International Journal of Management Science
Vol 14, No 1, May 2008

Inapproximability of the Max-cut Problem with Negative Weights

Sung-Pil Hong*

Department of Industrial Engineering, Seoul National University, Korea,

(Received: April 2, 2008 / Revised: April 16, 2008 / Accepted: April 16, 2008)

ABSTRACT

We show that when a max-cut problem is allowed native-weight edges, to decide if the problem has a cut of a positive weight is NP-hard. This implies that there is no polynomial time algorithm which guarantees a cut whose objective value is no less than $1/p(\langle I \rangle)$ times the optimum for any polynomially computable polynomial p, where $\langle I \rangle$ denotes the encoding length of an instance I.

Keyword: Maximum Cut, Negative Weights, NP-hard, Sparsest Cut

1. Introduction

This short note was motivated by [3] which investigated polynomially solvable cases of the maximum cut problem, or Max-Cut characterized by the signs of the objective coefficients. They extended the problem to have negative weights.

Problem 1.1 Given an undirected graph $G = (V, E)$, and the nonnegative edge weights w_e, $e \subseteq E$, find a cut, $(W; V \setminus W)$ of V of the maximum sum of weights of edges in the cut.

When only nonnegative edges are allowed, Max-Cut is a well-known NP-hard problem that has been intensively studied in various contexts.

Their observation begins with that Max-Cut is equivalent to the minimum cut problem, by negative weights, which is polynomially solvable via the maximum flow

* E- mail: sphong@snu.ac.kr

problem. If the number of positive edges are small enough, [3] showed that the polynomiality is maintained. When the subgraph induced by positive edges has a node cover whose size is $O(\log n^k)$, then Max-Cut is solvable in polynomial time. If, on the other hand, the minimum cover size is $\Omega(n^{1/k})$, then, they also showed, Max-Cut is strongly NP-hard.

This note focuses on the negative edges. Once the number of positive edges is large enough for the problem to be NP-hard then how the negative edges affect the intractability? Do they mitigate it or worsen it? The meaning of the question lies in that Max-Cut with positive weights only is readily approximable. We can easily construct a cut whose weight is at least the half of the optimum (See, e.g. [4]). In their seminal work, Goemans and Williamson [1] developed a randomized algorithm that guarantees 0.878-approximation.

In this paper, we show that when negative weights are allowed no approximation is possible for Max-Cut under the premise, $P \neq NP$. For any polynomially computable function $p(\langle I \rangle)$ of the input size, $\langle I \rangle$ of a Max-Cut instance, I, guaranteeing a solution whose objective value is $1/p(\langle I \rangle)$ times the optimum is impossible in polynomial time.

2. Inapproximability

We now show that the sparsest cut problem is polynomially reducible to Max-Cut with negative weights.

Problem 2.1: Sparsest cut problem [2]: Suppose $G=(V,E)$ is an undirected graph with nonnegative edge capacities c_e. Consider a set $S(\subseteq V \times V)$ of k pairs of nodes, $\{(s_1,t_1),\cdots,(s_k,t_k)\}$. For each pair (s_l,t_l), we assign a demand d_l. For a cut $(W; V \setminus W)$, let $d(W)$ be the total demand separated by the cut: $d(W)=\sum_{l:|\{s_l,t_l\}\cap W|=1} d_l$. Also, denote, by $c(W)$, the capacity of edges in the cut $(W; V \setminus W)$. Find a $W \subseteq V$ such that the ratio, $d(W)/c(W)$ is maximized.

Consider the binary search query to solve Problem 2.1:

$$\text{Given } \lambda \in \mathbb{Q}_+, \quad \text{is there a cut } (W; V \setminus W) \text{ such that } \frac{d(W)}{c(W)} > \lambda \text{ ?} \qquad (1)$$

Notice that we can make all the data, c and d, integral while keeping the problem size polynomially bounded. Then, by typical arguments, in $\log(k\,|\,E\,|\,c_{\max}d_{\max})$ queries with $\lambda \in [\,1/|\,E\,|\,c_{\max}, kd_{\max}\,]$, we can find the optimum of Problem 2.1. Here, $c_{\max}(d_{\max})$ is the maximum value of an edge cost (a demand, respectively).

The answer to the query is "yes" if and only if the optimal value of the following problem is positive:

$$\max_{W \subseteq V}\{d(W) - \lambda c(W)\}\,. \qquad (2)$$

We now reduce (2) to Max-Cut with negative edges. In doing so, for simplicity, we will use c_{ij} instead of λc_{ij}. For the construction of Max-Cut instance, we use the same graph $G = (V, E)$ from the given sparsest cut problem. Assign the negative weight, $w_{ij} = -c_{ij}$ to every edge $ij \subseteq E$. For each pair, (s_l, t_l) from S, create the edge s_lt_l if it is not in the original graph. Assign the positive weight $w_{s_lt_l} = d_l$ to the edge s_lt_l, $l = 1, 2, \cdots k$. If the edge is original and hence already assigned the negative weight $-c_{s_lt_l}$, then we add d_l to it. The final weight is then $w_{s_lt_l} = d_l - c_{s_lt_l}$.

It is easy to see that the Max-Cut instance, I, constructed above is equivalent to (2). Hence, the answer to the query, (1) is "yes" if and only if I has a proper cut $(W; V \setminus W)$ $(W \neq V)$ having positive objective value. This implies if there is a polynomial algorithm which guarantees a cut whose objective value is no less than $1/p(\langle I \rangle)$ times the optimum, we can verify in a polynomial time whether the answer to the query (2) is "yes" or "no". Hence, we can solve the sparsest cut problem polynomially, which is impossible under the premise NP $\neq$ P.

3. Open problem

We have shown that when the number of the negative edges is arbitrary, Max-Cut is not approximable within any factor. It will be interesting so see, when the size of negative edges decreases, how the approximability changes accordingly.

References

[1] Goemans, M. X. and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. ACM* 42 (1995) 1115-1145.

[2] Matula, D. W. and F. Shahrokhi, Sparsest cuts and bottleneck in graphs, Discrete Appl. Math., 27 (1990) 113-123.

[3] McCormick, S. T., M. R. Rao, and G. Rinaldi, Easy and difficult objective functions for max cut, Math. Program, Ser. B 94 (2003) 459-466.

[4] Vazirani, V. V., Approximation algorithms, Springer, Berlin, 2001.