<table>
<thead>
<tr>
<th>項目</th>
<th>内容</th>
</tr>
</thead>
<tbody>
<tr>
<td>タイトル</td>
<td>私的近似化のセットカバー問題：理論と応用への応用拡張 (Theory of Computer Science and Its Applications)</td>
</tr>
<tr>
<td>著者</td>
<td>Yashiro, Masatoshi; Tanaka, Keisuke</td>
</tr>
<tr>
<td>引用</td>
<td>数理解析研究所講究録 (2007), 1554: 48-55</td>
</tr>
<tr>
<td>発行日</td>
<td>2007-05</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/2433/80973">http://hdl.handle.net/2433/80973</a></td>
</tr>
<tr>
<td>形式</td>
<td>Departmental Bulletin Paper</td>
</tr>
<tr>
<td>テキストバージョン</td>
<td>publisher</td>
</tr>
<tr>
<td>右</td>
<td>Kyoto University</td>
</tr>
</tbody>
</table>
Abstract — Private approximation, introduced by Feigenbaum, Ishai, Malkin, Nissim, Strauss, and Wright, allows us to find approximate solutions with disclosing as little information as possible. In STOC 2006, Beimel, Carmi, Nissim, and Weinreb studied the private approximation for both the vertex cover and the max exact 3SAT problems. In this paper, we consider the set cover problem where the costs of all sets are polynomially bounded. We show that there exists neither a deterministic nor a randomized private approximation. We also consider the case that the frequencies of all elements are equal. We show that in this case there exist no deterministic private approximation.

Keywords: private approximation, set cover problem.

1 Introduction

Private approximation is an algorithm which is more efficient than exact computation and that maintain the privacy of the data, that is, the output of private approximation does not leak any information of the input.

Feigenbaum, Ishai, Malkin, Nissim, Strauss, and Wright [?] introduced the notion of the private approximation of functions. Roughly speaking, an approximation function $\hat{g}$ is called private approximation with respect to the target function $g$, if $\hat{g}(x)$ reveals no more information about $x$ than $g(x)$ does. More formally, there exists a probabilistic polynomial time simulator $\hat{M}$ such that the distribution of the simulation output $\hat{M}(g(x))$ is indistinguishable from $\hat{g}(x)$. They proposed a function (two-party protocol) which is the private approximation with respect to that for computing the hamming distance between two binary vectors. They also proposed the private approximations of several natural $\#P$-hard problems. After [?], several private approximations were proposed [?, ?]?.

Approximation algorithm is currently one of the main research fields in computer science. The design of algorithms for approximating $NP$-hard problems has attracted substantial attention in the last few decades, as did the research on proving hardness of approximation. Halevi, Krauthgamer, Kushilevitz, and Nissim [?] discussed the private approximation of NP-hard problems. They proved that there exists no private approximation for computing the size of minimum vertex cover within approximation ratio $n^{1-\epsilon}$. Their proof used the sliding-window reduction that translates a SAT instance $\phi$ to an instance $G$ of the vertex cover problem. If $\phi$ is satisfiable then $G$ has the vertex cover of size $z$, otherwise any vertex cover for $G$ is of size at least $z + 1$. The definition of the private approximation in [?] is almost the same as that by Feigenbaum, Ishai, Malkin, Nissim, Strauss, and Wright [?].

Beimel, Nissim, Carmi, and Weinreb [?] studied the private approximation of both the vertex cover and the max exact 3SAT problems, and Beimel, Hallak, and Nissim [?] studied the private approximation of both the vertex cover and the clustering problem. In order to consider search problems, Beimel et al. [?] proposed a definition of the private approximation which is different from that in [?]. In their definition, an algorithm $A$ is a private approximation with respect to a privacy structure $\mathcal{R}$, which is an equivalent relation, if the outputs of executing $A$ on two $\mathcal{R}$-equivalent inputs are computational indistinguishable. Under their definition, they showed that there exists neither a deterministic nor a randomized private approximation of the search problem for a minimum vertex cover within
approximation ratio \( n^{1-\epsilon} \).

In this paper, we consider the private approximation of the set cover problem. The vertex cover problem which studied by [? ,?] is essentially the special case of the set cover problem where the frequency of all elements are equal to 2. Therefore, by the result of Beimel et al. [?], we can see there exists no private approximation of the set cover problem where the frequency of all elements are equal to 2. However, in the other case, that is, the frequencies of some elements are not equal to 2, it is not clear whether there exists private approximation of the set cover problem.

In this paper we consider the private approximation of the set cover problem. In the previous paper [? ,?], only the vertex cover problem whose costs of all not fixed that vertices are fixed. In particular, we consider the set cover problem where the costs of all sets are polynomially bounded. We show that there exists neither a deterministic nor a randomized private approximation. We also consider the case that the frequencies of all elements are equal. We show that in this case there exist no deterministic private approximation.

Due to lack of space, the proofs are omitted from this paper. See the full version [?].

2 The Set Cover Problem

In this section, we describe the set cover problem and the frequency.

Definition 2.1 (Set Cover Problem). Let \( U \) be a set of \( m \) elements, \( S = \{S_1, \ldots, S_n\} \) a collection of subsets of \( U \), and \( c : S \to \mathbb{Q}^+ \) a cost function. We say the set \( C \subseteq \{1, \ldots, n\} \) of indices is called a cover of \( U \) if the collection of \( S_i \) (\( i \in C \)) covers all elements in \( U \), that is, \( \bigcup_{i \in C} S_i = U \). Given \( (U, S, c) \), the set cover problem is to find a minimum cost cover of \( U \).

Usually, the size of the instance \((U, S, c)\) of the set cover problem is considered as the number of the elements in \( U \). In this paper, we consider the size of the instance of the set cover problem as the number of sets in \( S \). Therefore, the number of the elements in each set in \( S \) is restricted to polynomial of the number of the sets in \( S \).

In this paper, we consider "polynomial-cost set cover problem" where the cost of each set is polynomial in the problem size. Let "Set Cover" be a polynomial-cost set cover problem.

Definition 2.2 (Frequency). We define the frequency of an element to be the number of sets the element is in. A useful parameter is the frequency of the most frequent element. Let us denote this by \( f \).

We call the problem where all elements in \( U \) have the equivalent frequency as "set cover problem with fixed frequency".

3 The Approximation and the Private Algorithm

First, we describe the definition of the approximation. The following definition of the approximation can be applied to minimization problems. The definition for maximization problems is similar.

Definition 3.1 (Approximation of the Search Problem). Let \( g \) be a function, \( A \) an algorithm for a search problem, and \( c \) a cost function. We say that \( A \) is an \( \alpha \)-approximation of \( g \) if it runs in polynomial time and for all input \( x \),

\[
\sum_{y \in A(x)} c(y) \leq \alpha \sum_{y \in g(x)} c(y).
\]

Next, we describe the definition of the private algorithm, following [?]. We describe the privacy structure which is necessary to define the private algorithm.

Definition 3.2 (Privacy Structure). A privacy structure \( \mathcal{R} \subseteq \{0,1\}^* \times \{0,1\}^* \) is an equivalence relation on instances. For \((x, y) \in \mathcal{R}, \) we use the notation \( x \equiv \mathcal{R} y \).

We only discuss on the privacy structures of the form \( \mathcal{R} = \bigcup_{n \in \mathbb{N}} \mathcal{R}_n \), where \( \mathcal{R}_n \) is an equivalence relation among the instances of size \( n \), such as \( S \) with \( n \) sets.

We now define the private algorithm. We say that an algorithm \( A \) is private with respect to a privacy structure \( \mathcal{R} \) if the results of executing \( A \) on two \( \mathcal{R} \)-equivalent inputs are computationally indistinguishable.

Definition 3.3 (Private Algorithm). Let \( \mathcal{R} \) be a privacy structure. A probabilistic polynomial-time algorithm \( A \) is private with respect to \( \mathcal{R} \) if for every polynomial-time algorithm \( D \) and for every positive polynomial \( p(\cdot) \), there exists some \( n_0 \in \mathbb{N} \) such that for every \( x, y \in \{0,1\}^* \), \( x \equiv \mathcal{R} y \), and \( |x| = |y| \geq n_0 \),

\[
\left| \Pr[D(A(x), x, y) = 1] - \Pr[D(A(y), x, y) = 1] \right| \leq \frac{1}{p(|x|)}.
\]

That is, when \( x \equiv \mathcal{R} y \), any algorithm \( D \) cannot distinguish if the input of \( A \) is \( x \) or \( y \).

Next, in order to define the private approximation of the search problem, we define the privacy structure, following [?]. We can regard the decision and the search problems as follows by using the bivariate relation.
4.1 Definitions

In this section, we describe some definitions.

First, we describe the definition of the private approximation of Set Cover.

Definition 4.1 (Private Approximation of the Set Cover Problem). An algorithm $A$ is a private $\alpha$-approximation algorithm for minSC if:

- $A$ is a $\alpha$-approximation algorithm for minSC, and
- $A$ is private with respect to $R_{\minSC}$.

In order to analyze the private approximation of the vertex cover problem, Beimel et al. [?] employed “critical vertices” and “relevant vertices”. We also employ the notion of “critical” and “relevant” for Set Cover.

Definition 4.2 (Critical Set and Relevant Set). Let $U$ be a set of $m$ elements, $S = \{S_1, S_2, \ldots, S_n\}$ a collection of sets and $c$ a cost function of $S$. We say that $S_i$ is critical for $(U, S, c)$ if every minimum set cover of $(U, S, c)$ contains $S_i$. We say that $S_i$ is relevant for $(U, S, c)$ if there exists at least one minimum set cover of $(U, S, c)$ that contains $S_i$.

Next, we present the problem related to Definition 4.1.

Definition 4.3 (The Relevant Set / Non-Critical Set Problem).

Input: a Set $U$, a collection $S = \{S_1, S_2, \ldots, S_n\}$ of sets, and a cost function $c$.

Output: “$S_i$ is relevant for $(U, S, c)$” or “$S_i$ is non-critical for $(U, S, c)$”.

We next define two special set cover problems. When we construct the algorithm for the Relevant / Non-Critical Set problem in Section ??, they are helpful.

Definition 4.4 ($U^2$, $S^2$, $c^2$) and $(U(t, u), S(t, u), c(t, u))$. Let $U$ be a set that contains $m$ elements $e_1, e_2, \ldots, e_m$, $S = \{S_1, \ldots, S_n\}$ a collection of subsets of $U$, $c$ a cost function $S \rightarrow \mathbb{Q}^+$, $I$ a collection of empty sets. For $(U, S, c)$ and for any $S_i \in I$ and $S_i \in S$, we define $(U^2, S^2, c^2)$ and $(U(t, u), S(t, u), c(t, u))$ as follows.

The collection $S^2$ of sets is defined as $S^2 = \{S_1, \ldots, S_{2m}\} \cup I$ where $S_{i+n} := \{e_{n+i} | e_i \in S_i\}$. The set $U^2$ is defined as $U^2 = \{e_1, \ldots, e_{2m}\}$.

The function $c^2$ is defined as $c^2(S_i) = c(S_i)$ (for $1 \leq i \leq n$), $c^2(S_{i+n}) = c(S_{i-1})$ (for $1 \leq i \leq 2n$), and $c^2(S_i) = 1$ (for $S_i \in I$).

The collection $S(t, u)$ of sets is defined as $S(t, u) = \{S_1, \ldots, S_n\} \cup I$ where $S_i = S_i \cup \{e^*, e^{**}\}$, $S_u = S_u \cup \{e^*\}$, and $S_{u+n} = S_{u+n} \cup \{e^{**}\}$ for some
4.2 Proofs

In this section, we show that there exists no private approximation algorithm of Set Cover with respect to $\mathcal{R}_{\text{minSC}}$ if $P \neq NP$.

**Theorem 4.5.** Let $\epsilon > 0$ be a constant and $f$ a frequency. If $P \neq NP$, then there is no deterministic private $f^*$-approximation algorithm of the search problem of minSC.

This proof is similar to that in [?]. The outline of the proof is as follows:

1. We construct a Relevant or Non-Critical for Set Cover algorithm from the private approximation algorithm $A$ with respect to $\mathcal{R}_{\text{minSC}}$.
2. We construct a greedy algorithm that efficiently solves the NP-hard problem from the Relevant or Non-Critical for Set Cover algorithm.
3. If $P \neq NP$, this is a contradiction. Thus there is no private approximation algorithm $A$ for Set Cover with respect to $\mathcal{R}_{\text{minSC}}$.

In Algorithm ??, we describe a greedy algorithm of Set Cover given an access to the algorithm which decides relevant or non-critical (we call this algorithm Relevant or Non-Critical for Set Cover). We will show that the algorithm Relevant or Non-Critical for Set Cover can be constructed by using oracle access to private approximation algorithms of Set Cover later on.

**Algorithm 1 (Greedy Minimum Set Cover)**

Input: a collection of sets $S = \{S_1, \ldots, S_n\}$, a cost function $c : S \to \mathbb{Q}^+$, and a set $U$ of $m$ elements.

1. Set $C_u = \emptyset$.
2. If $U = \emptyset$ return $C_u$.
3. Pick a set $S_i \in S$ and execute the algorithm Relevant or Non-Critical for Set Cover on $(U, S, c)$ and $S_i$.
4. If the answer is "Relevant",
   (a) Delete all the elements included in $S_i$ from both $U$ and sets $S_j$ in $S$.
   (b) $S \leftarrow S \setminus \{S_i\}$.
   (c) $C_u \leftarrow C_u \cup \{i\}$.
   (d) Go to STEP2.
5. If the answer is "Non-Critical",
   (a) $S \leftarrow S \setminus \{S_i\}.$
   (b) Go to STEP2.

The following claim shows the correctness of the greedy algorithm.

**Claim 4.6.** If the algorithm Relevant or Non-Critical for Set Cover is polynomial and correct then the algorithm Greedy Minimum Set Cover is polynomial and correct.

We next construct the Relevant or Non-Critical for Set Cover algorithm from a private approximation algorithm for minSC. We adopt the idea of [?].

**Claim 4.7.** Let $S = \{S_1, S_2, \ldots, S_n\}$ be a collection of sets, $c$ a cost function $S \to \mathbb{Q}^+$, $e^*$ an element such that $e^* \notin U$. We choose $i, j \in \{1, \ldots, n\}$ arbitrary $i \neq j$, and define $S^* = \{S_1^*, \ldots, S_n^*\}$ where $S_i^* = S_i \cup \{e^*\}$, $S_j^* = S_j \cup \{e^*\}$, and $S_k^* = S_k$ for $k \neq i, j$. We also define $U^* = U \cup \{e^*\}$ and $c^*(S_i^*) = c(S_i^*)$.

Then, If $S_j$ is critical for $(U, S, c)$, then $(U, S, c) \equiv_{\mathcal{R}_{\text{minSC}}} (U^*, S^*, c^*)$.

We can prove the two claims with respect to $(U^2, S^2, c^2)$ and $(U^{(t,u)}, S^{(t,u)}, c^{(t,u)})$ defined in Definition ?? We use these claims for the proof of the correctness of Algorithm ??.

**Claim 4.8.** If $S_u$ is critical for $(U, S, c)$, then $(U^2, S^2, c^2) \equiv_{\mathcal{R}_{\text{minSC}}}(U^{(t,u)}, S^{(t,u)}, c^{(t,u)})$.

Claim 4.9. If $S_u$ is not relevant for $(U, S, c)$, then $S_i$ is critical for $(U^{(t,u)}, S^{(t,u)}, c^{(t,u)})$.

Next, by using a private $f^*$-approximation algorithm we describe the Relevant or Non-Critical for
Algorithm 2 (Relevant or Non-Critical for Set Cover)

Input: \((U, S, c, S_u)\)
1. Let \(I\) be a set of \(2df^* + 1\) empty sets. (where \(d = \sum c(S_i)\))
2. Construct the collection of sets \(S^2\) from \(S^1\) and \(I\).
3. Execute \(A\) on \((U^2, S^2, c^2)\) and get the output \(W^2\) of \(A\).
4. Choose any set \(S_t \in I \setminus W^2\).
5. Construct the collection of sets \(S_{(t,u)}\) from \(S, I, S_t, S_u\).
6. Execute \(A\) on \(S_{(t,u)}\), and get the output \(W_{(t,u)}\) of \(A\).
7. If \(W^2 \neq W_{(t,u)}\), return "Non-Critical". Else return "Relevant".

We can show the following claim.

Claim 4.10. Let \(A\) be a deterministic private approximation algorithm for minSC, \(U\) a set of \(m\) elements, \(S = \{S_1, \ldots, S_n\}\) a collection of subsets of \(U\), and \(c\) a cost function, and denote \(A((U, S, c))\) be a cover of \((U, S, c)\) that corresponding to indices outputted by \(A\). Then for any set \(S_t \in S \setminus W\), the set \(S_t\) is not critical for \((U, S, c)\).

We must prove Algorithm ?? is correct and running time is polynomial. We can prove the correctness by proving the following two claims. In the proofs of the following claims, we use Claim ??.

Claim 4.11. If \(W^2 \neq W_{(t,u)}\), then \(S_u\) is not critical for \((U, S, c)\).

Claim 4.12. If \(W^2 = W_{(t,u)}\), then \(S_u\) is relevant.

Finally, we show that there is a set chosen in STEP 4 of Algorithm ??.

Claim 4.13. Let \(\epsilon \geq 0\). There is a set \(S_t \in I\) such that \(S_t \in I \setminus W^2\).

Therefore we proved Theorem ??.

Remark 4.14. Since \(d\) is polynomial in the size of problem, the sizes of \(S^2\) and \(S_{(t,u)}\) are polynomial in the size of problem. Therefore we prove Theorem ??.

5 Randomized Private Approximation of the Set Cover Problem

In this section, we show that there exists no randomized private approximation algorithm of Set Cover. The outline of the proof is similar to that in ??). We execute the approximation algorithm \(k\) times to decide whether the set is Relevant or Non-Critical. We prove several claims that correspond to those in Section ??.

We use the Algorithm ??, and we use Algorithm ?? as Relevant or Non-Critical for Set Cover in ??.

Algorithm 3 (Randomized Relevant or Non-Critical for Set Cover)

Input: \((U, S, c, S_u)\).
1. If \(S\) contains less than \(n_2\) sets (where \(n_2 = \max\{n_0, n_1\}\)), then find if \(S_u\) is relevant for \((U, S, c)\) or non-critical for \((U, S, c)\) using exhaustive search.
2. Let \(I\) be a set \(4df^* + 2\) sets. (where \(d = \sum c(S_i)\))
3. Construct the family of sets \((U^2, S^2, c^2)\) from \((U, S, c)\) and \(I\) as in Definition ??.
4. Execute \(k\) times the algorithm \(A\) on \((U^2, S^2, c^2)\).
5. Choose a set \(S_t \in I\) such that \(S_t\) appears at most \(k/2\) times in \(A((U^2, S^2, c^2))\) in the \(k\) executions.
6. Construct the family of sets \((U_{(t,u)}, S_{(t,u)}, c_{(t,u)})\) from \((U, S, c)\), \(I\), \(S_t\), and \(S_u\) as in Definition ??.
7. Execute \(k\) times Algorithm \(A\) on \(S_{(t,u)}\).
8. If \(t \in A((U_{(t,u)}, S_{(t,u)}, c_{(t,u)}))\) in at least \(0.75k\) of the \(k\) executions, then return "Non-Critical". Else return "Relevant".

We can show the following claims.

Claim 5.1. There is a set \(S_t \in I\) such that index \(t\) appears at most \(k/2\) times in \(A((U^2, S^2, c^2))\) out of the \(k\) executions.

Claim 5.2. There exists a constant \(n_1\) such that if

- \((U^2, S^2, c^2)\) contains at least \(n_1\) sets,
- \(\text{Pr}[t \in A((U^2, S^2, c^2))] < 0.55\), and
- \(\text{Pr}[t \in A((U_{(t,u)}, S_{(t,u)}, c_{(t,u)}))] > 0.6\),

then \(S_u\) is not critical for \((U, S, c)\).

Claim 5.3. If \(\text{Pr}[\{t, u\} \cap A(U, S, c)] = 0 \leq 0.8\) then \(S_u\) is relevant for \((U, S, c)\).

Claim 5.4. Let \(k > 2^{O(\log(df^*)}\). Algorithm Randomized Relevant or Non-Critical for Set Cover returns the correct answer with probability \(1 - 2^{-O(k)}\).
From the above claims, we can prove the following main theorem.

**Theorem 5.5.** Let $\epsilon > 0$ be a constant. If $RP \neq NP$, then there is no randomized private $f^*$-approximation algorithm for Set Cover.

**proof.** By Claim ?? and Claim ??, if there is a randomized private $f^*$-approximation algorithm for Set Cover, then there is a randomized algorithm for the exact search problems for minSC. This algorithm is transformed to the algorithm for decision problem of Set Cover (given $(U, S, c)$ and $x \in Q^+$, decide whether there is a cover of cost at most $x$). Since this problem is NP-complete, it contradicts $RP \neq NP$. □

6 Private Approximation of the Set Cover Problem with the Fixed Frequency

When the frequency of all elements $U$ are 2, the set cover problem is essentially the same as the vertex cover problem. Therefore, in this section, we consider the situation that the frequencies of all $e \in U$ are equal.

We show that there exists no randomized private approximation algorithm of Set Cover with the fixed frequency. The strategy of the proof is similar to that in the previous section, however, the construction of the greedy algorithm depends on whether the cost is fixed or not.

First, we describe the greedy algorithm for the case that the cost is not fixed. In this case, it is easy to construct the greedy algorithm.

**Algorithm. 4 (Greedy Minimum Set Cover with fixed frequency (cost is not fixed))**

Input: a collection of sets $S = \{S_1, \ldots, S_n\}$, a cost function $c: S \rightarrow Q^+$, and a set $U$ of $m$ elements.

1. Set $C_* = \emptyset$.
2. If $U = \emptyset$ return $C_*$.
3. Pick an element $e_i \in U$ and make a list of all sets that include $e_i$. We define this list as $L = \{L_1, \ldots, L_m\}$.
4. Set $c' \leftarrow c$ and $j = 1$.
5. Execute Algorithm Relevant or Non-Critical for Set Cover with fixed frequency on $(U, S, c)$ and $S_j$ (where $S_j = L_j$).
6. If the answer is "Relevant"
   (a) Delete all the elements included in $S_j$ from both $U$ and sets $S_i$ in $S$.
   (b) $S \leftarrow S \setminus \{S_j\}$.
   (c) $c \leftarrow c'$.
   (d) $C_* \leftarrow C_* \cup \{j\}$.
   (e) Go to STEP 2.
7. If the answer is "Non-Critical"
   (a) $c(S_j) \leftarrow c(S_j) + 1$.
   (b) $j \leftarrow j + 1$ and go to STEP 5.

We can show that this algorithm is polynomial and correct.

**Claim 6.1.** If the algorithm Relevant or Non-Critical for Set Cover with fixed frequency is polynomial and correct and the cost of each set is not fixed then the algorithm Greedy Minimum Set Cover with fixed frequency is polynomial and correct.

In Algorithm ??, if the cost is fixed, we can not execute STEP 6-a. Therefore we transform from Algorithm ?? to Algorithm ??.

**Algorithm. 5 (Greedy Minimum Set Cover with fixed frequency (cost is fixed))**

Input: a collection of sets $S = \{S_1, \ldots, S_n\}$, a cost function $c: S \rightarrow Q^+$, and a set $U$ of $m$ elements.

1. Set $C_* = \emptyset$.
2. If $U = \emptyset$ return $C_*$.
3. Pick an element $e_i \in U$ and make a list of all sets that include $e_i$. We define this list as $L = \{L_1, \ldots, L_l\}$.
4. Set $S' \leftarrow S$, and $j = 1$.
5. If $j \leq l$,
   (a) Execute Algorithm Relevant or Non-Critical for Set Cover with fixed frequency on $(U, S, c)$ and $S_j$ (where $S_j = L_j$).
   (b) If the answer is "Relevant"
      i. Delete all the elements included in $S_j$ from both $U$ and all sets $S_i$ in $S$.
      ii. $S \leftarrow S \setminus \{S_j\}$.
      iii. $C_* \leftarrow C_* \cup \{j\}$.
      iv. Go to STEP 2.
   (c) If the answer is "Non-Critical"
      i. If $|S_i| = 1$
         A. $j \leftarrow j + 1$ and go to STEP 5.
      ii. Else
         A. Divide $S_i$ into $|S_i| = k$ sets $S_{i_1}, \ldots, S_{i_k}$.
         B. $S \leftarrow (S \setminus S_i) \cup S_{i_1} \cup \ldots \cup S_{i_k}$.
         C. $j \leftarrow j + 1$ and go to STEP 5.
6. If $j = l$,
   (a) $C_* \leftarrow C_* \cup h$ where $S_h = L_1$.
   (b) $S \leftarrow S \setminus \{S_h\}$.
   (c) Go to STEP 2.

We can show that this algorithm is polynomial and correctness.
Claim 6.2. If Algorithm Relevant or Non-Critical for Set Cover with fixed frequency is polynomial and correct and the cost of each set is fixed, then Algorithm Greedy Minimum Set Cover with fixed frequency is polynomial and correct.

Next we consider the algorithm Relevant or Non-Critical for Set Cover with fixed frequency. This is the Algorithm ?? where \((U(t, u), S(t, u), c(t, u))\) is replaced with \(\langle U(t, u), S(t, u), c(t, u) \rangle \) of \(S(t, u) \cup I\) where \(S_T = \{e^*, e^{**}\}\), \(S_u = S_u \cup \{e^*\}\), \(S_u+n \cup \{e^{**}\}\), \(S_k = \{e^*\} (1 \leq i \leq f - 1)\), and \(S_k = \{e^{**}\} (f \leq i \leq 2(f - 2))\) such that \(e^*, e^{**} \notin U^2\). The set \(U(t, u) \cup I\) defined as \(U(t, u) \cup I = U^2 \cup \{e^*, e^{**}\}\), and \(c(t, u) = 2(f - 2)\).

We can easily see \(\langle U^2, S^2, c^2\rangle\) and \(\langle U(t, u) \cup I, S(t, u) \cup I\rangle\) are Set Cover with fixed frequency. Finally, we can prove Claims ??, ??, and ?? and ?? in a similar way as those for the proof of Claims ??, ??, and ??, respectively.

The following two claims are used in the proofs of Claim ?? and Claim ??.

Claim 6.4. If \(S_u\) is critical for \(\langle U, S, c\rangle\), then \(\langle U, S, c\rangle \equiv R_{\text{minSCfixed}} \langle U(t, u) \cup I, S(t, u) \cup I\rangle\).

Claim 6.5. If \(S_u\) is not relevant for \(\langle U, S, c\rangle\), then \(S_u\) is critical for \(\langle U(t, u) \cup I, S(t, u) \cup I\rangle\).

The following two claims guaranteed the correctness of Algorithm ??.

Claim 6.6. If \(W^2 \neq W(t, u, k_1, ..., k_{2(f-2)})\) then \(S_u\) is not critical for \(\langle U, S, c\rangle\).

Claim 6.7. If \(W^2 = W(t, u, k_1, ..., k_{2(f-2)})\) then \(S_u\) is relevant.

We can prove the following theorem from the above claims and Claim ??.

Theorem 6.8. Let \(\epsilon > 0\) be a constant and \(f\) a frequency. If \(P \neq NP\), then there is no deterministic private \(f^*\)-approximation algorithm of the search problem for minSCfixed.

7 Concluding Remarks

In this paper, we have considered the set cover problem where the costs of all sets are polynomially bounded. We have shown that there exists neither a deterministic nor a randomized private approximation. We have also considered the case that the frequencies of all elements are equal. We have shown that in this case there exist no deterministic private approximation.

In this paper, we have proved only when the size of the problem is defined as the number of the sets. It might be interesting to consider the problem where the size of the problem is defined as the number of elements. It might be also interesting to consider whether NP-hard problems other than the set cover problem have the private approximation algorithms or not.

Halevi et al. ?? discussed the leakage of information about the approximation algorithms for the minimum set cover problem. Beimel et al. ?? also discussed that for the vertex cover and the exact 3SAT problems. It might be interesting to consider the leakage of information about the approximation algorithms for the minimum set cover problem.

References


