DEPTH$(d(n))$ = langs comp. w/ depth $d(n)$ ckts

Recall NC^1 = DEPTH$(\log n)$

Since $\underline{C(f) \leq 2^{D(f)}}$,

NC^1 = poly-size, $O(\log n)$-depth circuits

= poly-size formulas!

__Def.__ $f: \{0,1\}^* \to \{0,1\}$ is __super-log-depth__

if $\exists$ __unbdd__ $s: \mathbb{N} \to \mathbb{N}$ s.t. $\forall^\infty n$, $D(f_n) \geq s(n) \cdot \log C(f_n)$

(i.e. $C(f_n) \leq 2^{D(f_n)/s(n)}$)

Call f "super-log-depth" if this is true.

__Prop.__ $\exists$ super-log-depth $f \Rightarrow P \not\subseteq NC^1$

__Pf:__ $NC^1 = P \Rightarrow$ CircEval $\in NC^1$

Let $f_n \in B_n$. Let E be a ckt for CircEval on inputs of size $O(C(f_n) \log C(f_n))$ w/ depth$(E) \leq O(\log C(f_n))$.

Let C compute f_n & min.

Define $E'(\underline{x}) := E(C, \underline{x})$

hardcoded.

$E' \equiv f_n \Rightarrow D(f_n) \leq O(\log C(f_n))$ $\forall n$. $\square$

CircEval is "hardest" for depth minimization:

lower depth of ckts for CircEval

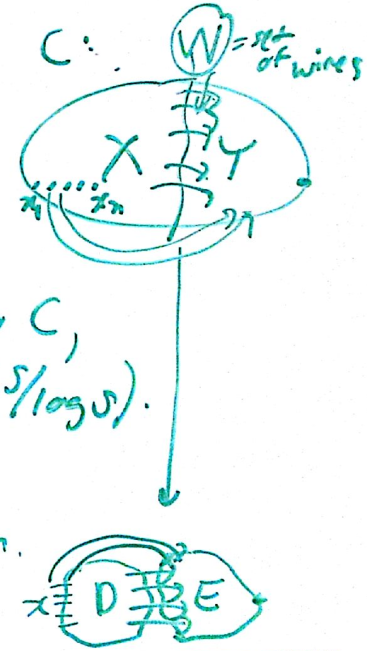
$\Rightarrow$ lower depths of "all" ckts.

Proving $D(f) \gg \log C(f)$ looks hard... more later.

Depth upper bound:

Thm [Paterson-Valiant '76]

$$D(f) \leq O(C(f)/\log C(f))$$



Pf Outline: Let C be a min ckt of size s.

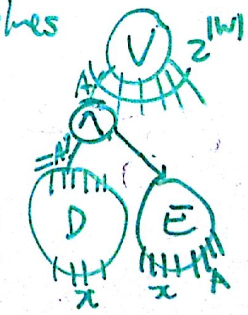
1. Find partition of gates of C into "balanced" subckts X + Y

2. Construct two ckts C_1, C_2 equiv to C,
   Claim: min depth of $C_1, C_2 \leq O(s/\log s)$.

2.1 C_1: take min ckt for X
min ckt for Y, compare them.

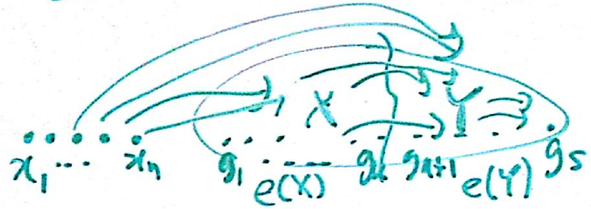
Good if $|W|$ is large

2.2 C_2: take OR of all $2^{|W|}$ possible values on wires W on a given input.
For each value assignment A,
take AND of two ckts:
- verify that X outputs A
- verify Y on A outputs 1.



Good when $|W|$ is small.

Pf. Ind on # of wires in $C = O(s)$



Find cut (X, Y) of s gates s.t. $e(X) \approx e(Y)$.

Define
$\mu(X,Y) = 2 \cdot e(X) + m - e(C)$

$e(X)$: total # wires in subckt X (including from inputs)

m: # wires of X that are inputs to Y

$e(C)$: # wires in C

Claim: $\exists$ cut (X, Y) s.t. $|\mu(X,Y)| \leq 2$

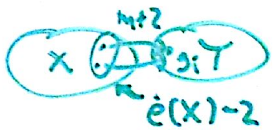
Pf. Start with $X = \{g_1, \dots, g_s\}$, $Y = \emptyset \Rightarrow \mu(X,Y) = e(C)$
($e(X) = e(C)$, $m = 0$)

$X = \{g_1, \dots, g_i\}$ → Move gate g_i from X to Y:
$Y = \{g_{i+1}, \dots, g_s\}$
— decreases m by ≤ 1
— decreases $e(X)$ by ≤ 2.
(lose the two input wires to g)

(if m increases, it's by 2, and e(X) decreases by 2.)

$\Rightarrow \mu(X,Y)$ decreases by ≤ 5

[does not increase! if m increases, it's by 2 but e(X) decreases by 2.]



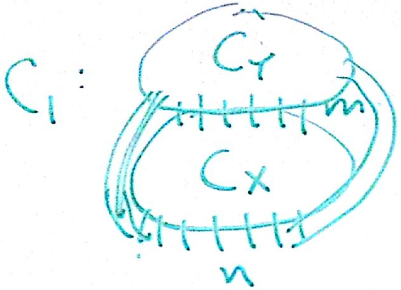
For $X = \emptyset$, $Y = \{g_1, \dots, g_s\}$,
$\mu(X,Y) = -e(C)$
($m = 0$, $e(X) = 0$). $\Rightarrow \exists\, i$ s.t. $\mu(X,Y) \in \{-2,-1,0,1,2\}$. □

Idea: $e(X) + e(Y) = e(C)$
$e(X) \approx \frac{e(C) - m}{2}$

$e(Y) \approx \frac{e(C) + m}{2}$

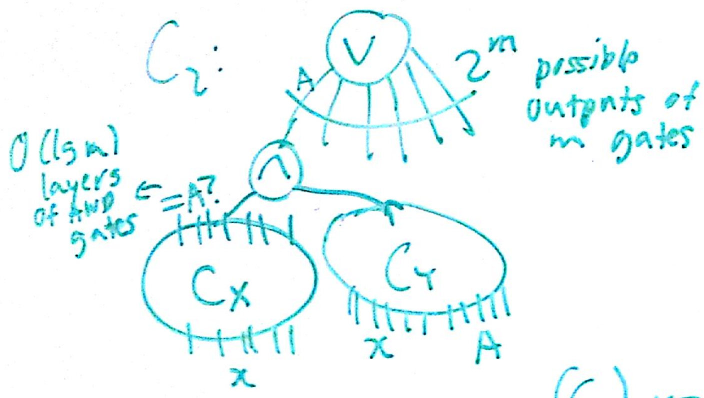
Let C_X = min-depth ckt for X w/ n inputs, m outputs ← (could imagine modifying C_X's)
C_Y = min " " " Y w/ n+m inputs, 1 output

C_1:



$d(C_1) = d(C_X) + d(C_Y)$

C_2:



$d(C_2) = m + 1 + \max\{d(C_X) + O(\log m), d(C_Y)\}$

$C' :=$ min depth ckt of C_1, C_2

$x = e(C_X), y = e(C_Y) < \overset{=z}{e(C)}$, so by induction

$d(C_X) \le k\, x/\log x + k\, y/\log y$

$d(C_Y) \le m + 1 + \max\{k\,x/\log x + O(\log m), k\,y/\log y\}$

<u>By claim,</u>

$x = \frac{z-m}{2} \pm c$

$y = \frac{z+m}{2} \pm c$

<u>WTS:</u> $d(C') \le kz/\log z$

<u>Cases</u> ① $m \le \varepsilon \cdot z/\log z$.

Then $d(C_2) \le \varepsilon \cdot z/\log z + 1 + \max\{k\,x/\log x + O(\log m), k\,y/\log y\}$.

Set $\varepsilon > 0$ suff. small)

$\le kz/\log z$

② $m > \varepsilon \cdot z/\log z$

[Removed $\frac{z}{\log z}$ wires from X, made $\frac{z}{\log z}$ new ckts, outputs are inputs to Y]

$x + y \le \left(1 - \frac{\varepsilon}{\log z}\right) z \Rightarrow d(C_1) = k\left(x/\log x + y/\log y\right)$

↑ # wires in X   ↑ # wires in Y

$x \le \frac{z-m}{2} \le c$ $\ldots \le k\, z/\log z.$

$\Rightarrow x \le \frac{z}{2}\left(1 - \varepsilon/\log z\right)$

Cor: Every $O(n)$-size ckt has an equivalent $2^{O(n/\lg n)}$-size formula.

Pf. $O(n)$-size ckt $\mapsto O(n/\lg n)$-depth ckt

$\mapsto 2^{O(n/\lg n)}$-size formula. □

Thm [HPV] $\mathrm{TIME}(t) \subseteq \mathrm{SPACE}(t/\log t)$

$\mathrm{TIME}(t) \approx \mathrm{SIZE}(t)$, $\mathrm{SPACE}(t/\log t) \Leftarrow \mathrm{DEPTH}(t/\lg t)$

Idea: "Graph" of a computation of time t:

Nodes = time steps
edges = data dependencies between time steps.

Pick cut $(X, Y) = (\{s_1, \ldots, s_{t/2}\}, \{s_{t/2+1}, \ldots, s_t\})$

Let m = # nodes in X with output to Y.

Cases 1. m is small:

- try all possible $2^{O(m)}$ guesses for data being read / state at these steps of X

$O(m)$ space.

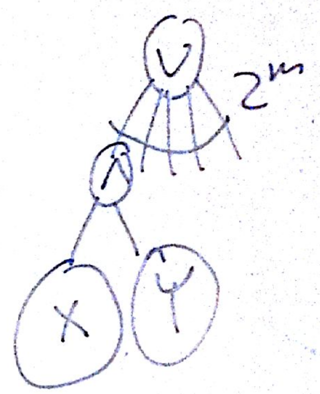
Recurse on X, then on Y:

- Verify each data guess is correct

Sim. X from start, separately on each

Erase space used for verification.

- Sim. starting from Y, assuming guesses correct.



(Run Y using $O(m)$ bits of guesses)

2. m is large.

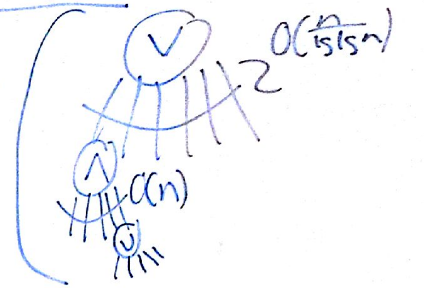
Sim. starting from Y. When Y needs to know data from s_i of X, sim. X from start to get s_i.

(Erase work when done with call)

Get a similar recurrence.

Depth reduction.

Thm [Valiant '77] Every ckt of $O(n)$ size, $O(\log n)$ depth is equiv. to a depth-three ckt of $2^{O(n/\lg\lg n)}$ size.



Key: Depth Reduction Lemma.

Let G be a DAG m edges, s.t.
depth = longest path = d
For all k, can find $\frac{km}{\log_2 d}$ edges

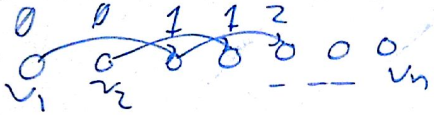
s.t. removal from $G \mapsto G'$ w/ depth $\le d/2^k$.

E.g. $O(n)$ edges $\Rightarrow$ can delete $O(n/\log n)$ edges
$n/2$ depth & depth $< n/100$.

"Hit" long paths & split them.

Pf. Let $d = 2^e - 1$ for simplicity.

$\ell(v_i)$ $\triangleq$ longest path ending in v_i



$\ell(v_i) \in \{0,1\}^e$.

(note: $\forall (u,v) \in E$ $\ell(u) \ne \ell(v)$)

Partition edges: $E_i = \{(u,v) \mid$ first bit in which $\ell(u) \ne \ell(v)$ differ $= i^{th}$ bit$\}$

i.e., $\ell(u)[i] \ne \ell(v)[i]$
$\forall j < i$, $\ell(u)[j] = \ell(v)[j]$

E.g. $\ell(u) = 0\cdots0$
$\ell(v) = 010\cdots0$
$\Rightarrow (u,v) \in E_2$.

Claim: $E_1, \ldots, E_e$ is a partition of E.

Remove the k smallest E_i's from G. $E_{i_1}, \ldots, E_{i_k}$.

Their union has $\leq \frac{km}{\log m}$ edges.

! $\to$ Can rewrite the labels $\ell(v)$ using only $e-k$ bits!

Remove bits at positions $i_1, \ldots, i_k$ from all labels $\mapsto \ell'$.

Still have: $\forall (u,v) \in E \Rightarrow \ell'(u) < \ell'(v)$!

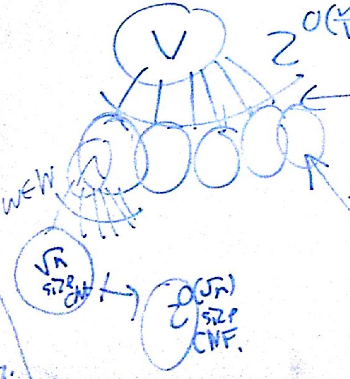
(Removed all edges in which those bits are first flipped!)

$\Rightarrow$ Longest path is now $\leq 2^{d-k} = d/2^k$. $\square$

Pf: C w/ n inputs, size $\leq cn$, depth $\leq c \log n$.

Red. to depth three

Find $\frac{ckn}{\log(c \log n)} = O\left(\frac{n}{\log \log n}\right)$ wires W in C s.t. removal $\Rightarrow C'$ of depth $\leq \frac{c \log n}{2^k}$.



$2^{O\left(\frac{n}{\log \log n}\right)} = 2^{|W|}$

$\leftarrow$ Guess all possible values that could be on these wires W.

AND: verify these values for each wire in W.

depth $\leq \frac{c \log n}{2^k} \Rightarrow$ size $\leq 2^{\frac{c \log n}{2^k}} = n^{c/2^k}$.

Can represent w/ CNF of $2^{O(n^{c/2^k})}$ size.

$k = \log(c) + 1 \Rightarrow$ CNF of size $2^{O(\sqrt{n})}$.

Cor: f isn't in $2^{n/1000}$ size depth-3 ckts

$\Rightarrow f$ isn't in $O(n)$ size $O(\log n)$ depth.