Relaxations in Practical Clustering and Blockmodeling

Stefan Wiesberg and Gerhard Reinelt
Institut für Informatik, Universität Heidelberg
Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
E-mail: stefan.wiesberg/gerhard.reinelt@informatik.uni-heidelberg.de

Keywords: network analysis, clustering, blockmodeling

Received: June 21, 2014

Network analysts try to explain the structure of complex networks by the partitioning of their nodes into groups. These groups are either required to be dense (clustering) or to contain vertices of equivalent positions (blockmodeling). However, there is a variety of definitions and quality measures to achieve the groupings. In surveys, only few mathematical connections between the various definitions are mentioned. In this paper, we show that most of the definitions used in practice can be seen as certain relaxations of four basic graph theoretical definitions. The theory holds for both clustering and blockmodeling. It can be used as the basis of a methodological analysis of different practical approaches.

Povzetek: Pri razdeljevanju omrežij na podskupine so pristopi opredeljeni kot eni od štirih teoretičnih skupin.

1 Introduction

The structure of large networks is usually not comprehensible to the human beholder. Especially, if the network has not been designed by a human architect, but rather evolved over time in a complex (natural) process. Examples for such networks are social (friendship, mailing, scientific collaboration, advice giving), economic (trading between countries or companies), chemical (protein-protein reactions), biological (food chain), or internet link networks. Nevertheless, researchers in these fields use the networks to gain insight into their structural makeup. To this end, a first step is most often the reduction of the network's complexity with the help of algorithms. A common approach is to reduce the high number of nodes in the network. The idea of *blockmodeling approaches* is to group the nodes such that the number of groups is much lower than the number of nodes. The grouping is done in a way that leads to patterns in the network's links. We distinguish two kinds of patterns: Patterns of link *density* (Section 1.1) and of link *existence* (Section 1.2). An example for patterns of link density is given in Figure 1: On the left-hand side, we see a random drawing of a graph $G = (V, E)$. In the center, we see a partition of V into four groups A, B, C, D, indicated by four different colors, such that a density pattern becomes apparent. Densely connected are the group pairs AB, BD, DD, CD, CA, sparsely connected are AA, BB, CC, AD, BC. Note that we use a merely intuitive definition of density here for motivational reasons; strict mathematical definitions will be introduced subsequently.

Before we explain the patterns of link density, we formalize a vertex grouping of a graph G with vertex set V and edge set E as a vertex coloring. This is possible since every vertex coloring $\phi : V \to [c]$, where $[c] = \{1, 2, \ldots, c\}$, naturally defines a partition of V into the color classes. W.l.o.g. we assume that ϕ is surjective, i.e., all c colors are used. In this paper, we assume that our network is given as an undirected graph $G = (V, E)$. More general cases, in which there are weights (on the arcs or on group pairs) or multiple arc types are not treated here.

1.1 Patterns of link density

The goal of the grouping which is discussed in this section is to group the vertices in a way such that for each pair of groups, there are either very *many* or very *few* links between the groups. In other words, we search for a *pattern of link density* in the network.

Once such a pattern has been found, the network's complexity has been reduced in the following sense: One can now shrink the groups to single vertices, and connect two such vertices by an edge if the corresponding groups were densely connected prior to the shrinking. The shrunk graph for the example in Figure 1 is depicted on the right-hand side of the figure.

Let us formalize the *pattern* notion. Given a coloring ϕ, the pattern specifies for each pair of color groups whether they are interpreted to be densely or sparsely connected. A pattern is usually notated as a binary square matrix I. Its dimension is the number of groups. An entry I_{AB} is 1 if groups A and B are interpreted to be densely connected, and 0 if they are interpreted to be sparsely connected. The matrix I representing the pattern is usually called *image matrix*. It is symmetric as the network graph is undirected. The graph whose adjacency matrix is the image matrix is called *image graph*. Figure 1 (right) shows the image graph to the density pattern described in the caption text of the

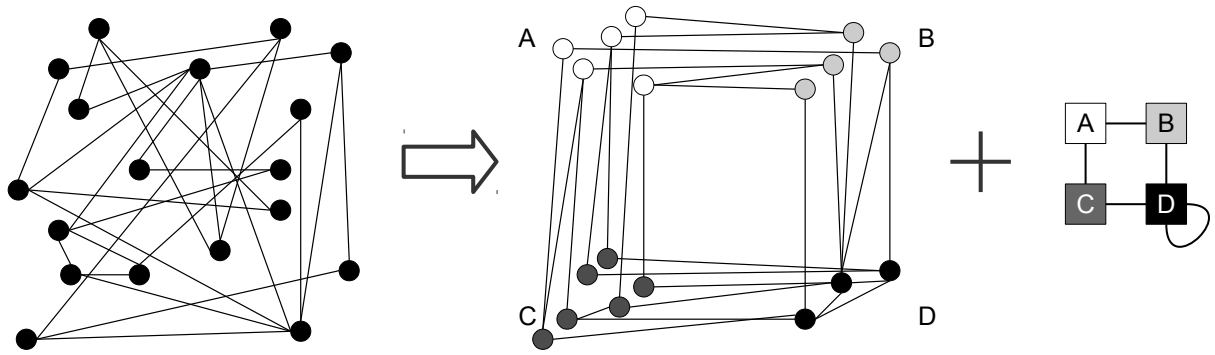

Figure 1: An exemplary density pattern.

figure. Note that the image graph can be seen as a simplification of the network structure: There is an edge in the image graph wherever there are many edges in the original network, and no edge wherever there are only few edges.

For a given network graph G, one is hence interested in a coloring ϕ of the vertices together with a density pattern. Such a pair (ϕ, I) of a coloring ϕ and its interpretation, an image matrix I of appropriate dimension, is called a *blockmodel*. The process of computing a good blockmodel for a given network is sometimes called *blockmodeling*.

1.2 Patterns of link existence

Density patterns imply that for the vertices in a group A, it holds that either all of them have very many or all of them have very few links to the vertices in a group B. In a pattern of link *existence*, however, one demands that either many vertices in group A have *at least one* link into group B or almost no vertex in group A has a link to group B. Analogously to the density pattern case, we can define an image matrix. It encodes for each pair of groups which of the two cases are interpreted to exist in the given coloring. The image graph then visualizes a pattern of connectivity. If an edge exists between groups A and B, then almost every vertex in A is connected to B, and vice versa. Otherwise, the groups are almost disconnected.

1.3 Fixing patterns

To find a suitable number of groups is generally part of the blockmodeling process. In practical blockmodeling, however, it is sometimes set a priori to a small fixed value. Moreover, the whole pattern is sometimes fixed a priori. The blockmodeling then reduces to the search for a coloring which matches the given pattern best. This is useful to test whether an assumed pattern actually exists in the network. A prominent example of pattern fixing is the *clustering problem*. Here, one searches for density patterns. The number c of groups is fixed to a small value and the image matrix is fixed to the $c \times c$ identity matrix. The blockmodeling hence consists of the search for a coloring with c colors such that the color groups themselves are dense, whereas their interconnections are sparse. In case that c is not fixed, the family of all identity matrices is considered as the set of feasible patterns.

1.4 Outline of the paper

Literature shows a large variety in practical blockmodeling approaches. Not only are they distinct in the way they use a priori fixings, they also differ in the ways they measure the quality of a given blockmodel for a given network. Usually, the search for clusters, link density and link existence patterns are treated separately. There are separate methods and publications for each of the three problem types.

In this paper, we present a new classification of the approaches. This classification holds for all (non-stochastic) clustering and blockmodeling approaches which quantify the quality of blockmodels and are reported in the following survey books: *Social Network Analysis* by Wasserman and Faust [16], *Network Analysis* by Brandes and Erlebach [6] (except *conductance*), and *Community Detection in Graphs* by Fortunato [10].

The search for an ideal blockmodel can usually be formulated as a graph coloring problem. By our classification, we show that the practical approaches can be seen as methods to optimize very specific relaxations of these problems. They are the same in clustering, link density and link existence patterns search.

Section 2 presents the graph coloring problems, which are relaxed in practical approaches. Section 3 explains the three types of relaxations that are used. Each type is illustrated with practical examples from the survey books. Finally, Section 4 summarizes and gives an outlook on applications of the classification.

2 Ideal blockmodels

In this section, we define *ideal* blockmodels of link density and existence. In an *ideal* blockmodel (ϕ, I) for link density, *all* links exist in the dense parts and *no* links exist in the sparse ones. In an ideal blockmodel for link existence, either *all* or *no* vertex in A have a neighbor in B.

Ideal *colorings* can be similarly defined. The reason is that in an ideal blockmodel (ϕ, I), the image matrix I can be directly constructed from ϕ: The entry I_{AB} is 0 if and only if there is no edge from A to B. We hence call a coloring ϕ *ideal* if the blockmodel (ϕ, I) is ideal, where I is constructed as explained.

There are three graph theoretical definitions of ideal colorings. They will be presented in the next three subsections.

2.1 The subgraph definition

In ideal blockmodels for density patterns, certain subgraphs are either complete or empty. These subgraphs can be defined as follows. Given a coloring ϕ, there is one such subgraph $G_{\phi,A,B}$ for every pair (A, B) of colors. It is obtained from G by deleting all vertices but the ones colored with A or B and deleting all edges but those connecting an A-colored with a B-colored vertex. $G_{\phi,A,B}$ is hence bipartite for $A \neq B$. Note that all of these subgraphs are edge disjoint. A similar observation can be made for ideal link existence blockmodels: That all vertices in A have at least one neighbor in B, and vice versa, is equivalent to the statement that $G_{\phi,A,B}$ contains no isolated vertices.

We have seen that *clustering* is a special case of link density, where the image matrix is a priori fixed. However, there is a common variant of clustering. It only requires the color groups to be dense, but does *not* require their interconnections to be sparse. In other words, only the diagonal image matrix entries are given. We include this variant into our classification scheme as it is widely used. It corresponds to Part (i) of the following definition of ideal colorings. Part (ii) defines ideal link density and Part (iii) ideal link existence colorings. See Figure 2 for examples.

Definition 1. *Given a graph G, a c-coloring $\phi : V \to [c]$ of its vertex set is*

(i) an ideal clique c-coloring*, if for all $A \in [c]$, the graph $G_{\phi,A,A}$ is complete.*

(ii) an ideal structural c-coloring*, if for all color pairs $A, B \in [c]$, the graph $G_{\phi,A,B}$ is either empty or a complete (complete bipartite for $A \neq B$) graph.*

(iii) an ideal regular c-coloring*, if for all color pairs $A, B \in [c]$, the graph $G_{\phi,A,B}$ is either empty or contains no isolated vertices.*

2.2 The node pair definition

We have seen that ideal colorings can be defined by subgraph characterizations. Alternatively, they can be defined by properties of same-colored vertices. In a clique c-coloring, every two vertices with the same color are connected by an edge. In a structural c-coloring, two vertices with the same color have exactly the same neighboring vertices in G. In a regular c-coloring, two vertices with the same color have exactly the same colors in their neighborhoods. Let $N(u)$ denote the set of vertices that are adjacent to vertex u. The following definition is hence equivalent to the subgraph definition above. See Lorrain and White [12] for details.

Definition 2. *Given a graph G, a c-coloring $\phi : V \to [c]$ of its vertex set is an*

(i) ideal clique c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $uv \in E$.*

(ii) ideal structural c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $N(u) \setminus \{v\} = N(v) \setminus \{u\}$.*

(iii) ideal regular c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $\{\phi(w) \mid w \in V, uw \in E\} = \{\phi(w) \mid w \in V, vw \in E\}$.*

2.3 The single node definition

A definition from the perspective of *single* vertex is only possible with respect to a fixed image matrix I. In this case, the following single node definition is equivalent to the two definitions above.

Definition 3. *Given a graph G and a $c \times c$ image matrix I, a c-coloring $\phi : V \to [c]$ of G's vertex set is*

(i) an ideal clique c-coloring*, if for all $u \in V$: u is adjacent to all $v \in V$ with $\phi(v) = \phi(u)$.*

(ii) an ideal structural c-coloring w. r. t. I*, if for all $u \in V$ and all $C \in [c]$: u is adjacent to all $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 1$, and to no $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 0$.*

(iii) an ideal regular c-coloring w. r. t. I*, if for all $u \in V$ and all $C \in [c]$: u is adjacent to at least one $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 1$, and to no $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 0$.*

3 Relaxations

For a given graph G, one can theoretically compute a coloring from Definition 1 or 2 to obtain an ideal coloring (and thus ideal blockmodel). However, this is usually not done in practice. In Section 3.1, we list some common reasons for this decision. In Section 3.2, we show that the approaches used in practice can be interpreted as the solution of an optimization problem on a relaxed problem definition.

3.1 Reasons for relaxations

There are several reasons for the use of relaxations instead of directly searching for ideal blockmodels. We list four of them.

1. *Non-existence of solutions.* An ideal coloring might only exist if a large number of colors is used.

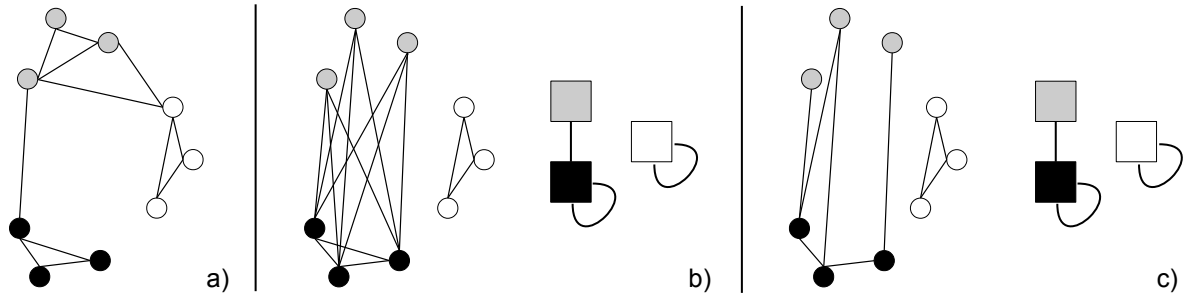

Figure 2: Ideal clique (a), structural (b), and regular (c) 3-colorings. In b) and c), the corresponding image graph is depicted.

2. *Real-world modeling reasons.* The definition might be too restrictive for the application at hand. For example, the graph theoretical definition of *clique* might be too strict to describe friendship cliques in social networks, where some edges can be missing.

3. *Involvement of statistics.* The relaxations allow to define statistically profound criteria for the quality of colorings, instead of the purely graph theoretical ones.

4. *Robustness against measuring errors.* The extraction of graphs from complex networks can be erroneous, especially in biological or chemical networks. However, a regular coloring on a graph can turn non-regular by the deletion or addition of one single edge. Relaxations are hence useful to limit the influence of these errors on the colorings.

3.2 General relaxation

In this section, we show how ideal blockmodels are relaxed in practice. Denote by $CC(c, G)$ the set of all clique c-colorings of the vertices of G. Analogously, we define $SC(c, G)$ and $RC(c, G)$ for structural and regular c-colorings. As a shorthand, we simply write $X(G)$ in a statement which holds for any fixed type (CC, SC, RC) and any fixed number c of colors. Practitioners, often implicitly, enlarge the set $X(G)$ of feasible colorings to a set $X_L(G) \supseteq X(G)$ and assign a penalty value $p(\phi) \geq 0$ to each member ϕ of the enlarged set $X_L(G)$. Afterwards, they solve the optimization problem of finding a coloring ϕ^* in $X_L(G)$ with the minimum penalty value $p(\phi^*)$. We now show that this is usually done in the following way: The set $X(G)$ of feasible colorings is enlarged by dropping some of the requirements in the definition of X. Furthermore, the penalty function p is not arbitrary, but measures the degree of violation against the *dropped* requirements. The optimization problem to be solved is thus:

(MIN-P) Given the set $X_L(G)$ and the penalty function $p : X_L(G) \to \mathbb{R}_0^+$, find a $\phi^* \in X_L(G)$ which minimizes p.

That is, among the colorings satisfying the non-dropped requirements, find the one which violates the dropped requirements to the least possible extent. As a convention, a penalty value of 0 is assigned to the colorings in $X(G)$, as they do not violate any dropped requirements (*compatibility requirement*, see Doreian et. al. [9]). Hence, a coloring satisfying the original definition X is always an optimum solution to (MIN-P).

We will now classify literature by the type of relaxation used. As we are considering the relaxation of ideal colorings, three types of relaxations come to mind: The relaxation of the coloring definition, of the node pair ideality definition and the subgraph ideality definition. Indeed, these possibilities are widely used. In Section 3.3, we will look at the cases where the general definition of coloring is relaxed. Sections 3.4 and 3.5 treat the ideality definition relaxations respectively.

3.3 Coloring relaxations

In Definition 1 and 2 for ideal colorings, the definition of "coloring" itself can be relaxed. If we use the binary variables x_{vA} to express whether vertex v is colored with A ($x_{vA} = 1$) or not ($x_{vA} = 0$), the requirement "to be a c-coloring" can be decomposed into the following sub requirements:

$$\sum_{A=1}^{c} x_{vA} = 1 \quad \text{for all } v \in V, \tag{1}$$

$$\sum_{v \in V} x_{vA} \geq 1 \quad \text{for all } A \in [c], \tag{2}$$

$$0 \leq x_{vA} \leq 1 \quad \text{for all } v \in V, A \in [c], \tag{3}$$

$$x_{vA} \in \mathbb{Z} \quad \text{for all } v \in V, A \in [c]. \tag{4}$$

Example (Fuzzy Colorings.) In some applications, it is meaningful for a vertex to get several colors at the same time. E.g., a person might be a member of several clubs. In this case, requirement (1) is dropped. Alternatively, a vertex might be allowed to consist of color fractions that sum up to 1, such as 50% red, 30% green and 20% blue. In this case, requirement (4) is dropped. One speaks of *fuzzy colorings* or *partitions* in both of these cases of relaxation. Usually, there is no penalty for a vertex to have more than one color at the same time. That is, the penalty function p is usually constant with respect to the coloring requirements.

Example (Number of Colors.) For many applications, a good choice for the number of colors is not a priori known

and hence not fixed to a certain value c. That is, the requirement that c colors must be used is dropped. As small numbers are usually more suitable for interpretation, the penalty function p might be defined to assign each coloring ϕ the number of colors used by ϕ. The lower the number of colors, the less the amount of penalty. As an example, the algorithm CATREGE [4] solves (MIN-P) for such a p and X=RC. I.e., given a graph, it finds a regular c-coloring with the smallest possible c.

3.4 Single node and node pair relaxations

In single node relaxations, the properties for a single vertex to contribute to an ideal coloring are relaxed. As we have seen in Definition 3, single node definitions are only possible if the image matrix I is fixed. An example are the nodal degree relaxations for clusterings, i.e., for Part (i).

Example (Nodal Degree Relaxations.) Seidman and Foster [15] relax the requirement that every vertex must be adjacent to all other vertices of the same color by the requirement that every vertex can be non-adjacent to at most k other vertices of the same color. In an ideal coloring, the resulting subgraphs are hence not cliques, but so-called k-*plexes*. Usually, the relaxation is not penalized. That is, p is constant, say $p \equiv 0$. The search for an ideal blockmodel is hence simply the search for a partition of the vertices into k-plexes. Instead of k-plexes, the similar k-cores are sometimes used.

We now turn to the more common node *pair* relaxations. Here, the properties for same-colored vertex pairs in Definition 2 are relaxed. Two forms of p are most commonly used, which will be explained by the following two examples: p is either constant or decomposable over the set of all vertex pairs.

Example (Sociometric Cliques.) Alba [1] finds the graph theoretical definition of *clique* to be not perfectly appropriate to describe friendship (or sociometric) cliques in social networks. He thus relaxes its definition to so-called n-cliques. Here, two same-colored vertices do not need to be connected by an edge. They need to be connected by a path of length at most n, which relaxes the edge connection requirement. If no penalties are introduced, the problem (MIN-P) merely consists in the search for *any* partition into n-cliques. Similar to the n-clique are the n-clan and n-club relaxations [13].

We now treat a second common type of node pair relaxation: The *vertex similarity approaches*. The idea is to consider for each vertex pair separately, whether it should be same-colored or not. In this special case of (MIN-P), the penalty function p can thus be decomposed over all vertex pairs, i.e., $p(\phi) = \sum_{u,v \in V} p_{uv}\delta(\phi(u), \phi(v))$. Here, $p_{uv} \geq 0$ are real numbers and δ denotes the Kronecker function. It is 1 if $\phi(u) = \phi(v)$ and 0 otherwise. In literature, the numbers p_{uv} are often called *(dis)similarity values*. The relaxation technique of using such a decomposable function is called *indirect blockmodeling approach* by Doreian et al. [9].

Example (Structural Equivalence.) For X=SC, several functions p of the above form have been proposed. This propositions were made indirectly by a specification of the values p_{uv}. They quantify how much a coloring violates this dropped requirement, that is, to quantify how similar two vertices are with respect to common neighbors. See Leicht, Holme, and Newman [11] for an overview on these functions.

3.5 Subgraph relaxations

In subgraph relaxations, the requirements of Definition 1 for ideal blockmodels are relaxed.

Assume a practitioner is interested in regular 4-colorings on a given graph $G = (V, E)$. However, such a coloring does not exist on G. It is then reasonable to consider a 4-coloring ϕ to be a good solution, if it is not regular on G, but turns regular if G is changed by a very small amount. Following this idea, the best 4-coloring is the one that requires the lowest amount of changes in G in order to become regular. Possible changes are usually the deletion and addition of edges. That is, requirements of the forms "$uv \in E$" and "$uv \notin E$" are dropped. If they are penalized by the function p, then the coloring ϕ^* which requires the lowest amount of edge changes in G will be the optimal solution to (MIN-P).

In order to define a suitable penalty function p, we first need to define a function d to measure the amount of edge changes. More precisely, d measures the distance of two graphs $G = (V, E)$ and $H = (V, F)$ on the same vertex set V. A simple but common exemplary form of such a d is given by

$$d(G, H) = \sum_{u,v \in V, u \neq v} |A(G)_{u,v} - A(H)_{u,v}|, \qquad (5)$$

where A denotes the adjacency matrix of the graph. The function counts the number of different entries in the adjacency matrices of G and H. More complex distance functions are discussed below. The function d measures the distance of G to a single graph H. We can also measure its distance to a set of graphs $\mathbf{H}$, by defining the distance $d(G, \mathbf{H})$ as the distance of G to its closest element in $\mathbf{H}$. That is,

$$d(G, \mathbf{H}) := \min_{H \in \mathbf{H}} d(G, H).$$

To measure how much G has to be changed, it is compared to sets of ideal graphs $\mathcal{H}(\phi)$, on which ϕ perfectly satisfies the requirements. In our example, $\mathcal{H}(\phi)$ is defined such that ϕ is 4-regular on all $H \in \mathcal{H}(\phi)$. The penalty function for (MIN-P) is hence $p(\phi) = d(G, \mathcal{H}(\phi))$.
We now give more details on this procedure. First, we will see how ideal graphs $\mathcal{H}(\phi)$ can be defined. Then, we give an overview on the distance functions $d(G, H)$ which are used in practice. Afterwards, a common variant of this procedure is discussed, which does not relax G, but several subgraphs of G simultaneously. We close by some examples on how graph relaxation is used in literature.

Ideal, Worst and Average Graphs

Given an ideal coloring definition X (for example CC, SC, RC), a graph $G = (V, E)$ and a coloring ϕ of its vertices, the set $\mathcal{H}(\phi)$ of ideal graphs can be naturally defined. It is the set of all graphs H with the same vertex set as G, such that ϕ is an X-coloring on H. Definition 1 gives a characterization of these graphs. In the case of clustering, i. e., X = CC, the ideal graphs are those in which vertices of the same color induce complete graphs. Note that for every $\phi : V \to [c]$, the set $\mathcal{H}(\phi)$ is non-empty.

Alternatively, one can define $\mathcal{H}(\phi)$ to be the set of *worst* graphs instead of ideal ones. Worst graphs can be easily defined for CC and SC. This is because their subgraph characterization in Definition 1 use empty and complete graphs only. As "being empty" and "being complete" are opposite extremes, one can define worst graphs by interchanging the words "empty" and "complete" in the definition. E. g., in a worst graph for clustering (CC) no cluster contains any edges. If worst graphs are used, the distance of the closest graph to G needs to be maximized instead of minimized.

A third alternative has been used for CC and SC: G is compared to average graphs. For clustering, the subgraphs are hence neither empty nor complete, but have an average density. The distance of G to the average graphs $\mathcal{H}(\phi)$ can then be positive or negative, depending on whether G is worse (sparser) or better (denser) than average. The same holds hence for the penalty function. It is usually used as a reward function $\overline{p}$: The farther G is from average in the positive direction, the larger $\overline{p}$ is, and the better ϕ is.

Overview on Distance Functions

We already stated the most simple distance function to measure the distance between two graphs on the same vertex set:

$$d(G, H) = \sum_{u,v \in V, u \neq v} |A(G)_{u,v} - A(H)_{u,v}|,$$

It counts the number of edges to be added or deleted (changed) in G to obtain the ideal graph H. See Figure 3 for an example for structural 3-colorings (X = SC). The distance $d(G, \mathcal{H}(\phi))$ of the depicted coloring ϕ of the drawn graph G is 3. The reason is that 3 changes are at least necessary to obtain a structural 3-coloring: Add two edges from gray to black and delete one edge within white. Hence, the penalty value for this coloring is $p(\phi) = 3$.

If G is compared to *average* graphs, the absolute value function is a problem. Here, we want to distinguish whether G is worse or better than average. Hence, the following function is more suitable in this case.

$$d(G, H) = \sum_{u,v \in V, u \neq v} (A(G)_{u,v} - A(H)_{u,v}). \quad (6)$$

The adjacency matrix of H is possibly weighted, as average graphs usually do not have binary edge weights.

There is a third function for the case that *vertices* are relaxed instead of edges. More precisely, if requirements

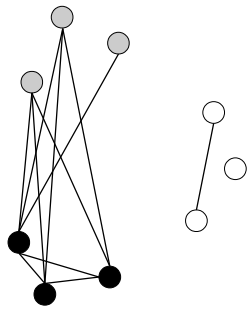

Figure 3: Example for the distance function (5) when applied to a structural 3-coloring problem.

of the form "$v \in V$" are relaxed. Note that the opposite requirement "$v \notin V$" is never relaxed, as the addition of vertices cannot contribute to the transformation of G into an ideal graph. For every coloring ϕ of the vertices in $G = (V, E)$, G is compared to a set of ideal graphs $\mathcal{H}(\phi)$. Every such graph $H = (V_H, E_H)$ in $\mathcal{H}(\phi)$ has a vertex subset $V_H \subseteq V$ and the edge set $E_H = E(V_H)$. That is, H can be obtained from G by deleting vertices together with their incident edges. A distance function needs to measure the amount of vertices to be deleted to transform G into H.

$$d(G, H) = |V(G)| - |V(H)|. \quad (7)$$

Beside these linear functions, several non-polynomial functions have been proposed. Being derived from general statistical matrix correlation measures, they can be used to compare the adjacency matrices of G and H. See Wasserman and Faust [16] or Arabie et al. [2] for an overview.

Combining Subgraph Penalties

In Definition 1, the ideal coloring conditions are formulated as requirements for the subgraphs $G_{\phi,A,B}$ of G. In the widely used *direct blockmodeling approach*, these subgraphs are relaxed separately. That is, there is a separate penalty value for each subgraph. However, the same distance function d is used for each subgraph. Whether the separate relaxations of the subgraphs is equivalent to the relaxation of G itself depends on the choice of d. In direct blockmodeling, we have single penalty values $p_{AB}(\phi) = d(G_{\phi,A,B}, \mathbf{H}_{\phi,A,B})$ for the subgraphs. They need to be combined to a total penalty value $p(\phi)$. In most cases, the p_{AB} are simply summed up:

$$p(\phi) = \sum_{A,B \in [c]} p_{AB}(\phi). \quad (8)$$

For clustering (X=CC), the sum runs clearly only over those (A, B) with $A = B$. If scaling is used, the factor is usually $1/m_{AB}$, where m_{AB} is the number of possible edges in the subgraph $G_{\phi,A,B}$. More precisely, $m_{AB} = |A| \cdot |B|$ if $A \neq B$, $m_{AA} = |A| \cdot (|A| - 1)$, and

$$p(\phi) = \sum_{A,B \in [c]} \frac{1}{m_{AB}} \cdot p_{AB}(\phi). \quad (9)$$

In some approaches, the squares of the penalties are summed up instead. This mostly occurs in so-called χ^2 *approaches*.

$$p(\phi) = \sum_{A,B \in [c]} (p_{AB}(\phi))^2. \tag{10}$$

Besides the above scaling factor, a second one can be used here. The distance of $G_{\phi,A,B}$ to $H_{\phi,A,B}$ can be seen in relation to the maximum distance $d^{\max}_{\phi,A,B}$ of any graph, on the same vertex set, to $H_{\phi,A,B}$.

$$p(\phi) = \sum_{A,B \in [c]} m_{AB} \cdot \left(\frac{p_{AB}(\phi)}{d^{\max}_{\phi,A,B}} \right)^2. \tag{11}$$

Examples

We now give some examples on how this kind of relaxation is used in literature, either for coloring type CC, SC, or RC. For each example, we need to specify the following modeling choices:

- Whether ideal, worst, or average graphs are used (and how average is defined).
- Whether edges or vertices are relaxed.
- How $p(\phi)$ is combined from the $p_{AB}(\phi)$.

Example (Cluster Performance). The *performance* of a clustering counts the number of missing edges within the clusters and adds the number of existing edges between the clusters. It is hence a measure for the clustering special case of $X = SC$. According to our classification, ideal graphs are used, edges are relaxed, and $p(\phi)$ is simply the sum of the $p_{AB}(\phi)$.

Example (Maximal Cluster Density.) A basic measure for the quality of a clustering (X=CC) on $G = (V, E)$ is the sum over all *intra-cluster densities* $\delta_{int}(V_i)$. They give the proportion of actual edges to theoretically possible edges within the i-th cluster:

$$\delta_{int}(V_i) = \frac{\text{\# internal edges of } V_i}{|V_i|(|V_i|-1)/2}.$$

The search for a coloring ϕ^* with maximum total intra-cluster density is a (MIN-P) problem. Ideal graphs are used, edges are relaxed, and the penalty values $p_{AB}(\phi)$ are linearly combined by Formula (9).

Example (Maximal Structural Density.) Wasserman and Faust explain a simple measure for structural colorings in their survey [16]. It is a generalization of the preceding example from clique to structural colorings. For each pair A, B of colors, they sum up the values $|I_{AB} - \Delta_{AB}|$. Here, I denotes the image matrix and Δ_{AB} denotes the density. The density is defined as the number of edges from A-colored to B-colored vertices, divided by the maximum possible number m_{AB} of such edges. Hence, ideal graphs are used, edges are relaxed, and the penalties $p_{AB}(\phi)$ are linearly combined by formula (9).

Example (Newman-Girvan-Modularity.) Newman and Girvan [14] present a well-known relaxation for clustering. They choose $H(\phi)$ to contain average graphs. More precisely, $H(\phi)$ consists of exactly one graph $H = (V, F)$. The edge weight of $uv \in F$ is $deg(u)deg(v)/2|E|$. This is precisely the probability of the edge to exist in a random graph with the same degree distribution as G. For this reason, H can be interpreted as the average graph w.r.t. to the degree distribution of G. Hence, average graphs are used, edges are relaxed, and the penalties $p_{AB}(\phi)$ are simply summed up (Formula (8)).

Note that $p(\phi)/2|E|$ is called the *modularity* of ϕ. The factor $1/2|E|$ is however constant and can thus be ignored in the solution of (MIN-P). Other so-called *Newman-like modularities* can be modeled analogously.

Example (Berkowitz-Carrington-Heil Index.) The index [8] is designed for structural colorings (X=SC). It compares G to an average graph H. The user is asked to specify an average density α from the interval between 0 and 1. H is then the complete graph with edge weights all α, letting its density equal α. The distance function d is (5), hence the most simple one. It is applied on subgraphs. Since the index is a χ^2 approach, the function $p(\phi)$ is composed as in (11).

Example (Vertex Relaxation.) Batagelj et. al. [3] relax vertices for regular colorings (X=RE). They use ideal graphs, relax vertices, and simply sum up the penalties $p_{AB}(\phi)$. However, they restrict the natural set $\mathcal{H}(\phi)$ of ideal graphs by allowing only those $H \in \mathcal{H}(\phi)$ for which it holds that whenever there is an edge $uv \in E$ and u is not in V_H, then v cannot be in V_H either. An optimization heuristic for this function is implemented in UCINET [5]. Brusco and Steinley [7] present an exact optimization algorithm based on an integer programming model.

4 Summary and conclusions

We present a classification for clustering and blockmodeling approaches used in practice. We show that these approaches are based on relaxations of graph theoretical coloring definitions. Basically, there are only three types of relaxations. The classification unifies link density pattern (including clustering) and link existence pattern approaches and shows the connections between them.

An obvious drawback of such a theory about used approaches is clearly its invalidity as soon as new kinds of approaches are invented. Furthermore, it does not yet cover approaches which penalize blockmodels in which the colors groups do not have similar sizes. An example is the *conductance approach* for clusterings. On the one hand, the function p minimizes the number of edges for $I_{AB} = 0$, which is a classical subgraph relaxation approach. On the other hand, p also minimizes size differences between the vertex groups. To classify this approach, the requirement for same group sizes needs to be added to the ideality definitions, such that a deviation can be penalized. We did not

include it as most approaches deal with this requirement indirectly: They exclude blockmodels with largely differing group sizes from the set $X_L(G)$ of feasible blockmodels.

However, we also see two kinds of practical benefits. First, the classification can be used to think about the "missing" approaches. For example, approaches which use average graphs usually compare G to a single average graph H, whose edge weights are fractional. This choice seems to be arbitrary, as one could also use a whole set $\mathcal{H}(\phi)$ of unweighted average graphs for the comparison to G. The latter idea is standard if ideal instead of average graphs are used. Second, the question which approach is the most suitable one for a given network can now be answered stepwise: Are ideal or average graphs more suitable, should edges or vertices be relaxed, should node pairs or subgraphs be relaxed, how should subgraph penalties be combined, etc.?

References

[1] R. D. Alba. A graph-theoretic definition of a sociometric clique. *Journal of Mathematical Sociology*, 3(1):113–126, 1973.

[2] P. Arabie, S. A. Boorman, and P. R. Levitt. Constructing blockmodels: How and why. *Journal of Mathematical Psychology*, 17(1):21–63, 1978.

[3] V. Batagelj, P. Doreian, and A. Ferligoj. An optimizational approach to regular equivalence. *Social Networks*, 14(1):121–135, 1992.

[4] S. P. Borgatti and M. G. Everett. Two algorithms for computing regular equivalence. *Social Networks*, 15(4):361–376, 1993.

[5] S. P. Borgatti, M. G. Everett, and L. C. Freeman. Ucinet for windows: Software for social network analysis. 2002.

[6] U. Brandes and J. Lerner. Structural similarity: spectral methods for relaxed blockmodeling. *Journal of Classification*, 27(3):279–306, 2010.

[7] M. J. Brusco and D. Steinley. Integer programs for one-and two-mode blockmodeling based on prespecified image matrices for structural and regular equivalence. *Journal of Mathematical Psychology*, 53(6):577–585, 2009.

[8] P. J. Carrington, G. H. Heil, and S. D. Berkowitz. A goodness-of-fit index for blockmodels. *Social Networks*, 2(3):219–234, 1980.

[9] P. Doreian, V. Batagelj, and A. Ferligoj. *Generalized blockmodeling*, volume 25. Cambridge University Press, 2005.

[10] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2009.

[11] E. Leicht, P. Holme, and M. Newman. Vertex similarity in networks. *Physical Review E*, 73(2):026120, 2006.

[12] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, 1(1):49–80, 1971.

[13] R. J. Mokken. Cliques, clubs and clans. *Quality and Quantity*, 13:161–173, 1979.

[14] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[15] S. Seidman and B. Foster. A graph-theoretic generalization of the clique concept. *Journal of Mathematical Sociology*, 6:139–154, 1978.

[16] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.