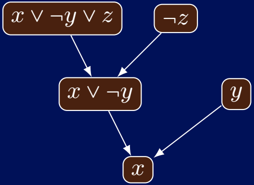
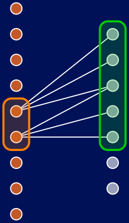
Random $\log(n)$-CNF are Hard for Cutting Planes (Again)





Dmitry Sokolov

Simons Institute
March 20, 2023

Proof Systems

Definition[Cook, Reckhow 79]

Proof system for $L \Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w \; \Pi(x, w) = 1$;
- (soundness) $\exists w \; \Pi(x, w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \ldots, D_\ell)$:

Proof Systems

Definition[Cook, Reckhow 79]

Proof system for L $\Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w\ \Pi(x,w) = 1$;
- (soundness) $\exists w\ \Pi(x,w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \ldots, D_\ell)$:

- $D_i \in \{C_i\}$;

Proof Systems

Definition[Cook, Reckhow 79]

Proof system for L $\Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w\ \Pi(x,w) = 1$;
- (soundness) $\exists w\ \Pi(x,w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \dots, D_\ell)$:

- $D_i \in \{C_i\}$;
- $\frac{A \vee x \quad B \vee \bar{x}}{A \vee B}$,
  $D_i \coloneqq A \vee B$;

Proof Systems

Definition[Cook, Reckhow 79]

Proof system for L $\Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w\ \Pi(x,w) = 1$;
- (soundness) $\exists w\ \Pi(x,w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \ldots, D_\ell)$:

- $D_i \in \{C_i\}$;
- $\frac{A \vee x \quad B \vee \bar{x}}{A \vee B}$, $D_i \coloneqq A \vee B$;
- $D_\ell = \varnothing$.

Proof Systems

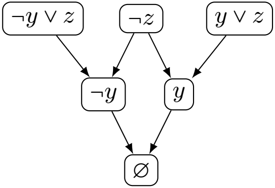
Definition[Cook, Reckhow 79]

Proof system for L $\Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w\ \Pi(x,w) = 1$;
- (soundness) $\exists w\ \Pi(x,w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \ldots, D_\ell)$:

- $D_i \in \{C_i\}$;
- $\frac{A \vee x \quad B \vee \bar{x}}{A \vee B}$, $D_i \coloneqq A \vee B$;
- $D_\ell = \varnothing$.



Proof Systems

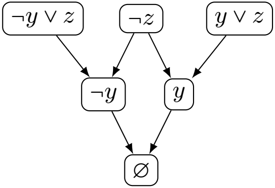
Definition[Cook, Reckhow 79]

Proof system for $L \Leftrightarrow$ poly-time algorithm Π: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$:

- (completeness) $x \in L \Rightarrow \exists w\ \Pi(x,w) = 1$;
- (soundness) $\exists w\ \Pi(x,w) = 1 \Rightarrow x \in L$.

Resolution: proof of $\varphi \coloneqq \bigwedge_i C_i$ is a sequence of clauses $(D_1, D_2, D_3, \ldots, D_\ell)$:

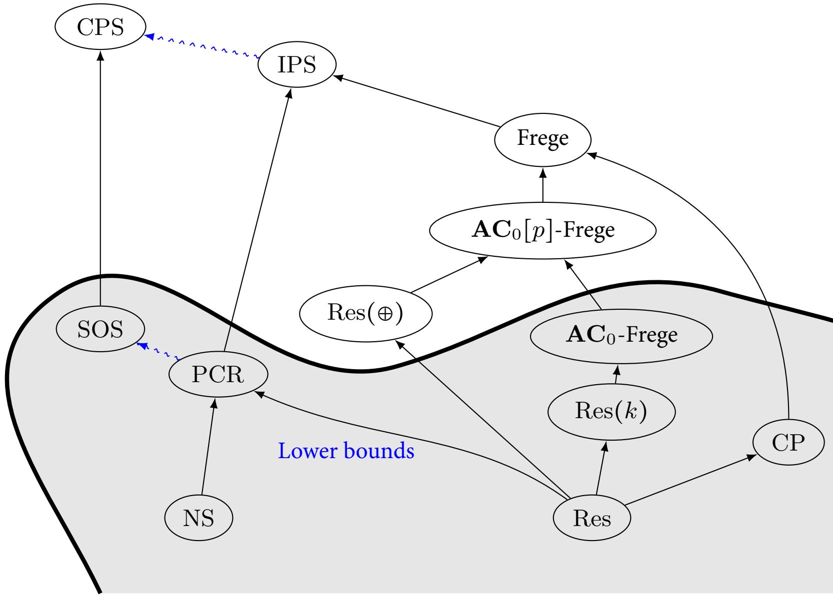
- $D_i \in \{C_i\}$;
- $\frac{A \vee x \quad B \vee \bar{x}}{A \vee B}$, $D_i \coloneqq A \vee B$;
- $D_\ell = \varnothing$.



Cutting Planes: proof is a sequence of inequalities over $\mathbb{Z}$ $(p_1 \geq 0, p_2 \geq 0, p_3 \geq 0, \ldots, p_\ell \geq 0)$:

- p_i is an encoding of $C \in \varphi$, $x_k \geq 0$ or $-x_k + 1 \geq 0$;
- $\frac{p_i \quad p_j}{p_k}$, $(p_i \geq 0) \wedge (p_j \geq 0)$ imply $(p_k \geq 0)$ over $\mathbb{Z}^n$;
- $p_\ell = 1$.

Lower bounds in proof complexity



Hard formulas for all proof systems

- If φ is unsatisfiable then there is a "proof" of unsatisfiability.

Hard formulas for all proof systems

- If φ is unsatisfiable then there is a "proof" of unsatisfiability.
  - And we can realize it in some proof system...

Hard formulas for all proof systems

- If φ is unsatisfiable then there is a "proof" of unsatisfiability.
  - And we can realize it in some proof system...
- Distribution on formulas?

Hard formulas for all proof systems

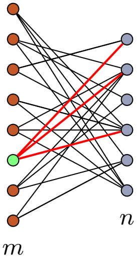
- If φ is unsatisfiable then there is a “proof” of unsatisfiability.
  - And we can realize it in some proof system...
- Distribution on formulas?
  - Fine. Counting argument do not work in proof complexity.

Hard formulas for all proof systems

- If φ is unsatisfiable then there is a "proof" of unsatisfiability.
  - And we can realize it in some proof system...
- Distribution on formulas?
  - Fine. Counting argument do not work in proof complexity.

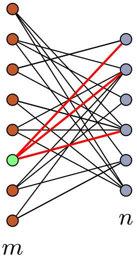
- Random Δ-CNF formulas
- Clique formulas
- Pseudorandom generator formulas

Random Δ-CNF



- m clauses;
- n variables;
- Δ neighbours: $\binom{n}{\Delta}$ possibilities;
- negations (uniformly at random);
- $\mathfrak{D} := \frac{m}{n}$ clause density.

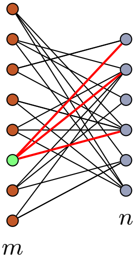
Random Δ-CNF



- m clauses;
- n variables;
- Δ neighbours: $\binom{n}{\Delta}$ possibilities;
- negations (uniformly at random);
- $\mathfrak{D} \coloneqq \frac{m}{n}$ clause density.

- $\mathfrak{D} > c_\Delta 2^\Delta \Rightarrow$ formula is unsat whp;

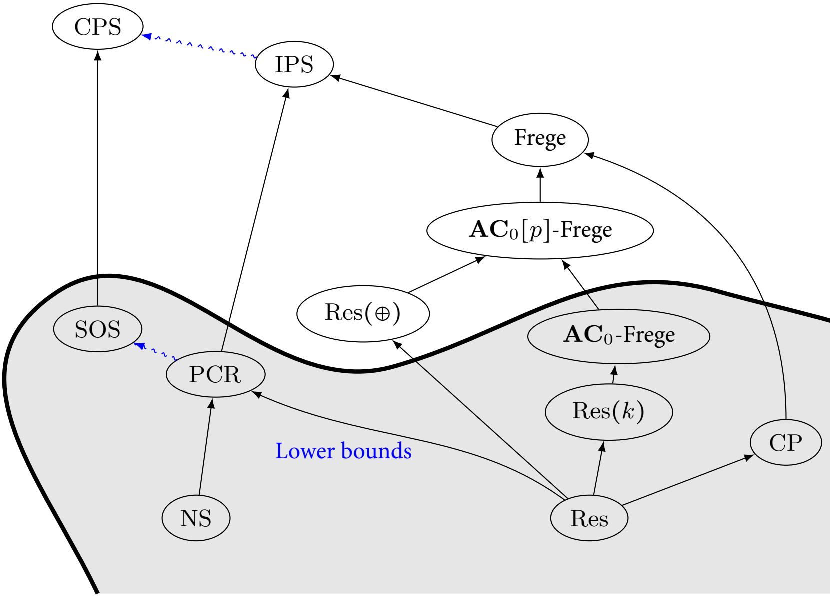
Random Δ-CNF



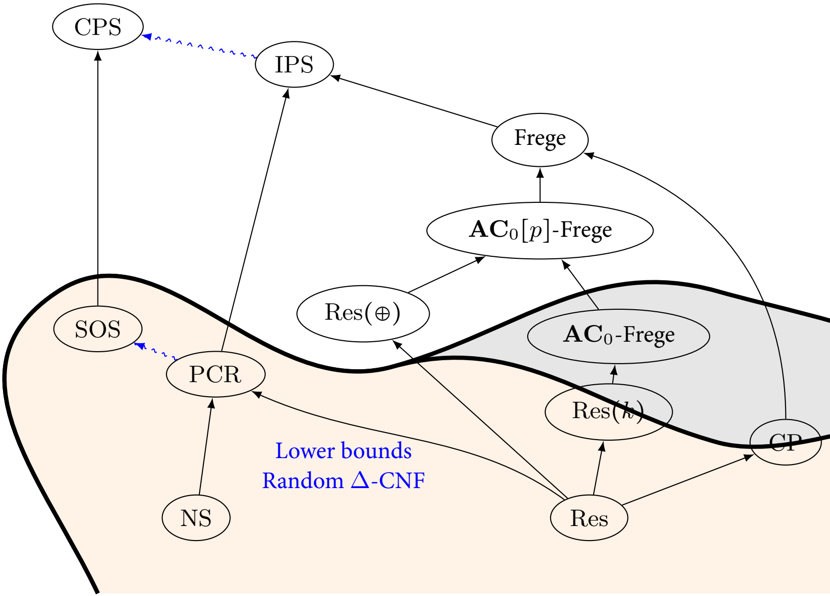
- m clauses;
- n variables;
- Δ neighbours: $\binom{n}{\Delta}$ possibilities;
- negations (uniformly at random);
- $\mathfrak{D} := \frac{m}{n}$ clause density.

- $\mathfrak{D} > c_\Delta 2^\Delta \Rightarrow$ formula is unsat whp;
- Fiege's conjecture: $\mathfrak{D} = \mathcal{O}(1) \Rightarrow$ no poly-time algorithm may "prove" unsatisfiability of random $\mathcal{O}(1)$-CNF.
  - Non-approximability of many problems.

Lower bounds in proof complexity



Lower bounds in proof complexity



Lower bounds

φ

Lower bounds

$$\varphi \Longrightarrow f_\varphi$$

f_φ is hard for monotone circuits $\Rightarrow$ φ is hard for CP

- [IPU 94, K96, P97] interpolation;
- [HP18, FPPR18] sertificate fo unsatisfiability.

Lower bounds

$$\varphi \Longrightarrow f_\varphi \Longrightarrow \text{mon ckt. lower bounds}$$

f_φ is hard for monotone circuits $\Rightarrow$ φ is hard for CP

- [IPU 94, K96, P97] interpolation;
- [HP18, FPPR18] sertificate fo unsatisfiability.

Monotone ckt. lower bounds

- [P97] approximation (clique);
- [HP18, FPPR18] Jukna's criteria.

We need monotone real circuits for the full version.

Lower bounds

$\varphi \Longrightarrow$ dag-like communication $\Longrightarrow f_\varphi \Longrightarrow$ mon ckt. lower bounds

f_φ is hard for monotone circuits $\Rightarrow \varphi$ is hard for CP

- [IPU 94, K96, P97] interpolation;
- [HP18, FPPR18] sertificate fo unsatisfiability.

Monotone ckt. lower bounds

- [P97] approximation (clique);
- [HP18, FPPR18] Jukna's criteria.

We need monotone real circuits for the full version.

Lower bounds

$\varphi \Longrightarrow$ dag-like communication $\Longrightarrow$ bottleneck counting

f_φ is hard for monotone circuits $\Rightarrow$ φ is hard for CP

- [IPU 94, K96, P97] interpolation;
- [HP18, FPPR18] sertificate fo unsatisfiability.

Monotone ckt. lower bounds

- [P97] approximation (clique);
- [HP18, FPPR18] Jukna's criteria.

We need monotone real circuits for the full version.

Unsat clause search problem $\mathsf{Search}_{\varphi}$ (Lovász et al. 1994)

$\varphi(x, y)$ is an unsatisfiable CNF formula:

- Alice gets $a \in \{0, 1\}^n$;
- Bob gets $b \in \{0, 1\}^n$;
- goal: find a clause $C \in \varphi$, such that $C(a, b) = 0$.

Unsat clause search problem Search_φ (Lovász et al. 1994)

$\varphi(x, y)$ is an unsatisfiable CNF formula:

- Alice gets $a \in \{0,1\}^n$;
- Bob gets $b \in \{0,1\}^n$;
- goal: find a clause $C \in \varphi$, such that $C(a,b) = 0$.

Balanced CNF: $\approx \Delta/2$ variables from each belongs to each player.

Unsat clause search problem Search_φ (Lovász et al. 1994)

$\varphi(x, y)$ is an unsatisfiable CNF formula:

- Alice gets $a \in \{0,1\}^n$;
- Bob gets $b \in \{0,1\}^n$;
- goal: find a clause $C \in \varphi$, such that $C(a, b) = 0$.

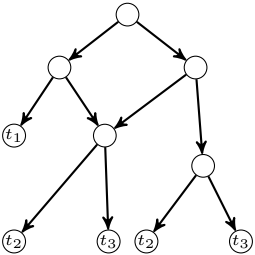
Balanced CNF: $\approx \Delta/2$ variables from each belongs to each player.

Theorem[Informal; Krajíček 98, Pudlak 99,S 17]

There is a CP-proof of φ of size $S \Rightarrow$ dag-like protocol for Search_φ of size S.

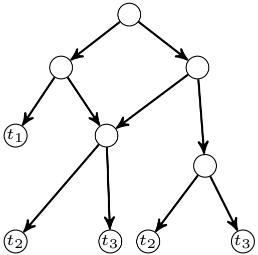
Dag-like protocols

- H is a graph with out degree 2, $\forall h \in H,\ R_h \subseteq X \times Y$;
- $R_{\text{root}} = X \times Y$;
- a, b are children of $h \Rightarrow R_h \subseteq R_a \cup R_b$;
- h is a leaf $\Rightarrow h$ is marked by common solution for R_h.

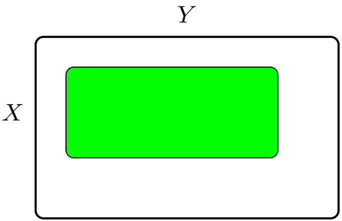


Dag-like protocols

- H is a graph with out degree 2, $\forall h \in H,\ R_h \subseteq X \times Y$;
- $R_{\text{root}} = X \times Y$;
- a, b are children of $h \Rightarrow R_h \subseteq R_a \cup R_b$;
- h is a leaf $\Rightarrow$ h is marked by common solution for R_h.



Rectangle (boolean) dag:



We need triangls instead of rectangles.

Proof Idea

- $\mu\colon X \cup Y \to H$ (partial mapping);
- $|\operatorname{Dom}(\mu)| = \Omega(\min(|X|, |Y|)) = 2^{n-\mathcal{O}(1)}$;
- $\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-f(n)}$.

Proof Idea

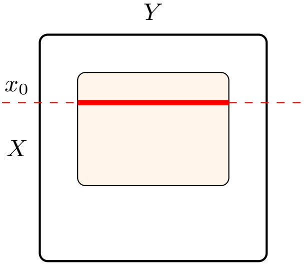
- $\mu: X \cup Y \to H$ (partial mapping);
- $|\text{Dom}(\mu)| = \Omega(\min(|X|, |Y|)) = 2^{n-\mathcal{O}(1)}$;
- $\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-f(n)}$.

Idea: $\mu(x) = h \Leftrightarrow h$ is the bottommost node where R_h contains "useful information" about x.

Proof Idea

- $\mu: X \cup Y \to H$ (partial mapping);
- $|\mathrm{Dom}(\mu)| = \Omega(\min(|X|, |Y|)) = 2^{n-\mathcal{O}(1)}$;
- $\forall h \in H, |\mu^{-1}(h)| \le 2^{n-f(n)}$.

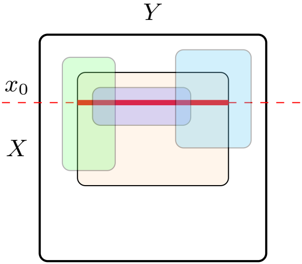
Idea: $\mu(x) = h \Leftrightarrow h$ is the bottommost node where R_h contains "useful information" about x.



Proof Idea

- $\mu: X \cup Y \to H$ (partial mapping);
- $|\mathrm{Dom}(\mu)| = \Omega(\min(|X|, |Y|)) = 2^{n-\mathcal{O}(1)}$;
- $\forall h \in H, |\mu^{-1}(h)| \le 2^{n-f(n)}$.

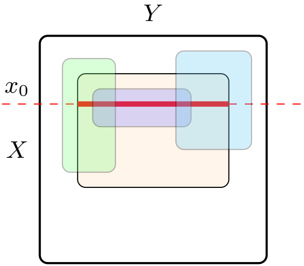
Idea: $\mu(x) = h \Leftrightarrow h$ is the bottommost node where R_h contains "useful information" about x.



Proof Idea

- $\mu: X \cup Y \to H$ (partial mapping);
- $|\text{Dom}(\mu)| = \Omega(\min(|X|, |Y|)) = 2^{n-\mathcal{O}(1)}$;
- $\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-f(n)}$.

Idea: $\mu(x) = h \Leftrightarrow h$ is the bottommost node where R_h contains "useful information" about x.



- $w(h, x_0) \coloneqq$ size of minimal monochr. covering
- $k \coloneqq n/\log(n)$
- $\mu(x_0)$ = the bottommost h such that $w(h, x_0) \geq k$.

Definition of μ

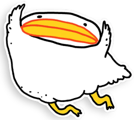
1. For all $h \in H$ from leafs to root.

Definition of μ

1. For all $h \in H$ from leafs to root.
2. $\forall x \in X, w(h, x) > k \Rightarrow$
   - $\mu(x) := h$;
   - erase $\{x\} \times Y$ from all rectangles in H.
3. $\forall y \in X, w(h, y) > k \Rightarrow$
   - $\mu(y) := h$;
   - erase $X \times \{y\}$ from all rectangles in H.

Definition of μ

1. For all $h \in H$ from leafs to root.
2. $\forall x \in X, w(h, x) > k \Rightarrow$
   - $\mu(x) := h$;
   - erase $\{x\} \times Y$ from all rectangles in H.
3. $\forall y \in X, w(h, y) > k \Rightarrow$
   - $\mu(y) := h$;
   - erase $X \times \{y\}$ from all rectangles in H.
4. Goto next h.



Lemma

At current node h

- before: $\forall z \in X \cup Y, w(h, z) \leq 2k$;
- after: $\forall z \in X \cup Y, w(h, z) \leq k$.

First property

Lemma

$|\mathrm{Dom}(\mu)| \geq \min(|X|, |Y|)/2.$

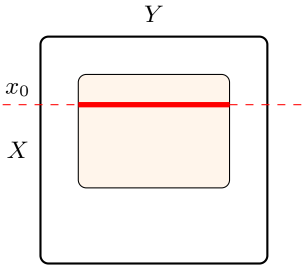
Proof.

First property

Lemma

$|\mathrm{Dom}(\mu)| \geq \min(|X|, |Y|)/2.$

Proof.

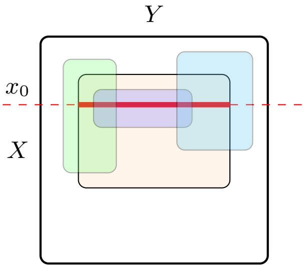


First property

Lemma

$|\mathrm{Dom}(\mu)| \geq \min(|X|, |Y|)/2.$

Proof.

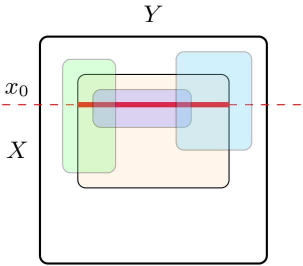


First property

Lemma

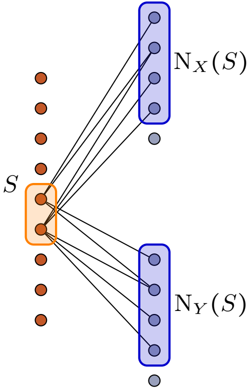
$|\mathrm{Dom}(\mu)| \geq \min(|X|, |Y|)/2.$

Proof.



$$w(\mathrm{root}, x_0) \leq k \Rightarrow \exists S \subseteq \varphi, |S| \leq k : \forall y \in Y_{\mathrm{root}}, S(x_0, y) = 0$$
$$\Rightarrow |Y_{\mathrm{root}}| \leq k/2^{\Delta} \cdot |Y|.$$

Expansion



- (r, Δ, c)-expander;
- $\forall S \subseteq L, |S| \leq r \Rightarrow$
  - $N_X(S) \geq c|S|$;
  - $N_y(S) \geq c|S|$.

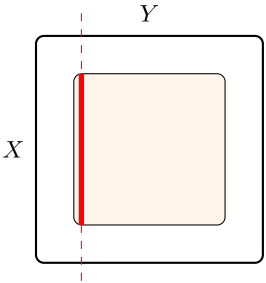
Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

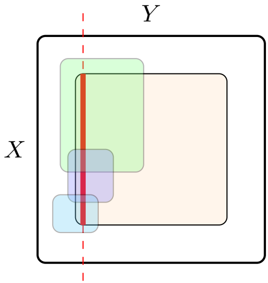
Proof.

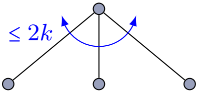


Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Proof.

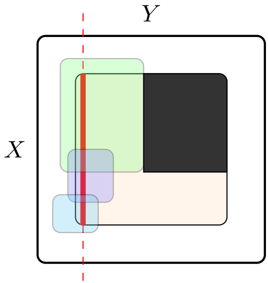


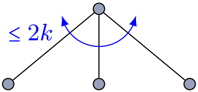


Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Proof.

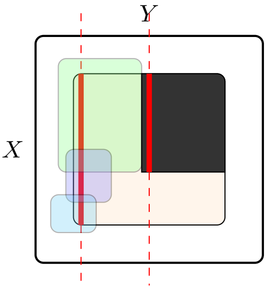


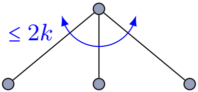


Lemma

$\forall h \in H, |\mu^{-1}(h)| \le 2^{n-\Omega(k \log k)}$.

Proof.

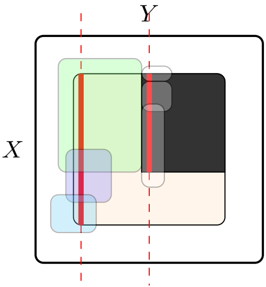


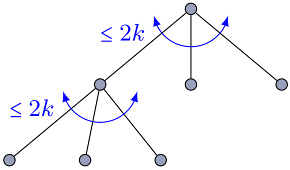


Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Proof.

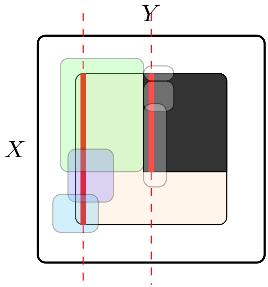


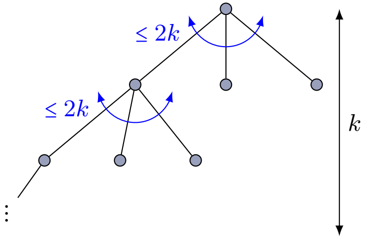


Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Proof.

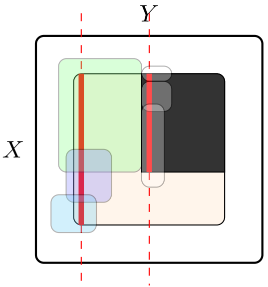


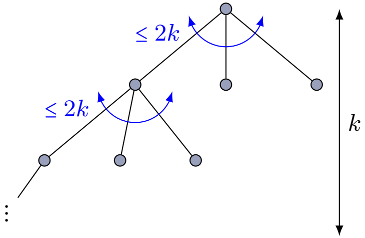


Lemma

$\forall h \in H, |\mu^{-1}(h)| \leq 2^{n-\Omega(k \log k)}$.

Proof.

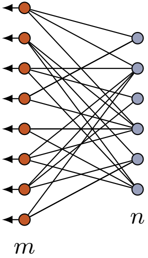




- $x_0 \in$ leaf $\Rightarrow \exists S \subseteq \varphi, |S| \leq k, x_0$ do not satisfy any clause in S.
- Expansion in $X \Rightarrow$ at most 2^{n-ck} such x.
- There are at most $(2k)^k$ leaves.
- Altogether: $|\mu^{-1}(h)| \leq 2^{n-ck+k\log 2k}$

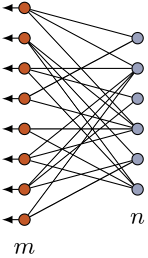
□

Open Problemas: Nisan–Wigderson Generators (naive encoding)



- Δ is the left degree;
- $P(x_1, \ldots, x_\Delta)$ is a predicate.

Open Problemas: Nisan–Wigderson Generators (naive encoding)



- Δ is the left degree;
- $P(x_1, \ldots, x_\Delta)$ is a predicate.

- Strategy do not work for balanced predicates;
- Upper bound if P is Parity;
- **P**/poly vs **NP**;

Open problems

- PRG. Other encodings.
- $\mathcal{O}(1)$-random CNF.
- “Sepataion” betweem CP and monotone circuits.