

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

© 2012 June Andrews
ALL RIGHTS RESERVED

COMMUNITY DETECTION IN LARGE NETWORKS

June Andrews, Ph.D.

Cornell University 2012

Graphs are used to represent various large and complex networks in scientific applications. In order to understand the structure of these graphs, it is useful to treat a set of nodes with similar characteristics as one community and analyze the community's behavior as a whole. Finding all such communities within the graph is the object of community detection. In our research, we compare dozens of existing community detection methods and develop a new class of algorithms for finding communities.

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 completing her doctoral degree in Applied Mathematics at Cornell University.



Figure 1: *Phil Andrews 1955 - 2011*

Here's to you Da.

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

Biographical Sketch	iii
Dedication	iv
Acknowledgements	v
Table of Contents	vi
List of Tables	vii
List of Figures	viii
1 Introduction	1
1.1 Community Detection	1
1.2 Graph Partitioning Methods	3
1.2.1 Top Down Approaches	3
1.2.2 Bottom Up Approaches	5
1.3 Overlapping Community Detection	6
1.3.1 Alpha Beta Clustering	6
1.4 More Approaches	8
1.5 Desired Improvements	8
1.6 Notation	9
2 A Framework for the Comparing Metric Based Detection Methods	11
2.1 Previous Comparisons	11
2.2 Individual Community Based Metrics	12
2.2.1 Internal and External Density	13
2.2.2 Study of Relevant Metrics	15
2.3 Set of Communities Based Metrics	25
2.3.1 Internal Density, External Density, and Conciseness	25
2.3.2 Study of Relevant Metrics	27
3 A New Metric: Linearity	31
3.1 Single Community Detection	31
3.2 Multiple Community Detection	33
4 Parallel Community Detection	37
4.1 Introduction of Properties and Statistical Significance	37
4.2 Algorithm	37
4.2.1 Seeds	37
4.2.2 Expansion	37
4.3 Probability of Correctness	37
4.4 Performance	37

5	Case Studies of Networks	38
5.1	Collaboration Networks	38
5.1.1	Karate Club	38
5.1.2	Astrophysics	39
5.1.3	Condensed Matter	39
5.1.4	High Energy Physics	39
5.1.5	General Relativity	39
5.2	Enron Email Network	39
5.3	Epinions Social Network	39
5.4	Gnutella P2P Network	39
5.5	Physics Citation Network	39
5.6	Web Graphs	39
5.6.1	Berkeley Webpage	39
5.6.2	Google	39
5.7	Wiki Network	39
5.7.1	Communication Network	39
5.7.2	Election Voting Network	39
5.8	Amazon Product Network	39
6	Evolution of Communities	40
7	Conclusions	41
A	Chapter 1 of appendix	42
	Bibliography	43

LIST OF TABLES

1.1	A subset of applications and one of their preferred community detection methods.	2
1.2	Notation	10
1.3	Introduced Functions	10
2.1	Communities that optimize each metric. A value of x , indicates that the optimization is independent of that value.	18

LIST OF FIGURES

1	<i>Phil Andrews 1955 - 2011</i>	iv
1.1	A simple graph of people and their friendships. The graph is regular enough to reveal two communities.	1
2.1	Level Sets in the (I, E) plane for different metrics of a single community. There are four ways the (I, E) space is categorized.	17
2.2	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 intermediary (I, E) values of adding nodes that optimize the metrics using Greedy Algorithm ??	20
2.3	Influence of size of community on the values of external density based metrics.	21
2.4	Tracing of communities found by volume through the IE plane for a maximum of 100 steps. 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 intermediary (I, E) values of adding nodes that optimize the metrics using Greedy Algorithm ??	22
2.5	The affect, increasing the size of the community has on volume, even for a constant $I(C)$ and $E(C)$	22
2.6	The progression of communities that optimize conductance. Note, both the entire graph and the ideal community optimize conductance. In the relativity and astrophysics networks, we stop following the progression of conductance once it becomes clear the entire graph will be engulfed. (In the case of the college football league, a local optimum was reached, but reports an undesirable value of conductance..)	24
2.7	Influence of size of community on the value of conductance. The object is to minimize conductance	25
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 conciseness at $ S = 1$, but does not optimize external density at $E(S) = 1$	28

2.9	The level sets of how MODULARITY treats the $I(C)$, $E(C)$ space for one community of size 9 in the CFL. Note the sharp transition from a region that heavily favors improvements in external density to a region that heavily favors improvements in internal density($E(C) < 0.1$).	29
2.10	Here we run the Louvain Algorithm [2] to maximize modularity. The $(I(S), E(S))$ path is each level of the dendrogram. The $(I(G), E(G))$ value for the entire graph is the diamond in the lower left. In the general relativity and astrophysics co-author networks, modularity does not present much of an improvement over $I(G)$ and has a much higher $E(G)$ value.	30
3.1	The two plots are boring and predictable.	33
3.2	Our linearity metric is tested exactly as previous single community metrics have been with the Greedy Algorithm ???. Notice that the paths are truncated paths of Conductance in Figure ???. JTODO add in all the final spots from previous metrics, also change b so that the final communities are more valued than the entire graph.	34
3.3	Compared to Known Solutions: For the karate club we find the two communities, but then expansion. For the college football league we find 13 communities, rather than the real 12, but this results in a better value in internal and external density. In the karate and football the black star corresponds to the value of known community solutions.	35
3.4	Linearity is in red. The (I, E) values modularity produces are inserted with a black square In relativity there are no known communities. For now we just show that good avlues of internal and external density can be found.	36

CHAPTER 1

INTRODUCTION

1.1 Community Detection

Consider an application that studies objects and the interactions between those objects. The application could study anything from people and their friendships, to papers and their citations; a variety of applications fall into this format. If we let nodes represent the objects and edges represent the interactions between those objects, we can store the application's data in a graph. While it can be possible for the application to draw conclusions by looking at every node within the graph, if the graph is large and complex, analyzing every node can be unmanageable and can produce incomprehensible results. We simplify the graph by finding communities of nodes. In particular, we want communities, whose members interact with each other in a particular way and interact with nonmembers of the community in a different way. If such a community is

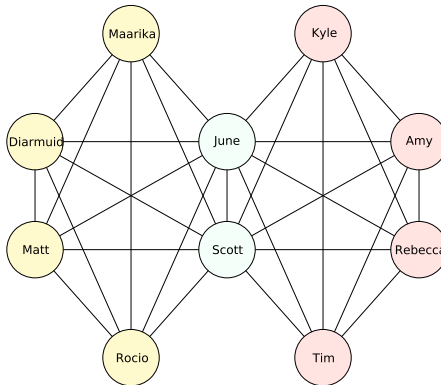


Figure 1.1: A simple graph of people and their friendships. The graph is regular enough to reveal two communities.

found, then two questions arise. How are members of the community related? How does the community interact with the rest of the graph? Given answers to these questions, we can comprehend what is happening in the graph at a local level. For social networks, we know that communities exist [?]. Due to the large, complex nature of social networks, communities can be hard to find. In order to find communities, we must develop the ability to see the forest through the trees. We have to be able to extract the communities of nodes from the interactions of the graph. This is the object of community detection.

Given a graph, there are two prominent questions community detection seeks to answer. The first, what is a community and the second, 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. Prior to 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 is 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 Propagation [?]
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

For many applications the object is to partition the graph into disjoint components. We call each component a community. There are an exponential number of possible partitions, but not every partition will provide useful information. While much analysis of useful or unuseful information must be left up to the application, there are two characteristics that most applications want in communities. The first is that nodes within a community be well connected. The second is that the community is not well connected to the rest of the graph. The definition of well connected is different for each community detection method.

There are two genres of finding good partitions of the graph, top down approaches that recursively cut the graph and bottom up approaches that union existing partitions.

1.2.1 Top Down Approaches

Top down approaches work by recursively dividing the graph, see Algorithm 1 for their structure. For methods in this category there are two necessary components. The first is the ability to tell if a set of nodes C is a community. The second, if a set of nodes is not a community, then the algorithm finds a way to divide the nodes, without splitting up any communities.

Algorithm 1: RECURSIVE PARTITIONING

Input: $G = (V, E)$

if V is a community **then**

return V

else

 divide V into C and $V - C$

return {RECURSIVE PARTITIONING(C), RECURSIVE PARTITIONING($V - C$)}

end if

Conductance

Conductance is a measure of a cut within the graph developed by JTODO [?]. For a given cut, if conductance is low, then there are relatively few edges crossing the cut. Intuitively, this implies that the cut does not divide a community. If further divisions do not improve conductance, then we have found a community.

$$\text{CONDUCTANCE}(C) = \frac{\sum_{u \in C, v \notin C} w(u, v)}{\sum_{u \in C, v \notin C} w(u, v) + \sum_{u, v \in C} w(u, v)} \quad (1.1)$$

While this algorithm is not in heavy use, conductance is used as a measure of whether or not other algorithms that cut the graph have split a community.

Betweenness and Centrality Measures

Betweenness and centrality measures were first presented by Givan and Newman [8]. The intuition is, if an edge lies between two communities, then several

shortest paths between nodes of the two communities will traverse the edge. We remove these edges to divide the network into components. When there are no preferential edges for shortest paths within a component, there are no more edges between communities, and the component is a community.

1.2.2 Bottom Up Approaches

Bottom up approaches work by unioning together subsets of nodes until the subset is a community. See Algorithm 2 for their structure. For algorithms in this category there are two components. The first is the determination of which subsets to union. The second is the determination of when a set of nodes is a community. To accomplish these, most bottom up approaches use a metric over the set of subsets. If no two subsets can be unioned to increase the metric then, every subset is a community.

Algorithm 2: RECURSIVE UNIONING

```

Input:  $S = \{C_1, C_2, \dots\}$ 
  if There exists  $C_i$  and  $C_j$ , such that  $C_i \cup C_j$  is a community then
    return RECURSIVE UNIONING( $\{S - C_i - C_j\} \cup \{C_i \cup C_j\}$ )
  else
    return  $S$ 
  end if

```

Modularity

The overwhelmingly popular metric in this category is modularity. Modularity was first presented by Newman [20]. The metric measures the distance between a provided set of communities and a randomly generated set of communities. Maximizing modularity finds the least random set of communities.

Fast algorithms have been developed for maximizing modularity. In this paper we use the Louvain Algorithm developed by Blondel et. al[2]. The same fast algorithm can be used for the similar metric, modularity ratio [?].

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 common. 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.

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 separately.

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 1.2 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. Here is the formal definition of an (α, β) community.

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 for this thesis. There are countless more methods. We briefly outline the most prominent of those methods.

- Kernighan-Lin Algorithm
- k -Clique Percolation
- Belief Propagation
- Hierarchy methods
- Principle Component 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.

1.6 Notation

We use the same notation throughout the thesis. A brief description of variables is listed in Table 1.2.

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 $u \in V$. We note that this assumption is not held in some of the literature we reference.
- *Edges* We presume that all edges exist and are weighted between 0 and 1. The edge weight function is $w : V \times V \rightarrow \mathbb{R}_{[0,1]}$ Unweighted graphs can easily be adapted into this notation.

We also introduce internal and external edges.

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 nonmember of C .*

Table 1.2: Notation

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

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
$\text{CONCISENESS}(S)$	Conciseness of a set of Communities, S , Definition 9

CHAPTER 2

A FRAMEWORK FOR THE COMPARING METRIC BASED DETECTION METHODS

Given the variety of community detection methods, we would like to know the differences and similarities between each. The experimental analysis from Lancichinetti and Fortunato [13] and Leskovec et. al [17] have given us some insights. Lancichinetti and Fortunato confirm that certain algorithms work better on certain networks. Leskovec and company analyze the communities produced by metric optimization algorithms. They find that as community size increases, the optimal communities diverge from our understanding of a strong community. We provide an explanation.

The networks used for comparison are described in depth in Chapter 5.

2.1 Previous Comparisons

Communities are defined by two characteristics, members of the community interact with each other in a particular way and interact with nonmembers of the community in a different way. Generally, a community is accepted as a set of nodes that interact frequently with each other and interact infrequently with nonmembers of the community. The first characteristic corresponds to a high internal density, see Definition 4. The second corresponds to a low external density, see Definition 5. Detection methods vary in their formulation of density and their prioritization of finding communities with higher internal density verse lower external density.

All known comparisons have been experimental. The experiments are run

by first selecting a set of networks. Then, each detection method finds the communities within each network. Finally, using a set of metrics the communities found by each method are compared. Lancichinetti and Fortunato [13] compared three popular partition algorithms with generated graphs and used normalized mutual information as the comparison metric. Their results conclude that partition algorithms are fast and work well for non-overlapping communities. Leskovec et. al [17] conducted a broader study. They used eight classes of algorithms over 40 networks and compared the results with a series of metrics covered in this chapter.

2.2 Individual Community Based Metrics

Here, we explore metrics that evaluate the strength of a single community. There are three uses of such metrics. The first is for use in a Top Down (Section 1.2.1) or Bottom Up style (Section 1.2.2) style algorithm to find a partitioning of the network. The second is to find a single community within the network. The third is to compare communities found by complex detection techniques. The later use is more common for these metrics, of which conductance is the most popular [?].

The two components of a community are the internal and external density. Measuring both of these components requires a two dimensional space. The metrics used to compare communities are one dimensional. Each metric compresses the two dimensional space into a real number and loses information about the quality of the community. We visualize how each metric compresses the two dimensional space with the use of level sets. Given these level sets, we

can describe what communities optimize the metric. The conclusions from this analysis apply to all networks.

2.2.1 Internal and External Density

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 number of edges that exist between members of the community, internal edges, compared to the number of all edges that could exist within the community. Hence, $I(C) : C \rightarrow \mathbb{R}_{[0,1]}$, where*

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

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 $I(C) = 1$. The closer a community, C is to an *internal density* value of 1, the close it is to being a clique. If the graph is indeed a weighted graph, where w has values between 0 and 1, the same evaluation applies. In weighted graphs, *internal density* reflects how close a community is to being a maximally weighted clique.

Definition 5 (External Density) *External density is the number of edges that exist between a member of the community and a nonmember of the community, external edges, compared to all possible edges that could exist leaving the community:*

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

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 $E(C) = 0$.

There are other representations of $I(C)$ and $E(C)$ that vary how the $|C|$ and $|V|$ terms are used. The analysis and conclusions that follow are not sensitive to such variations.

With our parameterization, all communities can be mapped to a point $(I(C), E(C))$ in the square $\mathbb{R}_{[0,1]} \times \mathbb{R}_{[0,1]}$. Communities with certain values do not correspond to our understanding of a strong community. Such values are $(0, \frac{1}{2})$, $(\frac{1}{2}, 1)$, etc. However, a community mapped to $(\frac{1}{2}, 0)$ corresponds to a strong community. We define the strongest possible community to be ideal.

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

$$I(C) = 1$$

$$E(C) = 0.$$

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 allow the approximation. Additionally, we introduce variable, k , representing the portion 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 [?].

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

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

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

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

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

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

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

- INTERNAL DENSITY as a metric, previously existed before our definition of $I(C)$, [?]. However, we stick to our definition of $I(C)$ for intuitive reasoning

and note in previous work internal density represents the mirror image of our definition.

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

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

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

With this parameterization of the metrics, we can already draw some conclusions. 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 most datasets.

For the metrics that can be parameterized in terms of $I(C)$ and $E(C)$, we can use level sets. Level sets are a way to visually categorize a space. Let us pick the metric conductance. An optimal value of conductance is 0. We can find all points of $(I(C), E(C))$ (without knowing C) that evaluate to $\text{CONDUCTANCE}(C) = 0$. These points form a line in the (I, E) space. Now, we find all the points of $(I(C), E(C))$ that have a conductance value of $\text{CONDUCTANCE}(C) = \delta$. These points will also form a line in the (I, E) space. Because of the continuity of conductance, any community, C , that evaluates to an (I, E) point that lies between these two lines must have a conductance value of $\text{CONDUCTANCE}(C) \in (0, \delta)$. In this way we can visually categorize the space.

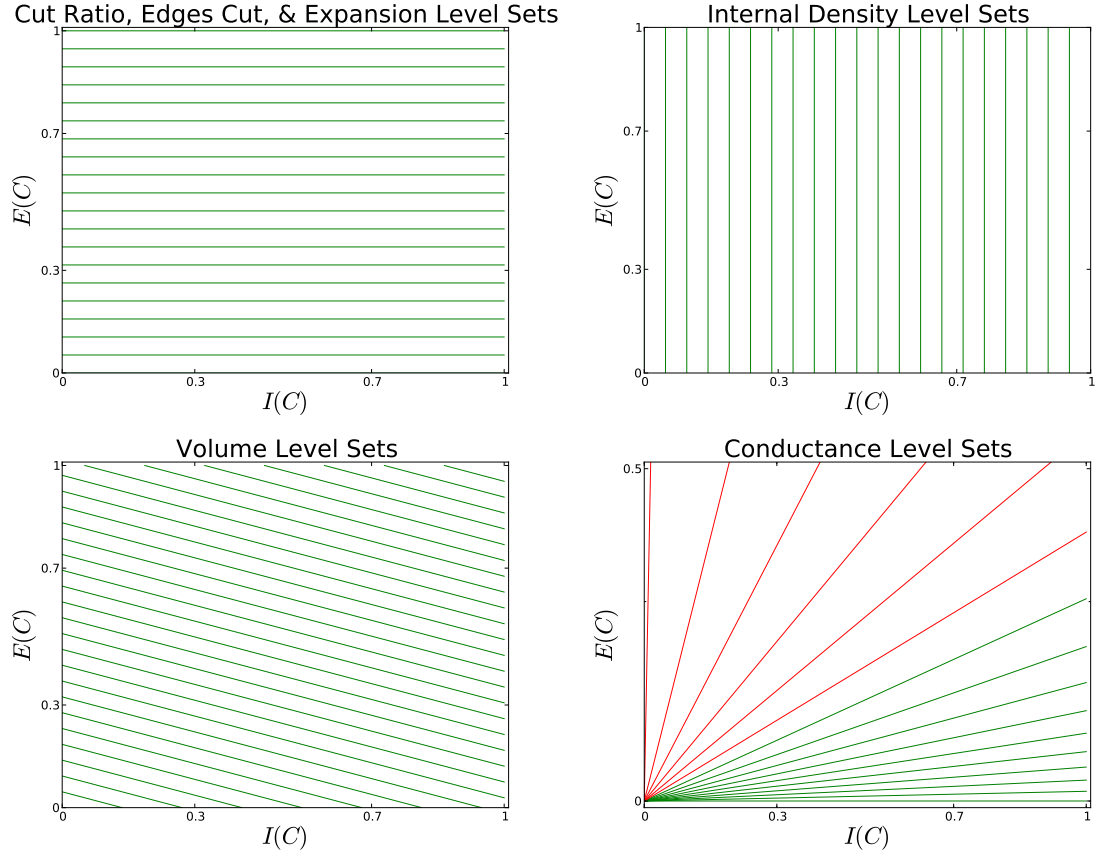


Figure 2.1: Level Sets in the (I, E) plane for different metrics of a single community. There are four ways the (I, E) space is categorized.

In level set figures, any two points in the I, E plane connected by a curve have the same metric value. In our Greedy Algorithm 3, 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 complete the analogy as the metrics are discrete.

While it is possible to draw conclusions now from the level sets, we proceed with finding communities based on these metrics. In doing so, we confirm and expand experimental results.

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.

Greedy Algorithm

The Greedy Single Community Metric Optimization Algorithm 3 takes as input a community and a metric. The algorithm then expands the community, one node at a time, until the metric can not be improved. The resultant community is a local optimum of the metric.

Some metrics require minimization rather than maximization, this algorithm can be adapted accordingly. In the following sections, we use the algorithm by starting with a metric and a subset of two connected nodes. The algorithm produces a series of nested communities, each with an increasing metric score. For each nested community, we compute their $(I(C), E(C))$. This gives us a path through the (I, E) plane. We can use level sets to explain the pattern of node selection that increases the metric.

Algorithm 3: 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 C

Expansion, Edges Cut, and Cut Ratio

We now consider metrics that are functions of $E(C)$ and not of $I(C)$: EXPANSION, EDGES CUT, and CUT RATIO. To understand these metrics we plot how they categorize the (I, E) plane with level sets and how iterations of the greedy algorithm choose communities in the (I, E) plane.

For these three metrics, their definitions vary, but their level sets are identical, as shown in Figure 2.1. The level set of $E(C) = 0$ corresponds to the metric's optimal set of communities. These communities that are disconnected from the rest of the graph, and can have an arbitrary internal density. These metrics favor decreases in external density over increases in internal density. In fact, for a community at any position in the (I, E) plane, the node that decreases external density the most will be chosen by the greedy algorithm, rather than a node that improves internal density. The effect of which is visible in the greedy algorithm's path through the (I, E) plane.

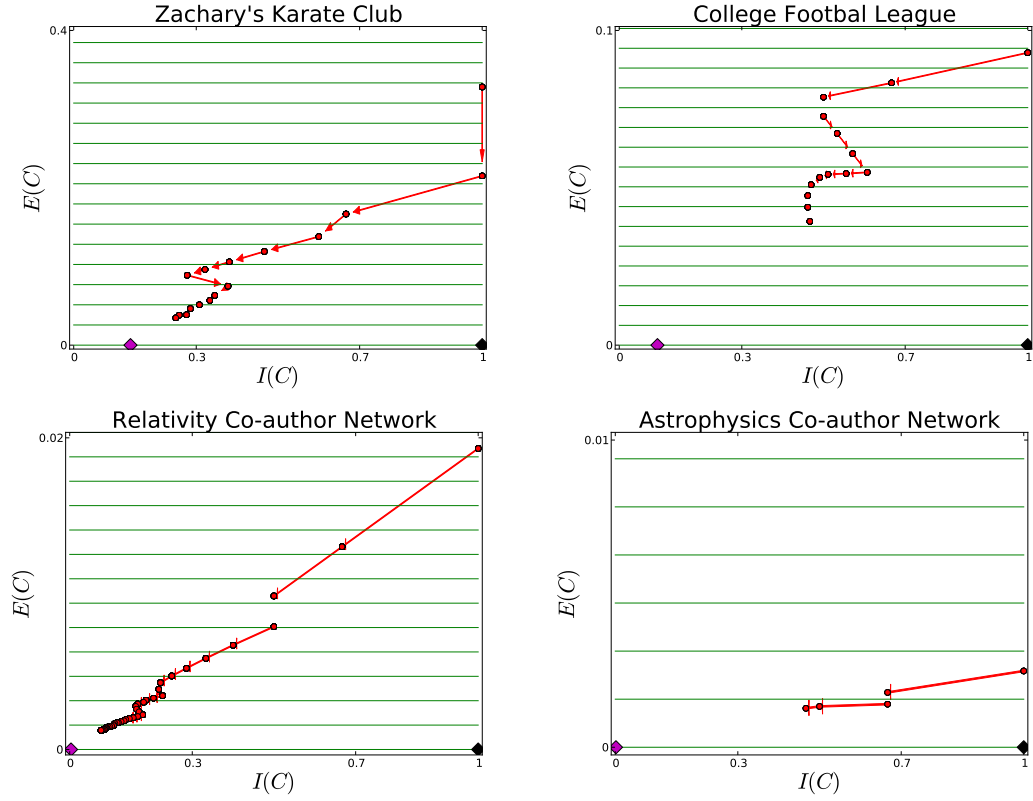


Figure 2.2: External Density based metrics(CUT RATIO, EDGES CUT, and EXPANSION) optimized in different networks. The lower left diamond is the $(I(C), E(C))$ point corresponding to a community of the entire graph. The lower right diamond is the $(I(C), E(C))$ point corresponding to an ideal community. The path corresponds to the intermediary (I, E) values of adding nodes that optimize the metrics using Greedy Algorithm 3

Because all of these metrics only respond to changes in external density, the order of nodes the greedy algorithm adds to the community does not vary between the three metrics. The difference between the three metrics is when they terminate. Termination in this case is determined by the size of the community, $k = \frac{|C|}{|V|}$. 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. 2.3.

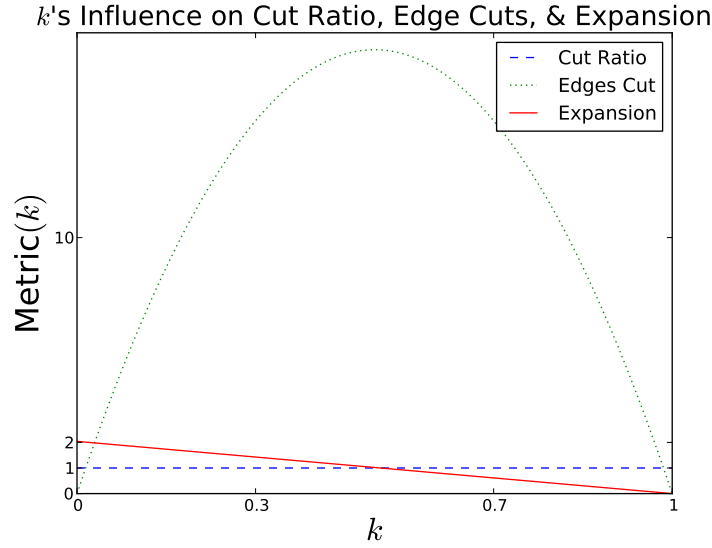


Figure 2.3: Influence of size of community on the values of external density based metrics.

Internal Density as Previously Defined

Internal density is a function of internal density and is unresponsive to changes in the the external density. Hence, only cliques and subsets of cliques optimize internal 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, as seen in Figure 2.1. The greedy algorithm augments our input of two connected nodes to the largest clique it can find(if forced to), as two connected nodes are already a clique.

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

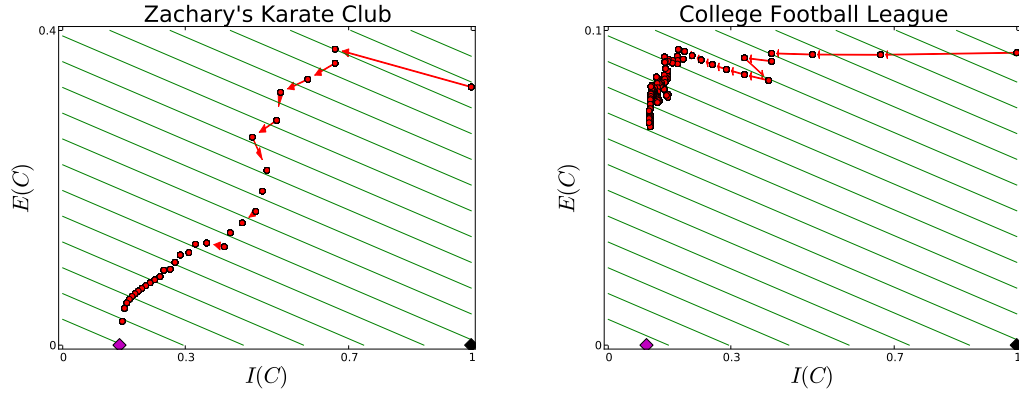


Figure 2.4: Tracing of communities found by volume through the IE plane for a maximum of 100 steps. The lower left diamond is the $(I(C), E(C))$ point corresponding to a community of the entire graph. The lower right diamond is the $(I(C), E(C))$ point corresponding to an ideal community. The path corresponds to the intermediary (I, E) values of adding nodes that optimize the metrics using Greedy Algorithm 3.

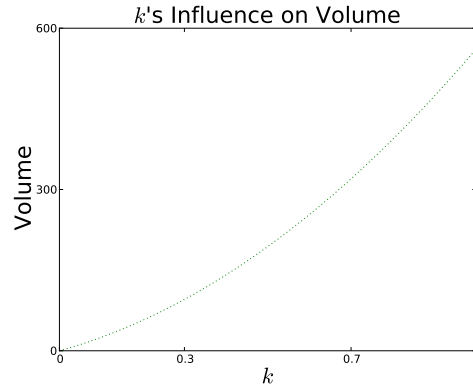


Figure 2.5: The affect, increasing the size of the community has on volume, even for a constant $I(C)$ and $E(C)$.

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

Conductance

For conductance the level sets are rays radiating from $(I, E) = (0, 0)$, see Fig. 2.1. As the rays come closer to horizontal, $E(C) = 0$, conductance is closer to optimal. Near $E(C) = 0$, changes to internal density have little effect on the value of conductance. Improvements in conductance come from modifying the community to decrease $E(C)$ as much as possible. If the rays are away from $E(C) = 0$, then improvements to internal density have a larger impact on conductance.

We now analyze the performance of the Greedy Algorithm 3 with conductance and four networks, results are displayed in Figure 2.6. In the College Football League, the greedy algorithm finds communities in the (I, E) plane where improvements in I and E are balanced. The final community found corresponds to our notion of a good community. For Zarchary’s Karate Club, the greedy algorithm begins to enter the region where external density determines conductance and returns a community of debateable quality. This effect is more emphasized in the relativity and astrophysics co-author networks. The greedy algorithm initially returns communities in the region of the (I, E) plane with balanced weightings between internal and external density. When external density reaches the region of low external density, the level sets show that small improvements to external density at the cost of lower internal density dramatically improve conductance.

This is the cause of the problem found by Leskovec et. al [17]. As a metric, conductance either incorporates internal density, as in the small College Football communities, or does not incorporate internal density, as in the larger Relativity and Astrophysics Co-author communities.

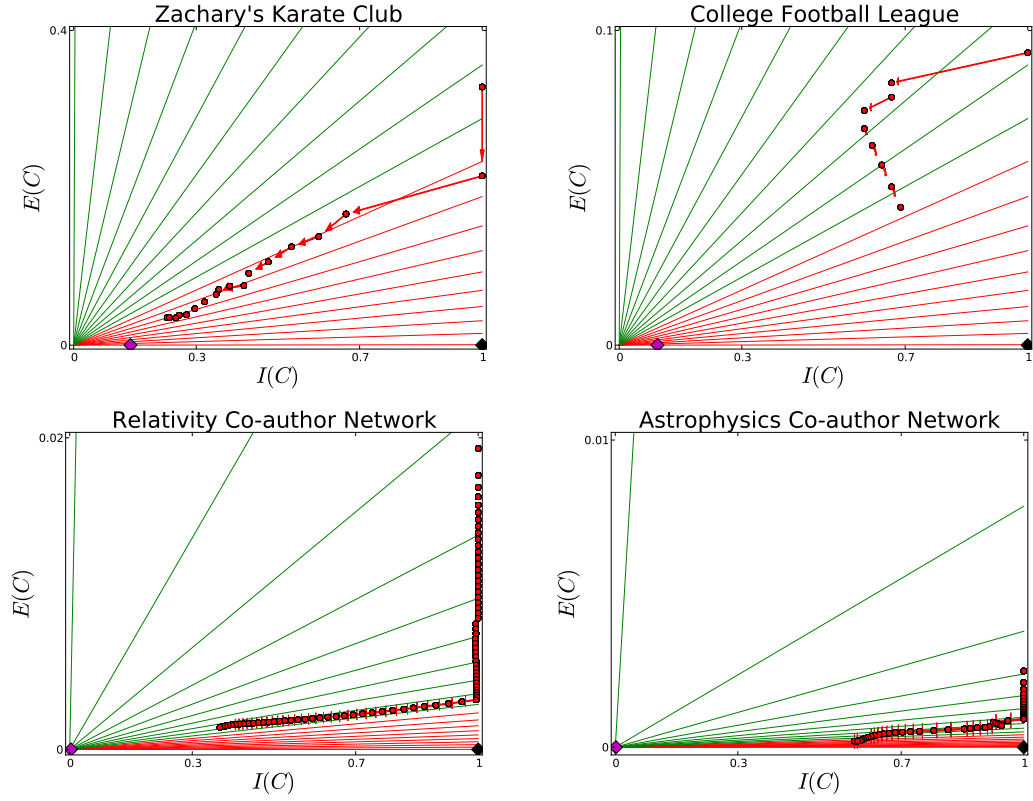


Figure 2.6: The progression of communities that optimize conductance. Note, both the entire graph and the ideal community optimize conductance. In the relativity and astrophysics networks, we stop following the progression of conductance once it becomes clear the entire graph will be engulfed. (In the case of the college football league, a local optimum was reached, but reports an undesirable value of conductance..)

This problem is amplified by the effect a community's size has on conductance. Now we fix the I, E ratio and observe how changes in $|C| = k|V|$ affect conductance, see Figure 2.7. Conductance always values a larger community more favorably. As long as the community is of small to moderate size and has a large $E(C)$ value, the greedy algorithm will return communities that to correspond to our intuition that an ideal community.

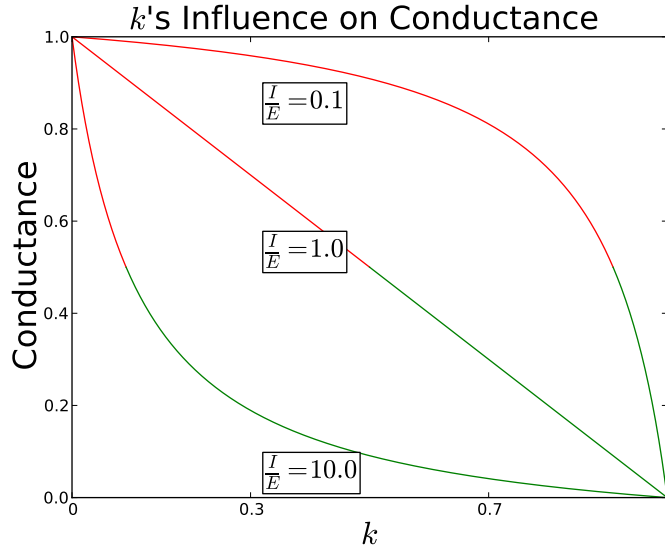


Figure 2.7: Influence of size of community on the value of conductance.
The object is to minimize conductance

2.3 Set of Communities Based Metrics

We now explore metrics that evaluate the strength of a set of communities, $S = \{C_1, C_2, \dots, C_n\}$. Several community detection methods are based on finding a partitioning of the network that optimizes such a metric. The most popular of these metrics is, modularity developed by Newman [20].

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, $S = \{C_1, C_2, \dots, C_n\}$. We begin as we did for single communities and consider the characteristics of a good set of communities. A set of good communities is a set of cliques such that every edge is within some community and every community is a maximal

clique. 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, $S = \{C_1, C_2, \dots, C_n\}$, the internal density of the set is the sum of the number of edges that do exist within each community compared to the maximal number of edges that could exist.*

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

Definition 8 (External Density of a Set of Communities) *In a set of communities, S , the EXT_EDGES is the set of edges not covered by any community. External density is the number of edges in EXT_EDGES compared to the number of edges in the graph.*

$$E(S) = \frac{\sum_{(u,v) \in \text{EXT_EDGES}} w(u, v)}{\sum_{u,v \in V} w(u, v)} \quad (2.10)$$

Definition 9 (Conciseness of a Set of Communities) *Conciseness is the size of S .*

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

Our choice of defining the parameters, allows the analysis of any set of communities, including overlapping communities. In particular, our definition of internal density for a set of communities, allows nodes to be placed in multiple communities. External density is independent of overlapping communities, as well as conciseness.

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

$$I(S) = 1$$

$$E(S) = 0$$

$$|S| = \text{number of connected components of the network.}$$

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 optimize those two parameters. Failure to evaluate the third parameter reveals an undesired characteristic 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 the most popular of these metrics. It compares the number of internal edges found, to the number of expected edges in a random graph. Modularity was developed by Newman in [20] and has found wide spread use due to the fast algorithms for maximizing modularity. In particular, the use of dendrograms in the Louvain Algorithm [2] runs in minutes for large networks.

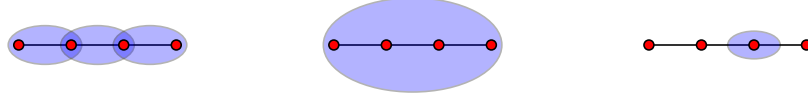


Figure 2.8: The communities that optimize 2 out of 3 parameters. Nodes are in red, lines are edges, and communities are blue ellipses. The left community configuration optimizes $I(S) = 1$ and $E(S) = 0$, but not conciseness at $|S| = 3$. The middle configuration optimizes $E(S) = 0$, $|S| = 1$, but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

There is not a closed form parameterization of modularity in terms of our definitions of $I(S)$, $E(S)$, and $|S|$. However, for each module's contribution there is a closed form parameterization in terms of internal and external density for a single community, $I(C)$ and $E(C)$. If we allow, $p = \frac{|C|(|C|-1)}{2L}$ and $q = \frac{|C|(|V|-|C|)}{2L}$, where L is the number of edges in the graph then:

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

We first note that if there exists a set of disjoint cliques in the graph, only a partitioning of each clique into a module maximizes modularity. Modularity already aligns more strongly with our understanding of strong communities than previous metrics.

We can not plot the level sets for modularity over a set of communities, but we can plot the level sets for the contribution to modularity from each commu-

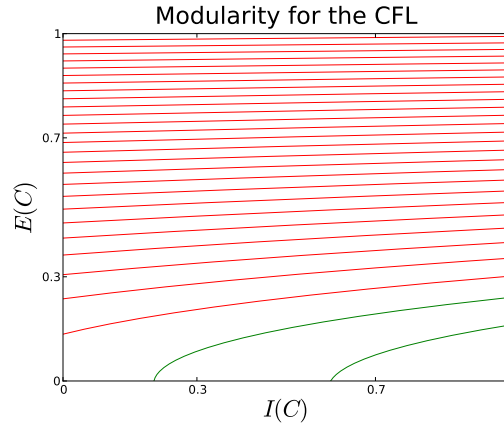


Figure 2.9: The level sets of how MODULARITY treats the $I(C)$, $E(C)$ space for one community of size 9 in the CFL. Note the sharp transition from a region that heavily favors improvements in external density to a region that heavily favors improvements in internal density ($E(C) < 0.1$).

nity. In Figure 2.9 we find that modularity is a two part optimization. When $E(C)$ is large, modularity maximization attempts to decrease $E(C)$ as quickly as possible. Once a threshold of $E(C)$ is crossed, modularity maximization attempts to increase $I(C)$ as quickly as possible. The transition between these two phases of optimization is sudden and revealed by a dramatic turn in the level set curves. The larger the graph the more sudden this transition.

JTODO They say modularity has a resolution limit, but that is just because modularity at first tries to optimize external density which is prone to joining communities together. [?]

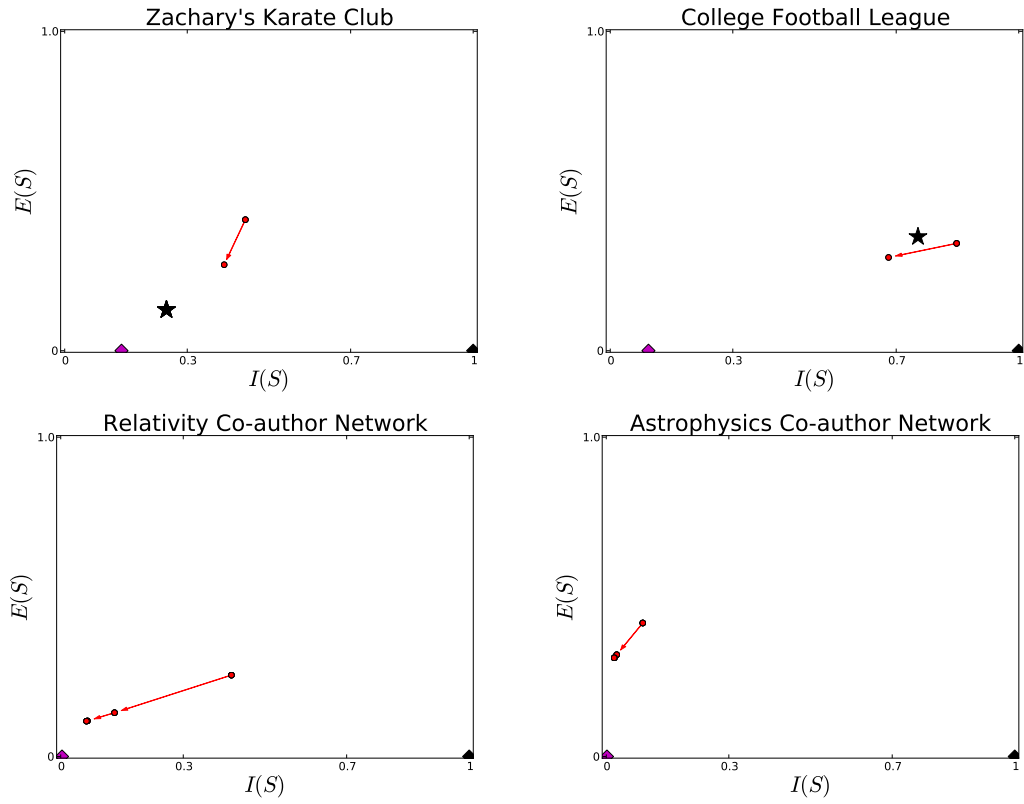


Figure 2.10: Here we run the Louvain Algorithm [2] to maximize modularity. The $(I(S), E(S))$ path is each level of the dendrogram. The $(I(G), E(G))$ value for the entire graph is the diamond in the lower left. In the general relativity and astrophysics co-author networks, modularity does not present much of an improvement over $I(G)$ and has a much higher $E(G)$ value.

CHAPTER 3

A NEW METRIC: LINEARITY

Metrics in community detection fall into two categories. Either they measure one of two characteristics of communities or they are inconsistent. In measuring only one characteristic, edges cut, cut, ratio, and expansion highly value poorly connected communities. In being unpredictable, conductance and modularity, sometimes produce strong communities and sometimes, especially as the size of the communities increases, return poorly connected communities[?].

In this chapter we will present metrics for single communities and sets of communities that measure both characteristics and are consistent.

3.1 Single Community Detection

For single communities there are two characteristics: internal density, $I(C)$, and external density, $E(C)$. We continue the use Definitions 4 and 5 from our analysis on single community metrics. We wish to create a metric with the following properties:

- Only a disconnected clique maximizes the metric. Ie, a unique maximum occurs at $(I(C), E(C)) = (1, 0)$.
- The metric is consistent. Ie, if $(I(C), E(C)) = (x, y)$ and the community can be improved by (δ_I, δ_E) and $M(x, y)$ is the metric valuation of C then:

$$M(x + \delta_I, y + \delta_E) - M(x, y) = M(\delta_I, \delta_E), \quad (3.1)$$

does not depend on (x, y) .

A linear metric satisfies these constraints.

Definition 11 (Linearity) $\text{LINEARITY}(C)$ is a metric for single communities with a linear weighting of internal density, $I(C)$, and external density, $E(C)$.

$$\text{LINEARITY}(C) = aI(C) - bE(C) \quad (3.2)$$

The constants a and b are restricted to $a, b \in (0, 1]$.

In some applications, we may want to relax the consistency constraint. An application could be interested in specific values for the size of the community, internal and external density. If this is the case, we suggest a polynomial approach to building a metric.

Definition 12 (General Metric) The general metric for evaluating any single community is a sum of polynomial functions on internal and external density.

$$\text{GENERAL}(C) = \sum_{i=0} f_i(C)I(C)^i - g_i(C)E(C)^i \quad (3.3)$$

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

We recommend that once your metric is built, you check the plots of level sets and size of community impact. In particular, check that:

- The (I, E) values that have local and global maximums correspond to communities you want to find.
- That the metric aids finding those communities with level sets; the gradient of the level sets leads to the desired communities. Again, this is only a visual check. Graphs and networks are discrete. The progression of an algorithm that maximizes the metric will not be able to follow the gradient in all cases, but it will stay as close as possible to the gradient.

Given, our LINEARITY(C) metric, we now do just that.

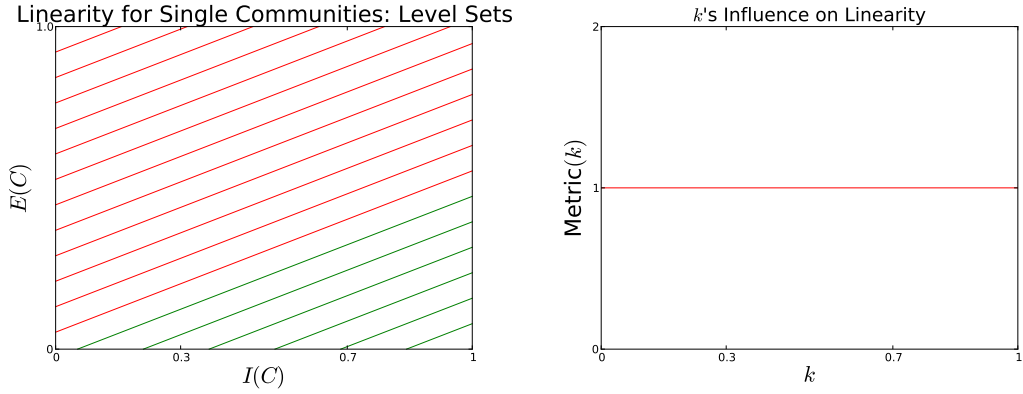


Figure 3.1: The two plots are boring and predictable.

We now put our metric through the same analysis for the other single community metrics. We chose our parameters a and b by setting $a = 1$ and searching for b that did not cause the greedy algorithm to return the entire graph but still had the greedy algorithm expand the initial community. Cover a method to automatically find a and b in the next section. JTODO

JTODO The benefits are clarity and uniformity. Independent of community size.

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

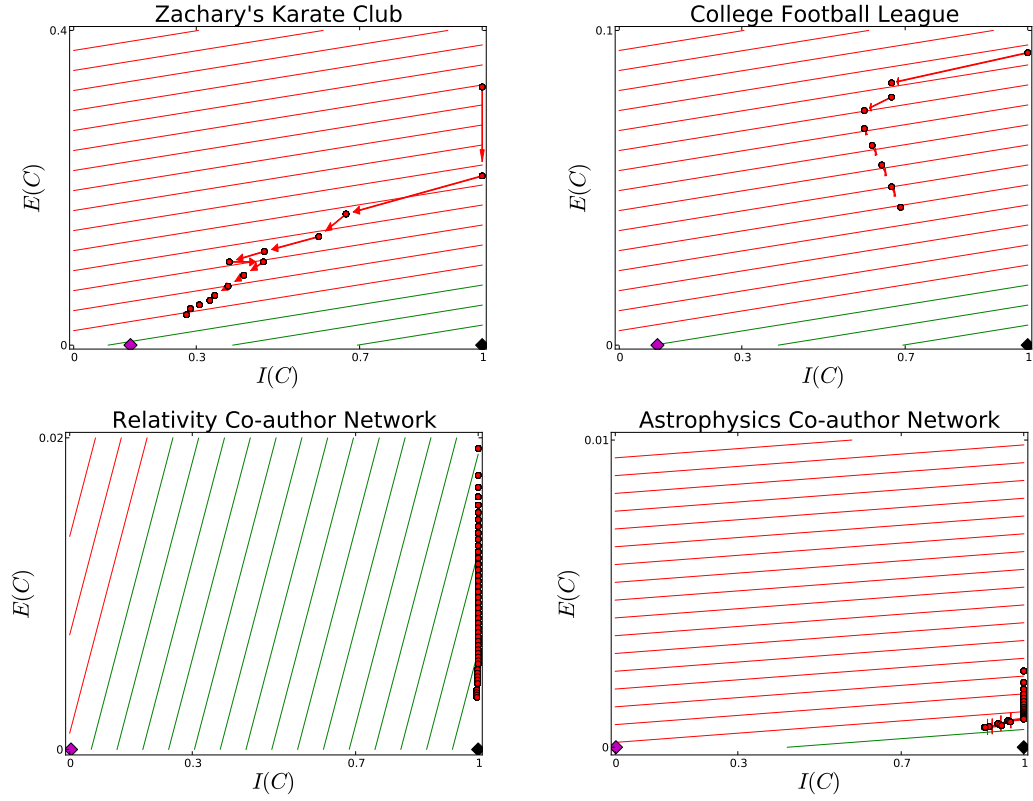


Figure 3.2: Our linearity metric is tested exactly as previous single community metrics have been with the Greedy Algorithm 3. Notice that the paths are truncated paths of Conductance in Figure 2.6. JTODD add in all the final spots from previous metrics, also change b so that the final communities are more valued than the entire graph.

improvements in each of the parameters is set by the user.

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

$$\text{LINEARITY}(S) = aI(S) - bE(S) - c|S|, \quad (3.4)$$

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:*

$$\text{GENERAL}(S) = \sum_{i=0} f_i(S)I(S)^i - g_i(S)E(S)^i - h_i|S|^i \quad (3.5)$$

From our experience of working with bottom up approach algorithms, we find that you must always grab improvements to internal density when possible. If an external density improvement is taken rather than an internal it is extremely unlikely that you will ever be able to return to that internal density value. The reverse is not true about external density. With the expansion of a partition into slightly overlapping communities external density can be greatly decreased.

Our algorithm for maximizing linearity is then a three part.

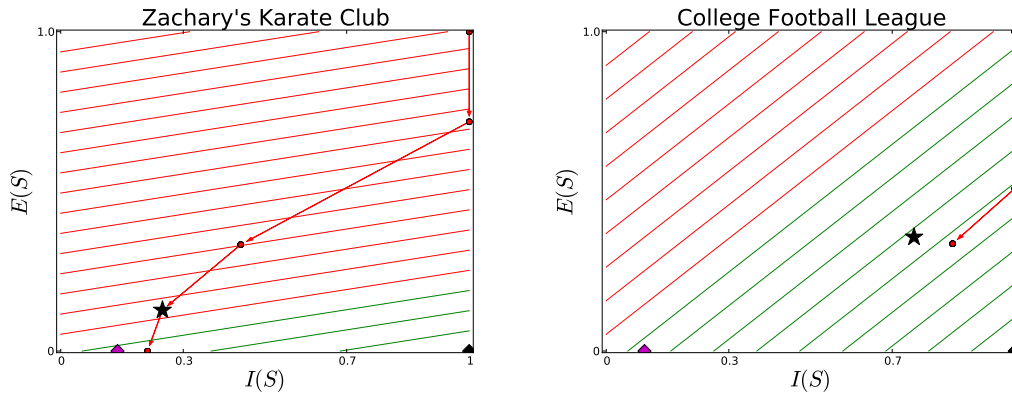


Figure 3.3: Compared to Known Solutions: For the karate club we find the two communities, but then expansion. For the college football league we find 13 communities, rather than the real 12, but this results in a better value in internal and external density. In the karate and football the black star corresponds to the value of known community solutions.

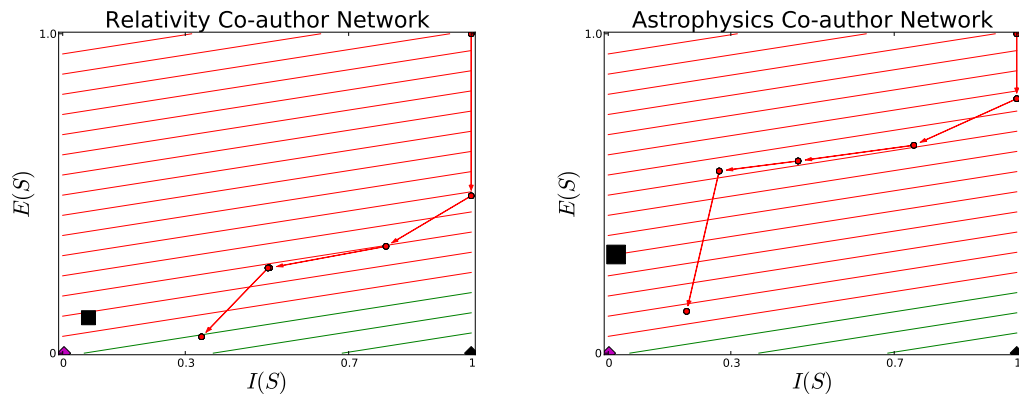


Figure 3.4: Linearity is in red. The (I, E) values modularity produces are inserted with a black square In relativity there are no known communities. For now we just show that good avlues of internal and external density can be found.

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 Collaboration Networks

5.1.1 Karate Club

[22]

5.1.2 Astrophysics

5.1.3 Condensed Matter

5.1.4 High Energy Physics

5.1.5 General Relativity

5.2 Enron Email Network

5.3 Epinions Social Network

5.4 Gnutella P2P Network

5.5 Physics Citation Network

5.6 Web Graphs

5.6.1 Berkeley Webpage

5.6.2 Google

5.7 Wiki Network

5.7.1 Communication Network

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] [23] [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.*, JTOD0, 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*, JTOD0, 2005.
- [6] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72(027104 JTOD0), 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 biological networks. *PNAS*, 99(12):7821–7826, June 2002.
- [9] M. B. Hastings. Community detection as an inference problem. *Archive JTOD0*, 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 JTOD0), 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] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.
- [23] 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.