Discrete Applied Mathematics 121 (2002) 15–26

DISCRETE
APPLIED
MATHEMATICS

Complexity of finding dense subgraphs ☆

Yuichi Asahiro[a, *], Refael Hassin[b], Kazuo Iwama[c, 1]

[a] *Department of Computer Science and Communication Engineering, Kyushu University, Fukuoka 812-8581, Japan*

[b] *Department of Statistics & Operations Research, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel*

[c] *School of Informatics, Kyoto University, Kyoto 606-8501, Japan*

Received 13 January 1998; received in revised form 29 March 2001; accepted 9 April 2001

Abstract

The k-$f(k)$ dense subgraph problem(($k, f(k)$)-DSP) asks whether there is a k-vertex subgraph of a given graph G which has at least $f(k)$ edges. When $f(k)=k(k-1)/2$, $(k, f(k))$-DSP is equivalent to the well-known k-clique problem. The main purpose of this paper is to discuss the problem of finding *slightly* dense subgraphs. Note that $f(k)$ is about k^2 for the k-clique problem. It is shown that $(k, f(k))$-DSP remains NP-complete for $f(k)=\Theta(k^{1+\varepsilon})$ where ε may be any constant such that $0<\varepsilon<1$. It is also NP-complete for "relatively" slightly-dense subgraphs, i.e., $(k, f(k))$-DSP is NP-complete for $f(k)=ek^2/v^2(1+\mathrm{O}(v^{\varepsilon-1}))$, where v is the number of G's vertices and e is the number of G's edges. This condition is quite tight because the answer to $(k, f(k))$-DSP is always yes for $f(k)=ek^2/v^2(1-(v-k)/(vk-k))$ that is the average number of edges in a subgraph of k vertices. Also, we show that the hardness of $(k, f(k))$-DSP remains for regular graphs: $(k, f(k))$-DSP is NP-complete for $\Theta(v^{\varepsilon_1})$-regular graphs if $f(k)=\Theta(k^{1+\varepsilon_2})$ for any $0<\varepsilon_1,\varepsilon_2<1$. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Dense subgraph; Regular graph; NP-complete; Threshold of complexity

1. Introduction

Let $f(k)$ be a function called an edge density function. The k-$f(k)$*dense subgraph problem* (($k, f(k)$)-DSP) asks, given a graph G of v vertices and e edges and an integer k, whether there is a k-vertex subgraph which has at least $f(k)$ edges. When $f(k)=k(k-1)/2$, $(k, f(k))$-DSP is equivalent to the well known k-clique problem

☆ An earlier version of this paper was presented at ISAAC '95.

* Corresponding author.

E-mail address: asahiro@csce.kyushu-u.ac.jp (Y. Asahiro).

[1] Supported by Grant-in-Aid for Scientific Research, Ministry of Education, Japan, No. 08244105, No. 08558024, and No. 07458061.

0166-218X/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0166-218X(01)00243-8

[12]. Therefore the problem is a generalization of the k-clique problem and is obviously NP-complete.

Obtaining a clique, which can be regarded as "the densest" subgraph, is thus intractable. By contrast, it is no wonder that obtaining "sparse" subgraphs is easy. For example, if $f(k)=k$, then $(k, f(k))$-DSP can be solved in polynomial time by a simple dynamic programming (see Section 3). If the density does not have to be more than average, then the problem is again not hard: Note that the average number of edges in a k-vertex subgraph is $(ek(k-1))/(v(v-1))$ $(=ek^2/v^2(1-(v-k)/(vk-k))$. By using the probabilistic method [1], we can show that there always exists a subgraph of k vertices and at least the above average number of edges. Therefore if we consider the case $f(k)=(ek(k-1))/(v(v-1))$, the answer of $(k, f(k))$-DSP is always *yes* and actually there is a simple polynomial-time algorithm that can find such a subgraph (see Section 3). Thus the problem is easy if the required density is low and becomes hard if it is high. A natural question is whether it is hard when the required density is "slightly" high.

In this paper, we show that the answer to this question is yes. Two edge density functions are introduced, one is slightly dense in terms of its absolute value and the other in terms of its relative value. Main results include: $(k, f(k))$-DSP remains NP-complete (1) for $f(k)=\Theta(k^{1+\varepsilon_1})$, (2) for $f(k)=ek^2/v^2(1+\mathrm{O}(v^{\varepsilon_1-1}))$, and also, (3) the result of (1) holds for a class of regular graphs, i.e., $(k, f(k))$-DSP is NP-complete for $\Theta(v^{\varepsilon_1})$-regular graphs with $f(k)=\Theta(k^{1+\varepsilon_2})$ where ε_1 and ε_2 may be any constants such that $0<\varepsilon_1, \varepsilon_2<1$. The edge density function for (2) is quite tight since there always exists a subgraph which has at least $ek^2/v^2(1-(v-k)/(vk-k))$ edges as mentioned above.

One can think of several applications of $(k, f(k))$-DSP. Among others, we shall briefly mention its application to the security of generating random test-instances for the CNF satisfiability problem [3]. When generating test-instances for evaluating the performance of combinatorial algorithms empirically, one of our concerns is that the algorithms could be tuned so as to run fast especially for the benchmarks by exploiting their generation method. It turns out that $(k, f(k))$-DSP is closely related to the security in this sense and its intractability is a good news to claim the hardness of the unnatural tune-up of the algorithms mentioned above.

2. Related works

$(k, f(k))$-DSP is the decision version of the *maximum edge subgraph problem* (*MES*): For a given graph $G=(V,E)$ with nonnegative edge weights and a positive integer $k \leqslant |V|$, we are required to find a k-vertex subgraph which has maximum weights among all the k-vertex subgraphs in G. Several approximation algorithms for MES are known: For general MES, Feige, Kortsarz and Peleg developed the algorithm whose approximation ratio is $\mathrm{O}(|V|^{1/3}\log|V|)$ [7]. (Here, *approximation ratio* means the supremum of *OPT*/*A* over all instances, where *OPT* is the weights of the optimal solution and A is the weights of the solution found by the algorithm.) A simple greedy algorithm can solve this problem with approximation ratio of $(|V|/2k + 1/2)^2$ [4]. If given graphs satisfy the triangle inequality, then MES can be approximable within 2 [11], although it remains to be NP-hard [14]. Moreover, there is a polynomial time approximation scheme(PTAS) if we restrict instances to unweighted and dense graphs satisfying that $|E| = \Theta(|V|^2)$ [2]. Recently, Czygrinow proposed a fully polynomial time approximation scheme(FPTAS) for the dense case [6].

One of the other related problems is the densest subgraph problem [13]. Its objective is finding a subgraph with the maximum average degree for a given graph $G=(V,E)$. For this problem we can choose any number of vertices as the vertex set of a subgraph, In contrast to MES which requires to find subgraphs of the fixed number k of vertices. The densest subgraph problem can be solved in $\mathrm{O}(|V||E|\log(|V|^2/|E|))$ [9].

The complexity of $(k, f(k))$-DSP exhibits a so-called *threshold* behavior which plays an important role in the complexity analysis. Actually, there are many problems whose complexities jump at some point as the value of some parameter of the problem grows. For example, as for the k-CNF satisfiability problem, if the number k of literals included in a clause increases 2 to 3, its complexity changes from P to NP-complete [5]. The graph colorability problem is easy with 2 colors, but NP-complete with 3 colors [10]. Also, k-CLIQUE is solvable in polynomial time when k is a fixed constant but turned to be NP-complete for bigger $k = \Omega(n^\varepsilon)$, where n is the number of vertices of a given graph and ε is any small constant. Similar to those examples, the complexity of $(k, f(k))$-DSP varies based on the parameter $f(k)$.

3. Main results

In this paper, a *graph* $G=(V,E)$ always means an undirected, unweighted, simple graph. $|V|$ is denoted by v and $|E|$ by e. Also in this paper, a *subgraph* G' is determined only by a set $V' \subseteq V$ of vertices. Namely, G' is the so-called induced subgraph, or $G'=(V',E')$, where $E' = \{(v_1,v_2) \mid (v_1,v_2) \in E \text{ and } v_1, v_2 \in V'\}$. Also in this paper, the number, x, always shows the least integer that is greater than or equal to x, i.e., $\lceil x \rceil$.

Theorem 1. *$(k, f(k))$-DSP is NP-complete for $f(k) = \Theta(k^{1+\varepsilon})$ where ε may be any positive constant less than one.*

Theorem 2. *$(k, f(k))$-DSP is NP-complete for $f(k) = ek^2/v^2(1 + \mathrm{O}(v^{\varepsilon-1}))$ where ε may be any positive constant less than one.*

The proofs will be given in Section 4. These results for two types of density are obtained at the same time by one reduction. As for the first criteria in Theorem 1, the better bound is claimed, that is, $(k, f(k))$-DSP is NP-complete for $f(k) = k + k^\varepsilon$ [8]. Based on the proofs of Theorems 1 and 2, we can extend the result to the case for regular graphs. The proof of the next theorem will be shown in Section 5.

Theorem 3. *$(k, f(k))$-DSP is still NP-complete even if the input graph is $\Theta(v^{\varepsilon_1})$-regular and $f(k) = \Theta(k^{1+\varepsilon_2})$ for any $0 < \varepsilon_1 < 1$ and $0 < \varepsilon_2 < 1$.*

Note that slightly changing the proofs of Theorems 1 and 3 gives us the restriction of the number of the vertices, namely, $(k, f(k))$-DSP is still NP-complete under same conditions of those theorems with the additional restriction $k = \Theta(v^{\varepsilon_3})$ for any $0 < \varepsilon_3 < 1$.

The following two propositions state the cases that are solvable in polynomial time. Theorem 2 is fairly tight due to Proposition 5 (recall that $ek^2/v^2(1-(v-k)/(vk-k))$ is the average number of edges in a k-vertex subgraph).

Proposition 4. *There is a deterministic polynomial time algorithm which finds a k-vertex subgraph with at least k edges, if one exists.*

Proof. Suppose that the given graph consists of connected components C_i's, and let n_i and m_i denote the number of vertices and edges of C_i, respectively. Also, define $l_i = m_i - n_i$.

Step 1: If there exists a connected component of at least k vertices, which includes a cycle of size at most k, then output a connected k-vertex induced subgraph which contains this cycle. Note that the shortest cycle can be found by breadth first search. In the following we assume that if the size of a connected component is larger than k, then its smallest cycle is of length $> k$.

Step 2: Classify the connected components into three groups, S^+, S^0, and S^- in terms of l_i's: Let $S^+ = \{C_1, \ldots, C_\alpha\}$, $S^0 = \{C_{\alpha+1}, \ldots, C_\beta\}$, and $S^- = \{C_{\beta+1}, \ldots, C_\gamma\}$, where (i) $n_i < k$ and $l_i > 0$ for $1 \leqslant i \leqslant \alpha$, (ii) $n_i < k$ and $l_i = 0$ for $\alpha + 1 \leqslant i \leqslant \beta$, and (iii) $n_i \geqslant k$ or $l_i = -1$ for $\beta + 1 \leqslant i \leqslant \gamma$. Note that S^- contains trees and big components whose size $\geqslant k$, so that any connected subgraph of size $\leqslant k$ is a tree. Every component in S^0 is a tree plus one additional edge. Note that S^+, S^0, or S^- might be empty. We can assume that $l_1 \geqslant \cdots \geqslant l_\alpha > 0$ and $n_{\beta+1} \geqslant \cdots \geqslant n_\gamma$. Let δ denote the least integer such that $\sum_{i=1}^{\delta} n_i \geqslant k$.

Step 3: (a) If $S^+ \neq \emptyset$, goto step 4. (b) If $S^+ = \emptyset$ and $S^0 = \emptyset$, output *no*. (c) Otherwise ($S^+ = \emptyset$ and $S^0 \neq \emptyset$), goto step 5.

Step 4: If $\sum_{i=1}^{\delta-1} l_i \geqslant 1$ for some $\delta \leqslant \gamma$, output $\bigcup_{i=1}^{\delta-1} C_i$ and a $k - \sum_{i=1}^{\delta-1} n_i$ vertex connected subgraph in C_δ. Otherwise (i.e., if $\sum_{i=1}^{\delta-1} l_i \leqslant 0$ which means $C_{\delta-1} \in S^-$), output *no*.

Step 5: Compute the size c_i of the cycle for each $C_i \in S^0$. We can consider the problem as a variation of Subset Sum Problem: Is there a subset $S' \subseteq S^0$ for which we can select an integer $c_i \leqslant s(i) \leqslant n_i$ for each $C_i \in S'$ such that $\sum_{C_i \in S'} s(i) = k$? Use dynamic programming to solve this problem. □

Proposition 5. *There is a deterministic polynomial time algorithm which finds a subgraph with at least $ek^2/v^2(1-(v-k)/(vk-k))$ edges.*

Proof. One such algorithm is the following greedy algorithm presented in [4]. $G[V']$ denotes the subgraph of G induced by a set $V' \subseteq V$ of vertices.

Step 1: $G' \leftarrow G, V' \leftarrow V$.

Step 2: Select a minimum-degree vertex u from G'. $V' \leftarrow V' - \{u\}$ and $G' \leftarrow G[V']$.

Step 3: Repeat step 2 until G' has k vertices. Then output G'. □

Note that if G is a random graph then it is easy to see that the average number of edges included in k-vertex subgraph is $ek^2/v^2(1-(v-k)/(vk-k))$. Proposition 5 says that we can select this dense subgraph for *any* particular graph. Recall that the density function in Theorem 2 is $ek^2/v^2(1 + O(v^{\varepsilon-1}))$. Thus ek^2/v^2 is an important border: If the density is slightly larger than ek^2/v^2, then the problem is hard in general. The problem becomes easy if the density is slightly smaller than ek^2/v^2.

4. Proof of Theorems 1 and 2

It is obvious that $(k, f(k))$-DSP is in NP. To prove its NP-hardness, we reduce a clique problem to $(k, f(k))$-DSP. Here we consider the restricted clique problem which asks whether there exists an n-vertex complete graph(n-*clique*) in a given $2n$-vertex graph $G=(V,E)$. It can be easily shown that this clique problem is still NP-complete by reduction from the general k-clique problem as follows: For an input graph I of n vertices, we add an $n-k$-vertex complete graph K_{n-k}, complete bipartite connection between I and K_{n-k}, and k isolated vertices. The consequent graph has $2n$ vertices, and has an n-clique if and only if a k-clique exists in I.

Let $f(k)=nm(2n-1)+n(n-1)/2$, where m is a polynomial in n and determined later. Construct a graph $H=(V',E')$ composed of a copy G' of $G=(V,E)$ and m complete graphs, each of which has $2n$ vertices. H has $|V'|=2n(m+1)$ vertices and $|E'|=|E|+nm(2n-1)$ edges in total. Then set $k=2nm+n$. This construction of H can be done in polynomial time obviously.

Lemma 6. *Suppose that there are m complete graphs $I_1,\ldots,I_m$ of $2n$ vertices and one (not necessarily complete) graph G of $2n$ vertices. Take $2nm+n$ vertices among those $2nm+2n$ ones. Then the number of induced edges becomes maximum when we take all the $2nm$ vertices of $I_1,\ldots,I_m$ (and other n ones from G).*

Proof. Suppose that we take $2nm-d$ vertices from $I_1,\ldots,I_m$ and $n+d$ vertices from G. (See Fig. 1 for $m=2$ and $d=3$.) Then one can easily see that the number of induced edges does not decrease if we abandon (any) d (three in Fig. 1) vertices from G and take d vertices from I_1 and I_2, since I_1 and I_2 are both complete. This statement is obviously true for general m and d. Thus the lemma holds. □

This lemma shows that a most dense k-vertex subgraph of H consists of all the $2n$-vertex cliques and an n-vertex subgraph of G'. If the number of edges in this

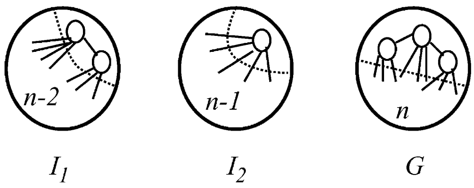

Fig. 1. Proof of Lemma 4.1.

subgraph is $f(k)$, then the number of edges in the subgraph taken from G' must be $n(n-1)/2$; namely it must be a clique. Conversely, if G' has a clique of size n, then it is obviously possible to take a k-vertex subgraph of $f(k)$ edges.

For the proof of Theorem 1, it remains to show that given $0<\varepsilon<1$, $f(k)$ can be chosen so that it meets the condition $f(k)=\Theta(k^{1+\varepsilon})$. For any given fixed $0<\varepsilon<1$ we choose $m=n^{1/\varepsilon-1}$, so that $k=2n^{1/\varepsilon}+n$ and $f(k)=2n^{1+1/\varepsilon}-n^{1/\varepsilon}+n(n-1)/2$. Thus, roughly speaking, $k^{1+\varepsilon}/2^{\varepsilon+1}<f(k)<k^{1+\varepsilon}/2^{\varepsilon-1}$, that is $f(k)=\Theta(k^{1+\varepsilon})$. Then, since $f(k)=|E'|k^2/|V'|^2(1+\mathrm{O}(m^{-1}))$ and $m=\mathrm{O}(|V'|^{1-\varepsilon})$, we also obtain Theorem 2 from Theorem 1.

5. Regular graphs

In this section we consider regular graphs. We construct a slightly different graph G' from H to prove Theorem 3. The outline is as follows: First we prove similar results as in the proof of Theorem 1 in Sections 5.1 and 5.2. Then in Section 5.3, we show how to make involved graphs regular while keeping the NP-completeness of the problem.

5.1. Transformation of graphs

This time, we reduce another variation of the clique problem, asking whether there is an n-clique in a given $3n$-vertex graph(also NP-complete). First of all, we set $f(k)=\frac{27}{2}n^2m-\frac{3}{2}nm-\frac{11}{2}n^2+\frac{1}{2}n$, where m is a polynomial in n and is determined later. The condition $f(k)=\Theta(k^{1+\varepsilon})(0<\varepsilon<1)$ will be satisfied by selecting proper m.

Construct a graph $G'=(V',E')$ composed of a copy $G_0=(V_0,E_0)$ of $G=(V,E)$ and m complete graphs $G_1=(V_1,E_1),\ldots,G_m=(V_m,E_m)$, each of which has $3n$ vertices (see Fig. 2). Then edges are placed as a complete bipartite connection between V_i and V_{i+1} for $1\leqslant i\leqslant m-1$ and between V_m and V_0. Such edges are $(3n)^2m$ in total. Also $3n$ edges are added between G_0 and G_1 so that each vertex in G_0 is connected to exactly one vertex in G_1 and vice versa. Thus G' has $|V'|=3n(m+1)$ vertices and $|E'|=\frac{27}{2}n^2m-\frac{3}{2}nm-9n^2+|E|$ edges in total. Recall that we did not need these connections between V_i and V_{i+1} in the proof of Theorem 1. The role of these new edges will be described later.

As for k, we select the values $k=3nm+n$. It is easy to see that this construction of G' and k can be carried out in polynomial time in n, since m is polynomial in n.

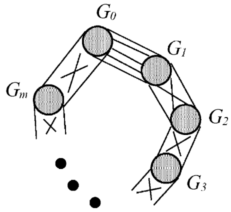

Fig. 2. The graph G'.

It remains to show that given $0 < \varepsilon < 1$, $f(k)$ can be chosen so that it meets the condition $f(k) = \Theta(k^{1+\varepsilon})$. For any given fixed $0 < \varepsilon < 1$ we choose $m = n^{1/\varepsilon - 1}$, so that $k = 3n^{1/\varepsilon} + n$ and $f(k) = \frac{27}{2}n^{1+1/\varepsilon} - \frac{3}{2}n^{1/\varepsilon} + \frac{11}{2}n^2 + \frac{1}{2}n$. Thus, roughly speaking, $k^{1+\varepsilon} < f(k) < \frac{9}{2 \cdot 3^\varepsilon}k^{1+\varepsilon}$, that is $f(k) = \Theta(k^{1+\varepsilon})$.

5.2. NP-completeness

In this section, we will prove that there exists an n-clique in G if and only if there exists a subgraph of $k = 3nm + n$ vertices and at least $f(k) = \frac{27}{2}n^2 m - \frac{3}{2}nm - \frac{11}{2}n^2 + \frac{1}{2}n$ edges in G'. $G[W]$ denotes the subgraph of G induced by a vertex set $W \subseteq V$.

Lemma 7. *If there exists an n-clique in G, G' includes a subgraph of k vertices and $f(k)$ edges.*

Proof. Suppose that G contains an n-clique. Let V_0' be the set of the vertices which forms this n-clique. It is easy to see that $G'[V_0' \cup V_1 \cup \cdots \cup V_m]$ has k vertices and $f(k)$ edges. □

Lemma 8. *Every subgraph of G' induced by $V_1 \cup \cdots \cup V_m$ and n vertices in V has less edges than $f(k)$, if G does not have n-cliques.*

Proof. Let V_0' be any set of n vertices in G. Since $G[V_0']$ is not an n-clique, $G'[V_0' \cup V_1 \cup \cdots \cup V_m]$ has less edges than $f(k)$. □

We shall call the above selection of vertices, i.e., choosing all vertices of V_i's and some n vertices in V, *Selection A*.

Lemma 9. *Selection A can induce a k-vertex subgraph of G' with maximum number of edges.*

Proof. We introduce the notion of "moving vertices". Let S be any set of k vertices. Then S can be written as

$$S = (S_A - S_{cut}) \cup (\overline{S_A} \cap S_{paste}),$$

by using S_A, $\overline{S_A}$, S_{cut}, and S_{paste} where S_A is a set of the k vertices selected by Selection A, $\overline{S_A}$ is its complement, $S_{cut} \subseteq S_A$, and $S_{paste} \subseteq \overline{S_A}$. Since $|S| = |S_A| = k$, $|S_{cut}|$ and $|S_{paste}|$ are the same. Namely, we can select S by first selecting some S_A by Selection A, then removing S_{cut} from S_A and the same number of vertices $(= S_{paste})$ are selected from $\overline{S_A}$. We shall say that *S can be selected by moving $|S_{cut}|$ vertices*. When we delete S_{cut} from S, the number of induced edges decreases by, say, e^-, and when we add S_{paste}, the number of induced edges increases by, say, e^+. If it always holds that $e^- - e^+ \geqslant 0$, $G[S_A]$ has the maximum number of edges among all k-vertex subgraphs, that we would like to claim.

Let S be a selection of k vertices. Then any such selection can be obtained from Selection A by the following procedure:

(1) move x_1 vertices from V_1 to V_0

(2) move x_2 vertices from V_2 to V_0

$\vdots$

(m) move x_m vertices from V_m to V_0

where each $x_i \geqslant 0$ and $0 < \sum_{j=1}^{m} x_i \leqslant 2n$. Note that $\sum_{j=1}^{m} x_i$ is the total number of vertices moved from V_i's to V_0.

Let S_i be the set of k vertices obtained by executing the above procedures only (1) through (i), i.e., the set of vertices at the intermediate step to construct S from S_A. Note that $S_0 = S_A$.

To prove the lemma, we will show $G'[S_A]$ always has more edges than $G'[S_i]$. Let e_i^- be the number of edges in the subgraph that reduces by removing x_i vertices from $G'[S_{i-1}]$. Then by adding x_i vertices in $V_0 \cap \overline{S_{i-1}}$, the number of edges of the subgraph increases. Let e_i^+ be that increasing number. We will show that $\sum_{i=1}^{m} (e_i^- - e_i^+) \geqslant 0$ by induction on i.

Suppose $i = 1$. If we remove x_1 vertices in V_1 from $G'[S_A]$,

$$e_1^- \geqslant \frac{x_1(x_1 - 1)}{2} + (3n - x_1)x_1 + 3nx_1 + \alpha,$$

where $x_1(x_1 - 1)/2$ is the number of edges among those x_1 vertices, $(3n - x_1)x_1$ is the number of edges between the x_1 vertices and the remaining vertices of V_1, $3nx_1$ is the number of edges between the x_1 vertices in V_1 and V_2, and α is the number of edges between the x_1 vertices in V_1 and V_0.

When adding x_1 vertices in $V_0 \cap \overline{S_A}$, e_1^+ achieves the maximum number when those x_1 vertices form a complete graph, and every one of these x_1 vertices is connected with all n vertices in V_0 selected by S_A. Therefore,

$$e_1^+ \leqslant \frac{x_1(x_1 - 1)}{2} + nx_1 + 3nx_1 + \beta,$$

where nx_1 is the number of the edges between the x_1 vertices and n vertices in V_0, which have been selected by S_A before this move, and $3nx_1$ and β is the number of edges between V_0 and V_m, and V_1 and V_0, respectively. Note that $\beta \leqslant \max\{(3n - x_1) - (n - \alpha), x_1\}$, because of the way of connection.

Therefore,

$$\begin{aligned} e_1^- - e_1^+ &\geqslant (2n - x_1)x_1 + \alpha - \beta \\ &\geqslant (2n - x_1)x_1 + \min\{-x_1, x_1 - 2n\}. \end{aligned}$$

It then follows that $e_1^- - e_1^+ \geqslant 0$ for both cases that $\min\{-x_1, x_1 - 2n\} = -x_1$ and $x_1 - 2n$.

Suppose that $\sum_{i=1}^{j-1}(e_i^- - e_i^+) \geqslant 0$.

Case 1: $j \leqslant m - 1$.

Similarly as the case $i = 1$, when we add x_j vertices in $V_0 \cap \overline{S_{j-1}}$, e_j^+ becomes maximum when those x_j vertices form a complete graph, and every one of those x_j vertices is connected with all vertices in V_0.

$$e_j^- = \frac{x_j(x_j - 1)}{2} + (3n - x_j)x_j + (3n - x_{j-1})x_j + 3nx_j,$$

where the number of edges between those x_j vertices and the remaining vertices in V_j corresponds to $(3n - x_j)x_j$, and the third and fourth terms are the number of edges between V_{j-1} and V_j and the number of edges between V_j and V_{j+1}, respectively.

Since $\sum_{h=1}^{j-1} x_h$ vertices are added to V_0 so far, $n + \sum_{h=1}^{j-1} x_h$ vertices are selected in V_0 by S_{j-1}. The maximum number of edges between those vertices in V_0 and moved x_j vertices is $(n + \sum_{h=1}^{j-1} x_h)x_j$ when they are connected by a complete bipartite connection. Therefore,

$$e_j^+ \leqslant \frac{x_j(x_j - 1)}{2} + \left(n + \sum_{h=1}^{j-1} x_h\right)x_j + 3nx_j + x_j,$$

and then,

$$\begin{aligned} e_j^- - e_j^+ &\geqslant \left(5n - x_{j-1} - \sum_{h=1}^{j} x_h - 1\right)x_j \\ &\geqslant (3n - x_{j-1} - 1)x_j. \end{aligned}$$

Hence, if $x_j > 0$ then $x_{j-1} \leqslant n - 1$ and $e_j^- - e_j^+ \geqslant 2nx_j > 0$. Therefore $\sum_{i=1}^{j}(e_i^- - e_i^+) \geqslant 0$.

Case 2: $j = m$.

The difference from the Case 1 is due to the edges between V_j and $V_{j+1}(= V_0)$. The difference can be seen in the last terms in the following two inequalities for e_m^- and e_m^+. By S_{m-1}, $n + \sum_{h=1}^{m-1} x_h$ vertices in V_0 and $3n$ vertices in V_m are selected. So by moving x_m vertices, the decreasing number of edges between V_m and V_0 is $(n + \sum_{h=1}^{m-1} x_h)x_m$, and increasing number between V_m and V_0 is $(3n - x_m)x_m$, because x_m vertices were removed from V_m.

$$e_m^- \geqslant \frac{x_m(x_m - 1)}{2} + (3n - x_m)x_m + (3n - x_{m-1})x_m + \left(n + \sum_{h=1}^{m-1} x_h\right)x_m,$$

$$e_m^+ \leqslant \frac{x_m(x_m - 1)}{2} + \left(n + \sum_{h=1}^{m-1} x_h\right)x_m + x_m + (3n - x_m)x_m.$$

Therefore,

$$e_m^- - e_m^+ \geqslant (3n - x_{m-1} - 1)x_m.$$

As in the (Case 1), $\sum_{i=1}^{j}(e_i^- - e_i^+) \geqslant 0$. □

From the Lemmas 8 and 9, we obtain the following lemma.

Lemma 10. *If there does not exist an n-clique in G, then any k-vertex subgraph of G' has less edges than $f(k)$.*

5.3. Making the graphs regular

In this section, we outline the rest of the proof. Note that G' is not regular but almost regular, i.e., vertices in V_2 through V_m have the same degree. Let $D_{\max}$ be the maximum degree of vertices in G'. To regularize the graph G', we introduce a new $D_{\max}$-regular graph $G_R = (V_R, E_R)$, whose detailed construction will be given later, and then G_R is connected to G' in the following manner: (i) Each vertex v in G whose degree, say, d, is less than $D_{\max}$ is connected to $D_{\max} - d$ vertices in G_R. Then (ii) if v was connected to v_1 and v_2 in G_R, then the original edge between v_1 and v_2 is removed (and hence the degree of v_1 and v_2 does not change).

For example, suppose that we wish to increase the degrees of four vertices a, b, c, d, where the degree of a and b is $D_{\max} - 2$, and the degree of c and d is $D_{\max} - 1$. We have to increase six degrees in total. In this case, we remove three edges in G_R, e.g., (v_1, v_2), (v_3, v_4) and (v_5, v_6). Then we connect between the endpoints of these edges and vertices a, b, c, and d by six edges, i.e., by (v_1, a), (v_2, a), (v_3, b), (v_4, b), (v_5, c), and (v_6, d), so that each of four vertices has degree $D_{\max}$.

The main idea of the construction is similar to the previous one: The selection which induces the maximum number of edges will be the same as Selection A in G' described in Section 5.2. Namely, if any vertex is selected from G_R instead of G', then the number of edges induced by such a selection decreases compared to Selection A.

Now we describe the detailed construction of G_R and connection between G' and G_R. See Fig. 3. In the graph G', the maximum degree $D_{\max}$ of the vertices is $9n - 1$ and we have to increase the degrees of the vertices in V and V_1. The number D which we increase is $D = 6n(3n - 1) - 2|E|$ in total.

G_R is a composition of $Dl(m + 1)/2$ copies of the $3n$-vertex complete graph, where l is a polynomial in n. Let those complete graphs be $G_{R,1}, \ldots, G_{R,Dl(m+1)/2}$. Then for $1 \leqslant i \leqslant Dl(m + 1)/2 - 1$, we place $9n^2$ edges between $G_{R,i}$ and $G_{R,i+1}$ by complete bipartite connection. The same connection is placed between $G_{R,Dl(m+1)/2}$ and $G_{R,1}$ as well.

The rule of connecting G' and G_R is as follows: For $1 \leqslant i \leqslant D/2$, remove one edge from $G_{R,mi}$. Then connect between endpoints of those removed edges and vertices with less degree than $D_{\max}$ in G' as described above.

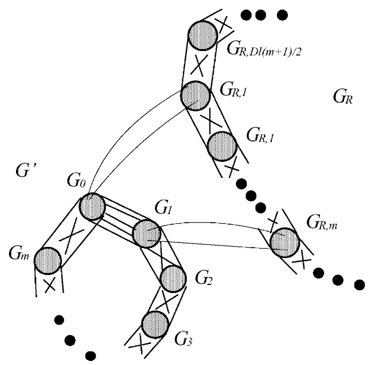

Fig. 3. Proof for regular graphs.

By connecting G_R and G' like this, we obtain the $9n-1$-regular graph G'' of $3n(m+1+Dl(m+1)/2)$ vertices. If we set $k=3nm+n$ and $f(k)=\frac{27}{2}n^2m-\frac{3}{2}nm-\frac{11}{2}n^2+\frac{1}{2}n$, then we can show that there exists an n-clique in G iff G'' consists of k-vertex subgraph with at least $f(k)$ edges. Its proof is very similar to the one in Section 5.2.

It should be noted that we placed some edges between $3n$-vertex complete graphs in Section 5.2, although no edge existed there in Section 4. The reason we introduce such edges is as follows: Those edges force us to choose vertices from G' instead of G_R to maximize the number of edges of k-vertex subgraph: If we regard each $3n$-vertex complete graph as a single vertex, then we can induce a (kind of) cycle constituted by those vertices in the case of G'. However we cannot obtain this kind of "cycles" from G_R because the size of the cycle is too large.

6. Concluding remarks

We proved that several restricted instances for $(k, f(k))$-DSP are NP-complete. As a variant of this problem, we can consider a problem which asks, given a graph and k, whether there is a k-vertex subgraph with *exactly* $f(k)$ edges. For extreme cases $f(k)=k(k-1)/2$ and $f(k)=0$, this problem is equal to k-clique problem and k-independent set problem, respectively. We can show that this problem is NP-complete for $f(k)=\Theta(k^{1+\varepsilon})$ as the original DSP.

References

[1] N. Alon, J.H. Spencer, P. Erdös, The Probabilistic Method, Wiley, New York, 1992.

[2] S. Arora, D.Karger, M. Karpinski, Polynomial time approximation schemes for dense instances of NP-hard problems, Proceedings of the 27th ACM Symposium on Theory of Computing, 1995, pp. 284–293.

[3] Y. Asahiro, K. Iwama, E. Miyano, Random generation of test instances with controlled attributes, DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 26 (1996) 377–393.

[4] Y. Asahiro, K. Iwama, H. Tamaki, T. Tokuyama, Greedily finding a dense subgraph, Proceedings of the Fifth Scandinavian Workshop on Algorithm Theory, Lecture Notes in Computer Science, Vol. 1097, Springer, New York, 1996, pp. 136–148.
[5] S.A. Cook, The complexity of theorem-proving procedures. Proceedings of the Third ACM Symposium on Theory of Computing, 1971, pp. 151–158.
[6] A. Czygrinow, Maximum dispersion problem in dense graphs, Oper. Res. Lett. 27 (2000) 223–227.
[7] U. Feige, G. Kortsarz, D. Peleg, The dense k-subgraph problem, unpublished manuscript.
[8] U. Feige, M. Seltser, On the densest k-subgraph problem, Technical Report CS97-16, Faculty of Mathematics and Computer Science, The Weizmann Institute of Science.
[9] G. Gallo, M.D. Grigoriadis, R.E. Tarjan, A fast parametric maximum flow algorithm and applications, SIAM J. Comput. 18 (1) (1989) 30–55.
[10] M.R. Garey, D.S. Johnson, L. Stockmeyer, Some simplified NP-complete graph problems, Theoret. Comput. Sci. 1 (1976) 237–267.
[11] R. Hassin, S. Rubinstein, A. Tamir, Approximation algorithms for maximum dispersion, Oper. Res. Lett. 21 (1997) 133–137.
[12] R.M. Karp, Reducibility among Combinatorial Problems, Complexity of Computer Computations, Plenum Press, New York, 1972, pp. 85–103.
[13] E.L. Lawler, Combinatorial Optimization: Networks and matroids, Holt, Rinehart and Winston, New York, 1976.
[14] S.S. Ravi, D.J. Rosenkrantz, G.K. Tayi, Heuristic and special case algorithms for dispersion problems, Oper. Res. 42 (1994) 299–310.