AOC; Assembling Overlapping Communities

Akhil Jakatdar[1], Tandy Warnow*[1], and George Chacko†[1,2]

[1]Department of Computer Science, University of Illinois Urbana-Champaign, Urbana, IL 61801
[2]Office of Research, Grainger College of Engineering, University of Illinois Urbana-Champaign, Urbana, IL 61801

August 19, 2022

Abstract

Through discovery of meso-scale structures, community detection methods support understanding complex networks. To this purpose, a variety of community detection approaches have been developed. Some have been applied to bibliometric data and have resulted in useful discovery. Many community finding methods, however, rely on disjoint clustering techniques, in which node membership is restricted to one community or cluster. This strict requirement limits the ability to inclusively describe communities since some nodes may reasonably be assigned to many communities. We previously reported a scalable and modular pipeline that discovers disjoint research communities from the scientific literature. We now present Assembling Overlapping Clusters (AOC), a complementary meta-method for overlapping communities as an option that addresses the disjoint clustering problem. We present findings from the use of AOC on a network of over 13 million nodes that captures recent research in the very rapidly growing field of extracellular vesicles in biology.

*warnow@illinois.edu
†chackoge@illinois.edu

1 Introduction

A community in a network generally refers to a group of nodes that are more densely connected with each other than to nodes outside the community. Various flavors of this definition exist, and the terms community and cluster are sometimes used interchangeably (**??**).

We are motivated by the problem of identifying and characterizing research communities in the network of modern scientific activity. Research communities represent scientific specialization (**??**) that evolves in response to influences such as new research paradigms, policy effects, collaboration practices, and increasing globalization. We are interested in scalable community detection methods for identifying research communities as they emerge and grow to maturity. We would also like to understand the extent to which these communities overlap.

Beyond identifying communities, we are interested in their structure and the roles played by community members. The observation by **?** that a community of oxidative phosphorylation researchers consisted of a small core of influential researchers and a much larger transient population drew attention to core-periphery or center-periphery structure in research communities. Core-periphery patterns have also been reported in other networks using different techniques, such as block modeling and k-core decomposition, arguing for some degree of ubiquity in their occurrence (**?????**).

One approach to identifying research communities is through analyzing citation patterns in the scientific literature. The underlying assumption is that members of a research community are more likely to cite each other's work than the work from outside their community. Drawing upon the rich literature in graph theory, this question can be framed as a community finding problem where a community is defined as a set of vertices in a graph that exhibit stronger connectivity to each other than to vertices outside such a community.

Thus, in the graph (or network) of scientific literature, citation-dense areas suggest the existence of communities of publications. Accordingly, a community of publications can be defined by edge-density and its researcher members can then be inferred from the authorship of these publications (**??**).

We stress that citation density alone does not make a confirming argument for the existence of a research community. However, community finding techniques are valuable in being able to efficiently search large datasets for communities, reducing them to smaller units of data that can then be examined with complementary analytical techniques that include the use of human judgment.

A considerable literature exists on community finding in graphs that re-

flects a diversity of perspectives and solutions. Hence various approaches can be drawn upon in identifying research communities, of which we cite a few (**????**). Community finding and clustering approaches have been applied to the scientific literature (**????????**). Most community finding approaches focus on disjoint partitioning, where a vertex or node is only assigned to only one community. This is a limitation when analyzing citation networks, since some nodes can reasonably be assigned to more than one community; for example, publications introducing widely used methods may be integral to the research activities of multiple communities.

We recently reported Iterative K-core Clustering or IKC (**?**), a recursive algorithmic approach based on the k-core property (**??**), which helps identify densely connected parts of a graph–the cores of core-periphery structures. Specifically, a k-core in a graph is a maximal connected subgraph where every node in the subgraph has at least k neighbors in the graph. The largest value for k for which a k-core exists in a graph is called its "degeneracy". IKC recursively extracts disjoint k-cores from a graph beginning with the largest value of k for which a k-core exists, and then reducing k until some user-specified lower bound on k is reached. IKC returns those clusters it finds that have positive modularity. However, IKC does not enforce global modularity maximization (**?**), which has theoretical limitations (such as the resolution limit).

We implemented IKC as the first step in a tunable modular pipeline to identify communities with core-periphery structure. Subsequent steps in the pipeline include breaking large cores and adding peripheral nodes to each core to construct communities with core-periphery structure. While IKC is a necessary step in the pipeline, the remaining steps are optional. We tested the ability of IKC to recursively extract k-cores from a network where nodes were publications and edges were citations. The input was a network of greater than 14 million articles centered around the rapidly growing field of extracellular vesicle biology (**?**). Using this pipeline, we were able to reduce a large network to two principal communities of interest that were robust to various option settings (**?**, Figure 5) from this large dataset.

However, IKC produces disjoint clusters, and some nodes can reasonably be assigned to more than one community. As noted above, this limitation impacts articles that describe widely used methods but is also relevant, in theory at least, to articles reporting discovery that are influential in more than one community. Thus, a need exists for methods that can produce meaningful overlapping clusters.

Several methods have been developed that address the limitation of disjoint clustering by producing overlapping clusters. (**???????**), although these do not appear to enjoy much use in scientometric studies and were not designed to address core-periphery structure in the scientific literature.

An interesting approach is the two-step procedure where the input graph is first transformed into a line graph whose nodes represent edges in the original graph (**?**). The line graph is then clustered and this output can be mapped back to the input graph to generate overlapping clusters. This general approach has been used by others on citation graphs (**??**). However, line graph techniques are not very scalable, since the size of a line graph is much larger than the size of its input graph. For example, the network that we studied in **?** would grow from 13,989,436 nodes and 92,051,051 edges to 92,051,051 nodes and 160,428,881,121 edges (Supplementary Materials), which presents a challenge to clustering software. Further, the line graph approach has not been adapted to detect core-periphery structure.

To address the limitation of disjoint clusters with IKC, we developed "Assembling Overlapping Clusters" (AOC), a scalable meta-method that takes the output of IKC and makes multiple community assignments from a list of candidate nodes, while enforcing criteria for cores. AOC can be used as an optional step in the IKC pipeline at the discretion of the user. We present results from AOC applied to the cores generated by the IKC method, and discuss the results and discovery made from them.

While the overarching question that motivates our work is identifying and describing research communities, this article is focused on the more narrow question of extending the IKC method towards inclusively identifying the cores of core-periphery communities and towards more general use in community detection. To test AOC, we use a citation network centered around the recent extracellular vesicle literature, a very rapidly growing field in biology (**?**). We use the terms core, community, and cluster interchangeably in this article.

2 Materials and Methods

2.1 Methods

Motivated by the graph-theoretic concept of k-cores (**??**), we have previously constructed a clustering pipeline we refer to as the Iterative K-core Clustering (IKC) pipeline in **?**, This scalable and tunable method takes, as input, two parameters k and p with $k > p \geq 1$, and computes a clustering of a given network N into disjoint clusters where each cluster has a "core" component, and a "periphery". This clustering is designed to satisfy several criteria: (i) the core is connected, has positive modularity (m-valid), and each node in the center is adjacent to at least k other nodes in the center (k-valid), and (ii) every node in the periphery of a cluster is adjacent to at least p center nodes in the cluster (p-valid). Thus, membership in the core of a cluster requires a

greater degree of connectivity to the other center nodes than membership in the periphery.

The IKC pipeline has three basic steps, where the second and third steps are optional. The first step (the iterative k-core extraction algorithm) produces disjoint clusters that are both k-valid and m-valid (i.e., km-valid), where each cluster has positive modularity and each node in each cluster is adjacent to at least k other nodes in the cluster; these form the centers or cores of the communities. The optional second and third steps breaks these clusters into smaller clusters and adds peripheries to the clusters respectively. Note that the parameter k is used to define the centers and the parameter p is used to define the periphery. If the only objective is cores or centers of communities, then the pipeline can be run using only the first step (or optionally also with the second step if smaller communities are desired).

The *Assembling Overlapping Clusters (AOC)* method, presented herein, builds on the first step of the IKC pipeline (k-core extraction). To run AOC, the user specifies two additional parameters: the set of nodes that being considered for membership in clusters and the criterion for membership. The two criteria for membership in the core of cluster C we consider are: (i) to have at least $MCD(C)$ neighbors in the center of C, where $MCD(C)$ is the minimum core degree (i.e., the minimum number of center neighbors of any node within C) and (ii) to have at least k neighbors in the core of C. We note that k (the parameter for condition (i)) has been used to construct the IKC clustering; hence, for every cluster C, it follows that $MCD(C) \geq k$ and so (ii) is a weaker condition than (i). We refer to the first membership criterion as AOC_m and the second as AOC_k.

Thus, whether running AOC_m or AOC_k on the IKC clustering, both pipelines produce a set of potentially overlapping clusters, each of which has positive modularity and where every node in every cluster has at least k other nodes in the cluster. This study is focused on core structure but the clustering process could be extended to the same optional second and third steps from the IKC pipeline, which would break up the large clusters and add peripheral nodes.

The input to AOC is a clustering of a network into disjoint clusters, a value for k, the criterion it uses (either AOC_m or AOC_k), and the list of candidate nodes. If this clustering is produced by IKC, by construction the clustering will be km-valid. The list of candidate nodes (the first of the two additional parameters) can be constructed from node-network characteristics such cluster membership, total degree, in-degree, or out-degree, or some other basis for candidate selection such as publication venue or funding sources.

Once candidate nodes are chosen, AOC constructs overlapping clusters by adding all candidate nodes to all clusters that meet the selected inclusion

criterion (AOC_m or AOC_k). Another condition that a candidate node must satisfy when added to a cluster is that inclusion of candidate node to the given cluster preserves positive modularity within the cluster.

The overlapping cluster construction stage of this method adds these candidate nodes as non-core members of their new clusters, while original members of the clusters are labelled as core members of the clusters they were originally a part of. This approach has two benefits: first, it maintains the original "definition" of the core part of the community (rather than potentially allowing gradual expansion of this core group) and second, it provides computational efficiency since the construction stage can take place in a single pass, does not depend on the order in which the nodes are added, and makes the test for meeting the AOC_k or AOC_m criterion very fast.

The resulting clustering generated from the overlapping clusters construction stage can now contains nodes in multiple clusterings, a property not found in the IKC method. Moreover, every cluster that is produced is still km-valid. Furthermore, if AOC_m is used as the criterion, then the MCD of each cluster is preserved.

2.2 Data

Citation network We previously generated a citation network (**?**) representing the exosome literature and more generally the extracellular vesicle literature (**??**) from the Dimensions database (**?**) in the Google cloud. For the present study, we curated this network to deplete it of both retracted articles and relatively high-referencing articles. Retractions were identified from a database kindly provided by Retraction Watch (**?**) and matched to nodes in the network using digital object identifiers (DOIs). Any article with 250 or more references was also removed.

While the network in **?** consisted of 14,695,475 nodes and 99,663,372 edges, the network resulting from removing retracted and high-referencing articles comprised 13,989,436 nodes and 92,051,051 edges. Thus, 706,039 nodes and 7,612,321 edges were removed. We refer to this network as the Curated Exosome Network (CEN). Its largest connected component consists of 13,988,426 nodes and accounts for 99.99% of the CEN.

Marker nodes. To identify exosome-relevant publications and communities, we re-used a set of marker nodes described in **?**. These 1,218 markers are the cited references combined from 12 different recent review articles on exosomes and extracellular vesicles. Of these, 1,021 are present in the CEN and were used to identify clusters relevant to extracellular vesicles research.

Random networks. We also explored clustering on random networks, specifically using both Erdős-Renyi (ER) graphs and configuration models. The ER

graphs analyzed in this paper were generated using the Python package NetworkX (nx) using the random graph function that requires four parameters: number of input vertices, number of input edges, random seed value (for reproducibility) and a Boolean value for whether the randomly generated graph is directed or not. Using this function, 100 ER graphs were generated using seed values from 0 to 99 inclusive. An example of a command used to generate a single graph is 'nx.gnm_random_graph(n=13989436, m=92051051, seed=0, directed=True)'. We also constructed configuration null models where the edges of the input network were randomized while preserving the total number of nodes, the degree of each node, and the publication year of each cited node in a citing-cited node pair (**?**).

3 Results and Discussion

As we explain in the preceding sections, AOC is a meta-method for overlapping communities that takes as input (i) a k-core based clustering such as IKC, (ii) a set of candidate nodes for consideration of membership in multiple communities, and (iii) a parameter k or m that defines the criterion for membership (Materials and Methods). We now examine the properties of non-disjoint clusterings produced using IKC followed by AOC. We analyze the effects of AOC on IKC clusters using either the nodes in non-singleton IKC clusters as candidates (the second input parameter) or high-degree nodes in singleton clusters. We also study the distribution of marker nodes in IKC clusters enhanced by AOC and finally, examine overlap across AOC clusters.

3.1 Characterizing the Curated Exosome Network

In an initial exploratory experiment, we clustered the curated exosome network (CEN) using IKC where $k \in \{10, 20, 30, 40, 50\}$, and we refer to these clusterings as IKC_k10, IKC_k20, etc. At the value of k with maximum coverage (k=10) in the CEN, 128 km-valid cores ($k \geq 10$ and modularity > 0) containing a total of 535,165 nodes (3.8% of the CEN network) are discovered. These cores range in size from 14 to 214,877, with a median core size of 79, and minimum core degree (MCD) varying from 10 to 53 with median MCD of 16. Thus, the CEN is a 53-degenerate graph consisting of 13,989,436 vertices.

The core sizes and MCD values for this curated exosome network are very close to the core sizes and MCD values of the original exosome network (prior to curation) studied in **?**. Specifically, the impact of curation reduced coverage from 4.2% to 3.8%, reduced the highest MCD value among the cores from 56 to 53, reduced the median core size from 85 to 79, and increased the number

of clusters from 119 to 128. Thus, the benefit of curation, which removed retracted and high-referencing articles, does not come at a steep cost.

To examine random network effects that could contribute to observed results from IKC, we generated 10 replicates of a configuration null model where the edges of the input network were randomized while preserving the total number of nodes, the degree of each node, and the publication year of each cited node in a citing-cited node pair (**??**). These shuffled networks were then clustered using IKC_k10. In all 10 cases, only a single km-valid core with MCD of 15 was extracted, although the number of nodes in each of these cores varied (Figure **??**). The median size of this cluster across 10 replicates was 435,216, roughly double the size of the largest cluster generated from the unperturbed network. These observations suggest that under these controlled conditions of randomization, the collective affinity of community members expressed as internal edge density is disrupted and a single core results. We did not run IKC at higher values of k on the shuffled networks since by definition no clusters would have been found. We also did not run AOC on the output from IKC with k=10 either, since AOC cannot add new members when the output has only one cluster and the candidate node list is restricted to membership of that cluster.

We also generated 100 random Erdős-Renyi graphs with the same number of nodes and edges as the CEN network and clustered them with IKC with $k = 10$ (Supplementary Material). No km-valid clusters were generated from these Erdős-Renyi graphs. We did not run AOC on the IKC output from the Erdős-Renyi graphs, since there were no clusters to use as input.

The distinct differences in results from the real world CEN network and the two random network models we explored show that the results seen in IKC clustering on a real world network are unlikely to be the result of random effects alone.

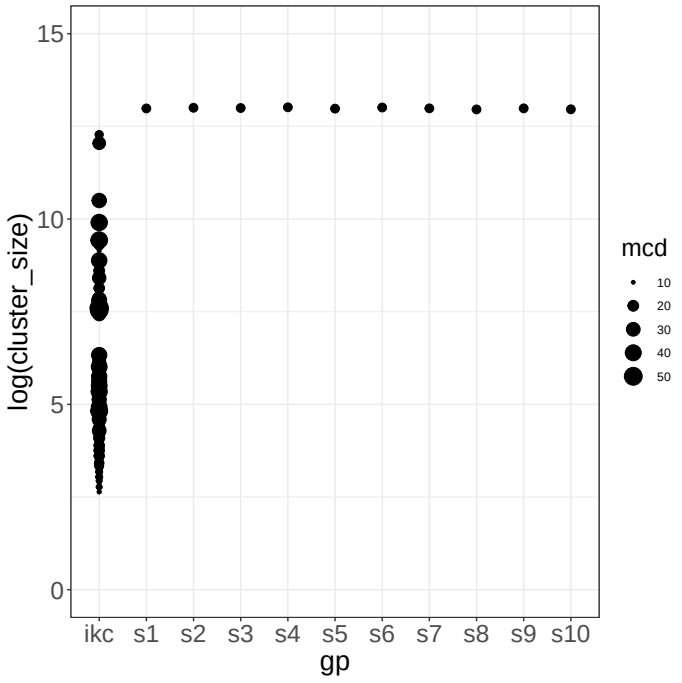

Figure 1: IKC clustering of Configuration Null Model. IKC clustering with k=10 of the original CEN network produced 128 clusters with MCD ranging from 10 to 52. The edges of the CEN network were randomly shuffled while preserving degree distribution for each node and the year of publication for citing and cited nodes. The resultant networks were clustered with IKC with k=10 (IKC_k10). In all 10 cases, a single k-core with MCD=15 resulted, although the size of this core varied slightly between replicates (s1, s2, ..., s10). Cluster size is shown on the y-axis in natural log units.

3.2 Effect of AOC on IKC clusters

As we assert, a limitation of disjoint clustering methods is that restricting membership to one community excludes assignment to other communities where a node may have both role and influence. Since clustering with IKC occurs iteratively with the densest core being extracted first, nodes in an extracted core are not considered for membership in cores that are subsequently extracted.

Accordingly we asked whether AOC could redistribute nodes from disjoint cores generated by IKC to other cores in the same clustering. In this experiment, we set the algorithmic parameters for AOC as follows: (i) the clustering produced by IKC with the CEN network was used as input to AOC with

$k \in \{10, 20, 30, 40, 50\}$; (ii) the set of candidate nodes was every node within the cores generated by IKC; and (iii) the criterion for membership was either (i) AOC_m or AOC_k (defined above).

	AOC_m	# clusters that do not change	# clusters that increase
1	ikc10	33	95
2	ikc20	10	34
3	ikc30	8	14
4	ikc40	3	3
5	ikc50	1	0
	AOC_k	# clusters that do not change	# clusters that increase
1	ikc10	17	111
2	ikc20	2	42
3	ikc30	4	18
4	ikc40	2	4
5	ikc50	1	0

Table 1: The number of clusters that change or do not change in size, after AOC_m or AOC_k treatment.

By construction, the number of cores cannot change by running AOC under any setting of its algorithmic parameters. However, core sizes can increase, with increases resulting from AOC_k being at least as large as increases resulting from AOC_m. Of interest, therefore, is how the algorithmic parameters, such as the value for k and the specified subset of nodes, impact the increase in size, and how node properties, such as degree, influence the number of clusters they are added to.

For both AOC_m and AOC_k a subset of the clusters increases in size (Table **??**). Figure **??** shows the distribution of cluster sizes generated by AOC relative to the core sizes from the IKC run at various values for k. Approximately 74% and 87% of 128 cores increase in size with AOC_m and AOC_k treatment respectively when IKC with k=10 is used as the input clustering. AOC treatment of IKC clustering, therefore, results in an increase in cluster sizes that is is inversely related to the value of k used in IKC and is more pronounced with AOC_k than with AOC_m. For reasons of coverage, we used IKC with k=10 in all subsequent experiments.

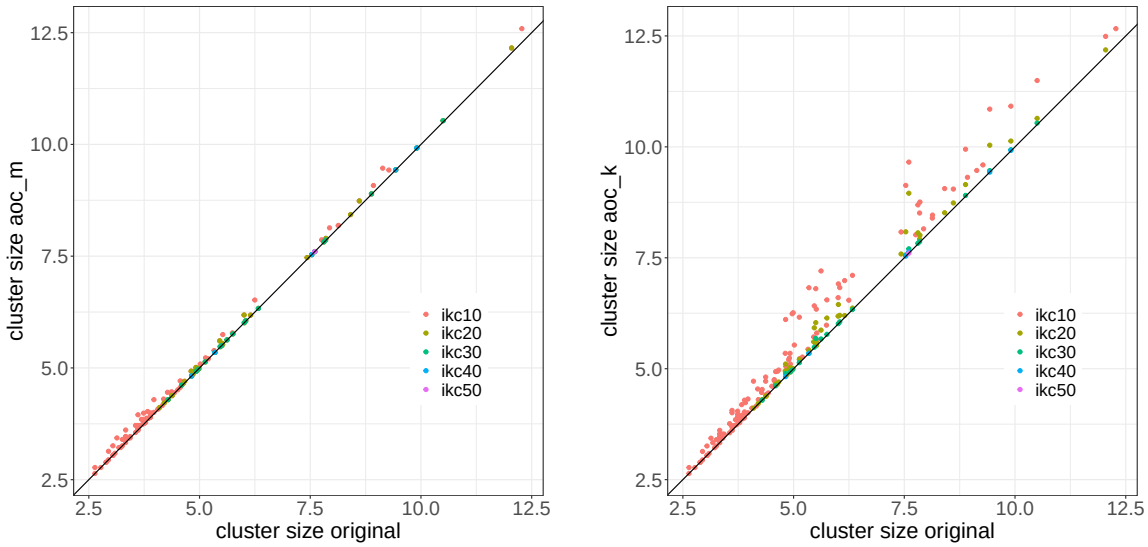

Figure 2: Comparison of cluster sizes between disjoint (IKC) and overlapping (AOC) clusters. Clusters were generated from the CEN network by IKC using values of k ranging from 10 to 50. These clusters were then enriched through the AOC process enforcing either *mcd* (left panel) or k (right panel). The input to AOC was the clustering produced by IKC and the set of candidate nodes to be assigned additional clusters was all nodes in non-singleton IKC clusters. Points that lie on the diagonal indicate no change in cluster size after AOC treatment. A natural log scale is used for both axes.

We then examined the number of clusters a candidate node was assigned to after AOC treatment of clusterings generated by IKC. For AOC_k treatment of IKC_k10 clusters, 54% of nodes in non-singleton clusters were assigned to between 2 and 24 clusters in a progressively decreasing manner, with roughly 26% of nodes assigned to 2 clusters and a single node being assigned to 24 different clusters. The remaining 46% of the nodes were assigned to a single cluster. AOC_m, in comparison to AOC_k, results in fewer multiple cluster assignments because of its more stringent membership criterion (Figure **??**).

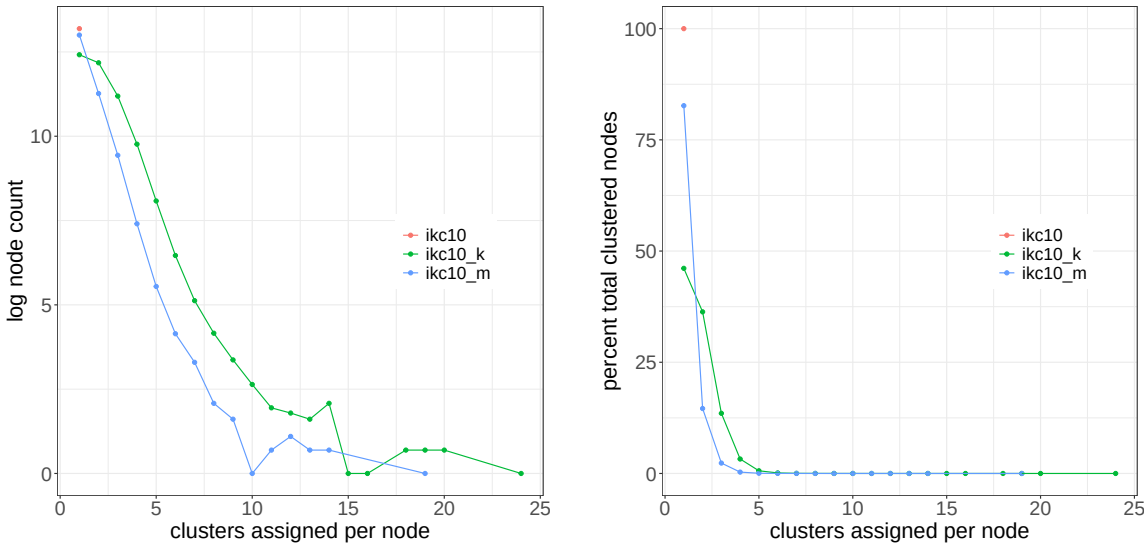

Figure 3: AOC selectively assigns nodes to multiple clusters. The count of nodes plotted against how many clusters a node was assigned to after AOC treatment enforcing either k (ikc10_k, green) or mcd (ikc10_m, blue); these are shown as natural log counts (left panel) or percentages of the number of nodes (right panel) in non-singleton clusters. One node is assigned to 24 different clusters in the case of AOC_k. The single red point in both plots (top left) indicates that all the nodes in the input IKC_k10 clustering are in single clusters.

3.3 Experiment 3: Which node properties impact multiple assignments?

To ask whether node degree is associated with the number of clusters the node is assigned to by AOC, we examined clustered nodes in the CEN and compared their total in-graph degree to the number of clusters they were assigned for either in AOC_m or AOC_k treatment of IKC_k10 clustering of the CEN (Figure **??**). We partitioned the nodes into five groups based on their in-network degree (in_degree + out_degree), with group 1 containing the nodes with the smallest total degree (less than 100), group 2 with nodes of total degree between 100 and 999, group 3 with nodes of total degree between 1,000 and 9,999, group 4 with nodes of total degree between 10,000 and 99,999, and group 5 the nodes with total degree at least 100,000.

Over 98% of these 535,165 nodes are in groups 1 and 2; 50.9% in group 1, 47.6% in group 2, 1.5% in group 3, 0.03% in group 4, and 0.0007% in group 5. Although group classification is based on total degree, any publication in

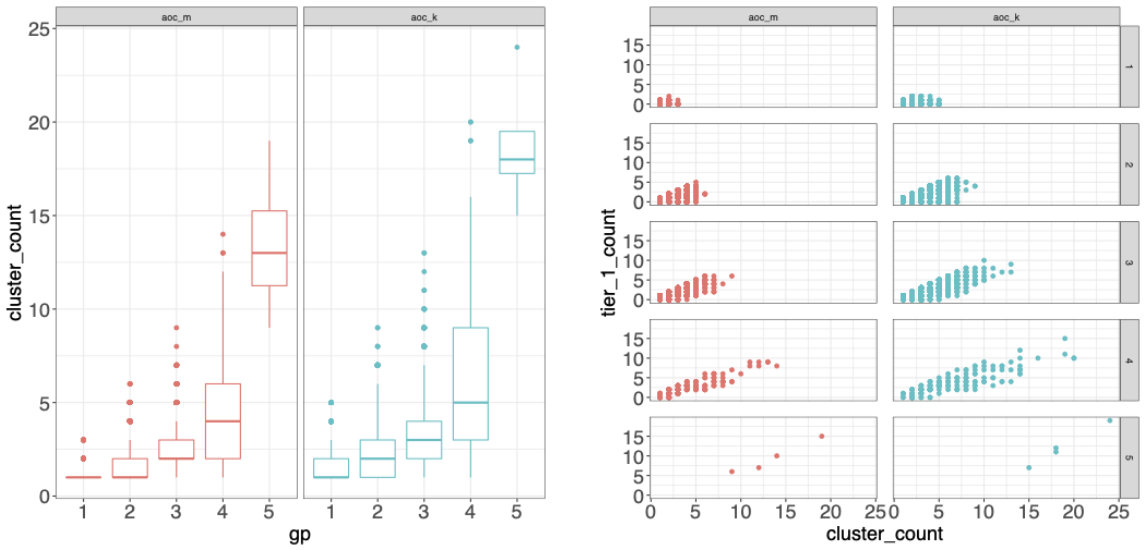

Figure 4: Cluster and tier assignments by node degree. The subfigure on the left shows the distribution of numbers of clusters each publication belongs to, and the subfigure on the right shows cluster count by Tier 1 membership for each group; each subfigure presents these results for both AOC_m and AOC_k. Nodes are partitioned into five groups based on their total in-network degree (in_degree + out_degree) with group 1 containing the nodes with the smallest total degree and group 5 with the nodes with the largest total degree. Group 1 has the largest number of observations and group 5 has the smallest. Group statistics (group; class limit; number of nodes): [gp1; <100; 272,395], [gp2; 100-999; 254,832], [gp3; 1,000-9,999; 7,773], [gp4; 10,000-99,999; 161], and [gp5; ≥100,000; 4].

groups 2-5 with high total degree must have high in-degree (in-network citation count). Membership in group 3 reflects a large citation count (at least 750 in-graph citations), and membership in groups 4 or 5, the top 169 nodes by total degree, reflects ultra-large citation counts (at least 9,750 in-graph citations).

We first compare clusterings produced by AOC_m and AOC_k (Figure **??**, left subfigure). For both AOC_m and AOC_k, the number of clusters that any node is assigned to increases as we move from group 1 to group 5, showing that in general total degree is associated with the number of clusters that a node is assigned to. The number of cluster assignments per node is larger for AOC_k than for AOC_m, which is not unexpected since AOC_m is a more stringent membership criterion. Hence, in-network citation count is associated with the number of clusters that a publication is a core member of.

However, for both AOC_k and AOC_m, the distributions for each group are overlapping, revealing potentially interesting differences between publications that are not explained just by citation count. Examining AOC_k, for example, we see the following trends. The largest number of clusters any node is assigned to is 25 and the smallest number is 1. All the nodes assigned to 14 or more communities are in groups 4 or 5, and so have total degree at least 10,000. In addition, group 5 publications belong to a minimum of 14 clusters.

In contrast, every other group has publications that belong to only 1 cluster. The largest number of communities for publications in groups 1 and 2 is 9, group 3 publications appear in at most 13 communities, group 4 publications appear in up to 20 communities, and group 5 publications appear in up to 25 communities. Since only group 5 publications appear in more than 20 communities, we conclude that, under the conditions of our clustering, an ultra-high in-graph citation count is necessary for assignment to a large number of communities. Results for AOC_m are similar but with reductions in the total number of clusters each publication can be in, which follows because AOC_m is a more restrictive condition than AOC_k.

While there appears to be an association between the degree of a node and the likelihood of it being assigned to multiple clusters, there are instances where nodes of high degree are assigned to only one or two clusters. For example, for AOC_k, five publications are found in Group 4 that are assigned to only one cluster (**?????**). Four of these five describe methods. All five were published in or before 1995 (1949-1995), have high in-degree (12,741 to 32,927) and very low out-degree in our data (0-11). Whether this low out-degree contributed to restricted cluster assignment merits follow up and opens up the questions of breadth and dependence (**?**) in publication communities, as well as that of data quality. This shows that high total degree is not sufficient for membership in many communities and underscores the case for mixed methods approaches.

Another perspective that provides additional insight into publications is

their "tier" within their communities. In **?**, we proposed a tier classification for nodes in a cluster, in which Tier 1 refers to the nodes in the top 10th percentile with respect to intra-cluster citations. Thus, when measuring in-degree within the cluster, a Tier 1 node is in the top 10 percent compared to all other nodes in its cluster.

We observe that nodes assigned to multiple clusters are more likely to have Tier 1 status (Figure **??**, right subfigure). The Tier 1 count also increases for AOC_k compared to AOC_m. The greater Tier I count for AOC_k is likely a combination of larger clusters and the in-degree of nodes within them.

For groups 1–4, there are some publications that are Tier 1 in all communities they belong to, some that are never in Tier 1, but the majority are in between. However, while the four publications in group 5 are Tier 1 for at least 7 communities for AOC_k, only one is in Tier 1 for more than 10 communities.

Interestingly, under AOC_k there are four publications in Group 4 that are in Tier 1 for at least 10 communities, two that are Tier 1 in strictly more than 10 communities, and one of these is Tier 1 in 15 communities. Also in AOC_k, we find publications in group 2 that are Tier 1 in up to 7 communities, publications in group 3 that are Tier 1 in 8 or more communities, and a publication in group 4 that is in Tier 1 for 15 communities. Results under AOC_m also show similar trends but with lower total Tier 1 counts, consistent with the reduced number of clusters that each publication belongs to.

These trends show that while total degree is correlated with the number of clusters a publication belongs to and how many clusters it is Tier 1 for, these values are not determined just by total degree. Thus, the cluster membership and Tier status within their communities provides complementary insights into the publications that goes beyond citation count.

3.4 High Degree Singleton Nodes

IKC clustering of the CEN results in 3.8% coverage. A large number of nodes of high in-network degree are assigned to singleton clusters and not to cores. In the case of the CEN clustered by IKC_k10, 15,039 nodes in the top 1% (by degree) of nodes in the network are assigned to singleton clusters. We examine here whether this population can be reduced with AOC.

With AOC_m using these 15,039 "singleton" nodes as candidates, no additional assignments are made and so the output of AOC_m is identical to IKC_k10. With the lower stringency AOC_k, however, 7,459 of 15,039 (49.6%) of the singleton nodes are assigned to IKC_k10 clusters, with all nodes assigned to at most 5 clusters, most nodes assigned to only one cluster, and only one node assigned to 5 clusters (Figure **??**).

In terms of cluster size increases, the effect is also mild: 71% of the 128 clusters in this AOC_k treatment do not increase in size. In contrast, when the candidates are nodes that are not singletons (so that they belong to a non-singleton cluster), 74% and 87% of the 128 clusters increase in size with AOC_m and AOC_k respectively (Section 3.2).

Thus, AOC enables nodes previously assigned to singleton clusters to be incorporated into cores, and while these assignments may not impact the clusters significantly, these assignments may shed light into the roles of these publications within the network. Whether this option is useful will depend on the purpose of clustering and the evaluation criteria designed by users for a specific study.

3.5 Experiment 4: Marker node concentrations

In the previous experiments, we examined the effects of AOC from a graph-theoretic one, aimed at generalizability. We now introduce a contextual perspective, in which we interpret findings relative to a field of interest. In this case, we are studying communities in the field of extracellular vesicle (EV) research. Context is addressed in two ways. We use, as input to IKC, a citation network (CEN) that is enriched in the recent extracellular vesicle literature. IKC reduces this large network to 128 cores that contain 3.8% of the nodes in the network. From these 128 cores, we identify a subset of of interest by using a set of markers derived from the cited references of recent reviews of the extracellular vesicle field that were authored by different researchers (**?**). Under the assumption that cores enriched in marker nodes are relevant to extracellular vesicle research, we further reduce the data under consideration to those cores. We now assess how AOC impacts marker node concentration (Figure **??**).

The count of cores with non-zero marker counts varied between treatments. For IKC, 17 of 128 cores exhibited non-zero marker counts. Clusters 3, 4, and 25 are notable in accounting for 87.5% of 1021 markers after IKC clustering. Because of this substantial coverage of marker nodes, from the perspective of EV biology, clusters 3, 4, and 25 are of obvious interest and offer a significant reduction in the amount of information to be studied qualitatively.

After IKC+AOC treatment, 20 of 128 (AOC_m) and 31 of 128 (AOC_k) cores respectively contained non-zero marker counts, which is consistent with their relatively stringent and permissive design. After AOC_m treatment of IKC clusters, clusters 3, 4, and 25 contained 30.4%, 42.5%, and 93.1%, respectively, of all markers. After AOC_k treatment of IKC clusters, clusters 3, 4, and 25 contained 69.5%, 71.8%, and 94.6% of all markers respectively. Thus, clusters 3, 4, and 25 all expand more significantly under AOC_k than under

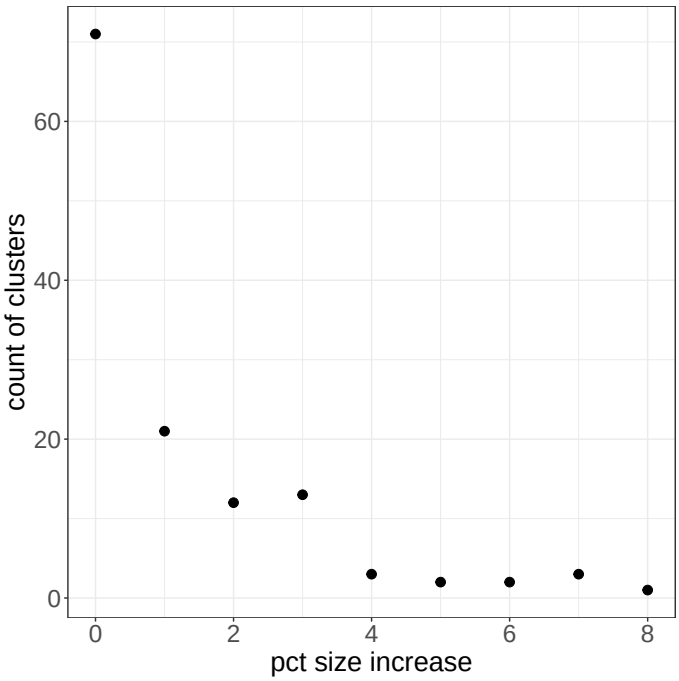

Figure 5: AOC_k but not AOC_m incorporates high-degree singleton nodes from IKC_k10 into clusters. 15,039 nodes belonging to the top 1% of nodes by total degree in the CEN were assigned to singleton clusters after IKC_k10 clustering. These 15,039 "singleton" nodes were used as candidate nodes for AOC_m and AOC_k treatment of IKC_k10 clusters. After AOC_m treatment, none of these 15,039 nodes were incorporated into any of the 128 clusters resulting from IKC_k10. After the more permissive AOC_k protocol, 7,459 (49.6%) were incorporated into one or more of the 128 IKC_k10 clusters. The count of clusters is plotted against percent increase in cluster size for all 128 clusters. The maximum increase in cluster size was 8%. The majority of the clusters (71%) did not change in size. After AOC_k treatment, 540,883 nodes (99%) of the nodes were assigned to one cluster, 1504 to 2 clusters, 210 to 3 clusters, and 26 to 4 clusters; one node was assigned to 5 clusters.

AOC_m. We also note that cluster 1 contained 20.7% of the marker nodes in IKC+AOC_k, indicating that cluster 1 is also significantly enriched for marker nodes by AOC_k. These data suggest that the recursive approach of IKC results in markers being segregated by disjoint clustering, but this effect can be remediated by post-processing using AOC.

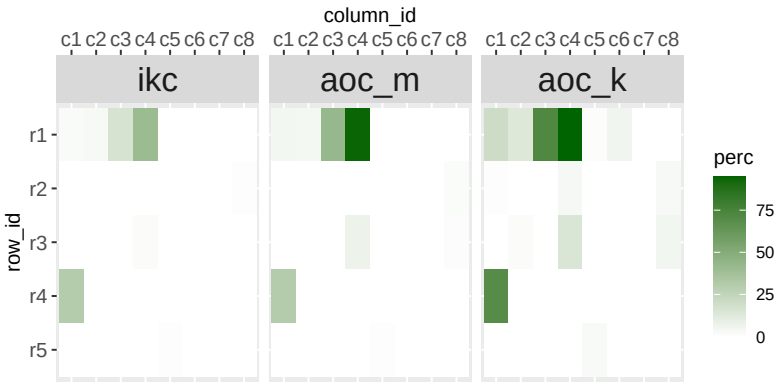

Figure 6: Marker Node Enrichment with AOC. We show marker node counts in 40 clusters (5 rows with 8 clusters per row) before and after AOC. The count of clusters with non-zero marker node counts is maximal in the case of AOC_k (right panel), with 31 clusters containing markers. Notably, the proportion of 1,021 marker nodes in the network increases from 40.7% in cluster 4 (r1,c4) of IKC clustering to 93.1% after AOC_m to 94.6% after AOC_k. The proportion of markers in cluster 25 (r4,c1) is the same for IKC and AOC_m but increases to 69.5% under the more permissive conditions of AOC_m. Data are shown for clusters where 1% or more of the markers are present in any of IKC, AOC_m, or AOC_k. *Perc*: Percentage of 1,021 marker nodes found in a cluster.

3.6 Examining overlap between clusters

The use of marker nodes is one approach to identify clusters of relevance. After enrichment by AOC_k or AOC_m, clusters will overlap, and the overlap consists of a mixture of marker and non-marker nodes.

Accordingly, we examined overlap between clusters after AOC_m or AOC_k treatment of IKC clusters, and we specifically relationships between clusters 3, 4, and 25, which were previously shown to be rich in marker nodes. Weighted edges were drawn between clusters based on the Jaccard Coefficient (ratio of intersection/union) for overlap. A threshold of the median Jaccard Coefficient from all values was set to permit an edge. Because some clusters do not have sufficient intersection with any others they do not have any incident edges in these graphs.

These networks, shown in Figure **??** (left shows the result for AOC_m and right shows the result for AOC_k), contain only those clusters that have at least one edge. Note that the left network (AOC_m) has fewer nodes than the right network (AOC_k) which indicates that augmentation using AOC_m does not produce substantial overlap for as many clusters as augmentation by AOC_k.

Beginning with the AOC_m network, we note that the network forms five connected components, with clusters 3 and 4 in the same component and connected by an edge. In contrast, cluster 25 is in a separate component. Thus, clusters 3 and 4 identified in IKC have substantial overlap after AOC_m, indicating some shared research questions or approaches, but cluster 25 reflects a different population.

Turning to IKC+AOC_k, we see a larger number of clusters, indicating that more of the clusters had sufficient overlap with other clusters to be retained in the network visualization. Interestingly here we see that clusters 3, 4, and 25 are all pairwise connected by edges, indicating that all three have have shared publications after AOC_k. Hence, what we see here is that the overlap after AOC_m between cluster 25 and the other two clusters is too weak to be considered significant, but is sufficient after AOC_k to be considered significant. Since AOC_m represents are more restrictive criterion, this shows that clusters 3 and 4 have a strong relationship to each other and a somewhat weaker relationship to cluster 25 but one that is nevertheless worth noting.

While both AOC_k and AOC_m offer insights into the EV research community structure, AOC_m provides more specific information about the research communities, since it is a local constraint that preserves the MCD of each core, compared to AOC_k which only enforces the value of k across all the cores and may therefore add weakly linked nodes to the IKC cores. On the other hand, the examination of clusters 3, 4, and 25 given above shows that

by combining information obtained from both AOC_m and AOC_k, additional insights can be obtained. Therefore, we allow both options and allow users to choose between them.

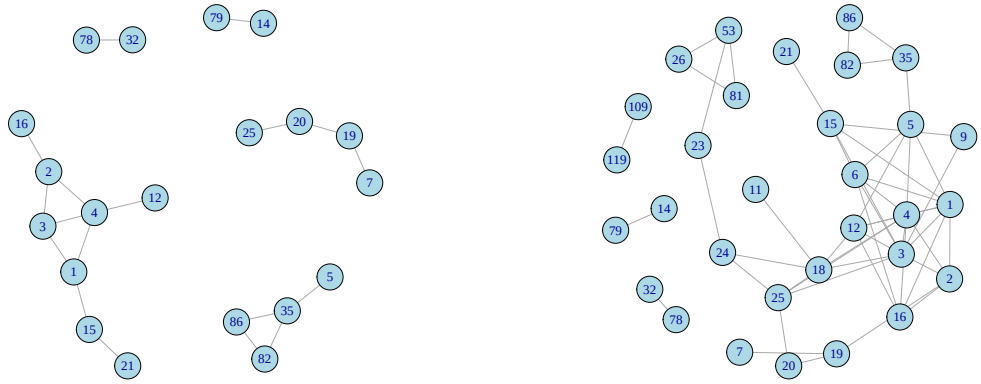

Figure 7: Overlapping clusters produced by IKC_k10 + AOC. Overlapping clusters were generated from CEN data using IKC_k10 followed by either treatment with either AOC_m (left) or AOC_k (right). The set of candidate nodes presented to AOC was all nodes in non-singleton IKC clusters. Edges are drawn between clusters based on the Jaccard Coefficient for node overlap and are visible if the JC exceeds the median value for all pairs. Cluster numbers in both panels correspond to cluster numbers from the input IKC clustering.

4 Conclusions

We developed AOC as a meta-method for overlapping clusters that serves as an option for users of the IKC pipeline. We sought to offer multiple options to users and this is achievable through varying the input data, the k setting for IKC, the two AOC options for membership, and the choice of candidate nodes. In this effort, we are positioned between method development and exploratory discovery.

One objective was to construct a modular community finding pipeline that is relatively subject-independent. Using IKC, a k-core based approach, we have identified citation-dense communities of publications. We have enriched these clusters by applying two variants of AOC. In this respect, we are able to address a limitation of our original IKC method that arises by its restriction to producing disjoint clustering (as a result of extracting k-cores in decreasing order of the value of k). This limitation prevents a node captured in a k-core from being considered for inclusion in a subsequently extracted k-core. Post-processing with AOC overcomes this limitation.

Since membership in multiple clusters following AOC occurs for many nodes, and is not completely predictable based on degree within the network, a benefit of AOC is that examination of the clusters a publication belongs to, and the role of the publication within these clusters, may provide additional insights into the roles of publications within the network that go beyond evaluation based on citation count. Such studies require expertise in the disciplines for the publications, and thus provides opportunities for specialists for future investigation.

A second objective was to study the extracellular vesicle literature. We have sought to include human experience and intent (**?**) in AOC through controlling the input data and enabling contextual evaluation in the form of externally identified markers. The results with the CEN suggest that AOC tends to enrich those clusters, already rich in markers. On the one hand, this may not be very useful in identifying the most marker-dense clusters but it does provide a more complete description for follow-on studies.

Finally, we note that AOC could be applied to clustering outputs from algorithms other than IKC. Further, it could also be engineered to accommodate new membership criteria such as average cluster degree. This is the subject of future work.

Competing Interests

The authors have no competing interests. AJ is presently in the graduate program at Princeton University; his contributions to this manuscript were made while he was a computer science major at the University of Illinois Urbana-Champaign.

Funding Information

TW receives funding from the Grainger Foundation. Research reported in this manuscript was supported by the Google Cloud Research Credits program through award GCP19980904 to GC.

Data Availability

Access to the bibliographic data analyzed in this study requires access from Digital Science. Code generated for this study is freely available from our Github site (**?**). Retraction data are available from The Center For Scientific Integrity, the parent nonprofit organization of Retraction Watch, subject to a standard data use agreement. Dimensions data were made available by Digital Science through the free data access for scientometrics research projects program.

Acknowledgments

We thank Srijan Sengupta from North Carolina State University for critical advice. We thank Alison Abritis and Ivan Oransky from Retraction Watch for helpful suggestions and for making data available. We thank Digital Science, Google, and the Grainger Foundation.