Lower Bounds

Emanuele Viola

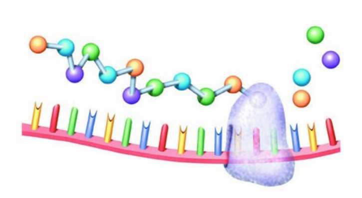
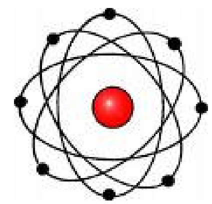
Columbia University

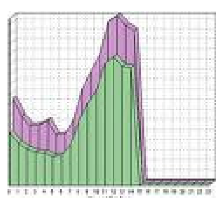
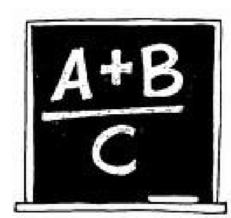
February 2008

Computation

- Efficient computation is fundamental to Science

  Increasingly important to many fields

  biology  physics 

  economics  mathematics 

- Goal: understand efficient computation

Lower Bounds

- Goal: Show that natural problems cannot be solved with limited resources (e.g., time, memory,…)

  E.g.: Cannot factor n-digit number in time n^2

- Fundamental enterprise, basis of cryptography

- Widespread belief: very challenging area

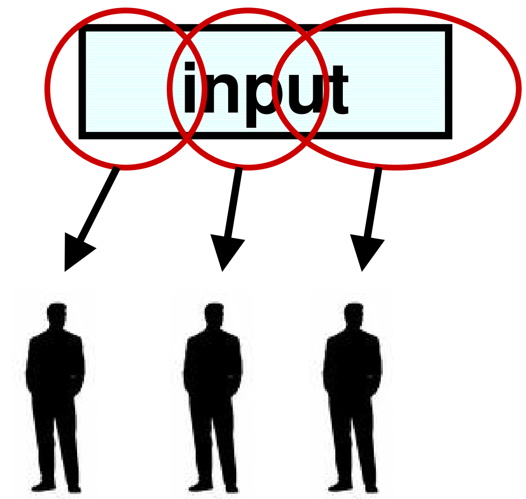
- This talk: Lower bounds for various resources
  surprising connections

Communication complexity

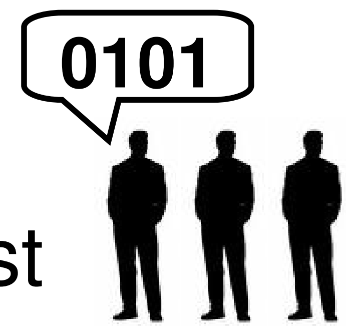
[Yao, Chandra Furst Lipton '83]

- Task: Compute function f : **input** $\rightarrow \{0,1\}$

- Input distributed among collaborating players



- Cost = how many bits players must broadcast



- E.g.: For 2 players computing "$x =^? y$" costs $\Theta(|x|)$

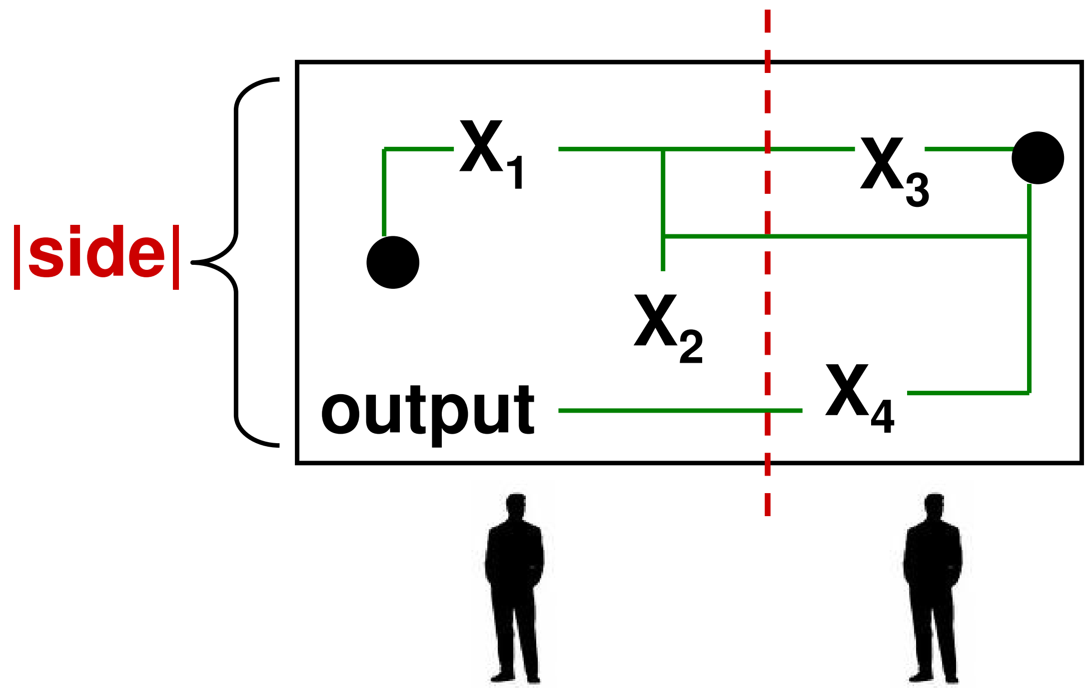
Application: CHIP design

[…, Lipton Sedgewick '81]

- Task: Design CHIP for $f : \{0,1\}^n \rightarrow \{0,1\}$

  Side length measure:
  wire width

  Wires carry 1 bit
  per time step

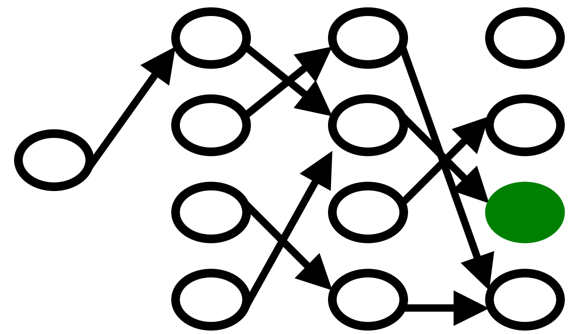


- 2-players simulate CHIP sending |side| bits per step

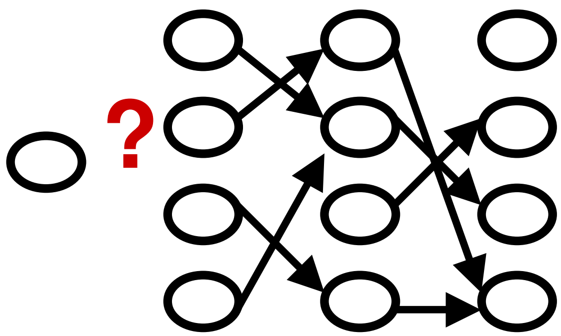
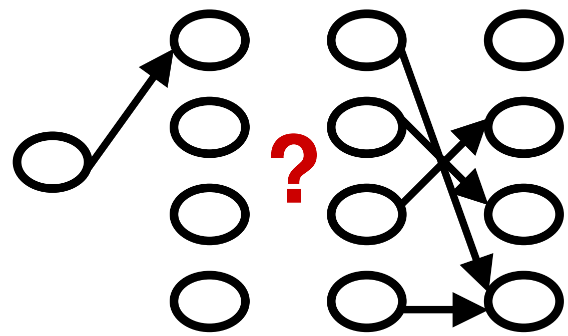
- Theorem: |side| x time > 2-player cost of f

Pointer chasing

- Input: directed depth-k graph
  Output: node reached from source



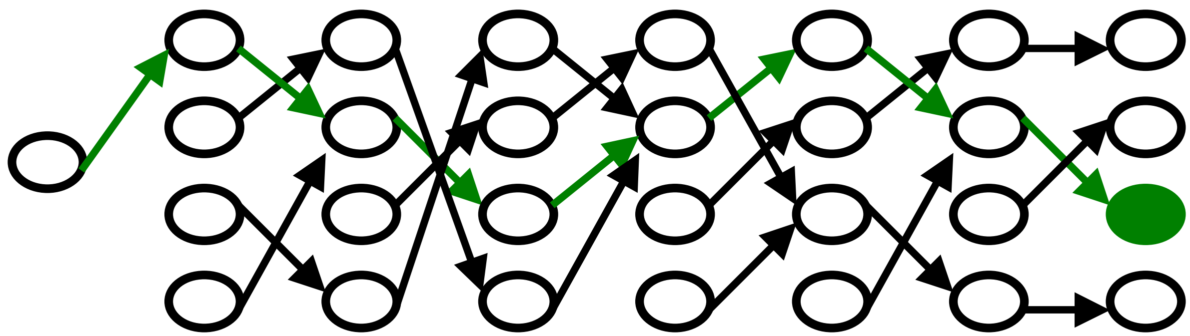
- k players speak in turn; i-th knows all but depth-i edges

Player 1:  Player 2: 

- High cost for log(|graph|) players ⇒ breakthrough
  Question[<1996]: 4 players?

Our results

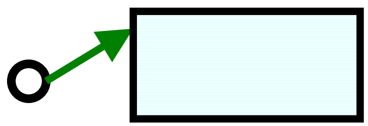
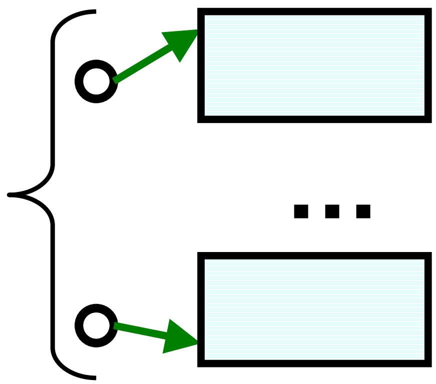
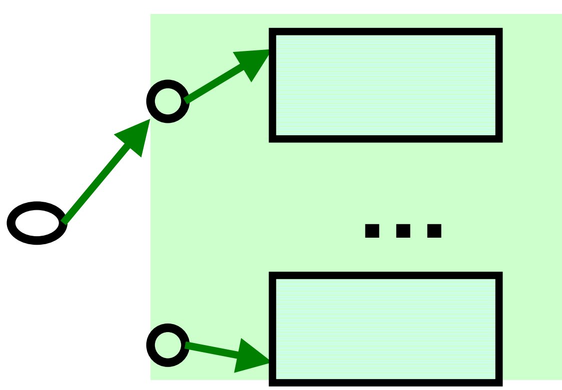
[V. Wigderson; FOCS '07 special issue]



- Theorem[VW] To chase a pointer in graph of depth k k players must communicate $\geq |\text{graph}|^{1/k}$ bits
  - Handle up to $k = \log(|\text{graph}|)^{1/3}$

- Applications:
  round hierarchy for communication
  multiple-pass streaming algorithms

Proof Idea

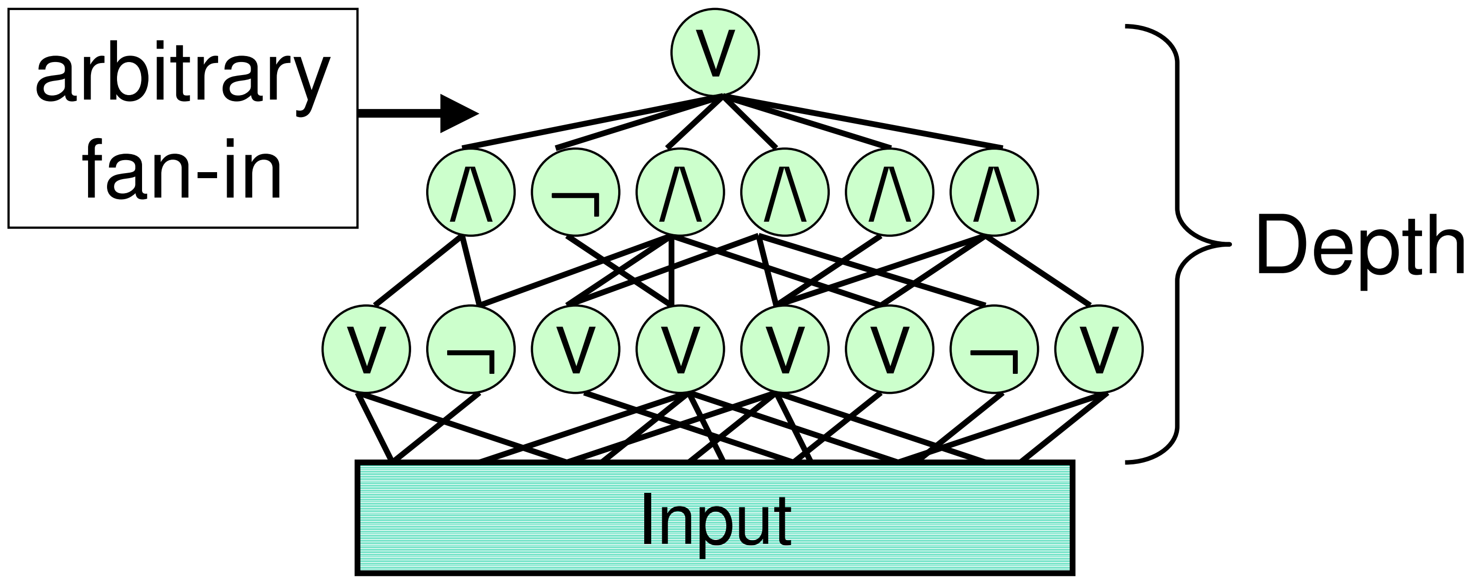
- Induction on depth = number of players
- Assume 1 chasing  high cost for k players
- $\Rightarrow$ 100 chasings  high cost for k players for most graphs
- $\Rightarrow$ 1 chasing  high cost for k+1 players
  New player's message won't help

Q.e.d.

Outline

- Communication complexity
- Circuit complexity
- Randomness vs. time

Circuits



- Gates:

  ∧ = AND

  V = OR

  ¬ = NOT

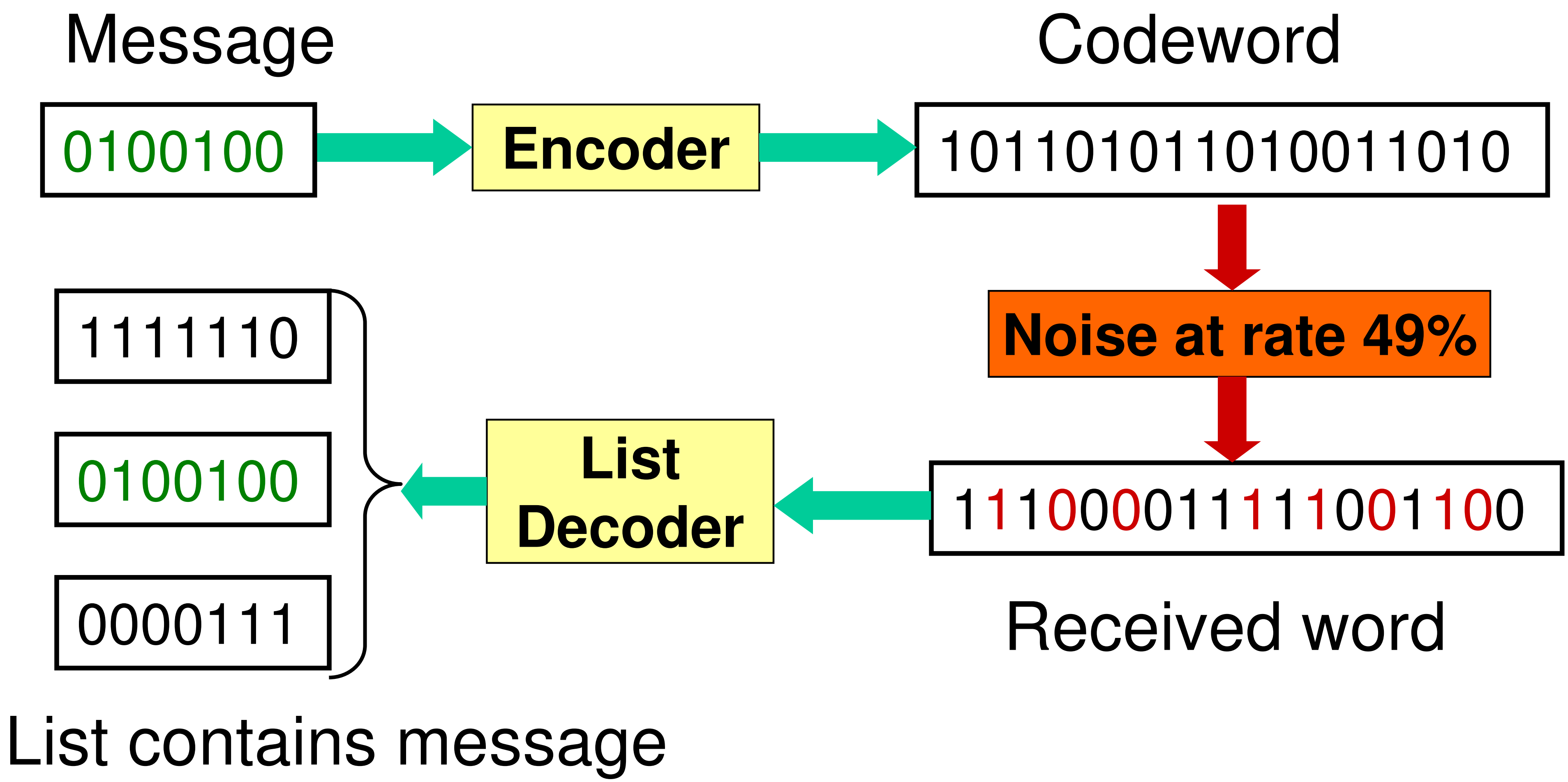
- Resource: size = number of gates

- Poly-size constant-depth = constant parallel time

- Theorem[Yao, Beigel Tarui, Hastad Goldman]

  Communication lower bound (polylog players)
  ⇒ circuit lower bound (small-depth, Mod gates)

Error correcting codes

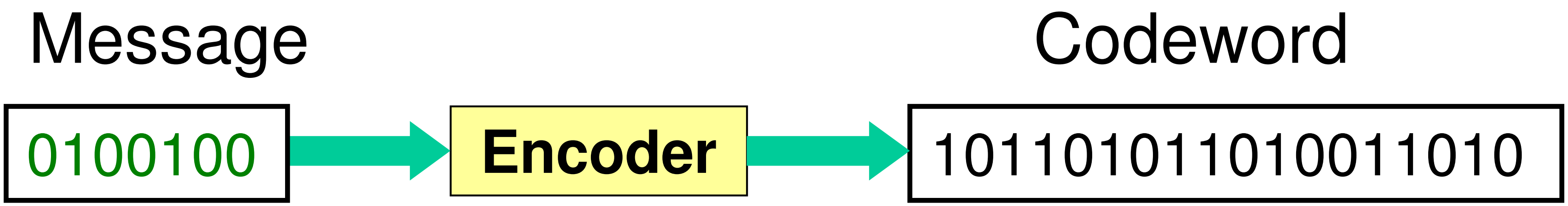


- Question: Complexity of encoder, decoder?

Motivation: average-case complexity

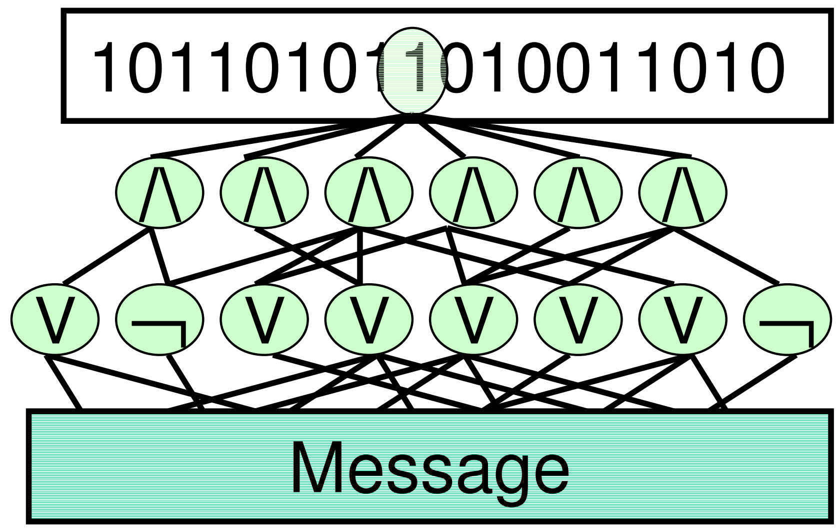
Our results: Encoding needs parity

[V.; J. Comp. Complexity]



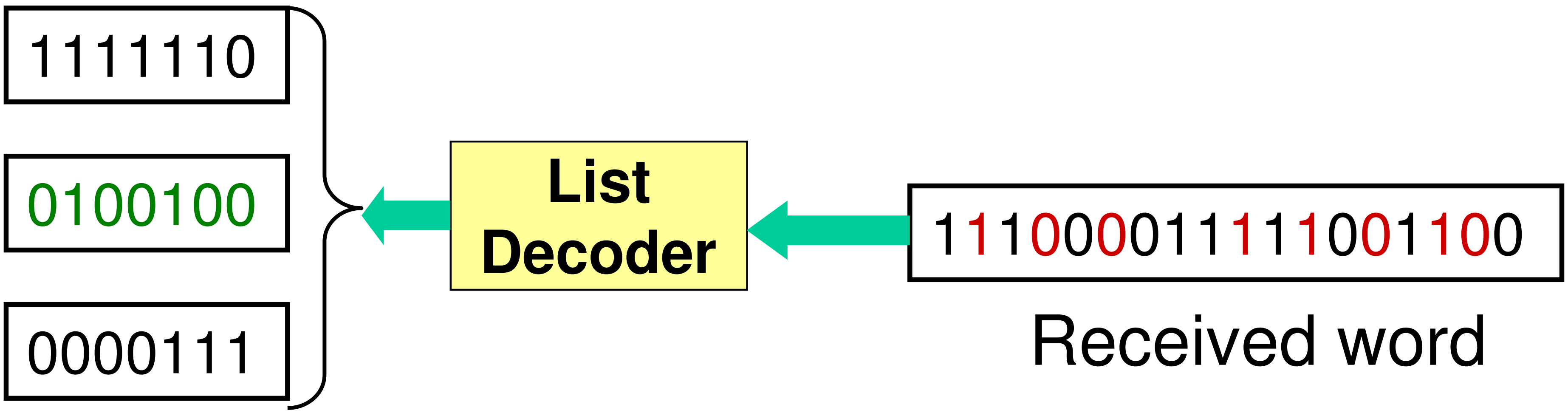
- Parity $\oplus(x_1,\ldots,x_n) := 1 \Leftrightarrow \sum_i x_i$ odd
  sufficient for encoding (e.g., linear codes)

- Theorem[V]:
  Cannot encode with small size,
  small depth, $\neg$, $\vee$, $\wedge$ gates.
  Parity is necessary

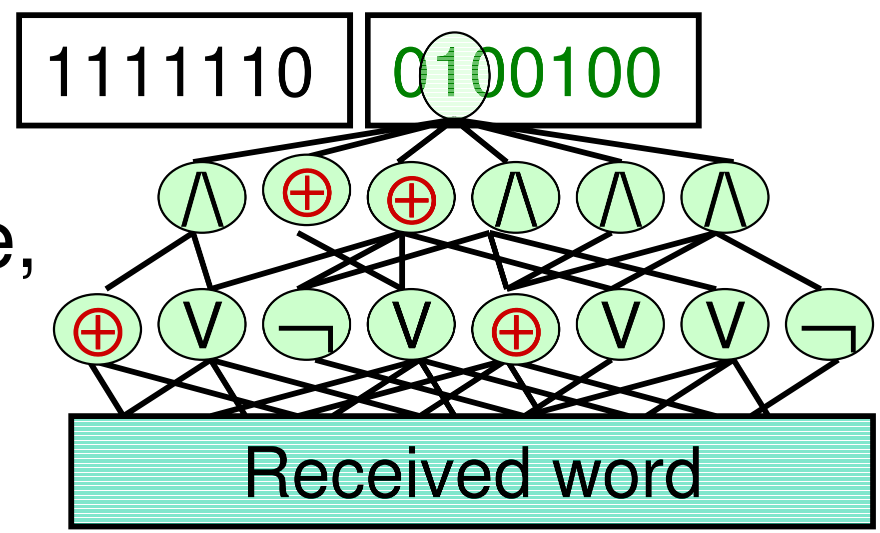


Our results: Decoding needs majority

[Shaltiel V.; STOC '08]



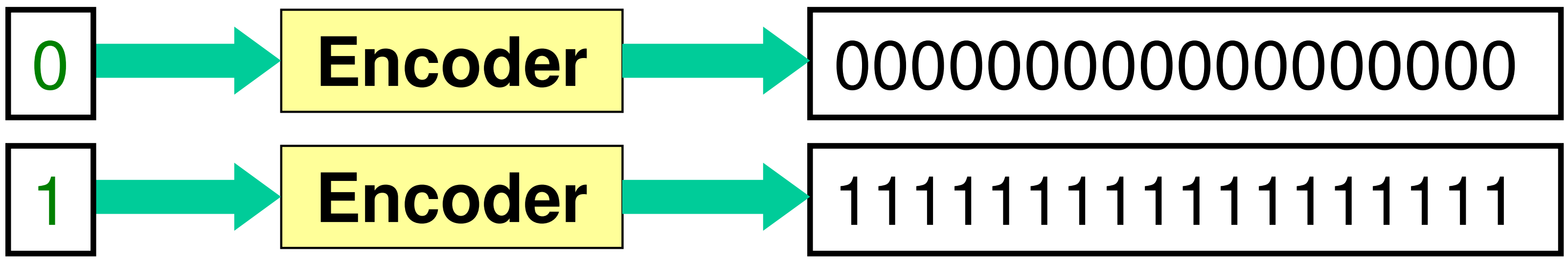
- Often more involved than encoding



- Theorem[SV]:
  Cannot decode with small size, small depth ¬, ∨, ∧, ⊕ gates.
  Majority is necessary

Proof idea: Decoding needs majority

- Repetition code:



- Decoder: Majority(101010111010101001) = 1

- Theorem[SV]: This happens in every code
  - Acknowledgment: Madhu Sudan

- Main difficulty: Large lists. Use information theory.

Outline

- Communication complexity
- Circuit complexity
- Randomness vs. time

Randomness vs. Time

- Probabilistic Time: for every x, Pr [M(x) errs] < 1%

- Deterministic simulation?

Brute force: probabilistic time t $\subseteq$ deterministic time 2^t

Belief: probabilistic time t $\subseteq$ deterministic time $t^{O(1)}$

- Surprise: Belief $\Leftrightarrow$ circuit lower bounds

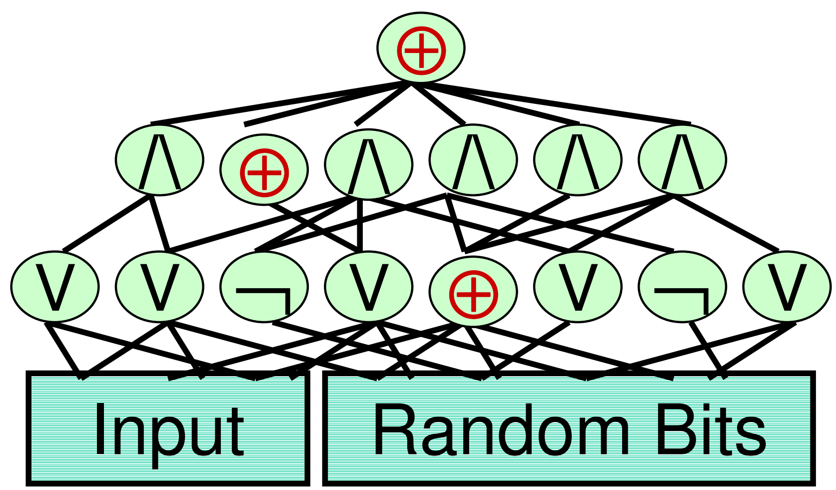
  by Babai Fortnow Kabanets Impagliazzo Nisan Wigderson…

Our Results

[V.; SIAM J. Comp., SIAM student paper prize 2006]

- Theorem[V]: Poly(n)-size probabilistic constant-depth circuits with $\neg$, $\vee$, $\wedge$, log(n) parity gates

  $\subseteq$ Deterministic Time($2^{n^{\varepsilon}}$) ($\subset$ trivial Time($2^{n^{O(1)}}$))

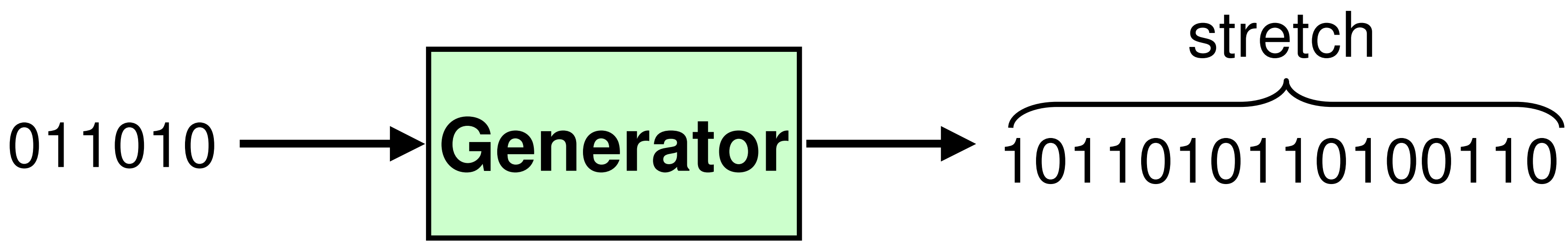


- Richest probabilistic circuit class in Time($2^{n^{\varepsilon}}$)

- Proof: Lower bound $\Rightarrow$ pseudorandom generator

Our Results

[Bogdanov V.; FOCS '07 special issue]

- 

  Output "looks random" to polynomials, e.g. $x_1 \cdot x_2 + x_3$

- Theorem[Bogdanov V.] Optimal stretch generator
  (I) Unconditionally: for degree 2,3
  (II) Under conjecture: for any degree

- Theorem [Green Tao, Lovett Meshulam Samorodnitsky]:
  Conjecture false

Our latest result
[V.; CCC ‘08]

- Theorem[V.]: Optimal stretch generator for any degree d.

  (Despite the conjecture being false)

- Improves on [Bogdanov V.] and [Lovett]

- Also simpler proof

BPP vs. Poly-time Hierarchy

- Probabilistic Polynomial Time (BPP):

  for every x, Pr [M(x) errs] < 1%

- Recall belief: BPP = P

  Still open: BPP $\subseteq$ NP ?

- Theorem[Sipser Gács, Lautemann '83]: BPP $\subseteq \Sigma_2$ P

- Recall NP = Σ_1 P $\rightarrow$ $\exists$ y M(x,y)

  Σ_2 P $\rightarrow$ $\exists$ y $\forall$ z M(x,y,z)

The Problem We Study

- More precisely [Sipser Gács, Lautemann] give

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

- Question: Is quadratic slow-down necessary?

- Motivation: Lower bounds

  Know $\Sigma_1\mathrm{Time}(n) \neq \mathrm{Time}(n)$ on some models

  [Paul Pippenger Szemeredi Trotter, Fortnow, …]

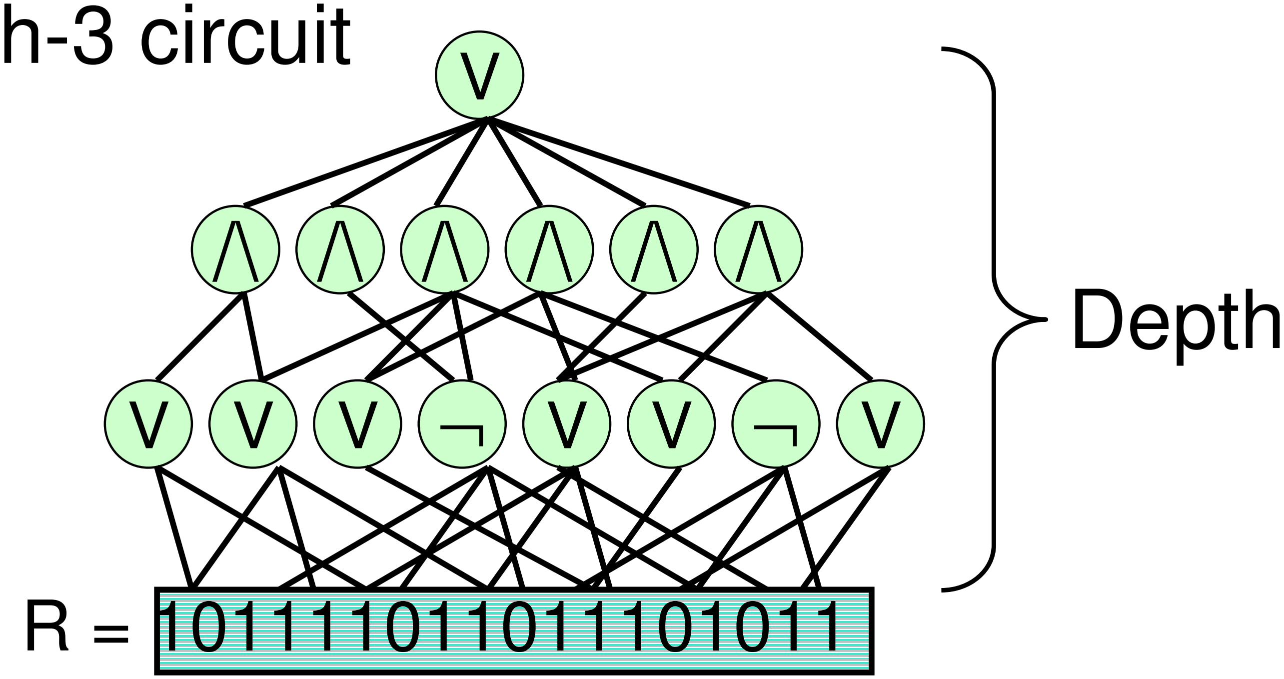
  Technique: speed-up computation with quantifiers

For $\Sigma_1\mathrm{Time}(n) \neq \mathrm{BPTime}(n)$ can't afford $\Sigma_2\mathrm{Time}(t^2)$

Approximate Majority

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

  Do not care if $\Pr_i[R_i = 1] \sim 50\%$ (approximate)
- Model: Depth-3 circuit



The connection

[Furst Saxe Sipser '83]

$M(x;r) \in$ BPTime(t) → $R = 1101\,1\,011101011$

$|R| = 2^t$ → $R_i = M(x;i)$

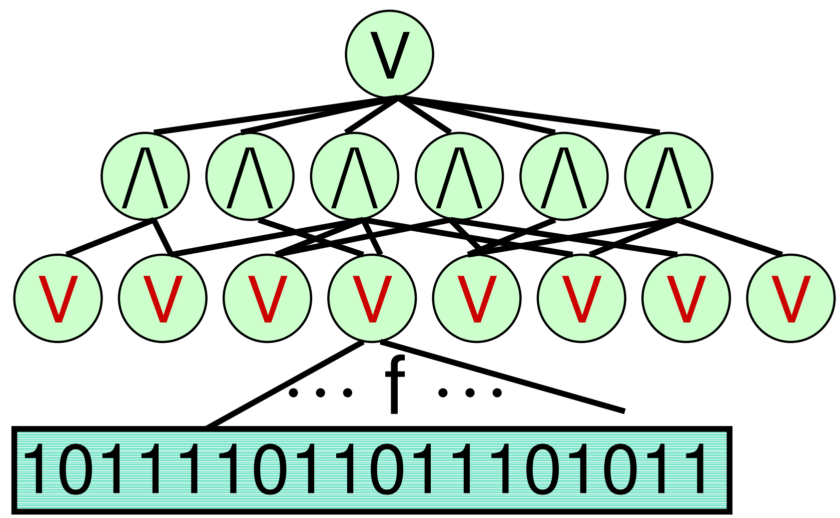
Compute M(x):

Tell $Pr_r[M(x;r) = 1] > 99\%$
from $Pr_r[M(x;r) = 1] < 1\%$

→ Compute Appr-Maj

BPTime(t) $\subseteq \Sigma_2$Time(t')
$= \exists\, \forall$ Time(t')

→



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

→ **Bottom fan-in f = t' / t**

Our Results

[V.; CCC ’07]

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

- Corollary: Quadratic slow-down necessary for black-box techniques:

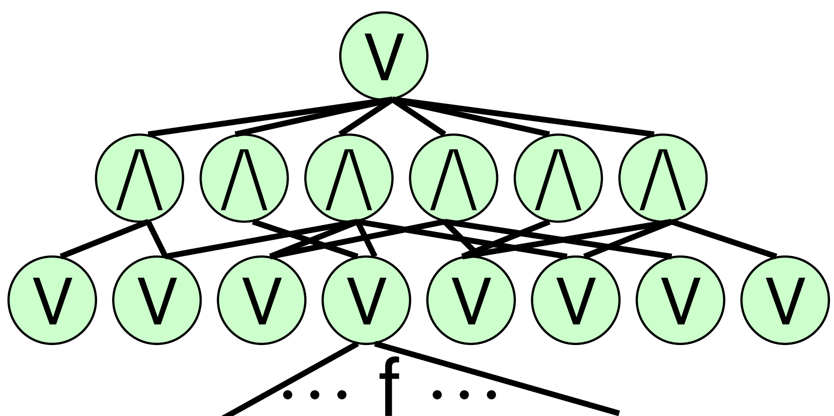
$$\mathrm{BPTime}^{A}(t) \not\subseteq \Sigma_2\mathrm{Time}^{A}(t^{1.99})$$

- Theorem[Diehl van Melkebeek, V]:
  $\mathrm{BPTime}(t) \subseteq \Sigma_3\mathrm{Time}(t \cdot \log^5 t)$

- For time, the level is the third

Our Negative Result

- Theorem[V]: $2^{N^{\varepsilon}}$-size depth-3 circuits for Approximate Majority on N bits have bottom fan-in $f > (\log N)/10$
- Note: $2^{\Omega(N)}$ bound $\Rightarrow$ bound for log-depth circuits [Valiant]
- Recall:



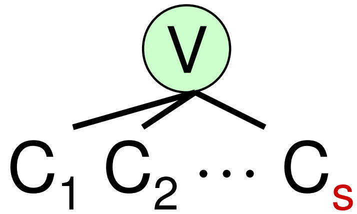
R = 1011110110111101011 |R| = N

tells $R \in$ YES := { R : $\Pr_i [\, R_i = 1] > 99\%$ }

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

Proof

- Circuit: OR of $s=2^{N^{\varepsilon}}$ CNF



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

$\underbrace{(x_1 \vee x_2 \vee \neg x_3)}_{\text{clause size = fan-in}}$

- 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.

  $\Pr_{R \in \text{YES}}[C(x) = 1] \geq 1/s = 1/2^{N^{\varepsilon}}$

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

- Claim: Impossible if C has clause size < (log N)/10

Either $\Pr_{R \in YES}[C(x)=1] < 1/2^{N^{\varepsilon}}$ 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 : $\Pr_{R \in YES}[C(x) = 1] < 1 / 2^{N^{\varepsilon}}$

  Case |S| tiny : Fix few variables and repeat

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

Case |S| BIG

- $|S| \geq N^{\delta} \Rightarrow$ have $N^{\delta} / \log N$ disjoint clauses Γ_i
  - Can find Γ_i greedily

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

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

$$\leq \prod_i (1 - 1/100^{(\log N)/10}) \leq \prod_i \left(1 - 1/N^{1/2}\right)$$

$$= (1 - 1/N^{1/2})^{(N^{\delta}/\log N)} \leq 1/2^{N^{\varepsilon}} \quad ✓$$

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

Case |S| tiny

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

- Note: S covering $\Rightarrow$ clauses shrink

  Example

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

- Repeat

  Consider smallest covering S', etc.

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

Finish up

- Recall: Repeat $\Rightarrow$ shrink clauses
  So repeat at most (log N)/10 times

- When you stop:
  - Either smallest covering size > N^{δ} ✓
  - Or C = 1
    - Fixed $\leq N^{\delta}$ (log N) /10 << N vars.
    - Set rest to 0 $\Rightarrow$ $R \in NO : C(R) = 1$ ✓

Q.e.d.

Conclusion

- Lower bounds: rich area, surprising connections

- Communication complexity, pointer chasing [VW]

⬇

- Circuit complexity, encoding vs. decoding [V,SV]

⬇ ⬆

- Time vs. Randomness

  Constant-depth circuits, polynomials [V,BV,V]

  BPP vs. poly-time hierarchy [V]

  Circuit lower bound for approximate majority