outbalancing the advantage in F_1-score.”

References:

[4] Science, 2014, 344(6191):1492-6.

[5] Science Advances, 2019, 5(10):eaax3770.

Regarding community labels, in general they are not necessarily related with network structure, even if said to define ground-truth communities. This point needs to be discussed somewhere in the text.
Response: We thank the reviewer for this great suggestion. We elaborate on this point in the main text as “The choice(s) of the ground truth(s) are crucial and there might be ``alternative'' ground truths that emerge from unsupervised clustering analysis and are validated a posteriori. The notion of alignment between ground truth and structure is indeed crucial to obtain good clusters [6]. For example, in the well known Zachary Karate Club network, the metadata of nodes can also be their gender, age, major, ethnicity, however, most of which are irrelevant to the community structure when interested in understanding the split of the club, but might be relevant to understand other type of community structure.

Apart from evaluations based on ground-truth labels, various evaluation criteria purely based on network structure (e.g., optimizing modularity, conductance, cut) have been proposed, however, they may deviate from the real generating process of networks and will not be suitable for all scenarios. For example, maximizing modularity cannot generate good partitions in ecological networks, as herbivores in the same community will not prey on each other, thus there are no dense connections within the same ecological community. In this sense, if there can be some ground-truth labels, using F1 score is a more objective evaluation.

Reference:
[6] IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(4):1663-72.

Other more technical comments:

In Table I and II, the quality of the clusters is only measured as a function of the number of communities and the overlap with labels, F1-score. Results for other quality measures need to be supplemented. For instance, in other publications, it is has been seen that when the network contains well-separated non-overlapping communities, conductance may be a better scoring function, see Knowledge and Information Systems volume 42, pages 181-213 (2015).
Response: Thank you for your great suggestion. We added a subsection in the Supplementary Materials to introduce the conductance, and also made comparisons between algorithms on conductance (see Supplementary Table 3 in the revised version), and we find that our algorithm still outperforms other algorithms under this criterion, the LS algorithm achieves the lowest conductance in four out of seven networks.

	Louvain	GDG	LS
Karate	0.29	0.43	**0.21**
Football	**0.28**	0.60	0.29
Polbooks	0.24	0.21	**0.06**
Polblogs	**0.33**	0.48	0.41
Cora	0.14	0.42	**0.11**
Citeseers	0.08	0.24	**0.03**
PubMed	**0.14**	0.95	0.22

Supplementary Table 3. Comparisons between the parition performance of GDG (geodesic density gradient) [27], Louvain , and LS algorithms on networks when evaluated by conductance. The LS algorithm is of the smallest conductance in four out of seven networks.

As illustrated in Fig 1, the identification of local leaders based on local dominance by creating a forest of DAGs requires to visit nodes in a certain order, and results can change depending on the

order. Some experiments need to be performed using networks with a diversity of topological features to show how this can impact the behavior of LS.

Response: We thank the reviewer for this great advice. We did not explicitly comment on this in the original draft, but the algorithm itself is almost not affected by the order in which the nodes are visited. The order in which the algorithm visits the nodes is mainly controlled by the random intial seed, and we show the results in the Table below for ten implementations with different random seeds (label as 0~9) for the value of F1 score. From Table 1, the standard deviation of the results with different random seeds (i.e., different access sequences) is almost zero and the results are almost identical in most cases. This indicates that our algorithm is not strongly affected by different traversal orders.

	Implementations										μ	σ
	0	1	2	3	4	5	6	7	8	9		
Karate	0.832	0.832	0.832	0.832	0.832	0.832	0.832	0.832	0.832	0.832	0.832	1.17×10^{-16}
Polbooks	0.796	0.796	0.796	0.796	0.796	0.796	0.796	0.796	0.796	0.796	0.796	0
Polblogs	0.692	0.692	0.692	0.692	0.692	0.692	0.692	0.692	0.692	0.692	0.692	1.17×10^{-16}
Cora	0.325	0.325	0.325	0.326	0.325	0.326	0.325	0.326	0.325	0.326	0.325	4.67×10^{-4}
Citeseer	0.455	0.454	0.455	0.454	0.454	0.455	0.454	0.454	0.455	0.455	0.454	3.89×10^{-4}
PubMed	0.456	0.455	0.454	0.454	0.455	0.454	0.455	0.454	0.455	0.456	0.455	5.96×10^{-4}
Football	0.347	0.346	0.339	0.377	0.333	0.306	0.359	0.361	0.36	0.368	0.35	2.04×10^{-2}

Table 1 The F1-score of LS algorithm on seven datasets with multiple implementations. Each implementation, which is denoted by a digit from 0 to 9, is of a different random seed. μ and σ represent the average and standard deviations of F1-scores in ten implementations.

The experiments with synthetic networks for the detection of multiscale community structure is restricted to the Ravasz-Barabasi network model (or even simpler models), which is very far from capturing the multiscale organization of real networks, so it cannot be claimed that LS works on detecting multiscale structure as a general statement.

Response: We thank the reviewer for this great observation. It would be better and we would love to illustrate the results of detecting multi-scale communities structure on real networks, however, we didn't find networks with multi-scale ground truth labels to facilitate such a study. So, we make experiments on multiscale community structure generated by SBM models, which is a common practice on studying multi-scale communities structure (see Fig. 3 in the main text). When the degree distribution of networks is relatively heterogeneous, which is closer to real networks, we can identify the first level and second level community centers via obvious gaps. Between the four first-level centers (which are denoted as a2, d1, c1, b1, respectively) and others, there is a first obvious gap, and then there is a second obvious gap that separates all second level centers with other follower nodes.

Reviewer #2 (Remarks to the Author):

*In this manuscript, Shang et al propose a **novel** algorithm for community detection based on the identification of centers/leaders of each cluster. They validate their algorithm with synthetic data and then they apply their algorithm to real data and to the clustering of high-dimensional vector data.*

I think that the manuscript is well-written.
Response: We thank the reviewer for these positive comments.

Unfortunately, I do not think that this manuscript is suitable for publication in Nature Communications as I discuss in what follows.

It has already been 20 years since network scientists started looking for communities in networks. This means that in 20 years we have learnt a lot of things about this particular problem. One of the things that we have learnt is that as time goes by, we will probably be able to find a new heuristic that works better than others in identifying communities of a specific network (see for instance Peel et al. in https://www.science.org/doi/10.1126/sciadv.1602548). In practical, terms this does not mean that there could be methods that are superior to others, in general, to perform a certain task (community detection in this case) (see for instance Guimera in https://www.science.org/doi/10.1126/sciadv.1602548 and Peixoto in https://arxiv.org/abs/2210.09186 for a related discussion). Finding a 'better' method is expected and not surprising, but this does not mean that it is a big advance with respect to the state of the art. The fact that the method is easy to explain and then is fast are not reasons good enough to justify another method for this same problem. I think that a few years back, this would have been a great advance to the state of the art, but right now, I do not think this is the case anymore.
Response: We thank the reviewer for these great suggestions and thanks for recognizing that this method would have been a great advance a few years ago. We agree that there will be no algorithm superior to all other methods in all cases, which is proved by the "no free lunch theorem" by Peel et al., and we have cited suggested references to further back up such an idea. After making comparisons with more algorithms, it is clear that most algorithms can only score the first place in some network, and we added following remarks in the revised version:
"We note that the best performing algorithms are well distributed among the benchmarks, which reflects that real networks generally have different generating mechanisms that are better captured by some algorithms than others \cite{rosvall2008maps,peel2017ground}. This suggests that achieving optimal performance across all scenarios is highly improbable \cite{peixoto2023implicit}, aligning with the No Free Lunch theorem \cite{peel2017ground}."

We do believe that our method still provides an advance in the community and cluster detections field. Apart from being easy to explain and fast, our algorithm also provides connections between community detection and clustering analysis, which are regarded as similar tasks but lacks a unified understanding and connection. Different from algorithms rely optimization, spreading or random walk dynamics, information compression, spectral analysis and inferences, which all require the global information of the whole network, our algorithm provides a new perspective on tackling community detection with local information. After detecting local leaders, which only require information of neighbors, the local BFS for local leaders can be performed without traversing the whole network. With the concept of community center, which was overlooked over the past two decades, and knowledge from network science, we can further improve performance on traditional clustering analysis. We assume that our works further prove the power of network science. In addition, the identification of local leaders can be of broad applications, for example, identifying the seminal works

in the citation networks, and better study the splitting and merging dynamics of communities and hierarchical structure with local leaders. Most importantly, our algorithm is based on different mechanisms to capture the notion of community network. As such, our analysis proves that it provides additional insight in empirical systems and could become a useful addition to the network toolbox available to data scientists.

In particular, what concerns me the most is that the authors are ignoring developments in the analysis of the structure of complex networks that in my mind have qualitatively changed our ability to draw conclusions from network data in a principled way. I think that network inference methods are objectively a principled approach (as opposed to heuristic or descriptive - see https://arxiv.org/abs/2112.00183 for a lengthy discussion) to solve the community detection problem. These methods can explain our inability to detect communities in the very sparse limit (Decelle et al https://arxiv.org/abs/1109.3041), can help us eliminate the 'resolution limit' and identify the hierarchical organization of complex networks (Peixoto https://journals.aps.org/prx/abstract/10.1103/PhysRevX.4.011047) and are amenable to many different kinds of networks (weighted - Aicher et al. https://arxiv.org/abs/1404.0431 , directed - Larremore et al. https://journals.aps.org/pre/abstract/10.1103/PhysRevE.90.012805 , or hypergraphs https://www.nature.com/articles/s41467-022-34714-7 to name a few). All of this is possible with clearly made assumptions (for instance relying on stochastic block models as generative models, but we could use others) and removing arbitrariness from the process. In the manuscript the authors overlook completely these approaches, which are also free from the caveats that heuristic community detection methods suffer from (for instance, returning a set of communities for graphs that have none - an inference approach would return 1 community).

Response: We thank the reviewer for these great suggestions on community detection along the direction of statistical inference, which has indeed achieved great progress in recent years. We would like to emphasise that our approach is quite distinct from this line of work, as it does not explicitly search for communities in terms of a model to infer. Nonetheless, for the sake of completeness, we have referenced related papers as exemplified methods of inferential algorithms, and added the following paragraph in the main text:

"Another important type of method is inferential ones, which provides powerful tools without arbitrariness and further advances our understanding of community structure of networks \cite{peixoto2023descriptive}. Inferential algorithms, which generally relies on SBM as generative models, can explain the inability to detect communities in the very sparse limit \cite{decelle2011asymptotic}, help eliminate the resolution limit in Louvain algorithm and detect hierarchical structure of complex networks \cite{peixoto2014hierarchical}. Inferential methods can be adapted to different types of networks ranging from weighted networks \cite{aicher2015learning} to directed ones \cite{larremore2014efficiently} and hypergraphs \cite{contisciani2022inference}. However, inferential methods are generally computationally expensive \cite{peixoto2019bayesian,peixoto2014hierarchical}. The inferential algorithm described in Refs. \cite{peixoto2019bayesian,peixoto2014hierarchical} has a much better performance than the LS on Football network (see Supplementary Table 3), whose degree distribution is quite homogeneous."

In addition, our LS algorithm is less susceptible to caveats that most heuristic community detection methods suffer from. For example, for a lattice network, our LS algorithm will also return just 1 community (see Fig. 2A in the main text) and for ER networks, our algorithm returns a way smaller number of communities (as we commonly assume that ER networks are of no community structure in the thermodynamic limit and thus used as the null model, see Fig. 2B in the main text) than the Louvain algorithm that optimizes modularity (see Fig. 2E). And our algorithm doesn't have

the problem of resolution limit, as the LS method only utilizes local information, which is not affected by the size of the whole network.

I understand that the authors might think that it is not necessary to use inferential approaches for community detection and they are right. The problem is not the method itself, but what we use these communities for at a later stage: the conclusions I draw cannot depend on the arbitrary decisions I am making in defining communities and I should be able to explain how the assumptions affect the outcome. Unfortunately, the heuristic methods the authors propose have a number of arbitrary choices that are hard to justify and have an impact on the outcome of the method in unknown ways. First, they assume that communities have some central authority around which they are 'built'. This feels like a very bold unjustified assumption to make. The fact that network scientists have been looking for hubs for a long time does not mean that this is the right assumption to make (we could have been wrong all along!). Also, there are tons of other centrality metrics the authors could have used for the same purpose (why didn't the authors consider those? This may seem like a very naive comment, but I am using it to illustrate another 'arbitrary' choice). Finally, the authors at some point use visual inspection to decide where to 'cut the hierarchical tree' they obtain from their algorithm into communities. That is to say, this approach also suffers from the same problem you find when you use hierarchical clustering to find clusters of vector data: you do not have robust criterion of where to cut the tree. This last issue reflects the fact that because of the heuristic approach it is not possible to make a principled criterion to remove arbitrariness.

Response: We thank the reviewer for these comments. First, assuming that communities have some central authority (i.e., community centers) is at least plausible. For example, for the Karate Club, the president and the instructor are certainly two centers of communities. When the two leaders split, the network also split. And the community center is a common concept in the context of social networks (e.g., opinion leaders; the instructor and the president in the Zachary Karate Club network) [1,2], technology networks (e.g., hub nodes on the Internet), and geographical networks (where regions are often organised around central cities or physical entities, which is summarised as the famous ``central place theory by German geographer Walter Christaller). Equivalently, using SBM as the generative model of real networks is also an assumption, which may deviate from the real generative process of some networks, and we can't rule out other assumptions/hypotheses.

Second, in our LS algorithm, the reason why we choose the degree is mainly for the speed, and we can certainly choose other centrality measures. The seemingly arbitrariness may bring more flexibility to the algorithm. For example, changing to different centrality measures can reflect different focus or criterion on evaluating the importance of nodes when we try to find communities.

Third, for determining the number of communities, this can be better approached by more sophisticated algorithms, for example, Bayesian Change Point Detection (BOCPD) [3] that involves using a probabilistic model to represent the data and updating beliefs about change points as new observations arrive. Though BOCPD has some hyperparameters, they are often less sensitive to tuning.

Finally, our algorithm makes the notion of community centre explicit and well-defined, but there exist other types of algorithms, which implicitly search to assign higher-degree/central nods to different communities. For instance, it is known that modularity optimisation has a tendency for separating hubs due to its (configuration) null model.

References:

[1] Valente TW, Pumpuang P. Identifying opinion leaders to promote behavior change. Health education & behavior. 2007, 34(6):881-96.

[2] Burt RS. The social capital of opinion leaders. The Annals of the American Academy of Political and Social Science. 1999, 566(1):37-54.
[3] Adams RP, MacKay DJ. Bayesian online changepoint detection. 2007, arXiv preprint arXiv:0710.3742.

In my mind this is a very fast method, useful for some applications, and I am sure could be well received for people wanting to find 'clusters.' However, I am afraid it is not at the frontier of network science. In my wishlist, I would like that the authors, in a new version of the manuscript, try to make a fair comparison of more stat-of-the art inferential methods and make a better case of when one should consider using their approach as opposed to an inferential one (in case in which both give similar answers, which ones are those? When is the proposed method finding undesirable solutions - i.e when is assuming the existence of central nodes a very bad assumption?). The Louvain method is not at the frontier for knowledge in network science anymore.

Response: We thank the reviewer for these great suggestions and for recognizing the practicality of our method. In the revised version, we make detailed comparisons between the inferential method introduced in Refs. [4,5], which is representative of inferential methods and is well implemented and optimized, and the LS algorithm (see Supplementary Table 3 in the revised version). In addition, we also compare this inferential algorithm with other popular community detection algorithms on real networks (see Table 2 in the main text).

We find that when the degree distribution of the network is relatively homogeneous (e.g., Football network and networks generated by standard SBM), using inferential method is generally better; When the degree distribution is heterogeneous, the LS algorithm is usually better (see the Table below). As the concept of local leader is plausible in a heterogeneous setting, when the network is homogeneous, there would be no dominant leaders at all (e.g., ER network is of no obvious local leaders).

	N	E	Nc	Inferential		LS	
				F1	Nc	F1	Nc
Karate	34	78	2	0.653	1	**0.83**	**2**
Football	115	613	10	**0.865**	**11**	0.35	6
Polbooks	105	441	3	0.771	**3**	**0.80**	2
Polblogs	1,490	19,090	2	0.324	22	**0.69**	**3**
Cora	2,708	5,429	7	0.309	20	**0.33**	**7**
Citeseers	3,264	9,072	6	0.399	10	**0.45**	**7**
PubMed	19,717	44,327	3	0.074	53	**0.46**	**8**

Supplementary Table 3. Comparisons between an inferential algorithm [28, 29] and LS on networks with ground truth community labels.

And we also revised Table 2 in the main text to incorporate the comparisons with this inferential method.

	Karate	Football[59]	Polbooks	Polblogs	Cora	Citeseers	Pubmed	Time Complexity
Spin glass[64]	0.61	0.92	0.62	0.88	0.33	0.22	0.21	NA
GN[8]	0.59	0.84	**0.80**	0.74	0.32	0.23	0.18	$O(N^3)$
GDG[48]	**0.91**	0.60	0.75	0.66	0.29	**0.63**	0.19	$O(dtN^2)$
Walktrap[65]	0.51	0.88	0.79	0.88	0.29	0.14	0.16	$O(N^2 \log N)$
Spectral[23]	0.62	0.54	0.70	0.89	0.32	0.25	**0.46**	$O(N^2 \log N)$
Inferential[52,56]	0.65	0.87	0.77	0.32	0.31	0.40	0.07	$O(N^2 \log N)$
Fastgreedy[25]	0.75	0.56	0.78	0.89	**0.39**	0.28	0.32	$O(N \log N)$
Infomap[26]	0.76	**0.96**	0.69	0.80	0.07	0.04	0.01	$O(N \log N)$
LPA[30]	0.88	0.79	0.69	**0.91**	0.22	0.11	0.18	$\sim O(E)$
Louvain[9]	0.63	0.87	0.70	0.85	0.32	0.27	0.20	$\sim O(E)$
LS	0.83	0.35	**0.80**	0.69	0.33	0.45	**0.46**	$O(E)$

Table 2. Comparisons with classical community detection algorithms on real networks with ground-truth community labels. The algorithm with the highest F_1 score is highlighted in bold, and the second highest one is highlighted by underline. Overall, our LS algorithm have a pretty good performance, it is ranked first or second for five out of seven networks when compared to other popular algorithms, some of which are slower but more accurate ones. And our algorithm is the fastest one. For the GDG algorithm, d is the dimension of embedding space, and t is the number of iterations.

References:

[4] Peixoto, T. P. Bayesian stochastic blockmodeling. In Advances in network clustering and blockmodeling, vol. 11, 289–332 (Wiley Online Library, 2019)

[5] Peixoto, T. P. Hierarchical block structures and high-resolution model selection in large networks. Phys. Rev. X 4, 011047 (2014).

Reviewers' comments:

Reviewer #1 (Remarks to the Author):

First of all, I would like to thank the authors for their work which I think has significantly improved the manuscript.

The comparisons with other methods are valuable and give a better context to the results shown in the manuscript.

That being said, I do not think that the manuscript is a good fit for Communications Physics. The reason for this is that I do not think that the method proposed in the manuscript significantly advances our ability to extract knowledge from networks that we did not have before. The authors present an arguably fast heuristic method that performs at most comparably well to other methods (a few times better, the rest of the times worse). However, the results have not convinced me that this is a significant advance with respect to previous methods other than speed.

The clustering application is interesting, the authors essentially provide a networks based approach to k-means by looking at the local authority instead of the centroid of a subset of the data. In here results are a tad confusing since that authors highlight both F1 and smaller number of clusters.

I think that the problem in here is that because this is an heuristic method, there is no 'intrinsic' way of estimating the balance between goodness of the model and the number of parameters in the model (i.e number of clusters). Again, the results, while of technical merit, they are, in my opinion, not above the bar for Communications Physics.

As other comments related to this version of the manuscript and answer to reviewers:

1. I think that the question about using labels as ground truth is not well addressed. Using labels as ground truth has caveat (which have already been pointed out in the literature) that we should not take lightly. For real networks, there is at least one other way to test whether the algorithm identifies an good underlying network structure such as link prediction which some authors have used recently. It would be nice to address these issues in this type of works.

2. The question about how the different levels in multi-scale networks are selected form a previous report has not been addressed (or at least it was not clear to me).

3. It would be nice for the authors to explain when comparing to other methods what parameters they used. This is necessary for reproducibility of the results. For instance in the inferential method they use

(Peixoto PRX 2014), do they use a hierarchical prior, degree corrected or non-degree corrected SBM? How do they select the best model. In heuristic models, how many times do they run each algorithm? Details are also important when comparing with other methods. Same for the DBB clustering method they consider as a reference.

Reviewer #2 (Remarks to the Author):

I have reviewed the revised version of the manuscript proposing the Local Search community detection algorithm, which defines communities as basins of influence of high degree nodes serving as local central authorities. The manuscript has been extensively revised in response to my comments and those of the other referee. The proposed definition of community may be useful in specific practical situations and is easily generalizable to alternative definitions of influence. Furthermore, the speed of the method is a significant advantage when dealing with very large heterogeneous networks, say of millions of nodes, for which other methods with complexity scaling as N^2 or above would be impractical. Considering that there is no "the Method" for community detection in networks, an additional method offering advantages in some situations can be valuable for the interested community.

In my opinion, the manuscript is suitable for publication in Communications Physics. I have one additional suggestion. It would be useful to include a table in the main text wit basic statistics of the evaluated real networks, such as size, average degree, and maybe the rest of the metrics reported in SI Table 1, and also the level of heterogeneity in the degree distribution, assortativity, and clustering coefficient.

Reviewer #3 (Remarks to the Author):

In their manuscript, the authors develop an efficient community detection algorithm named Local Search (LS), which leverages the innovative concept of local dominance to identify community centers using local information. This methodology effectively uncovers local hierarchies within networks, setting it apart from traditional approaches by eliminating the need for heuristic optimization of a global objective function. Their work validates the LS algorithm's performance on synthetic benchmarks and empirical networks with known community structures, highlighting its efficiency, lower susceptibility to noise, and its comparative advantage over classical clustering methods in certain cases.

After careful consideration of the manuscript, the additional reports provided, and the authors' responses to the reviewers' comments from the previous round of revision, I find myself in alignment with the perspective of Reviewer #2. The critiques presented by Reviewer #1, while certainly valid and emphasizing the manuscript's theoretical limitations in comparison to inferential methods like those in Peixoto's work, do not diminish the revised paper's contribution to the advancement of community detection in networks. This contribution is particularly valuable for practitioners like myself, who prioritize methods that are swift, interpretable, and versatile. The potential impact of this work on the scientific community, especially for researchers and practitioners seeking innovative approaches to clustering and community detection, underscores the merit of its publication in Communications Physics.

Building on this foundation, I would like to offer a few targeted suggestions to further refine the manuscript. These comments are intended to enhance the paper's clarity and impact, addressing aspects ranging from overarching conceptual concerns to specific technical details:

1. Restrictions on graphs: At the beginning of the paper, it is very difficult to understand what type of graphs can be analyzed with Local Search (LS). Figure 1 suggests that only simple graphs can be considered, which is almost confirmed on page 14 since “We start with an undirected network with N nodes and E edges, for example, see Fig. 1A.” The limitation regarding self-loops is not explicitly mentioned, but the algorithm seems to consider that V(u), the set of adjacent vertices of u, cannot contain u itself. The code provided is not clear either, since the authors use nx.Graph objects, which, in principle, can contain self-loops. Please clarify the restrictions as soon as possible in the paper.

2. Directed graphs: The discussion concludes by talking about the possibility of applying LS to weighted graphs. I think the case of directed graphs should also be mentioned, since this problem is not as straightforward. One possibility is to define two types of leaders: the "integrators" (high in-degree) and the "influencers" (high out-degree), leading to two types of clustering. However, the authors may have already investigated this problem of great interest, as most networks are expected to be weighted and directed (as in neuroscience).

3. Caption of Fig 6: I think that "E" should be replaced by "G" in "(C) and (E) show the partitions obtained from the Louvain method."

4. Caption of Fig 6: "(D and H) show the results" should be "(D) and (H) show the results."

5. Page 2, just before "where $V(u)$ is the set of neighboring nodes of u: Curly brackets are missing in the expression for k_ν.

6. Reading the Author contributions, I feel that R. Li should be considered for the first author position. The authors might want to consider a different ordering.

7. Statements like "LS algorithm is linear" could be more precise by saying "LS algorithm is linear in the number of edges."

8. p7: becomes more noticeable -> becomes more noticeable.

9. p7: comparisons between algorithm -> comparisons between algorithms.

10. p7: algorithms, which generally relies -> algorithms, which generally rely.

11. p8: LS on Football network -> LS on the Football network

12. p8: remove comma in "While, our LS algorithm".

13. p8: replace "are" by "is" in "The choice(s) of the ground truth(s) are".

14. p13: "the LS method is not depend on the structure". Correct to "the LS method does not depend on the structure..."

15. p13: "every object are in a direct " -> "every object is in a direct".

16. p13 "optimization algorithm that utilize" -> "optimization algorithms that utilize".

17. p1-15 Consistency with the use of "centre" versus "center" depending on whether British or American English is being used.

Reviewers' comments:

Reviewer #1 (Remarks to the Author):

First of all, I would like to thank the authors for their work which I think has significantly improved the manuscript.
The comparisons with other methods are valuable and give a better context to the results shown in the manuscript.

Response: We thank the reviewer for these positive comments.

That being said, I do not think that the manuscript is a good fit for Communications Physics. The reason for this is that I do not think that the method proposed in the manuscript significantly advances our ability to extract knowledge from networks that we did not have before. The authors present an arguably fast heuristic method that performs at most comparably well to other methods (a few times better, the rest of the times worse). However, the results have not convinced me that this is a significant advance with respect to previous methods other than speed.

Response: We thank the reviewer for this comment. First of all, Peel et al.'s work (2019, Science Advances, e1602548) on "no free lunch" theorem proves that there will be no algorithm outperforms other algorithms in all situations. In this work, we show that our LS algorithm works pretty well on heterogeneous networks, which ranks first in 2 out of 7 networks and ranks second in 2 out of 7 when compared to other popular methods, and no other methods is overall better than ours (GDG ranks first in 2 out of 7, other methods at most rank first in 1 out of 7 or at most rank second in 2 out of 7, please see Table 3 in the revised main text). On homogeneous networks, such as Football and ER networks, where degree heterogeneity is too weak to facilitate meaningful local dominance, our algorithm does not work well. But, again, we want to point out that heterogeneous networks are more common in real world, and our algorithm would have broad applications.

The clustering application is interesting, the authors essentially provide a networks based approach to k-means by looking at the local authority instead of the centroid of a subset of the data. In here results are a tad confusing since that authors highlight both F1 and smaller number of clusters.
I think that the problem in here is that because this is an heuristic method, there is no 'intrinsic' way of estimating the balance between goodness of the model and the number of parameters in the model (i.e number of clusters). Again, the results, while of technical merit, they are, in my opinion, not above the bar for Communications Physics.

Response: We thank the reviewer for this comment. Please allow us to elaborate on this point. The number of clusters is not a parameter of our LS algorithm, but a result. Our algorithm can identify the number of cluster, not like k-means that requires setting this before doing clustering.

As other comments related to this version of the manuscript and answer to reviewers:
1. I think that the question about using labels as ground truth is not well addressed. Using labels as ground truth has caveat (which have already been pointed out in the literature) that we should not take lightly. For real networks, there is at least one other way to test whether the algorithm identifies an good underlying network structure such as link prediction which some authors have used recently. It would be nice to address these issues in this type of works.

Response: We thank the reviewer for this comment. Different criteria have their own caveats. Using ground truth labels is a common practice, but it usually requires domain knowledge and detailed survey, which is difficult to obtain for large-scale networks. While using structure-related indicators (e.g., modularity or performance on link prediction) seems to be more objective, but it may not relate to the real generating process of the network, which may render the practical value of obtained partitions.

2. The question about how the different levels in multi-scale networks are selected form a previous report has not been addressed (or at least it was not clear to me).

Response: We thank the reviewer for the comment. Please allow us to elaborate again. For multiscale community structure generated by SBM models, where there are four first-level community and each comprises four second-level communities (see Fig. 3 in the main text). Our LS algorithm can identify the first level and second level community centers via obvious gaps. Between the first four centers (which are first-level ones and are denoted as a2, d1, c1, b1, respectively) and other centers, there is a first obvious gap; and then there is a second obvious gap that separates all second level centers with other follower nodes.

3. It would be nice for the authors to explain when comparing to other methods what parameters they used. This is necessary for reproducibility of the results. For instance in the inferential method they use (Peixoto PRX 2014), do they use a hierarchical prior, degree corrected or non-degree corrected SBM? How do they select the best model. In heuristic models, how many times do they run each algorithm? Details are also important when comparing with other methods. Same for the DBB clustering method they consider as a reference.

Response: We thank the reviewer for the comment. We discover that inferential methods seem to work well on homogeneous networks, which might relate to its underlying assumption based on SBM, and this is worth future closer investigations. For the inferential method we use (Peixoto PRX 2014), we use the "minimize_blockmodel_dl" function in graph_tool.inference and its default setting of the package. For DDB clustering algorithm, we also use the package released by authors when generating Fig. 6H, which is also the same as the results presented in (Pizzagalli et al., Sci. Adv., 2019, 5: eaax3770), and use their default setting for determining the distance threshold. For other clustering results related to DDB, we directly refer to the results presented in their original work.

Reviewer #2 (Remarks to the Author):

I have reviewed the revised version of the manuscript proposing the Local Search community detection algorithm, which defines communities as basins of influence of high degree nodes serving as local central authorities. The manuscript has been extensively revised in response to my comments and those of the other referee. The proposed definition of community may be useful in specific practical situations and is easily generalizable to alternative definitions of influence. Furthermore, the speed of the method is a significant advantage when dealing with very large heterogeneous networks, say of millions of nodes, for which other methods with complexity scaling as N^2 or above would be impractical. Considering that there is no "the Method" for community

detection in networks, an additional method offering advantages in some situations can be valuable for the interested community.

In my opinion, the manuscript is suitable for publication in Communications Physics. I have one additional suggestion. It would be useful to include a table in the main text wit basic statistics of the evaluated real networks, such as size, average degree, and maybe the rest of the metrics reported in SI Table 1, and also the level of heterogeneity in the degree distribution, assortativity, and clustering coefficient.

Response: Thank you very much for your positive comments. We have added a table with basic statistics of networks used in this work (Fig. 1 in the revised version).

	N	E	$\langle k \rangle$	$\langle d \rangle$	$\langle CC \rangle$	ρ	α
Karate	34	78	4.59	2.443	0.256	-0.476	1.781
Football[47]	115	613	10.66	2.397	0.407	0.162	–
Polbooks	105	441	8.40	2.841	0.348	-0.128	1.791
Polblogs	1,222	16,717	27.36	2.747	0.226	-0.221	1.415
Cora	2,485	5,609	4.08	5.738	0.117	-0.055	1.645
Citeseers	2,110	3,668	3.48	10.257	0.171	-0.024	2.074
PubMed	19,717	44,327	4.50	2.764	0.060	-0.044	2.227

Table 1. Basic statistics of networks. N is the number of nodes in the network, E is the number of edges, $\langle k \rangle$ is the average degree of the network, $\langle d \rangle$ refers to the average shortest path length between all node pairs, $\langle CC \rangle$ refers to the average clustering coefficient, ρ refers to assortativity, and α refers to the power-exponent of the degree distribution if it can be reasonably well fitted by a power law.

Reviewer #3 (Remarks to the Author):

In their manuscript, the authors develop an efficient community detection algorithm named Local Search (LS), which leverages the innovative concept of local dominance to identify community centers using local information. This methodology effectively uncovers local hierarchies within networks, setting it apart from traditional approaches by eliminating the need for heuristic optimization of a global objective function. Their work validates the LS algorithm's performance on synthetic benchmarks and empirical networks with known community structures, highlighting its efficiency, lower susceptibility to noise, and its comparative advantage over classical clustering methods in certain cases.

After careful consideration of the manuscript, the additional reports provided, and the authors' responses to the reviewers' comments from the previous round of revision, I find myself in alignment with the perspective of Reviewer #2. The critiques presented by Reviewer #1, while certainly valid and emphasizing the manuscript's theoretical limitations in comparison to inferential methods like those in Peixoto's work, do not diminish the revised paper's contribution to the advancement of community detection in networks. This contribution is particularly valuable for practitioners like myself, who prioritize methods that are swift, interpretable, and versatile. The potential impact of this work on the scientific community, especially for researchers and practitioners seeking innovative approaches to clustering and community detection, underscores the merit of its publication in Communications Physics.

Response: Thank you so much for your positive comments.

Building on this foundation, I would like to offer a few targeted suggestions to further refine the

manuscript. These comments are intended to enhance the paper's clarity and impact, addressing aspects ranging from overarching conceptual concerns to specific technical details:

1. Restrictions on graphs: At the beginning of the paper, it is very difficult to understand what type of graphs can be analyzed with Local Search (LS). Figure 1 suggests that only simple graphs can be considered, which is almost confirmed on page 14 since "We start with an undirected network with N nodes and E edges, for example, see Fig. 1A." The limitation regarding self-loops is not explicitly mentioned, but the algorithm seems to consider that V(u), the set of adjacent vertices of u, cannot contain u itself. The code provided is not clear either, since the authors use nx.Graph objects, which, in principle, can contain self-loops. Please clarify the restrictions as soon as possible in the paper.

Response: We thank the reviewer for this comment. Our algorithm can deal with self-loops. In the revised version of the code uploaded to Github, we added an input parameter to the LS algorithm, when the input parameter "self_loop" is False, which is the default setting, it neglects all self-loops; when users determine that self-loops are meaningful, they can set the parameter "self_loop" as True, it first traverse all self-loops to add corresponding degree or weight to the focal node, and then in the later procedures, these self-loops will not affect the adjacency, i.e., a node with a self-loop will not be identified as a neighbor of itself. We added more descriptions on this aspect at the step 1 of LS algorithm in the Methods: "*Our algorithm neglects self-loops in default, but if self-loops are meaningful for calculating degree of nodes, setting the input parameter ``self_loop'' of the algorithm as True will increase the degree of nodes accordingly, and nodes with self-loops will not be considered as neighbors of themselves.*"

2. Directed graphs: The discussion concludes by talking about the possibility of applying LS to weighted graphs. I think the case of directed graphs should also be mentioned, since this problem is not as straightforward. One possibility is to define two types of leaders: the "integrators" (high in-degree) and the "influencers" (high out-degree), leading to two types of clustering. However, the authors may have already investigated this problem of great interest, as most networks are expected to be weighted and directed (as in neuroscience).

Response: Thank you for this great suggestion. Extending our algorithm to directed graphs is indeed one of our ongoing work. We have added more discussions on this aspect, and thanks for suggesting these two types of leaders, we had been struggling with good names for them. In the discussion section, we added following sentences: *"Extending LS algorithm to directed networks is worth closer investigations in the future. In directed networks, two types of local leaders, the ``integrators" (determined by in-degree) and the ``influencers" (by out-degree), might be needed, which can lead to two types of clustering. The influence of edge directionality should be closely examined, as influence may propagate in the reverse direction of the directed edge. For example, on Twitter, information often flows from a user to their followers. Additionally, directionality affects the calculation of path lengths between nodes."*

3. Caption of Fig 6: I think that "E" should be replaced by "G" in "(C) and (E) show the partitions obtained from the Louvain method."

4. Caption of Fig 6: "(D and H) show the results" should be "(D) and (H) show the results."

5. Page 2, just before "where $V(u)$ is the set of neighboring nodes of u: Curly brackets are missing in the expression for k_ν.

Responses to Comments 3-4: All these are corrected and highlighted in red.

6. Reading the Author contributions, I feel that R. Li should be considered for the first author position. The authors might want to consider a different ordering.

Response: R. Li thanks so much for this suggestion. R. Li is in a transition from an independent researcher to a PhD supervisor, and the last corresponding author might be kind of a recognition for that, and we would prefer to keep the current author ordering.

7. Statements like "LS algorithm is linear" could be more precise by saying "LS algorithm is linear in the number of edges."

Response: We have change the phrase as "*our LS algorithm is linear in the number of edges*" and "*Our LS algorithm is of a linear time complexity in terms of the number of edges (see Methods and Supplementary Table 1*)".

8. p7: becomes more noticable -> becomes more noticeable.
9. p7: comparisons between algorithm -> comparisons between algorithms.
10. p7: algorithms, which generally relies -> algorithms, which generally rely.
11. p8: LS on Football network -> LS on the Football network
12. p8: remove comma in "While, our LS algorithm".
13. p8: replace "are" by "is" in "The choice(s) of the ground truth(s) are".
14. p13: "the LS method is not depend on the structure". Correct to "the LS method does not depend on the structure..."
15. p13: "every object are in a direct " -> "every object is in a direct".
16. p13 "optimization algorithm that utilize" -> "optimization algorithms that utilize".

Responses to Comments 8-16: Thank you so much for your comments on improving the clarity of our work. We have revised related expressions according to your suggestions.

17. p1-15 Consistency with the use of "centre" versus "center" depending on whether British or American English is being used.

Response: We go through the manuscript to change all centre as center. We greatly appreciate your careful attention to all these details of our manuscript, which is crucial for improving the clarity of our work. Thank you once again.