THEORY OF COMPUTING
www.theoryofcomputing.org

A note on the hardness of approximating the K-WAY HYPERGRAPH CUT problem

Chandra Chekuri* Shi Li†

June 18, 2020

Abstract: We consider the approximability of K-WAY HYPERGRAPH CUT problem: the input is an edge-weighted hypergraph $G = (V, \mathcal{E})$ and an integer k and the goal is to remove a min-weight subset of the edges such that the residual graph has at least k connected components. When G is a graph this problem admits a $2(1-1/k)$-approximation [Saran and Vazirani, SIAM J. Comput. 1995]. However, there has been no non-trivial approximation ratio for general hypergraphs. In this note we show, via a very simple reduction, that an α-approximation for K-WAY HYPERGRAPH CUT implies an α^2-approximation for the DENSEST K-SUBGRAPH problem. Our reduction combined with the hardness result of [Manurangsi STOC'17] implies that under the Exponential Time Hypothesis (ETH), there is no $n^{1/(\log\log n)^c}$-approximation for K-WAY HYPERGRAPH CUT where $c > 0$ is a universal constant and n is the number of nodes.

K-WAY HYPERGRAPH CUT is a special case of k-WAY SUBMODULAR MULTIWAY PARTITION and hence our hardness applies to this latter problem as well. These hardness results are in contrast to a 2-approximation for closely related problems where the goal is to separate k given terminals [Chekuri and Ene, FOCS'11], [Ene et al. SODA'13].

*Supported in part by NSF grants CCF-1526799 and CCF-1907937.
†Supported in part by NSF grants CCF-1566356, CCF-1717134 and CCF-1844890.

ACM Classification: F.2.2, G.2.0

AMS Classification: 68Q17

Key words and phrases: Hardness of Approximation, k-Way Hypergraph-Cut

1 Introduction

We consider the following problem.

K-WAY HYPERGRAPH CUT: Let $G = (V, \mathcal{E})$ be a hypergraph with edge weights given by $w : \mathcal{E} \to \mathbb{R}_+$. Given an integer k, find a min-weight subset of edges $\mathcal{E}' \subseteq \mathcal{E}$ such that $G - \mathcal{E}'$ has at least k connected components. Equivalently find a partition of V into k non-empty sets $V_1, V_2, \ldots, V_k$ such that the weight of the hyperedges that cross the partition[1] is minimized.

K-WAY HYPERGRAPH CUT is known as the K-CUT problem when the input is a graph and is one of the well-studied variants of graph partitioning problems. K-WAY HYPERGRAPH CUT is a special case of a more general submodular partitioning problem defined below.

k-WAY SUBMODULAR PARTITION (K-WAY SUB-MP): Let $f : 2^V \to \mathbb{R}_+$ be a non-negative submodular set function[2] over a finite ground set V. The k-way submodular partition problem is to find a partition $V_1, \ldots, V_k$ of V to minimize $\sum_{i=1}^{k} f(V_i)$ under the condition that each part $V_i \neq \emptyset$. An important special case is when f is symmetric and we refer to it as K-WAY SYM-SUB-MP.

We refer the reader to [14, 5] to see why K-WAY HYPERGRAPH CUT is a special case of K-WAY SUB-MP. The K-CUT problem is not only a special case of K-WAY HYPERGRAPH CUT but it is also a special case of K-WAY SYM-SUB-MP. When k is part of the input K-CUT is NP-Hard [8] and hence all the problems we discussed so far are also NP-Hard. K-WAY SYM-SUB-MP admits a $2(1-1/k)$-approximation [15, 19] and hence also K-CUT [17]. For K-WAY HYPERGRAPH CUT a $2\Delta(1-1/k)$-approximation easily follows from the $2(1-1/k)$-approximation for K-CUT; here Δ is the rank of the hypergraph (the maximum size of any hyperedge). On the other hand, in the general case, the known approximation algorithms for K-WAY HYPERGRAPH CUT and K-WAY SUB-MP provide an approximation ratio of $(k-1)$ [19]. Despite a claim of APX-Hardness for K-CUT in [17] (attributed to Papadimitriou), no proof has been published in the literature; Manurangsi [13] showed that K-CUT does not admit a $(2-\varepsilon)$-factor for any fixed $\varepsilon > 0$ under the Small Set Expansion Hypothesis. As far as we are aware, prior to our work, no better hardness result was know for K-WAY HYPERGRAPH CUT.

In this note we show that a good approximation for K-WAY HYPERGRAPH CUT would imply a good approximation for the DENSEST K-SUBGRAPH problem which has been extensively investigated and has been shown to be conditionally hard.

DENSEST K-SUBGRAPH: Given a graph $G = (V, E)$ and integer ℓ, find a subset $S \subseteq V$ of ℓ nodes to maximize the number of edges in the induced graph $G[S]$.

In the preceding definition we used ℓ instead of k to denote the number of nodes in the subgraph to be found. This is to avoid notational confusion since k is used in the K-WAY HYPERGRAPH CUT problem. The current best approximation for DENSEST K-SUBGRAPH is $O(n^{1/4+\varepsilon})$ [1]; note that an ℓ-approximation is easy. Although the problem is expected to be quite hard to approximate, the known hardness results are weak; a PTAS for DENSEST K-SUBGRAPH can be ruled out only under the assumption that NP $\not\subseteq \cap_{\varepsilon>0}$BPTIME$(2^{n^{\varepsilon}})$ [11]. Polynomial-factor integrality gaps for several strong SDP relaxations are known [2]. In a breakthrough result, Manurangsi [12] showed that under the Exponential Time

[1] A hyperedge e crosses a partition of the vertex set if e properly intersects at least two parts of the partition.

[2] A set function $f : 2^V \to \mathbb{R}$ is submodular iff $f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$ for all $A, B \subseteq V$. Moreover, f is symmetric if $f(A) = f(V - A)$ for all $A \subseteq V$.

Hypothesis (ETH), DENSEST K-SUBGRAPH is hard to be approximate to a factor better than $n^{1/(\log\log n)^c}$ where n is the number of nodes in the input graph and $c > 0$ is a universal constant. We state his result more precisely in the theorem below.

Theorem 1.1 ([12])**.** *There exists a constant $c > 0$ such that the following holds, assuming ETH. No polynomial time algorithm can, given a graph G with n vertices and a positive integer $\ell \leq n$, distinguish between the following two cases:*

- *G contains an ℓ-clique as a subgraph.*
- *Every ℓ-node subgraph of G has at most $\binom{\ell}{2}/n^{1/(\log\log n)^c}$ edges.*

To formally state our result it is more convenient to relate the approximation ratio to the parameter s of a given graph (or hyper-graph) which is the sum of the number of nodes and edges (or hyper-edges). The above theorem gives an $s^{1/(\log\log s)^{c'}}$-hardness for DENSEST K-SUBGRAPH for some $c' > 0$, since s and n are polynomially related for graphs. On the other hand, the tight instances for the algorithm of [1] for DENSEST K-SUBGRAPH have $|E| = \Theta(|V|^{3/2})$. For these instances, it is not known how to obtain an approximation ratio better than $O(|V|^{1/4}) = O(s^{1/6})$. Our main theorem is the following:

Theorem 1.2. *A polynomial-time $\alpha(s)$ approximation algorithm for* K-WAY HYPERGRAPH CUT *implies a polynomial-time $\frac{\ell-1}{\ell-2}(\alpha(s+1))^2$-approximation algorithm for* DENSEST K-SUBGRAPH.

Combining the preceding theorem with Theorem 1.1, we obtain the following.

Corollary 1.3. *Assuming ETH, there is no $s^{1/(\log\log s)^c}$-approximation for* K-WAY HYPERGRAPH CUT, *where $c > 0$ is a universal constant.*

The reduction in the proof of Theorem 1.2 creates instances of K-WAY HYPERGRAPH CUT in which s is upper bounded by a fixed polynomial in n where n is the number of nodes of the hypergraph. Therefore, the preceding corollary in fact shows that, assuming ETH, there is no $n^{1/(\log\log n)^c}$-approximation for K-WAY HYPERGRAPH CUT, where $c > 0$ is some universal constant.

When k is a fixed constant one can reduce K-WAY HYPERGRAPH CUT and K-WAY SUB-MP to solving $O(n^{k-1})$ instances of the "terminal" version of these problems which have a $2(1-1/k)$ approximation. We refer the readers to [5, 7] for more details on these related problems.

2 Proof of Theorem 1.2

Let $(G = (V,E), \ell)$ be an instance of DENSEST K-SUBGRAPH. We construct a hypergraph $H = (A, \mathcal{F})$ as follows. For each edge $e \in E$ we create a node a_e and add it to A. Moreover we add a new special node r to A. Thus $A = \{r\} \cup \{a_e \mid e \in E\}$. For each node $v \in V$ we add a hyperedge f_v to $\mathcal{F}$ where $f_v = \{r\} \cup \{a_e \mid e \in \delta_G(v)\}$ where $\delta_G(v)$ is the set of edges in E that are incident to v in G. Thus H is basically the hypergraph obtained from G by flipping the role of nodes and edges and then adding the extra node r to each hyperedge. We also observe that $|A| + |\mathcal{F}| = 1 + |V| + |E| = s + 1$.

For a subset $S \subseteq V$, we let $E_G(S)$ denote the set of edges in E with both endpoints in S. The following is a simple but useful claim about the relationship between G and H.

Claim 2.1. *For any* $1 \le \ell \le |V|$, *if there is set* $S \subseteq V$ *with* $|S| = \ell$ *and* $|E_G(S)| = L-1$ *then the* K-WAY HYPERGRAPH CUT *instance on* H *with* $k = L$ *has a cut of value at most* ℓ. *Moreover, given any* $F \subseteq \mathcal{F}$ *of size* $|F| = \ell'$ *such that* $H - F$ *has* L' *connected components, there is a subset* $S' \subseteq V$ *such that* $|S'| = |F| = \ell'$ *and* $|E_G(S')| = L' - 1$.

Proof. Consider a set $F \subseteq \mathcal{F}$ of hyperedges in H. Suppose we remove them from H. Let $V_F = \{v \in V \mid f_v \in F\}$ be the nodes in G that correspond to the hyperedges in F. Then a node $a_e \in A$ corresponding to an edge $e = uv$ is separated from r in H iff both $u, v \in V_F$; in this case the node a_e becomes an isolated node in $H - F$. Thus the number of connected components in $H - F$ is precisely equal to $|E_G(V_F)| + 1$. This correspondence proves both parts of the claim. □

Suppose we have an $\alpha(s)$ approximation for K-WAY HYPERGRAPH CUT. We will obtain an $\frac{\ell-1}{\ell-2}\alpha(s+1)^2$-approximation for DENSEST K-SUBGRAPH as follows. Let (G, ℓ) be a given instance of DENSEST K-SUBGRAPH. First assume that we know the optimum solution value L for the given instance. We construct the hypergraph H as described and give H and $k = L+1$ to the $\alpha(s)$ approximation algorithm for K-WAY HYPERGRAPH CUT. By Claim 2.1 there is an optimum solution to the K-WAY HYPERGRAPH CUT instance on H of value at most ℓ.

Thus, the approximation algorithm will output a set $F \subseteq \mathcal{F}$ such that $|F| \le \alpha(s+1) \cdot \ell$ such that $H - F$ has at least $L+1$ connected components. By the second part of the claim we can obtain a set $S' \subseteq V$ such that $|S'| \le \alpha \cdot \ell$ and $|E_G(S')| \ge L$. Then we shall output a random subset $S' \subseteq S$ of size ℓ as the solution for the DENSEST K-SUBGRAPH problem. For simplicity, we let $\alpha = \alpha(s+1)$. Then, the expected number of edges induced by S is

$$\begin{aligned} |E_G(S')| \cdot \frac{\ell}{|S'|} \cdot \frac{\ell-1}{|S'|-1} &\ge L \cdot \frac{\ell}{\alpha \cdot \ell} \cdot \frac{\ell-1}{\alpha \cdot \ell - 1} = \frac{L}{\alpha}\left(\frac{1}{\alpha} - \frac{1-1/\alpha}{\alpha\ell - 1}\right) \\ &\ge \frac{L}{\alpha}\left(\frac{1}{\alpha} - \frac{1}{\alpha\ell - \alpha}\right) = \frac{\ell-2}{\ell-1} \cdot \frac{L}{\alpha^2}. \end{aligned}$$

In the above sequence, we used $\alpha \ge 1$ and assumed $\ell \ge 2$. One can indeed efficiently and deterministically find a set $S \subseteq V$ of size ℓ such that $|E_G(S)| \ge \frac{\ell-2}{\ell-1} \cdot \frac{L}{\alpha^2}$, using the method of conditional expectations. This holds since conditioned on the event that S contains a given set of vertices, the expectation of $|E_G(S)|$ can be computed easily. Since L is the optimum value for the given instance of DENSEST K-SUBGRAPH, we obtain the desired $\frac{\ell-1}{\ell-2} \cdot \alpha^2 = \frac{\ell-1}{\ell-2} \cdot (\alpha(s+1))^2$-approximation. The assumption that the algorithm knows the value L can be easily removed by trying all possible values of L from 0 to $|E(G)|$. This finishes the proof of Theorem 1.2.

3 Discussion and open problems

We proved conditional hardness of K-WAY HYPERGRAPH CUT. An important open question is to obtain hardness of approximation for K-WAY HYPERGRAPH CUT under the standard $P \ne NP$ assumption. At this point we do not even have APX-Hardness. For K-WAY SYM-SUB-MP Santiago [16] has shown an exponential lower bound on the number of value oracle queries required to obtain an approximation ratio strictly below 2. Can one show exponential query lower bounds for K-WAY SUB-MP even for super-constant approximation factors? This question was raised in [16] based on a preliminary version of this paper.

For any fixed constant k, K-CUT in graphs can be solved in polynomial time [8]; there are several different algorithms for this problem by now and we refer the reader to [6, 10] for a discussion of recent work and other pointers. It was an open problem whether K-WAY HYPERGRAPH CUT can be solved in polynomial time when k is a fixed constant. Recently a randomized polynomial algorithm was developed in [4], and very recently a deterministic algorithm is claimed in [3]. The complexity status of K-WAY SUB-MP is open for any fixed $k > 3$; for $k \leq 3$ there is a polynomial time algorithm [14] building upon [18]. For K-WAY SYM-SUB-MP a polynomial-time algorithm is known for $k \leq 4$ [9] and the complexity status is open for any fixed $k > 4$.

References

[1] ADITYA BHASKARA, MOSES CHARIKAR, EDEN CHLAMTAC, URIEL FEIGE, AND ARAVINDAN VIJAYARAGHAVAN: Detecting high log-densities: An $O(n^{1/4})$ approximation for densest k-subgraph. In *Proceedings of the Forty-second ACM Symposium on Theory of Computing*, STOC '10, pp. 201–210, New York, NY, USA, 2010. ACM. 2, 3

[2] ADITYA BHASKARA, MOSES CHARIKAR, ARAVINDAN GURUSWAMI, VENKATESAN VIJAYARAGHAVAN, AND YUAN ZHOU: Polynomial integrality gaps for strong sdp relaxations of densest k-subgraph. In *Proceedings of the Twenty Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 388–405. SIAM, 2012. 2

[3] KARTHEKEYAN CHANDRASEKARAN AND CHANDRA CHEKURI: Hypergraph k-cut for fixed k in deterministic polynomial time. Manuscript under submission, April 2020. 5

[4] KARTHEKEYAN CHANDRASEKARAN, CHAO XU, AND XILIN YU: Hypergraph k-cut in randomized polynomial time. *Mathematical Programming*, pp. 1–29, 2019. Preliminary version in Proc. of ACM-SIAM SODA 2018. 5

[5] CHANDRA CHEKURI AND ALINA ENE: Approximation algorithms for submodular multiway partition. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pp. 807–816. IEEE, 2011. 2, 3

[6] CHANDRA CHEKURI, KENT QUANRUD, AND CHAO XU: LP relaxation and tree packing for minimum k-cut. *SIAM J. on Discrete Mathematics*, 32:1334–1353, 2020. 5

[7] ALINA ENE, JAN VONDRÁK, AND YI WU: Local distribution and the symmetry gap: Approximability of multiway partitioning problems. In *Proc. of ACM-SIAM SODA*, pp. 306–325, 2013. 3

[8] O. GOLDSCHMIDT AND D.S. HOCHBAUM: A polynomial algorithm for the k-cut problem for fixed k. *Mathematics of Operations Research*, pp. 24–37, 1994. 2, 5

[9] F. GUINEZ AND M. QUEYRANNE: The size-constrained submodular k-partition problem. Unpublished manuscript. Available at https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxmbGF2aW9ndWluZXpob21lcGFnZXxneDo0NDVlMThkMDg4ZWRlOGI1. See also https://smartech.gatech.edu/bitstream/handle/1853/43309/Queyranne.pdf, 2012. 5

[10] ANUPAM GUPTA, DAVID G. HARRIS, EUIWOONG LEE, AND JASON LI: Optimal bounds for the *k*-cut problem, 2020. ArXiv, 2005.08301. 5

[11] SUBHASH KHOT: Ruling out ptas for graph min-bisection, dense *k*-subgraph, and bipartite clique. *SIAM J. Comput.*, 36(4):1025–1071, December 2006. 2

[12] PASIN MANURANGSI: Almost-polynomial ratio eth-hardness of approximating densest k-subgraph. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pp. 954–961, 2017. 2, 3

[13] PASIN MANURANGSI: Inapproximability of maximum biclique problems, minimum k-cut and densest at-least-k-subgraph from the small set expansion hypothesis. *Algorithms*, 11(1):10, Jan 2018. 2

[14] K. OKUMOTO, T. FUKUNAGA, AND H. NAGAMOCHI: Divide-and-conquer algorithms for partitioning hypergraphs and submodular systems. *Algorithmica*, 62(3):787–806, 2012. 2, 5

[15] M. QUEYRANNE: On optimum size-constrained set partitions. Combinatorial Optimization Workshop, AUSSOIS, 1999. 2

[16] RICHARD SANTIAGO: *Multi-Agent Submodular Optimization: Variations and Generalizations*. Ph. D. thesis, McGill University, 2019. 4, 5

[17] HUZUR SARAN AND VIJAY V. VAZIRANI: Finding *k* cuts within twice the optimal. *SIAM J. Comput.*, 24(1):101–108, February 1995. 2

[18] MINGYU XIAO: Finding minimum 3-way cuts in hypergraphs. *Information Processing Letters*, 110(14-15):554–558, 2010. Preliminary version in TAMC 2008. 5

[19] L. ZHAO, H. NAGAMOCHI, AND T. IBARAKI: Greedy splitting algorithms for approximating multiway partition problems. *Mathematical Programming*, 102(1):167–183, 2005. 2

AUTHORS

Chandra Chekuri
Professor
Department of Computer Science
University of Illinois
Urbana, IL 61801
chekuri [at] illinois [dot] edu
http://chekuri.cs.illinois.edu

Shi Li
Assistant Professor
Department of Computer Science and Engineering
University at Buffalo
Buffalo, NY 14260
shil [at] buffalo [dot] edu
https://www.cse.buffalo.edu/~shil/

ABOUT THE AUTHORS

CHANDRA CHEKURI received his B. Tech in Computer Science and Engineering from Indian Institute of Technology, Madras (now Chennai) in 1993 and received his PhD in Computer Science from Stanford University in 1998 under the supervision of the late Rajeev Motwani. After finishing his PhD he worked eight formative years at Lucent Bell Labs before moving to University of Illinois. He is interested in many topics but mainly works in algorithm design, combinatorial optimization, graphs, and mathematical programming. He has very much enjoyed teaching since he was in middle school. His dreams of running a marathon and to learn a musical instrument have not yet come to fruition in spite of lack of effort on his part. His childhold was spent mostly in the town of Rajahmundry on the banks of the beautiful river Godavari where his parents still live and his children like to visit.

SHI LI received his PhD in 2013 from the Department of Computer Science at Princeton University, under the supervision of Moses Charikar. From 2013 to 2015, he was a research assistant professor at Toyota Technological Institute at Chicago. His research interest is algorithm design.