A Good Neighborhood Hierarchical Model for Communities in Social Networks

Maria Florina Balcan
School of Computer Science
Georgia Institute of Technology
Atlanta, GA 30332
ninamf@cc.gatech.edu

Yingyu Liang
School of Computer Science
Georgia Institute of Technology
Atlanta, GA 30332
yliang39@gatech.edu

Abstract

Community detection in social networks has in recent years emerged as an invaluable tool for describing and quantifying the interactions in the networks. In this paper we propose a theoretical model that captures both the neighboring structure and the hierarchical nature of the communities. We further present an efficient algorithm and prove that it successfully detects all communities satisfying the property defined in our model.

1 Introduction

The structure of social networks has been extensively studied over the past several years in many disciplines, ranging from mathematics and computer science to sociology and biology. A significant amount of recent work in this area has focused on the development of community detection algorithms, since community structure reflects how entities in a network form meaningful groups such that interactions within the groups are more active compared to those between the groups and the outside world. The discovery of these communities allows for understanding the structure of the underlying network, or decision making in the network [6, 7, 15, 16].

While many heuristics and optimization criteria have been proposed [8, 11, 12, 13, 17], there is no established consensus on the definition of communities, partially due to the fact that there are a wide variety of networks of fundamentally different natures, and the applications may have different requirements as to what properties the communities to be detected should possess. Theoretical models aiming to capture common intuitions about communities across different types of networks are thus valuable in providing insights into the community structures of social networks and guiding the design of new algorithms.

Generally, a community should be thought of as a subset that has more interactions between its members than between its members and the remainder of the network. This is captured by some recently proposed models, such as [1, 2, 8]. However, recent studies suggest that networks often exhibit hierarchical organization, in which communities can contain groups of sub-communities, and so forth over multiple scales. For example, this can be observed in ecological niches in food webs, modules in biochemical networks or groups of common interest in social websites [18, 10, 5]. It is also shown empirically and theoretically that hierarchical structures can simultaneously explain and quantitatively reproduce many commonly observed topological properties of networks [4, 19, 9]. Therefore, hierarchical nature should also be reflected in modeling communities.

In this paper we take a step towards building community model that reflects both the tight connection and the hierarchical nature of communities. Inspired by [3], we introduce and study a notion of communities satisfying good neighborhood property. We further present an efficient algorithm and prove that it successfully detects all communities satisfying good neighborhood property. Both our model and algorithm are based on the neighborhood structure of the network that encompasses a variety of settings. In the remainder of the paper, we formalize our model in Section 2, and then present and analyze our algorithm in Section 3. Finally in Section 4 we provide a discussion of our results and future work.

2 Good Neighborhood Hierarchical Model

A social network is typically represented as a graph $G = (S, E)$ on a set of $n = |S|$ points[1], where the edges could be undirected or directed, unweighted or weighted. In this paper, however, we assume our access to the network data is via a neighborhood function N which given an point x and a threshold t outputs a list $N_t(x)$ containing the t nearest neighbors of x in S. This neighborhood function can be obtained directly by sorting for each point x all the other points y according to the weights of the edges (x, y) (we assume WLOG that the weights are in $[0, 1]$ and the weight of an edge not in E is regarded as 0). As pointed out in [2], we also have an alternative approach to convert the observed graph into the neighborhood function. More specifically, we regard the observed graph as a realization of some underlying unobserved set of relations, and thus we can lift the graph to an affinity system based on various beliefs about how the latent relations generate the observed graph, and then sort the points according to the affinity system to get the neighborhood function. For example, based on the belief that the graph serves as an approximation spanner of the underlying affinity system [14], we can define the affinity between points x and y as $1/d(x, y)$, where $d(x, y)$ is the shortest path distance between x and y. Note that the results of appropriate lifting procedures can better reflect the true relationships between entities, and thus the conversion can address the challenging issue of sparsity in the observed graph. More detailed discussion can be found in [2].

In the remainder of the section, we define our model based on the neighborhood function. We begin with the following notion of compact blobs, which will serve as a building block for our model.

Definition 1. *A subset A of points is called a compact blob, if out of the $|A|$ nearest neighbors, any point $p \in A$ has at most αn neighbors outside A, i.e. $|N_{|A|}(p) \setminus A| \leq \alpha n$; any point $q \notin A$ has at most αn neighbors inside A, i.e. $|N_{|A|}(q) \cap A| \leq \alpha n$.*

Note that the notion of compact blobs is inspired by the α-good neighborhood property defined in [3]. The notion captures the desired property of communities to be detected: members in the community have much more interactions with other members inside the community and have much less interaction with those outside. However, in practice, the notion may seem somewhat restricted. First, it requires all the members in the community have most interactions with other members inside the community, which may not be the case in real life. For example, some members in the boundary may have more interactions with the outside world, i.e. they have more than αn neighbors from outside. Also, we want to reflect the hierarchical nature of communities observed in the networks. Based on this consideration, we define the weak α-good neighborhood property formally as follows.

Definition 2. *A community C satisfies the weak α-good neighborhood property if*

- *any point $p \in C$ falls into a compact blob $A_p \subseteq C$ of size at least $6\alpha n$,*
- *at least $5/6$ fraction of points in A_p have all but at most αn nearest neighbors from C out of their $|C|$ nearest neighbors,*
- *any point q outside C have at most αn nearest neighbors from C out of their $|C|$ nearest neighbors.*

Informally, the first condition requires that every point falls into a sufficiently large compact subset of its community. This condition formalizes the local neighborhood structure that each member interacts actively with a subset in the community. The second condition requires that for a majority of points in such subsets, most of their nearest neighbors are from the community. This condition formalizes more global neighborhood structure about how the locally compact subsets relate to each other. The third condition then formalizes how the community is separated from the outside. Note that we no longer require all the members in the community have most interactions inside; we only require each member interacts with sufficiently many members and a majority of members in these local groups interact actively. Also note that the notion is hierarchical in nature: the locally compact

[1] To distinguish from the nodes in the hierarchy our algorithm builds, we refer to the entities in S as points.

subsets can be viewed as communities in lower levels. More generally, we can replace the first condition by requiring that p falls into either a compact blob or a community satisfying the weak α-good neighborhood property, leading to a recursive definition. Our results still hold for the more complicated definition, but we use the current simpler one for ease of presentation.

Next we further take into consideration possible noise that arises in practice. Naturally, we can relax the weak α-good neighborhood to the weak (α, ν)-good neighborhood defined as follows. Informally, it requires that the target community satisfies the weak α-good neighborhood property after removing a few bad points B. For convenience, we call the other points in $S \setminus B$ good points.

Definition 3. *A community C satisfies the weak (α, ν)-good neighborhood property if there exist a subset of bad points B of size at most νn, such that*

- *any good point $p \in G = C \setminus B$ falls into a compact blob $A_p \subseteq C$ of size at least $6(\alpha+\nu)n$,*
- *at least $5/6$ fraction of points in A_p have all but at most αn nearest neighbors from G out of their $|G|$ nearest neighbors in $S \setminus B$,*
- *any good point q outside $C \cup B$ have at most αn nearest neighbors from G out of their $|G|$ nearest neighbors in $S \setminus B$.*

3 Hierarchical Community Detection Algorithm

In the section, we propose an algorithm for detecting communities satisfying the weak (α, ν)-good neighborhood property. The goal of our algorithm is to output a set of communities such that each community satisfying the weak (α, ν)-good neighborhood property is close to one in the output. To be precise, we say that a community C is ν-close to another community C' if $|C \setminus C'| + |C' \setminus C| \leq \nu n$. We first describe the details in Algorithm 1, and then present the analysis in Theorem 1.

Algorithm 1 Hierarchical Community Detection Algorithm

Input: neighborhood function N on a set of points S, $n = |S|$, $\alpha > 0, \nu > 0$.

1. Initialize $\mathcal{C}'$ to be a set of singleton points.
2. Initialize the threshold $t = 6(\alpha + \nu)n + 1$.
3. Build F_t on S by connecting any $x, y \in S$ if x, y share more than $t - 2(\alpha+\nu)n$ neighbors out of their t nearest neighbors.
4. Build H_t on $\mathcal{C}'$ as follows. For any two current clusters u, v:
   If they are both singleton subsets, connect them when they share more than νn singleton subsets as neighbors in common in F_t.
   Otherwise, for any $x \in u, y \in v$, let $S_t(x, y)$ denote the number of points in $u \cup v$ they share as neighbors in common in F_t. Connect u, v when $\text{median}_{x \in u, y \in v} S_t(x, y)$ is more than $(|u| + |v|)/4$.
5. For any connected component in H_t:
   If the union of the subsets in the component contains more than $4(\alpha + \nu)n$ points, merge these subsets.
6. If $t < n$, let $t = t + 1$, then go to Step 3.

Output: Hierarchy T with single elements as leaves and internal nodes corresponding to the merges performed.

Now we prove that the algorithm successfully outputs a hierarchy such that any community satisfying the weak (α, ν)-good neighborhood property is close to one of the nodes in the hierarchy. Formally,

Theorem 1. *Algorithm 1 outputs a hierarchy such that any community satisfying the weak (α, ν)-good neighborhood property is ν-close to a node in the hierarchy. The algorithm runs in time $O(n^{\omega+1})$, where $O(n^{\omega})$ is the state of the art for matrix multiplication.*

The theorem follows from Lemma 3 and Lemma 4. First, we have the following lemma: for a good point p, all the points in the compact blob A_p it falls in cannot be merged with good points outside A_p before they are all merged together.

Lemma 1. *For any good point p, when $t \leq |A_p|$, good points from A_p will not be merged with good points outside A_p. At the end of the threshold $t = |A_p|$, all points in A_p have been merged into a subset.*

Proof. We prove this by induction on t. The claim is clearly true initially. Now assume for induction that at the beginning of a threshold $t \leq |A_p|$, in $\mathcal{C}'$ good points from A_p are not merged with good points outside A_p, i.e. any subset can contain good points from only one of A_p and $S \setminus B \setminus A_p$. We now analyze the properties of the graphs F_t and H_t, and show that at the end of the current threshold, the claim is still true.

First, as long as $t \leq |A_p|$, the graph F_t has the following properties.

- No good point x in A_p is connected to a good point y outside A_p. By the definition of compact blob,out of the t nearest neighbors, x has at most $(\alpha+\nu)n$ neighbors outside A_p. For $y \in S \setminus B \setminus A_p$, y has at most $(\alpha+\nu)n$ neighbors in A_p. Then x, y have at most $t - 4(\alpha+\nu)n$ common neighbors, so they are not connected.
- No bad point z is connected to both a good point x in A_p and a good point y outside A_p. We know that out of the t nearest neighbors, x has at most $(\alpha+\nu)n$ neighbors outside A_p. So if z is connected to x, then z must have more than $t-3(\alpha+\nu)n$ neighbors in A_p. Since y has at most $(\alpha+\nu)n$ neighbors in A_p, we have that y, z share less than $t - 2(\alpha+\nu)n$ neighbors, so they are not connected.

Based on the properties of F_t and the inductive assumption that any subset can contain good points from only one of A_p and $S \setminus B \setminus A_p$, we show that the graph H_t has the following properties.

- No subset u containing good points from A_p is connected to a subset v containing good points outside A_p. Note that the fraction of bad points in u or v is at most $1/4$. Then the number of pairs (x, y) with good points $x \in u$ and $y \in v$ is at least $\frac{3}{4}|u| \times \frac{3}{4}|v| > |u||v|/2$, i.e. more than half of the pairs (x, y) with $x \in u$ and $y \in v$ are pairs of good points. This means there exist good points $x^* \in u, y^* \in v$ such that $S_t(x^*, y^*)$ is no less than $\mathrm{median}_{x\in u, y\in v} S_t(x, y)$. By the properties of F_t, x^*, y^* have no common neighbors. Therefore, u and v are not connected.
- If a subset w contains only bad points, then it cannot be connected to both a subset containing good points from A_p and a subset containing good points outside A_p. Suppose it is connected to u which contains good points from A_p. Note that since w contains only bad points, it must contain only a single point z. Also note that the fraction of bad points in u is at most $1/4$. So there exists a good point $x^* \in u$ such that $S_t(x^*, z) \geq \mathrm{median}_{x\in u} S_t(x, z)$. Then we have $S_t(x^*, z) > (|u| + |v|)/4 > \nu n$. Since in F_t, x^* is only connected to good points from A_p and bad points, z and x^* must share some common neighbors from A_p, i.e. z is connected to some good points in A_p. Similarly, if w is connected to a subset containing good points outside A_p, then the point in w must connect to some good point outside A_p. But this is contradictory to the fact that in F_t no bad point is connected to both a good point in A_p and a good point outside A_p.

By the properties of H_t, no connected component contains both good points in A_p and good points outside A_p. So at the end of this threshold t, the claim is still true. Then by induction, we know that when $t \leq |A_p|$, we will not merge good points from A_p with good points outside A_p.

Next we show that at the end of the threshold $t = |A_p|$, we will merge all points in A_p into a subset. First, at this threshold, all good points in A_p are connected in F_t. Any good point in A_p has at most $(\alpha+\nu)n$ neighbors outside A_p, so when $t = |A_p|$, any two good points x, y in A_p are connected, and thus they share at least $|A_p|$ common neighbors in F_t. Second, all subsets containing good points in A_p are connected in H_t. If no good points in A_p have been merged, then these singleton points will be connected in H_t since they share at least $|A_p|$ singleton subsets as common neighbors in F_t. If some good points in A_p have already been merged into non-singleton subsets, we can show that in H_t these non-singleton subsets will be connected to each other and connected to singleton subsets containing good points from A_p. For any such pair of subsets u, v, we know that the fraction of bad points in u or v is at most $1/4$, so there exist good points $x^* \in u, y^* \in v$ such that $\mathrm{median}_{x\in u, y\in v} S_t(x,y)$ is no less than $S_t(x^*, y^*)$. Since x^*, y^* are connected to all good points in A_p in F_t, $S_t(x^*, y^*)$ is no less than the number of good points in u and v. So $\mathrm{median}_{x\in u, y\in v} S_t(x,y) \geq S_t(x^*, y^*) > (|u|+|v|)/4$, and thus u, v are connected in H_t. Therefore, all points in A_p are merged into a subset. □

The following is a useful consequence of Lemma 1.

Lemma 2. *In Algorithm 1, if a subset u satisfies that for any good point $p \in u$, $A_p \subseteq u$, then there exist a subset of good points $P \subseteq u$, such that $\{A_p : p \in P\}$ is a partition of $u \setminus B$.*

Proof. We have $u \setminus B = \cup_{p\in u\setminus B} A_p$. We only need to show that sets in $\{A_p : p \in u \setminus B\}$ are laminar, i.e. for any $p, q \in u \setminus B$, either $A_p \cap A_q = \emptyset$ or $A_p \subseteq A_q$ or $A_q \subseteq A_p$. Assume for contradiction that there exist A_p and A_q such that $A_p \setminus A_q \neq \emptyset, A_q \setminus A_p \neq \emptyset$ and $A_p \cap A_q \neq \emptyset$. Without loss of generality, suppose $|A_p| \leq |A_q|$. Then by Lemma 1, at the end of the threshold $t = |A_p|$, we have merged all good points in A_p into a subset. Specifically, this means that we have merged $A_p \cap A_q$ with $A_p \setminus A_q$. So for $t \leq |A_q|$, we have merged good points in A_q with good points outside A_q, which is contradictory to Lemma 1. □

By the above lemmas, for any good point p, the subset A_p will be formed before points in it are merged with good points outside. Once these subsets are formed, we can show that subsets in the same target community will be merged together before they are merged with those from other communities, and thus the hierarchy produced has a node close to the target community. Formally, we have the following result.

Lemma 3. *For any community C satisfying the weak (α, ν)-good neighborhood property, $\mathcal{C}' \setminus B$ in Algorithm 1 is always laminar to $C \setminus B$, i.e. for any $C' \in \mathcal{C}'$, either $(C' \setminus B) \cap (C \setminus B) = \emptyset$ or $(C' \setminus B) \subseteq (C \setminus B)$ or $(C \setminus B) \subseteq (C' \setminus B)$. Furthermore, there is a node u in the hierarchy produced such that $u \setminus B = C \setminus B$.*

Proof. we will show by induction on t that: for any community C satisfying the weak (α, ν)-good neighborhood property,

- at the end of threshold t, $\mathcal{C}' \setminus B$ is laminar to $C \setminus B$,
- at the end of threshold t, for any C such that $|C \setminus B| \leq t$, we have merged all points in $C \setminus B$ into a subset.

These claims are clearly true initially. Assume for induction that they are true for the threshold $t-1$, we now show that they are also true for the threshold t.

We first show that the laminarity is preserved. The laminarity is broken only when we connect in H_t two subsets u, v such that u is a strict subset of C after removing the bad points, and v is a subset containing good points from outside. If there is a good point $p \in u$ such that $A_p \not\subseteq u$, then by Lemma 1, they cannot be connected. So we only need to consider the other case when for any good point $p \in u$, $A_p \subseteq u$. For convenience, we call a point great if it is a good point in C, and it has less than αn neighbors outside $C \setminus B$ out of the $|C \setminus B|$ nearest neighbors in $S \setminus B$. We now show that u, v are not connected in H_t. Since $u \setminus B$ is a strict subset of $C \setminus B$, by induction on the second claim, we have $t \leq |C \setminus B|$. Then great points in u and points in v share at most $2(\alpha+\nu)n < t - 2(\alpha+\nu)n$ common neighbors, so they are not connected in F_t. By Lemma 2 and the second condition of the weak (α, ν)-good neighborhood property, we know that at least $5/6$ fraction of points in $u \setminus B$ are great points. Then there exist a great point $x^* \in u$ and a point $y^* \in v$ such that $S_t(x^*, y^*)$ is no less than $\mathrm{median}_{x\in u, y\in v} S_t(x,y)$. Since in F_t great points in u are not connected to points in v, we have $S_t(x^*, y^*) \leq (|u| + |v|)/4$. So $\mathrm{median}_{x\in u, y\in v} S_t(x,y) \leq (|u| + |v|)/4$ and u, v are not connected in H_t. Therefore, the laminarity is preserved.

Next we show that at the end of the threshold $t = |C \setminus B|$, all points in $C \setminus B$ are merged into a subset. By Lemma 1, all good points in $C \setminus B$ are now in sufficiently large subsets. We claim that any two of these subsets u, v are connected in H_t, and thus will be merged. Again by Lemma 2, we know at least $5/6$ fraction of points in $u \setminus B$ or $v \setminus B$ are great points, and thus there exist great points $x^* \in u, y^* \in v$ such that $S_t(x^*, y^*)$ is no more than $\mathrm{median}_{x \in u, y \in v} S_t(x, y)$. Notice that all great points in u are connected to great points in v in F_t, since they share at least $t - 2(\alpha + \nu)n$ neighbors. Then $S_t(x^*, y^*) \geq 3(|u| + |v|)/4 > (|u| + |v|)/4$, and thus $\mathrm{median}_{x \in u, y \in v} S_t(x, y) > (|u| + |v|)/4$. Therefore, any two subsets containing good points from $C \setminus B$ are connected in H_t and thus are merged.

So the two claims hold for all t, specially for $t = n$. Then the algorithm must stop after this threshold, and we have the lemma as desired. □

Lemma 4. *Algorithm 1 has a running time of* $O(n^{\omega+1})$.

Proof. To implement the algorithm, we introduce some data structures. For any $x \in S$, if y is within the t nearest neighbors of x, let $I_t(x, y) = 1$, otherwise $I_t(x, y) = 0$. Initializing I_t takes $O(n^2)$ time. Next we compute $CN_t(x, y)$, the number of common neighbors between x and y. Notice that $CN_t(x, y) = \sum_{z \in S} I_t(x, z) I_t(y, z)$, so $CN_t = I_t I_t^T$. Then we can compute the adjacent matrix F_t (overloading notation for the graph F_t) from CN_t. These takes $O(n^\omega)$ time.

To compute the graph H_t, we introduce the following data structures. Let $FS_t(x, y) = 1$ if x, y are singleton subsets and $F_t(x, y) = 1$, and let $FS_t(x, y) = 0$ otherwise. Let $NS_t = FS_t(FS_t)^T$, then for two singleton subsets x, y, $NS_t(x, y)$ is the number of singleton subsets they share as neighbors in common in F_t. Let $FC_t(x, y) = 1$ if x and y are in the same subset and $F_t(x, y) = 1$, and let $FC_t(x, y) = 0$ otherwise. Let $S_t(x, y) = NS_t(FC_t)^T + FC_t(NS_t)^T$, then for two points $x \in u, y \in v$ where u, v are two non-singleton subsets, $S_t(x, y)$ is the number of points in $u \cup v$ they share as neighbors in common in F_t. Based on NS_t and S_t we can build the graph H_t. All these take $O(n^\omega)$ time.

When we perform merge or increase the threshold, we need to update the data structures, which takes $O(n^\omega)$ time. Since there are $O(n)$ merges and $O(n)$ thresholds, Algorithm 1 takes time $O(n^{\omega+1})$ in total. □

4 Discussion

Note that our algorithm outputs a hierarchy such that any community satisfying the good neighborhood property is close to some node in it. However, not all nodes in the hierarchy correspond to meaningful communities. For example, a leaf containing a single point is generally not considered a meaningful community. We can perform a further processing step that checks all the $O(n)$ nodes and filter out undesired ones. The check can be specific to the applications, so we do not include such a step in our algorithm. Also note that the input parameters α and ν in the algorithm denote the compactness of the communities. So we can tune these parameters to obtain communities with specific level of compactness, or vary them to obtain communities with all levels of compactness. This is useful since applications may have different requirements for the desired communities.

For future work, we plan to perform empirical study of our model and algorithm on real-world data sets. Another direction would be to speed up the algorithm and adapt it to large-scale scenarios.

References

[1] Sanjeev Arora, Rong Ge, Sushant Sachdeva, and Grant Schoenebeck. Finding overlapping communities in social networks: toward a rigorous approach. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, EC '12, pages 37–54, New York, NY, USA, 2012. ACM.

[2] M.F. Balcan, C. Borgs, M. Braverman, J. Chayes, and S.H. Teng. Finding endogenously formed communities. In *SODA 2013*, 2013.

[3] M.F. Balcan and P. Gupta. Robust hierarchical clustering. In *Proceedings of the Conference on Learning Theory (COLT)*, 2010.

[4] Aaron Clauset, Cristopher Moore, and M. E. J. Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98–101, May 2008.

[5] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):066111+, December 2004.

[6] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.

[7] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, June 2002.

[8] Jing He, John Hopcroft, Hongyu Liang, Supasorn Suwajanakorn, and Liaoruo Wang. Detecting the structure of social networks using (α, β)-communities. In *Proceedings of the 8th international conference on Algorithms and models for the web graph*, WAW'11, pages 26–37, Berlin, Heidelberg, 2011. Springer-Verlag.

[9] Jon Kleinberg. Complex networks and decentralized search algorithms. In *In Proceedings of the International Congress of Mathematicians (ICM*, 2006.

[10] M. Cosentino Lagomarsino, P. Jona, B. Bassetti, and H. Isambert. Hierarchy and feedback in the evolution of the Escherichia coli transcription network. *Proceedings of the National Academy of Sciences*, 104(13):5516–5520, March 2007.

[11] Jure Leskovec, Kevin J. Lang, and Michael Mahoney. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 631–640, New York, NY, USA, 2010. ACM.

[12] Nina Mishra, Robert Schreiber, Isabelle Stanton, and Robert Endre Tarjan. Clustering social networks. In *WAW*, pages 56–67, 2007.

[13] Nina Mishra, Robert Schreiber, Isabelle Stanton, and Robert Endre Tarjan. Finding strongly knit clusters in social networks. *Internet Mathematics*, 5(1):155–174, 2008.

[14] G. Narasimhan and M. Smid. Geometric spanning networks. *Cambridge University Press*, 2007.

[15] M. E. J. Newman. Detecting community structure in networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 38(2):321–330, March 2004.

[16] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, June 2006.

[17] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658–2663, March 2004.

[18] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabási. Hierarchical Organization of Modularity in Metabolic Networks. *Science*, 297(5586):1551–1555, August 2002.

[19] Michael Schweinberger and Tom A. B. Snijders. Settings in social networks: A measurement model. *Sociological Methodology*, 33:307–341, 2003.