COMS 6998: Uncond. Lower Bds. and Derand. **Spring 2024**

# Introducing Braverman's Theorem

*Lecturer: Rocco Servedio* *Scribe: Shiv Kampani*

## 1 Bounded Independence Fools $\mathsf{AC}^0$

In this section, we will discuss Braverman's Theorem which states that any $k$-wise independent RV fools $\mathsf{AC}^0$. Before formally stating and proving this theorem, we will motivate PRGs for $\mathsf{AC}^0$.

**Motivation.** In short, $\mathsf{AC}^0$ is fascinating. Previously, we have proved both worst-case and average-case lower bounds for $\mathsf{AC}^0$, constructing a pseudorandom generator for the class is a natural extension. Here is a **brief history** of this problem:

| | | |
|---|---|---|
| 1990 | Linial & Nisan made conjecture that polynomial-size constant depth circuits ($\mathsf{AC}^0$) is fooled by any $k$-wise independent distribution for $k \sim \text{polylog}(n)$. | [LN90] |
| 2007 | Bazzi proved the conjecture for $d = 2$ (50 page paper). | [Baz07] |
| 2008 | Razborov presented a simplified proof for $d = 2$ (4 page paper). | [Raz08] |
| 2009 | Braverman proved the conjecture for all $d$ | [Bra09] |

It is worth mentioning that this PRG is neither the simplest, nor the first, nor with the best parameters. The current state-of-the-art for an $\epsilon$-PRG for $\mathsf{AC}^0_{s,d}$ has seed-length $k = (\log s)^{d+O(1)} \cdot \log \frac{1}{\epsilon}$ [ST19] (to achieve this result, we need to make use of the **multi-switching lemma**). Nevertheless, $k$-wise independence is "general" machinery (we do not need to make contrived constructions for random variables).

**Intuition** (for conjecture). Why is $k$-wise independence "good enough" to fool $\mathsf{AC}^0$? There are several reasons for this:

1. $k$-wise independence fools $\mathcal{DT}_k$, $k$-juntas ($\mathcal{J}_k$), etc.

2. $k$-wise independence fools real polynomials of degree-$k$ (can be shown by definition and application of $\triangle$-inequality).

3. Since the 90's it has been known that circuits in $\mathsf{AC}^0$ are well-approximated by low-degree real polynomials. There are two such approximators of interest:

   (a) Point-wise approximator (Beigel, Reingold, Spielman) [BRS91]

   (b) $L_2$ approximator (Linial, Mansour, Nisan) [LMN89]

Neither of the approximators above, however, is a sandwiching approximator. The idea behind Braverman's proof was to combine these approximators in a clever way to construct a sandwiching polynomial approximator.

**Theorem 1** (Braverman's Theorem)**.** *Let $k = \left(\log \frac{s}{\epsilon}\right)^{O(d)}$. Let $\mathbf{D}$ be any $k$-wise independent RV over $\{0,1\}^n$. Then $\mathbf{D}$ $\epsilon$-fools $\mathsf{AC}^0_{s,d}$.*

**Remark.** Braverman's original proof was worse; it had a $O(d^2)$ exponent for $k$.
Before proving Braverman's theorem, we will prove both polynomial approximators.

## 1.1 Point-wise approximator

**Lemma 2** ("point-wise" BRS approximator)**.** *Let $f \in \mathsf{AC}^0_{s,d}$. Let $\mathcal{D}$ be any distribution over $\{0,1\}^n$. $\exists$ a real polynomial $p$ : $\Pr_{\boldsymbol{x}\sim\mathcal{D}}[p(\boldsymbol{x}) = f(\boldsymbol{x})] \geq 1-\epsilon$, where:*

*(i) $deg(p) \leq \left(\log \frac{s}{\epsilon}\right)^{O(d)}$*
*(ii) $\forall x \in \{0,1\}^n \; |p(x)| \leq \exp\left(\log \frac{s}{\epsilon}\right)^{O(d)}$*

*Proof.* (Lemma 2) We will begin by proving the existence of polynomial "point-wise" approximators for OR gates. An OR gate with fan-in $t$ computes $f(x) = x_1 \vee \cdots \vee x_t$. We will perform a probabilistic construction by describing a distribution over polynomials, making random draws from the distribution, and showing that they "do well" with high probability.

Let $v_0 := \{x_1, \ldots, x_t\}$. Let $p_0(x) := x_1 + \cdots + x_t$. For $i \in 1, \ldots, \log_2(t)+1$, let $\boldsymbol{v}_i$ be constructed from $\boldsymbol{v}_{i-1}$ by independently and randomly discarding each variable in $\boldsymbol{v}_{i-1}$ with probability 1/2. Let $\boldsymbol{p}_i(x) := \sum\limits_{x_j \in \boldsymbol{v}_i} x_j$.

**Fact 3.** $p_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_{\log_2(t)+1}$ *are all deg-*1 *polynomials mapping* $\{0,1\}^t \to [0,t]$

The idea behind these polynomials is a process of **random sifting** to isolate a single 1 from the variables $x_1, \ldots, x_t$. $p_0(x)$ "fails" at approximation when the input $x$ contains multiple 1's, since it leads to a sum $> 1$. But, among the "sifted" polynomials, there is a good chance that some polynomial is exactly 1. Note that in the special case that the input is $0^t$, all polynomials are 0 (correct approximators).

Fix any input assignment $z \in \{0,1\}^t : z \neq 0^t \implies z_1 \vee \cdots \vee z_t = 1$.

**Claim 4.** $\Pr[$*at least one of* $p_0(z), \boldsymbol{p}_1(z), \ldots, \boldsymbol{p}_{\log_2(t)+1}(z) = 1] \geq 1/3$

*Proof.* We will consider three mutually-exclusive cases:

(a) **Case:** $\forall i = 1, \ldots, \log_2(t) + 1 \;\; \boldsymbol{p}_i(z) > 1$.

Each fixed variable has a $1/2t$ chance of surviving all stages of sifting. Thus, the probability that any variable $z_j$ survives all the stages of sifting $\leq 1/2$ (union bound). Thus, as a coarse upper bound, $\Pr[(a)] \leq 1/2$.

(b) **Case:** $p_0(z) = 1$.

(c) **Case:** for some stage $i \in [1, \log_2(t)]$, $\boldsymbol{p}_i(z) > 1$ and $\boldsymbol{p}_{i+1}(z) \leq 1$.

For any $j$, given the value of $\boldsymbol{p}_j(z)$, we have:

- $\Pr[\boldsymbol{p}_{j+1}(z) = 0] = \left(\frac{1}{2}\right)^{\boldsymbol{p}_j(z)}$ (all the 1-variables in $z$ need to be discarded in iteration $j+1$ for $\boldsymbol{p}_{j+1}(z) = 0$).
- $\Pr[\boldsymbol{p}_{j+1}(z) = 1] = \boldsymbol{p}_j(z) \cdot \left(\frac{1}{2}\right)^{\boldsymbol{p}_j(z)}$ (all but **exactly one** of the 1-variables in $z$ need to be discarded in iteration $j+1$ for $\boldsymbol{p}_{j+1}(z) = 0$).

For stage $i$, in the worst-case, $\boldsymbol{p}_i(z) = 2$. In this specific case, we are given that $\boldsymbol{p}_i(z) > 1$ and $\boldsymbol{p}_{i+1}(z) \leq 1$. Thus, $\Pr[\boldsymbol{p}_{i+1}(z) = 1 | \boldsymbol{p}_i(z) > 1 \wedge \boldsymbol{p}_{i+1}(z) \leq 1] = \frac{\boldsymbol{p}_j(z)}{1+\boldsymbol{p}_j(z)} \geq 2/3$ (worst-case probability).

Consider the cases in which our claim event (at least one of the polynomials equals 1) is true. As we have shown, in case (a) our claim event is always false, in case (b) our claim event is always true, and in case (c) our claim event is true with probability $\geq 2/3$. Since we showed $\Pr[(a)] \leq 1/2$, either (b) or (c) occurs with probability $\geq 1/2$. Thus, the probability that our claim event is true $= \frac{1}{2}(\Pr[(b)|\text{not }(a)] + \frac{2}{3}\Pr[(c)|\text{not }(a)]) \geq \frac{1}{2} \cdot \frac{2}{3} = \frac{1}{3}$, which proves our claim. ■

To prove Lemma 2, we need a **single** polynomial approximator with amplified success probability. Define $\boldsymbol{r}(x) \coloneqq \prod_{i=0}^{\log_2 t+1}(1-\boldsymbol{p}_i(x))$. $\boldsymbol{r}(x)$ is a polynomial of degree $\leq \log_2(t)+1$ that maps $\{0,1\}^t \to [-t^{O(\log t)}, t^{O(\log t)}]$. When $x = 0^t, \boldsymbol{r}(x) = 1$ since $\forall i, \boldsymbol{p}_i(x) = 0$. For any $x \neq 0^t, \boldsymbol{r}(x) = 0$ with probability $\geq 1/3$ (from Claim 4).

To improve success probability, let $\boldsymbol{r}'(x) \coloneqq$ product of $O(\log \frac{1}{\epsilon})$ independent $\boldsymbol{r}(x)$'s. By definition, $\boldsymbol{r}'(x)$ is a polynomial of degree $\leq O(\log \frac{1}{\epsilon} \cdot \log t)$ that maps $\{0,1\}^t \to [-t^{O(\log \frac{1}{\epsilon} \cdot \log t)}, t^{O(\log \frac{1}{\epsilon} \cdot \log t)}]$. Again, we have $\boldsymbol{r}'(0^t) = 1$. We also have:

$$\text{Let } x \neq 0^t,\ \Pr[\boldsymbol{r}'(x) \neq 0] \leq \left(\frac{2}{3}\right)^{O(\log \frac{1}{\epsilon})} \leq \epsilon \quad \dots \quad \text{(independence)}$$

Finally, define $\boldsymbol{a}(x) \coloneqq 1 - \boldsymbol{r}'(x)$. Based on the properties of $\boldsymbol{r}'(x)$, we have:

$$\forall x \ \Pr_{\boldsymbol{a}}[\boldsymbol{a}(x) = x_1 \vee \dots \vee x_t] \geq 1-\epsilon \quad \dots \quad (*)$$

Consider Table 1. Each column corresponds to an $x \in \{0,1\}^t$. Each row corresponds to a polynomial outcome of $a$ with respect to the randomness used to generate $\boldsymbol{a}$. A check-mark (✓) in cell $(i,j)$ indicates that $a_i(x_j) = x_{j_1} \vee x_{j_2} \vee \dots \vee x_{j_t}$. This is an illustration of the **minimax theorem**.

| | $x_0 = 0^t$ | $x_1 = 0^{t-1}1$ | $\dots$ |
|---|---|---|---|
| $a_1$ | ✓ | $\dots$ | $\ddots$ |
| $a_2$ | ✓ | $\dots$ | $\ddots$ |
| $\vdots$ | $\vdots$ | $\ddots$ | $\ddots$ |

Table 1: Check-mark (✓) Matrix (Minimax Theorem)

Clearly, there are $2^t$ columns in the check-mark matrix. By statement $(*)$, we have that, in every column (corresponding to some $x_j$), the probability of any cell having ✓is $\geq 1-\epsilon \implies$ ✓-density in every column $\geq 1-\epsilon$. For any distribution $\mathcal{D}$ over columns, the matrix ✓-density according to this distribution $\mathcal{D}$ is $\geq 1-\epsilon$. Finally, this implies that there must exist a row in the matrix (corresponding to some polynomial outcome $a_i$) such that the ✓-density in that row is $\geq 1-\epsilon$. Formally:

$$\forall \text{ distributions } \mathcal{D}, \exists \text{ polynomial outcome of } a : \Pr[a(\boldsymbol{x}) = \boldsymbol{x}_1 \vee \cdots \vee \boldsymbol{x}_t] \geq 1-\epsilon \quad (**)$$

The degree of $a$ is equal to the degree of $\boldsymbol{r}' \leq O(\log \frac{1}{\epsilon} \cdot \log t)$.

So far, we have shown the existence of polynomial "point-wise" approximators for OR gates. In general, Boolean circuits (in the De Morgan basis) may make use of OR, AND, NOT gates. Thus, we need to show the existence of approximators for AND, NOT gates with guarantee $(**)$:

($\neg$) NOT gates take only a single bit as input and flip its value. Thus, $a(x) = 1-x$ is a degree-1 polynomial that exactly computes $\neg x$.

($\wedge$) We can show that there exist polynomial approximators for AND gates with the same degree and guarantee $(**)$ as the OR gate approximator using the NOT gate polynomial.

**Proof sketch:** Observe that AND, OR are dual-functions: $\text{AND}(x_1, \ldots, x_t) = \text{NOT}(\text{OR}(\text{NOT}(x_1), \ldots, \text{NOT}(x_t)))$. For any distribution $\mathcal{D}$, take the polynomial approximator for OR, i.e., $a(x)$. Replace every literal $x_i$ in $a$ with $(1-x_i)$. Let the updated polynomial be $a'(x)$. Define $a''(x) := 1 - a'(x)$. Clearly, from the check-mark matrix, $a''(x)$ is a polynomial approximator for AND on distribution $\mathcal{D}$ with same degree. ■

Any circuit $C$ computing $f \in \mathsf{AC}^0_{s,d}$ will have $\leq s$ gates by definition. Consider any gate $g$ (either OR, AND, NOT) in circuit $C$. $g$ takes in at most $s$ inputs. Let $\mathcal{D}'$ be the distribution of inputs into gate $g$ when the circuit input $\boldsymbol{x} \sim \mathcal{D}$. By our proofs for OR, AND, NOT gates, for any $\mathcal{D}'$, there exists a polynomial $a$ that "point-wise" approximates gate $g$ with error $1-\epsilon/s$. $a$ has degree at-most $O(\log \frac{s}{\epsilon} \cdot \log s)$.

We can replace every gate in $C$ with the corresponding polynomial approximator and call the result $p(x)$. By union bound $p(x)$ correctly computes every gate in $C$, and therefore correctly computes $f$, with probability $\geq 1 - \frac{s}{\epsilon} \cdot s = 1-\epsilon$.

$p(x)$ has degree $O(\log \frac{s}{\epsilon} \cdot \log s)^d \leq \left(\log \frac{s}{\epsilon}\right)^{O(d)}$ (satisfies (i)).

In our proof, we showed that $|\boldsymbol{r}'(x)| \leq t^{O(\log \frac{1}{\epsilon} \cdot \log t)} \implies |a(x)| \leq t^{O(\log \frac{1}{\epsilon} \cdot \log t)}$. For any gate $g$, let $a_g$ be the approximator. $|a_g(x)| \leq s^{O(\log \frac{s}{\epsilon} \cdot \log s)} = \exp\big(O(\log \frac{s}{\epsilon} \cdot \log^2 s)\big)$. Thus, for the polynomial $p(x)$, $|p(x)| \leq |a_g(x)|^d \leq \exp\big(\log \frac{s}{\epsilon}\big)^{O(d)}$ (satisfies (ii)).

We call $p(x)$ the BRS approximator. This concludes our proof of Lemma 2. ■

## 1.2 $L_2$ approximator

**Theorem 5** ("$L_2$" LMN approximator)**.** *Let $f \in \mathsf{AC}^0_{s,d}$. $\exists$ a real polynomial $p_2$ of degree $O(\big(\log \frac{s}{\epsilon}\big)^d)$ :* $\mathop{\mathbb{E}}_{\mathbf{U}\sim\mathcal{U}}\Big[(f(\mathbf{U}) - p_2(\mathbf{U}))^2\Big] \leq \epsilon$.

**Remark.** Like the BRS approximator, the LMN approximator does not necessarily sandwich $f(x)$ since $|f(x) - p_2(x)|$ may (rarely) be large.

**Notation.** Let $W^k(f) \coloneqq \sum_{S\subseteq[n],|S|=k} \widehat{f}(S)^2$ and $W^{\geq k}(f) \coloneqq \sum_{S\subseteq[n],|S|\geq k} \widehat{f}(S)^2$.

**Fact 6.** *Let $f : \{0,1\}^n \to \{\pm 1\}$. $W^{\geq 0}(f) = \sum_{S\subseteq[n]} \widehat{f}(S)^2 = 1$ (Parseval's Identity).*

*Proof.* (Theorem 5) The idea behind our proof is to show that for $f \in \mathsf{AC}^0_{s,d}$, the Fourier "tail weight" is very small. Consider the following theorem:

**Theorem 7.** *If $f \in AC^0_{s,d}$, then: $\forall r \quad W^{>r}(f) \leq 2s \cdot 2^{\frac{-r^{1/d}}{20}}$*

Intuitively, Theorem 7 states that a property of all functions $f \in \mathsf{AC}^0_{s,d}$ is that as one goes "higher up" in the Boolean hypercube, the Fourier tail weight drops exponentially.

We will use Theorem 7 to prove Theorem 5. Take $r = \big(20 \log \frac{2s}{\epsilon}\big)^d$. For this value of $r$, $2s \cdot 2^{\frac{-r^{1/d}}{20}} \leq \epsilon$. We define the polynomial approximator $p_2$ as the truncated (at level $r$) part of the Fourier representation of $f$. Explicitly:

$$p_2(x) \coloneqq \sum_{|S|\leq r} \widehat{f}(S)\, \chi_S(x) = \sum_{|S|\leq r} \widehat{f}(S) \prod_{i\in S} x_i$$

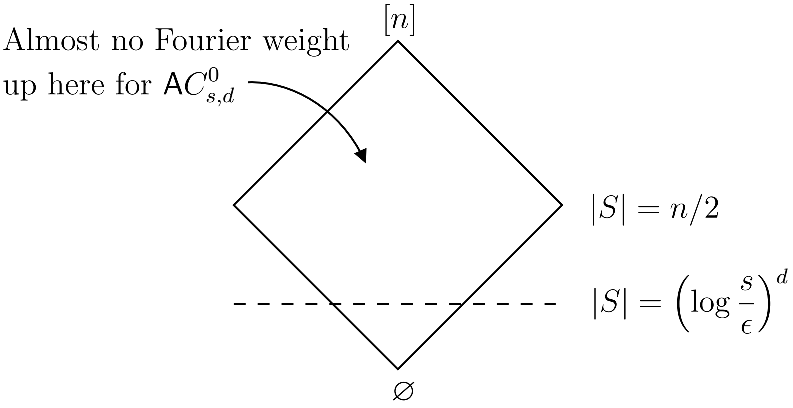
Almost no Fourier weight up here for $\mathsf{AC}^0_{s,d}$
$[n]$
$|S| = n/2$
$|S| = \left(\log \frac{s}{\epsilon}\right)^d$
$\varnothing$

Figure 1: Theorem 7 on Boolean Hypercube

Clearly, the degree of $p \le r = O(\left(\log \frac{s}{\epsilon}\right)^d)$ as required. All that remains is to prove that $p_2$ is a valid $L_2$ polynomial approximator for $f$. By Parseval's Identity:

$$\begin{aligned}\mathop{\mathbb{E}}_{\mathbf{U}\sim\mathcal{U}}\left[(f(\mathbf{U}) - p_2(\mathbf{U}))^2\right] &= \sum_{S\subseteq[n]} \left(\widehat{f - p_2}(S)\right)^2 \\ &= \sum_{S\subseteq[n]} \left(\widehat{f}(S) - \widehat{p_2}(S)\right)^2 \\ &= \sum_{|S|>r} \left(\widehat{f}(S)\right)^2 = W^{>r}(f) \le \epsilon \quad \text{(Theorem 7)}\end{aligned}$$

This completes our proof of Theorem 5. All that remains is to prove Theorem 7. ■

*Proof.* (Theorem 7) To prove this Fourier concentration theorem, we will make use of two helpful lemmas:

**Lemma 8** (Håstad's Switching Lemma)**.** *Let $f \in \mathsf{AC}^0_{s,d}$. Fix $t$ and let $\boldsymbol{\rho} \sim R_p$ (random restriction with $*$ probability $p$) with $p \le \frac{1}{10^d \cdot t^{d-1}}$. Then:*

$$\Pr_{\boldsymbol{\rho}\sim R_p}\left[\text{DT-depth}(f \upharpoonright \boldsymbol{\rho}) \ge t\right] \le s \cdot 2^{-d}$$

We have already proved Lemma 8 in a previous lecture.

**Lemma 9.** *For any $f : \{\pm 1\}^n \to \{\pm 1\}$, any $p \le 1/10$, we have:*

$$W^{\ge t/p}(f) \le 2 \cdot \mathop{\mathbb{E}}_{(\boldsymbol{J},\boldsymbol{z})\sim R_p}[W^{\ge t}(f_{\boldsymbol{J},\boldsymbol{z}})]$$

**Notation.** $(\boldsymbol{J}, \boldsymbol{z}) \sim R_p$ defines a random restriction and is equivalent to $\boldsymbol{\rho}$. $\boldsymbol{J} \subseteq [n]$ is the subset of variables that survived the restriction ($*$'s) and $\boldsymbol{z} \in \{\pm 1\}^{[n]\setminus \boldsymbol{J}}$ is the vector of assignments to the remaining non-$*$ variables.

We will prove Lemma 9 in the next lecture. We will now show that, given both lemmas above, we can prove Theorem 7.

Let $p = \frac{1}{10r^{\frac{d-1}{d}}} \implies pr = \frac{r^{1/d}}{10}$. Let $t = \frac{pr}{2} = \frac{r^{1/d}}{20}$. Thus: $\frac{1}{10^d \cdot t^{d-1}} = \frac{2^{d-1}}{10r^{\frac{d-1}{d}}} = 2^{d-1}p \geq p$.

From Lemma 8, $\Pr_{\boldsymbol{\rho}\sim R_p}[\text{DT-depth}(f \restriction \boldsymbol{\rho}) \geq t] \leq s \cdot 2^{-t}$. Now, we will compute the expected Fourier weight above $t = \frac{r^{1/d}}{20}$ of $f \restriction \boldsymbol{\rho} = f_{\boldsymbol{J},\boldsymbol{z}}$ (restricted function):

$$\begin{aligned}
\mathop{\mathbb{E}}_{(\boldsymbol{J},\boldsymbol{z})\sim R_p}[W^{\geq t}(f_{\boldsymbol{J},\boldsymbol{z}})] &= W^{\geq t}(f_{\boldsymbol{J},\boldsymbol{z}}) \cdot \Pr_{\boldsymbol{\rho}\sim R_p}[\text{DT-depth}(f \restriction \boldsymbol{\rho}) \geq t] \\
&\quad + 0 \cdot \Pr_{\boldsymbol{\rho}\sim R_p}[\text{DT-depth}(f \restriction \boldsymbol{\rho}) < t] \\
&\leq W^{\geq 0}(f_{\boldsymbol{J},\boldsymbol{z}}) \cdot \Pr_{\boldsymbol{\rho}\sim R_p}[\text{DT-depth}(f \restriction \boldsymbol{\rho}) \geq t] \\
&= \Pr_{\boldsymbol{\rho}\sim R_p}[\text{DT-depth}(f \restriction \boldsymbol{\rho}) \geq t] \qquad \ldots \quad (\text{Fact } 6) \\
&\leq s \cdot 2^{-t}
\end{aligned}$$

Since $t = \frac{r^{1/d}}{20} \implies t/p = r/2$. Finally, from Lemma 9, $W^{\geq r/2}(f) \leq 2s \cdot 2^{-t} \implies W^{\geq r}(f) \leq 2s \cdot 2^{-t} = 2s \cdot 2^{\frac{-r^{1/d}}{20}}$. This completes our proof of Theorem 7. ■

## References

[Baz07] L.M.J. Bazzi. Polylogarithmic independence can fool DNF formulas. *IEEE Computer Society Press*, pages 63–73, 2007. Prelim. ver. in *Proc. of FOCS'07*. 1

[Bra09] M. Braverman. Poly-logarithmic independence fools $\mathsf{AC}^0$ circuits. *Journal of the ACM*, 57:1–10, 2009. Prelim. ver. in *Proc. of CCC'09*. 1

[BRS91] R. Beigel, N. Reingold, and D. Spielman. The perceptron strikes back. *IEEE*, 1991. Prelim. ver. in *Proc. of SCT'91*. 3a

[LMN89] N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, fourier transform, and learnability. *IEEE*, 1989. Prelim. ver. in *Proc. of FOCS'89*. 3b

[LN90] N. Linial and N. Nisan. Approximate inclusion-exclusion. *Combinatorica*, 10:349–365, 1990. Prelim. ver. in *Proc. of STOC'90*. 1

[Raz08] A. Razborov. A simple proof of bazzi's theorem. *Electronic Colloquium on Computational Complexity (ECCC)*, 2008. 1

[ST19] R. Servedio and L. Tan. Improved pseudorandom generators from pseudorandom multi-switching lemmas. *Theory of Computing*, 2019. Prelim. ver. in *Proc. of RANDOM'19*. 1