# The Complexity of Hardness Amplification and Derandomization

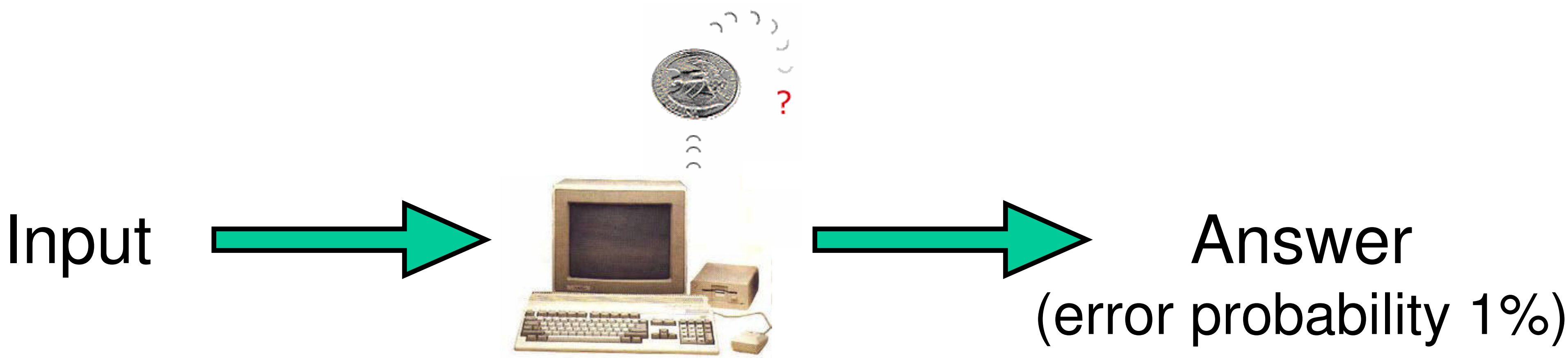

Emanuele Viola

Harvard University


Ph.D. Defense

May 2006

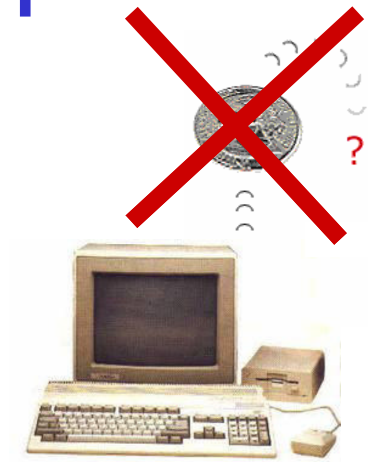
# Randomness in Computation




- Useful throughout Computer Science
  - Cryptography
  - Learning Theory
  - Complexity Theory

- Question: Is Randomness necessary?

# Derandomization




- Goal: remove randomness

- Why study derandomization?

- Breakthrough [R '04]:
  Connectivity in logarithmic space (SL = L)

- Breakthrough [AKS '02]:
  Primality in polynomial time (PRIMES $\in$ P)

# Randomness vs. Time

- Goal:
  simulate randomized computation deterministically

- Trivial Derandomization:
  If A uses n random bits, enumerate all $2^n$ possibilities

  Probabilistic polynomial-time $\subseteq$ exponential time
  BPP $\subseteq$ Time($2^{poly(n)}$)

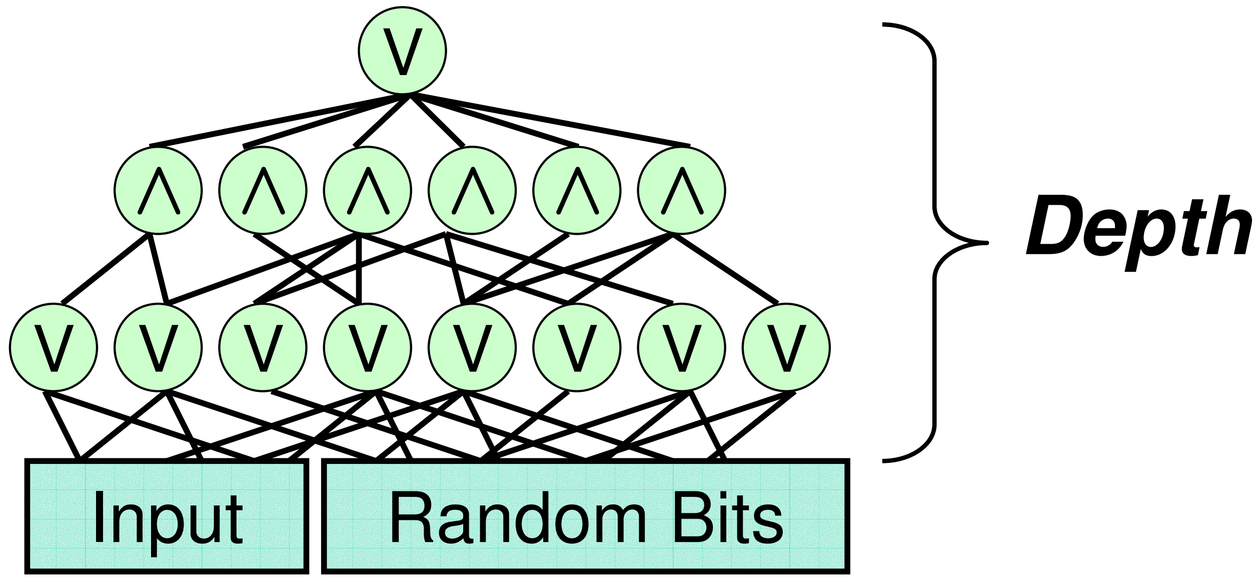
- Strong Belief: BPP = P ( Time(poly(n)) )
  Complexity Assumptions $\Rightarrow$ BPP = P [BFNW,NW,IW,…]

# Outline

- Overview of derandomization
- Derandomization of restricted models
  - Application: Hardness Amplification in NP
  - New derandomization
- Derandomization of general models
  - BPP vs. PH
  - Proof of Lower Bound

# Constant-Depth Circuits

- Probabilistic constant-depth circuit (BP $AC^0$ )




- Theorem [N '91]: BP $AC^0 \subseteq \text{Time}(n^{\text{polylog } n})$
  - Compare to BP P $\subseteq \text{Time}(2^{\text{poly}(n)})$

# Application: Avg-Case Hardness of NP

- Study hardness of NP on random instances
  - Natural question, essential for cryptography

- Currently cannot relate to P $\neq$ NP [FF,BT,V]

- Hardness amplification

  Definition: $f : \{0,1\}^n \rightarrow \{0,1\}$ is $\delta$-hard if

  for every efficient algorithm M : $\Pr_x[M(x) \neq f(x)] \geq \delta$

f → **Hardness Amplification** → f '

.01-hard → (1/2 - $\varepsilon$)-hard

# Previous Results

- Yao's XOR Lemma: $f'(x_1,\ldots, x_n) := f(x_1) \oplus \cdots \oplus f(x_n)$
  $f' \approx (1/2 - 2^{-n})$-hard, almost optimal

- Cannot use XOR in NP: $f \in NP \not\Rightarrow f' \in NP$

- Idea: $f'(x_1,\ldots, x_n) = C( f(x_1),\ldots, f(x_n) )$, C monotone
  - e.g. $f(x_1) \wedge ( f(x_2) \vee f(x_3) )$. $f \in NP \Rightarrow f' \in NP$

- Theorem [O'D]: There is C s.t. $f' \approx (1/2 - 1/n)$-hard

- Barrier: No monotone C can do better!

# Our Result on Hardness Amplification

- Theorem [HVV]: Amplification in NP up to $\approx 1/2 - 2^{-n}$
  - Matches the XOR Lemma

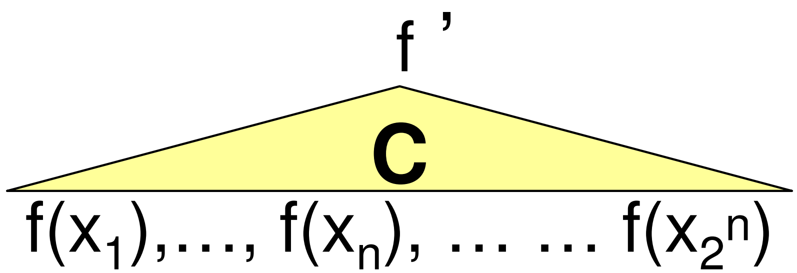
- Technique: Derandomize!

  Intuitively, $f' := C(\ f(x_1),\ldots, f(x_n),\ \ldots\ \ldots\ f(x_{2^n})\ )$

  $f'$ $(1/2 - 1/2^n)$-hard by previous result

  Problem: Input length = $2^n$

  Note C is constant-depth




  Derandomize: input length $\to$ n, keep hardness

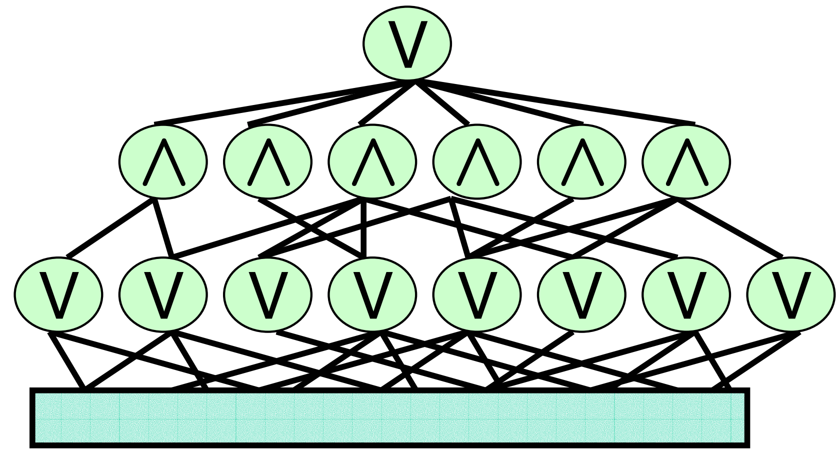
# Outline

- Overview of derandomization
- Derandomization of restricted models
  - Application: Hardness Amplification in NP
  - New derandomization
- Derandomization of general models
  - BPP vs. PH
  - Proof of Lower Bound

# Previous Results

- Recall Theorem [N]:

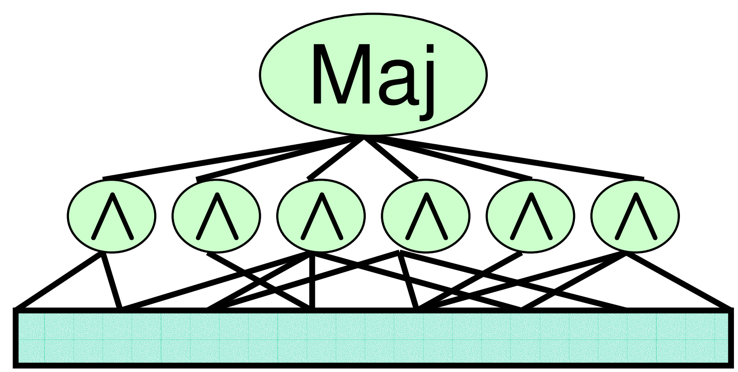
  BP $AC^0 \subseteq$ Time($n^{\text{polylog } n}$)




- But $AC^0$ is weak: Majority $\notin$ $AC^0$
  - Majority($x_1,\ldots,x_n$) := $\sum_i x_i > n/2$ ?

- Theorem [LVW]:

  BP Maj AND $\subseteq$ Time($2^{n^{\varepsilon}}$)

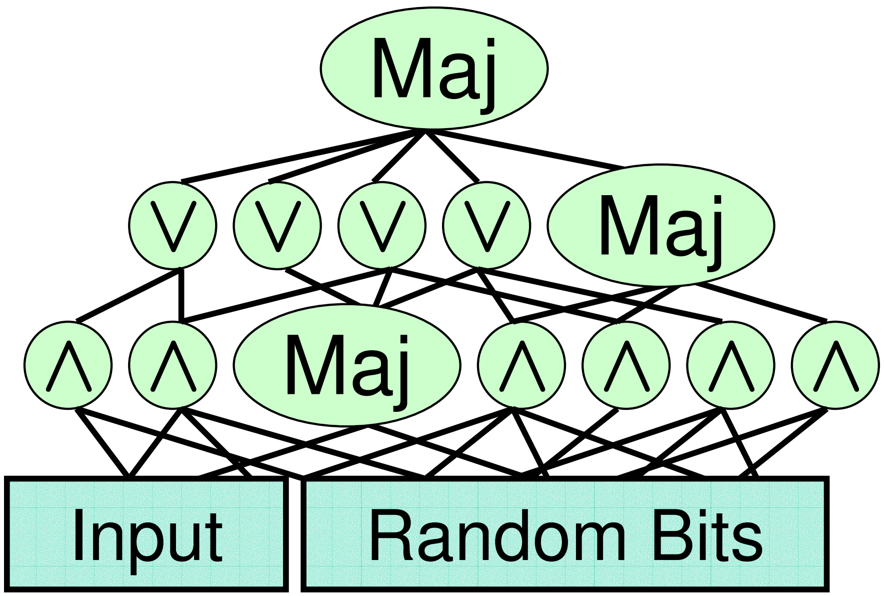



- Derandomize incomparable classes

# Our New Derandomization

- Theorem **[V] :** BP Maj $AC^0 \subseteq$ Time($2^{n^{\varepsilon}}$)

  Derandomize
  constant-depth circuits
  with few Majority gates =




- Improves on [LVW]. Slower than [N] but richer
  richest probabilistic circuit class in Time($2^{n^{\varepsilon}}$)

- Techniques: Communication complexity +
  switching lemma [BNS,HG,H,HM,CH]

# Outline

- Overview of derandomization
- Derandomization of restricted models
  - Application: Hardness Amplification in NP
  - New derandomization
- Derandomization of general models
  - BPP vs. PH
  - Proof of Lower Bound

# BPP vs. POLY-TIME HIERARCHY

- Probabilistic Polynomial Time (BPP):
  for every x, Pr [ M(x) errs ] $\leq$ 1%

- Strong belief: BPP = P [NW,BFNW,IW,…]
  Still open: BPP $\subseteq$ NP ?

- Theorem [SG,L; '83]: BPP $\subseteq \Sigma_2$ P

- Recall
  NP = $\Sigma_1$ P $\rightarrow$ $\exists$ y M(x,y)
  $\Sigma_2$ P $\rightarrow$ $\exists$ y $\forall$ z M(x,y,z)

# The Problem we Study

- More precisely [SG,L] give

  $\mathrm{BPTime}(t) \subseteq \Sigma_2\mathrm{Time}(t^2)$

- Question[Rest of this Talk]:

  Is quadratic slow-down necessary?

- Motivation: Lower bounds

  Know NTime ≠ Time on some models [P+,F+,…]

  Technique: *speed-up* computation with quantifiers

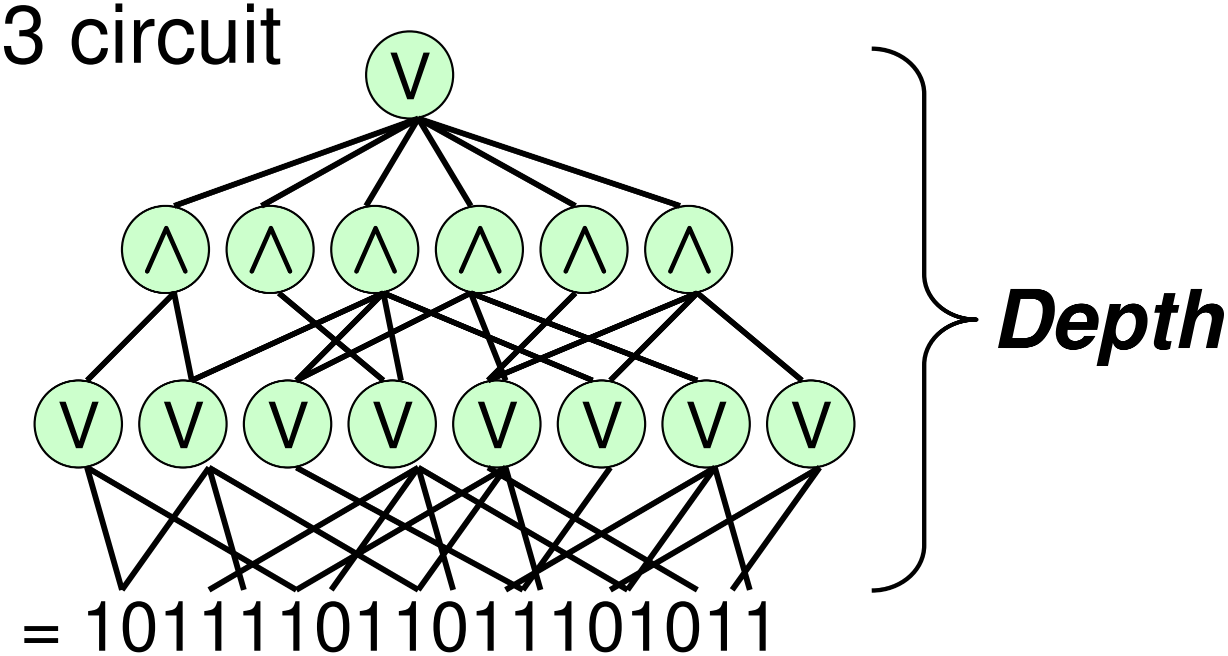
  To prove NTime ≠ BPTime cannot afford $\Sigma_2\mathrm{Time}(t^2)$

# Approximate Majority

- Input: R = 1011110110111101011
- Task: Tell $\Pr_i[R_i = 1] \geq 99\%$ from $\Pr_i[R_i = 1] \leq 1\%$

  Do not care if $\Pr_i[R_i = 1] \sim 50\%$ (approximate)

- Model: Depth-3 circuit



# The connection [FSS]

$M(x;u) \in \text{BPTime}(t)$ ⇒ $R = 1101101110101 1$

$|R| = 2^t$ ↳ $R_i = M(x;i)$

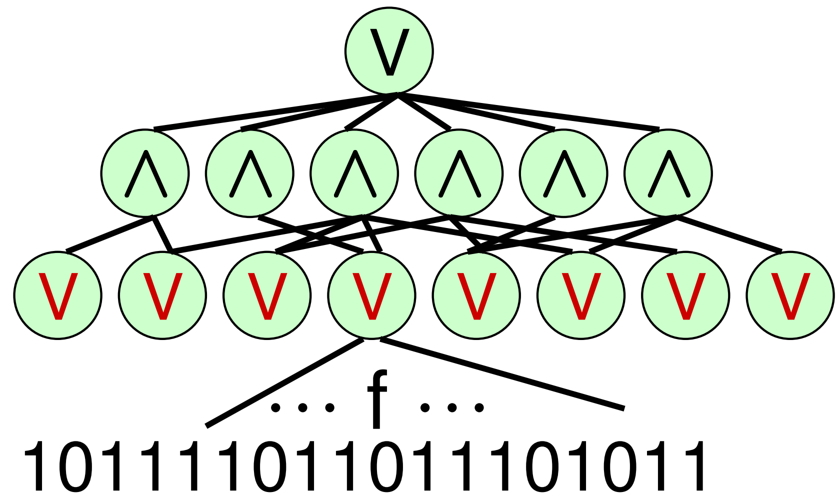
Compute M(x):

Tell $\Pr_u[M(x) = 1] \geq 99\%$
from $\Pr_u[M(x) = 1] \leq 1\%$

⇒ Compute Appr-Maj

$\text{BPTime}(t) \subseteq \Sigma_2 \text{Time}(t')$
$= \exists \forall \text{ Time}(t')$

⇒




**Running time t'**
– run M at most t'/t times

⇒ **Bottom fan-in f = t' / t**

# Our Results

- **Theorem[V] :** Small depth-3 circuits for Approximate Majority on N bits have bottom fan-in $\Omega(\log N)$

- **Corollary**: Quadratic slow-down necessary for relativizing techniques:

$$\mathrm{BPTime}^{A}(t) \not\subseteq \Sigma_2\mathrm{Time}^{A}(t^{1.99})$$

- **Theorem**[DvM,V]: BPTime (t) $\subseteq \Sigma_3$Time (t $\cdot$ $\log^5$ t)
  - Previous result [A]: BPTime (t) $\subseteq \Sigma_{O(1)}$Time( t )

- For time, the level is the third!

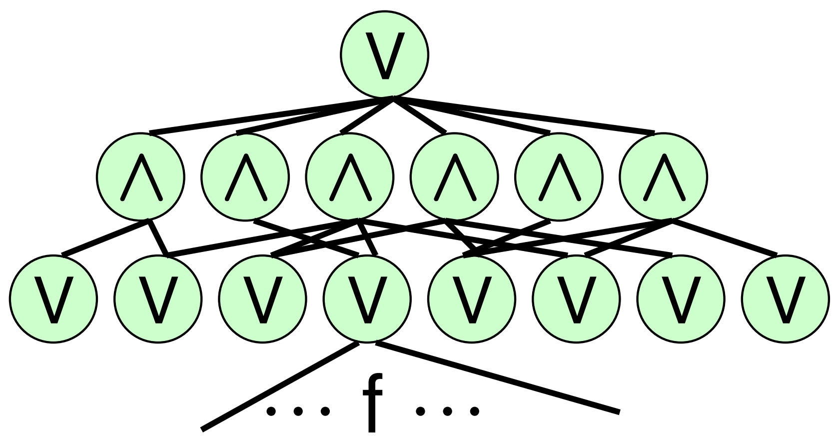
# Outline

- Overview of derandomization
- Derandomization of restricted models
  - Application: Hardness Amplification in NP
  - New derandomization
- Derandomization of general models
  - BPP vs. PH
  - Proof of Lower Bound

# Our Negative Result

- **Theorem[V]:** $2^{N^{\varepsilon}}$-size depth-3 circuits for Approximate Majority on N bits have bottom fan-in $f = \Omega(\log N)$

- Recall:



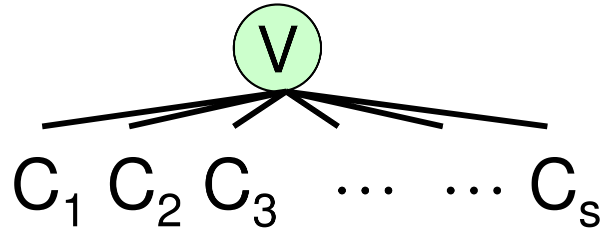

R = 1011110110111101011 $|R| = N$

Tells $R \in$ YES := { R : $Pr_i$ [ $R_i$ = 1] $\geq$ 99% }

from $R \in$ NO := { R : $Pr_i$ [ $R_i$ = 1] $\leq$ 1% }

# Proof

- Circuit is OR of s depth-2 circuits




- By definition of OR :

  $R \in YES \Rightarrow$ some $C_i\ (R) = 1$

  $R \in NO \Rightarrow$ all $C_i\ (R) = 0$

- By averaging, fix $C = C_i$ s.t.

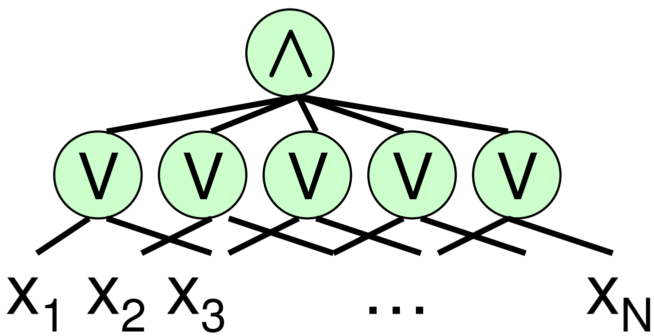
  $\Pr_{R \in YES}[C\ (x) = 1\ ] \geq 1/s$

  $\forall\ R \in NO \Rightarrow C\ (R) = 0$

- Claim: Impossible if C has bottom fan-in $\leq \varepsilon \log N$

# CNF Claim

- Depth-2 circuit $\Rightarrow$ CNF




$(x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_4) \wedge (x_5 \vee x_3)$

bottom fan-in $\Rightarrow$ **clause** size

- **Claim:** All CNF C with clauses of size $\varepsilon \cdot \log N$

  Either $\Pr_{R \in YES}[C(x) = 1] \leq 1 / 2^{N^{\varepsilon}}$
  or there is $R \in NO : C(x) = 1$

- Note: Claim $\Rightarrow$ Theorem

Either $\Pr_{R \in YES}[C(x)=1] \leq 1/2^{N^{\varepsilon}}$ or $\exists\, R \in NO : C(x) = 1$

# Proof Outline

- Definition: $S \subseteq \{x_1, x_2, \ldots, x_N\}$ is a covering if every clause has a variable in S

  E.g.: $S = \{x_3, x_4\}$ $C = (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_4) \wedge (x_5 \vee x_3)$

- Proof idea: Consider **smallest** covering S

  Case |S| BIG : $\Pr_{R \in YES}[C(x) = 1] \leq 1 / 2^{N^{\varepsilon}}$

  Case |S| tiny : Fix few variables and repeat

Either $\Pr_{R \in YES} [C(x)=1] \leq 1/2^{N^{\varepsilon}}$ or $\exists\, R \in NO : C(x) = 1$

# Case |S| BIG

- $|S| \geq N^{\delta} \;\Rightarrow$ have $N^{\delta} / (\varepsilon \cdot \log N)$ disjoint clauses $\Gamma_i$
  - Can find $\Gamma_i$ greedily

- $\Pr_{R \in YES} \left[C(R) = 1\right] \leq \Pr \left[\, \forall\, i, \Gamma_i(R) = 1 \,\right]$

$$= \prod_i \Pr[\, \Gamma_i(R) = 1] \qquad \text{(independence)}$$

$$\leq \prod_i (1 - 1/100^{\varepsilon \log N}) = \prod_i \left(1 - 1/N^{O(\varepsilon)}\right)$$

$$= \left(1 - 1/N^{O(\varepsilon)}\right)^{|S|} \leq e^{-N^{\Omega(1)}} \quad \checkmark$$

Either $Pr_{R \in YES}[C(x)=1] \leq 1/2^{N^{\varepsilon}}$ or $\exists\, R \in NO : C(x) = 1$

# Case |S| tiny

- $|S| < N^{\delta}$ $\Rightarrow$ Fix variables in S
  - Maximize $Pr_{R \in YES}[C(x)=1]$

- Note: S covering $\Rightarrow$ clauses shrink

  Example

  $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_3) \wedge (x_5 \vee \neg x_4)$ $\boxed{x_3 \leftarrow 0,\ x_4 \leftarrow 1}$ $\Rightarrow$ $(x_1 \vee x_2) \wedge (x_5)$

- Repeat

  Consider smallest covering S', etc.

Either $Pr_{R \in YES}[C(x)=1] \leq 1/2^{N^{\varepsilon}}$ or $\exists\, R \in NO : C(x) = 1$

# Finish up

- Recall: Repeat $\Rightarrow$ shrink clauses
  So repeat at most $\varepsilon \cdot \log N$ times

- When you stop:
  Either smallest covering size $\geq N^{\delta}$ ✓
  Or C = 1
  Fixed $\leq (\varepsilon \cdot \log N)\, N^{\delta} \ll N$ vars.
  Set rest to 0 $\Rightarrow R \in NO : C(R) = 1$ ✓

Q.E.D.

# Conclusion

- Derandomization: powerful technique

- Restricted models: Constant-depth circuits ($AC^0$)
  - Derandomization of $AC^0$ [N]
  - Application: Hardness Amplification in NP [HVV]
  - Derandomization of $AC^0$ with few Maj gates [V]

- General models: BPP vs. PH
  - $\mathrm{BPTime}(t) \subseteq \Sigma_2\mathrm{Time}(t^2)$ [SG,L]
  - $\mathrm{BPTime}(t) \not\subseteq \Sigma_2\mathrm{Time}(t^{1.99})$ (w.r.t. oracle) [V]
    Lower Bound for Approximate Majority

**Thank you!**