A Note on a Connection Between Small Set Expansions and Modularity Clustering

Bhaskar DasGupta

Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607

Devendra Desai

Department of Computer Science, Rutgers University, Piscataway, NJ 08854

Abstract

In this note, we explore a connection between two seemingly different problems from two different domains: the *small-set expansion* problem studied in unique games conjecture, and a popular community finding approach for social networks known as the *modularity clustering* approach. We show that a sub-exponential time algorithm for the small-set expansion problem leads to a sub-exponential time constant factor approximation for some hard input instances of the modularity clustering problem.

Keywords: Small-set expansion, modularity clustering, social network
2010 MSC: 68Q25, 68W25

1. Introduction and Definitions

All graphs considered in this note are *undirected* and *unweighted*[1]. Let $G = (V, E)$ denote the given input graph with $n = |V|$ nodes and $m = |E|$ edges, let d_v denote the degree of a node $v \in V$, and let $A(G) = \left[a_{u,v}(G)\right]$

Email addresses: dasgupta@cs.uic.edu (Bhaskar DasGupta), devdesai@cs.rutgers.edu (Devendra Desai)

URL: www.cs.uic.edu/~dasgupta (Bhaskar DasGupta), paul.rutgers.edu/~devdesai (Devendra Desai)

[1]Our result can be extended for the more general case of directed weighted graphs by using the correspondence of these versions with unweighted undirected graphs as outlined in [3, Section 5.1].

denote the adjacency matrix of G, *i.e.*, $a_{u,v}(G) = \begin{cases} 1, & \text{if } \{u,v\} \in E \\ 0, & \text{otherwise.} \end{cases}$ Since our result spans over two distinct research areas, we summarize the relevant definitions from both research fields [1, 5] below for convenience.

($\mathfrak{a}$) By a "set of (k) *communities*" we mean a partition of the set of nodes V into (k) non-empty parts.

($\mathfrak{b}$) If G is d-regular for some given d, then its symmetric *stochastic walk* matrix is denoted by $\widehat{A}(G)$, and is defined as the $n \times n$ real symmetric matrix $\widehat{A}(G) = \left[\frac{a_{u,v}(G)}{d} \right]$.

($\mathfrak{c}$) For a real number $\tau \in [0,1)$, the τ*-threshold rank* of G, denoted by $\mathrm{rank}_\tau(G)$, is the number of eigenvalues λ of $\widehat{A}(G)$ satisfying $|\lambda| > \tau$.

($\mathfrak{d}$) For a subset $\emptyset \subset S \subset V$ of nodes, the following quantities are defined:

- The (normalized) *measure* of S is $\mu(S) = \frac{|S|}{n}$.
- The (normalized) *expansion* of S is $\Phi(S) = \frac{\left| \left\{ \{u,v\} \,|\, u \in S,\, v \notin S,\, \{u,v\} \in E \right\} \right|}{\sum_{v \in S} d_v}$
- The (normalized) *density* of S is $\mathsf{D}(S) = 1 - \Phi(S)$.
- The *modularity* value of S is $\mathsf{M}(S) = \frac{1}{2m} \left(\sum_{u,v \in S} \left(a_{u,v} - \frac{d_u d_v}{2m} \right) \right)$

($\mathfrak{e}$) The modularity of a set of communities $\mathbf{S}$ is $\mathsf{M}(\mathbf{S}) = \sum_{S \in \mathbf{S}} \mathsf{M}(S)$.

($\mathfrak{f}$) The goal of the *modularity k-clustering* problem on an input graph G is to find a set of at most k communities $\mathbf{S}$ that *maximizes* $\mathsf{M}(\mathbf{S})$. Let $\mathsf{OPT}_k(G) = \max_{\mathbf{S} \text{ is a set of at most k communities}} \left\{ \mathsf{M}(\mathbf{S}) \right\}$ denote the optimal modularity value for a modularity k-clustering; it is easy to verify that $0 \leq \mathsf{OPT}_k(G) < 1$.

($\mathfrak{g}$) The goal of the *modularity clustering* problem on G is to find a set of (unspecified number of) communities $\mathbf{S}$ that *maximizes* $\mathsf{M}(\mathbf{S})$. Let $\mathsf{OPT}(G)$ denote the optimal modularity value for a modularity clustering; obviously, $\mathsf{OPT}(G) = \mathsf{OPT}_n(G)$.

($\mathfrak{h}$) $\exp(\xi)$ denotes $2^{c\xi}$ for some constant $c > 0$ that is independent of ξ.

The modularity clustering problems as described above is *extremely popular* in practice in their applications to biological networks [7, 8] as well as to social networks [4–6]. For relevant computational complexity results for modularity maximization, see [2, 3]. The following results from [3] demonstrate the computational hardness of $\mathsf{OPT}_2(G)$ and $\mathsf{OPT}(G)$ even if G is a regular graph.

Theorem 1.1. [3]

(a) *For every constant $d \geq 9$, there exists a collection of d-regular graphs G of n nodes such it is* NP*-hard to decide if* $\mathsf{OPT}_2(G) \geq \frac{1}{2} - \frac{2c}{dn}$ *or if* $\mathsf{OPT}_2(G) \leq \frac{1}{2} - \frac{2c+2}{dn}$ *for some positive $c = O(\sqrt{n}\,)$.*

(b) *There exists a collection of $(n-3)$-regular graphs G of n nodes such it is* NP*-hard to decide if* $\mathsf{OPT}(G) > \frac{0.9388}{n-4}$ *or if* $\mathsf{OPT}(G) < \frac{0.9382}{n-4}$.

2. Our Result

Theorem 2.1. *Let G be a d-regular graph. Then, for some constant $0 < \varepsilon < 1/2$, there is an algorithm $\mathcal{A}_\varepsilon$ with the following properties:*

- *$\mathcal{A}_\varepsilon$ runs in sub-exponential time,* i.e., *in time* $\exp(\delta\, n)$ *for some constant $0 < \delta = \delta(\varepsilon) < 1$ that depends on ε only.*
- *$\mathcal{A}_\varepsilon$ correctly distinguishes instances G of modularity clustering with* $\mathsf{OPT}(G) \geq 1 - \varepsilon$ *from instances G with* $\mathsf{OPT}(G) \leq \varepsilon$.

(Note that we make no claim if $\varepsilon <$ $\mathsf{OPT}(G) < 1 - \varepsilon$.)

Proof.[2] Set $\varepsilon = 10^{-6}$. We assume that G is d-regular, and either $\mathsf{OPT}(G) \geq 1 - 10^{-6}$ or $\mathsf{OPT}(G) \leq 10^{-6}$.

Preliminary Algebraic Simplification

Let $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ be a set of communities of G. The objective function $\mathsf{M}(\mathbf{S})$ can be equivalently expressed as follows via simple algebraic manipulation [2, 4–6]. Let m_i denote the number of edges whose both endpoints are in S_i, m_{ij} denote the number of edges one of whose endpoints is in S_i and the other in S_j and $D_i = \sum_{v \in S_i} d_v$ denote the sum of degrees of nodes in S_i. Then, $\mathsf{M}(\mathbf{S}) = \sum_{S_i \in \mathbf{S}} \left(\frac{m_i}{m} - \left(\frac{D_i}{2m}\right)^2 \right)$.

[2]We have made no significant attempts to optimize the constants in Theorem 2.1.

We will provide an approximation for $\mathsf{OPT}_2(G)$ and then use the result that $\mathsf{OPT}_2(G) \geq \frac{\mathsf{OPT}(G)}{2}$ proved in [3]. Note that if if $\mathsf{OPT}(G) \leq 10^{-6}$ then obviously $\mathsf{OPT}_2(G) \leq 10^{-6}$, whereas if $\mathsf{OPT}(G) \geq 1 - 10^{-6}$ then $\mathsf{OPT}_2(G) \geq \frac{1}{2} - \frac{10^{-6}}{2}$. Consider a partition $\mathbf{S}$ of V into exactly two sets, say S and $\overline{S} = V \setminus S$ with $0 < \mu(S) \leq 1/2$. By Lemma 2.2 of [3], $\mathsf{M}(S) = \mathsf{M}(\overline{S})$ and thus

$$\mathsf{M}(\mathbf{S}) = 2 \times \left(\frac{m_1}{m} - \left(\frac{|S|}{n} \right)^2 \right) \;=\; 2 \times \left(\frac{\frac{1}{2}\, \mathsf{D}(S)\, d\, |S|}{\frac{1}{2}\, d\, n} - \mu(S)^2 \right)$$

$$= 2 \times \big(\mathsf{D}(S)\, \mu(S) - \mu(S)^2 \big)$$

Thus, letting $\mathsf{D} = \mathsf{D}(S), \mu = \mu(S)$ and $\Phi = \Phi(S) = 1 - \mathsf{D}$, the goal of modularity 2-clustering is to maximize the following function f over all possible valid choices of D and μ:

$$f(\mu, \mathsf{D}) = 2 \times \left(\mu\, \mathsf{D} - \mu^2 \right) = 2 \times \big(\mu(1 - \Phi) - \mu^2 \big)$$

Let $\mathbf{S}^\star = \{ S^\star, \overline{S^\star} \}$ be an optimal solution for modularity 2-clustering of G, with $\mathsf{D} = \mathsf{D}^\star, \mu = \mu^\star, \Phi = \Phi^\star$ (and thus $\mathsf{OPT}_2(G) = f(\mu^\star, \mathsf{D}^\star)$). Obviously,

$$\left| \mu^\star - \frac{\mathsf{D}^\star}{2} \right| < \frac{\mathsf{D}^\star}{2}$$

$$f\left(\frac{\mathsf{D}^*}{2} + \delta, \mathsf{D}^* \right) = f\left(\frac{\mathsf{D}^*}{2} - \delta, \mathsf{D}^* \right) \text{ for any positive } \delta > 0$$

Note that we need to show that, if $\mathsf{OPT}_2(G) = f(\mu^\star, \mathsf{D}^\star) > \frac{1}{2} - \frac{10^{-6}}{2}$, then there is an algorithm $\mathcal{A}_\varepsilon$ as described in Theorem 2.1 that outputs a valid choice of μ and D, say μ' and D', such that $f(\mu', \mathsf{D}') > 10^{-6}$.

Guessing $\mathsf{D}^\star$

Note that there are at most $\mathrm{O}\left(d\, n^2\right)$ choices for $\mathsf{D}^\star$ since $\mathsf{D}^\star$ is of the form $i/(j\,d)$ for $j \in \{1, 2, \ldots, n/2\}$ and $i \in \{1, 2, \ldots, j\, d\}$. In the sequel, we will run our algorithm for each choice of $\mathsf{D}^\star$ and take the best of these solutions. Thus, it will suffice to prove our approximation bound assuming we have guessed $\mathsf{D}^\star$ exactly.

In the remainder of the proof, we will make use of results for small-set expansion from [1]. The description is self-contained, and the reader *will* *not need any prior knowledge of expansion properties of graphs.* Remember that we assume that $f(\mu^\star, \mathsf{D}^\star) > \frac{1}{2} - \frac{10^{-6}}{2}$ and thus $\mu^\star > \frac{1}{2} - \frac{10^{-6}}{2}$. Since $\mu^\star \leq 1/2$, this implies $\mathsf{D}^\star = \frac{1-10^{-6}}{4\mu} + \mu > \sqrt{1-10^{-6}} > 1 - 10^{-6}$, and thus $\Phi^\star = 1 - \mathsf{D}^\star < 10^{-6}$.

Case I: G has a small threshold rank, *i.e.*, $\mathrm{rank}_{1-10^{-6}}(G) < n^{10^{-1}}$

The following result, restated below under the assumption of this case in our terminologies after instantiation of parameters with specific values and trivial algebraic simplification, was proved by Arora, Barak and Steurer in [1] in the bigger context of obtaining sub-exponential algorithms for unique games in PCP theory.

Theorem 2.2. [1][3] *There exists a* $\left(\exp\left(n^{10^{-1}}\right)\mathrm{poly}(n)\right)$*-time algorithm that outputs a subset* $\emptyset \subset S \subset V$ *such that* $0.92\,|\,S^\star| \leq |\,S| \leq 1.08\,|\,S^\star|$*, and* $\Phi(S) \leq \Phi(S^\star) + 0.08$*.*

We run the algorithm in Theorem 2.2, and return $\{\, S, \overline{S}\,\}$ as our solution. Note that:

$$\Phi(S) \leq \Phi^\star + 0.08 < 0.080001 \implies \mathsf{D}(S) > 1 - 0.080001 = 0.919999$$
$$0.92\mu^\star \leq \mu(S) \leq 1.08\mu^\star \implies 0.4599 \leq \mu(S) \leq 0.54$$

and thus

$$\begin{aligned} f(\mu(S), \mathsf{D}(S)) &= 2 \times \mu(S) \times \big(\mathsf{D}(S) - \mu(S)\big) \\ &> 2 \times 0.4599 \times \big(0.919999 - 0.54\big) > 10^{-6} \end{aligned}$$

Case II: Remaining Case, *i.e.*, $\mathrm{rank}_{1-10^{-6}}(G) \geq n^{10^{-1}}$

The following result, restated below in our terminologies after instantiation of parameters with specific values, was again proved in [1].

Theorem 2.3. [1][4] *Let H be a regular graph of r nodes with* $\mathrm{rank}_{1-10^{-5}}(H) \geq r^{10^{-1}}$*. Then, there is an algorithm that*

- *runs in* $\mathrm{poly}(r)$ *time, and*

[3] Instantiate Theorem 2.2 in [1] with $\eta = 10^{-4}$ and $\varepsilon = 10^{-6}$.
[4] Instantiate Theorem 2.3 in [1] with $\eta = 10^{-4}$ and $\gamma = 10^{-1}$.

- *finds a subset S of nodes of H with $|S| \leq r^{1-10^{-3}}$ and $\Phi(S) \leq 10^{-2}$.*

Our strategy is to use the algorithm in Theorem 2.3 *repeatedly*[5] to extract "high-rank parts" from G. Namely, we compute in polynomial time an ordered partition of nodes $\left(T_1, T_2, \ldots, T_k, V \setminus \cup_{i=1}^{k} T_i\right)$ such that each T_i is obtained by using the algorithm in Theorem 2.3 on graph G_i induced by the set of nodes $V \setminus \cup_{j=1}^{i-1} T_i$, and the last (possibly empty) graph G'' induced by the set of nodes $V'' = V \setminus \cup_{i=1}^{k} T_i$ satisfy $\text{rank}_{1-10^{-6}}(G'') < |V''|^{10^{-1}}$. Let G' be the graph induced by the set of nodes $V' = \cup_{i=1}^{k} T_i$.

Case II(a) $\left|S^\star \cap V''\right| \geq {|S^*|}/{2}$.

Let S_1^* be the set containing an arbitrary ${|S^*|}/{2}$ elements from the set $S^\star \cap V''$. Note that $\mu\left(S_1^*\right) = {\mu^*}/{2}$ and $\Phi\left(S_1^*\right) \leq 2\,\Phi^*$. We now use Theorem 2.2 on the graph G'' with $|S^*|$ replaced by ${|S^*|}/{2}$ to output a set $S \subseteq V''$ of nodes such that

$$\Phi(S) \leq 2\,\Phi^\star + 0.08 < 0.080002 \implies \mathsf{D}(S) > 1 - 0.080002 = 0.919998$$
$$0.46\mu^\star \leq \mu(S) \leq 0.54\mu^\star \implies 0.229 < \mu(S) \leq 0.27$$

and thus

$$\begin{aligned} f(\mu(S), \mathsf{D}(S)) &= 2 \times \mu(S) \times \big(\mathsf{D}(S) - \mu(S)\big) \\ &> 2 \times 0.229 \times (0.919998 - 0.27) > 10^{-6} \end{aligned}$$

Case II(b) $\left|S^\star \cap V''\right| < {|S^*|}/{2}$.

Since $|S^*| \geq \left(\frac{1}{2} - \frac{10^{-6}}{2}\right) n$ and $|T_j| \leq \left|V \setminus \cup_{\ell=1}^{j-1} T_\ell\right|^{1-10^{-3}} < n^{1-10^{-3}}$ for any j, there exists an index i such that $\frac{|S^*|}{2} - n^{1-10^{-3}} < \left|\cup_{j=1}^{i} T_j\right| < \frac{|S^*|}{2} + n^{1-10^{-3}}$. Notice that the graph induced by the set of nodes $S = \cup_{j=1}^{i} T_j$ satisfy $\Phi(S) \leq 10^{-2}$ and, since $\left(0.5 - \frac{10^{-6}}{2}\right) n \leq |S^*| \leq n$, we have

$$\frac{|S^*|}{2} - n^{1-10^{-3}} < |S| = \left|\cup_{j=1}^{i} T_j\right| < \frac{|S^*|}{2} + n^{1-10^{-3}} \implies 0.24 < \mu(S) < 0.51$$

[5] [1] points out how to "re-regularize" the remaining graph each time a set of nodes have been extracted by adding appropriate number of self-loops of weight $1/2$.

$$\text{and thus,} \quad f(\mu(S), \mathsf{D}(S)) = 2 \times \mu(S) \times (\mathsf{D}(S) - \mu(S)) > 2 \times 0.24 \times (0.99 - 0.51) > 10^{-6}$$

□

Further Research

An interesting open question is whether it is possible to prove the converse of Theorem 2.1, *i.e.*, can we use a sub-exponential approximation algorithm for modularity maximization to design a sub-exponential algorithm for small-set expansion problems ? If possible, this may lead to an alternate interpretation of unique games via communities in social networks.

Acknowledgements We thank Mario Szegedy for pointing out reference [1].

References

[1] S. Arora, B. Barak and D. Steurer. *Subexponential Algorithms for Unique Games and Related Problems*, 51st annual IEEE Symposium on Foundations of Computer Science, 563-572, 2010.

[2] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski and D. Wagner, *On Modularity Clustering*, IEEE Transaction on Knowledge and Data Engineering, 20 (2), 172-188, 2007.

[3] B. DasGupta and D. Desai. *Complexity of Newman's Community Finding Approach for Social Networks*, to appear in Journal of Computer & System Sciences.

[4] E. A. Leicht and M. E. J. Newman, *Community Structure in Directed Networks*, Physical Review Letters, 100, 118703, 2008.

[5] M. E. J. Newman, *Modularity and community structure in networks*, PNAS, 103 (23), 8577-8582, 2006.

[6] M. E. J. Newman and M. Girvan, *Finding and evaluating community structure in networks*, Physical Review E, 69, 026113, 2004.

[7] R. Guimer'a, M. Sales-Pardo and L. A. N. Amaral. *Classes of complex networks defined by role-to-role connectivity profiles*, Nature Physics, 3, 63-69, 2007.

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