LECTURE 26

- Lovasz Local Lemma
  - constructive version

(3SAT with $K=3$)

K-SAT:

Variables: $x_1 \dots x_n$ boolean $\{0,1\}$

Clauses: $C_1 \dots C_m$

$$C_i = x_{i_1} \vee \bar{x}_{i_2} \vee \bar{x}_{i_3} \vee \bar{x}_{i_4} \dots \vee x_{i_k}$$

Goal: Find $x \in \{0,1\}^n$ satisfying all the clauses

C_i is true / satisfied

$x \vee y \vee z \vee \bar{w}$

"0 0 0 1"

$x=0 \quad y=0 \quad z=0 \quad w=1$

$x_{i_1} \vee x_{i_2} \vee \dots \vee \neg x_{i_k}$

– $\exists$ exactly one / unique violating assignment.

$$\Pr\left[\begin{array}{c}\text{random assignment} \quad k\text{-SAT} \\ \text{satisfies a clause } C_i\end{array}\right] = \frac{2^k - 1}{2^k}$$

$$= 1 - 1/2^k$$

[Lovasz local lemma]

<u>Theorem:</u> $\forall$ k-SAT formula with

(variable-degree $< \boxed{2^{k-20}/k}$

- $\exists$ a satisfying assignment

AND one can efficiently find it.

Variable degree = Maximum # of clauses a variable occurs in.

INPUT: K-SAT ϕ

SOLVE(ϕ)

- Pick random assignment
  $x \in \{0,1\}^n$.

Outer Lopp

- While some clause C is violated
  Fix(C)

Fix(C)

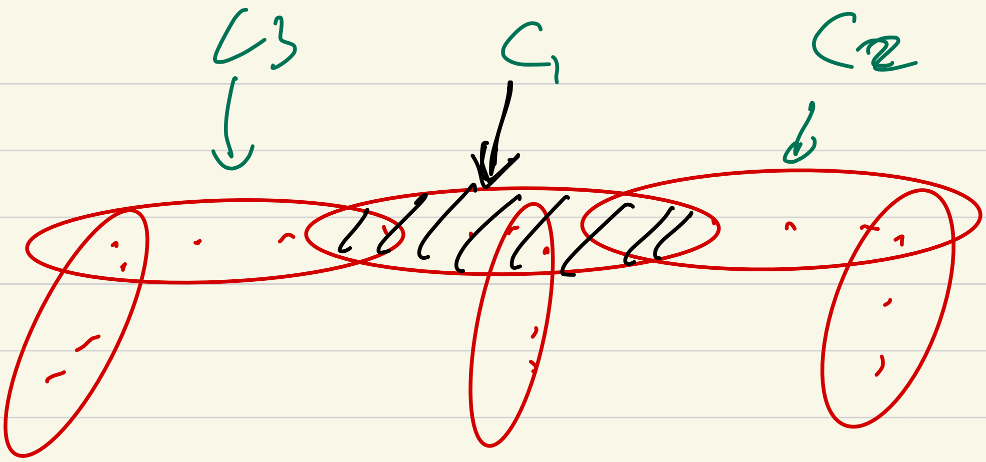
- Assign random values to C.
- While there is some neighboring clause D that is violated

  Fix(D)

  (D might be = C)

- Terminate if $> m^2$ calls.

Terminate??



<u>Claim:</u> # of satisfied clauses increases with each iteration of outer loop.

<u>Proof:</u> Consider some clause D,

D is satisfied before Fix(C) $\Longrightarrow$ D is satisfied after Fix(C) returns.

D $\rightarrow$ satisfied

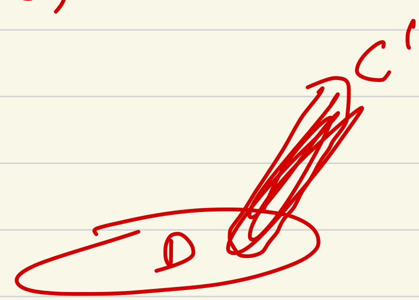
unsatis $\downarrow$ Fix(C)

satis

unsatis

satisfied C

unsatisfied



C will be satisfied at the end of $\mathrm{Fix}(C)$.

$\Rightarrow$ At least one extra clause.

Corollary: Outer loop runs for $< M$ times.

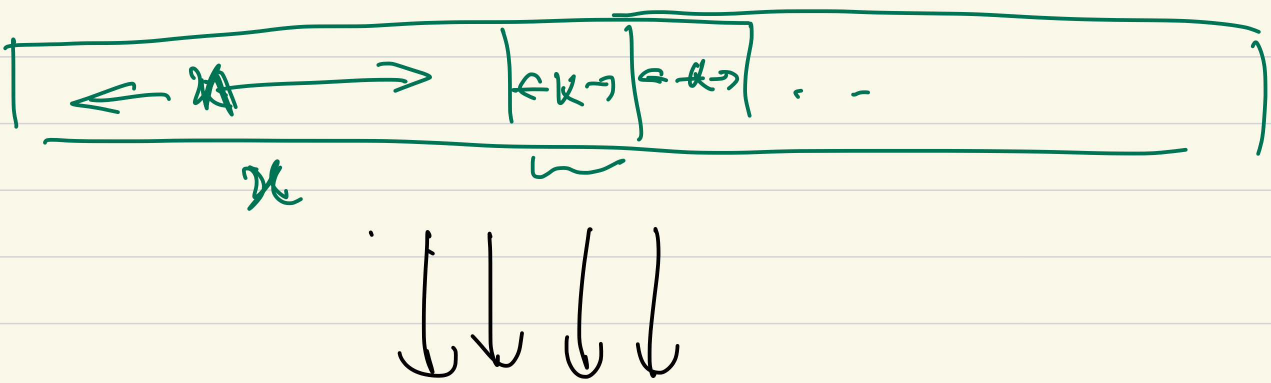
# of clauses.

PROOF

IDEA:

ALG: Execution Φ $\qquad$ 0

- Random bits
  - initial $x \in \{0,1\}^n$
  - $\forall$ clause C, Fix(C)
    - need k bits



DEBUGGER

recieves L bits

but can reconstruct $\gg L$ bits

of algorithm's random bits.

ALG: Execution Φ

- Random bits
  - initial $x \in \{0,1\}^n$
  - $\forall$ clause C, Fix(C)
    - need k bits

(1) $m \log m + s \log R$

(2) $2s$

(3) n

$m \log m + s \log R + 2s + n$

$n + sk$

SEND:

1) $\forall$ clause C, Fix(C)
   send name of "C"

2) $\forall$ return / termination of fix(C) send "return" $\rightarrow$ 2 bits ($O(1)$) per Fix

3) After s calls to Fix send current assignment $\rightarrow$ n bits

$\leq m \log m + s \log R$

#TOPLEVEL CALLS $\cdot (\log m)$ + #recursive calls $(\log R)$

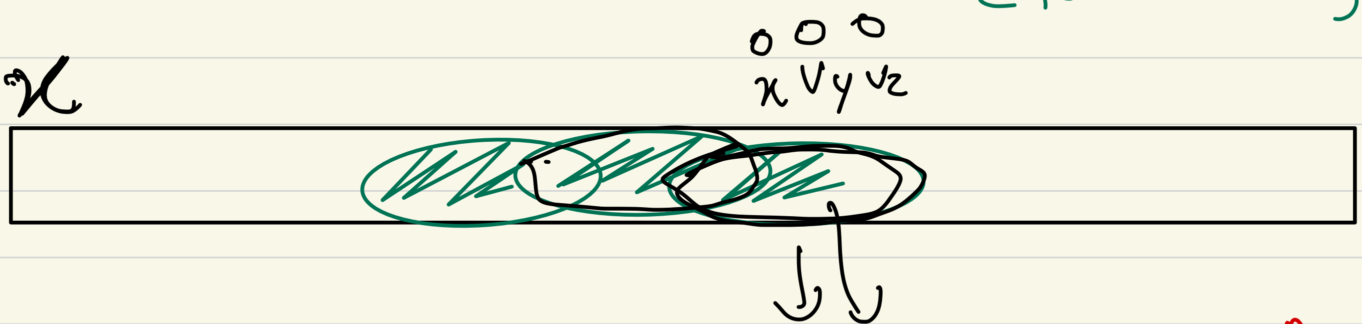
DEBUGGER

Claim: Given the clauses on which Fix is called & final assignment $y \in \{0,1\}^n$

Debugger can reconstruct

→ $x \in \{0,1\}^n$

→ all the intermediate clause assignments.



C_l $(1\,0\,1)$

$y \in \{0,1\}^n$

C_1

C_2

C_3

⋮

C_{10}

C_1 ← all random bits of Alg

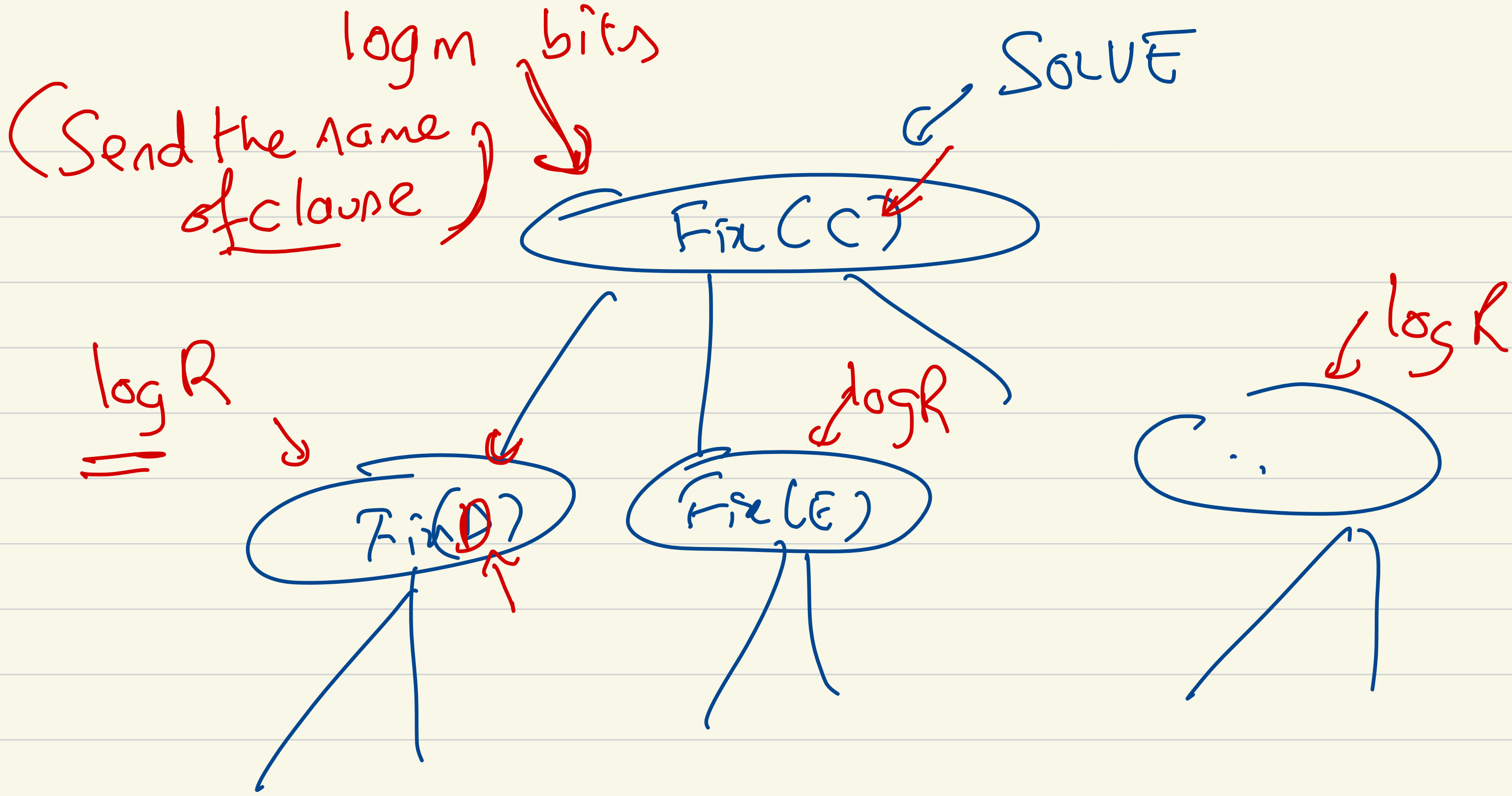
$\boxed{C_l}$

$n + sK$

Debugger can recover

1) initial $x \in \{0,1\}^n$ (n bits)

2) all clause assignments K



Suppose each clause has R neighbours

$\rightarrow$ log R bits

Alg Sends

$$\frac{\boxed{m\log m + s\log R + 2s + n}}{n + sK}$$

Debugger reconstructs

If $K > (\log R + 2)$ then

for $s > m^2 + n^2$, a contradiction

R = max degree of a clause

$\leq \begin{pmatrix}\text{max degree} \\ \text{of variable}\end{pmatrix} \cdot K$

$\leq \left(2^{K-20}/K\right) \cdot K \approx 2^{K-20}$

LOVASZ LOCAL LEMMA

- Bad Events: $\mathcal{E}_1 \ldots \mathcal{E}_m$

over a probability space

[- Pick a random $x \in \{0,1\}^n$
$\mathcal{E}_i \to$ clause C_i violated]

- $\Pr[\text{Bad Event } \mathcal{E}_i] \leq p \quad \forall i = 1 \ldots m$

$$\left[\Pr[x \text{ violates } C_i] \doteq 1/2^k \right]$$

GOAL:

$$\Pr\left[\bigcap_{i=1}^{m} \overline{\mathcal{E}_i}\right] = \Pr[\text{no Bad Event}] \geq ??$$

[UNION BOUND]

m bad events $\mathcal{E}_1, \ldots \mathcal{E}_m$

$$\Pr\left[\bigcup_{i=1}^{m} \mathcal{E}_i\right] \leq \sum_{i=1}^{m} \Pr[\mathcal{E}_i]$$

$$\leq p \cdot m$$

$$\Rightarrow \Pr\left[\bigcap \overline{\mathcal{E}_i}\right] \geq 1 - pm$$

[INDEPENDENT EVENTS]

$$\Pr\left[\bigcap_{i=1}^{m} \overline{\mathcal{E}_i}\right] = \prod_{i=1}^{m} \Pr(\overline{\mathcal{E}_i}]$$

$$\geq (1-p)^m > 0$$

Let $\Gamma(i)$ = neighborhood of event $\mathcal{E}_i$

in that

$\mathcal{E}_i$ is independent of $\overline{\Gamma(i)}$.

$$\Pr[\mathcal{E}_i] = \Pr\left[\mathcal{E}_i \,\middle|\, \bigcap_{\substack{j \in A \\ A \subseteq \overline{\Gamma(i)}}} \mathcal{E}_j\right]$$

Thm: If $\Pr(\mathcal{E}_i) \leq p \quad \forall i$

and $|\Gamma(i)| \leq d \quad \forall i$

AND $p(d+1) \cdot e < 1$ (2.718)

then $\Pr\left[\bigcap \overline{\mathcal{E}}_i\right] \geq \left(\frac{d}{d+1}\right)^n >$