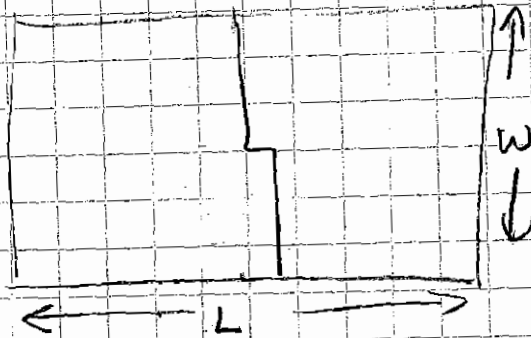


6.896
4/26/04
L19.1VLSI lower bounds

Lemma A network with bisection width B has area $\Omega(B^2)$.

Pf.



$$B \leq W + 1 \Rightarrow A = LW \geq W^2 \geq (B-1)^2 = \Omega(B^2) \quad \square$$

Good recursive bisection \Rightarrow small area.

Bisection width lower bounds on computationShifting:

Input: x_0, x_1, \dots, x_{n-1} data

$S \in \{0, \dots, n-1\}$ control

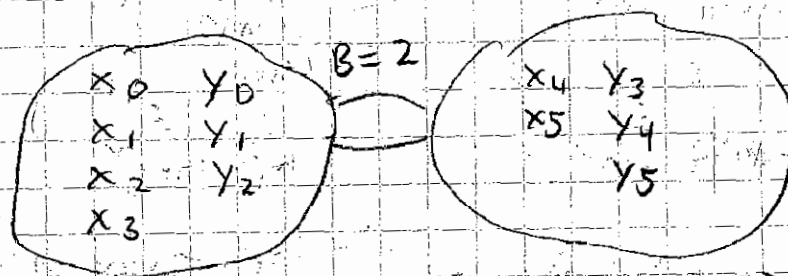
Output: y_0, y_1, \dots, y_{n-1} & $y_i = x_{(i-S) \bmod n}$

Network	$T(n)$	$B(n)$	$A(n)$
Linear array	$\Theta(n)$	$\Theta(1)$	$\Theta(n)$
Tree	$\Theta(n)$	$\Theta(1)$	$\Theta(n)$
$\sqrt{n} \times \sqrt{n}$ mesh	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$	$\Theta(n)$
n -input butterfly	$\Theta(\lg n)$	$\Theta(n)$	$\Theta(n^2)$

Let $B(n) = \#$ edges cut to bisect outputs

If $I(n) =$ worst-case #bits to cross this bisection, then $B(n)T(n) \geq I(n)$.

Ex.



$S=0 \Rightarrow$ 1 bit crosses bisection (L to R)

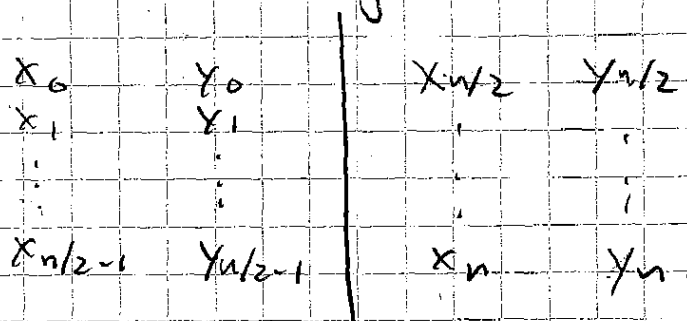
$S=3 \Rightarrow$ 3 bits cross - worst case $\Rightarrow I=3, T \geq B/I = 3/2$.

6.896
4/26/04
L19.2

How do we know 3 bits must cross?
Might there be a clever encoding?

"Fooling argument"

Ex. Sup. we have following bisection:



$s = n/2 \Rightarrow$ intuitively $n/2$ bits must cross (L to R)

Claim: $B(n) \cdot T(n) \geq n/2$.

Pf. (fooling arg.) Sup. $B(n) \cdot T(n) < n/2$

communication patterns on $B(n)$ wires (L to R)
over time $T(n) = 2^{B(n)T(n)} < 2^{n/2}$

values for $x_0, \dots, x_{n/2-1} = 2^{n/2}$

$\therefore \exists$ 2 distinct $x'_0, \dots, x'_{n/2-1}$ and $x''_0, \dots, x''_{n/2-1}$
that produce identical comm patterns.

RHS of circuit can't distinguish \Rightarrow produces
same values for $y_{n/2}, \dots, y_n$ for both \Rightarrow must
operate wrong for one. Contradiction. \square

6.896

4/26/04

L19.3

Thm. For any bisection of outputs, $B(n)T(n) \geq n/2$

Pf. Consider arb. bisection

<u>Ex.</u>	y_0	x_0	y_1	x_3
	y_2	x_1	y_3	
	y_4	x_2	y_5	
		x_4		
		x_5		

Make an $n \times n$ table:

	x_0	x_1	x_2	x_3	x_4	x_5
0		X				X
1	X		X	X	X	
5	2		X			X
3	X		X	X	X	
4		X				X
5	X		X	X	X	

X if shift of s causes x_i to cross bisection

Every column contains $n/2$ X's.

\therefore Average # X's per row = $n/2$.

\Rightarrow some row contains $\geq n/2$ X's.

(some shift causes $n/2$ bits to cross)

Network	$B(n)$	$T(n)$
Linear array	$\Theta(1)$	$\Omega(n)$
Tree	$\Theta(1)$	$\Omega(n)$
$\sqrt{n} \times \sqrt{n}$ mesh	$\Theta(\sqrt{n})$	$\Omega(\sqrt{n})$
n -input butterfly	$\Theta(n)$	$\Omega(1)$

6.896

4/26/04

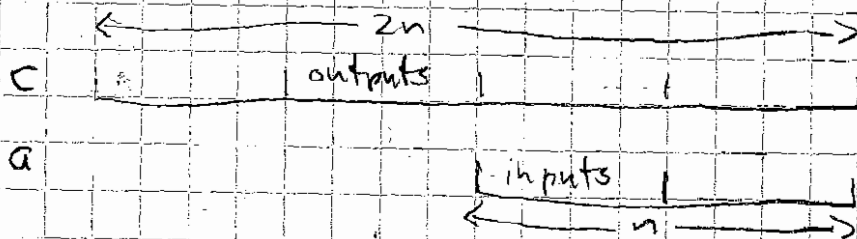
L19.4

Theorem Any circuit for shifting n bits has
 $AT^2 = \Omega(n^2)$

PF. $AT^2 = \Omega(B^2 T^2)$
 $= \Omega(n^2)$ \square

Theorem Any circuit for multiplying two n -bit #'s
 has $AT^2 = \Omega(n^2)$.

PF. $c = ab$. Let b take on powers of 2.
 Essentially corresp. to shift problem.



Consider bisection of outputs.
 Build shift/input matrix as before: $n \times n/2$

	$a_{n/2}$	$a_{n/2+1}$	\dots	a_n
0		x	x	
1	x			x
⋮				
s			x	
⋮		x		x
n-1	x			

$n/4$ x's per column

$n^2/8$ x's in matrix

\Rightarrow some row has $n/8$ x's.

$\therefore BT \geq \Omega(n)$

\square

Also, FFT, convolution, sorting, routing, etc.

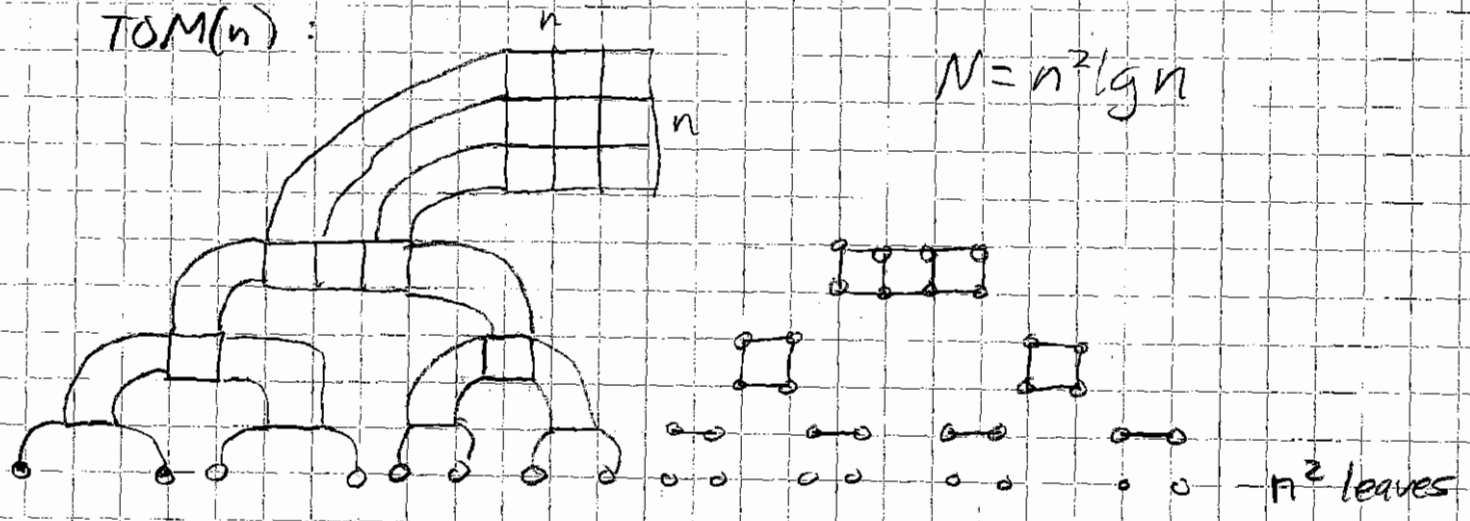
Exam: In class on May 12. Prob. session May 10.

6.896
4/28/04
L20.1

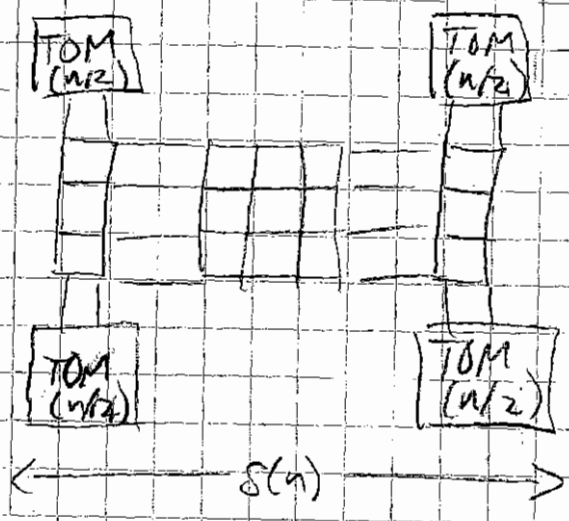
General Layout Strategy

Tree of meshes (not mesh of trees)

TOM(n):



Area:



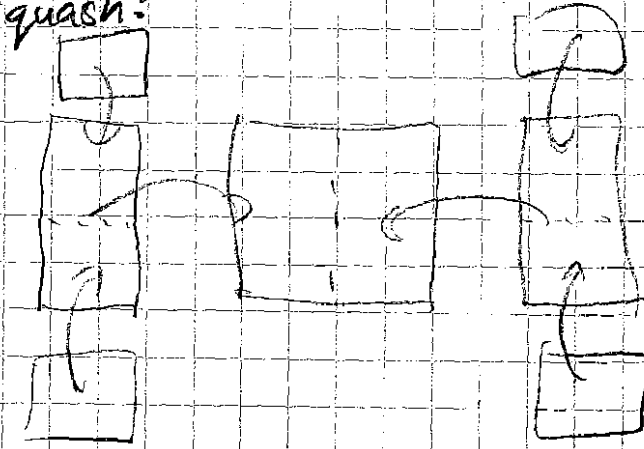
$$S(n) = 2S(n/2) + n$$

$$= \Theta(n \lg n)$$

$$A(n) = \Theta(n^2 \lg^2 n)$$

6.896
4/28/04
L20.2

Fold and squash:



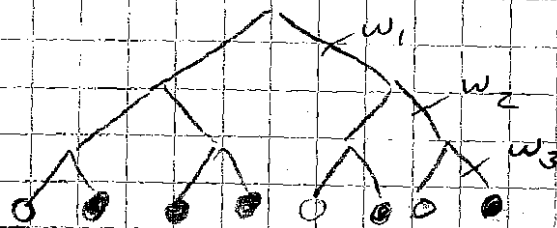
$n^2 \times \lg n$ layers $\xrightarrow{\text{squash}}$ $\Theta(n^2 \lg^2 n)$ area.

Truncated TOM: $\text{TOM}(n, k)$ - top k levels.
Area = $\Theta(n^2 k^2)$

Decomposition trees

T is a (w_1, w_2, \dots, w_r) decomposition tree for $G = (V, E)$:

1. Vertices in V mapped to leaves of T .
2. Edges in E run through links of T .
3. #edges leaving subtree rooted at depth i is $\leq w_i$



For $1 < \alpha \leq 2$, G has a (w, α) decomp tree if it has a $(w, w/\alpha, w/\alpha^2, \dots, O(1))$ decomp tree.

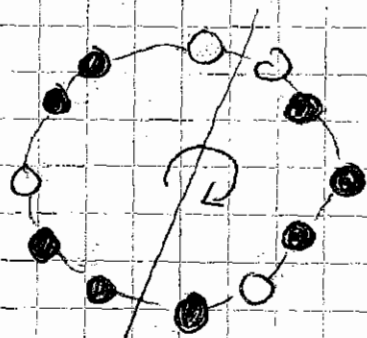
A decomp tree is balanced if all subgraphs at the same depth have same # vertices to within 1.

6.896
4/28/04
L20.3Layout strategy

1. Start with $(w, \sqrt{2})$ decomp tree.
2. Balance the decomp tree
3. Embed the balanced tree in trunc TOM
4. Use trunc TOM layout to yield $O(w^2 \lg^2 n)$ area layout.

Balancing decomp trees

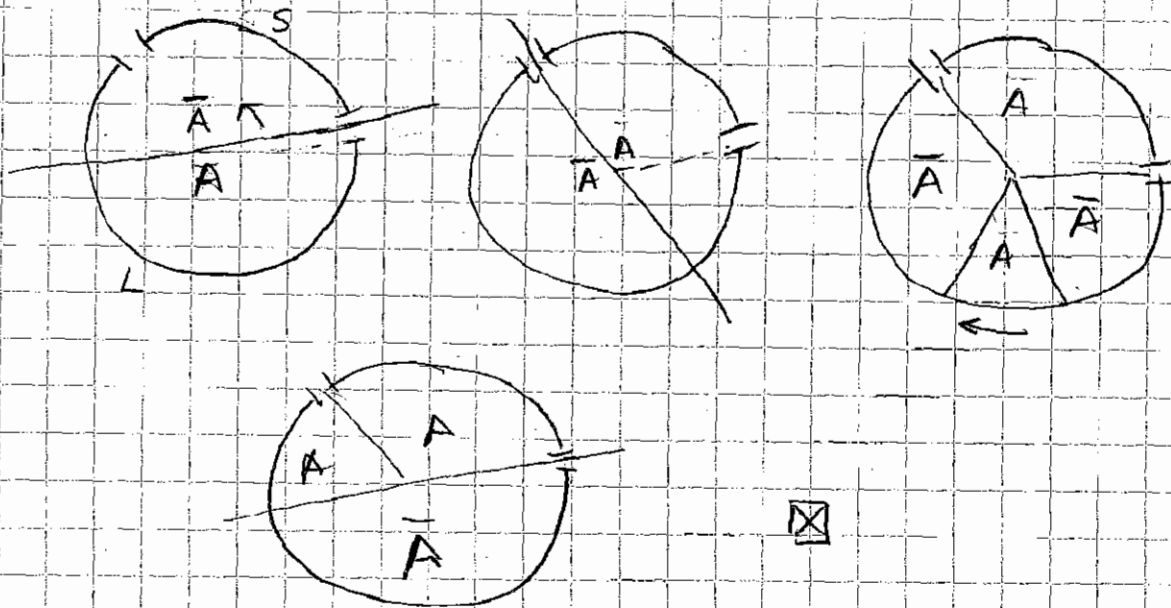
Warm-up: Necklace with black and white pearls.
How many cuts to divide into 2 sets, each with half the pearls of each color?



2 cuts suffice.
Continuity argument.

Lemma. Consider any 2 strings composed of an even # of black pearls and an even # of white pearls. By making at most 2 cuts, the pearls can be partitioned into 2 sets, each containing 2 strings, such that each set has $1/2$ the pearls of each color.

Pf. (Continuity arg.)

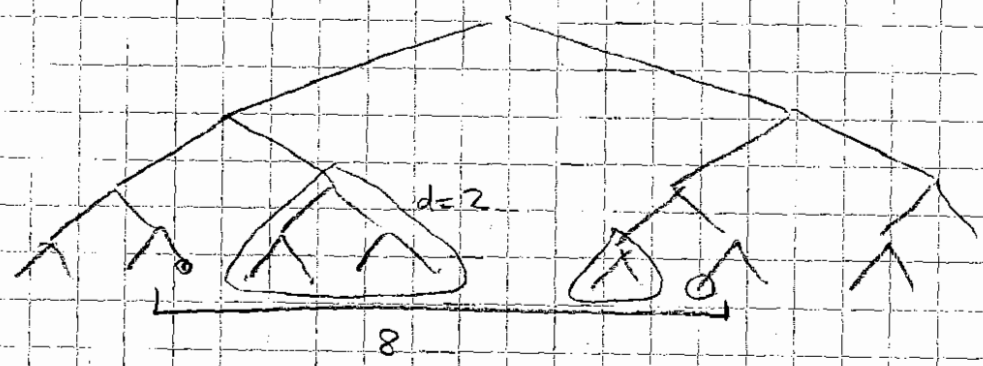


□

6.896
4/28/04
L20.4

Lemma. Let T be cbt drawn with n leaves on a straight line, and consider any set S of k consecutive leaves of T . Then, \exists a forest F of complete binary subtrees of T $\$$

1. $S = \{\text{leaves of } F\}$
2. at most 2 trees of F have any given height.
3. depth of largest tree in F is $\leq \lg k$

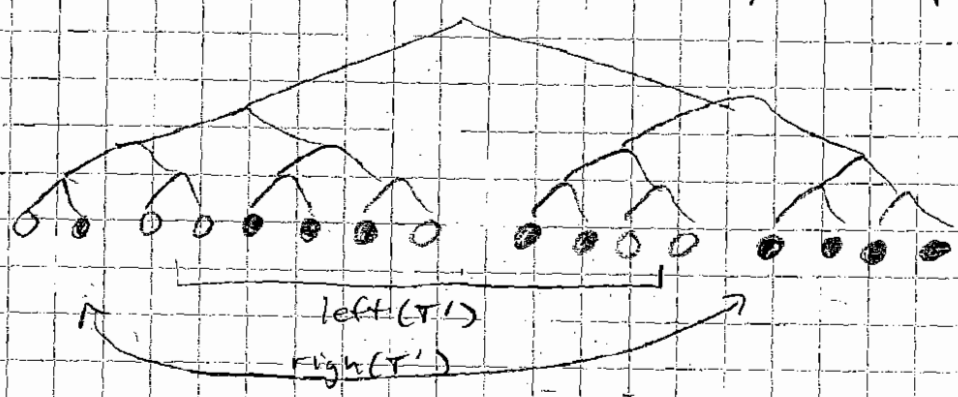


Pf. F be forest of maximal cbt's whose leaves lie only in S . (1) & (3) follow. Use induction to prove (2). \square

Thm. Let G be a graph on n vertices that has a $\langle w_1, w_2, \dots, w_r \rangle$ decomp tree T . Then, G has a $\langle w'_1, w'_2, \dots, w'_{\lceil \lg n \rceil} \rangle$ balanced decomp tree T' , where

$$w'_i = 4 \sum_{k=i}^r w_k$$

Pf. Color leaves of T : 1 = node of G , 0 = empty.



Recursively split B & W leaves evenly. Each stage has ≤ 2 strings of consec. leaves from T , each of which has ≤ 2 cbt's of a given height.

6.896
4/28/04
L20.5

Total # wires leaving a string
 \leq sum of wires leaving each of its cbl's.

$$w_i' \leq 4 \sum_{k \in i} w_k \quad \square$$

Corollary A graph with a (w, α) decomp tree, α const,
 has an $(O(w), \alpha)$ balanced decomp tree.

Pf. Sum is geometric:

$$w_i' = 4 \sum_{k \in i} w_k$$

$$\leq 4 \sum_{k \in i} \frac{w}{\alpha^{k-1}}$$

$$\leq \frac{4w}{\alpha^{i-1}} \left(\frac{\alpha}{\alpha-1} \right)$$

Graph has $(4w\alpha/(\alpha-1), \alpha)$ decomp tree. \square

Next week: Embed in trunc TOM \Rightarrow layout.
 Area-universal networks.

<<Exam issues>>