SAT Reduces to the Minimum Circuit Size Problem with a Random Oracle

Tingqiang Xu, Yusi Chen, Jingyi Lyu

IIS, THU

2024/06/03

1 Background

2 Main Result

3 Intuition

4 Proof Sketch

5 Discussion

6 References

1 Background

2 Main Result

3 Intuition

4 Proof Sketch

5 Discussion

6 References

Minimum Circuit Size Problem (MCSP)

The $\mathcal{O}$**-oracle Minimum Circuit Size Problem** is defined by:

- **Input:** A Boolean function $f : \{0,1\}^n \to \{0,1\}$.
- **Output:** The minimum s such that an $\mathcal{O}$-oracle circuit[1] computing f whose size is at most s exists.

Often denoted by $\mathrm{MCSP}^{\mathcal{O}}$.

[1] Fan-in 2 Boolean circuit with access to circuit family $\{G_n\}$ that computes $\mathcal{O}$ over $\{0,1\}^*$.

Why Care about MCSP?

MCSP is mysterious:

- NP-Complete?
- Hard to approximate / Hard on average?
- Has **any** non-trivial algorithm?

and useful:

- Has connections to structural complexity, cryptography......
- MCSP is NP-Complete $\Rightarrow$ EXP $\neq$ ZPP
- (Some version of) MCSP is hard on average $\Rightarrow$ OWF exists

Bounded Frequency Set Cover

The τ-**Frequency Set Cover Problem** is defined by:

- **Input:** Subsets $S_1, \cdots, S_m \subset [n]$ with property that for all $i \in [n]$, i appears in those subsets exactly τ times.
- **Output:** The minimum θ such that $J \subset [m]$ exists with $\bigcup_{j \in J} S_j = [n]$ and $|J| \leq \theta$.

1. Background
2. Main Result
3. Intuition
4. Proof Sketch
5. Discussion
6. References

Main Result (Informal)

Approximating Bounded Frequency Set Cover is known NP-hard [DGKR03].

This paper: Deterministic, uniform, polynomial-time reduction with access to **random oracle** $\mathcal{O}$ from Bounded Frequency Set Cover to $\mathsf{MCSP}^{\mathcal{O}}$.

Hardness Result of [DGKR03]

Theorem [DGKR03]: Let τ be any sufficiently large integer constant. Given an instance of τ-Frequency Set Cover over universe $[n]$ (n is a power of two[2]), it is NP-hard to distinguish:

- YES case: There is a set cover of size $\frac{2}{\tau}n$;
- NO case: There is no set cover within size $\frac{n}{3}$.

We refer this promise problem as **Gap τ-Frequency Set Cover**.

[2] Can be assumed universally by padding argument.

Hardness Result of This Paper

Theorem [Ila23]: There is a deterministic polynomial-time algorithm A such that with probability $1 - 2^{-\Omega(n)}$ over the choice of random oracle $\mathcal{O}$, $A^{\mathcal{O}}$ is a many-one reduction from Gap τ-Frequency Set Cover[3] over universe $[n]$ to the promise problem of MCSP, formulated as:

- **Input:** A binary string $x \in \{0,1\}^*$.
- **Output YES** if $\mathrm{MCSP}^{\mathcal{O}}(x) \leq \theta(|x|)$;
- **Output NO** if $\mathrm{MCSP}^{\mathcal{O}}(x) \geq \theta(|x|) + \Omega(\frac{|x|}{\log |x|})$.

where θ is some function of $|x|$.

[3] Equivalent to reduction from SAT regarding the previous theorem.

Reduction

Input: An instance $S_1, \cdots, S_m \subset [n]$ of Gap τ-Frequency Set Cover for some power-of-two n.

Parameter: Secret key length $\lambda(n)$ computable in time $O(\log \lambda(n))$.

Oracle: For all powers-of-two n, we have random oracle[4] $\mathcal{O}_n : [n] \times \{0,1\}^\lambda \times \{0,1\}^{2\lambda} \to \{0,1\}^\lambda$.

[4] This is easy to produce given a general random oracle $\{0,1\}^* \to \{0,1\}$.

Reduction (Cont'd)

Algorithm:

1. Pick $sk_1, \cdots, sk_m \in \{0,1\}^\lambda$ uniformly at random.
2. For all $i \in [n]$, pick $v_i \in \{0,1\}^\lambda$ uniformly at random.
3. For all $i \in [n]$ and $k \in [\tau]$, let $c_{i,k}$ be a uniformly random element of the set $\{c \in \{0,1\}^{2\lambda} : \mathcal{O}_n(i, sk_j, c) = v_i\}$, where j is the index of the k-th set containing i.[5]
   - By rejection sampling.
4. Output the $4\tau n\lambda$-bit truth table of function $f : [n] \times [\tau] \times \{0,1\} \times [2\lambda] \to \{0,1\}$ given by

$$f(i,k,b,d) = \begin{cases} d\text{'th bit of } c_{i,k} & b = 0 \\ d\text{'th bit of } v_i & b = 1 \end{cases}$$

[5] i.e., $j = \mathrm{Index}(i,k)$.

Understand the Reduction

- sk_j: **secret key**; v_i: **message**.
- $c_{i,k}$: a random "**encoding**" of an "**encryption**" of message v_i, using the secret key sk_j.
- One might hope the "optimal" way to compute f is to memorize all $c_{i,k}$ as well as all secret keys sk_j in an **optimal set cover**, then decrypt to find values of v_i.
- A **deterministic** reduction since all randomness come from a general random oracle, invoked in some specific way.

1 Background

2 Main Result

3 Intuition

4 Proof Sketch

5 Discussion

6 References

Why This Works?

Before introducing the proof of reduction (maybe we won't have such time), it is important to see the intuition behind this construction.

MCSP is one of **meta-complexity** problem, and another important family of meta-complexity problem is **Kolmogorov complexity**. In [Ila23], the author proved similar result, and actually get inspiration from some variation of Kolmogorov complexity.

Kolmogorov Complexity

The t-**time bounded Kolmogorov complexity** is defined by

- **Input:** $x \in \{0,1\}^n$.
- **Output:** The minimum description length of Turing machine M, such that M running on empty input outputs x within time t.

Sometimes we consider another string $y \in \{0,1\}^*$, we say the **conditional Kolmogorov complexity** of x given y is defined same way *but with y as input of M*.
Often denoted by $\mathsf{K}^t(x)$ or $\mathsf{K}^t(x|y)$.

Symmetry of Information

Information theory: $H(XY) = H(Y) + H(X|Y)$.
Kolmogorov complexity: $\mathsf{K}^{\infty}(xy) \approx \mathsf{K}^{\infty}(y) + \mathsf{K}^{\infty}(x|y)$. [ZL70]
Approximating conditional version of K^t is NP-**hard** [HIR23], then we expect such association also holds for K^t, i.e.

$$\mathsf{K}^t(xy) \approx \mathsf{K}^t(y) + \mathsf{K}^t(x|y)$$

Then we can estimate $\mathsf{K}^t(x|y)$ by unconditional K^t, which proves NP-hardness of approximating K^t. And (without surprise) this is **false**......

Counter-Example

......at least when one-way function exists. Consider the case $y = f(x)$ where f is OWF, we can approximately think

- $\mathrm{K}^t(y) \approx |y|$, which is most of the case (up to approximation of additive constant);
- $\mathrm{K}^t(x|y) \approx \mathrm{K}^t(x) \approx |x|$, since OWF cannot leak any information to compute x from $y = f(x)$.

However, to compute xy, we can hard-code f and x, then compute y from $f(x)$. This in general costs $|x|$. If $|x| = |y|$, then symmetry of information on this case fails.

Pseudo Symmetry of Information

[Ila23] stated a generalized idea. Assume $\tilde{y}$ is some encoding of y that one can easily decode to y. If there is some encoding scheme such that

$$\mathsf{K}^t(x\tilde{y}) \approx \mathsf{K}^t(x|y) + \mathsf{K}^t(\tilde{y})$$

Then this scheme holds **pseudo symmetry of information**.

Pseudo Symmetry of Information: Construction

Specifically, since we assume decoding is easy, we trivially have $LHS \leq RHS$. To satisfy the other direction, intuitively, we need the encoding has enough **randomness** that diminishes latent relationship between x, y (for example, an OWF), so $\tilde{y}$ cannot be obtained from x easily.

Let $\mathcal{O} : \{0,1\}^{\lambda} \to \{0,1\}$ be a random oracle. For $x, y \in \{0,1\}^n$, we randomly sample $r_i \in \{0,1\}^{\lambda}$ such that $\mathcal{O}(r_i) = y_i$, and let $\tilde{y} \in \{0,1\}^{n\lambda}$ be concatenation of all r_i.

Vulnerability

However in the OWF example, we can hack as follow:

- Hard-code x and compute $y = f(x)$.
- For $(\lambda - 1)$-bit prefix r_i' of r_i, with probability $\frac{1}{2}$ we have $\mathcal{O}(r_i'0) \neq \mathcal{O}(r_i'1)$, then we can determine the unique true value of r_i just from the prefix.

We shall see when hard-coding $\tilde{y}$, with high probability, about $\frac{n}{2}$ bits can be preserved. Therefore

$$\mathsf{K}^t(x\tilde{y}) \approx n + n\left(\lambda - \frac{1}{2}\right)$$

$$\mathsf{K}^t(x|y) + \mathsf{K}^t(\tilde{y}) \approx n + n\lambda$$

Which contradicts definition.

Reflection

There should be some nice property of encoding that ensures enough **incompressibility**, which prevents adversary from cheating by hard-coding a compressed version of $\tilde{y}$.

Intuitively, it suffices to take random oracle with **longer output**, then inferring r_i from a prefix is impossible.

Pseudo Symmetry of Information: Construction Revisited

Consider $\mathcal{O}: \{0,1\}^{2\lambda} \to \{0,1\}^{\lambda}$. Divide y into n/λ blocks, encode them by random preimage of $\mathcal{O}$, obtain the concatenation $\tilde{y} \in \{0,1\}^{2n}$. By the same compression method, since we are limited to t oracle queries, we can only compress $O(\log t)$ bits for each block. Therefore, we (informally) conjectures

$$\mathsf{K}^t(x\tilde{y}) \geq n + \frac{n}{\lambda}(\lambda - O(\log t))$$

When λ overwhelms $\log t$ (for example, $\lambda = \gamma \log t$ for sufficiently large γ), we arrive at $\mathsf{K}^t(x\tilde{y}) \geq (3-\epsilon)n$ and finally we have pseudo symmetry of information approximately achieved.

Formal Results

The author formally proved the following result, for some similar meta-complexity measurement pK (probabilistic Kolmogorov complexity). It says:

Theorem [IIa23]: Let n be a power of two, $x, y \in \{0,1\}^n$, $t = \text{poly}(n)$, $\lambda \geq \Omega(\log n)$. Let $\Delta = \frac{n}{100}$. With probability at least $1 - O(2^{-\Delta/2})$ over encoding by random oracle $\mathcal{O}$ we have

$$\mathrm{pK}^{t''}(x|y) - \Delta \leq \mathrm{pK}^{t',\mathcal{O}}(x\tilde{y}) - 2n \leq \mathrm{pK}^{t}(x|y) + \Delta$$

Where t', t'' are polynomials of t.

Cool, So What?

We can see how this idea is used in the reduction construction. $c_{i,k}$ is a random encoding of encryption on messages, which means there are two steps establishing the soundness of f:

- Encryption step: Use sk to encrypt v to ciphertext c'. Then $\mathrm{K}^t(v|c')$ is large, unless you hard-code sk. This step couples size of Set Cover into the length of hard-coding of sk, which establishes relationship between two problems;
- Encoding step: Encode c' to c by random oracle preimage. Pseudo symmetry of information established, thus

$$\mathrm{K}^t(f) = \mathrm{K}^t(vc) \approx \mathrm{K}^t(v|c') + \mathrm{K}^t(c)$$

So the optimal way of computing f is to hard-code $sk_j, c_{i,k}$ over optimal set cover and decode to v_i.

1 Background

2 Main Result

3 Intuition

4 Proof Sketch

5 Discussion

6 References

Assumptions

We assume the reduction algorithm is equipped with a general random oracle $\mathcal{O} : \{0,1\}^* \to \{0,1\}$, and to compute the λ bits of $\mathcal{O}_n(x)$ for some x, it works as

$$\mathcal{O}_n(x) = \mathcal{O}(1, 1^{\log n}, 0, 1^1, 0, x) \cdots \mathcal{O}(1, 1^{\log n}, 0, 1^\lambda, 0, x)$$

Furthermore, for all “fresh randomness” used by sampling sk, v, c, we assume they are from another oracle $\mathcal{R}_n$ induced by $\mathcal{O}$:

$$\mathcal{R}_n(x) = \mathcal{O}(1, 1^n, 0, 1^{2^\lambda}, 0, x)$$

The two oracles can be evaluated in $O(|x| + \lambda + \log n)$ and $O(|x| + 2^\lambda + n)$.[6]

[6]We will see taking $\lambda = \Omega(\log n)$ suffices later, so the running time is polynomial.

Running Time

The reduction may not halt on rejection sampling, but with satisfyingly small probability.

Theorem 1. With probability at least $1 - 2^{-n2^{\lambda}}$ over choice of $\mathcal{O}$, the reduction runs in time at most poly$(n, 2^{\lambda})$.

Proof. Each sampling of $c_{i,k}$ success with probability $2^{-\lambda}$. The total failure probability on generating $c_{i,k}$ is

$$(1 - 2^{-\lambda})^{T} \leq e^{-T2^{-\lambda}}$$

So taking $T = \text{poly}(n, 2^{\lambda})$ suffices.

Upper Bound on YES Case

To construct a circuit of f, we construct the following:

- A subcircuit that given $(i, j) \in [n] \times [\tau]$ computes $c_{i,k}$;
- A subcircuit that given $q \in [OPT]$ computes sk_{j_q} (the private key of q-th set in optimal cover);
- A subcircuit that given $i \in [n]$ computes (q, k) such that $i \in S_{j,q}$ and $j_q = \text{Index}(i, k)$.

We can compute f by combining those subcircuits and querying $\mathcal{O}$.

Upper Bound on YES Case (Cont'd)

[Ila23] proposed a construction for any $f : \{0,1\}^n \to \{0,1\}^m$ with size $(1+o(1))\frac{m2^n}{n \log m}$ (Omitted)[7], which gives:

Lemma 2. On a YES instance (with $OPT \leq 2n/\tau$), if the reduction outputs f, then there is a constant-depth O-oracle circuit for f of size at most

$$(1+o(1))[\frac{2\lambda n\tau}{\log(n\tau\log(2\lambda))} + \frac{2\lambda n/\tau}{\log 2n/\tau} + 2\log(\tau n)n/\log n + O(\tau + \lambda + \log n)]$$

[7] Based on result of [Lup70].

Lower Bound on NO Case: Main Lemma

[Ila23] proved a result that bounds the probability a deterministic adversary with limited number of $\mathcal{O}$ queries outputs f.

Lemma 3. Assume $\lambda \geq \Omega(\log n)$. Fix any deterministic decision tree P that makes $q \leq 2^{\lambda/O(\tau)}$ queries of length at most $2^{\lambda/O(\tau)}$ to $\mathcal{O}$ and then outputs a string. Fix any NO instance, and let f be the output of reduction with same oracle $\mathcal{O}$, then over choices of oracle,

$$\Pr(P^{\mathcal{O}} = f) \leq 2^{-(1-o(1))(2\lambda n\tau + n\lambda/4)}$$

Main Lemma Proof Sketch

The proof of **Lemma 3** is overwhelmingly long and full of probability technique stuffs.

Intuitively the proof scheme is to **reveal all the randomness of $\mathcal{O}$ (used in P or reduction) by steps**, and at each step we can get some information, as well as bounds on some random variables.

Finally we will be able to bound the probability that P and reduction gives same f.

Main Lemma Proof Sketch: Step 1

Step 1: Reveal values of $\mathcal{O}$ and $\mathcal{O}_n$ that P queries. Assume the inputs on which values of $\mathcal{O}_n$ are revealed by P.

- After this step, output of P is fixed. Assume it gives truth table of f' in same form of f, we can extract “faked” $c'_{i,k}$ and v'_i values.
- We say $i \in [n]$ has w-**collision** if there is v that

$$|\{(i, sk, c) \in Q : \mathcal{O}_n(i, sk, c) = v\}| \geq w$$

  and define C_w as the random variable given by number of $i \in [n]$ with w-collision, then there exists some tail bound on C_w.

Main Lemma Proof Sketch: Step 2

Step 2: Reveal values of $\mathcal{R}_n$ that determine the secret keys in reduction.

- Recall that $\mathcal{R}_n(i) = \mathcal{O}(0, 1^n, 0, 1^{2^\lambda}, 0, i)$, but P only queries $\mathcal{O}$ on input length $\leq 2^{\lambda/O(\tau)}$, so $sk_1, \cdots, sk_m$ **are still uniform** conditioned on information in **Step 1**.
- Secret keys are fixed in **Step 2**, so we can **check the validity of faked ciphertext** $c'_{i,k}$. Let B be the number of pair (i, k) that we do not know decryption of $c'_{i,k}$, or the decryption is inconsistent with v'_i. We can lower bound B with C_w.

Main Lemma Proof Sketch: Step 3

Step 3: Reveal $\mathcal{O}_n$ on inputs corresponding to each $c'_{i,k}$.

- After this step we are able to check for all $c'_{i,k}$ if they decrypt correctly, i.e., if $\mathcal{O}_n(i, sk_{\text{Index}(i,k)}, c'_{i,k} = v'_i$ holds. If all decryption are correct, we say f' is a **valid encoding**.
- We can upper bound the probability that f' is valid encoding by

$$2^{-\lambda n(\tau-1)-OPT\cdot\lambda+m\log(4mq)+\tau n\log(\tau q)+\tau\log n+\log m}$$

Main Lemma Proof Sketch: Step 4 & 5

Step 4: Reveal the rest of $\mathcal{O}_n$.

Step 5: Reveal the remaining values of $\mathcal{R}_n$ that determine v_i.

- Still by the query length argument, values of $\mathcal{R}_n$ is uniformly random conditioned on prior information.
- We are able to bound the probability of $f = f'$, conditioned that f' is valid encoding.

Take $q \leq 2^{\lambda/(128\tau)}$ and recall $m \leq \tau n$ from definition of τ-Frequency Set Cover, as well as $\lambda \geq \Omega(\log n)$, we get on a NO instance:

$$\Pr(f = f') \leq 2^{-(1-o(1))2\lambda n\tau - \lambda n/4}$$

Lower Bound on NO Case

Using **Lemma 3** we are able to give the probabilistic result on lower bound of MCSP over f:

Lemma 4. Assume $\lambda \geq \Omega(\log n)$. Fix any NO instance, with probability at least $1 - 2^{-(1-o(1))n\lambda/8}$ over choice of $\mathcal{O}$, the reduction outputs an f such that

$$\mathsf{MCSP}^{\mathcal{O}}(f) \geq \frac{2\lambda n\tau + n\lambda/8}{\log(2\lambda n\tau + n\lambda/8)}$$

Main Theorem

Combining **Lemma 2** and **Lemma 4**, we can see the complexity gap between reduction result of YES instance and NO instance is

$$
\begin{aligned}
&\frac{2\lambda n\tau + n\lambda/8}{\log(2\lambda n\tau + n\lambda/8)} - (1+o(1))[\frac{2\lambda n\tau}{\log(n\tau\log(2\lambda))} + \frac{2\lambda n/\tau}{\log 2n/\tau} + 2\log(\tau n)n/\log n + O(\tau+\lambda+\log n)] \\
\geq&(1-o(1))\frac{2\lambda n\tau + n\lambda/8}{\log n} - (1+o(1))\frac{2\lambda n\tau + 2\lambda n/\tau}{\log n} \\
\geq&(1-o(1))\frac{n\lambda(1/8 - 2/\tau)}{\log n} \\
\geq&\Omega(\frac{|f|}{\log|f|})
\end{aligned}
$$

Where the probability of existing an Gap τ-Frequency Set Cover instance failing this gap is at most

$$
n^{\tau m}(2^{-(1-o(1))n\lambda/8} + 2^{-n2^{\lambda}}) \leq n^{\tau^2 n}2^{-(1-o(1)n\lambda/8} \leq 2^{-(1-o(1))n\lambda/8+\tau^2 n\log n} \leq 2^{-\Omega(n)}
$$

1 Background

2 Main Result

3 Intuition

4 Proof Sketch

5 Discussion

6 References

Contribution

With non-uniform reduction (replacing the uniform one with oracle access), we can informally write the result as

$$\mathsf{NP} \subset \mathsf{P}^{\mathsf{MCSP}^{\mathcal{O}}}/\mathsf{poly}$$

This is a very strong evidence that MCSP might be NP-Complete: Might be able to instantiate $\mathcal{O}$ with real-world hash functions and prove a similar result.[8]

This proof bypasses barrier results like implication of $\mathsf{EXP} \neq \mathsf{ZPP}$, [9] as well as limitations of oracle-independent reductions.

[8] Such instantiation is not always possible [CGH04], but is successful most of the time in real world.

[9] Because in fact $\mathsf{ZPP}^{\mathcal{O}} = \mathsf{P}^{\mathcal{O}}$ with random oracle.

Properties of Random Oracle

The random oracle plays an important role both intuitively and technically. Some useful properties:

- **No partial information on preimage.** One cannot distinguish the distribution of suffix of r, conditioned on knowing $y = \mathcal{O}(r)$ and a prefix of r, or only knowing $y = \mathcal{O}(r)$. This mainly supports incompressibility of random oracle encoding in pseudo symmetry of information.
- **Pairwise independence.** This supports the query length argument in formal proof.
- **Collision-resistance.** So you cannot cheat by potentially producing some collisions to bypass the secret key requirement.

If we can utilize some real-world hash function with these properties (even just approximately), maybe instantiation of $\mathcal{O}$ is possible.

1. Background
2. Main Result
3. Intuition
4. Proof Sketch
5. Discussion
6. **References**

References

[CGH04] Ran Canetti, Oded Goldreich, and Shai Halevi.
The random oracle methodology, revisited, 2004.

[DGKR03] Irit Dinur, Venkatesan Guruswami, Subhash Khot, and Oded Regev.
A new multilayered pcp and the hardness of hypergraph vertex cover, 2003.

[HIR23] Yizhi Huang, Rahul Ilango, and Hanlin Ren.
Np-hardness of approximating meta-complexity: A cryptographic approach, 2023.

[Ila23] Rahul Ilango.
Sat reduces to the minimum circuit size problem with a random oracle, 2023.

[Lup70] O. B. Lupanov.
On a method of circuit synthesis, 1970.