Randomness and Computation
Joint Workshop "New Horizons in Computing" and "Statistical Mechanical Approach to Probabilistic Information Processing" (18-21 July, 2005, Sendai, Japan)

Dense subgraph problem revisited

Akiko Suzuki and Takeshi Tokuyama [1]
Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan

We consider the weighted *dense subgraph problem* (often called the maximum dispersion problem or dense k-subgraph problem) defined as follows:

> Consider a weighted graph $G = (V, E)$, where $|V| = n$ and each edge e has a nonnegative weight $0 \le w(e) \le 1$. Given a natural numbers $k \le n$, find a subgraph $H = (X, F)$ of G such that $|X| = k$ and $w(F) = \sum_{e \in F} w(e)$ is maximized.

Its bipartite version is as follows:

> Consider a weighted bipartite graph $G = (U, V, E)$, where $|U| = m$, $|V| = n$ and each edge e has a nonnegative weight $0 \le w(e) \le 1$. Given two natural numbers $m' \le m$ and $n' \le n$, find a subgraph $H = (X, Y, F)$ of G such that $|X| = m'$, $|Y| = n'$ and $w(F) = \sum_{e \in F} w(e)$ is maximized.

We note the condition $0 \le w(e) \le 1$ is given since it is convenient for presenting our theoretical results, although we can define each problem without this condition. We say *unweighted dense subgraph problem* if $w(e) = 1$ for each edge. We define the density Δ of the output subgraph H to be $\Delta = \frac{2w(F)}{k(k-1)}$ for the non-bipartite case and $\Delta = \frac{w(F)}{m'n'}$ for the bipartite case. In other words, density is the ratio of the weight sum to the number of edges in a clique (or a bipartite clique) of the same size. We mainly consider the case where the density of the optimal output subgraph is high (e.g., $\Delta = \Omega(1)$), and aim to design efficient approximation algorithms.

Previous work

The densest subgraph problem that finds a subgraph with the maximum average degree without any size constraint of the subgraph can be solved in polynomial time [5]. However, the unweighted dense subgraph problem (where k is given) is NP-hard since the max-clique problem is reduced to it.

Therefore, several theoretical approximation algorithms have been proposed in the literature for the dense subgraph problem. We use a convention that an algorithm has an approximation ratio $r > 1$ for a maximization problem if its objective value is at least r^{-1} times the optimal value. In practice, a greedy algorithm removing the smallest weighted-degree vertex one by one often works well; however, its approximation ratio is $2n/k$ (ignoring smaller terms) if $k < n/3$, and it is asymptotically tight [2]. One nice feature of this algorithm is that it gives a constant approximation ratio if $k = \Omega(n)$; however, the linear dependency of the ratio in n is not satisfactory from the theoretical point of view. The current best algorithm has an approximation ratio that is slightly better than $n^{1/3}$, and it is conjectured that there exists some constant ϵ such that n^ϵ approximation is hard, although no nontrivial lower bound for the approximation ratio is known .

Unweighted dense subgraph problems with some additional density conditions on the input/output graph have been also considered. In particular, when the optimal subgraph has $\Omega(n^2)$ edges (thus, $\Delta = \Omega(1)$ and $k = \Omega(n)$), PTAS algorithms are known [1, 3]. However, in many applications of the dense subgraph problem, it is desired to solve the problem with only a density assumption on the optimal output graph. An n^ϵ-approximation algorithm with a time complexity $O(n^{1/\epsilon})$ is known for the case $\Delta = \Omega(1)$ under an additional condition that the average degree of the input graph is $\Omega(k)$ [4]. It is also claimed in [4] that if the optimal subgraph is a clique, there is an $n^{O((1/\epsilon)\log(n/k))}$ time algorithm to have an $(1 + \epsilon)$-approximation solution: This time complexity is polynomial only if $k = \Omega(n)$. As far as the

[1] E-mail: (akiko,tokuyama)@dais.is.tohoku.ac.jp

authors know, no constant-ratio algorithm is known for $k = o(n)$ even for the unweighted problem. It is an interesting question whether we can relax the requirement $k = \Omega(n)$, and this is one of our motivations of this research. Indeed, one may suspect this is difficult, since the maximum clique problem is known to be hard to approximate if the approximation ratio is majored by the number of vertices of the clique.

Next, let us consider the bipartite dense subgraph problem. Although the bipartite dense subgraph problem looks easier than the general dense subgraph problem, it has not been revealed how easier it is. Indeed, it is not much easier. The problem is $\mathcal{NP}$-hard, since the $\mathcal{NP}$-hard edge-maximizing bipartite clique problem is reduced to this problem. Note that bipartite clique problem is polynomial-time soluble if the criterion is to maximize the number of vertices.

Our contribution

For the bipartite dense subgraph problem, let $p = m/m'$, and $q = n/n'$. We have an algorithm with a time complexity $O(mn2^{O(\Delta^{-1}\epsilon^{-2}\log p \log q)})$ that outputs $H_0 = (X_0, Y_0, F_0)$ such that $|X_0| = m'$, $|Y_0| = n'$ and its weighted density Δ_0 satisfies $(1+\epsilon)\Delta_0 \geq \Delta$ for any positive $\epsilon < 1$. The time complexity implies that we have PTAS if either (1) Δ and $\min(p, q)$ are constants, (2) both p and q are constants and $\Delta = \Omega(\log^{-1} n)$, or (3) Δ is a constant and $\max(p, q) = 2^{O(\sqrt{\log mn})}$. We also give a polynomial-time approximation algorithm with an approximation ratio $(\min(p, q))^{\epsilon}$ for any constant $\epsilon > 0$ only with the density condition $\Delta = \Omega(1)$. These results imply that Δ is the principal parameter to control the computational complexity. We remark that $\Delta = \Omega(1)$ and $\min(p, q)$ is small in data mining applications as discussed later.

As direct consequences of the bipartite problem, we have the following results for the non-bipartite dense subgraph problem: We give an algorithm with an approximation ratio $(n/k)^{\epsilon}$ if the optimal solution has $\Omega(k^2)$ edges (or $w(F) = \Omega(k^2)$ for the weighted problem). This improves the previous n^{ϵ} approximation ratio of [4] when k is large, and also the additional condition on the average degree of the input graph is removed. Moreover, we give a $(2+\epsilon)$-approximation algorithm for any $\epsilon \leq 1$ that runs in $O(n^2 2^{O(\Delta^{-1}\epsilon^{-2}\log^2(n/k))})$ time. This implies that we have a $(2+\epsilon)$-approximation polynomial-time algorithm if $\Delta = \Omega(1)$ and $n/k < 2^{\sqrt{\log n}}$. As far as the authors know, this is the first constant-ratio polynomial-time algorithm for the dense subgraph problem that works for some $k = o(n)$.

Technically, we apply the framework of Arora *et al.* [1] that was developed to solve combinatorial problems on a dense input graph. Original framework of [1] is as follows: First formulate the bipartite dense subgraph problem into a quadratic integer programming (QIP) problem. Next, take a small sample set of variables, and guess values of sampled variables in order to transform the QIP to an integer programming (IP) problem. Then, solve its LP relaxization and apply randomized rounding to have an approximate solution of the IP. Finally, we restore a solution of the QIP assuming that our guess is correct. For the bipartite problem, the IP can be precisely solved, thus we do not need to handle error caused by the randomized rounding. Moreover, the QIP is bilinear, and hence the error due to the restoration can be more precisely analyzed. This enables to replace the density condition of the input graph with that of the output graph.

Our complexity results are purely theoretical, and not practically useful as they are since the exponents are high and dependent on parameters. However, our contribution includes novel analysis of a core algorithm of common heuristics, and will help to give a guideline to tune such heuristics.

References

[1] S. Arora, D. Karger, and M. Karpinski, Polynomial Time Approximation Scheme for Dense Instances of $\mathcal{NP}$-hard problems, *Proc. STOC'95* (1995), pp. 284–293.

[2] Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama, Greedily Finding Dense Subgraphs, *Journal of Algorithms,* **34** (2000), 203-221.

[3] A. Czygrinow, Maximum Dispersion Problem in Dense Graphs, *Operation Research Letters* **27** (2000) 223-227.

[4] U. Feige, G. Kortsarz, D. Peleg, The Dense k-Subgraph Problem, *Algorithmica* **29** (2001) 410-421.

[5] G. Gallo, M.D.Grigoriadis, R. E. Tarjan, A Fast Parametric Maximum Flow Algorithm and Applications, *SIAM J. Comput.* **18** (1989) 30-55.