On the Approximability of the Sum-Max Graph Partitioning Problem

Rémi Watrigant, Marin Bougeret, Rodolphe Giroudeau, Jean-Claude König

On the approximability of the Sum-Max graph partitioning problem*

R. Watrigant, M. Bougeret, R. Giroudeau, and J.-C. König

LIRMM-CNRS-UMR 5506-161, rue Ada 34090 Montpellier

Abstract In this paper we consider the classical combinatorial optimization graph partitioning problem with Sum-Max as objective function. Given a weighted graph $G = (V, E)$ and a integer k, our objective is to find a k-partition $(V_1, \dots, V_k)$ of V that minimizes $\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \max_{u \in V_i, v \in V_j} w(u, v)$, where $w(u, v)$ denotes the weight of the edge $\{u, v\} \in E$. We prove, in addition to the $\mathcal{NP}$ and $W[1]$ hardnesses (for the parameter k), that there is no ρ-approximation algorithm for any $\rho \in O(n^{1-\epsilon})$, given any fixed $0 < \epsilon \leq 1$ (unless $\mathcal{P} = \mathcal{NP}$), improving the previous $1 + \frac{1}{k}$ lower bound of [5]. Lastly, we present a natural greedy algorithm with an approximation ratio better than $\frac{k}{2}$.

1 Introduction

Graph partitioning problems are classical combinatorial optimization problems, where the objective is to partition vertices of a given graph into k clusters, according to one or several criteria. In this article we focus on minimizing the sum of the largest edge between each pair of clusters. More formally, we study the following optimization problem:

SUM-MAX GRAPH PARTITIONING
Input: a graph $G = (V, E)$, $w : E \to \mathbb{N}$, $k \leq |V|$
Output: a k-partition $(V_1, ..., V_k)$ of V
Goal: minimize $\displaystyle\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \max_{\substack{u \in V_i \\ v \in V_j}} w(u, v)$

All graphs studied here are supposed to be simple, non oriented and connected, unless otherwise stated. We also consider the unweighted version of this problem, U-SUM-MAX GRAPH PARTITIONING, where $w(e) = 1$ for all $e \in E$.

This article is organized as follows: the next section is devoted to computational lower bounds, while Section 3 is devoted to approximability.

2 Computational lower bounds

Theorem 1. U-SUM-MAX GRAPH PARTITIONING *(and thus the weighted version) is $\mathcal{NP}$-hard, and cannot be approximated within a $O(n^{1-\epsilon})$ factor for any fixed $0 < \epsilon \leq 1$, unless $\mathcal{P} = \mathcal{NP}$.*

Proof. We show that a $\rho(n')$-approximation algorithm for U-SUM-MAX GRAPH PARTITIONING implies a $(1 + 2\rho(n + 1))$-approximation for the INDEPENDENT SET, where n' (resp. n) is the number of vertices in the input graph of U-SUM-MAX GRAPH PARTITIONING (resp. INDEPENDENT SET). Thus,

* This work has been funded by grant ANR **2010** BLAN **021902**

any $O(n'^{(1-\epsilon)})$-approximation (with $0 < \epsilon \leq 1$) for U-SUM-MAX GRAPH PARTITIONING would imply a $O(n^{1-\epsilon})$ for INDEPENDENT SET, which is known to be impossible unless $\mathcal{P} = \mathcal{NP}$ [3].

Let I be an instance of INDEPENDENT SET, composed of a graph $G = (V, E)$ with $|V| = n$ vertices and $|E| = m$ edges. For any $k \leq n$, we build the following instance I'_k of U-SUM-MAX GRAPH PARTITIONING: I'_k is composed of a graph $G' = (V', E')$ which is a copy of G plus an universal vertex ω. Notice that I'_k can be computed in polynomial time, and recall that the problem ask for a k-partition of minimum cost in G'. It is easy to verify the following claims:

claim 1 : A minimum independent set in G (of size $\alpha(G)$) implies an $(\alpha(G)+1)$-partition in G' of cost $\alpha(G)$ (each vertex of the independent set forms a cluster, and all remaining ones form the latest), and conversely (because of the universal vertex).

claim 2 : Given a k-partition P_k of G', one can define the graph G_{P_k} where vertices are clusters, and two vertices are adjacent if and only if the two clusters are adjacent. We denote by m_k the number of edges of G_{P_k}. An independent set of size C in G_{P_k} corresponds to an independent set of the same size in G (the cluster containing ω cannot be part of any independent set since it is an universal vertex).

claim 3 : Given a graph with n vertices and m edges, it is possible to compute an independent set of size $\frac{n^2}{2m+n}$, using a derandomized version of the Turan's theorem [4].

Suppose that $\mathcal{A}$ is a $\rho(n')$-approximation algorithm for U-SUM-MAX GRAPH PARTITIONING, and let us build the following algorithm for INDEPENDENT SET, given the instance I defined previously.

Algorithm 1 algorithm for INDEPENDENT SET

for $k = 1$ to n **do**
  build the instance I'_k as previously
  let P_k be the partition of cost m_k returned by Algorithm $\mathcal{A}$ on I'_k
  use claim 3 to obtain an independent set in G_{P_k} (and thus in G by claim 2) of size $x_k = \frac{k^2}{2m_k+k}$
end for
return the independent set of maximum size among all independent sets built in the algorithm.

Analysis of Algorithm 1: Since $\alpha(G) \in \{1, ..., n\}$, there is a step k_0 such that $k_0 = \alpha(G) + 1$. Let $\mathcal{B}$ be the size of the independent set returned by Algorithm 1. By construction, we have:

$$\mathcal{B} \geq \frac{k_0^2}{2m_{k_0} + k_0} = \frac{(\alpha(G)+1)^2}{2m_{k_0} + (\alpha(G)+1)}$$

Since $\mathcal{A}$ return a $\rho(n') = \rho(n+1)$-approximated solution (G' contains $n+1$ vertices), we have $m_{k_0} \leq \rho(n+1) m^*_{k_0}$, where $m^*_{k_0}$ is the value of an optimal solution for I'_{k_0}. As claim 1 implies that $m^*_{k_0} = \alpha(G)$, we get:

$$\mathcal{B} \geq \frac{(\alpha(G)+1)^2}{2\rho(n+1)\alpha(G) + \alpha(G) + 1} \geq \frac{1}{1 + 2\rho(n+1)} \alpha(G)$$

Which concludes the proof. □

The reduction given above can also be used to construct an *FPT reduction* from INDEPENDENT SET parameterized by k (which is a known $W[1]$-complete problem) to U-SUM-MAX GRAPH PARTITIONING parameterized by the number of clusters. Thus, we deduce the following proposition:

Proposition 1. SUM-MAX GRAPH PARTITIONING *(and its unweighted version) parameterized by the number of clusters is* $W[1]$*-hard.*

3 A polynomial-time approximation algorithm

In this section we consider the simple greedy algorithm for SUM-MAX GRAPH PARTITIONING given by Algorithm 2. Actually, this algorithm corresponds to the SPLIT algorithm of [2], which gives a $(2-2/k)$-approximation algorithm for MIN-K-CUT.

Algorithm 2 a greedy algorithm for SUM-MAX GRAPH PARTITIONING

```
Sort E in non decreasing order of weights (ties are broken arbitrarily)
j ← 0
for i = 1 to k − 1 do
  while G has i connected components do
    G ← G\{e_j}
    j ← j + 1
  end while
end for
return connected components of G
```

Theorem 2. *Let* $\theta = \max\{\frac{w(e)}{w(e')} : e, e' \in E, e \neq e', w(e') \geq w(e)\}$. *Then, Algorithm 2 is a* $(1+(\frac{k}{2}-1)\theta)$*-approximation algorithm.*

A complete proof of Theorem 2 can be found in [5].

Remark 1. The ratio of Theorem 2 is tight, for graphs such that Algorithm 2 builds a clique over the clusters (*i.e.* all clusters are adjacent), while an optimal solution is a tree.

4 Conclusion

We investigated a variant of the classical graph partitioning problem with sum-max as objective function, and established some computational and approximability results. Especially, the approximation lower bound indicates that the $\frac{k}{2}$-approximation algorithm of [5] is somehow optimal. Thus, it should encourage other investigations in the development of a $O(n^{f(k)})$ exact algorithm, like for the MIN-K-CUT problem [1], or on the contrary an $\mathcal{NP}$-completeness result for some fixed k.

References

1. O Goldschmidt and D. S. Hochbaum. Polynomial algorithm for the k-cut problem. In *FOCS*, pages 444–451, 1988.
2. H. Saran and V. V. Vazirani. Finding k cuts within twice the optimal. *Journal of Computing*, 24:101–108, 1995.
3. L. Trevisan. Inapproximability of combinatorial optimization problems. *CoRR*, cs.CC/0409043, 2004.
4. P. Turán. An extremal problem in graph theory. *Mat Fiz Lapok*, pages 436–452, 1941.
5. R. Watrigant, M. Bougeret, R. Giroudeau, and J.-C. König. Sum-max graph partitioning problem. submitted to ISCO 2012.