Temporal Development of Overlapping Communities in Co-Authorship Networks

Alexa Schlegel

Freie Universität Berlin, Institute of Computer Science
alexa.schlegel@gmail.com

Abstract. TODO TODO TODO

Keywords: co-authorship networks, scientific collaboration networks, overlapping community detection, temporal analysis, clique percolation, dynamic communities

1 Introduction and Motivation

A social network is a representation of a social structure consisting of actors (e.g. individuals, affiliations) and their social interactions. The network model conceptualizes (e.g. social, economic, political) structure as lasting pattern of interactions between actors. [**?**]

A co-authorship networks is a (scientific) collaboration network, where actors are scientists and the social interaction is the collaboration work resulting in publishing a paper together. Networks are in mathematical terms graphs, so they consist of nodes (vertex) and links (edges). In a co-authorship graph nodes are individual authors who are considered connected if the have co-aothored a paper. [**?**]

Usually those networks are used to study patterns of scientific collaboration in different fields, their development over time, identify key people regarding different measurement, detection and evolution of communities and many more. [**?**,**?**,**?**]

This seminar paper is motivated by a dataset consisting of 2277 publications related to geochemistry about stable metal isotopes (calcium, magnesium, lithium, siliium, etc.). This data was used to build an cummulated co-authorship network with an very simple approach to model collaboration: The dataset was collected using the bibliographic database scopus. It is an almost complete dataset starting from 1997/98 until 2014. The dataset consists of authors and co-athors, title, year, abstract and affiliations. Basic netowrk analysis had already been conducted for example centrality measurement, community detection, degree distribution and average degree.

As the dataset includes temporal information, the question arises how do communities within this type of co-authorship network evlolve over time. In geochemistry people often tend to focus on one isotope system, and stick to this topic for a long time. Do researchers in geochemistry really focus on one topic or do they change the field over time?

The goal of this seminar paper therefeore is to find and explain an already established and well understood/studied method for analysing the temporal development of communities within co-authorship networks. The main point is to understand the undelying algorithm in detail and to find out implications and challenges when applying this method to the geochemistry dataset.

2 Related Work

I am foucsing on community detection algorithms in general and especially for coaothorship networks. A second topic is how to track communities over time and how to model scientific collaboration in relation to edge weights.

"Most real world networks contain parts in which nodes are more highly conected to each other than to the rest of the network, those sets are usually called clusters, communities, cohesive groups or modules, they have no wiedely accepted, unique definition." [**?**] The way "more highly connected" is preciesely defined makes the difference between community detection algorithms. It depends on the heuristic which is employed to identify communiteis. [**?**]

Fortunato [**?**] gives an extended overview about different algorithms regarding community detection. To explain any would exceed the scope of this work. For example traditional methods include algorithms based on graph partitioning, hicharchical clustering and spectral clustering. There are also divisive and agglomerative algorithms. The best known one from this category is Girvan and Newman [**?**].

Most algorihtms determine distinctive sets of communities (a node can only belong to a single community), but most real world networks consists of overlapping and nested communities (e.g friendship network of a person) [**?**]. Fortunato [**?**] gives also an overview about algorithms detecting overlapping communities. The best known and widely cited is *clique percolation method* by Palla et. al [**?**]. There is an implementation provided by the authors called CFinder[1], and also an implementation in the R igraph library[2].

Not only overlapping communities exists in networks, but also communities are structured in an hirarchical way. Recent work on algorithms detecting hirachical and overlapping sturctures include [**?**,**?**,**?**].

"The main phenomena occurring in the lifetime of a community are: birth, growth, contraction, merger with other communities, split, death." [**?**] (see figure 1) The detection of dynamic communities has been studied by Palla et. al [**?**] using CPM for each snapshot of the graph. Further information on dynamic communities can be found in section 13 of [**?**].

As CPM is widely used method for detecting overlapping communities I will go with this method. The following describes the CPM and also the temporal algorithm mainly based on those tow papers [**?**] and [**?**].

[1] CFinder is a tool which implements the clique percolation method by Palla and finding and visualizes overlapping dense groups of nodes in networks. `http://www.cfinder.org/`, 05.12.2015, 22:43

[2] `http://igraph.wikidot.com/community-detection-in-r#toc0`, 05.12.2015, 22:45

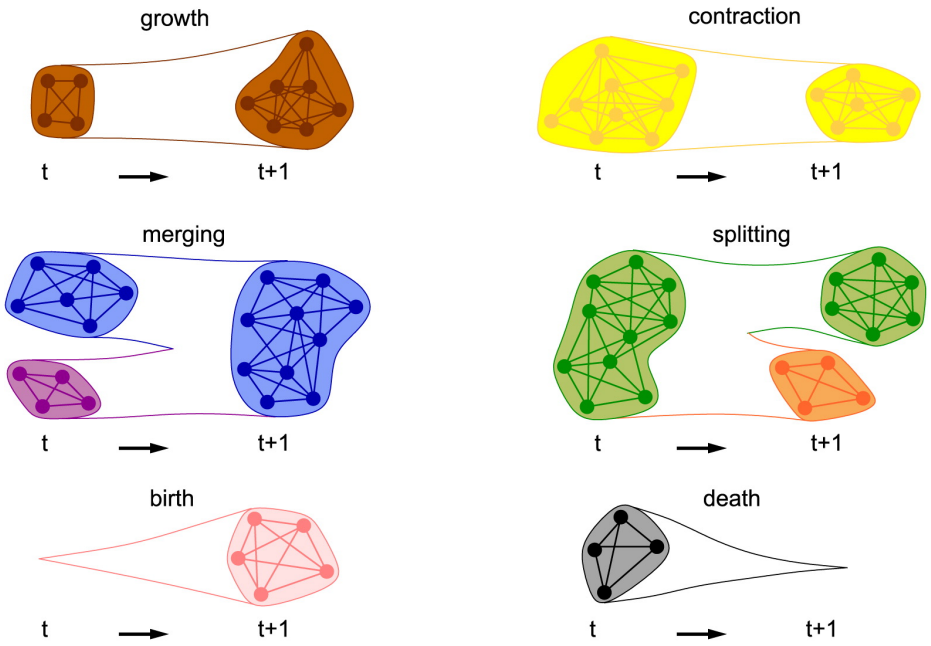
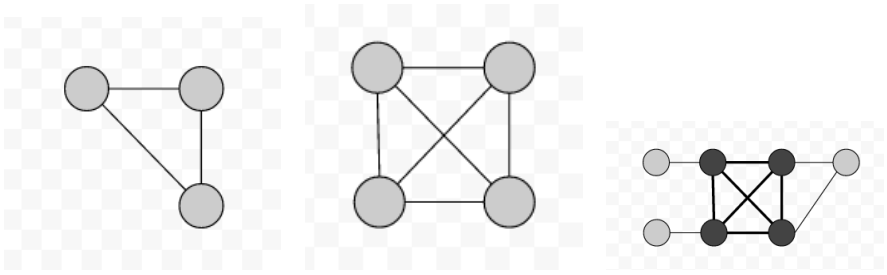


Fig. 1: lifetime of a community

3 Clique Percolation Method

The clique percolation method (CPM) developed by Palla et. al [**?**] is used to identify overlapping communities in unweighted and undirected networks. The authors tested and evaluated their method using a co-authorship network, protein-protein interaction network, word associations network and a network constructed from variables within the source code of an open source ftp programm. The following section explains the algorithm and summarizes the main findings of the paper regardin co-authorship networks.

The commmunity definition the algorithm implies, relies on the fact that a community consists of fully connected subgraphs, called *cliques*, that share many nodes. k-*cliques* are fully connected subgraphs with k nodes. In figure 3 example of $\{3, 4\}$-cliques are given. A community, in this context, is called a k-*clique-community*, which is defined as a union of all k-cliques, which can be reached from each other through a number of *adjacent* k-*cliques*. Two k-cliques are called adjacent if they share $k-1$ nodes. In figure 4 the so called k-clique template rolling is visualized for a 3-clique. A k-clique-community is best described by relocating one node until all nodes of the community are discovered.



3-cliquel 4-clique 4-clique inside a graph

Fig. 3: Examle of cliques

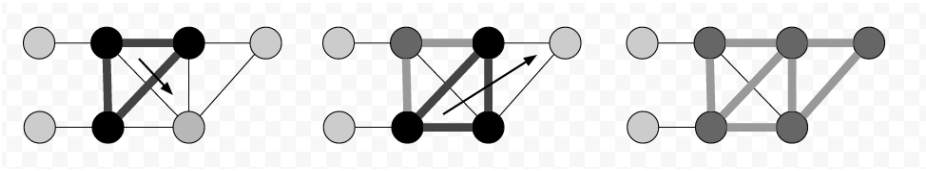


Fig. 4: Template rolling for a 3-clique

3.1 Algorithm

The starting point for the algorithm is a undirected and unweighted graph. In section 3.2 we talk about generating the unweighted co-authorship graph. Based on the explained community definition the algorithm consists of the following steps:

(1) Find all *maximal cliques*.
(2) Prepare clique-clique overlap matrix (clique graph).
(3) Treshold the matrix with a certain k.
(4) Each connected component in the clique graph form a k-clique-community.

(1) Find all maximal cliques Instead of finding all k-cliques (it is a polynomial problem) as a first step. The algorithm searches for all maximal cliques, this can be done in exponential time the authors state. Maximal cliques cannot be subsets of larger cliques. That is why they are detected in decreasing order of their size. The largest possible clique size s_{max} is determinde by the maximal degree d_{max} found in the network.

(1) Determine $s = s_{max}$.
(2) Repeatedly choose a node v from the graph:
  (2.1) Extract all cliques of size s containing v.
  (2.2) Delete the node and its edges.
(3) When no nodes are left set $s = s - 1$ and start with (2) on the original graph.

The set of already found cliques do influence the found cliques in later steps, as the later found cliques are smaller. The detailed algorithm for step (2.1) finding

cliques of size s of v can be looked up in supplemtary material to the paper on section 1.1.2, page 3. The result of this step is a set of all maximal cliques, an example can be seen in figure 2a. This set contains n_c cliques.

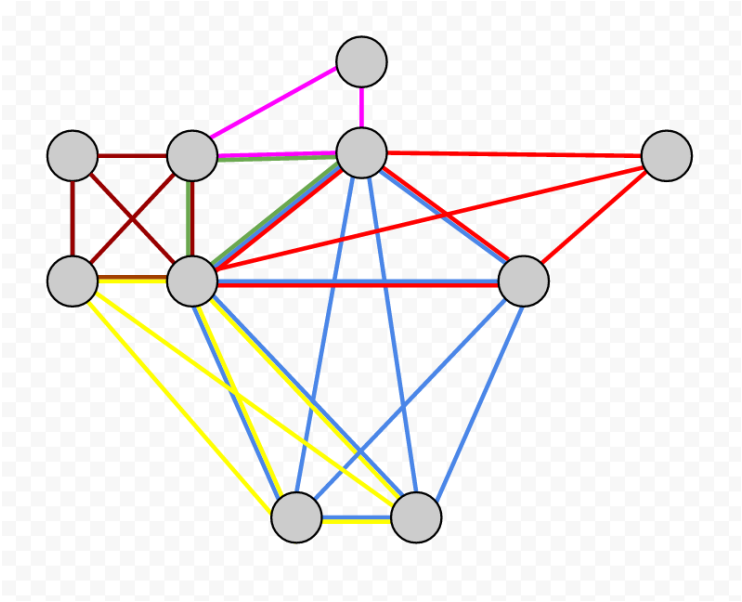


Fig. 5: All maximal cliques in a graph. Each color represents a maximal clique.

(2) Prepare clique-clique overlap matrix (clique graph) The dimension of the overlap matrix is $n_c \times n_c$. Each row and column represent a clique. All non-diagonal matrix elements represent the number of nodes those cliques share. The diagonal entries represent the size of the corresponding clique. This matrix is only generated once. In figure 6 the clique-clique overlap matrix of graph in figure 5 and the coressponding clique graph is shown.

(3) Treshold the matrix All off-diagonal entries smaller than $k-1$ and diagonal entries smaller than k are set to 0. The remaining elements are set to 1, resulting in a binary matrix, representing a network of cliques. The tresholding can be done repeatedly without calculating a new matrix. See figure 7.

(4) All connected components represent a community In the resulting graph we just need to look for connected components, those represent the k-clique-communities. See figure 8.

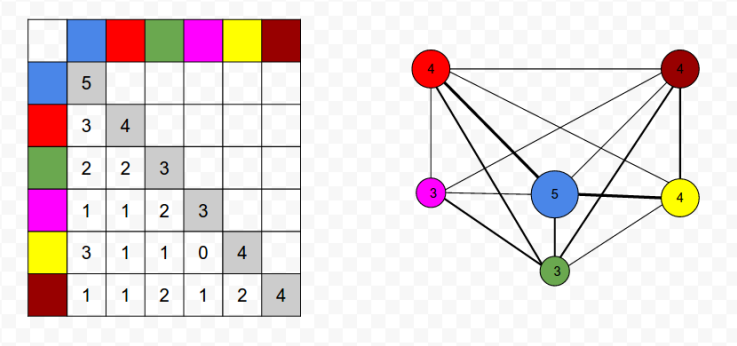


Fig. 6: The clique-clique overlap matrix and the coresponding clique graph. The number inside the node is the number of nodes belonging to that community, the width of the edge represents the shared nodes.

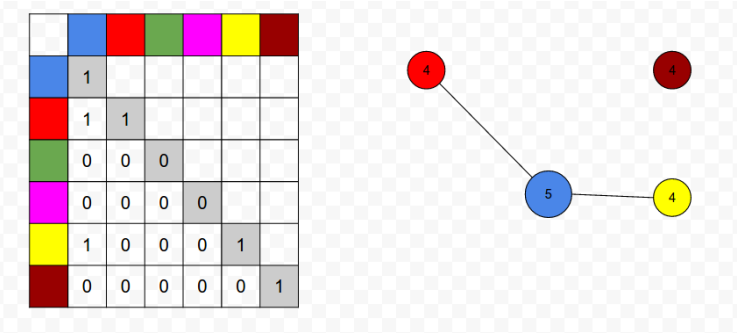


Fig. 7: This is the resulting clique graph after the treshholding with $k = 4$. All connected components do represent a k-clique-community. There are only two communities.

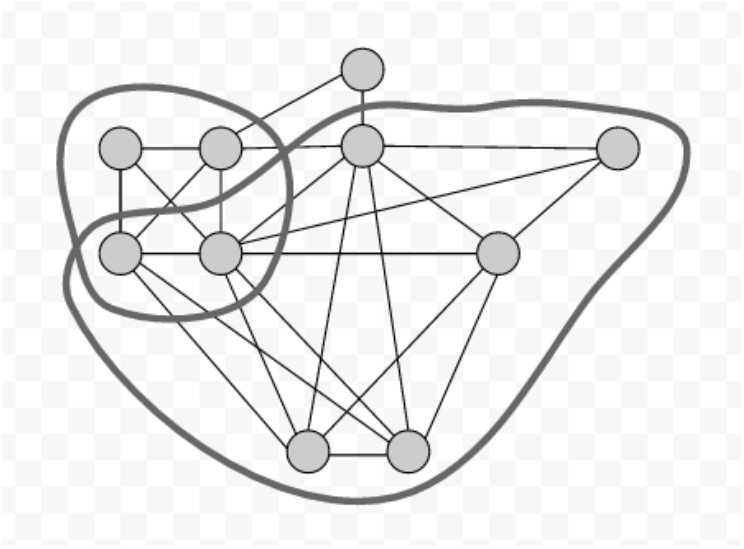


Fig. 8: Here the two resulting k-clique-communities are shown.

3.2 Construction of the co-authorship and choosing the right k and trashold.

The edge weights are calculated with $n/(n-1)$ with n as number of auther per publication. This results in modeling the collaboration between fewer authors as more intensive that looking at a higher number of authors. To generate an unweighted network all edges need to be removed under a certain treshold w^*, all others are replaced by weight 1. Increasing w^* results in fewer edges throughout the network, but stronger links and fewer detected communities.

In addition a good k needs to be chosen. As k is increased the detected k-clique-communities shrink, but at the same time become more cohesive (stronger connected).

So a good balance between k and w^* needs to be found resulting in a community structure which is highly structured.

The authors state that k is usually betweem between 3 and 6. If the number of links in a network are increased above some point a giant component evolves and covers the underlying community structure. So for each k the treshold w^* is lowered until the larges component becomes twice as big as the second largest component. For really choosing the best k some further characteristics describing the community structure need to be investigated, see Supplementary Information of [**?**] for more details.

3.3 Summary of measured statistics

A node i of a network has the following characteristics:

membership number m_i ... number of communities the node belongs to
overlap size $s^{\mathtt{ov}}_{\alpha,\beta}$ ... between two communities α and β
community degree $d^{\mathtt{com}}_{\alpha}$ ... number of links a community α has
community size $s^{\mathtt{com}}_{\alpha}$ ... number of nodes belonging to community α

The distributions of all characteristics are measured and analysed.

3.4 Main Findings of the paper

Overlaps in networks are significant. The distributions introduced in the paper (community size, community degree, overlap size, membership number) reveal universal features of networks. The network of communities has non-trivial correlations and specific scaling properties. [**?**]

4 Community Evolution based on CPM

Palla et. al [**?**] developed an algorithm based on clique percolation (see section 3) that allows the investigation of overlapping communities over time. They uncovered basic relationships characterizing community evolution within a co-authorship network and a phone-call network. The following section describes the main steps of the algorithm. A short summary of their main findings can be found in section 4.4.

4.1 Algorithm

The starting point of the algorithm is a set of undirected and unweighted graphs for each timestep. How to produce those temporal graphs is explained later in section 4.2. In general the algorithm uses the clique percolation method to find communities in each temporal graph, and matches the communities of consecutive timesteps.

(1) Extract communities with CPM for each graph g_t at time step t.
(2) Match set of communities at consecutive time steps of graph g_t, g_{t+1}, as follows:
  (2.1) Construct joint graph $g_{\cup} = g_t \cup g_{t+1}$.
  (2.2) Extract communities V with CPM in joint graph $g_{\cup}$.
  (2.3) For each extracted community V_i:
    – Extract communities in g_t and g_{t+1} that are contained in V_i.
    – Calculate relative overlap for each pair.
    – Match communities in decending order.
(3) Gap filling.

The following describe the steps (2) and (3) in detail. Step (1) was explained in section 3 already.

Matching Communities For matching communities, the *relative node overlap* $C(A, B)$ between two nodes A and B, in a simple way, is defined as follows:

$$C(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

As overlapping communities are allowed the matching from consecutive time steps in descending order of their relative node overlap can lead to missmatching. For example when small communities gain a lot of members or vice versa. An example for this problem is given in figure X [TODO-PIC]. As a solution, for each time steps t and $t+1$ a joint graph $g_\cup$ is constructed, containing all links from both networks. Let D be the set of communities at time step t and E the set of communities at time step $t+1$. The set of communities from the joint graph $g_\cup$ are extracted using CPM again and are called V. For any community $D_i \in D$ or $E_j \in E$ exactly one community $V_k \in V$ can be found. For checking weather E_i or D_j is contained in V_k the links are compared instead of nodes. For each community $V_i \in V$ the set of communities $D_i^k \in D$ and $E_j^k \in E$ contained in V_i are extracted. Now the relative node overlap between every possible pair can be calculated as follows

$$C_{i,j}^k = \frac{\left|D_i^k \cap E_j^k\right|}{\left|D_i^k \cup E_j^k\right|}$$

and the pairs can be matched in descending order.

In figures X three examples are given: figXa is a simple matching of a propagating community, figXb showing two merging communities with one community dying and figXc showning the splitting of a community into two communities with one community is new born.

Gap Filling In some cases a community which was disintegrated at a certain time step suddenly reappear in a later timestep, due to low publishing rates for example. That means a newborn community includes a formerly dead community. This problem is overcome by just fillig the gap with the last step of the almost disintegrated community.

4.2 Construction of the temporal co-authorship network

Events in the co-authorship are paper publications. The social connection between people writing a paper together usually starts before the event and last for some time after the event. The higher the fequency the closer the relationship [**?**].

The edge weight resulting from one paper is $n/(n-1)$ with n authors. The *link weight* between to nodes a and b at a certain time t is calculated as

$$w_{a,b}(t) = \sum_i w_i e^{\frac{-\lambda|t-t_i|}{w_i}}$$

The summation runs over all collaboration event in which a and b are involved. The event i occurs at time t_i and the corresponding edge weight at this time is

called w_i. This function which is a decay functions kind of models the strenght of collaboration between authors over time considering all event ever occured in time.

A treshhold w^* is used to only include certain edges to the emporal graph. So for each timestep a graph can be constructed based on the collaboration strenght per time step.

The authors of the paper used $w^* = 1.0$ for the co-authorship network, they said nothing about λ at all. (maybe explain how w/k was chosen?)

The dataset containes 142 month of publications, but I could find in the paper how the data was aggregated, because in the figues it looks like 50 timesteps in total.

4.3 Summary of variables an measured statistics

For evaluating the quality the overall coverage of the community structure (ratio of nodes contained in at least one community) is measured. Also the distribution of community size is measured.

4.4 Main Findings of the paper

The paper summarized differences between large and small communities and their development over time. Small communities live longer if the members stay the same over time. If members in small communities change frequently, they only live for a short time. Large communities live longer if members are changed permanently, if members stay the same they die quickly.

5 Implications regarding the geochemistry dataset

This part describe the steps I would need to undertake to carry out a temporal analysis of communities within the geochemistry dataset or with any other dataset, which forms a collaboration network.

(1) Genrate slices of the network for each year.
(2) Find function for calculating edge weights.
(3) Apply edge weights to each snapshot.
(4) Find out k.
(5) Adjust treshold w^* according to k.
(6) Do CPM for each snapshot.
(7) Conduct community mapping.

It is also important to look at basic statistics after (1) timeslicing. For example to look at how many publications per timeslice, degree distributions, nodes on average per timeslice to check of timeslincing was done right or needs to be adjusted. The function for calculating edge weights (2) could be used from the paper (see section 4.2), but it would need to be checked if the resulting networks (3) are still valid. For findng the right k (4) and w^* (5) and extended analysis of network caracteristics like coverage of community structure and distribution if community size

(see 4.3) would need to be carried out. Also the quality of the detected communities would need evaluation, comparing density within the community to outside the community. Using CFinder or R for CMP (6) detection. It is open to me if there is an implementation for the community mapping (7), if not I would neet to implement it. Time complexity will be no problem because my dataset is not very big. About 2.000 paper resulting in about 500 nodes.

6 Future Work

As this paper dicusses only one method an extensive literature research should follow, finding out more methods for that problem. Also a closer look on recent work would be good. The results from this dataset should be compared to the results of Palla. Maybe some time should be invested for finding a good representation for edge weights calculation, related to collaboration definition. The visuaization of the community evolution remains open and would be a broad topic, but exciting to investigate further.