УДК 510.52

O. V. Verbitsky

A NOTE ON THE APPROXIMABILITY OF THE DENSE SUBGRAPH PROBLEM

O. V. Verbitsky. *A note on the approximability of the dense subgraph problem*, Matematychni Studii, **22** (2004) 198–201.

The Dense Subgraph Problem is, given a graph G and an integer a, to determine the maximum number of edges in a subgraph of G induced on a vertices. This optimization problem is well known to be NP-hard. We prove that finding an approximate solution with an absolute error guarantee of $a^{2-\epsilon}$ is still NP-hard for each $\epsilon > 0$.

О. В. Вербицкий. *Замечание об аппроксимируемости задачи о плотном подграфе* // Математичні Студії. – 2004. – Т.22, №2. – С.198–201.

Задача о плотном подграфе заключается в определении, по заданному графу G и целому числу a максимального количества ребер в подграфе индуцированном графом G на некотором множестве a вершин. Хорошо известно, что эта оптимизационная задача является NP-трудной. Мы доказываем, что нахождение приближенного решения с абсолютной погрешностью $a^{2-\epsilon}$ есть все еще NP-трудной задачей для каждого $\epsilon > 0$.

As usually, we call the number of vertices and the number of edges of a graph by its *order* and *size* respectively. We denote the vertex set and the edge set of a graph G, respectively, by $V(G)$ and $E(G)$. Given graphs G and S, we call S a *subgraph* of G if $V(S) \subseteq V(G)$ and $E(S) \subseteq E(G)$. Given graphs G and H, let $\sigma(G, H)$ denote the maximum size of a graph S which is isomorphic to a subgraph of G and to a subgraph of H. We use the standard notation K_a for the complete graph on a vertices and $K_{a,b}$ for the complete bipartite graph with vertex classes of a and b vertices. Recall that the *clique number* of a graph G, denoted by $\omega(G)$, is equal to the maximum a such that G contains a subgraph K_a.

The Largest Common Subgraph is an optimization problem of computing $\sigma(G, H)$ for two given graphs. A few other well-known optimization problems are naturally presentable as specifications of Largest Common Subgraph:

Max Cut: Given G of order n, determine $\sigma(G, K_{n-1,n-1})$.

Max Dense Subgraph (also called Max Edge Subgraph): Given G and a, determine $\sigma(G, K_a)$.

Max Bisection: Given G of even order n, determine $\sigma(G, K_{n/2,n/2})$.

All these problems are classical examples of NP-hard optimization problems [7]. Since the existence of an efficient algorithm for an NP-hard problem seems unplausible, it is reasonable to try to solve such a problem at least approximately. Let π be the objective function of

2000 *Mathematics Subject Classification:* 68Q25, 68R10.

an optimization problem. We assume that π, whose domain is the space of instances of the problem, takes on non-negative integer values. Let c be a function mapping the space of instances to the set of real numbers strictly larger than 1. In particular, c can be constant. We say that an algorithm A *solves* (or *approximates*) *the problem with an approximation ratio of* c if on every input I the algorithm outputs a number $A(I)$ such that

$$\pi(I) \le A(I) \le c(I) \cdot \pi(I).$$

There are polynomial-time algorithms for MAX CUT with approximation ratio of 1.139 [8] and for MAX BISECTION with approximation ratio of 1.431 [6, 12]. For MAX DENSE SUBGRAPH an efficient approximation with ratio of $O(n^\delta)$ for some $\delta < 1/3$ is obtained in [5]. Moreover, if a is linearly related to n, the approximation ratio becomes constant. On the other hand, approximating MAX CUT within ratio of $17/16 - \epsilon$ is NP-hard for each $\epsilon > 0$ [10].

Despite an essential progress in getting inapproximability results (see [3]), it is unknown whether LARGEST COMMON SUBGRAPH is hard to approximate with every constant ratio and whether MAX DENSE SUBGRAPH or MAX BISECTION is hard to approximate with some ratio. A possible explanation of this situation is given in [9, Remark 7]. It is therefore reasonable to confine oneself to proving inapproximability results for the latter problems with a stronger concept of an approximate solution. As above, let π be the objective function of an optimization problem and let e be a function of an input taking on non-negative real values. We say that an algorithm A *solves* (or *approximates*) *the problem with an absolute error guarantee of* e if on every input I the algorithm outputs a number $A(I)$ such that

$$\pi(I) \le A(I) \le \pi(I) + e(I).$$

The question if a problem can be efficiently solved by such an algorithm is quite sensible: There are NP-hard problems approximable with constant absolute error guarantee of 1. An example is given by the problem of computing the edge-chromatic number of a graph, that is, the minimum number of colors that can be assigned to edges of the graph so that adjacent edges are colored differently. While this graph invariant is either equal to or one greater than the maximum vertex degree [1], its computation is NP-hard [11].

In many cases the NP-hardness of finding an approximate solution with an absolute error guarantee follows from the NP-hardness of finding a precise solution by a standard *padding argument*. As an example, we apply this argument to LARGEST COMMON SUBGRAPH.

Proposition 1. *Given graphs G and H, we will denote their orders by n and m respectively and suppose that $n \ge m$. Let ϵ be an arbitrary positive real. Then, unless NP=P, no polynomial-time algorithm approximates $\sigma(G, H)$ with absolute error guarantee of $m^{1-\epsilon}$.*

Proof. Assuming that such an algorithm A exists, we show how, given a graph G and a number a, to efficiently recognize if $\omega(G) \ge a$. This will give the claim as the latter problem is NP-complete. Let sG denote the graph consisting of s vertex-disjoint copies of G. If $\omega(G) \ge a$, we have $\sigma(sG, sK_a) = s\binom{a}{2}$, whereas if $\omega(G) < a$, we have $\sigma(sG, sK_a) \le s\binom{a}{2} - s$. We can distinguish between the two cases by setting $s = a^{\lceil 1/\epsilon \rceil}$ and running A on input (sG, sK_a). If $\omega(G) \ge a$, we obtain $A(sG, sK_a) \ge s\binom{a}{2}$. If $\omega(G) < a$, we obtain $A(sG, sK_a) \le (s\binom{a}{2} - s) + (sa)^{1-\epsilon} < s\binom{a}{2}$. □

A disadvantage of this argument is that, using it, we apparently cannot attain an absolute error guarantee of $m^{2-\epsilon}$ in Proposition 1. Furthermore, the argument does not go through for

Max Dense Subgraph just because the union of complete graphs is not a complete graph. Luckily, we are able to handle the Max Dense Subgraph problem involving more powerful means.

Theorem 1. *Let ϵ be an arbitrary positive real. Then, unless NP=P, no polynomial-time algorithm approximates $\sigma(G, K_a)$ with absolute error guarantee of $a^{2-\epsilon}$.*

The proof is based on two well-known results. The first is a cornerstone of the extremal graph theory.

Proposition 2. (Turán's theorem, see e.g. [4]) *Let $a \geq b > 1$ and $a = (b-1)d + r$, where $0 \leq r < b-1$. Let $f(a,b)$ denote the maximal possible size of a graph G of order a with $\omega(G) < b$. Then*

$$f(a,b) = a(a-1)/2 - (b-1)d(d-1)/2 - dr.$$

The second result we use is related to the PCP theorem in complexity theory.

Proposition 3. (Bellare, Goldreich, Sudan [2]) *Given a graph G, we will denote its order by n. For each $\delta \in (0, 1/4)$, the problem of distinguishing between the cases $\omega(G) \geq n^{1/4}$ and $\omega(G) < n^{\delta}$ is NP-hard.*

Armed with these heavy guns, we can now prove Theorem 1 with minor efforts. Given ϵ, set $\delta = \epsilon/5$. Assume that there is a polynomial-time algorithm A approximating $\sigma(G, K_a)$ with absolute error guarantee of $a^{2-\epsilon}$. In view of Proposition 3, it suffices to convert it into a polynomial-time algorithm distinguishing between graphs with $\omega(G) \geq n^{1/4}$ and $\omega(G) < n^{\delta}$.

Let $a = \lceil n^{1/4} \rceil$ and $b = \lceil n^{\delta} \rceil$. Note that

$$\begin{aligned} \sigma(G, K_a) &= \tbinom{a}{2} && \text{if } \omega(G) \geq n^{1/4}; \\ \sigma(G, K_a) &\leq f(a,b) && \text{if } \omega(G) < n^{\delta}. \end{aligned}$$

Here $f(a,b)$ is the extremal function given by Proposition 2. A routine calculation shows that the difference between values of $\sigma(G, K_a)$ in the two cases is at least $(1/2 - o(1))n^{1/2-\delta}$.

Running the algorithm A on input (G, K_a), we obtain $A(G, K_a) \geq \binom{a}{2}$ if $\omega(G) \geq n^{1/4}$ and $A(G, K_a) \leq \binom{a}{2} - (1/2 - o(1))n^{1/2-\delta} + a^{2-\epsilon}$ if $\omega(G) < n^{\delta}$. By the choice of δ, in the latter case we have $A(G, K_a) < \binom{a}{2}$ provided n is large enough. This does allow us to distinguish between graphs with the larger cliques and graphs with the smaller cliques.

The proof of Theorem 1 is complete. Since Max Dense Subgraph is a specification of the Largest Common Subgraph problem, we also obtain a strengthening of Proposition 1.

Corollary 1. *Proposition 1 holds true for an absolute error guarantee of $m^{2-\epsilon}$.*

REFERENCES

1. Визинг В.Г. *Хроматический класс мультиграфа,* Кибернетика, **3** (1965), 29–39. English translation available: Vizing V.G. *The chromatic class of a multigraph,* Cybernetics, 1:32–41 (1965).

2. Bellare M., Goldreich O., Sudan M. *Free bits, PCPs, and non-approximability – towards tight results,* SIAM J. Computing, **27** (1998), no.3, 804–915.

3. Crescenzi P., Kann V. *A compendium of NP optimization problems,* On-line version available at *http://www.nada.kth.se/~viggo/wwwcompendium/*

4. Erdős P., Spencer J. Probabilistic methods in combinatorics. Akadémiai Kiadó, Budapest (1974). Russian translation available: Эрдёш П., Спенсер Дж. Вероятностные методы в комбинаторике. М.: Мир, 1976.

5. Feige U., Kortsarz G., Peleg D. *The Dense k-Subgraph problem,* Algorithmica, **29** (2001), no.3, 410–421.

6. Frieze A., Jerrum M. *Improved approximation algorithms for MAX k-CUT and MAX BISECTION,* Algorithmica, **18** (1997), 67–81.

7. Garey M.R., Johnson D.S. Computers and Intractability. A guide to the theory of NP-completeness, W.H. Freeman, San Francisko (1979). A Russian translation available: Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи, М.: Мир, 1982.

8. Goemans M., Williamson D. *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming,* J. of the ACM **42** (1995), no 6, 1115–1145.

9. Khanna S., Motwani R., Sudan M., Vazirani U. *On syntactic versus computational views of approximability,* SIAM J. Computing **28** (1999), 164–191.

10. Håstad J. *Some optimal inapproximability results.* In: Proc. of the 29-th ACM Ann. Symp. on Theory of Computing, (1997), 1–10.

11. Holyer I. *The NP-completeness of edge-coloring,* SIAM J. Computing **10** (1981), no.4, 718–720.

12. Ye Y. A. 699 approximation algorithm for MAX BISECTION. Manuscript, 1999.

Department of Algebra and Mathematical Logic
Faculty of Mechanics and Mathematics
Kyiv National University, Ukraine

Received 5.05.2004