Kolmogorov Complexity and its Applications to OWF

Term Paper: Karthik Gajulapalli

June 4, 2022

Abstract

We present a rough sketch of a recent result by Pass and Liu that provides a characterization of OWF via the average case hardness of the time-bounded kolmogorov complexity problem. We also show two proofs of Brassards Theorem which presents a barrier to base the existence of OWP on $P \neq NP$.

1 Preliminaries

1.1 P and NP

Definition 1. P: Set of languages decidable by a Machine running in polynomial time.

Definition 2. NP: Set of langauges where given a short witness proof of membership can be decided in polynomial time.

Note that in the above definition finding a short witness may turn out to be very hard. The P vs NP question essentially formalizes this question to ask if finding such witness can be done quickly (polynomial time).

Definition 3. coNP: The negation of the set of NP languages. Or rather coNP is the set of languages that have short witness for non-membership.

We know that $P \subseteq NP \cap coNP$ But there are some problems in $NP \cap coNP$ that we don't believe to be in NP. For example Factoring is in $NP \cap coNP$. A proof of membership is just the prime factors, and a proof of non-membership would be a primality test. Other interesting problems lying here are Discrete Log and Graph Isomorphism.

Definition 4. NP-Hard: A language, L, is said to be NP-hard if given any language in NP there exists a polynomial transformation to an instance of L.

Thus being able to solve a NP-hard language in polynomial time would imply the existence of an algorithm for any NP language thereby resolving the open question with the answer P = NP.

Lemma 5. *If a language L which is NP-Hard is also in $coNP$, then $NP = coNP$*

Proof. if $L' \in NP$, then there exists reduction R such that $R(L') = L$ but since L is in coNP, L' is also in coNP.
If L' $\in$ coNP then $\overline{L'} \in NP$. So there is a reduction R, that takes $R(\overline{L'})$ to L but since L is in coNP, $\overline{L'}$ is in coNP, so L' must be in NP. □

1.2 MCSP and Kolmogorov Complexity

Definition 6. MCSP: (Minimum Circuit Size Problem) is given a truth table for some boolean function f, and a size parameter s, is there a circuit of size s that computes f.

MCSP has a very special place in complexity theory. Leonid Levin actually delayed publishing his theory of NP-hardness because he wanted to prove that MCSP is NP-hard. While its obvious that MCSP is in NP, we don't know if it is NP-intermediate or even NP-hard. We don't believe that MCSP would be in P, since a no instance to an MCSP query implies a lowerbound, something complexity researchers have been struggling with for years.

Kolmogorov Complexity was introduced to formalize the intuition of what it means for one string to be more random then another. For example why do we think that 000000000111111 is more random looking then 010111010110111. Kolmogorov's claim was that stings that appear less random are highly compressible while stings that appear more random don't have good compressions.

Definition 7. Kolmogorov Complexity: The Kolmogorov Complexity of a string s, can be thought of as the compressibility of a string, i.e. what is the smallest $< M, x >$ program-input pair that when simulated by the Universal Turing Machine outputs s.

The problem with the above definition is finding the kolmogorov complexity of a string is that there is no bound on how long M can run and that makes it undecidable.

We define 3 decidable measures of Komogorov below:

Definition 8. $\mathbf{K^t(x)}$ (t-time bounded Kolmogorov Complexity):

$$K^t(x) = \min_{\pi \in \{0,1\}^*} \{|\pi| : U(\pi, 1^{t(|x|)}) = x\}$$

While the change above makes the K^t decidable, the next definions parametrize the time t to also include it in the measure

Definition 9. Kt(x) or **MKtP**:

$$Kt(x) = \min_{\pi \in \{0,1\}^*, t \in \mathbb{N}} \{|\pi| + \lceil \log t \rceil : U(\pi, 1^t) = x\}$$

Definition 10. KT(x) or **MKTP**:

$$KT(x) = \min_{\pi \in \{0,1\}^*, t \in \mathbb{N}} \{|\pi| + \lceil t \rceil : U(\pi, 1^t) = x\}$$

Fact 11. *$K^t(x), Kt(x), KT(x) \leq |x| + c$.*
The upperbound is just the machine that hardwires the string as its output (description)

To connect MCSP, and Kolmogorov Complexity, consider a boolean string, s, s.t $|s| = 2^l$ for some l. We could think of this string as encoding the truth table of some boolean function with l inputs. From [Lup70] we know that random functions have high circuit complexity, and correspondingly random strings have high Kolmogorov Complexity. More formal connections can be found in [All21].

1.3 One Way Functions, PRG's

Definition 12. OWF: Let $f : \{0,1\}^* \rightarrow \{0,1\}^*$ be a polynomial-time computable function.f is said to be a one-way function of for every PPT algorithm $\mathcal{A}$, there exists a negligible function μ sucht hat for all $n \in \mathbb{N}$:

$$Pr[x \leftarrow \{0,1\}^n; y = f(x) : \mathcal{A}(1^n, y) \in f^{-1}(f(x))] \leq \mu(n)$$

Definition 13. Weak OWF: Let $f : \{0,1\}^* \rightarrow \{0,1\}^*$ be a polynomial-time computable function.f is said to be a α-weak OWF if for every PPT algorithm $\mathcal{A}$, for all $n \in \mathbb{N}$:

$$Pr[x \leftarrow \{0,1\}^n; y = f(x) : \mathcal{A}(1^n, y) \in f^{-1}(f(x))] < 1 - \alpha(n)$$

Theorem 14. ***Yao-Amplification*** *[Yao82]: If weak one-way functions exist then there exists a one-way function.*

Definition 15. Computational Indistinguishability: Two ensembles $\{A_n\}_{n\in\mathbb{N}}$ and $\{A_n\}_{n\in\mathbb{N}}$ are said to be $\epsilon(.)$-indistighuishable, if for every PPT machine D (a distinguisher) whose running time is polynomial, there exists some $n_0 \in \mathbb{N}$ so that $\forall n \geq n_0$:

$$|Pr[D(1^n, A_n) = 1] - Pr[D(1^n, B_n) = 1]| < \epsilon(n)$$

Definition 16. PRG: Let $g : \{0,1\}^n \rightarrow \{0,1\}'(n)$ be a polynomial-time computable function. g is said to be an $\epsilon(.)$-PRG if for any PPT algorithm $\mathcal{A}$ (whose running time is polynomial), for all sufficiently large n,:

$$|Pr[x \leftarrow \{0,1\}^n : \mathcal{A}(1^n, g(x)) = 1] - Pr[y \leftarrow \{0,1\}^{m(n)} : \mathcal{A}(1^n, y) = 1]| < \epsilon(n)$$

Theorem 17. ***HILL*** *[HILL99]: Existence of OWF implies existence of PRG*

Definition 18. Conditionally Secure entropy-preserving pseudorandom generator:

- **(pseudorandomness)**: $\{\mathcal{G}(\mathcal{U}_n|E_n)\}_{n\in\mathbb{N}}$ and $\{\mathcal{U}_{n+\gamma\log n}\}$ are $\mu()$-indistinguishable
- **(entropy-preserving):** For all large $n \in \mathbb{N}, H(G((U)_n|E_n)) \geq n - \alpha\log n$

1.4 Average Case Complexity

Definition 19. Heur$_{\mathbf{neg}}$BPP : $L \in Heur_{neg}BPP$ if forall polynomial $p(.)$, there exists a probabilistic polynomial-time heuristic $\mathcal{H}$, such that for all n ($\frac{1}{p(.)} - HoA$):

$$Pr[x \leftarrow \{0,1\}^n : \mathcal{H}(x) = L(x)] \geq 1 - \frac{1}{p(n)}$$

Definition 20. Avg$_{\mathbf{neg}}$BPP : $L \in Avg_{neg}BPP$if forall polynomial $p(.)$, there exists a probabilistic polynomial-time heuristic $\mathcal{H}$, such that for all n:

$$Pr[\mathcal{H}(x) \in \{L(x), \perp\}] \geq 0.9$$

and

$$Pr[x \in \{0,1\}^n : \mathcal{H}(x) =\perp] \leq \frac{1}{p(n)}$$

Essentially in the first definition our Heursitic algorithm is allowed to make 2-sided error, while our second definition considers error-less Heuristics. The Heuristic either outputs the correct answer or $\perp$("I don't know").

2 Brassard Worst-Case Barrier

Brassard [Bra83] showed a very simple proof that said that if you tried to construct OWP based on hardness from assumption of P != NP, you will show that $NP = coNP$ which would collapse the polynomial hierarchy. We provide two versions of proof of the theorem.

Theorem 21. *(Complexity Version): Given an injective function $f : \{0,1\}^n \to \{0,1\}^n$ where $f \in P$ and $P \neq NP$. If f^{-1} is $NP - Hard$ then $NP = coNP$*

Proof. Consider the language $Q = \{< y, x > | f^{-1}(y) \geq x\}$. If f^{-1} is NP-hard then so is Q. Since you can solve f^{-1} with oracle queries to Q (just do binary search). Note that Q is also in NP since the pre-image will act as a witness.
Note that $\overline{Q} = \{< y, x > | f^{-1}(y) < x\}$ is also in NP since the same binary search idea works here too. This means that $Q \in coNP$ and is NP-hard. Which by lemma 5 implies $NP = coNP$

□

Theorem 22. *(Cryptography Version): If you construct OWP from hardness of a language L then for any reduction R, and PPT A that inverts f with non-neg() probability, then R^A decides $L \implies L \in$ coNP*

Proof. Consider the set of queries $q_1, ..., q_{t(n)}$ made by the reduction R to the f^{-1} oracle in deciding if $x \in L$.
We can rewrite this as $x \in L$ iff $R(x, q_1, ...q_{t(n)}) = 1$. So the answers to the queries to f^{-1} act as a witness. Note that since this is a permutation you are always guaranteed to have a unique pre-image for every image.
We now need to show that deciding $\overline{L}$ is also in NP. So we essentially just need to send the x_i s.t $f(x_i) = q_i$.
$\overline{R}$ given access to the x_i can create an instance of R and then flip the bit. Thus $L \in coNP$.

□

Corollary 23. *Any deterministic PKE that is NP-hard to invert would imply NP = coNP*

3 $\mathrm{K^t}$ HoA iff OWF exist [LP21]

Theorem 24. *The following are equivalent:*

(a) The existence of One Way Functions

(b) The existence of a polynomial $t(n) > 0$, such that K^t is mildly hard-on-average

(c) $\forall d, \epsilon > 0$, it is mildly hard-on-average to approximate K^t better than $d - \log n$.

Proof. (sketch)

- $b \implies a$
  Let $y = f(x)$, then we want to make inverting y becoming the equivalent of knowing the Kolmogorov complexity of y.
  Here is the construction: input of f is $l||\pi'$, output is running $l||U(\pi)$ where π is the first l

bits of π'.
If we could invert f, then we just run our algorithm on all $i||y$. The smallest i will be the kolmogorov complexity of the string with h.p.

- OWF $\implies$ EP-PRG
  idea here is to first show that any regular OWF is actually EP-PRG.
  Then convert a OWF into a regular function (S-OWF) under some conditional event.

- if EP-PRG G, then K^t mildy HoA to $d \log n$ approximate The key point here is that $K^t(G(x))$ will be really small (since G, x defines a small compression), while K^t($$) will be very high. So K^t oracle will break pseudorandomness property a contradiction.

□

4 Future Directions

Pass and Liu extend their framework to almost get OWF from the very weak assumption of EXP != BPP. The idea is to change the measure from time bounded Kolmogorov complexity to MKtP problem which is known to be EXP-complete. Unfortunately to make their construction work they would need to show that MKtP hard for errorless heuristics would imply that MKtP is hard for heuristics with two sided errors. But doing this would imply that $P \neq NP$. A question to go around this barrier would be to maybe prove the same result with a stronger assumption like PSPACE $\neq$ BPP?

Another very interesting question would be if we can get OWF from the average case hardness of MCSP. This would be really interesting since it would provide a bridge between a highly studied problem in complexity and its connections to cryptography. Rahul Santhanam [San20] proves something along this lines but makes use of an universality conjecture about locally samplable distributions. Since no barrier is known about proving such a result with no assumptions, it seems to be very interesting if we can get a characterization of OWF with no assumptions from the average case hardness of MCSP.

References

[All21] Eric Allender. Vaughan jones, kolmogorov complexity, and the new complexity landscape around circuit minimization. *New Zealand journal of mathematics*, 52:585–604, 2021.

[Bra83] Gilles Brassard. Relativized cryptography. *IEEE Transactions on Information Theory*, 29(6):877–894, 1983.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[LP21] Yanyi Liu and Rafael Pass. On one-way functions from np-complete problems. *Cryptology ePrint Archive*, 2021.

[Lup70] Oleg B Lupanov. On a method of circuit synthesis. *Journal of Symbolic Logic*, 35(4), 1970.

[San20] Rahul Santhanam. Pseudorandomness and the minimum circuit size problem. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[Yao82] Andrew C Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*, pages 80–91. IEEE, 1982.