Electronic Colloquium on Computational Complexity - Reports Series 1997 - available via:

ECCC
TR97-004

FTP: ftp.eccc.uni-trier.de:/pub/eccc/
WWW: http://www.eccc.uni-trier.de/eccc/
Email: ftpmail@ftp.eccc.uni-trier.de with subject 'help eccc'

Approximating Dense Cases of Covering Problems

Marek Karpinski* Alexander Zelikovsky†

Abstract

We study dense instances of several covering problems. An instance of the set cover problem with m sets is dense if there is $\epsilon > 0$ such that any element belongs to at least ϵm sets. We show that the dense set cover problem can be approximated with the performance ratio $c \log n$ for any $c > 0$ and it is unlikely to be NP-hard. We construct a polynomial-time approximation scheme for the dense Steiner tree problem in n-vertex graphs, i.e. for the case when each terminal is adjacent to at least ϵn vertices. We also study the vertex cover problem in ϵ-dense graphs. Though this problem is shown to be still MAX-SNP-hard as in general graphs, we find a better approximation algorithm with the performance ratio $\frac{2}{1+\epsilon}$. The *superdense* cases of all these problems are shown to be solvable in polynomial time.

*Dept. of Computer Science, University of Bonn, 53117 Bonn, and the International Computer Science Institute, Berkeley. Research partially done while visiting Dept. of Computer Science, Princeton University. Research supported by DFG Grant KA 673/4-1, and by the ESPRIT BR Grants 7097 and EC-US 030 and by DIMACS. Email: marek@cs.uni-bonn.de

†Dept. of Computer Science, University of Bonn, 53117 Bonn. Visiting from Institute of Mathematics Akademiei 5, Kishinev 277028, and the Dept. of Computer Science, University of Virginia. Research partially supported by Volkswagen Stiftung and Packard Foundation. Email: alexz@cs.virginia.edu

1 Dense Set Cover Problem

We start with the Dense Set Cover Problem. Let $X = \{x_1, ..., x_k\}$ be a finite set and $P = \{p_1, ..., p_m\} \subseteq 2^X$ be a family of its subsets. The `Set Cover Problem` (SCP) asks for a minimum size sub-family M of P such that $X \subseteq \cup\{p|p \in M\}$.

The greedy heuristic gives $1 + lnk$ approximation for SCP [5]. Moreover, SCP cannot be approximated to within less than $\ln k$-factor unless $NP \subseteq DTIME[n^{loglogn}]$ [6].

The B-*sparse* SCP has a constant upper bound $B > 1$ on the number of sets in P which cover the same element of X. The Vertex Cover Problem is a well-known representative of B-sparse SCP (B=2). There is a simple B-approximation algorithm for this problem. From the other side, the B-sparse SCP is MAX SNP-complete.

In an ϵ-*dense* SCP, any element of X belongs to at least $\epsilon|P|$ sets for some $\epsilon < 1$.

We will analyze the greedy heuristic applied to ϵ-dense SCP. This heuristic repeatedly choose a maximum size set in P, remove its elements from X and all other sets in P. All chosen sets form the output set cover *Greedy*.

Lemma 1 *The size of Greedy is at most* $\log_{1/(1-\epsilon)} k$.

Proof. At first we will show that the maximum size of a set in P is at least ϵk. Consider a bipartite graph $G = (P \cup X, E)$ where $x \in X$ and $p \in P$ are adjacent if and only if $x \in p$. The degree of any $x \in X$ is at least ϵm, so the number of edges in this graph is at least ϵmk and, therefore, there is a set $p \in P$ with degree at least ϵm.

Each iteration of the greedy heuristic does not decrease density, since all elements which belong to the chosen set are removed from X. So the size of X after the ith iteration is at most $(1-\epsilon)^i k$. $\diamondsuit$

This lemma shows that the size of the optimal set cover is $O(\log k)$. So we cannot expect that the ϵ-dense SCP is NP-complete, since a simple $O(m^{O(\log k)})$-time exhaustive search chooses the optimal solution.

Theorem 1 *Unless* $NP \subseteq DTIME[n^{\log n}]$, *the* ϵ-*dense SCP is not* NP-*complete.*

Note that $O(\log k)$ is the tight bound for the performance ratio of the greedy heuristic applied to ϵ-dense SCP. To show this for $\epsilon = \frac{1}{2}$, we can construct an instance of this problem with the size of optimal solution of $O(\log k)$ and then add two sets A and B such that $A \cup B = X$, $A \cap B = \emptyset$. On the other hand, unlike to the general case of SCP, we may decrease the constant factor as far as we want.

Lemma 2 *For any* $c > 0$ *and* $1 > \epsilon > 0$, *there is a* $c \ln k$-*approximation algorithm for* ϵ-*dense SCP.*

Proof. Indeed, let transform an instance of ϵ-dense SCP to an instance of $(1-(1-\epsilon)^2)$-dense SCP in the following way. Consider a family $P^2 = \{p \cup q : p, q \in P\}$. It is easy to see that any solution for SCP with the family P^2 gives a solution for initial SCP. An ϵ-density means that at most $(1-\epsilon)m$ sets do not contain a given element of X. But then at most $(1-\epsilon)^2 m^2$ sets in P^2 do not contain a given element of X.

Lemma 1 implies that such transformation decrease the performance ratio of the greedy algorithm twice. $\diamondsuit$

Theorem 1 arises the following two open problems:

Problem 1 *Can ϵ-dense SCP be solved in polynomial time?*

Problem 2 *Can ϵ-dense SCP be approximated in polynomial time to within constant factor?*

Further densification leads to polynomial solvability of SCP. The *δ-superdense* SCP is the case of SCP where each element of X is covered by at least $m - o(m^\delta)$ sets of P for some $\delta < 1$.

Theorem 2 *The δ-superdense SCP can be solved in polynomial time.*

Proof. Let each element of X is covered by at least $m - \gamma m^\delta = m(1 - \gamma m^{\delta-1})$ sets of P for some $\gamma < m^{1-\delta}$. By Lemma 1 for $\epsilon = 1 - \gamma m^{\delta-1}$, the size of optimal solution is at most

$$\log_{\gamma^{-1} m^{1-\delta}} k = \frac{1}{(1-\delta)(1-\log_m \gamma)} \log_m k.$$

Thus, exhaustive search for finding an exact solution has at most $k^{((1-\delta)\delta)^{-1}}$ cases to consider. $\diamondsuit$

2 Dense Steiner Tree Problem

Consider a connected graph $G = (V, E)$ with a *terminal* set $S \subseteq V$. The `Steiner Tree Problem` (STP) asks for a minimum size tree within G which spans all terminals from S. Further, $d(F)$ denotes the length of a graph F, $|S| = k$ and $|V| = n$. A well-known *minimum spanning tree* heuristic (MSTH) [9] finds a minimum spanning tree M of a weighted complete graph $G' = (S, E', c)$, where the weight of any edge equals to the length of the shortest path between its ends in G. Then MSTH replaces all edges of M with the corresponding paths in G and extracts a tree from the subgraph obtained.

An optimal Steiner tree contains also non-terminals. Each such vertex of degree at least 3 is called a *Steiner point.* It is easy to see that there are at most $k-2$ Steiner points. Using MSTH we can find an optimal Steiner tree if we add all Steiner points to the terminal set.

Remark 1 *An optimal Steiner tree can be found exactly in $O(n^k)$ time.*

MSTH gives 2-approximation for STP [9] and the best up-today polynomial-time approximation guarantee is about 1.644 [7]. From the other side, STP is known to be MAX SNP - complete [4].

In the *B-sparse* STP the degree of any vertex is bounded by a constant B. It is known that STP in the rectilinear metric (a sub-case of 4-sparse STP) is NP-complete but the question whether it is MAX SNP-hard or not is still open.

In an *ϵ-dense* instance of STP (for some $\epsilon < 1$) any terminal has at least ϵn neighbors outside S.

Note that for $\epsilon > \frac{1}{2}$, ϵ-dense STP is a sub-case of Network STP with distances 1 and 2 which is still MAX SNP-complete [4]. The Rayward-Smith heuristic [8] was proposed for the latter problem in [4]. It achieves a better approximation guarantee ($\frac{4}{3}$) then MSTH which has the tight bound 2 as for the general case. MSTH also does not differ the dense and general case of STP.

If the number of terminals is small enough, i.e. $k \leq \frac{1}{\epsilon}$, then we can find an exact solution in polynomial time. Otherwise, we apply to the dense STP the following variant of Rayward-Smith heuristic (or the greedy algorithm [10]).

Algorithm DSTP

(0) $SP \leftarrow \emptyset$;
$\mathcal{C} \leftarrow \{\{s\} : s \in S\}$
(1) while $|\mathcal{C}| > \frac{1}{\epsilon}$ do
find $v \in V \setminus S$ with the maximum size of
$D(v) = \{C \in \mathcal{C} : C$ contains a neighbor of $v\}$
$SP \leftarrow SP \cup v$;
$\mathcal{C} \leftarrow \mathcal{C} \setminus D(v) \cup \{\cup_{C \in D(v)} C\}$;
(2) find an optimal Steiner tree T for a terminal set $S \cup SP$.

Let $\mathcal{C}$ consist of sets $C_1, ..., C_r$ after Step (1) of Algorithm DSTP. Let add edges between all terminals of the same set $C_i, i = 1, ..., r$. The length of the optimal Steiner tree in the graph G' obtained cannot be longer than in G. There is an optimal Steiner tree OPT' in G' containing spanning trees M_i for each set $C_i, i = 1, ...r$. If we contract any such tree M_i to a vertex, then OPT' appears to be an optimal Steiner tree M_0 spanning vertices corresponding to C_i. Thus, the edge set of OPT' is a union of edges of $M_i, i = 0, 1, .., r$.

Algorithm DSTP constructs some Steiner trees M_i' in G for terminals of C_i (step (1)) and then finds the shortest tree M_0' spanning $M_i', i = 1, ..., r$ (step (2)). M_0' cannot be longer that M_0, since M_0 also spans M_i'. Remark 1 implies that an exhaustive search in Step (2) can be executed in time $O(n^{1/\epsilon})$.

An approximation ratio of Algorithm DSTP is at most

$$\frac{\sum_{i=0}^r d(M_i')}{\sum_{i=0}^r d(M_i)} \leq \frac{\sum_{i=1}^r d(M_i')}{\sum_{i=1}^r d(M_i)} = \frac{k - r + |SP|}{k - r} \leq 1 + \frac{|SP|}{k - \frac{1}{\epsilon}}. \quad (1)$$

The size of SP equals to the number of iterations in Step (1). Each iteration of (1) decreases the size of $\mathcal{C}$ by at least $\epsilon|\mathcal{C}| - 1$. Thus, after i-th iteration $|\mathcal{C}| \leq (k - \frac{1}{\epsilon})(1-\epsilon)^i + \frac{1}{\epsilon}$. The procedure (1) interrupts when $|\mathcal{C}| < \frac{1}{\epsilon} + 1$, so

$$|SP| \leq \log_{1/(1-\epsilon)}(k - \frac{1}{\epsilon}).$$

Thus, (1) implies the following

Lemma 3 *An approximation ratio of Algorithm DSTP is at most*

$$1 + \frac{\log_{1/(1-\epsilon)}(k - \frac{1}{\epsilon})}{k - \frac{1}{\epsilon}} \quad . \qquad \diamondsuit$$

Given an arbitrary approximation ratio $1 + \gamma$, $\gamma > 0$, our strategy is to solve exactly in polynomial time (for fixed ϵ and γ) instances of DSTP with small number of terminals, i.e. when k satisfies the following inequality

$$\frac{\log_{1/(1-\epsilon)}(k - \frac{1}{\epsilon})}{k - \frac{1}{\epsilon}} \leq \gamma.$$

If the number of terminals is sufficiently big, then we apply Algorithm DSTP. Thus we obtain the following

Theorem 3 *There is a polynomial-time approximation scheme for the ϵ-dense STP.* $\diamondsuit$

It is not difficult to see that there is a polynomial time reduction of the ϵ-dense SCP to the ϵ-dense STP and vice versa, thus, the problem of polynomial time solvability of ϵ-dense STP is equivalent to Problem 1.

Similarly to SCP, we define δ-*superdense* STP to be the case of STP where any terminal has at least $n - o(n^\delta)$ neighbors outside S.

Corollary 1 *The δ-superdense STP can be solved exactly in polynomial time.*

3 Dense Vertex Cover Problem

Vertex Cover Problem (VCP). *Given a graph $G = (V, E)$, find a minimum size vertex set $OPT \subseteq V$ such that at least one end of any edge belongs to OPT.*

The following algorithm is suggested for VCP in ϵ-dense graphs, i.e., in graphs where any vertex has at least ϵn neighbors for some $\epsilon > 0$ ($|V| = n$). Let $O(v)$ denote the set of neighbors of a vertex v, $G(V')$ denote a subgraph induced by a vertex set $V' \subseteq V$ and 2VC denote the well-known 2-approximation algorithm for VCP.

Algorithm DVC

```
for all v ∈ V
do V' ← V \ (O(v) ∪ {v});
  find a vertex cover VC(v) for G(V') using 2VC;
  VC(v) ← O(v) ∪ VC(v);
APPR ← arg min_{v∈V} |VC(v)|.
```

Let $v \notin OPT$. Then $O(v) \subseteq OPT$ since all edges incident to v should be covered by OPT. Moreover, $O(v)$ covers all edges between $O(v)$ and the corresponding V'. So the rest of vertices of OPT cover the edges of $G(V')$.

Let $OPT' = OPT - O(v)$. The output vertex cover of 2VC applied to V' has a size at most $\min\{2|OPT'|, |V'|\}$. So the approximation ratio can be bounded as follows.

$$\frac{|APPR|}{|OPT|} \leq \frac{|O(v)| + \min\{2|OPT'|, |V'|\}}{|O(v)| + |OPT'|} \leq \min\{\frac{|O(v)| + 2|OPT'|}{|O(v)| + |OPT'|}, \frac{n}{|O(v)| + |OPT'|}\}$$

If $2|OPT'| \leq (1-\epsilon)n$, then

$$\frac{|APPR|}{|OPT|} \leq \frac{\epsilon n + 2|OPT'|}{\epsilon n + |OPT'|} = 2 - \frac{1}{1 + \frac{|OPT'|}{\epsilon n}}$$

Thus, the more $|OPT'|$ corresponds to the more bound for the approximation ratio. Therefore,

$$\frac{|APPR|}{|OPT|} \leq 2 - \frac{1}{1 + \frac{0.5(1-\epsilon)n}{\epsilon n}} = \frac{2}{1+\epsilon}.$$

If $2|OPT'| \geq (1-\epsilon)n$, then we obtain the same bound for the approximation ratio as follows

$$\frac{|APPR|}{|OPT|} \leq \frac{n}{\epsilon n + 0.5(1-\epsilon)n} = \frac{2}{1+\epsilon}.$$

Theorem 4 *The algorithm DVC has an approximation ratio at most* $\frac{2}{1+\epsilon}$ *for* ϵ*-dense graphs.*

Theorem 5 *The* ϵ*-dense Vertex Cover Problem is MAX SNP-hard.*

Proof. (Sketch.) Starting with an instance of the Vertex Cover Problem in a graph G with n vertices we dencify it joining all vertices of a clique of size $\frac{\epsilon}{1-\epsilon}n$ with all vertices of G. The resulting graph is ϵ-dense and, therefore, if we have an α-approximation for DVC, then the reduction above gives $\alpha(1+\epsilon)$-approximation algorithm for the general problem which is MAX SNP-hard. $\diamondsuit$

Further densification (as for SCP and STP) leads to decreasement of approximation complexity.

We say that an instance of VCP is δ-superdense if the degree of any vertex is at least $n - o(n^\delta)$. Theorem 4 implies

Corollary 2 *The δ-superdense VCP has a polynomial-time approximation scheme.*

Acknowlegdement

We thank Sanjeev Arora and Dorit Hochbaum for number of interesting discussions.

References

[1] S. Arora, D. Karger, and M. Karpinski. Polynomial Time Approximation Schemes for Dense Instances of NP-Hard Problems.In *Proc. of 27th ACM Symp. on Theory of Computing*, 284–293, 1995.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and hardness of approximation problems. In *Proc. of 33d Annual IEEE Symp. on Foundations of Computer Science*, 14–23, 1992.

[3] M. Bellare, S. Goldwasser, C. Lund, and A. Russel. Efficient probabilistically checkable proofs and application to approximation. In *Proc. of 25th Annual ACM Symp. on Theory of Comp.*, 294–304, 1993.

[4] M. Bern and P. Plassmann. The Steiner problems with edge lengths 1 and 2. *Inform. Process. Lett.* 32: 171–176, 1989.

[5] Chvatal. A greedy-heuristic for the set-covering problem. *Math. Operations Res.* 4: 233–235, 1979.

[6] U. Feige. A threshold of $\ln n$ for approximating set cover. In *Proc. of 28th ACM Symp. on Theory of Comp.*, 314-318, 1996.

[7] M. Karpinski and A. Zelikovsky. New approximation algorithms for the Steiner tree problem. *Journal of Combinatorial Optimization* 1: 1–19, 1997.

[8] V. J. Rayward-Smith, The computation of nearly minimal Steiner trees in graphs, *International J. Math. Ed. Sci. Tech.* 14: 15–23, 1983.

[9] H. Takahashi and A. Matsuyama. An approximate solution for the Steiner problem in graphs. *Math. Japonica*, 24: 573–577, 1980.

[10] A. Z. Zelikovsky. An 11/6-approximation algorithm for the network Steiner problem. *Algorithmica* 9: 463–470, 1993.