On Modularity Clustering

Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Görke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner

Abstract—Modularity is a recently introduced quality measure for graph clusterings. It has immediately received considerable attention in several disciplines, particularly in the complex systems literature, although its properties are not well understood. We study the problem of finding clusterings with maximum modularity, thus providing theoretical foundations for past and present work based on this measure. More precisely, we prove the conjectured hardness of maximizing modularity both in the general case and with the restriction to cuts and give an Integer Linear Programming formulation. This is complemented by first insights into the behavior and performance of the commonly applied greedy agglomerative approach.

Index Terms—Graph clustering, graph partitioning, modularity, community structure, greedy algorithm.

1 Introduction

Graph clustering is a fundamental problem in the analysis of relational data. Studied for decades and applied to many settings, it is now popularly referred to as the problem of partitioning networks into communities. In this line of research, a novel graph clustering index called *modularity* has recently been proposed [1]. The rapidly growing interest in this measure prompted a series of follow-up studies on various applications and possible adjustments (for example, see [2], [3], [4], [5], [6]). Moreover, an array of heuristic algorithms has been proposed to optimize modularity. These are based on a greedy agglomeration [7], [8], spectral division [9], [10], simulated annealing [11], [12], or extremal optimization [13] to name a few prominent examples. Although these studies often provide plausibility arguments in favor of the resulting partitions, we know of only one attempt to characterize properties of clusterings with maximum modularity [2]. In particular, none of the proposed algorithms has been shown to produce optimal partitions with respect to modularity.

In this paper, we study the problem of finding clusterings with maximum modularity, thus providing theoretical foundations for the past and present work based on this measure. More precisely, we prove the conjectured hardness of maximizing modularity both in the general case and the restriction to cuts and give an integer linear programming (ILP) formulation to facilitate optimization without enumeration of all clusterings. Since the most commonly employed heuristic to optimize modularity is based on greedy agglomeration, we investigate its worst-case behavior. In fact, we give a graph family for which the greedy approach yields an approximation factor no better than two. In addition, our examples indicate that the quality of greedy clusterings may heavily depend on the tie-breaking strategy utilized. In fact, in the worst case, no approximation factor can be provided. These performance studies are concluded by optimally partitioning some previously considered networks, which yields further insight.

This paper is organized as follows: Section 2 shortly introduces preliminaries, formulations of modularity, and an ILP formulation of the problem. The basic and counterintuitive properties of modularity are observed in Section 3. Our $\mathcal{NP}$-completeness proofs are given in Section 4, followed by an analysis of the greedy approach in Section 5. The theoretical investigation is extended by characterizations of the optimum clusterings for cliques and cycles in Section 6. Our work is concluded by revisiting examples from previous work in Section 7 and a brief discussion in Section 8.

2 Preliminaries

Throughout this paper, we will use the notation in [14]. More precisely, we assume that $G = (V, E)$ is an undirected connected graph with $n := |V|$ vertices and $m := |E|$ edges. Denote by $\mathcal{C} = \{C_1, \dots, C_k\}$ a partition of V. We call $\mathcal{C}$ a *clustering* of G and the C_i, which are required to be nonempty, *clusters*. $\mathcal{C}$ is called *trivial* if either $k = 1$ or $k = n$. We denote the set of all possible clusterings of a graph G with $\mathcal{A}(G)$. In the following, we often identify a cluster C_i with the induced subgraph of G, that is, the graph $G[C_i] := (C_i, E(C_i))$, where

$$E(C_i) := \{\{v, w\} \in E : v, w \in C_i\}.$$

Then, $E(\mathcal{C}) := \bigcup_{i=1}^{k} E(C_i)$ is the set of *intracluster edges*, and $E \setminus E(\mathcal{C})$ is the set of *intercluster edges*. The number of intracluster edges is denoted by $m(\mathcal{C})$, and the number of intercluster edges by $\overline{m}(\mathcal{C})$. The set of edges that have one

- U. Brandes is with the Department of Computer and Information Science, University of Konstanz, Box D 67, 78457 Konstanz, Germany. E-mail: brandes@inf.uni-konstanz.de.
- D. Delling, M. Gaertler, R. Görke, and D. Wagner are with the Faculty of Informatics, Universität Karlsruhe (TH), ITI Wagner, Box 6980, 76128 Karlsruhe, Germany. E-mail: {delling, gaertler, goerke, wagner}@ira.uka.de.
- M. Hoefer is with the Department of Computer Science, RWTH Aachen, Lehrstuhl Informatik I, Ahornstrasse 55, 52074 Aachen, Germany. E-mail: mhoefer@cs.rwth-aachen.de.
- Z. Nikoloski is with the Max-Planck Institute of Molecular Plant Physiology, Bioinformatics Group, Am Mühlenberg 1, 14476 Potsdam-Golm, Germany. E-mail: nikoloski@mpimp-golm.mpg.de.

Manuscript received 6 Mar. 2007; revised 27 June 2007; accepted 5 Sept. 2007; published online 20 Sept. 2007.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2007-03-0096. Digital Object Identifier no. 10.1109/TKDE.2007.190689.

end node in C_i and the other end node in C_j is denoted by $E(C_i, C_j)$.

2.1 Definition of Modularity

Modularity is a quality index for clustering. Given a simple graph $G = (V, E)$, we follow [1] and define the *modularity* $\mathrm{q}(\mathcal{C})$ of a clustering $\mathcal{C}$ as

$$\mathrm{q}(\mathcal{C}) := \sum_{C\in\mathcal{C}}\left[\frac{|E(C)|}{m} - \left(\frac{|E(C)| + \sum_{C'\in\mathcal{C}}|E(C,C')|}{2m}\right)^2\right]. \tag{1}$$

Note that C' ranges over all clusters so that edges in $E(C)$ are counted twice in the squared expression. This is to adjust proportions, since edges in $E(C, C')$ and $C \neq C'$ are counted twice as well, once for each ordering of the arguments. Note that we can rewrite (1) into a more convenient form:

$$\mathrm{q}(\mathcal{C}) = \sum_{C\in\mathcal{C}}\left[\frac{|E(C)|}{m} - \left(\frac{\sum_{v\in C}\deg(v)}{2m}\right)^2\right]. \tag{2}$$

This reveals an inherent trade-off. To maximize the first term, many edges should be contained in clusters, whereas the minimization of the second term is achieved by splitting the graph into many clusters, with small total degrees each. Note that the first term $|E(\mathcal{C})|/m$ is also known as *coverage* [14].

2.2 Maximizing Modularity via Integer Linear Programming

The problem of maximizing modularity can be cast into a very simple and intuitive ILP. Given a graph $G = (V, E)$ with $n := |V|$ nodes, we define n^2 decision variables $X_{uv} \in \{0, 1\}$, one for every pair of nodes $u, v \in V$. The key idea is that these variables can be interpreted as an equivalence relation (over V) and thus form a clustering. In order to ensure consistency, we need the following constraints, which guarantee

$$\begin{array}{lll} \text{reflexivity} & \forall\, u: \; X_{uu} = 1, & \\ \text{symmetry} & \forall\, u,\, v: X_{uv} = X_{vu}, \quad \text{and} & \\ \text{transitivity} & \forall\, u,v,w: \left\{\begin{array}{lcl} X_{uv} + X_{vw} - 2\cdot X_{uw} & \leq & 1 \\ X_{uw} + X_{uv} - 2\cdot X_{vw} & \leq & 1 \\ X_{vw} + X_{uw} - 2\cdot X_{uv} & \leq & 1. \end{array}\right. \end{array}$$

The objective function of modularity then becomes

$$\frac{1}{2m}\sum_{(u,v)\in V^2}\left(E_{uv} - \frac{\deg(u)\deg(v)}{2m}\right)X_{uv},$$

$$\text{with} \qquad E_{uv} = \begin{cases} 1, & \text{if } (u,v) \in E, \\ 0, & \text{otherwise.} \end{cases}$$

Note that this ILP can be simplified by pruning redundant variables and constraints, leaving only $\binom{n}{2}$ variables and $\binom{n}{3}$ constraints.

3 Fundamental Observations

In the following, we identify the basic structural properties that clusterings with maximum modularity fulfill. We first focus on the range of modularity, for which Lemma 3.1 gives the lower bound and upper bound.

Lemma 3.1. *Let G be an undirected and unweighted graph and let $\mathcal{C} \in \mathcal{A}(G)$. Then, $-1/2 \leq \mathrm{q}(\mathcal{C}) \leq 1$ holds.*

Proof. Let $m_i = |E(C)|$ be the number of edges inside cluster C and let $m_e = \sum_{C\neq C'\in\mathcal{C}}|E(C,C')|$ be the number of edges that have exactly one end node in C. Then, the contribution of C to $\mathrm{q}(\mathcal{C})$ is

$$\frac{m_i}{m} - \left(\frac{m_i}{m} + \frac{m_e}{2m}\right)^2.$$

This expression is strictly decreasing in m_e, and when varying m_i, the only maximum point is at $m_i = (m - m_e)/2$. The contribution of a cluster is minimized when m_i is zero and m_e is as large as possible. Supposing now that $m_i = 0$, using the inequality $(a + b)^2 \geq a^2 + b^2$ for all nonnegative numbers a and b, the modularity has a minimum score for two clusters where all edges are intercluster edges. The upper bound is obvious from our reformulation in (2) and has previously been observed [2], [3], [15]. It can only be actually attained in the specific case of a graph with no edges, where the coverage is defined to be 1. □

As a result, any bipartite graph $K_{a,b}$ with the canonic clustering $\mathcal{C} = \{C_a, C_b\}$ yields the minimum modularity of $-1/2$. The following results characterize the structure of a clustering with maximum modularity.

Corollary 3.2. *Isolated nodes have no impact on modularity.*

Corollary 3.2 directly follows from the fact that modularity depends on edges and degrees; thus, an isolated node does not contribute, regardless of its association to a cluster. Therefore, we exclude isolated nodes from further consideration in this work; that is, all nodes are assumed to be of degree greater than zero.

Lemma 3.3. *A clustering with maximum modularity has no cluster that consists of a single node with a degree of 1.*

Proof. Suppose, for contradiction, that there is a clustering $\mathcal{C}$ with a cluster $C_v = \{v\}$ and $\deg(v) = 1$. Consider a cluster C_u that contains the neighbor node u. Suppose there are a number of m_i intracluster edges in C_u and m_e intercluster edges that connect C_u to other clusters. Together, these clusters add

$$\frac{m_i}{m} - \frac{(2m_i + m_e)^2 + 1}{4m^2},$$

to $\mathrm{q}(\mathcal{C})$. Merging C_v with C_u results in a new contribution of

$$\frac{m_i + 1}{m} - \frac{(2m_i + m_e + 1)^2}{4m^2}.$$

The merge yields an increase of

$$\frac{1}{m} - \frac{2m_i + m_e}{2m^2} > 0,$$

in modularity, because $m_i + m_e \leq m$, and $m_e \geq 1$. This proves the lemma. □

Lemma 3.4. *There is always a clustering with maximum modularity, in which each cluster consists of a connected subgraph.*

Proof. Consider, for contradiction, a clustering $\mathcal{C}$ with a cluster C of m_i intracluster and m_e intercluster edges that consists of a set of more than one connected subgraph. The subgraphs in C do not have to be disconnected in G: They are only disconnected when we consider the edges $E(C)$. Cluster C adds

$$\frac{m_i}{m} - \frac{(2m_i + m_e)^2}{4m^2},$$

to $\mathrm{q}(\mathcal{C})$. Now, suppose we create a new clustering $\mathcal{C}'$ by splitting C into two new clusters. Let one cluster C_v consist of the component that includes node v, that is, all nodes, which can be reached from a node v with a path running only through nodes of C. That is, $C_v = \bigcup_{i=1}^{\infty} C_v^i$, where $C_v^i = \{w | \exists (w, w_i) \in E(C) \text{ with } w_i \in C_v^{i-1}\}$, and $C_v^0 = \{v\}$. The other nonempty cluster is given by $C - C_v$. Let C_v have m_i^v intracluster and m_e^v intercluster edges. Together, the new clusters add

$$\frac{m_i}{m} - \frac{(2m_i^v + m_e^v)^2 + (2(m - m_i^v) + m - m_e^v)^2}{4m^2},$$

to $\mathrm{q}(\mathcal{C}')$. For a, $b \geq 0$, obviously, $a^2 + b^2 \leq (a+b)^2$ and, hence, $\mathrm{q}(\mathcal{C}') \geq \mathrm{q}(\mathcal{C})$. □

Corollary 3.5. *A clustering of maximum modularity does not include disconnected clusters.*

Corollary 3.5 directly follows from Lemma 3.4 and from the exclusion of isolated nodes. Thus, the search for an optimum can be restricted to clusterings, in which clusters are connected subgraphs, and there are no clusters that consist of nodes with a degree of 1.

3.1 Counterintuitive Behavior

In the previous section, we listed some intuitive properties like connectivity within clusters for clusterings of maximum modularity. However, due to the enforced balance between coverage and the sums of squared cluster degrees, counterintuitive situations arise. These are nonlocality, scaling behavior, and sensitivity to satellites.

1. *Nonlocality.* At first glance, modularity seems to be a local quality measure. Recalling (2), each cluster contributes separately. However, the example presented in Figs. 1a and 1b exhibit a typical nonlocal behavior. In these figures, clusters are represented by color. By adding an additional node that is connected to the leftmost node, the optimal clustering is completely altered. According to Lemma 3.3, the additional node has to be clustered together with the leftmost node. This leads to a shift of the rightmost black node from the black cluster to the white cluster, although locally, its neighborhood structure has not changed.
2. *Sensitivity to Satellites.* A *clique with leaves* is a graph of $2n$ nodes that consists of a clique K_n and n *leaf* nodes of degree 1 such that each node of the clique is connected to exactly one leaf node. For a clique, we show in Section 6 that the trivial clustering, with $k = 1$, has maximum modularity. For a clique with leaves, however, the optimal clustering changes to $k = n$ clusters, in which each cluster consists of a connected pair of leaf and clique nodes. Fig. 1c shows an example.
3. *Scaling Behavior.* Figs. 1c and 1d display the scaling behavior of modularity. By simply doubling the graph presented in Fig. 1c, the optimal clustering is completely altered. Although in Fig. 1c, we obtain three clusters, each consisting of the minor K_2, the clustering with maximum modularity of the graph in Fig. 1d consists of two clusters, each being a graph that is equal to the one in Fig. 1c.

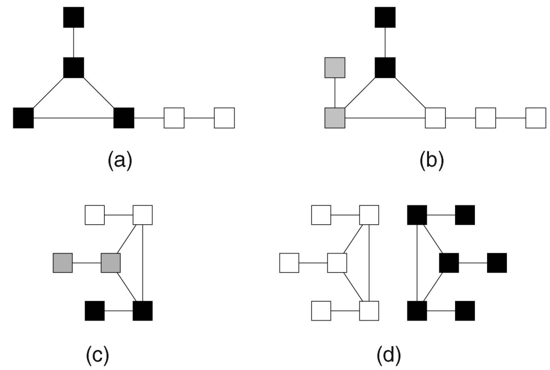

Fig. 1. (a) and (b) Nonlocal behavior. (c) A clique K_3 with leaves. (d) Scaling behavior. Clusters are represented by colors.

This behavior is in line with the previous observations in [2], [4] that the size and structure of clusters in the optimum clustering depend on the total number of links in the network. Hence, clusters that are identified in smaller graphs might be combined to a larger cluster in an optimum clustering of a larger graph. The formulation of (2) mathematically explains this observation as the modularity optimization strives at optimizing the trade-off between coverage and degree sums. This provides a rigorous understanding of the observations made in [2], [4].

4 $\mathcal{NP}$-Completeness

It has been conjectured that maximizing modularity is hard [8], but no formal proof was provided to date. We next show that the decision version of modularity maximization is indeed $\mathcal{NP}$-complete.

Problem 1 (MODULARITY). *Given a graph G and a number K, is there a clustering $\mathcal{C}$ of G, for which $\mathrm{q}(\mathcal{C}) \geq K$?*

Note that we may ignore the fact that, in principle, K could be a real number in the range $[-1/2, 1]$, because $4m^2 \cdot \mathrm{q}(\mathcal{C})$ is an integer for every partition $\mathcal{C}$ of G and is polynomially bounded in the size of G. Our hardness result for MODULARITY is based on a transformation from the following decision problem.

Problem 2 (3-PARTITION). *Given $3k$ positive integer numbers $a_1, \ldots, a_{3k}$ such that the sum $\sum_{i=1}^{3k} a_i = kb$ and $b/4 < a_i < b/2$ for an integer b and for all $i = 1, \ldots, 3k$, is there a partition of these numbers into k sets such that the numbers in each set sum up to b?*

We show that an instance $A = \{a_1, \ldots, a_{3k}\}$ of 3-PARTITION can be transformed into an instance $(G(A), K(A))$ of MODULARITY such that $G(A)$ has a clustering with

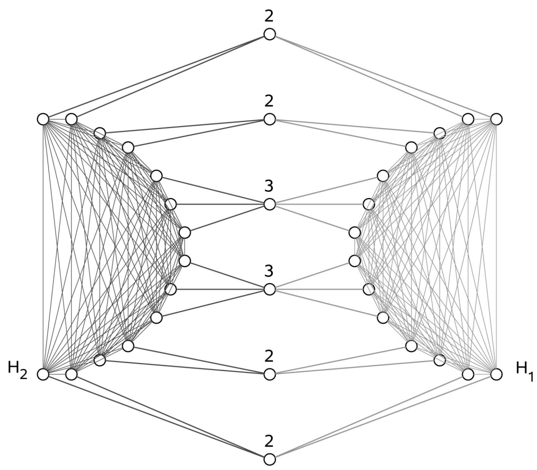

Fig. 2. An example graph $G(A)$ for the instance $A = \{2, 2, 2, 2, 3, 3\}$ of 3-PARTITION. Node labels indicate the corresponding numbers $a_i \in A$.

modularity at least $K(A)$ if and only if $a_1, \ldots, a_{3k}$ can be partitioned into k sets of sum $b = 1/k \cdot \sum_{i=1}^{k} a_i$ each.

It is crucial that 3-PARTITION is *strongly* $\mathcal{NP}$-complete [16]; that is, the problem remains $\mathcal{NP}$-complete, even if the input is represented in unary coding. This implies that no algorithm can decide the problem in time polynomial, even in the sum of the input values, unless $\mathcal{P} = \mathcal{NP}$. More importantly, it implies that our transformation only needs to be pseudopolynomial.

The reduction is defined as follows: Given an instance A of 3-PARTITION, construct a graph $G(A)$ with k cliques (completely connected subgraphs) $H_1, \ldots, H_k$ of size $a = \sum_{i=1}^{3k} a_i$ each. For each element $a_i \in A$, we introduce a single *element node* and connect it to a_i nodes in each of the k cliques in such a way that each clique member is connected to exactly one element node. It is easy to see that each clique node then has degree a, and the element node that corresponds to element $a_i \in A$ has degree ka_i. The number of edges in $G(A)$ is $m = k/2 \cdot a(a+1)$. See Fig. 2 for an example. Note that the size of $G(A)$ is polynomial in the unary coding size of A so that our transformation is indeed pseudopolynomial.

Before specifying bound $K(A)$ for the instance of MODULARITY, we will show three properties of maximum modularity clusterings of $G(A)$. Together, these properties establish the desired characterization of solutions for 3-PARTITION by solutions for MODULARITY.

Lemma 4.1. *In a maximum modularity clustering of $G(A)$, none of the cliques $H_1, \ldots, H_k$ is split.*

We prove the lemma by showing that every clustering that violates the above condition can be modified to strictly improve the modularity.

Proof. We consider a clustering $\mathcal{C}$ that splits a clique $H \in \{H_1, \ldots, H_k\}$ into different clusters and then show how we can obtain a clustering with strictly higher modularity. Suppose that $C_1, \ldots, C_r \in \mathcal{C}$, and $r > 1$ are the clusters that contain nodes of H. For $i = 1, \ldots, r$, we denote by n_i the number of nodes of H contained in cluster C_i, by $m_i = |E(C_i)|$ the number of edges between nodes in C_i, and by f_i the number of edges between nodes of H in C_i and element nodes in C_i. d_i is the sum of degrees of all nodes in C_i. The contribution of $C_1, \ldots, C_r$ to $\mathrm{q}(\mathcal{C})$ is

$$\frac{1}{m}\sum_{i=1}^{r} m_i - \frac{1}{4m^2}\sum_{i=1}^{r} d_i^2.$$

Now, suppose we create a clustering $\mathcal{C}'$ by rearranging the nodes in $C_1, \ldots, C_r$ into clusters $C', C_1', \ldots, C_r'$ such that C' contains exactly the nodes of clique H and each C_i', $1 \le i \le r$, the remaining elements of C_i (if any). In this new clustering, the number of covered edges reduces by $\sum_{i=1}^{r} f_i$, because all nodes from H are removed from the clusters C_i'. This labels the edges that connect the clique nodes to other nonclique nodes of C_i as intercluster edges. For H itself, there are $\sum_{i=1}^{r}\sum_{j=i+1}^{r} n_i n_j$ edges that are now additionally covered due to the creation of cluster C'. In terms of degrees, the new cluster C' contains a nodes of degree a. The sums for the remaining clusters C_i' are reduced by the degrees of the clique nodes, as these nodes are now in C'. Thus, the contribution of these clusters to $\mathrm{q}(\mathcal{C}')$ is given by

$$\frac{1}{m}\sum_{i=1}^{r}\left(m_i + \sum_{j=i+1}^{r} n_i n_j - f_i\right) - \frac{1}{4m^2}\left(a^4 + \sum_{i=1}^{r}(d_i - n_i a)^2\right).$$

Setting $\Delta := \mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C})$, we obtain

$$\begin{aligned}
\Delta &= \frac{1}{m}\left(\sum_{i=1}^{r}\sum_{j=i+1}^{r} n_i n_j - f_i\right) \\
&\quad + \frac{1}{4m^2}\left(\left(\sum_{i=1}^{r} 2d_i n_i a - n_i^2 a^2\right) - a^4\right) \\
&= \frac{1}{4m^2}\left((4m\sum_{i=1}^{r}\sum_{j=i+1}^{r} n_i n_j - 4m\sum_{i=1}^{r} f_i\right. \\
&\quad \left. + \left(\sum_{i=1}^{r} n_i\big(2d_i a - n_i a^2\big)\right) - a^4\right).
\end{aligned}$$

Using the equation that

$$2\sum_{i=1}^{r}\sum_{j=i+1}^{r} n_i n_j = \sum_{i=1}^{r}\sum_{j\neq i} n_i n_j,$$

by substituting $m = \frac{k}{2}a(a+1)$ and rearranging the terms, we get

$$\begin{aligned}
\Delta &= \frac{a}{4m^2}\left(-a^3 - 2k(a+1)\sum_{i=1}^{r} f_i\right. \\
&\quad \left. + \sum_{i=1}^{r} n_i\left(2d_i - n_i a + k(a+1)\sum_{j\neq i} n_j\right)\right) \\
&\ge \frac{a}{4m^2}\left(-a^3 - 2k(a+1)\sum_{i=1}^{r} f_i\right. \\
&\quad \left. + \sum_{i=1}^{r} n_i\left(n_i a + 2kf_i + k(a+1)\sum_{j\neq i}^{r} n_j\right)\right).
\end{aligned}$$

For the last inequality, we use the fact that $d_i \ge n_i a + kf_i$. This inequality holds, because C_i contains at least the n_i nodes of degree a from the clique H. In addition, it contains both the clique and element nodes for each edge counted in f_i. For each such edge, there are $k-1$ other edges that connect the element node to the $k-1$ other cliques. Hence, we get a contribution of kf_i in the degrees of the element nodes. Combining the terms n_i and one of the terms $\sum_{j\neq i} n_j$, we obtain

$$\begin{aligned}\Delta \geq\ & \frac{a}{4m^2}\left(-a^3 - 2k(a+1)\sum_{i=1}^{r} f_i\right) \\ &+ \frac{a}{4m^2}\left(\sum_{i=1}^{r} n_i\left(a\sum_{j=1}^{r} n_j + 2kf_i \right.\right. \\ &\left.\left.+ ((k-1)a+k)\sum_{j\neq i}^{r} n_j\right)\right) \\ =\ & \frac{a}{4m^2}\left(-2k(a+1)\sum_{i=1}^{r} f_i \right. \\ &\left.+ \sum_{i=1}^{r} n_i\left(2kf_i + ((k-1)a+k)\sum_{j\neq i}^{r} n_j\right)\right) \\ =\ & \frac{a}{4m^2}\left(\sum_{i=1}^{r} 2kf_i(n_i - a - 1))\right. \\ &\left.+ ((k-1)a+k)\sum_{i=1}^{r}\sum_{j\neq i}^{r} n_i n_j\right) \\ \geq\ & \frac{a}{4m^2}\left(\sum_{i=1}^{r} 2kn_i(n_i - a - 1)\right. \\ &\left.+ ((k-1)a+k)\sum_{i=1}^{r}\sum_{j\neq i}^{r} n_i n_j\right).\end{aligned}$$

For the last step, we note that $n_i \leq a-1$, and $n_i - a - 1 < 0$ for all $i = 1, \ldots, r$. Thus, increasing f_i decreases the modularity difference. For each node of H, there is at most one edge to a node that is not in H and, thus, $f_i \leq n_i$.

By rearranging terms and using the inequality $a \geq 3k$, we get

$$\begin{aligned}\Delta \geq\ & \frac{a}{4m^2}\sum_{i=1}^{r} n_i\left(2k(n_i - a - 1)\right. \\ &\left.+ ((k-1)a+k)\sum_{j\neq i}^{r} n_j\right) \\ =\ & \frac{a}{4m^2}\sum_{i=1}^{r} n_i\left(-2k + ((k-1)a-k)\sum_{j\neq i}^{r} n_j\right) \\ \geq\ & \frac{a}{4m^2}((k-1)a - 3k)\sum_{i=1}^{r}\sum_{j\neq i}^{r} n_i n_j \\ \geq\ & \frac{3k^2}{4m^2}(3k-6)\sum_{i=1}^{r}\sum_{j\neq i}^{r} n_i n_j.\end{aligned}$$

As we can assume that $k > 2$ for all relevant instances of 3-PARTITION, we obtain $\Delta > 0$. This shows that any clustering can be improved by completely merging each clique into a cluster. □

Next, we observe that the optimum clustering completely places at most one clique into a single cluster.

Lemma 4.2. *In a maximum modularity clustering of $G(A)$, every cluster contains at most one of the cliques $H_1, \ldots, H_k$.*

Proof. Consider a maximum modularity clustering. Lemma 4.1 shows that each of the k cliques $H_1, \ldots, H_k$ is entirely contained in one cluster. Assume that there is a cluster C that contains at least two of the cliques. If C does not contain any element nodes, then the cliques form disconnected components in the cluster. In this case, it is easy to see that the clustering can be improved by splitting C into distinct clusters, one for each clique. This way, we keep the number of edges within clusters the same; however, we reduce the squared degree sums of clusters.

Otherwise, we assume that C completely contains $l > 1$ cliques and, in addition, some element nodes of elements a_j, with $j \in J \subseteq \{1, \ldots, k\}$. Note that inside the l cliques, $la(a-1)/2$ edges are covered. In addition, for every element node that corresponds to an element a_j, there are la_j edges included. The degree sum of the cluster is given by the la clique nodes of degree a and a number of element nodes of degree ka_j. The contribution of C to $\mathrm{q}(\mathcal{C})$ is thus given by

$$\frac{1}{m}\left(\frac{l}{2}a(a-1) + l\sum_{j\in J} a_j\right) - \frac{1}{4m^2}\left(la^2 + k\sum_{j\in J} a_j\right)^2.$$

Now, suppose that we create $\mathcal{C}'$ by splitting C into C_1' and C_2' such that C_1' completely contains a single clique H. This leaves the number of edges covered within the cliques the same; however, all edges from H to the included element nodes eventually drop out. The degree sum of C_1' is exactly a^2 and, so, the contribution of C_1' and C_2' to $\mathrm{q}(\mathcal{C}')$ is given by

$$\begin{aligned}&\frac{1}{m}\left(\frac{l}{2}a(a-1) + (l-1)\sum_{j\in J} a_j\right) \\ &- \frac{1}{4m^2}\left(\left((l-1)a^2 + k\sum_{j\in J} a_j\right)^2 + a^4\right).\end{aligned}$$

Considering the difference, we note that

$$\begin{aligned}\mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C}) - =\ & -\frac{1}{m}\sum_{j\in J} a_j \\ &+ \frac{1}{4m^2}\left((2l-1)a^4 + 2ka^2\sum_{j\in J} a_j - a^4\right) \\ =\ & \frac{2(l-1)a^4 + 2ka^2\sum_{j\in J} a_j}{4m^2} \\ &- \frac{4m\sum_{j\in J} a_j}{4m^2} \\ =\ & \frac{2(l-1)a^4 - 2ka\sum_{j\in J} a_j}{4m^2} \\ \geq\ & \frac{9k^3}{2m^2}(9k-1) \\ >\ & 0,\end{aligned}$$

as $k > 0$ for all instances of 3-PARTITION.

Since the clustering is improved in every case, it is not optimal. This is a contradiction. □

The previous two lemmas show that any clustering can be strictly improved to a clustering that contains k *clique clusters* such that each one completely contains one of the cliques $H_1, \ldots, H_k$ (possibly plus some additional element nodes). In particular, this must hold for the optimum clustering as well. Now that we know how the cliques are clustered, we turn to the element nodes.

As they are not directly connected, it is never optimal to create a cluster that consists only of element nodes. Splitting such a cluster into singleton clusters, one for each element node, reduces the squared degree sums but keeps the edge coverage at the same value. Hence, such a split yields a clustering with strictly higher modularity. The next lemma shows that we can further strictly improve the modularity of a clustering with a singleton cluster of an element node by joining it with one of the clique clusters.

Lemma 4.3. *In a maximum modularity clustering of $G(A)$, there is no cluster composed of element nodes only.*

Proof. Consider a clustering $\mathcal{C}$ of maximum modularity and suppose that there is an element node v_i that corresponds to the element a_i, which is not part of any clique cluster. As argued above, we can improve such a clustering by creating a singleton cluster $C = \{v_i\}$. Suppose C_{min} is the clique cluster, for which the sum of degrees is minimal. We know that C_{min} contains all nodes from a clique H and, eventually, some other element nodes for elements a_j, with $j \in J$, for some index set J. The cluster C_{min} covers all $a(a-1)/2$ edges within H and $\sum_{j\in J} a_j$ edges to element nodes. The degree sum is a^2 for clique nodes and $k\sum_{j\in J} a_j$ for element nodes. As C is a singleton cluster, it covers no edges, and the degree sum is ka_i. This yields a contribution of C and C_{min} to $\mathrm{q}(\mathcal{C})$ of

$$\frac{1}{m}\left(\frac{a(a-1)}{2} + \sum_{j\in J} a_j\right) - \frac{1}{4m^2}\left(\left(a^2 + k\sum_{j\in J} a_j\right)^2 + k^2 a_i^2\right).$$

Again, we create a different clustering $\mathcal{C}'$ by joining C and C_{min} to a new cluster C'. This increases the edge coverage by a_i. The new cluster C' has the sum of degrees of both previous clusters. The contribution of C' to $\mathrm{q}(\mathcal{C}')$ is given by

$$\frac{1}{m}\left(\frac{a(a-1)}{2} + a_i + \sum_{j\in J} a_j\right) - \frac{1}{4m^2}\left(a^2 + ka_i + k\sum_{j\in J} a_j\right)^2.$$

Thus,

$$\begin{aligned}\mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C}) &= \frac{a_i}{m} - \frac{1}{4m^2}\left(2ka^2 a_i + 2k^2 a_i \sum_{j\in J} a_j\right)\\ &= \frac{1}{4m^2}\left(2ka(a+1)a_i - 2ka^2 a_i \right.\\ &\qquad \left. - 2k^2 a_i \sum_{j\in J} a_j\right)\\ &= \frac{a_i}{4m^2}\left(2ka - 2k^2 \sum_{j\in J} a_j\right).\end{aligned}$$

At this point, recall that C_{min} is the clique cluster with the minimum degree sum. For this cluster, the elements that correspond to the included element nodes can never sum to more than a/k. In particular, as v_i is not part of any clique cluster, the elements of nodes in C_{min} can never sum to more than $(a - a_i)/k$. Thus,

$$\sum_{j\in J} a_j \leq \frac{1}{k}(a - a_i) < \frac{1}{k} a$$

and, so, $\mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C}) > 0$. This contradicts the assumption that $\mathcal{C}$ is optimal. □

We have shown that for the graphs $G(A)$, the clustering of maximum modularity consists of exactly k clique clusters, and each element node belongs to exactly one of the clique clusters. Combining the above results, we now state our main result.

Theorem 4.4. *MODULARITY is strongly $\mathcal{NP}$-complete.*

Proof. For a given clustering $\mathcal{C}$ of $G(A)$, we can check in polynomial time whether $\mathrm{q}(\mathcal{C}) \geq K(A)$, so clearly, MODULARITY $\in \mathcal{NP}$.

For $\mathcal{NP}$-completeness, we transform an instance $A = \{a_1, \ldots, a_{3k}\}$ of 3-PARTITION into an instance $(G(A), K(A))$ of MODULARITY. We have already outlined the construction of the graph $G(A)$ above. For the correct parameter $K(A)$, we consider a clustering in $G(A)$ with the properties derived in the previous lemmas, that is, a clustering with exactly k clique clusters. Any such clustering yields exactly $(k-1)a$ intercluster edges, so the edge coverage is given by

$$\begin{aligned}\sum_{C\in\mathcal{C}} \frac{|E(C)|}{m} &= \frac{m - (k-1)a}{m}\\ &= 1 - \frac{2(k-1)a}{ka(a+1)} = 1 - \frac{2k-2}{k(a+1)}.\end{aligned}$$

Hence, the clustering $\mathcal{C} = (C_1, \ldots, C_k)$ with maximum modularity must minimize

$$\deg(C_1)^2 + \deg(C_2)^2 + \ldots + \deg(C_k)^2.$$

This requires a distribution of the element nodes between the clusters, which is as even as possible with respect to the sum of degrees per cluster. In the optimum case, we can assign to each cluster element nodes that correspond to elements that sum to $b = 1/k \cdot a$. In this case, the sum of degrees of element nodes in each clique cluster is equal to $k \cdot 1/k \cdot a = a$. This yields $\deg(C_i) = a^2 + a$ for each clique cluster C_i, $i = 1, \ldots, k$, and gives

$$\deg(C_1)^2 + \ldots + \deg(C_k)^2 \geq k(a^2 + a)^2 = ka^2(a+1)^2.$$

Equality holds only in the case that an assignment of b to each cluster is possible. Hence, if there is a clustering $\mathcal{C}$ with $\mathrm{q}(\mathcal{C})$ of at least

$$K(A) = 1 - \frac{2k-2}{k(a+1)} - \frac{ka^2(a+1)^2}{k^2a^2(a+1)^2} = \frac{(k-1)(a-1)}{k(a+1)},$$

then we know that this clustering must perfectly split the element nodes to the k clique clusters. As each element node is contained in exactly one cluster, this yields a solution for the instance of 3-PARTITION. With this choice of $K(A)$, the instance $(G(A), K(A))$ of MODULARITY is satisfiable only if the instance A of 3-PARTITION is satisfiable.

Otherwise, suppose the instance for 3-PARTITION is satisfiable. Then, there is a partition into k sets such that the sum over each set is $1/k \cdot a$. If we cluster the corresponding graph by joining the element nodes of each set with a different clique, we get a clustering of modularity $K(A)$. This shows that the instance $(G(A), K(A))$ of MODULARITY is satisfiable if the instance A of 3-PARTITION is satisfiable. This completes the reduction and proves the theorem. □

This result also naturally holds for the straightforward generalization of maximizing modularity in weighted graphs [17]. Instead of using the numbers of edges, the definition of modularity employs the sum of edge weights for edges within clusters, between clusters, and in the total graph.

4.1 Special Case: Modularity with a Bounded Number of Clusters

A common clustering approach is based on iteratively identifying cuts with respect to some quality measures (for example, see [18], [19], [20]). The general problem being $\mathcal{NP}$-complete, we now complete our hardness results by proving that the restricted optimization problem is hard as well. More precisely, we consider the two problems of computing the clustering with maximum modularity that splits the graph into exactly or at most two clusters. Although these are two different problems, our hardness result will hold for both versions; hence, we define the problem cumulatively.

Problem 3 (k-MODULARITY). *Given a graph G and a number K, is there a clustering $\mathcal{C}$ of G into exactly/at most k clusters, for which $\mathrm{q}(\mathcal{C}) \geq K$?*

We provide a proof by using a reduction that is similar to the one given recently for showing the hardness of the MinDisAgree[2] problem of correlation clustering [21]. We use the problem MINIMUM BISECTION FOR CUBIC GRAPHS (MB3) for the reduction:

Problem 4 (MB3). *Given a 3-regular graph G with n nodes and an integer c, is there a clustering into two clusters of $n/2$ nodes each such that it cuts at most c edges?*

This problem has been shown to be strongly NP-complete [22]. We construct an instance of 2-MODULARITY from an instance of MB3 as follows: For each node v from the graph $G = (V, E)$, we attach $n-1$ new nodes and construct an n-clique. We denote these cliques as $cliq(v)$ and refer to them as *node cliques* for $v \in V$. Hence, in total, we construct n different new cliques, and after this transformation, each node from the original graph has degree $n+2$. Note that a cubic graph with n nodes has exactly $1.5n$ edges. In our adjusted graph, there are exactly $m = (n(n-1)+3)n/2$ edges.

We will show that an optimum clustering, which is denoted as $\mathcal{C}^*$, of 2-MODULARITY in the adjusted graph has exactly two clusters. Furthermore, such a clustering corresponds to a minimum bisection of the underlying MB3 instance. In particular, we give a bound K such that the MB3 instance has a bisection cut of size at most c if and only if the corresponding graph has 2-modularity of at least K.

We begin by noting that there is always a clustering $\mathcal{C}$ with $\mathrm{q}(\mathcal{C}) > 0$. Hence, $\mathcal{C}^*$ must have exactly two clusters, as not more than two clusters are allowed. This shows that our proof works for both versions of 2-modularity, in which at most or exactly two clusters must be found.

Lemma 4.5. *For every graph constructed from a MB3 instance, there exists a clustering $\mathcal{C} = \{C_1, C_2\}$ such that $\mathrm{q}(\mathcal{C}) > 0$. In particular, the clustering $\mathcal{C}^*$ has two clusters.*

Proof. Consider the following partition into two clusters. We pick the nodes of $cliq(v)$ for some $v \in V$ as C_1 and the remaining graph as C_2. Then,

$$\begin{aligned}\mathrm{q}(\mathcal{C}) &= 1 - \frac{3}{m} \\ &\quad - \frac{(n(n-1)+3)^2 + ((n-1)(n(n-1)+3))^2}{4m^2} \\ &= \frac{2n-2}{n^2} - \frac{3}{m} = \frac{2}{n} - \frac{2}{n^2} - \frac{3}{m} \\ &> 0,\end{aligned}$$

as $n \geq 4$ for every cubic graph. Hence, $\mathrm{q}(\mathcal{C}) > 0$, and the lemma follows. □

Next, we show that in an optimum clustering, all the nodes of one node clique $cliq(v)$ are located in one cluster.

Lemma 4.6. *For every node $v \in V$, there exists a cluster $C \in \mathcal{C}^*$ such that $cliq(v) \subseteq C$.*

Proof. For contradiction, we assume that a node clique $cliq(v)$ for some $v \in V$ is split in two clusters C_1 and C_2 of the clustering $\mathcal{C} = \{C_1, C_2\}$. Let $k_i := |C_i \cap cliq(v)|$ be the number of nodes located in the corresponding clusters, with $1 \leq k_i \leq n-1$. Note that $k_2 = n - k_1$. In addition, we denote the sum of node degrees in both clusters, excluding nodes from $cliq(v)$, by d_1 and d_2:

$$d_i = \sum_{u \in C_i, u \notin cliq(v)} \deg(u).$$

Without loss of generality, assume that $d_1 \geq d_2$. Finally, we denote by m' the number of edges covered by the clusters C_1 and C_2.

We define a new clustering $\mathcal{C}'$ as $\{C_1 \setminus cliq(v), C_2 \cup cliq(v)\}$ and denote the difference of the modularity as $\Delta := \mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C})$. We distinguish two cases, depending in which cluster the node v was located with respect to $\mathcal{C}$. In the first case, $v \in C_2$, and we obtain

$$\begin{aligned}
\mathrm{q}(\mathcal{C}) &= \frac{m'}{m} - \frac{(d_1 + k_1(n-1))^2}{4m^2} \\
&\quad + \frac{(d_2 + (n-k_1)(n-1) + 3)^2}{4m^2}, \\
\mathrm{q}(\mathcal{C}') &= \frac{m' + k_1(n-k_1)}{m} \\
&\quad - \frac{d_1^2 + (d_2 + n(n-1) + 3)^2}{4m^2}, \qquad \text{and} \\
\Delta &= \frac{k_1(n-k_1)}{m} - \frac{d_1^2 + (d_2 + n(n-1) + 3)^2}{4m^2} \\
&\quad + \frac{(d_1 + k_1(n-1))^2}{4m^2} \\
&\quad + \frac{(d_2 + (n-k_1)(n-1) + 3)^2}{4m^2}.
\end{aligned}$$

We simplify the expression of Δ as follows:

$$\begin{aligned}
\Delta &= \frac{1}{4m^2}\Big(4mk_1(n-k_1) - d_1^2 - (d_2 + n(n-1) + 3)^2 \\
&\qquad + (d_1 + k_1(n-1))^2 \\
&\qquad + (d_2 + (n-k_1)(n-1) + 3)^2\Big) \\
&= \frac{1}{4m^2}\Big(4mk_1(n-k_1) + (2k_1^2 - 2nk_1)(n-1)^2 \\
&\qquad - 6k_1(n-1) + 2(d_1 - d_2)k_1(n-1)\Big) \\
&\geq \frac{k_1}{4m^2}\Big(4m(n-k_1) \\
&\qquad - 2(n-k_1)(n-1)^2 - 6(n-1)\Big).
\end{aligned}$$

We can bound the expression in the bracket in the following way by using the assumption that $d_1 \geq d_2$ and $1 \leq k_1 \leq n-1$:

$$\begin{aligned}
&(n-k_1)\Big(4m - 2(n-1)^2\Big) - 6(n-1) \\
&\geq (n-k_1)\Big(\underbrace{4m - 2(n-1)^2 - 6(n-1)}_{=:B}\Big).
\end{aligned} \tag{3}$$

Thus, it remains to show that $B > 0$. By filling in the value of m and using the facts that $2n^2(n-1) > 2(n-1)^2$ and $6n > 6(n-1)$ for all $n \geq 4$, we obtain $B > 0$ and, thus, the modularity strictly improves if all nodes are moved from $cliq(v)$ to C_2.

In the second case, the node $v \in C_1$, and we get the following:

$$\begin{aligned}
\mathrm{q}(\mathcal{C}) &= \frac{m'}{m} - \frac{(d_1 + k_1(n-1) + 3)^2}{4m^2} \\
&\quad + \frac{(d_2 + (n-k_1)(n-1))^2}{4m^2}, \\
\mathrm{q}(\mathcal{C}') &= \frac{m' + k_1(n-k_1)}{m} \\
&\quad - \frac{d_1^2 + (d_2 + n(n-1) + 3)^2}{4m^2}, \quad \text{and} \\
\Delta &= \frac{k_1(n-k_1)}{m} - \frac{d_1^2 + (d_2 + n(n-1) + 3)^2}{4m^2} \\
&\quad + \frac{(d_1 + k_1(n-1) + 3)^2}{4m^2} \\
&\quad + \frac{(d_2 + (n-k_1)(n-1))^2}{4m^2}.
\end{aligned}$$

We simplify the expression of Δ as follows:

$$\begin{aligned}
4m^2\Delta &= 4mk_1(n-k_1) + (2k_1^2 - 2nk_1)(n-1)^2 \\
&\quad - 6(n-k_1)(n-1) \\
&\quad + 2(d_1 - d_2)(k_1(n-1) + 3) \\
&\geq\ 4mk_1(n-k_1) - 2k_1(n-k_1)(n-1)^2 \\
&\quad - 6(n-k_1)(n-1)).
\end{aligned}$$

Recalling that $1 \leq k_1 \leq n-1$ and filling in the value of m, we obtain

$$\begin{aligned}
&4mk_1 - 2k_1(n-1)^2 - 6(n-1) \\
&= 2k_1(n^2(n-1) - (n-1)^2) + 6nk_1 - 6(n-1) > 0,
\end{aligned}$$

which holds for all $k_1 \geq 1$ and $n \geq 4$. In addition, in this case, the modularity strictly improves if all nodes are moved from $cliq(v)$ to C_2. □

The final lemma before defining the appropriate input parameter K for the 2-MODULARITY and thus proving the correspondence between the two problems shows that the clusters in the optimum clusterings have the same size.

Lemma 4.7. *In $\mathcal{C}^*$, each cluster contains exactly $n/2$ complete node cliques.*

Proof. Suppose, for contradiction, that one cluster C_1 has $l_1 < n/2$ cliques. For completeness of presentation, we use m' to denote the unknown (and irrelevant) number of edges covered by the clusters. The modularity of the clustering is given as follows:

$$\begin{aligned}
\mathrm{q}(\mathcal{C}^*) &= \frac{m'}{m} - \frac{l_1^2(n(n-1) + 3)^2}{4m^2} \\
&\quad - \frac{(n-l_1)^2(n(n-1) + 3)^2}{4m^2}.
\end{aligned} \tag{4}$$

We create a new clustering $\mathcal{C}'$ by transferring a complete node clique from cluster C_2 to cluster C_1. As the graph G is 3-regular, we lose at most three edges in the coverage part of modularity:

$$\begin{aligned}
\mathrm{q}(\mathcal{C}') &\geq \frac{m' - 3}{m} - \frac{(l_1 + 1)^2(n(n-1) + 3)^2}{4m^2} \\
&\quad + \frac{(n - l_1 - 1)^2(n(n-1) + 3)^2}{4m^2}.
\end{aligned} \tag{5}$$

We can bound the difference as follows:

$$\begin{aligned}
\mathrm{q}(\mathcal{C}') - \mathrm{q}(\mathcal{C}) \geq & -\frac{3}{m} + \frac{(l_1^2 + (n-l_1)^2}{4m^2} \\
& - \frac{(n-l_1-1)^2)(n(n-1)+3)^2}{4m^2} \\
&= -\frac{3}{m} + \frac{(2n-4l_1-2)}{n^2} \\
\geq & -\frac{3}{m} + \frac{2}{n^2} = \frac{2}{n^2} - \frac{6}{n^3 - n^2 + 3n} \\
& > 0,
\end{aligned}$$

for all $n \geq 4$. The analysis uses the fact that we can assume n to be an even number, so $l_1 \leq \frac{n}{2} - 1$ and, thus, $4l_1 \leq 2n - 4$.

This shows that we can improve every clustering by balancing the number of complete node cliques in the clusters, independent of the loss in edge coverage. □

Finally, we can state theorem about the complexity of 2-MODULARITY.

Theorem 4.8. *2-MODULARITY is strongly $\mathcal{NP}$-complete.*

Proof. Letting (G, c) be an instance of MB3, then we construct a new graph G' as stated above and define $K := 1/2 - c/m$.

As we have shown in Lemma 4.7 that each cluster of $\mathcal{C}^*$ that is an optimum clustering of G' with respect to 2-MODULARITY has exactly $n/2$ complete node cliques, the sum of degrees in the clusters is exactly m. Thus, supposing the clustering $\mathcal{C}^*$ meets the following inequality:

$$\mathrm{q}(\mathcal{C})^* \geq 1 - \frac{c}{m} - \frac{2m^2}{4m^2} = \frac{1}{2} - \frac{c}{m} = K,$$

then it is easy to see that the number of intercluster edges can be at most c. Thus, the clustering $\mathcal{C}^*$ induces a balanced cut in G with at most c cut edges. □

This proof is particularly interesting, as it highlights that maximizing modularity, in general, is hard due to the hardness of minimizing the squared degree sums on one hand, whereas in the case of two clusters, this is due to the hardness of minimizing the edge cut.

5 THE GREEDY ALGORITHM

In contrast to the above-mentioned iterative cutting strategy, another commonly used approach to find clusterings with good-quality scores is based on greedy agglomeration [14], [23]. In the case of modularity, this approach is particularly widespread [7], [8].

The greedy algorithm starts with the singleton clustering and iteratively merges those two clusters that yield a clustering with the best modularity; that is, the largest increase or the smallest decrease is chosen. After $n-1$ merges, the clustering that achieved the highest modularity is returned. The algorithm maintains a symmetric matrix Δ with entries $\Delta_{i,j} := \mathrm{q}(\mathcal{C}_{i,j}) - \mathrm{q}(\mathcal{C})$, where $\mathcal{C}$ is the current clustering, and $\mathcal{C}_{i,j}$ is obtained from $\mathcal{C}$ by merging clusters C_i and C_j. Note that there can be several pairs i and j such that $\Delta_{i,j}$ is the maximum. In these cases, the algorithm selects an arbitrary pair. The pseudocode for the greedy algorithm is given in Algorithm 1. An efficient implementation that uses sophisticated data structures requires $\mathcal{O}(n^2 \log n)$ runtime. Note that $n-1$ iterations is an upper bound, and one can terminate the algorithms when the matrix Δ contains only nonpositive entries. We call this property *single peakedness*, and it is proven in [8]. Since it is $\mathcal{NP}$-hard to maximize modularity in general graphs, it is unlikely that this greedy algorithm is optimal. In fact, we sketch a graph family where the above greedy algorithm has an approximation factor of 2 asymptotically. In order to prove this statement, we introduce a general construction scheme, as given in Definition 5.2. Furthermore, we point out instances where a specific way of breaking ties of merges yield a clustering with modularity of 0, whereas the optimum clustering has a strictly positive score.

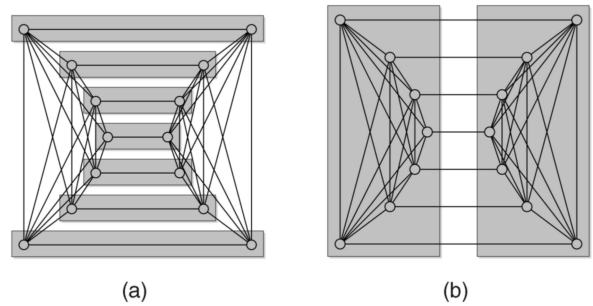

Fig. 3. (a) Clustering with modularity 0. (b) Clustering with modularity close to $\frac{1}{2}$.

Algorithm 1: GREEDY ALGORITHM FOR MAXIMIZING MODULARITY

```
Input: graph G = (V, E)
Output: clustering C of G
C ← singletons
initialize matrix Δ
while |C| > 1 do
    find {i, j} with Δ_{i,j} is the maximum entry in Δ
    merge clusters i and j
    update Δ
return clustering with highest modularity
```

Modularity is defined such that it takes values in the interval $[-1/2, 1]$ for any graph and any clustering. In particular, the modularity of a trivial clustering that places all vertices into a single cluster has a value of 0. We use this technical peculiarity to show that the greedy algorithm has an unbounded approximation ratio.

Theorem 5.1. *There is no finite approximation factor for the greedy algorithm for finding clusterings with maximum modularity.*

Proof. We present a class of graphs, on which the algorithm obtains a clustering of value 0 but for which the optimum clustering has a value close to $1/2$. A graph G of this class is given by two cliques (V_1, E_1) and (V_2, E_2) of size $|V_1| = |V_2| = n/2$, and $n/2$ matching edges E_m that connect each vertex from V_1 to exactly one vertex in V_2, and vice versa. See Fig. 3 for an example, with $n = 14$. Note that we can define modularity by associating weights $w(u,v)$ with every existing and nonexisting edge in G as follows:

$$w(u,v) = \frac{E_{uv}}{2m} - \frac{\deg(u)\deg(v)}{4m^2},$$

where $E_{uv} = 1$ if $(u,v) \in E$; otherwise, this is 0. The modularity of a clustering $\mathcal{C}$ is then derived by summing the weights of the edges covered by $\mathcal{C}$:

$$\mathrm{q}(\mathcal{C}) = \sum_{C \in \mathcal{C}} \sum_{u,v \in C} w(u,v).$$

Note that in this formula, we have to count twice the weight for each edge between different vertices u and v (once for every ordering) and once the weight for a nonexisting self loop for every vertex u. Thus, the change of modularity by merging two clusters is given by twice the sum of weights between the clusters.

Now, consider a run of the greedy algorithm on the graph in Fig. 3. Note that the graph is $n/2$-regular and thus has $m = n^2/4$ edges. Each existing edge gets a weight of $2/n^2 - 1/n^2 = 1/n^2$, whereas every nonexisting edge receives a weight of $-1/n^2$. As the self loop is counted by every clustering, the initial trivial singleton clustering has a modularity value of $-1/n$. In the first step, each cluster merge along any existing edge results in an increase of $2/n^2$. Of all these equivalent possibilities, we suppose that the algorithm chooses to merge along an edge from E_m to create a cluster C'. In the second step, merging a vertex with C' results in change of 0, because one existing and one nonexisting edge would be included. Every other merge along an existing edge still has a value of $2/n^2$. We suppose that the algorithm again chooses to merge two singleton clusters along an edge from E_m, creating a cluster C''. After that, observe that merging clusters C' and C'' yields a change of 0, because two existing and two nonexisting edges would be included. Thus, it is again optimal to merge two singleton clusters along an existing edge. If the algorithm continues to merge singleton clusters along the edges from E_m, it will, in each iteration, make an optimal merge that results in strictly positive increase in modularity. After $n/2$ steps, it has constructed a clustering $\mathcal{C}$ of the type depicted in Fig. 3a. $\mathcal{C}$ consists of one cluster for the vertices of each edge of E_m and has a modularity value of

$$\mathrm{q}(\mathcal{C}) = \frac{2}{n} - \frac{n}{2} \cdot \frac{4n^2}{n^4} = 0.$$

Due to the single peakedness of the problem [8], all following cluster merges can never increase this value; hence, the algorithm will return a clustering of value 0.

On the other hand, consider a clustering $\mathcal{C}^* = \{C_1, C_2\}$ with two clusters, one for each clique: $C_1 = V_1$ and $C_2 = V_2$ (see Fig. 3b). This clustering has a modularity of

$$\mathrm{q}(\mathcal{C}^*) = \frac{n(n-2)}{n^2} - 2\frac{4n^2}{16n^2} = \frac{1}{2} - \frac{2}{n}.$$

This shows that the approximation ratio of the greedy algorithm can infinitely be large, because no finite approximation factor can outweigh a value of 0, with one strictly greater than 0. □

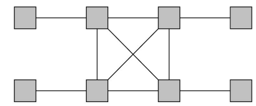

Fig. 4. The graph $K_4 \star_u P_1$.

The key observation is that the proof considers a worst-case scenario in the sense that greedy is, in each iteration, supposed to pick exactly the "worst" merge choice of several equivalently attractive alternatives. If greedy chooses, in an early iteration, to merge along an edge from E_1 or E_2, the resulting clustering will be significantly better. As mentioned earlier, this negative result is due to the formulation of modularity, which yields values from the interval $[-1/2, 1]$. For instance, with a linear remapping of the range of modularity to the interval [0, 1], the greedy algorithm yields a value of 1/3 compared to the new optimum score of 2/3. In this case, the approximation factor would be 2.

Next, we provide a decreased lower bound for a different class of graphs and no assumptions on the random choices of the algorithm.

Definition 5.2. *Let $G = (V,E)$ and $H = (V',E')$ be two nonempty simple undirected unweighted graphs and let $u \in V'$ be a node. The* product $G_{\star_u}H$ *is defined as the graph (V'',E'') with the node set $V'' := V \cup V \times V'$ and the edge set $E'' := E \cup E''_c \cup E''_H$, where*

$$E''_c := \big\{\{v,(v,u)\} \mid v \in V\big\}, \qquad \text{and}$$
$$E''_H := \big\{\{(v,v'),(v,w')\} \mid v \in V, v', w' \in V'', \{v',w'\} \in E\big\}.$$

An example is given in Fig. 4. The product $G \star_u H$ is a graph that contains G and, for each node v of G, a copy H_v of H. For each copy, the node in H_v that corresponds to $u \in H$ is connected to v. We use the notation (v,w') to refer to the copy of node w' of H, which is located in H_v. In the following, we consider only a special case. Let $n \geq 2$ be an integer, $H = (V',E')$ be an undirected and connected graph with at least two nodes, and $u \in V'$ be an arbitrary but fixed node. We denote by $\mathcal{C}^g_k$ the clustering obtained with the greedy algorithm applied to $K_n \star_u H$ that starts from singletons and performs at most k steps, which all have a positive increase in modularity. Furthermore, let m be the number of edges in $K_n \star_u H$. Based on the merging policy of the greedy algorithm, we can characterize the final clustering $\mathcal{C}^g_n$. It has n clusters, each of which includes a vertex v of G and the copy of H.

Theorem 5.3. *Let $n \geq 2$ be an integer and let $H = (V',E')$ be a undirected and connected graph with at least two nodes. If $2|E'| + 1 < n$, then the greedy algorithm returns the clustering $\mathcal{C}^g := \{\{v\} \cup \{v\} \times V' | v \in V\}$ for $K_n \star_u H$ (for any fixed $u \in H$). This clustering has a modularity score of*

$$4m^2 \cdot \mathrm{q}(\mathcal{C}^g) = 4m((|E'|+1) \cdot n) - n(2|E'|+1+n)^2.$$

The proof of Theorem 5.3, which relies on the graph construction described above, is available from the authors or can alternatively be found in an associated technical report [24]. The next corollary reveals that the clustering, in which G and each copy of H form individual clusters, has a greater modularity score. We first observe an explicit expression for modularity.

Corollary 5.4. *The clustering $\mathcal{C}^s$ is defined as*

$$\mathcal{C}^s := \{V\} \cup \{\{v\} \times V' | v \in V\},$$

and according to (2), its modularity is

$$4m^2 \cdot \mathrm{q}(\mathcal{C}^s) = 4m\Big(|E'|n + \binom{n}{2}\Big) - n(2|E'|+1)^2 - (n \cdot (n-1+1))^2.$$

If $n \geq 2$ and $2|E'|+1 < n$, then clustering $\mathcal{C}^s$ has higher modularity than $\mathcal{C}^g$.

Theorem 5.5. *The approximation factor of the greedy algorithm for finding clusterings with maximum modularity is at least 2.*

The quotient $\mathrm{q}(\mathcal{C}^s)/\mathrm{q}(\mathcal{C}^g)$ asymptotically approaches 2 for n going to infinity on $K_n \star_u H$, with H being a path of length $1/2\sqrt{n}$. The full proof of Theorem 5.5 is also available in [24].

6 Optimality Results

6.1 Characterization of Cliques and Cycles

In this section, we provide several results on the structure of clusterings with maximum modularity for cliques and cycles. This extends previous work, in particular [2], in which cycles and cycles of cliques were used to reason about global properties of modularity.

A first observation is that modularity can be simplified for general d-regular graphs as follows:

Corollary 6.1. *Let $G = (V, E)$ be an unweighted d-regular graph and let $\mathcal{C} = \{C_1, \dots, C_k\} \in \mathcal{A}(G)$. Then, the following equality holds:*

$$\mathrm{q}(\mathcal{C}) = \frac{|E(\mathcal{C})|}{dn/2} - \frac{1}{n^2}\sum_{i=1}^{k}|C_i|^2. \tag{6}$$

The correctness of the corollary can be read from the definition given in (2) and the fact that $|E| = d|V|/2$. Thus, for regular graphs, modularity only depends on cluster sizes and coverage.

6.1.1 Cliques

We first deal with the case of complete graphs. Corollary 6.2 provides a simplified formulation for modularity. From this rewriting, the clustering with maximum modularity can be directly obtained.

Corollary 6.2. *Let K_n be a complete graph on n nodes and let $\mathcal{C} := \{C_1, \dots, C_k\} \in \mathcal{A}(K_n)$. Then, the following equality holds:*

$$\mathrm{q}(\mathcal{C}) = -\frac{1}{n-1} + \frac{1}{n^2(n-1)}\sum_{i=1}^{k}|c_i|^2. \tag{7}$$

The simple proof of the corollary can be found in the Appendix. Thus, maximizing the modularity is equivalent to maximizing the squares of cluster sizes. Using the general inequality $(a+b)^2 \geq a^2 + b^2$ for nonnegative real numbers, the clustering with maximum modularity is the 1-clustering.

Theorem 6.3. *Let k and n be integers, K_{kn} be the complete graph on $k \cdot n$ nodes, and $\mathcal{C}$ be a clustering such that each cluster contains exactly n elements. Then, the following equality holds:*

$$\mathrm{q}(\mathcal{C}) = \left(-1 + \frac{1}{k}\right) \cdot \frac{1}{kn-1}.$$

For fixed $k > 1$ and as n tends to infinity, the modularity is always strictly negative but tends to zero. Only for $k = 1$ is the modularity zero and so is the global maximum.

As Theorem 6.3 deals with one clique, the following corollary provides the optimal result for k disjoint cliques.

Corollary 6.4. *The maximum modularity of a graph that consists of k disjoint cliques of size n is $1 - 1/k$.*

The corollary follows from the definition of modularity in (2). Corollary 6.4 gives a glimpse on how previous approaches have succeeded to upper bound modularity as it was pointed out in the context of Lemma 3.1.

6.1.2 Cycles

Next, we focus on simple cycles, that is, connected 2-regular graphs. According to (6), modularity can be expressed as given in (8) if each cluster is connected, which may safely be assumed (see Corollary 3.5):

$$\mathrm{q}(\mathcal{C}) = \frac{n-k}{n} - \frac{1}{n^2}\sum_{i=1}^{k}|C_i|^2. \tag{8}$$

In the following, we prove that clusterings with maximum modularity are balanced with respect to the number and the sizes of clusters. First, we characterize the distribution of cluster sizes for clusterings with maximum modularity by fixing the number k of clusters. For convenience, we minimize $F := 1 - \mathrm{q}(\mathcal{C})$, where the argument of F is the distribution of the cluster sizes.

Proposition 6.5. *Let k and n be integers, the set*

$$D^{(k)} := \left\{x \in \mathbb{N}^k \Big| \sum_{i=1}^{k} x_i = n\right\},$$

and the function $F : D^{(k)} \to \mathbb{R}$ defined as

$$F(x) := \frac{k}{n} + \frac{1}{n^2}\sum_{i=1}^{k} x_i^2 \qquad \text{for } x \in D^{(k)}.$$

Then, F has a global minimum at x^*, with $x_i^* = \lfloor \frac{n}{k} \rfloor$ for $i = 1, \dots, k-r$ and $x_i^* = \lceil \frac{n}{k} \rceil$ for $i = k-r+1, \dots, k$, where $0 \leq r < k$, and $r \equiv n \bmod k$.

Proposition 6.5 is based on the fact that, roughly speaking, evening out cluster sizes decreases F. We refer the reader to the Appendix for the full proof. Due to the special structure of simple cycles, we can swap neighboring clusters without changing the modularity. Thus, we can safely assume that clusters are sorted according to their sizes, starting with the smallest element. Then, x^* is the only optimum. Evaluating F at x^* leads to a term that only depends on k and n. Hence, we can characterize the

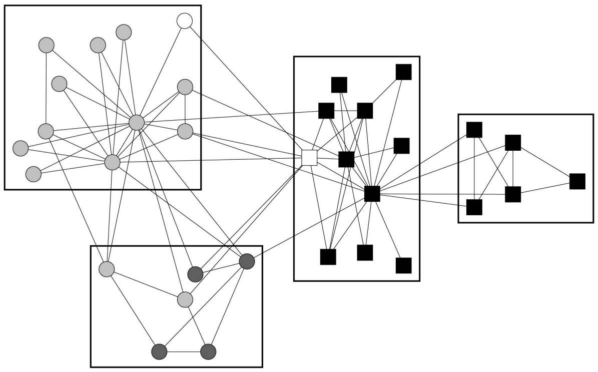

Fig. 5. Karate club network of Zachary [25]. The different clusterings are coded as follows: Blocks represent the optimum clustering (with respect to modularity), colors correspond to the greedy clustering, and shapes code the split that occurred in reality.

clusterings with maximum modularity only with respect to the number of clusters. The function to be minimized is given as follows:

Lemma 6.6. *Let C_n be a simple cycle with n nodes, $h : [1, \ldots, n] \to \mathbb{R}$ a function defined as*

$$h(x) := x \cdot n + n + \left\lfloor \frac{n}{x} \right\rfloor \left(2n - x \cdot \left(1 + \left\lfloor \frac{n}{x} \right\rfloor \right) \right),$$

and k^* be the argument of the global minimum of h. Then, every clustering of C_n with maximum modularity has k^* clusters.

The proof of Lemma 6.6 builds upon Proposition 6.5, and it can be found in the Appendix. Finally, we obtain the characterization for clusterings with maximum modularity for simple cycles.

Theorem 6.7. *Let n be an integer and C_n be a simple cycle with n nodes. Then, every clustering $\mathcal{C}$ with maximum modularity has k cluster of almost equal size, where*

$$k \in \left[\frac{n}{\sqrt{n + \sqrt{n}}} - 1, \frac{1}{2} + \sqrt{\frac{1}{4} + n} \right].$$

Furthermore, there are only three possible values for k for a sufficiently large n.

The rather technical proof of Theorem 6.7 is based on the monotonicity of h. This proof can also be found in the Appendix.

7 Examples Revisited

Applying our results about maximizing the modularity gained so far, we revisit three example networks that were used in related work [25], [26], [9]. More precisely, we compare published greedy solutions with respective optima, thus revealing two peculiarities of modularity. First, we illustrate a behavioral pattern of the greedy merge strategy, and second, we put the quality of the greedy approach into perspective.

The first instance is the karate club network of Zachary that was originally introduced in [25] and used for demonstration in [26]. The network models social interactions between members of a karate club. More precisely, friendship between the members is presented before the club split up due to an internal dispute. A representation of the network is given in Fig. 5. The partition that has resulted from the split is given by the shape of the nodes, whereas the colors indicate the clustering calculated by the greedy algorithm, and blocks refer to an optimum clustering maximizing modularity that has been obtained by solving its associated ILP. The corresponding scores of modularity are 0.431 for the optimum clustering, 0.397 for the greedy clustering, and 0.383 for the clustering given by the split. Even though this is another example in which the greedy algorithm does not optimally perform, its score is comparatively good. Furthermore, the example shows one of the potential pitfalls that the greedy algorithm can encounter: Due to the attempt of balancing the squared sum of degrees (over the clusters), a node with large degree (white square) and one with small degree (white circle) are merged at an early stage. However, using the same argument, such a cluster will unlikely be merged with another one. Thus, small clusters with skewed degree distributions occur.

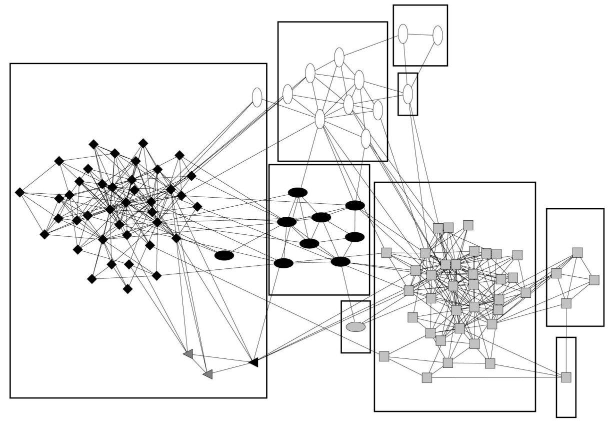

Fig. 6. The networks of books on politics compiled by V. Krebs. The different clusterings are coded as follows: Blocks represent the clustering calculated with GMC, colors correspond to the greedy clustering, and shapes code the optimum clustering (with respect to modularity).

The second instance is a network of books on politics, which is compiled by V. Krebs and used for demonstration in [9]. The nodes represent books on American politics bought from Amazon.com, and edges join pairs of books that are frequently purchased together. A representation of the network is given in Fig. 6. The optimum clustering maximizing modularity is given by the shapes of nodes, the colors of nodes indicate a clustering calculated by the greedy algorithm, and the blocks show a clustering calculated by Geometric MST Clustering (GMC), which is introduced in [27], using the geometric mean of coverage and performance, both of which are quality indices discussed in the same paper. The corresponding scores of modularity are 0.527 for the optimum clustering, 0.502 for the greedy clustering, and 0.510 for the GMC clustering. Similar to the first example, the greedy algorithm is suboptimal but is relatively close to the optimum. Interestingly, GMC outperforms the greedy algorithm, although it does not consider modularity in its calculations. This illustrates the fact that there probably are many *intuitive* clusterings close to the optimum clustering that all have

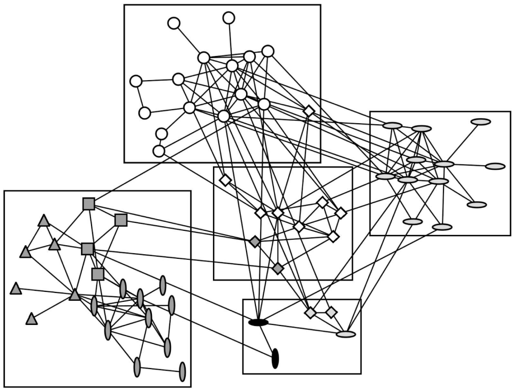

Fig. 7. Social network of bottlenose dolphins introduced in [28] and clustered in [29]. The different clusterings are coded as follows: Blocks represent the clustering with maximum modularity, colors represent the result of the greedy clustering, and shapes code the community structure identified with the *iterative conductance cut* algorithm presented in [20].

relatively similar values of modularity. In analogy to the first example, we observe the same merge artifact, namely, the two nodes represented as dark-gray triangles.

As the last example, Fig. 7 reflects the social structure of a family of bottlenose dolphins off the coast of New Zealand, as observed by Lusseau et al. [28], who logged frequent associations between dolphins over a period of 7 years. The clustering with optimum modularity (blocks) achieves a modularity score of 0.529 and, again, the greedy algorithm (colors) approaches this value with 0.496. However, structurally, the two clusterings disagree on the two small clusters, whereas a clustering based on *iterative conductance cutting* [20] (shapes) achieves the same quality (0.492) but disagrees with the optimum only on the smallest cluster and on the refinement of the leftmost cluster.

In summary, the three examples illustrated several interesting facts. First, an artificial pattern in the optimization process of the greedy algorithm is revealed: The early merge of two nodes, one with a high degree and one with a low degree, results in a cluster that will not be merged with another one later on. In general, this can prevent finding the optimum clustering. Nevertheless, it performs relatively well on the given instances and is at most 10 percent of the optimum. However, applying other algorithms that do not optimize modularity, we observe that the obtained clusterings have similar scores. Thus, achieving good scores of modularity does not seem to be too hard on these instances. On one hand, these clusterings roughly agree in terms of the overall structure; on the other hand, they differ in the numbers of clusters and even feature artifacts such as small clusters of size one or two. Considering that all three examples exhibit significant community structure, we thus predict that there are many intuitive clusterings being structurally close (with respect to lattice structure) and that most suitable clustering algorithms probably identify one of them.

8 Conclusion

This paper represents the first formal assessment to optimization of a popular clustering index known as modularity. We have settled the open question about the complexity status of modularity maximization by proving its $\mathcal{NP}$-hardness, in particular by proving $\mathcal{NP}$-completeness in the strong sense for the underlying decision problem. We show that this even holds for the restricted version, with a bound of two on the number of clusters. This justifies the further investigation of approximation algorithms and heuristics such as the widespread greedy approach. For the latter, we prove a first lower bound on the approximation factor. Our analysis of the greedy algorithm also includes a brief comparison with the optimum clustering, which is calculated via ILP on some real-world instances, thus encouraging a reconsideration of previous results. The following is a list of the main results derived from this paper:

- Modularity can be defined as a normalized trade-off between edges covered by clusters and squared cluster degree sums (see (1)).
- There is a formulation of modularity maximization as ILP (Section 2.2).
- There is a clustering with maximum modularity without singleton clusters of degree 1 and without clusters representing disconnected subgraphs. Isolated nodes have no impact on modularity (Corollary 3.2 and Lemmas 3.3 and 3.4).
- The clustering of maximum modularity changes in a global nontrivial fashion, even for the simplest graph perturbation (Section 3.1).
- For any clustering $\mathcal{C}$ of any graph G, the modularity value $\frac{1}{2} \leq \mathrm{q}(\mathcal{C}) < 1$ (Lemma 3.1).
- Finding a clustering with maximum modularity is $\mathcal{NP}$-hard, both for the general case and when restricted to clusterings with exactly or at most two clusters (Theorems 4.4 and 4.8).
- With the worst tie-breaking strategy, the greedy agglomeration algorithm has no worst-case approximation factor (Theorem 5.1). With an arbitrary tie-breaking strategy, the worst-case factor is at least 2 (Theorem 5.5).
- A clustering of maximum modularity for cliques of size n consists of a single cluster (Theorem 6.3), and for cycles of size n, this consists of approximately $\sqrt{n}$ clusters of size $\sqrt{n}$ each (Theorem 6.7).

For the future, we plan an extended analysis and the development of a clustering algorithm with provable performance guarantees. The special properties of the measure, its popularity in application domains, and the absence of fundamental theoretical insights hitherto render a further mathematically rigorous treatment of modularity necessary.

Appendix

Proof [of Corollary 6.2]. The coverage of $\mathcal{C}$ can be expressed in terms of cluster sizes as follows:

$$\begin{aligned}|E(\mathcal{C})| &= \binom{n}{2} - \sum_{i=1}^{k}\prod_{j>i}|C_i|\cdot|C_j| \\ &= \binom{n}{2} - \frac{1}{2}\sum_{i=1}^{k}\prod_{j\neq i}|C_i|\cdot|C_j| \\ &= \binom{n}{2} - \frac{1}{2}\sum_{i=1}^{k}|C_i|\cdot\sum_{j\neq i}|C_j| \\ &= \binom{n}{2} - \frac{1}{2}\sum_{i=1}^{k}|C_i|\cdot(n-|C_i|) \\ &= \binom{n}{2} - \frac{1}{2}\left(n^2 - \sum_{i=1}^{k}|C_i|^2\right) \\ &= -\frac{n}{2} + \frac{1}{2}\sum_{i=1}^{k}|C_i|^2.\end{aligned}$$

Thus, we obtain

$$\begin{aligned}\mathrm{q}(\mathcal{C}) &= -\frac{1}{n-1} + \frac{1}{n(n-1)}\sum_{i=1}^{k}|C_i|^2 - \frac{1}{n^2}\sum_{i=1}^{k}|C_i|^2 \\ &= -\frac{1}{n-1} + \frac{1}{n^2\cdot(n-1)}\sum_{i=1}^{k}|C_i|^2,\end{aligned}$$

which proves the equation. □

Proof [of Proposition 6.5]. Since k and n are given, minimizing F is equivalent to minimizing $\sum_i x_i^2$. Thus, let us rewrite this term:

$$\begin{aligned}\sum_{i=1}^{k}\Big(x_i - \frac{n}{k}\Big)^2 &= \sum_{i=1}^{k}x_i^2 - 2\frac{n}{k}\sum_{i=1}^{k}x_i + k\cdot\Big(\frac{n}{k}\Big)^2 \\ &= \sum_{i=1}^{k}x_i^2 - 2\frac{n^2}{k} + \frac{n^2}{k}, \\ \iff \qquad \sum_{i=1}^{k}x_i^2 &= \underbrace{\sum_{i=1}^{k}\Big(x_i - \frac{n}{k}\Big)^2}_{=:h(x)} + \frac{n^2}{k}.\end{aligned}$$

Minimizing F is equivalent to minimizing h. If r is 0, then $h(x^*) = 0$. For every other vector y, the function h is strictly positive, since at least one summand is positive. Thus, x^* is a global optimum.

Let $r > 0$. First, we show that every vector $x \in D^{(k)}$ that is close to $(\frac{n}{k}, \dots, \frac{n}{k})$ has (in principle) the form of x^*. Letting $x \in D \cap [\lfloor\frac{n}{k}\rfloor, \lceil\frac{n}{k}\rceil]^k$, then it is easy to verify that there are $k - r$ entries that have value $\lfloor\frac{n}{k}\rfloor$, and the remaining r entries have value $\lceil\frac{n}{k}\rceil$. Any "shift of one unit" between two variables having the same value increases the corresponding cost. Let $\epsilon := \lceil\frac{n}{k}\rceil - \frac{n}{k}$ and $x_i = x_j = \lceil\frac{n}{k}\rceil$. Replacing x_i with $\lfloor\frac{n}{k}\rfloor$ and x_j with $\lceil\frac{n}{k}\rceil + 1$ causes an increase in h by $5 + 2\epsilon > 0$. Similarly, the case of $x_i = x_j = \lfloor\frac{n}{k}\rfloor$ and the reassignment $x_i = \lceil\frac{n}{k}\rceil$ and $x_j = \lfloor\frac{n}{k}\rfloor - 1$ cause an increase of h by $2 > 0$.

Finally, we show that any vector of $D^{(k)}$ can be reached from x^* by "shifting one unit" between variables. Letting $x \in D^{(k)}$ and with loss of generality, we assume that $x_i \le x_{i+1}$ for all i. We define a sequence of elements in $D^{(k)}$ as follows:

1. $x^{(0)} := x^*$ and
2. if $x^{(i)} \neq x$, define $x^{(i+1)}$ as follows:

$$x_j^{(i+1)} := \begin{cases} x_j^{(i)} - 1 & \text{if } j = \min\{\ell | x_\ell^{(i)} > x_\ell\} =: L, \\ x_j^{(i)} + 1 & \text{if } j = \max\{\ell | x_\ell^{(i)} < x_\ell\} =: L', \\ x_j^{(i)} & \text{otherwise.} \end{cases}$$

Note that all obtained vectors $x^{(i)}$ are elements of $D^{(k)}$ and meet the condition of $x_j^{(i)} \le x_{j+1}^{(i)}$. Furthermore, we gain the following formula for the cost:

$$\sum_j \Big(x_j^{(i+1)}\Big)^2 = \sum_j \Big(x_j^{(i)}\Big)^2 + 2\Big(x_{L'}^{(i)} - x_L^{(i)} + 1\Big).$$

Since $L < L'$, one obtains $x_{L'}^{(i)} \ge x_L^{(i)}$. Thus, x^* is a global optimum in $D^{(k)}$. □

Proof [of Lemma 6.6]. Note that $h(k) = F(x^*)$, where F is the function of Proposition 6.5 with the given k. Consider first the following:

$$\begin{aligned}\sum_{i=1}^{k}(x_i^*)^2 &= (k-r)\cdot\Big\lfloor\frac{n}{k}\Big\rfloor^2 + r\cdot\Big\lceil\frac{n}{k}\Big\rceil^2 \\ &= (k-r)\frac{(n-r)^2}{k^2} + r\bigg(\frac{(n-r)}{k} + 1\bigg)^2 \\ &= \frac{n-r}{k}((n-r)+2r) + r = \frac{n^2-r^2}{k} + r \\ &= \frac{1}{k}\bigg(n^2 - \Big(n - \Big\lfloor\frac{n}{k}\Big\rfloor k\Big)^2\bigg) + n - \Big\lfloor\frac{n}{k}\Big\rfloor k \\ &= 2n\Big\lfloor\frac{n}{k}\Big\rfloor - k\Big\lfloor\frac{n}{k}\Big\rfloor^2 + n - \Big\lfloor\frac{n}{k}\Big\rfloor k \\ &= n + \Big\lfloor\frac{n}{k}\Big\rfloor\Big(2n - k\Big(\Big\lfloor\frac{n}{k}\Big\rfloor + 1\Big)\Big).\end{aligned}$$

Since maximizing the modularity is equivalent to minimizing the expression

$$k/n + 1/n^2\sum_i x_i^2$$

for $(x_i) \in \bigcup_{j=1}^{n} D^{(j)}$. Note that every vector (x_i) can be realized as clustering with connected clusters. Since we have characterized the global minima for fixed k, it is sufficient to find the global minima by varying k. □

Proof [of Theorem 6.7]. First, we show that the function h can be bounded by the inequalities given in (9) and is monotonically increasing (decreasing) for certain choices of k:

$$kn + \frac{n^2}{k} \le h(k) \le kn + \frac{n^2}{k} + \frac{k}{4}. \tag{9}$$

In order to verify the inequalities (9), let ϵ_k be defined as $n/k - \lfloor n/k\rfloor (\ge 0)$. Then, the definition of h can be rewritten as follows:

$$
\begin{aligned}
h(k) &= kn + n + \left\lfloor \frac{n}{k} \right\rfloor \left(2n - \left(1 + \left\lfloor \frac{n}{k} \right\rfloor\right)k\right) \\
&= kn + n + \left(\frac{n}{k} - \epsilon_k\right)\left(2n - \left(1 + \frac{n}{k} - \epsilon_k\right)k\right) \\
&= kn + n + \frac{2n^2}{k} - (1 - \epsilon_k)n \\
&\quad - \frac{n^2}{k} - 2n\epsilon_k + (1 - \epsilon_k)k\epsilon_k + n\epsilon_k \\
&= kn + \frac{n^2}{k} + (1 - \epsilon_k)\epsilon_k k.
\end{aligned}
$$

Replacing the term $(1-\epsilon_k)\epsilon_k k$ by a lower (upper) bound of 0 $(k/4)$ proves the given statements.

Second, the function h is monotonically increasing for $k \geq 1/2 + \sqrt{1/4 + n}$ and monotonically decreasing for $k \leq n/\sqrt{n + \sqrt{n}} - 1$. In order to prove the first part, it is sufficient to show that $h(k) \leq h(k+1)$ for every suitable k:

$$
\begin{aligned}
&h(k+1) - h(k) \\
&= (k+1)n + n + \left\lfloor \frac{n}{k+1} \right\rfloor \\
&\quad \left(2n - \left(1 + \left\lfloor \frac{n}{k+1} \right\rfloor\right)(k+1)\right) \\
&\quad - kn - n - \left\lfloor \frac{n}{k} \right\rfloor \left(2n - \left(1 + \left\lfloor \frac{n}{k} \right\rfloor\right)k\right) \\
&= n + 2n\left(\left\lfloor \frac{n}{k+1} \right\rfloor - \left\lfloor \frac{n}{k} \right\rfloor\right) - \left(1 + \left\lfloor \frac{n}{k+1} \right\rfloor\right)\left\lfloor \frac{n}{k+1} \right\rfloor \\
&\quad + k\left(\left(1 + \left\lfloor \frac{n}{k} \right\rfloor\right)\left\lfloor \frac{n}{k} \right\rfloor - \left(1 + \left\lfloor \frac{n}{k+1} \right\rfloor\right)\left\lfloor \frac{n}{k+1} \right\rfloor\right).
\end{aligned}
$$

Since $\lfloor \cdot \rfloor$ is discrete, and $|\lfloor x \rfloor - \lfloor x - 1 \rfloor| \leq 1$, one obtains

$$
\begin{aligned}
&h(k+1) - h(k) \\
&= \begin{cases} n - \lfloor \frac{n}{k} \rfloor^2 - \lfloor \frac{n}{k} \rfloor, & \text{if } \lfloor \frac{n}{k} \rfloor = \lfloor \frac{n}{k-1} \rfloor, \\ 3n - \lfloor \frac{n}{k} \rfloor^2 - \lfloor \frac{n}{k} \rfloor + 2k\lfloor \frac{n}{k} \rfloor, & \text{otherwise.} \end{cases}
\end{aligned} \tag{10}
$$

Since

$$
3n - \lfloor n/k \rfloor^2 - \lfloor n/k \rfloor + 2k\lfloor n/k \rfloor > n - \lfloor n/k \rfloor^2 - \lfloor n/k \rfloor,
$$

it is sufficient to show that $n - \lfloor n/k \rfloor^2 - \lfloor n/k \rfloor \geq 0$. This inequality is fulfilled if $n - (n/k)^2 - n/k \geq 0$. Solving the quadratic equations leads to $k \geq 1/2 + \sqrt{1/4 + n}$.

Using the above bound, for the second part, it is sufficient to show that

$$
kn + \frac{n^2}{k} - (k+1)n - \frac{n^2}{k+1} - \frac{k+1}{4} \geq 0, \tag{11}
$$

since this implies that the upper bound of $h(k+1)$ is smaller than (the lower bound of) $h(k)$. One can rewrite the left side of (11) as

$$
\begin{aligned}
&kn + \frac{n^2}{k} - (k+1)n - \frac{n^2}{k+1} - \frac{k+1}{4} = \\
&- n + \frac{n^2}{k(k+1)} - \frac{k+1}{4}.
\end{aligned}
$$

Since $h(k) - h(k+1)$ is monotonically decreasing for $0 \leq k \leq \sqrt{n}$, it is sufficient to show that $h(k) - h(k+1)$ is nonnegative for the maximum value of k. We show that the lower bound $h_-(k) := -n + n^2/(k+1)^2 - (k+1)/4$ is nonnegative:

$$
\begin{aligned}
h_-\left(\frac{n}{\sqrt{n + \sqrt{n}}} - 1\right) &= -n - \frac{n}{4\sqrt{n + \sqrt{n}}} \\
&\quad + \frac{n^2(n + \sqrt{n})}{n^2} \\
&= \sqrt{n} - \underbrace{\frac{n}{4\sqrt{n + \sqrt{n}}}}_{\leq \frac{1}{4}\sqrt{n}} \geq 0.
\end{aligned}
$$

In summary, the number of clusters k (of an optimum clustering) can only be contained in the given interval, since outside the function, h is either monotonically increasing or decreasing. The length of the interval is less than

$$
\frac{1}{2} + \underbrace{\sqrt{\frac{1}{4} + n} - \frac{n}{\sqrt{n + \sqrt{n}}}}_{=:\ell(n)} + 1.
$$

The function $\ell(n)$ can be rewritten as follows:

$$
\begin{aligned}
\ell(n) &= \frac{\sqrt{\left(\frac{1}{4} + n\right)\left(\sqrt{n + \sqrt{n}}\right)} - n}{\sqrt{n + \sqrt{n}}} \\
&\leq \frac{\left(n + \frac{1+\epsilon}{2}\sqrt{n}\right) - n}{\sqrt{n + \sqrt{n}}} \\
&\leq \frac{1+\epsilon}{2}\sqrt{\frac{n}{n + \sqrt{n}}},
\end{aligned} \tag{12}
$$

for every positive ϵ. Inequality (12) is due to the fact that

$$
\begin{aligned}
\left(\frac{1}{4} + n\right)\left(\sqrt{n + \sqrt{n}}\right) &\leq n^2 + n\sqrt{n} + \frac{1}{4}(n + \sqrt{n}) \\
&\leq n^2 + 2\frac{1+\epsilon}{2}n\sqrt{n} \\
&\quad + \frac{(1+\epsilon)^2}{4}n \\
&= \left(n + \frac{1+\epsilon}{2}\sqrt{n}^2\right),
\end{aligned}
$$

for a sufficiently large n. $\square$

ACKNOWLEDGMENTS

This work was partially supported by the DFG under grants BR 2158/2-3, WA 654/14-3, Research Training Group 1042 "Explorative Analysis and Visualization of Large Information Spaces" and by EU under grant DELIS (contract no. 001907).

REFERENCES

[1] M.E.J. Newman and M. Girvan, "Finding and Evaluating Community Structure in Networks," *Physical Rev. E,* vol. 69, no. 026113, http://link.aps.org/abstract/PRE/v69/e026113, 2004.

[2] S. Fortunato and M. Barthelemy, "Resolution Limit in Community Detection," *Proc. Nat'l Academy of Sciences,* vol. 104, no. 1, pp. 36-41, 2007.

[3] E. Ziv, M. Middendorf, and C. Wiggins, "Information-Theoretic Approach to Network Modularity," *Physical Rev. E,* vol. 71, no. 046117, 2005.

[4] S. Muff, F. Rao, and A. Caflisch, "Local Modularity Measure for Network Clusterizations," *Physical Rev. E,* vol. 72, no. 056107, 2005.

[5] P. Fine, E.D. Paolo, and A. Philippides, "Spatially Constrained Networks and the Evolution of Modular Control Systems," *Proc. Ninth Int'l Conf. Simulation of Adaptive Behavior (SAB '06),* 2006.

[6] M. Gaertler, R. Görke, and D. Wagner, "Significance-Driven Graph Clustering," *Proc. Third Int'l Conf. Algorithmic Aspects in Information and Management (AAIM '07),* pp. 11-26, June 2007.

[7] M.E.J. Newman, "Fast Algorithm for Detecting Community Structure in Networks," *Physical Rev. E,* vol. 69, no. 066133, 2004.

[8] A. Clauset, M.E.J. Newman, and C. Moore, "Finding Community Structure in Very Large Networks," *Physical Rev. E,* vol. 70, no. 066111, http://link.aps.org/abstract/PRE/v70/e066111, 2004.

[9] M. Newman, "Modularity and Community Structure in Networks," *Proc. Nat'l Academy of Sciences,* pp. 8577-8582, 2005.

[10] S. White and P. Smyth, "A Spectral Clustering Approach to Finding Communities in Graph," *Proc. SIAM Int'l Conf. Data Mining (SDM '05),* 2005.

[11] R. Guimerà, M. Sales-Pardo, and L.A.N. Amaral, "Modularity from Fluctuations in Random Graphs and Complex Networks," *Physical Rev. E,* vol. 70, no. 025101, 2004.

[12] J. Reichardt and S. Bornholdt, "Statistical Mechanics of Community Detection," *Physical Rev. E,* vol. 74, no. 016110, 2006.

[13] J. Duch and A. Arenas, "Community Detection in Complex Networks Using Extremal Optimization," *Physical Rev. E,* vol. 72, no. 027104, 2005.

[14] M. Gaertler, "Clustering," *Network Analysis: Methodological Foundations,* U. Brandes and T. Erlebach, eds. pp. 178-215, Springer-Verlag, http://springerlink.metapress.com/openurl.asp?genre=article&is sn=0302-9743&volume=3418&spage=178, Feb. 2005.

[15] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing Community Structure Identification," *J. Statistical Mechanics,* 2005.

[16] M.R. Garey and D.S. Johnson, *Computers and Intractability. A Guide to the Theory of* $\mathcal{NP}$-Completeness. W.H. Freeman and Co., 1979.

[17] M. Newman, "Analysis of Weighted Networks," technical report, Univ. of Michigan, July 2004.

[18] C.J. Alpert and A.B. Kahng, "Recent Directions in Netlist Partitioning: A Survey," *Integration: The VLSI J.,* vol. 19, nos. 1-2, pp. 1-81, http://vlsicad.cs.ucla.edu/~cheese/survey.html, 1995.

[19] E. Hartuv and R. Shamir, "A Clustering Algorithm Based on Graph Connectivity," *Information Processing Letters,* vol. 76, nos. 4-6, pp. 175-181, http://citeseer.nj.nec.com/hartuv99clustering.html, 2000.

[20] S. Vempala, R. Kannan, and A. Vetta, "On Clusterings: Good, Bad and Spectral," *Proc. 41st Ann. IEEE Symp. Foundations of Computer Science (FOCS '00),* pp. 367-378, 2000.

[21] I. Giotis and V. Guruswami, "Correlation Clustering with a Fixed Number of Clusters," *Proc. 17th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA '06),* pp. 1167-1176, http://portal.acm.org/citation.cfm?id=1109557.1109686#, 2006.

[22] T. Bui, S. Chaudhuri, F. Leighton, and M. Sipser, "Graph Bisection Algorithms with Good Average Case Behavior," *Combinatorica,* vol. 7, no. 2, pp. pp. 171-191, 1987.

[23] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys,* vol. 31, no. 3, pp. 264-323, 1999.

[24] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, "On Modularity: NP-Completeness and Beyond," Technical Report 2006-19, ITI Wagner, Faculty of Informatics, Universität Karlsruhe (TH), 2006.

[25] W.W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups," *J. Anthropological Research,* vol. 33, pp. 452-473, 1977.

[26] M.E.J. Newman and M. Girvan, "Mixing Patterns and Community Structure in Networks," *Statistical Mechanics of Complex Networks,* Lecture Notes in Physics 625, R. Pastor-Satorras, M. Rubi, and A. Diaz-Guilera, eds., pp. 66-87, Springer-Verlag, 2003.

[27] D. Delling, M. Gaertler, R. Görke, and D. Wagner, "Experiments on Comparing Graph Clusterings," Technical Report 2006-16, ITI Wagner, Faculty of Informatics, Universität Karlsruhe (TH), 2006.

[28] D. Lusseau, K. Schneider, O.J. Boisseau, P. Haase, E. Slooten, and S.M. Dawson, "The Bottlenose Dolphin Community of Doubtful Sound Features a Large Proportion of Long-Lasting Associations," *Behavioral Ecology and Sociobiology,* vol. 54, no. 4, pp. 396-405, 2003.

[29] M.E.J. Newman and M. Girvan, "Finding and Evaluating Community Structure in Networks," http://arxiv.org/abs/cond-mat/0308217, Aug. 2003.

Ulrik Brandes received the bachelor's degree from RWTH Aachen University in 1994 and the PhD degree and Habilitation degree from the University of Konstanz in 1999 and 2002, respectively. He was an associate professor at the University of Passau until 2003. He is currently a full professor of computer science at the University of Konstanz. His research interests include algorithmic graph theory, graph drawing, social network analysis, information visualization, and algorithm engineering.

Daniel Delling received the bachelor's degree in computer science from the Universität Karlsruhe, Germany, in 2006. He is currently working towards the PhD degree in the Chair of Algorithmics I, Faculty of Informatics, University of Karlsruhe. His research interests are computations of shortest paths in large, dynamic, time-dependent graphs, and graph clustering.

Marco Gaertler received the bachelor's degree in mathematics from the Universität Konstanz, Germany, in 2002 and the PhD degree in computer science from the University of Karlsruhe in 2007. His research interests are the clustering of graphs both in a static and in an evolving environment and the analytic visualization of networks.

Robert Görke received the bachelor's degree in technical mathematics from the Universität Karlsruhe, Germany, in 2005. He is currently working towards the PhD degree at the Chair of Algorithmics I, Faculty of Informatics, University of Karlsruhe. His research interests are the clustering of graphs both in a static and in an evolving environment and the analytic visualization of networks.

Martin Hoefer received the bachelor's degree in computer science from the Technische Universität Clausthal, Germany, in 2004 and the PhD degree in computer science from the University of Konstanz in 2007. His research interests are efficient graph algorithms, graph theory, algorithmic game theory, and combinatorial optimization.

Zoran Nikoloski received the BS degree in computer science from Graceland University, Lamoni, Iowa, in 2001 and the PhD degree in computer science from the University of Central Florida, Orlando, in 2005. From 2004 to 2007, he was a postdoctoral researcher in the Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Prague. He is currently a postdoctoral researcher at the Max-Planck Institute for Molecular Plant Physiology, Potsdam-Golm, and the Institute for Biochemistry and Biology, University of Potsdam, Potsdam, Germany.

Dorothea Wagner received the Diplom and PhD degrees in mathematics from the Rheinisch-Westfälische Technische Hochschule, Aachen, Germany, in 1983 and 1986, respectively, and the Habilitation degree from the Technische Universität Berlin in 1992. Until 2003, she was a full professor of computer science at the University of Konstanz. She is currently a full professor at the University of Karlsruhe. She is part of several editorial boards and is the vice president of the German Research Association (DFG). Her research interests include discrete optimization, graph algorithms, and algorithm engineering.