COMMUNITY DETECTION IN LARGE NETWORKS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

June Andrews

May 2012

COMMUNITY DETECTION IN LARGE NETWORKS

June Andrews, Ph.D.

Cornell University 2012

Networks are large and demand attention at being understood for these reasons.

With the impossibility of understanding a network a node at a time and the incompleteness of data, we seek to find clumps of data that exhibit cohesion. We call these communities.

With community detection we hope to better our understanding of large networks. This thesis makes advances towards understanding existing methods, introduces a greedy algorithm within current community detection and steps outside towards the creation of parallel community detection method.

Your abstract goes here. Make sure it sits inside the brackets. If not, your biosketch page may not be roman numeral iii, as required by the graduate school.

BIOGRAPHICAL SKETCH

June Andrews was born in San Diego, 1985. She attended University of California, Berkeley for her undergraduate degree in Electrical Engineering and Computer Science, with a minor in Applied Mathematics. She is now completeing her doctoral degree in Applied Mathematics at Cornell University.

Here's to you Da.

"There are more things in heaven and hell, Horatio, than are dreamt of in your philosophy." - the Bard.

ACKNOWLEDGEMENTS

It goes without saying, these people have been inspiring forces of nature to work with:

- Len Kulbacki
- Coach Wilson
- James Sethian
- Patricia Kovatch
- John Hopcroft
- Steve Strogatz
- Jon Kleinberg

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 History of Community Detection

In many applications, data can be stored in a graph where objects of interest are the nodes and interactions between the objects are the edges. If the nodes and edges form a very distinct pattern, say a circle or disconnected sets of cliques, the graph is easy to analyze. Of particular interest is if the graph contains a set of nodes, ie a community, whose members interact with each other in a particular way and interact with non-members of the community in a different way. If such a community is found, then two questions arise. How are members of the community related? How does the community interact with the rest of the graph? For social networks, we know that groups exist [**?**] and would like to answer those two questions. However, large social networks have thousands of nodes with complicated interactions and no simple pattern to reveal the communities. Hence, in order to find the communities, we must develop the ability to see the forest through the trees. This is the object of community detection.

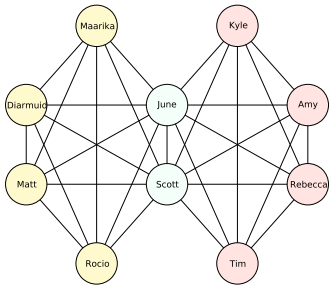

Figure 1.1: A simple graph of people and their friendships. The graph is regular enough to reveal two communities.

Given a graph, there are two prominant questions community detection seeks to answer. The first is, for the graph, what is a community? The second is, for the graph, what are the communities? Several approaches have been developed to answer these two questions, some with a particular application as motivation. We outline the coupling of a few sciences and one of their preferred detection methods in table 1.1. Before about the year 2002, most development of community detection was done within the fields of the applications. Since then, computer scientists have contributed a large volume of advances towards answering these two questions for applications in general. The first goal of this thesis to try and tie together a portion of these advances into a cohesive understanding of community detection. The second is to use our perspective to create fast and parallel algorithms.

Application	Community Detection Method
Parallel Computation Distribution	$k-$means clustering [**?**]
Physics	Belief Propogation [**?**]
Search Queries	[**?**]
Sociology	[**?**]
Storage of Large Matrices	Spectral Analysis [**?**]
Taxonomy	Neighbor Joining [**?**]

Table 1.1: A subset of applications and one of their preferred community detection methods.

1.2 Graph Partitioning Methods

1.2.1 Top Down Approaches

Betweenness and Centrality Measures

**********STOP**********

Betweenness and centrality measures were first presented by JTODO in [**?**]

To develop our intuition of betweenness, let us consider a network of roads. Say we are driving around in city C. If I want to travel between any two points in the city the density of roads allows me to take several different, even non-overlapping, routes. Now, if we wish to travel to another city, chances one of a limited number of highways will be traveled. ...

Conductance

JTODO find a concrete paper on conductance

Conductance is a measure of the value of a cut within the graph. The lower the value of the cut, the more likely that one side of the cut is indeed a community []. In particular, for a given cut between C and $\bar{C}$, the conductance value of the cut is the ratio of external edges of C to all edges that have an end point in C. If C and $\bar{C}$ are disconnected, the value of the cut is then 0, the minimal value of conductance, the most likely senario for predicting the existance of a community C.

1.2.2 Bottom Up Approaches

The question was, given a partition of the network, how good is that partition? Of course, the more distinct the partition, the more useful the found communities would be in coming to global conclusions. To measure the quality of a partition, several metrics exist. For each of these metrics then, finding the best partitioning is a matter of optimizing the metric.

The first use of a metric was in circuit board partitioning []. Since then modularity has become the overwhelmingly popular metric with other attempts as well.

Modularity

Here is modularity.

mention variationes

1.3 Overlapping Community Detection

We call two communities overlapping, if there exists a node that is a member of both communities. In practice these communities are not unusual. For example, think of the community of your colleagues and the community of your family. You are a member of both communities, and while they are different communities, they are overlapping. In fact, for most social networks, we expect there to be many overlapping communities. Without a careful definition of a community, there can be an exponential number of communities within a

network.

We now present two different approaches to finding overlapping communities.

1.3.1 Alpha Beta Clustering

In previous sections, communities were the partitions of a graph. Each node was placed in exactly one community. So if it was optimal to place node, n, in community C_1, then node n would not be placed in community C_2. Alpha beta clustering makes a change to this step. If adding node n to community C_2 has a high value, alpha beta clustering adds node n to community C_2, as well as C_1. This simple change dramatically restructures community detection. The new structure is a two part process:

1. Create a definition of a community that does not depend on other communities in the graph.
2. Find each community seperately.

We now present Mishra's et al [19] approach following these guidelines. Let us say the strength of a connection between a node and a community is the number of edges the node has to members of the community, denoted as $|E(n, C)|$. See table **??** for a list of all notation. Mishra et al [19] use this notion of strength to define a community satisfying the first guideline. In particular, no node outside of the community is more strongly connect to the community than any of the nodes inside the community. This an (α, β) community with a formal definition following.

Definition 1 ((α,β)– Community) *For community C, let:*

$$\begin{aligned} \alpha(C) &= \min_{n \in C} |E(n, C)| \\ \beta(C) &= \max_{n \notin C} |E(n, C)| \end{aligned}$$

If $\alpha(C) > \beta(C)$, then C is an (α,β) community.

Given this definition, Mishra et al[19] are able to find communities quickly and in parallel. In our development of a parallel algorithm we use the same guidelines.

1.4 More Approaches

So far we have introduced the community detection methods that have provided inspirations to this thesis. There are countless more methods. We briefly outline the most prominent of those methods.

- Swapping of Kernighan-Lin
- k-Clique Percolation
- Belief Propogation
- heirarchy methods
- Principle Component Analysis
- Spectral Analysis

1.5 Desired Improvements

In the field of community detection both algorithms and data sets are increasing in complexity. Hence, a useful theoretical result is the ability to compare and understand complex algorithms. Additionally, a useful experimental result is the ability to compute overlapping communities in parallel on large networks.

We deliver on the these results:

- A framework for comparing existing community detection methods.
- A community definition encouraging overlapping communities.
- A parallel algorithm with near perfect scalability to analyze large networks.

An incredible result would be the ability to automatically analyze the communities found in a network. We have not provided that result. But, we do some processing of the communities found in sample networks.

1.6 Notation

This thesis will use consistent notation and assumptions. Here, we provide a reference for all variables in table 1.2 and assumptions. There exist several similies in community detection, for clarity we mention them now. A network is also a graph. A node is also a vertex, person, paper, or any type of object within the network. An edge is also an interaction between two people, a citation between two papers, or a connection between any two objects within the network.

Table 1.2: Notation

Variable Name	Description	Constraints
V	Set of all nodes within the network	$\{u \mid u \in$ the network $\}$
u and v	Nodes	$u, v \in V$
$w(u, v)$	Edge Weight Function	$w : VxV \rightarrow \mathbb{R}_{[0,1]}$
G	Network or Graph	$G(V, E)$
C	Community	$C \subset V$
k	Fraction of nodes within C	$k = \frac{\lvert C \rvert}{\lvert V \rvert}$
$\lvert C \rvert$	Size of C	$\lvert C \rvert = k\lvert V \rvert$
S	Set of Communities	$S = \{C_1, C_2, \ldots, C_n\}$

Table 1.3: Introduced Functions

Function	Description
$I(C)$	Internal Density of a single Community, C, definition 4
$E(C)$	External Density of a single Community, C, definition 5
$I(S)$	Internal Density of a set of Communities, S, definition 7
$E(S)$	External Density of a set of Communities, S, definition 8
CONCISENESS(S)	Conciseness of a set of Communities, S, definition 9

The assumptions we make are:

- *Self-Loops.* We presume there are no self loops in the networks. As a node will always be in the same community as itself, self-loops provide redundant information. Accordingly, $w(u, u) = 0$, for all $u \in V$.

- *Edges* We presume that all edges exist and are weighted between 0 and 1. The edge weight function is $w : VxV \rightarrow \mathbb{R}_{[0,1]}$ Unweighted graphs can easily be adapted into this notation.

Definition 2 (Internal Edges) *Internal edges are edges between members of the same community C.*

Definition 3 (External Edges) *External edges are edges between a member of community C and a non-member.*

CHAPTER 2

A FRAMEWORK FOR THE COMPARING METRIC BASED DETECTION METHODS

2.1 Previous Comparisons

Given the variety of community detection methods, researchers have tried to compare them. In ..... they use conductance as a measure of the strength of communities produced by each method. In...

All comparisons have been of an experimental nature. A metric and data set are chosen and algorithms are compared. This methodology has lead to several comparison papers [] [] [], each of which offer general conclusions, but not theoretical results.

2.2 Individual Community Based Metrics

We now explore metrics that evaluate the strength of a single community. There are two uses of such metrics. The first is to recursively partition a network to find communities. This is done by finding the community, C, that maximizes the metric, partitioning the graph into C and $V - C$ and recursively partitioning the two subsets of the graph with the metric. The second, is to provide an intermediate way of evaluating the strengths of communities returned by more complex detection techniques. The later use is more common for these metrics, and for this purpose, CONDUCTANCE is the most popular intermediatory metric []. However, the benefit of using conductance in such a way has been

unclear and is used more as an implicitly mediatory metric to compare communities from different detection methods than as an understood evaluation of communities.

In the following section, we show that the use of any of the existing metrics to find heirarchical partitions of a network will not result in conventionally accepted strong communities, confirming []. We also show that the use of these one dimensional metrics to evaluate communities, hides revealing information about the communities.

Our framework is to choose parameters that pertain to the desired characteristics of a community, parameterize the existing metrics, and analyze how change in the values of the parameters affect the metrics. By choosing parameters to understand the metrics, we can avoid the network dependent analysis of previous work [] and draw conclusions for all networks.

2.2.1 Internal Density and External Sparsity

It is accepted that the most distinct community is a clique, disconnected from the rest of the graph. There are two characteristics of this ideal community. The first is that it has high connectivity between the nodes of the community. We deem this property, *internal density*. The second characteristic is that the community has low connectivity to the rest of the graph. We deem this property *external density*. Formal definitions follow.

Definition 4 (Internal Density) *Internal density is the ratio between number of edges that exist between members of the community, internal edges, to the number of all*

possible edges that could exist within the community. Hence, $I(C) : C \to \mathbb{R}_{[0,1]}$, where

$$I(C) = \frac{\sum_{u \in C} \sum_{v \in C} w(u, v)}{|C|(|C| - 1)}. \tag{2.1}$$

For a community C that has no edges between its members, the *internal density* will be minimized with, $I(C) = 0$. For a community C that is a clique, *internal density* will be maximized with $I(C) = 1$. Intuitively, the closer a community, C is to an *internal density* value of 1, the more sure we are the nodes have a meaningful connection that be used to observe aggregate behavior. If the graph is indeed a weighted graph, where w has values between 0 and 1, the same intuitions apply and *internal density* reflects how close a community is to being a maximally weighted clique.

Definition 5 (External Sparsity) *External sparsity is the fraction of edges that exist between a member of the community and a non-member of the community, external edges, to all possible edges that could exist leaving the community:*

$$E(C) = \frac{\sum_{u \in C, v \notin C} w(u, v)}{|C|(|V| - |C|)}. \tag{2.2}$$

For a community C that has all possible *external edges*, external density will be maximized at $E(C) = 1$. For a community C, disconnected from the rest of the graph, external density will be minimized at $E(C) = 0$. Intuitively, the closer a community is to having a value of $E(C) = 0$, the more complete the community is guaranteed to be.

There are other representations of $I(C)$ and $E(C)$ that vary the how the $|C|$ and $|V|$ terms are used. The analysis and conclusions that follow are not sensitive varying such variations. With our parameterization, we have that all communities can be mapped to a point $(I(C), E(C))$ that is in the square $\mathbb{R}_{[0,1]}x\mathbb{R}_{[0,1]}$, this is an easy to visualize space. We can now dictate what the ideal, or strongest, community is mathematically.

Definition 6 (Ideal Single Community) *A community, C, is ideal if it is an isolated clique, specifically has the following properties:*

$$\begin{aligned} I(C) &= 1 \\ E(C) &= 0. \end{aligned}$$

Metrics provide a one dimensional analysis of communities. Our proposal is to look at these two dimensions of communities explicitly, understand how metrics use these characteristics in their evaluation of a community, and from there draw our conclusions about metrics.

2.2.2 Study of Relevant Metrics

Given that we can map a community, C, to the point $(I(C), E(C))$, we now analyze how different metric based detection methods operate in the I, E plane. We cover six metrics that evaluate a single community. We use one approximation to simplify the equations, $|C| \approx |C| - 1$. This approximation has a larger impact on smaller communities, but most communities of interest are large enough to accomodate the approximation. Additionally, we introduce variable, k, representing the fraction of the nodes within community, C, such that $|C| = k|V|$

- CONDUCTANCE is the probability that a step in a random walk will leave the community [].

$$\text{CONDUCTANCE}(C) = \frac{(1-k)E(C)}{kI(C) + (1-k)E(C)} \tag{2.3}$$

- CUT RATIO is the fraction of existing to possible edges leaving the community [].

$$\text{CUT RATIO} = E(C) \tag{2.4}$$

- EDGES CUT is the number of edges connecting the community to the rest of the graph [].

$$\text{EDGES CUT} = k(1-k)|V|^2 E(C) \tag{2.5}$$

- EXPANSION the average number of edges leaving the community per node [].

$$\text{EXPANSION} = (1-k)|V|E(C) \tag{2.6}$$

- INTERNAL DENSITY as a metric, previously existed before our definition of $I(C)$, []. However, we stick to our definition of $I(C)$ for intutive reasoning and note in previous work internal density represents the mirror image of our definition.

$$\text{INTERNAL DENSITY} = 1 - I(C) \tag{2.7}$$

- VOLUME is the total degree of nodes within the community [].

$$\text{VOLUME} = |C|^2 I(C) + k(1-k)|V|^2 E(C) \tag{2.8}$$

Hence, we can put all previously described metrics in terms of $I(C)$, $E(C)$, $|C|$, $|V|$, and k. With a common parameterization of the metrics, we can already draw some inferences. All metrics, besides VOLUME and CONDUCTANCE are

a function of either $I(C)$ or $E(C)$, but not both. A metric that considers only $I(C)$ will be optimized by any clique. Which is a very restrictive definition of a community and finding all communities in the graph under such a definition is equivalent to finding all the cliques in a graph, a NP-hard problem. A metric that considers only $E(C)$ will be optimized by any disconnected component of the graph, including a community that includes the entire graph. While, it is possible to find all disconnected components in linear time, it also provides no information about reasonable datasets.

While these metrics are simple and easily understood by their parameterization, not all metrics have a closed form parameterization. We can push a step further and obtain a general methodology for gaining understanding of what communities certain metrics prefer. The methodology will be to use a simple greedy algorithm to incrementally improve a community according to a metric. The progress of the algorithm is then tracked within the (I, E) plane. While these paths may even seem random. We can visualize how a metric categorizes communities in the (I, E) plane with level sets. The progression of communities chosen by the metric can then be seen as a progression through categories of communities that optimize the metric.

Metric	Optimal C	$(I(C), E(C))$
CONDUCTANCE	G	$(x, 0)$
CUT RATIO	G	$(x, 0)$
EDGES CUT	G	$(x, 0)$
EXPANSION	G	$(x, 0)$
INTERNAL DENSITY	any clique	$(1, x)$
VOLUME	G	$(x, 0)$

Table 2.1: Communities that optimize each metric. A value of x, indicates that the optimization is independent of that value.

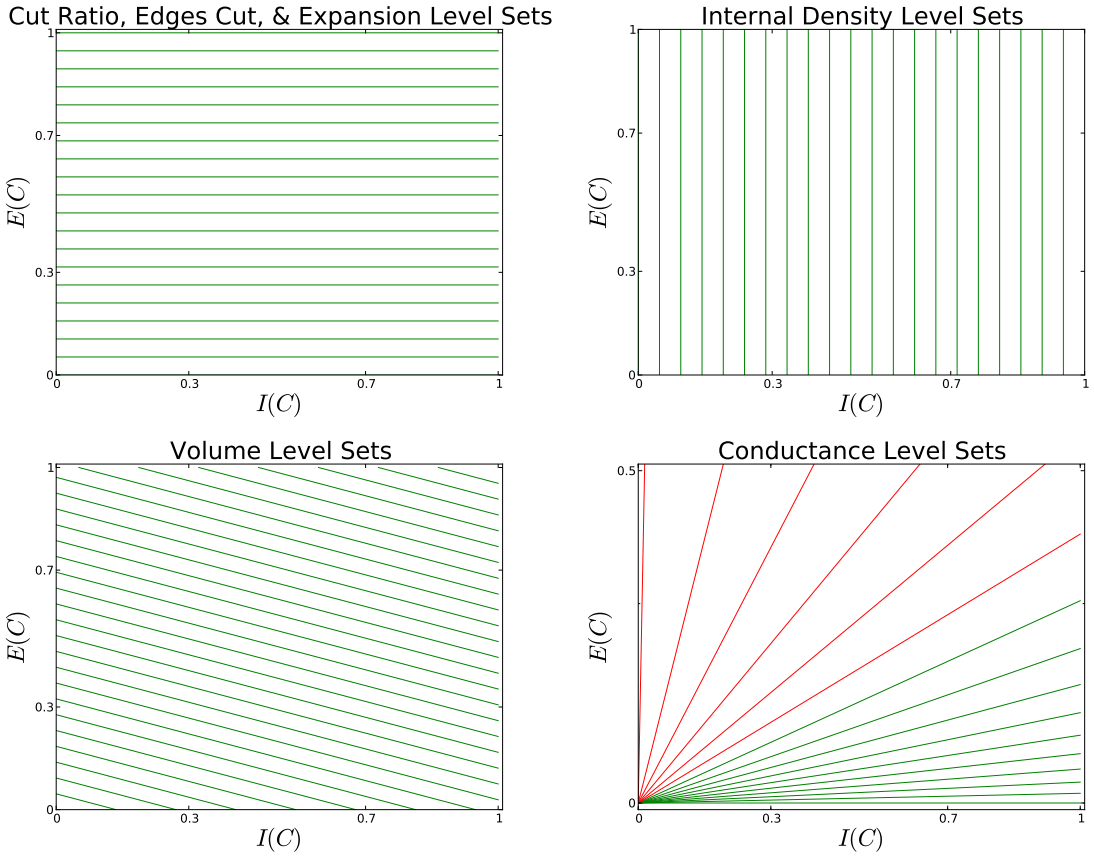

Figure 2.1: Level Sets in the (I, E) plane for different metrics of a single community.

Greedy Algorithm

The simple greedy algorithm is an expansion of the community, greedily engulfing the node that improves a given METRIC the most. The algorithm is the *Greedy Single Community Metric Optimization* Algorithm: 1.

Some metrics require minimization rather than maximization, this algorithm is easily adapted accordingly. For each metric we start the algorithm with a simple subset of two connected nodes and track the communities that result in improvements, according to a given metric. As the algorithm expands the com-

Algorithm 1: GREEDY SINGLE COMMUNITY METRIC OPTIMIZATION

Input: C, $G = (V, E)$, and METRIC

$inc = 1$

while $inc \geq 0$ and $C \neq V$ **do**

Let $u \in V$ maximize METRIC($C \cup u$).

$inc \leftarrow$ METRIC($C \cup u$)$-$ METRIC(C)

$C \leftarrow C \cup u$

end while

return S

munity, the metric determines which node is optimal at each step. We record this better community and for each metric, M, get a set of $C_1 \subset C_2 \subset \ldots \subset C_n$, such that the metric is increasing $M(C_i) < M(C_{i+1})$. While this is not an encompassing algorithm as it can only find local maximums, it will reveal a metrics biasenesses toward communities of certain $I(C)$, $E(C)$, values.

Expansion, Edges Cut, and Cut Ratio

Let us look at how $E(C)$ based metrics categorize the (I, E) plane. The metrics in this category are EXPANSION, EDGES CUT, and CUT RATIO. We can visualize how the plane is viewed by these metrics with level sets. For all points of the I, E plane, we evaluate the metric and graph the contour curves, or level sets[]. In level set figures, any two points in the I, E plane connected by a curve have the same metric value. In our greedy algorithm1, if the algorithm can add a node to the community that crosses a level set to a higher metric valuation the algorithm will add that node. Visually, the more level sets crossed by a change

to the community, corresponds to a higher change in the metric. Traditionally, level sets are used in this manner to show gradient descent to find a local minimum. The optimum that a gradient descent will find, can be found by traveling perpendicular to the level sets. While we find in practice this is a good analogy to understand the behavior of optimizing these metrics, we can not quite finish the analogy as the metrics are discrete. Hence, we can not quite use level sets to compute communities. But, we can use level sets as a visual categorization of the (I, E) plane according to a metric.

The level sets of cut ratio, edge cuts, and expansion, figure 2.8, across the I, E plane reveal the I, E values of the communities that optimize these metrics. While, the three metrics have different definitions, their level sets are identical only with different values. The level sets are horizontal lines, whose metric's are optimized by improvements in $E(C)$, irregardless of changes to $I(C)$. Accordingly, the node that has the greatest improvement in $E(C)$ is the same for all three metrics, and each metric results in the same path through the I, E plane.

Though it does not make a difference in our examples, the difference between these metrics is their treatment of $k = \frac{|C|}{|V|}$. Cut ratio is unresponsive to changes in the size of the community, while expansion linearly discounts against larger communities. Edges cut heavily favors very large or very small communities. See *Fig. 2.2*

Internal Density as previously defined

On the polar extreme of cut ratio, edge cuts, and expansion, is internal density. Internal density is a function of internal density and is unresponsive to changes

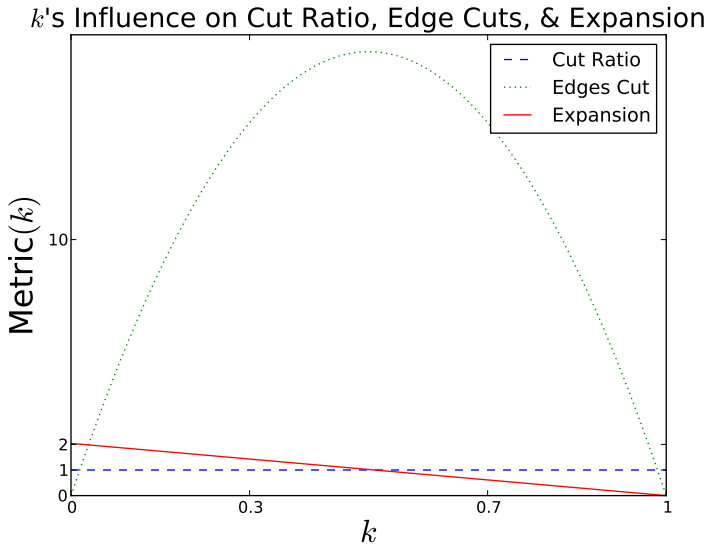

Figure 2.2: Influence of size of community on the values of external density based metrics.

in the the external density. We do not include indepth analysis, but rather a summary. The level sets of internal density are vertical lines in the I, E plane. The paths produced by the greedy algorithm grow out the original community to the largest clique it can find(if forced to), as the internal density of two connected notes is already maximal. Any clique is an optimization of internal density, irregardless of external connectivity.

Volume

A metric that takes both internal and external density into account is volume. The next conclusion is not apparent just from the equation parameterized in terms of internal and external density. However, observing the level sets of volume reveal that the optimal community is at $(I, E) = (0, 0)$ and volume as a metric is optimal for communities with low external density and low inter-

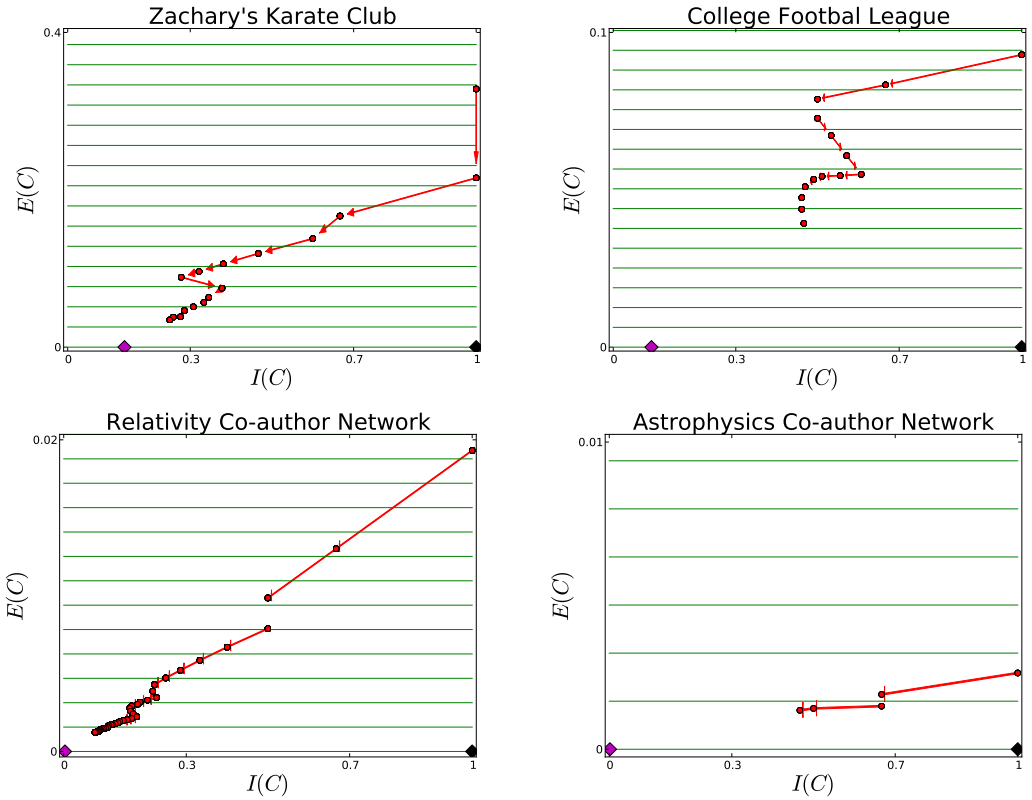

Figure 2.3: External Density based metrics(CUT RATIO, EDGES CUT, and EXPANSION) optimized in different networks. The lower left diamond is the $(I(C), E(C))$ point corresponding to a community of the entire graph. The lower right diamond is the $(I(C), E(C))$ point corresponding to an ideal community. The path corresponds to the intermediatory (I, E) values of adding nodes that optimize the metrics using greedy algorithm 1

nal density. Apart from communities of disconnected nodes, volume can best be optimized by a community encompassing the entire graph. Volume directly contradicts our intuition that communities should have good internal connectivity.

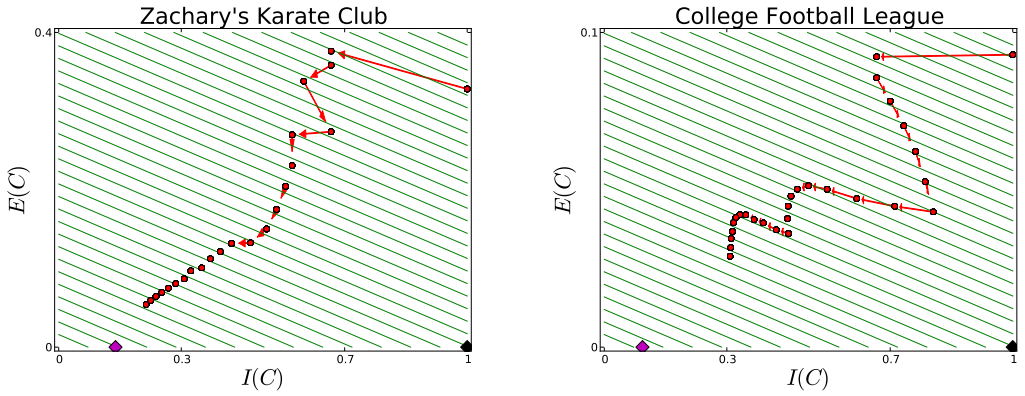

Figure 2.4: Tracing of communities found by volume through the IE plane. The paths are much longer, extending until the community includes the entire graph. We cut the paths short once they progress towards including the entire community. The lower left diamond is the $(I(C), E(C))$ point corresponding to a community of the entire graph. The lower right diamond is the $(I(C), E(C))$ point corresponding to an ideal community. The path corresponds to the intermediatory (I, E) values of adding nodes that optimize the metrics using greedy algorithm 1.

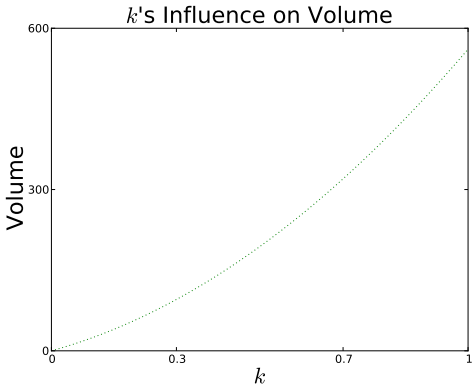

Figure 2.5: The affect, increasing the size of the community has on volume, even for a constant $I(C)$ and $E(C)$.

Conductance

For conductance the level sets are rays radiating from $(I, E) = (0, 0)$, see *Fig. 2.1*.

As the rays become closer to horizontal, $E = 0$, conductance becomes optimal.

Hence, no matter the data set, conductance desires communities with I, E values closer to $(I, E) = (x, 0)$, for any arbitrary $I(C) = x$. Including more nodes that continually increase $I(C)$ is rare in networks and much more common is decreasing $E(C)$ by encompassing more of the graph. Since, conductance is fairly, unresponsive to changes in $I(C)$, optimizing conductance results decreasing $E(C)$ as much as possible. How much conductance is biased towards small improvements in $E(C)$ verse large improvements in $I(C)$ depends on where in the I, E domain the seed communities originate. In the College Football League, the seed community begins a portion of the domain where improvements in I and E are balanced. For Zarchary's Karate Club, the seed community quickly falls into a portion of the I, E domain where small and easy improvements in E benefit conductance more than large and hard changes in I.

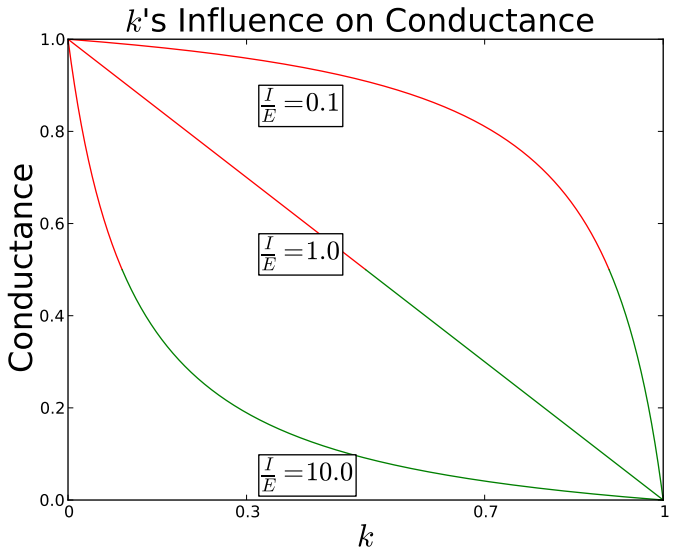

Figure 2.6: Influence of size of community on the value of conductance.

The last parameter that must be taken into account is k, the proportion of the graph covered by the community. Now we fix the I, E ratio and observe how changes in $|C| = k|V|$ affect conductance. The conclusion, If the expansion of

the community is greatly affected by k, then our previous analysis will yield to conductance favoring the largest possible community as demonstrated in *Fig. 2.6*. This affect can be seen in the change of C_1 to C_2 in the plotted paths in *Fig.* **??**, as the relative size change of adding just one node is large.

That conductance favors large disconnected communities, even with low internal density, is experimentally confirmed by []. However, as long as the community is of medium size and has a much larger $I(C)$ value than $E(C)$ conductance will make improvements that correspond to our intuition that an ideal community is internal dense and externally sparse.

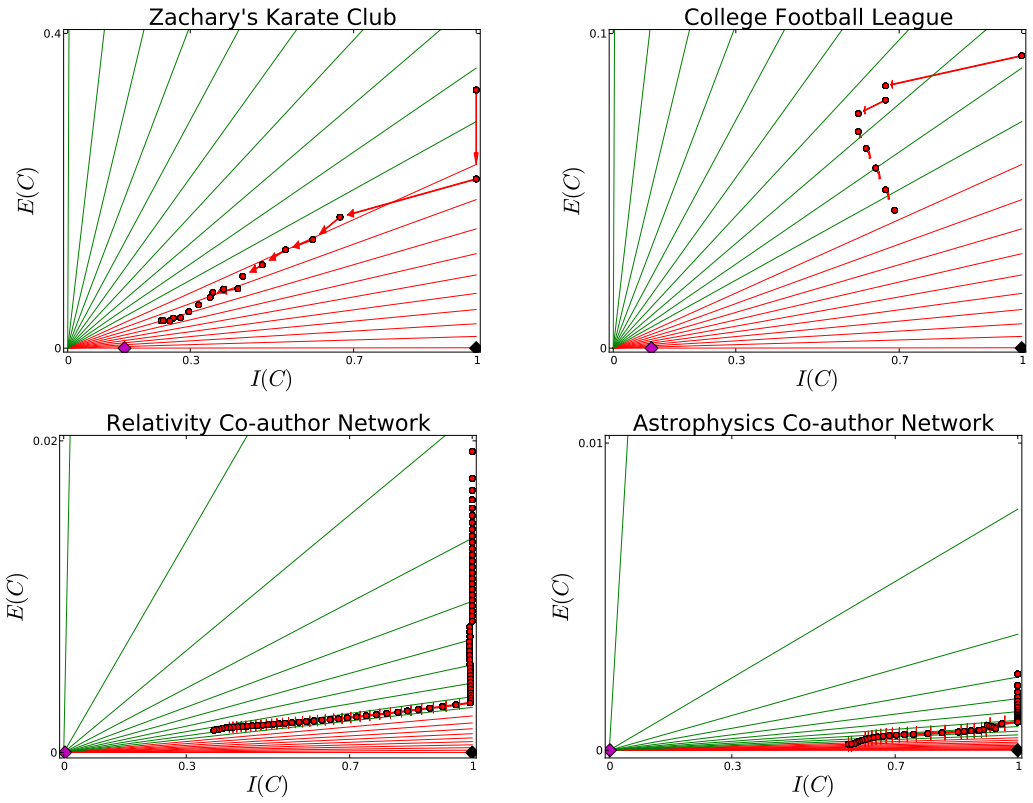

Figure 2.7: The progression of communities that optimize conductance. Note, the entire graph optimizes conductance, we stop following the progression of conductance once it becomes obvious the entire graph will be engulfed. (In the case of the college football league, a local optimum was reached.)

2.3 Set of Communities Based Metrics

We now explore metrics that evaluate the strength of a set of communities. A majority of these community detection methods are based on finding a partitioning of the network into communities. For a given network, community detection produces a set of communities, S, where each node u belongs to exactly one community, $C_i \in S$. The most popular of these detection methods is, maximizing modularity []. Modularity returns a single valued evaluation of a set of communities.

As with individual community based metrics, we will develop an understanding of the pertinent parameters of a set of communities. Through these parameters, there exist some closed form parameterizations for metrics on sets of communities. Primarily, though, we will build the space of these parameters that a set of communities may evaluate to and analyze how these metrics valuate communities of different parameters. Again, our conclusions are independent of a specific network.

2.3.1 Internal Density, External Density, and Conciseness

Our parameterization of internal and external density for single community metrics can not be directly applied to a set of communities, $S = \{C_1, C_2, ...C_n\}$. Though if we follow the same logic we will arrive at a similar parameterization. An ideal set of communities are cliques such that every edge is within some community. In addition, the community description of the network should not have an exponential number of communities, but rather some concise set of

communities. Hence an ideal set of communities has three parameters. Internal density is a representation of how close the set of communities is to being a set of cliques. External density is a representation of how close the set of communities are to covering all edges in the graph. Size of the set of communities is a representation of how concise the set of communities are. With the same methodology for parameterizing and understanding metrics of individual communities we proceed to parameterize metrics for sets of communities with *internal density*, *external density*, and *conciseness*. Formal definitions follow.

Definition 7 (Internal Density of a Set of Communities) *For a set of communities, $S = \{C_1, C_2, \ldots, C_n\}$, the internal density of the set is ratio between the sum of the number of edges that do exist within each community to the maximal value of that sum. The maximal value is achieved by all communities being cliques.*

$$I(S) = \sum_{C \in S} \frac{\sum_{u \in C} \sum_{v \in C} w(u, v)}{\sum_{C \in S} |C|(|C| - 1)} \tag{2.9}$$

Definition 8 (External Density of a Set of Communities) *In a set of communities S, the* EXT_EDGES *are the edges, not covered by any community. Hence, external density is the fraction of edges not covered by any community.*

$$E(S) = \frac{\sum_{(u,v) \in \text{EXT\_EDGES}} w(u, v)}{\sum_{u,v \in V} w(u, v)} \tag{2.10}$$

Definition 9 (Conciseness of a Set of Communities) *Conciseness is the size of the description necessary to describe all of the communities.*

$$\text{CONCISENESS}(S) = |S| \tag{2.11}$$

An useful byproduct of our choice of defining the parameters, is the ability to analyze, any set of communities, including overlapping communities. Our definition of internal density for a set of communities, allows nodes to be placed in multiple communities, but insists that a high internal density corresponds to a node being well connected to all communities it belongs to. External density is independent of overlapping communities, as well as conciseness.

To date we do not know of any previous metrics that have a closed form parameterization in terms of $I(S)$, $E(S)$, and $|S|$. We release a linear metric pertaining to these parameters in a later section.

Definition 10 (Ideal Set of Communities) *A set of communities, S, is ideal if it is a set of cliques that cover the graph in the fewest necessary communities:*

$$\begin{aligned} I(S) &= 1 \\ E(S) &= 0 \\ |S| &= 1. \end{aligned}$$

All three parameters are necessary to ensure a complete description of a set of communities. For any two parameters, there exists a set of communities that can maximize those two parameters, but not the third parameter, revealing an undesired property of the set of communities. Figure 2.8, illustrates the types of communities that can optimize for any two parameters.

2.3.2 Study of Relevant Metrics

Modularity is a partitioning of the network, and each partition is deemed a community. For each module its modularity is the difference of the existing internal

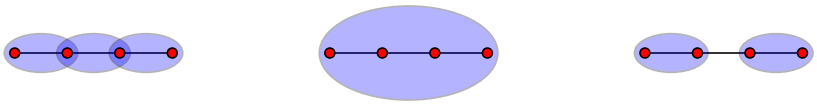

Figure 2.8: The communities that optimize 2 out of 3 parameters. Nodes are in red, lines are edges, and communities are blue ellipses. The left community configuration optimizes $I(S) = 1$ and $E(S) = 0$, but not conciseness at $|S| = 3$. The middle configuration optimizes $E(S) = 0$, $|S| = 1$, but not internal density at$I(S) = \frac{1}{2}$. The right configuration optimizes $I(S) = 1$ and nearly conciseness at $|S| = 2$, but does not optimize external density at $E(S) = \frac{1}{3}$

edges and expected number of internal edges, had the graph been random, with the same distribution of node degrees. The total modularity is then the sum of the modularity of each partition. Modularity was developed by [] and has found wide spread use due to the fast algorithms for maximizing modularity [**?**]. In particular, the use of dendograms in the Louvain Algorithm [**?**] runs in minutes for large networks.

There is not a closed form parameterization of modularity in terms of our definitions of $I(S)$, $E(S)$, and $|S|$. But we find $I(S)$ and $E(S)$ to be intuitive enough to provide the space in which to analyze metrics of sets of communities.

Still, we would like to be able to use level sets for some visual explanation of how modularity behaves with regard to some internal and external density. The parameterization of modularity we can provide is in terms of $I(C)$ and $E(C)$

for single communities. If we allow, $p = \frac{|C|(|C|-1)}{2L}$ and $q = \frac{|C|(|V|-|C|)}{2L}$ to be constants:

$$\text{MODULARITY}(S) = \sum_{C \in S} pI(C) - (pI(C) + qE(C))^2 . \tag{2.12}$$

We can draw loose conclusions from this parameterization. The first check is that a set of disjoint cliques are the only communities that optimize modularity. There are no counter intuitive communities that optimize modularity. Overall, modularity is a two part optimization. For seed communities, in the case of modularity individual nodes, the typical initial values are high in external density. When $E(C)$ is high it is the dominating term and the optimization is to decrease $E(C)$ as quickly as possible. Once, external density is not the dominating term the optimization is then center around trying to improve $I(C)$. The transition between these two phases of optimization is sudden as attested by the level sets.

We would like to recommend modularity should be designed to optimize for improvements in internal density first to find the dense cores of communities and then optimize over external density to cover more of the graph. The reasoning is that improvements in external density can be made at any time by increasing the size of communities. However, improvements in internal density can only be made while the communities are small and closer to cliques.

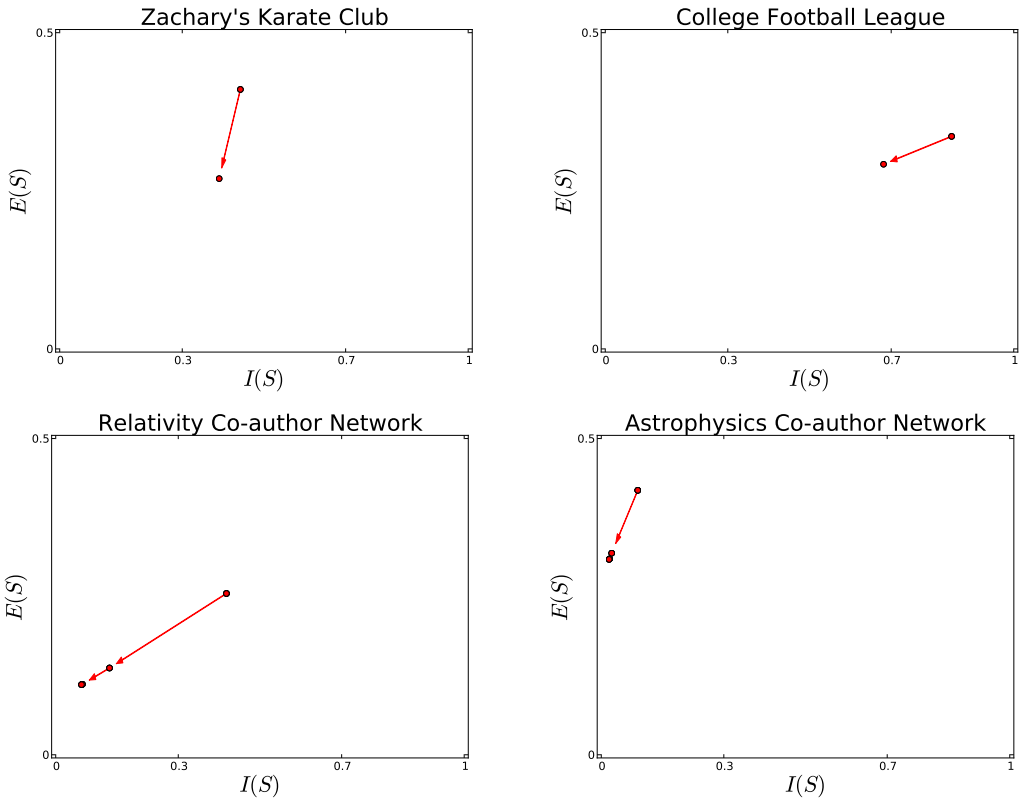

Figure 2.9: Here we plot the $(I(S), E(S))$ values produced by each level of the dendogram stages in maximizing modularity with the Louvain algorithm [].

CHAPTER 3

A NEW METRIC: LINEARITY

Through our understanding of how existing metrics handle the balance between internal and external density

3.1 Single Community Detection

We maintain that the variables of internal and external density incorporate our full intuition of communities, metrics are just a matter of balancing between the two. We have shown how previous metrics single communities balance the two or select one to optimize for. We propose a linear metric that transparently balances between the two variables.

Definition 11 (Linearity) *Our metric for single communities:*

$$\text{LINEARITY} = M_L(C) = aI(C) - bE(C) \tag{3.1}$$

Many metrics incorporated the size of the community into their weighting. Indeed, it is more significant for a large community to gain nearly the same internal density as small community. Accordingly, there is a generalization of LINEARITY that can account for any desired weighting between internal and external density, with weighting for any size of community.

Definition 12 (General Metric) *Our metric for single communities in its greatest generality:*

$$\text{GENERAL} = M_G(C) = \sum_{i=0} f_i(C)I(C)^i - g_i(C)E(C)^i \tag{3.2}$$

Where f_i and g_i can be any function of the size of a community.

Linearity behaves similarly to conductance, when conductance is in a region of the I, E plane where there is a fair balance between improvements of I and E. Unlike conductance though, linearity does not have a critical point in the I, E plane, where the balance is shifted towards only favoring improvements in E. The result is that with the same initial seeds in the Karate Club and CFL, linearity follows the same path as conductance for the first few expansions and stops, rather than engulfing the network.

3.1.1 Results

Theorem 3.1.1 (Single Community Optimization) *If a single metric is optimized and looses $delta_I$, it will not gain it back*

3.2 Multiple Community Detection

We now present a linear metric for sets of communities with regard to $I(S)$, $E(S)$, and $|S|$. It follows the same intuition of creating the linear metric for a single community. The ideal community is located at $(I(S), E(S), |S|) = (1, 0, 0)$ and accordingly the level sets are parallel planes emanating from around the ideal community. Hence, the linearity of the metric. How to balance between improvements in each of the parameters is set by the user.

Definition 13 (Linearity) *Our metric for a set of communities:*

$$\text{LINEARITY}(S) = aI(S) - bE(S) - c|S|, \tag{3.3}$$

where $a, b, c \geq 0$.

Depending on the application, communities of certain characteristics may be desired. Communities of size smaller than a certain size may be desired to be penalized or trade offs between improvements in $E(S)$ and $I(S)$ may depend on existing values. A general form of this equation is available.

Definition 14 (General Metric) *Our metric for single communities in its greatest generality:*

$$\text{GENERAL}(S) = \sum_{i=0} f_i(S) I(S)^i - g_i(S) E(S)^i - h_i |S|^i \tag{3.4}$$

3.2.1 Results

CHAPTER 4

PARALLEL COMMUNITY DETECTION

4.1 Introduction of Properties and Statistical Significance

4.2 Algorithm

4.2.1 Seeds

4.2.2 Expansion

4.3 Probability of Correctness

4.4 Performance

CHAPTER 5

CASE STUDIES OF NETWORKS

5.1 Amazon Product Network

5.2 Collaboration Networks

5.2.1 Astrophysics

5.2.2 Condensed Matter

5.2.3 High Energy Physics

5.2.4 General Relativity

5.3 Enron Email Network

5.4 Epinions Social Network

5.5 Gnutella P2P Network

5.6 Physics Citation Network

5.7 Web Graphs

5.7.1 Berkeley Webpage

5.7.2 Google

CHAPTER 6

EVOLUTION OF COMMUNITIES

CHAPTER 7

CONCLUSIONS

Above we have provided an indepth look at the details. Here we provide the summation of our results.

Finding communities is always a tradeoff. In metric based approaches between internal density and external sparsity. In significance based approaches the tradeoff is between specificity and sensitivity.

The number of communities a node belongs to follows a power law distribution.

Communities in citation networks evolve from a unioning of previous topics. However, not all papers that union topics produce successful communities.

[7] [21] [12] [20] [6] [2] [19] [11] [22] [1] [3] [13] [5] [17] [10] [18] [14] [9] [16] [4] [15] [8]

APPENDIX A

CHAPTER 1 OF APPENDIX

Appendix chapter 1 text goes here

BIBLIOGRAPHY

[1] R. Andersen, F. Chung, and K. Lang. Local graph partitioning using pagerank vectors. 2006.

[2] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *J. Stat. Mech.*, JTODO, 2008.

[3] A. Cappocci, V.D.P. Servedio, G. Calarelli, and F. Colaiori. Detecting communities in large networks. *Physica A*, 352:669–676, 2005.

[4] D. Chen, Y. Fu, and M. Shang. A fast and efficient heuristic algorithm for detecting community structures in complex networks. *Physica A*, 388:2741–2749, 2009.

[5] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, JTODO, 2005.

[6] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72(027104 JTODO), 2005.

[7] S. Fortunato. Community detection in graphs. *Physics Reports*, 486:75–174, November 2010.

[8] M. Girvan and M. E. J. Newman. Community structure in social and biolgical networks. *PNAS*, 99(12):7821–7826, June 2002.

[9] M. B. Hastings. Community detection as an inference problem. *Archive JTODO*, 2006.

[10] P. Hui, E. Yoneki, S. Chan, and J. Crowcroft. Distributed community detection in delay tolerant networks. Kyoto, Japan, August 2007. MobiArch.

[11] A. Jain. Data clustering: 50 years beyond k-means. 2008.

[12] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad, and spectral. *Journal of the ACM*, 51(3):497–515, May 2004.

[13] A. Lancichinetti and S. Fortunato. Community detection algorithms: A comparative analysis. *Physical Review E*, 80(056117 JTODO), 2009.

[14] A. Lancichinetti, S. Fortunato, and F. Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(046110 JTODO), 2008.

[15] A. Lancichinetti, M. Kivela, J. Saramaki, and S. Fortunato. Characterizing the community structure of complex networks. *PloS ONE*, 5, August 2010.

[16] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney. Statistical properties of community structure in large social and informatino networks. Beijing, China, April 2008. WWW 2008.

[17] J. Leskovec, K. Lang, and M. Mahoney. Empirical comparison of algorithms for network community detection. Raleigh, NC, April 2010. WWW 2010.

[18] A. Maiya and T. Berger-Wolf. Sampling community structure. Raleigh, NC, April 2010. WWW 2010.

[19] N. Mishra, R. Schreiber, I. Stanton, and R. Tarjan. Finding strongly-knit clusters in social networks. *Internet Mathematics*, 5:155–174, 2009.

[20] M. Newman. Modularity and community structure in networks. *PNAS*, 103(23):8577–8582, 2006.

[21] M. Porter, J. Onnela, and P. Mucha. Communities in networks. *Notices of the AMS*, 56(9):1082–1097, October 2009.

[22] S. Zhang, R. Wang, and X. Zhang. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A*, 374:483–490, 2007.