On Connected Two Communities

V. Estivill-Castro\textsuperscript{1} \hspace{1cm} Mahdi Parsa\textsuperscript{1}

\textsuperscript{1} School of Information and Communication Technology
Griffith University,
Nathan, Qld, 4111, Australia.

Abstract

We say that there is a community structure in a graph when the nodes of the graph can be partitioned into groups (communities) such that each group is internally more densely connected than with the rest of the graph. However, the challenge is to specify what is to be dense, and what is relatively more connected (there seems to exist an analogous situation to what is a cluster in unsupervised learning). Recently, Olsen (2012) provided a general definition that seemed to be significantly more generic than others. We make two observations regarding such definition. (1) First, we show that finding a community structure with two equal size communities is \textit{NP}-complete (UNIFORM 2-COMMUNITIES). The first implication of this is that finding a large community seems intractable. The second implication is that, since this is a hardness result for \( k = 2 \), the UNIFORM \( k \)-COMMUNITIES problem is not fixed-parameter tractable when \( k \) is the parameter. (2) The second observation is that communities are not required to be connected in Olsen (2012)'s definition. However, we indicate that our result holds as well as the results by Olsen (2012) when we require communities to be connected, and we show examples where using connected communities seems more natural.

Keywords: Community detection, graph partitioning, complexity, parameterized complexity

1 Introduction

Researchers are now focusing on analyzing the community structure (Boccaletti et al. 2006, Lancichinetti et al. 2010) of graphs and finding so called communities or modules (intuitively these are groups of nodes that are more densely connected to each other than with the rest of the graph). Exploring communities in graphs is important (Lancichinetti et al. 2010) because 1) communities uncover the graph at a coarse level, for example, formulating realistic mechanisms for its genesis and evolution 2) communities provide a new aspect for understanding dynamic processes occurring in the graph and 3) communities reveal relationships among the nodes that are not apparent when inspecting the graph as a whole.

Recently, there has been a large research focus on community structures in graphs (Condon & Karp 2001, Fortunato 2010, Gargi et al. 2011, Kevin J. Lang et al. 2009). However, the main problem is how to define communities in the first place. This is the essential issue tackled by most papers on the topic which have appeared in the literature (Fortunato 2010, references therein). Here we consider the most recent definition of community structure introduced by Olsen (2012). This definition is inspired by the planted \( l \)-partition model, and the hierarchical random graph model introduced by Condon & Karp (2001). Olsen (2012) was able to justify why this becomes a more suitable (and formal) definition of community and initiated the study of the complexity of finding communities by showing that it is \textit{NP}-complete to decide if a group of nodes can be extended to a community in some community structure.

We introduce this generic notion of community using the following notation. Let \( \Pi \) be a partition of the vertices \( V \) of a graph \( G = (V, E) \) \( (\Pi = \{C_1, C_2, \ldots, C_k\} \), with \( \emptyset \neq C_j \subset V \) for \( j = 1, \ldots, k \) and \( \bigcup_{j=1}^{k} C_j = V \), and \( C_j \cap C_{j'} = \emptyset \) for \( j \neq j' \). If \( i \in V \), then we denote the part vertex \( i \) belongs to by \( \Pi_i \). Let \( \forall i \in V \) be a vertex and \( S \subset V \), then \( N_i(S) \) is the number of vertices in \( S \) that are neighbors (adjacent) to the vertex \( i \) (a vertex is never considered adjacent to itself).

Definition 1.1. A community structure for an undirected connected graph \( G = (V, E) \) is a partition \( \Pi \) of \( V \) such that

1. \( |\Pi| \geq 2 \) (we have at least 2 communities),
2. \( |C| \geq 2 \) for all \( C \in \Pi \) (every community has at least 2 members) and
3. \( \forall i \in V, \forall C \in \Pi \) the following holds

\[
\frac{N_i(\Pi_i)}{|\Pi_i| - 1} \geq \frac{N_i(C)}{|C|}.
\]

Each set of the partition is called a community.

Olsen (2012) also showed that finding a community structure in a graph that does not contain \( S_n \) (the stars of \( n \) vertices), for \( n \geq 3 \) can be done in polynomial time. However, nothing could be said about the community structure, like if large communities could be found. Also, it was left open any claim whether finding community structures with few communities is tractable or not. Thus, we investigate the question of finding a community structure with two communities. That ensures one community is large as it must include at least half of the vertices. It turns out that this investigation reveals one more aspect regarding Definition 1.1. We direct the reader to the observation that communities are not
required to be connected. That is, each part \( C \) is not required to be connected. Why do we suggest communities to be connected? Because it is hard not to consider the connected components of a disconnected “community” more naturally as communities in themselves. In fact, the lack of links (vs links to other parts of the graph) suggest the connected components are not to be placed together. We also consider uniform community structure, that is, all communities have the same size. The uniform community structure has gained importance due to its application for clustering and detection of cliques in social, pathological and biological networks (Patkar & Narayanan 2003).

We start with a discussion on the complexity of finding 2-COMMUNITIES. Why we look at the problem of two communities rather than the problem with \( k \) communities? Because by showing the problem with 2 communities is hard, we are showing the problem with \( k \) communities is also hard. Why we look at equal size communities? Because this forces the communities to be large. It seems in practice, the larger a community, the more interesting. We prove that when we require the communities to have equal size the problem is \( NP \)-complete. This result suggests that other lines of attack may be required. For example, a very successful avenue of attack has recently been the application of parameterized complexity theory. Such approach can lead to polynomial-time algorithms on the size of the input (at the cost of exponential-time complexity on the parameter, which can be small in practical settings). A first natural parameter is the number \( k \) of communities. That is, to consider the question whether, for a given graph \( G \), there exists a community structure with exactly \( k \) communities. We call this problem \( k \)-COMMUNITIES. Because we will show that for \( k = 2 \), the problem UNIFORM \( k \)-COMMUNITIES (where communities are all of the same size) is \( NP \)-complete, the problem UNIFORM \( k \)-COMMUNITIES is not fixed-parameter tractable when \( k \) is the parameter. In other words, it is unlikely to have an algorithm for this problem with \( f(k) \cdot \text{poly}(|G|) \) time requirements, for some computable function \( f \).

### 2 Uniform Two-Communities is hard

In this section, we formally define our problem and then show our main hardness result (Theorem 2.1). Our proof is inspired by a hardness result for a graph partitioning problem (Bazgan et al. 2010). We prove this results in several steps.

**UNIFORM \( k \)-COMMUNITIES**

**Instance:** A graph \( G = (V, E) \).

**Parameter:** An integer \( k > 1 \).

**Question:** Does a community structure \( I = \{C_1, C_2, \ldots, C_k\} \) exist such that \( |C_i| \geq |C_j| \) for \( i, j = 1, \ldots, k \)?

**Theorem 2.1** UNIFORM 2-COMMUNITIES is \( NP \)-complete.

**Proof.** We prove that the problem UNIFORM 2-COMMUNITIES is \( NP \)-complete by reducing the graph 2-Clique problem to the UNIFORM 2-COMMUNITIES problem. The version of the CLIQUE problem that asks, for a given non-complete graph \( G \) of size \( n \) (\( n \) is even), whether there exists a complete subgraph of size at least \( n/2 \). This version of the CLIQUE problem is also \( NP \)-complete (Garey & Johnson 1979), and it is not hard to see that the version we will use (whether a graph has a clique of size \( n/2 \)) is also \( NP \)-complete. Now we construct our reduction and we will show that every Yes-instance of the CLIQUE problem maps to a Yes-instance of the UNIFORM 2-COMMUNITIES problem and vice versa.

**Construction 1** Let \( G = (V, E) \) be an instance of the CLIQUE problem with \( V = \{v_1, v_2, \ldots, v_n\} \) and \( E = \{e_1, e_2, \ldots, e_m\} \). We map it to an instance of the UNIFORM 2-COMMUNITIES problem. The set \( V \) is the original set of vertices in the instance of the CLIQUE problem; the set \( V’ = \{v’_1, \ldots, v’_n\} \) consists of as many mirror vertices as in the original set \( V \) of vertices. The set \( F = \{f_1, \ldots, f_{2^k+1}\} \) has two vertices \( f_{2^1}, f_{2^1+1} \) for each non-edge \( v_i \) with \( i = 1, \ldots, p \) and \( f_i \) is an additional vertex. The set \( T = \{f_1, f_2, \ldots, f_{2^k+1}\} \) also has two vertices \( f_{2^1}, f_{2^1+1} \) for each non-edge in the original instance of the CLIQUE problem, and also \( f_1 \) is an additional vertex.

We now describe the set of edges \( E’ \). The set \( E \) consists of all edges in \( V \) is in \( E’ \); that is \( E \subseteq E’ \). In the new instance, \( F \) and \( T \) are two cliques of size \( 2^p+1 \) (that is, in \( E’ \), all vertices of \( F \) are connected among themselves and also in \( E’ \), all vertices of \( T \) are connected among themselves). For \( j = 1, \ldots, n \), \( (v’_i, v’_j) \) is in \( E’ \). The edge set \( E’ \) contains some additional edges as follows:

- Each vertex \( t \in T \) connects to all vertices of \( V \).
- Each vertex \( f \in F \) connects to all vertices of \( V \), unless
  - \( f \) is of the form \( f_2 \) or \( f_{2^1+1} \)
  - \( v_i \) or \( v_j \) is the missing edge (with \( i < j \) in \( G \) corresponding to the pair \( (f_2, f_{2^1+1}) \)).

In this case, the vertex \( f_{2^1+1} \) connects to every vertex in \( V \setminus \{v’_i\} \), and \( f_{2^1} \) connects to every vertex in \( V \setminus \{v’_i\} \).

Finally, the edge \( (f_1, t_1) \) is in \( E’ \).

Note the following about this construction. First, the degree of all vertices in \( V’ \) is one, as these vertices are only connected to their mirror vertices. Second, the degree of every vertex \( t \neq t_1 \) in \( T \) is \( |V| + |T| - 1 \) and the degree of \( t_1 \) is \( |V| + |T| - n + 2p + 1 \). This is because \( t_1 \) is in clique \( T \) (degree \( |T| - 1 \)) and it is connected to each vertex in \( V \) and \( t_1 \) is additionally connected to \( f_1 \). Third, the degree of every vertex \( f \in F \) is at least \( |F| - 1 \) as it belongs to the clique \( F \). The vertex \( f_1 \) has degree \( |F| + |V| \), but the other vertices in \( F \) have degree \( |F| + |V| - 2 \), as each of these vertices looses one connection to one vertex in \( V \) that is an endpoint of a non-edge.

Figure 1 provides a more specific example of the reduction. Clearly, this construction can be performed in polynomial time. We only need to show that a Yes-instance of the first problem maps to a Yes-instance of the second problem and vice versa.

**Proposition 2.2** A Yes-instance of the CLIQUE problem maps to a Yes-instance of the UNIFORM 2-COMMUNITIES problem.
Proceeding of the Thirty-Sixth Australasian Computer Science Conference (ACSC 2013), Adelaide, Australia

\[
N_f(\Pi_1) \frac{|I_1| - 1}{|I_1| - 1} = \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|C| - |F|}{|I_2|} = N_f(\Pi_2) \frac{|I_2| - 1}{|I_2|}.
\]

Case 2: The vertex \( f \) connects to every vertex in \( C \) except one.

We recall that the degree of every vertex in \( F \) that is not \( f_1 \) is \( |F| + |V| - 2 \). Since \( |F| \geq 3 \), then we have

\[
N_f(\Pi_1) \frac{|I_1| - 1}{|I_1| - 1} = \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 2 + |C|}{|I_1| - 1} = \frac{|F| - 2 + |C|}{|I_1| - 1} = \frac{|C| - |F|}{|I_2|} = N_f(\Pi_2) \frac{|I_2| - 1}{|I_2|}.
\]

Case 3: \( f = f_1 \).

According to the construction, \( f \) connects to every vertex in \( C \) and also connects to \( t_1 \). Hence, we have

\[
N_f(\Pi_1) \frac{|I_1| - 1}{|I_1| - 1} = \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|F| - 1 + |C|}{|I_1| - 1} = \frac{|F| - 1 + |C|}{|I_1| - 1} \geq \frac{|C| - |F|}{|I_2|} = N_f(\Pi_2) \frac{|I_2| - 1}{|I_2|}.
\]

**Figure 1:** The dotted lines mean that there is no edge between the two end points of the line. A branch of four edges at \( f_1 \) and each vertex of \( T \) mean those vertices connect to all vertices of \( V \).
The last type of vertex in \( \Pi_1 \) that we check for Inequality (1) belongs to \( C' \), but these vertices have degree 1 in \( \Pi_2 \) and degree zero in \( \Pi_1 \), so this is immediate.

To complete the proof that we have a YES-instance of 2-COMMUNITIES we need to establish Inequality (1) for the vertices in \( \Pi_2 \). We start by showing that Inequality (1) holds for every vertex \( e \) in \( C \).

First, \( N_e(\Pi_2) = |T| + 1 + N_e(C) \), since \( e \) connects to all vertices in \( T \), all its neighbors in \( C \) and also connects to its mirror \( e' \). Second, assume \( x_e \) is the number of non-edges in \( C \) with endpoint in \( e \), then we have \( N_e(\Pi_2) = |T| + 1 + |C| - 1 - x_e = |T| + |C| - x_e \). Third, if there exists a missing edge \((e,v)\) with \( v \in C \), corresponding to this missing edge, there exists exactly a missing edge between \( e \) and a vertex \( f \) in \( F \). Therefore, \( N_e(\Pi_1) \) equals \( |F| - x_e + N_e(C) \). Since \( |C| = |C| \) and \( |C| \geq N_e(C) \), then we have

\[
\frac{N_e(\Pi_2)}{|\Pi_2| - 1} = \frac{|T| + |C| - 1}{|\Pi_2| - 1} \\
\geq \frac{|F| + N_e(C) - x_e}{|\Pi_1| - 1} \\
\geq \frac{|F| + |C| - x_e}{|\Pi_1|} \\
= \frac{N_e(\Pi_1)}{|\Pi_1|}.
\]

We now argue for the second type of vertices in \( \Pi_2 \). We show that every vertex \( t \) in \( T \) satisfies Inequality (1). Since \( |\Pi_1| = |\Pi_2|, |T| \geq 3 \) and \( |C| = |C| \) we have for every \( t \neq t_1 \)

\[
\frac{N_e(\Pi_2)}{|\Pi_2| - 1} = \frac{|T| + |C| - 1}{|\Pi_2| - 1} \\
= \frac{|T| + |C| - 1}{|\Pi_1| - 1} \\
\geq \frac{|C| + 1}{|\Pi_1|} \\
\geq \frac{N_e(\Pi_1)}{|\Pi_1|}.
\]

Similarly, for \( t = t_1 \) we have

\[
\frac{N_e(\Pi_2)}{|\Pi_2| - 1} = \frac{|T| + |C| - 1}{|\Pi_2| - 1} \\
= \frac{|T| + |C| - 1}{|\Pi_1| - 1} \\
\geq \frac{|C| + 1}{|\Pi_1|} \\
\geq \frac{N_e(\Pi_1)}{|\Pi_1|}.
\]

And to complete all vertices of \( \Pi_2 \) we consider the mirror vertices in \( C' \), but again these vertices have degree one to their community \( \Pi_2 \) and zero to the other part \( \Pi_1 \), so trivially they satisfy Inequality (1).

Therefore, a YES-instance of the CLIQUE problem maps to a YES-instance of the UNIFORM 2-COMMUNITIES problem.

Now we show that the reverse is true.

**Proposition 2.3** A YES-instance of the Uniform 2-COMMUNITIES problem maps to a YES-instance of the CLIQUE problem.

Suppose \( I = (G', \Pi_1, \Pi_2) \) is a YES-instance of the Uniform 2-COMMUNITIES problem. We justify the following observations to show the pre-image of \( I \) is a YES-instance of the CLIQUE problem.

**Observation 2.4** (about mirror vertices): In each YES-instance of 2-COMMUNITIES, the mirror vertices \( v'_j \) must be in the same community as \( v_j \), with \( j = 1, \ldots, n \).

**Proof:** If a mirror vertex \( v' \) is in community \( \Pi_1 \), and its corresponding vertex \( v \) is in community \( \Pi_2 \neq \Pi_1 \), then \( N_{v'}(\Pi_1) = 0 \), while \( N_{v}(\Pi_2) > 0 \). This contradicts that the vertex \( v' \) must satisfy Inequality (1).

**Observation 2.5** The set \( T \) cannot be cut by the community structure.

**Proof:** (by contradiction) Suppose \( T \) is divided in \( (T_1, T_2) \) with \( T_i \subseteq \Pi_i \) and \( i = 1, 2 \). Also assume that the set \( V \) is cut in \( (V_1, V_2) \) with with \( V_i \subseteq \Pi_i \) and \( i = 1, 2 \), where \( V_1 \) and \( V_2 \) could be empty. Moreover, assume that \( F \) is divided in \( (F_1, F_2) \) with \( F_i \subseteq \Pi_i \) and \( i = 1, 2 \). We will face the following cases where each one leads to a contradiction.

**Case 1:** \( T_1 = \{t_1\} \) and \( F_1 = \{f_1\} \).

In this case \( |T_2| = |T| - 1 \) and \( |F_2| = |F| - 1 \). Therefore, \( |T_2| \) is equal to \( |F_2| \) and both are equal to \( |T| - 1 \) because \( |T| = |F| \). Since \( \Pi \) is a community structure, then every \( \Pi \in T_2 \) must satisfy Inequality (1). But,

\[
\frac{N_e(\Pi_2)}{|\Pi_2| - 1} = \frac{|T_2| - 1 + |V_2|}{|F_2| + 2|V_2| + |T_2| - 1} \\
= \frac{|T| + |V_2| - 2}{2|T| + 2|V_2| - 3} < \frac{1}{2}.
\]

This statement contradicts the vertex \( t \) must satisfy Inequality (1).

**Case 2:** \( T_1 = \{t_1\} \) and \( f_1 \in F_2 \).

The neighbors of the vertex \( t_1 \) in \( \Pi_1 \) are all vertices in \( V_1 \) as \( T_1 = \{t_1\} \). Also, the neighbors of the vertex \( t_1 \) in \( \Pi_2 \) are all vertices in \( T_2 \), with also all vertices in \( V_2 \) and \( f_1 \). Therefore, we have

\[
\frac{N_e(\Pi_1)}{|\Pi_1| - 1} = \frac{|T_1| - 1 + |V_1|}{|F_1| + 2|V_1| + |T_1| - 1} \\
= \frac{1 - 1 + |V_1|}{|F_1| + 2|V_1|} \\
= \frac{|V_1|}{|F_1| + 2|V_1|} \\
\leq \frac{1}{2}.
\]
Since \(|T_1| = 1, |T_2| = |T| - 1| and |T| + 1 > |F| \geq |F_2|\), then we have
\[
\frac{N_t(\Pi_1)}{|\Pi_1| - 1} = \frac{|T_1| + |V_1| - 1}{|\Pi_1| - 1} \geq \frac{N_t(\Pi_2)}{|\Pi_2|} = \frac{|T_2| + |V_2|}{|\Pi_2|}. \tag{2}
\]
Also, each vertex \(t'\) in \(T_2\) must satisfy Inequality (1), therefore,
\[
\frac{N_{t'}(\Pi_2)}{|\Pi_2| - 1} = \frac{|T_2| + |V_2| - 1}{|\Pi_2| - 1} \geq \frac{N_{t'}(\Pi_1)}{|\Pi_1|} = \frac{|T_1| + |V_1|}{|\Pi_1|}. \tag{3}
\]
From Inequality (2) we get
\[
|\Pi_2|(|T_1| + |V_1|) \geq (|\Pi_1| - 1)(|T_2| + |V_2|) + |\Pi_2|. \tag{4}
\]
From Inequality (3) we get
\[
|\Pi_1|(|T_2| + |V_2| - 1) \geq (|\Pi_2| - 1)(|T_1| + |V_1|),
\]
or equivalently
\[
|\Pi_1|(|T_2| + |V_2| - 1) \geq |\Pi_2|(|T_1| + |V_1|) - (|T_1| + |V_1|). \tag{5}
\]
Combining Inequality (4) and Inequality (5) we arrive at
\[
|\Pi_1| + |\Pi_2| \leq (|T_1| + |T_2|) + (|V_1| + |V_2|).
\]
This inequality contradicts the fact that \(F\) is not empty.

**Observation 2.6** The set \(F\) can not be cut by the community structure.

**Proof:** Assume that \(F\) is cut into \((F_1, F_2)\) where \(F_1 \subseteq \Pi_1, F_2 \subseteq \Pi_2\) and \(F_1 \neq \emptyset\). Also assume that the original set \(Y\) is cut into \((V_1, V_2)\) with \(V_i \subseteq \Pi_i\) and \(i = 1, 2\), where \(V_1\) and \(V_2\) could be empty. Moreover, since \(T\) can not be split, without loss of generality we can assume that \(T = V_1 \cup V_2 \cup F_1, T = V_1 \cup V_2 \cup F_2\).

We show that \(F_2\) is empty or we have a contradiction.

Assume that \(F_2\) is not empty and let \(f \in F_2\). Then we will face the following cases.

**Case 1:** \(f = f_1\).

The neighbors of vertex \(f_1\) in \(\Pi_2\) are all vertices in \(V_2\), plus all vertices in \(F_2 - \{f\}\) and the vertex \(t_1\). Therefore, we have
\[
\frac{N_{f_1}(\Pi_2)}{|\Pi_2| - 1} = \frac{|F_2| + |V_2|}{|\Pi_2| - 1} - \frac{1}{2}.
\]
Similarly, the neighbors of the vertex \(f_1\) in \(\Pi_1\) are all vertices in \(V_1\), plus all vertices in \(F_1\), therefore,
\[
\frac{N_{f_1}(\Pi_1)}{|\Pi_1|} = \frac{|F_1| + |V_1|}{|\Pi_1|} \geq \frac{1}{2}
\]
This statement shows that the vertex \(t_1\) violates Inequality (1).

**Case 2:** \(f \neq f_1\) and \(f\) does not connect to a vertex of \(V_2\).

The neighbors of the vertex \(f\) in \(\Pi_2\) are all vertices in \(V_2\) except one, plus all vertices in \(F_2 - \{f\}\), hence,
\[
\frac{N_f(\Pi_2)}{|\Pi_2| - 1} = \frac{|F_2| - 1 + |V_2| - 1}{|\Pi_2| - 1} \leq \frac{1}{2}.
\]
Similarly, the neighbors of the vertex \(f\) in \(\Pi_1\) are all vertices in \(V_1\), plus all vertices in \(F_1\), therefore,
\[
\frac{N_f(\Pi_1)}{|\Pi_1|} = \frac{|F_1| + |V_1|}{|\Pi_1|} \geq \frac{1}{2}
\]
This contradicts the fact that the vertex \(f\) must satisfy Inequality (1).

**Case 3:** \(f \neq f_1, f\) does not connect to a vertex of \(V_1\) and \(|F_1| \geq 2\).

The neighbors of the vertex \(f\) in \(\Pi_2\) are all vertices in \(V_2\), plus all vertices in \(F_2 - \{f\}\), hence,
\[
\frac{N_f(\Pi_2)}{|\Pi_2| - 1} = \frac{|F_2| - 1 + |V_2|}{|\Pi_2| - 1} \leq \frac{1}{2}
\]
Similarly, the neighbors of the vertex \( v \) in \( \Pi_1 \) are all vertices in \( V_1 \) except one, plus all vertices in \( F_1 \). Moreover, the size of \( |F_1| \geq 2 \), therefore,

\[
\frac{N_f(\Pi_1)}{|\Pi_1|} = \frac{|F_1| + |V_1| - 1}{|F_1| + 2|V_1|} \geq \frac{1}{2}.
\]

Similar to Case 1 above, we have a contradiction that the vertex \( v \) violates Inequality (1).

**Case 4:** If \( f \neq f_1 \), \( f \) does not connect to a vertex of \( V_1 \) and \( |F_1| < 2 \).

Since \( F_1 \) is not empty, we must have \( |F_1| = 1 \), and by Case 1, \( F_1 = \{ f_1 \} \), while \( |F_2| = |F| - 1 \). Moreover, the vertex \( f_1 \) must satisfy Inequality (1). Therefore, we have

\[
\frac{N_f(\Pi_1)}{|\Pi_1|} = \frac{|V_1|}{1 + 2|V_1| - 1} = \frac{1}{2}.
\]

Now, to find the value of \( N_f(\Pi_2)/|\Pi_2| \), we note that \( f_1 \) is adjacent to all the vertices in \( V_2 \) and \( t_1 \). Moreover, \(|F_2| = |T| - 1 \). Thus,

\[
\frac{N_f(\Pi_2)}{|\Pi_2|} = \frac{|V_2| + |F_2| + 1}{2|V_2| + |F_2| + |T|} = \frac{|V_2| + |T|}{2|V_2| + 2|T| - 1} = \frac{1}{2}.
\]

This is a contradiction since the vertex \( f_1 \) must satisfy Inequality (1) for a 2-community.

**Observation 2.7** The set \( F \) and the set \( T \) do not belong to a same community.

**Proof:** (by contradiction) Assume \( V \) is cut in \( (V_1, V_2) \). Also, assume \( \Pi_1 = F \cup T \cup V_1 \cup V_2 \) and \( \Pi_2 = V_2 \cup V_2 \). Consider a vertex \( t \neq t_1 \) in \( T \). The neighbors of the vertex \( t \) in \( \Pi_1 \) are all vertices in \( V_1 \), plus all vertices in \( T \) \( \setminus \{ t \} \). Similarly, the neighbors of the vertex \( t \) in \( \Pi_2 \) are all vertices in \( V_2 \). Since \( \Pi_1, \Pi_2 \) is a community structure, then the vertex \( t \) must satisfy Inequality (1). Therefore, we have

\[
\frac{N_t(\Pi_1)}{|\Pi_1|} = \frac{|V_1| + |T| - 1}{|F| + 2|V_1| + |T| - 1} = \frac{N_t(\Pi_2)}{|\Pi_2|} = \frac{|V_2|}{2|V_2|}.
\]

By simplifying the above inequality we arrive at

\[
\frac{|V_1| + |T| - 1}{|F| + 2|V_1| + |T| - 1} \geq \frac{1}{2}.
\]

Now the above inequality implies that

\[
2 \cdot (|V_1| + |T| - 1) \geq |F| + 2|V_1| + |T| - 1,
\]

and hence

\[
2 \cdot |V_1| + 2 \cdot |T| - 2 \geq |F| + 2|V_1| + |T| - 1.
\]

Since \( |T| = |F| \), the last inequality implies that \(-2 \geq -1\), which is a contradiction. Therefore, \( T \) and \( F \) are not in a same community.

**Observation 2.8** If \((V_1, V_2)\) is a cut of \( V \) based on community structure \((\Pi_1, \Pi_2)\), then \( \Pi_1 = F \cup V_1 \cup V_2 \), \( \Pi_2 = T \cup V_2 \cup V_2 \) and \( V_1 \) is a clique.

**Proof:** (by contradiction) Assume \( V_1 \) is not a clique. There, there exist a missing edge between two vertices of \( V_1 \). Suppose \( v \in V_1 \) is one of the end points of the mentioned missing edge. Assume \( x_v \) is the number of missing edge in \( V_1 \) with one end in \( v \). Clearly \( x_v \geq 1 \). Also assume that \( y_v \) is the number of missing edge in \( V_2 \) with one end in \( v \).

Since \( \Pi_1, \Pi_2 \) is a community structure, the vertex \( v \) must satisfy Inequality (1), therefore we have

\[
\frac{N_v(\Pi_1)}{|\Pi_1|} = \frac{(|V_1| - 1) - x_v + |F| - (x_v + y_v)}{|V_1| - 1} \geq \frac{N_v(\Pi_2)}{|\Pi_2|} = \frac{|V_2| - y_v + |T|}{|\Pi_2|}.
\]

Since \( |\Pi_1| = |\Pi_2| \), therefore \( |V_1| = |V_2| \). Now we simplify Inequality (6) as follows.

\[
|\Pi_2|)((|V_1| - 1) - x_v + |F| - (x_v + y_v)) \geq ((|\Pi_1| - 1)(|V_2| - y_v + |T|)).
\]

Now we substitute \( \Pi_1 \) with \( \Pi_2 \), \( F \) with \( |T| \) and \( |V_1| \) with \( |V_2| \) as they are equal to each other. Therefore, we get

\[
|\Pi_2|((|V_2| - 1) - x_v + |T| - (x_v + y_v)) \geq ((|\Pi_2| - 1)(|V_2| - y_v + |T|)).
\]

After canceling equal values from the both sides of the inequality and simplifying it, then we arrive at

\[
|V_2| + |T| \geq |\Pi_2| + 2 \cdot x_v |\Pi_2|.
\]

But, the latter inequality represents a contradiction since \( x_v \geq 1 \) and the value of the left side of the above inequality is in fact less than \( |\Pi_2| \). Therefore \( V_1 \) is a clique.

**Observation 2.9** The size of \( V_1 \) is at least \( n/2 \).

**Proof:** Observation 2.8 shows that \( V_1 \) is a clique. Also we know that \( |\Pi_1| = |\Pi_2| \), therefore, \( |V_1| = |V_2| \). Hence, the size of \( |V_1| = n/2 \).

3 Some observations on the definition of community structure

As we alluded in the introduction, our aim was to investigate when can we find a large community within a community structure. Thus, we focused on the 2-COMMUNITIES problem since this ensures one community is large as it must include half of the vertices of the underlying graph. However, we discovered that requesting connectivity for each community changes the problem. According to Definition 1.1, communities are not required to be connected. That is, each community \( C \) in the community structure is not required to be connected. In other words, we generalized Definition 1.1.

**Observation 3.1** There are graphs that do not have a 2-community structure, if we demand that each community must be connected; but have a 2-community structure under Definition 1.1.
4 Classes of graphs with 2-communities

The example of Figure 3 enables us to reflect on what graphs accept 2-communities. In particular, since a community is a concept close to a cluster or a region of high density, a community structure with 2-communities must imply some low density between the communities. We can establish a relation between the notion of a cut in a graph and the notion of a 2-community structure. A cut in a graph $G= (V, E)$ is a partition $(\Pi_1, \Pi_2)$ of vertices of $G$, and is called balanced if $|\Pi_1| = |\Pi_2|$. The set of edges whose end points are in different subsets of the partition is called a cut set. A min-cut is a cut with the smallest cut-set size (and can be found in polynomial time, although it might not be balanced). Figure 4 illustrates a min-cut of size two.

We show that a balanced min-cut of a graph $G$ constitutes a 2-community structure.

**Observation 4.1** If $(\Pi_1, \Pi_2)$ is a cut of size two of a graph $G$ with cut-set $S$, then every vertex that is not an endpoint of an edge in $S$ satisfies Inequality (1).

This is immediate. Every vertex $v \in \Pi_1$, that is not an endpoint of an edge in the cut-set $S$, has no connections to the other side. Thus, the value of $N_v(\Pi_1)$ with $j \neq i$ is zero.

**Observation 4.2** If $(\Pi_1, \Pi_2)$ is a minimum cut of graph $G$ with $|\Pi_1| = |\Pi_2|$, then $(\Pi_1, \Pi_2)$ forms a 2-community structure.

**Proof:** Based on Observation 4.1, we only need to show that every vertex in the cut-set satisfies Inequality (1). Assume a vertex $v \in \Pi_1$ is an endpoint of an edge in the cut-set $S$. The number of neighbors of vertex $v$ in the set $\Pi_1$ is equal or greater that the number of neighbors of vertex $v$ in the set $\Pi_2$. Otherwise, we can make a smaller cut-set by moving vertex $v$ to the set $\Pi_2$ (contradicting the fact that the size of $S$ is minimum among all cut-sets). Therefore,

$$\frac{N_v(\Pi_1)}{|\Pi_1| - 1} \geq \frac{N_v(\Pi_2)}{|\Pi_2|},$$

since the size of $\Pi_1$ is equal to the size of $\Pi_2$. That is the vertex $v$ satisfies Inequality (1).

**Corollary 4.3** Paths and cycles with even number of vertices have a 2-communities structure.

The above corollary can be extended to the paths and cycles with odd number of vertices.

**Lemma 4.4** The 2-Communities problem for graphs with maximum degree two and $|V| \geq 3$ can be solved in polynomial time.

**Proof:** Let $G= (V, E)$ be a graph with maximum degree two. If $G$ is not a connected graph, then consider
any connected component $\Pi_1$ as one community and $\Pi_2 = V - \Pi_1$ as the second community. The partition $(\Pi_1, \Pi_2)$ forms a 2-community structure since there is no edge between the two sets. Thus, based on Observation 4.1, all vertices satisfy Inequality (1).

Assume now that $G$ is a connected graph. Since the maximum degree is at most two and the graph is connected, the graph $G$ is a path or a cycle. We can construct a two communities as follows.

Case 1: The graph $G$ is a path. We pick a vertex $v$ of degree one and add all vertices in a path of length $\lfloor |V|/2 \rfloor$ from $v$ into a set $\Pi_1$. The rest of vertices is placed in a set $\Pi_2$. It is not hard to see that all vertices in $\Pi_1$, and $\Pi_2$ satisfy Inequality (1). Hence $(\Pi_1, \Pi_2)$ is a 2-community structure.

Case 2: $G$ is a cycle. We pick a vertex $v$ of the cycle and add all vertices in a path of length $\lfloor |V|/2 \rfloor$ from $v$ into a set $\Pi_1$. Again, the rest of vertices is placed in a set $\Pi_2$. A similar argument to Case 1 shows that $(\Pi_1, \Pi_2)$ is a 2-community structure. $\square$

5 Conclusion and open problems

We studied the computational complexity of the uniform $k$-COMMUNITIES problem. We showed that this problem is $NP$-complete even for $k = 2$. The complexity of the problem is not known if we drop the uniformity (size of all communities are equal) condition as in the $k$-COMMUNITIES problem. This leads to observations for detecting a community structure of size two. We also showed that the known algorithm (Olsen 2012) for finding a community structure may find a solution that is very far from an optimal solution to the 2-COMMUNITIES problem. Moreover, we observed that there may exist graphs where some communities are not connected. Since requiring all communities to be connected is consistent with previous work, we suggest the definition should incorporate this requirement.

Our work here leads to several interesting open problems for finding a community structure with a specific property. We list some of them.

Problem 1: Determine the computational complexity of the uniform $k$-COMMUNITIES problem on different classes of graphs, such as planar graphs and regular graphs.

Problem 2: Determine the computational complexity of the $k$-COMMUNITIES problem.

Problem 3: Determine the computational complexity of finding a community structure with one community of size at least $k$.

Another interesting connection of the $k$-COMMUNITIES problem seems to be a relatively similar problem in the literature which is called the SPARSEST CUTS problem. A sparsest cut of a graph $G = (V, E)$ is a partition $(V_1, V \setminus V_1)$ having the minimum density

$$|\text{cut-set}(V_1)|/|V_1||V \setminus V_1|$$

among all partitions in the graph, where

$$\text{cut-set}(V_1) = \{e = \{u, v\} \in E \mid u \in V_1 \text{ and } v \notin V_1\}.$$

The SPARSEST CUTS problem is $NP$-hard; however, it can be solved in polynomial time on trees and planar triconnected graphs (Matula & Shahrkhi 1990). It is not hard to see that in paths and in cycles a sparsest cut is also a 2-community structure and vice versa. However, it would be interesting to know on what graph classes the concept of 2-community structure and of sparsest-cut are identical.

References


