# CS184a: Computer Architecture (Structures and Organization)

Day14: November 10, 2000

Switching



## Previously

- Role and Requirements for Interconnect
- Understood interconnect structure in terms of recursive bisection
  - *e.g.* Rent's Rule, Hierarchical Interconnect
- Using all necessary wires optimally
  - $O(n^{2p})$ growth
- Raised the question of mesh channel growth
  - w grow as n?

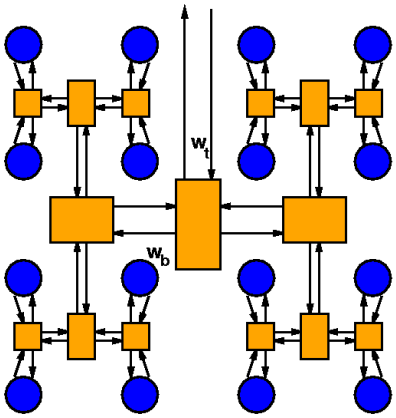
# Today

- Switching Requirements
  - use wires
  - reduce switching costs
  - allow routing
- Mesh Interconnect
- Flavor of Switch Timing



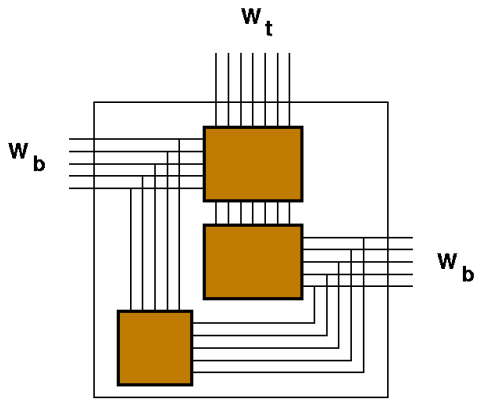
# Hierarchical

- Previously, focussed on wires
- What do switch boxes need to look like to use the wires?



# Straight-forward Case

- Build Crossbars




- Switches:
  - $w_t \% w_b$
  - $w_t \% w_b$
  - $w_b \% w_b$
  - Total: $2(w_t \% w_b) + w_b \% w_b$

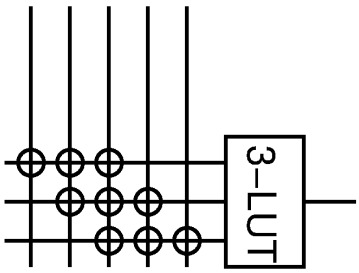


# Can we do better?

- Crossbar too powerful?
  - Does the specific down channel matter?
- What do we want to do?
  - Connect to *any* channel on lower level
  - Choose a subset of wires from upper level
    - order not important

# N choose K

- Exploit freedom to depopulate switchbox
- Can do with:
  - K(N-K+1) swtiches






# Crossover?

- Specific channel not matter on crossover, either
- But tricky
- Need to guarantee:
  - any subset free on left can be connected to free subset on right
  - can be done in $w_b^2/2$
  - for large $w_l/w_b$, can be done with existing connections

# Switching Costs

- How many switches total?
  - What is the switch growth with N?
- How much delay?
  - How does switch delay grow with N?



# Switch Delay

- Switch Delay: $2 \log_2(N_{tree})$
  - $N_{tree}$ = smallest subtree containing source and sink
  - Worst Case: $N_{tree} = N$

# Switch Area

- $w_l = 2^p w_b$
- $Nsb(l) = (2 \times 2^p + 1) w_b^2$
- $N(l) = N/2^l$
- $w_b(l) = c(2^l)^p$
- Total $= \Sigma N(l) * Nsb(l)$
- Total $\mu \Sigma (N/2^l)((2^l)^p)^2$
- Total $\mu N^{2p} [\Sigma (1 + 2/2^{2p} + \ldots)]$
- Total $\mu N^{2p}$



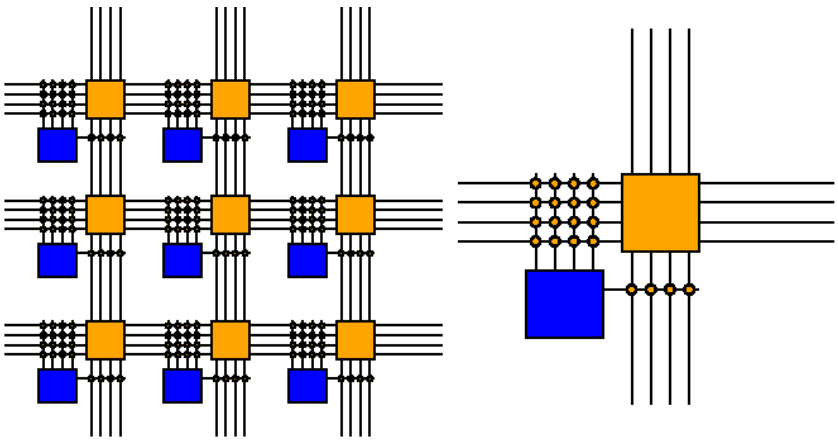
# Routing

- Trivial and guaranteed
  - assuming don't exceed channel capacities
  - according to the way we just designed the switch boxes
- Start at root switch box:
  - route subset to each side (k of m guarantee)
  - start crossover routes here
    - (space on sides and subset connect guaranteed)
  - recurse on left and right subtrees
- Essentially linear in number of switches

# Mesh

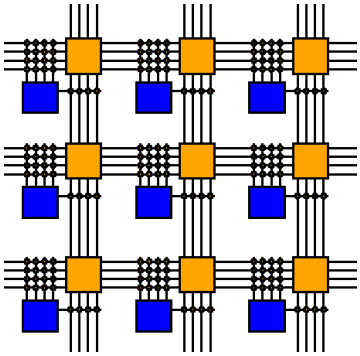


# Mesh



# Mesh Channels

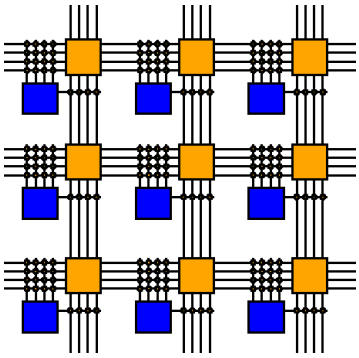
- Lower Bound on w?
- Bisection Bandwidth
  - goes as $cN^p$
  - $\sqrt{N}$ channels in bisection
  - $w \geq cN^p/\sqrt{N} = cN^{p-0.5}$





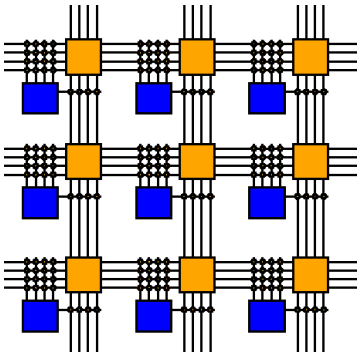
# Straight-forward Switching Requirements

- Total Switches?
- Switching Delay?



# Switch Delay

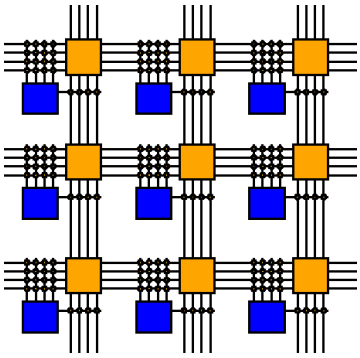
- Switching Delay: 2 Ö($N_{subarray}$)
  - worst case: $N_{subarray}$ = N





# Total Switches

- Switches per switchbox:
  - 4 3w´ w = $12w^2$
- Switches into network:
  - (K+1) w
- Switches per PE:
  - $12w^2$ +(K+1) w
  - $w^3 = cN^{p-0.5}$
  - Total $\mu N^{2p-1}$
- Total Switches: N*Sw/PE $\mu N^{2p}$



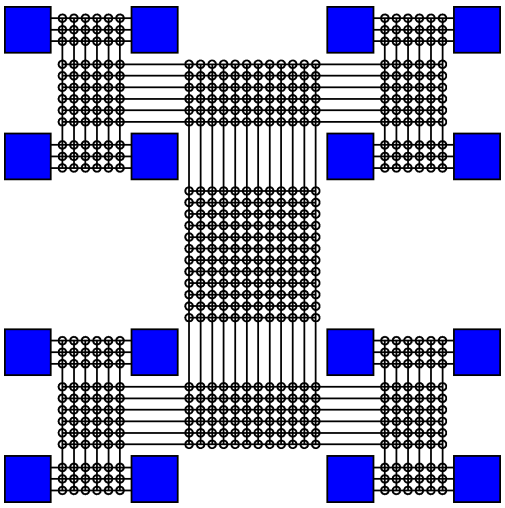
# Routability?

- Asking if you can route in a given channel width is:
  - NP-complete



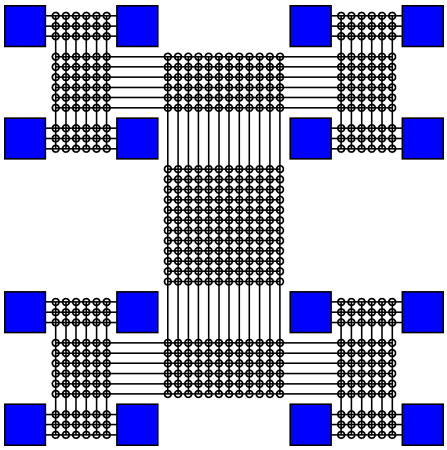
# Meshes and Trees

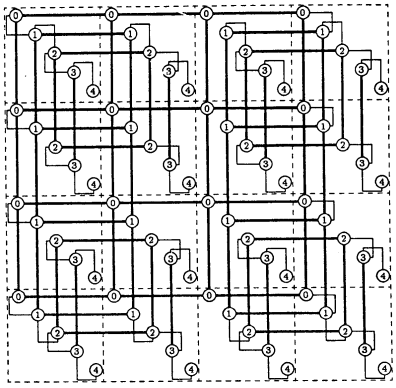
# Consider Full Population Tree



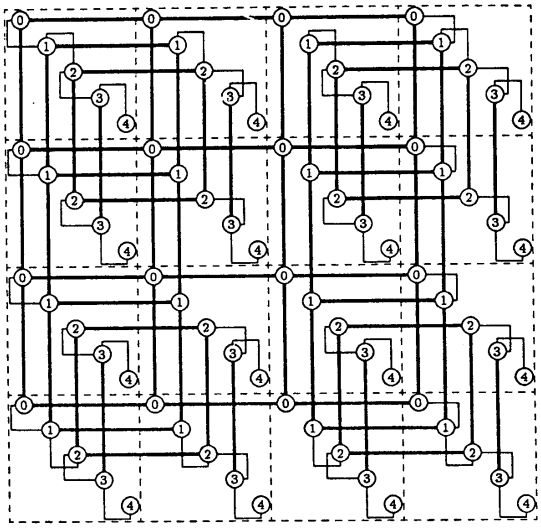


# Can Fold Up





# Gives Uniform Channels



Works nicely
p=0.5

[Greenberg and Leiserson, *Appl. Math Lett.* v1n2p171, 1988]



# How wide are channels?

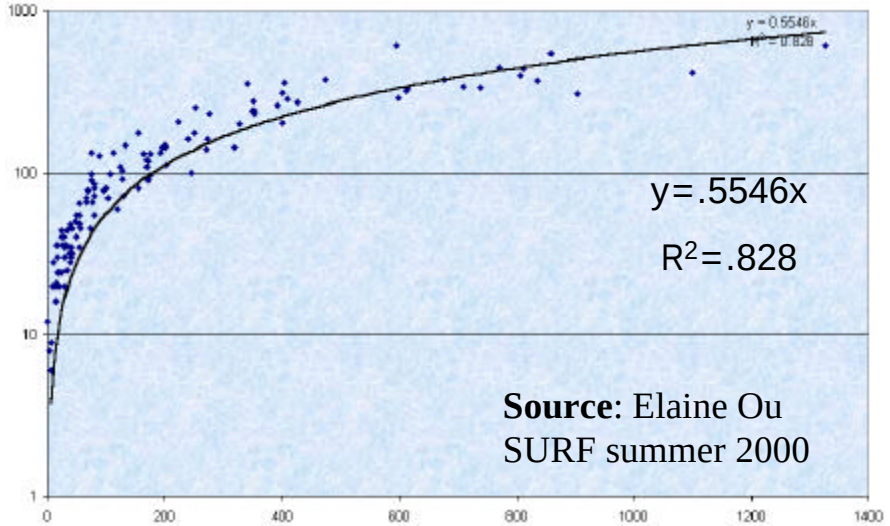
- $W = [w(l) + w(l-1)]/\sqrt{N} + [w(l-2) + w(l-3)]/\sqrt{(N/4)} + \ldots$
- $w_b(l) = c(2^l)^p$
- Share across $\sim 2^{(l/2)}$
- $W = cN^{p-0.5}(1 + 2^{0.5}/2^p + 2^{2 \cdot 0.5}/2^{2p} + \ldots)$
- $W \mu N^{p-0.5}$ $(p>0.5)$

# Implications?

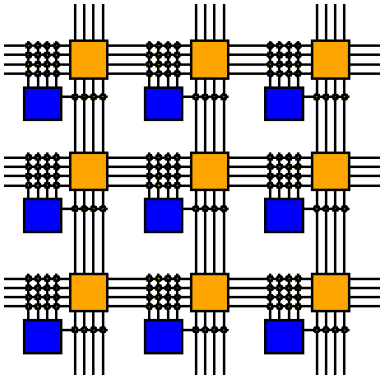
- On Mesh:
  - Upper bound on channel width
    - (assuming full population interconnect)
    - for something characterized by Rent's Rule c,p
    - can use folded hierarchical routing
    - $w \mu N^{p-0.5}$
    - Same as lower bound, different constant
- On Hierarchical:
  - with this layout:
  - channels within constant factor of mesh

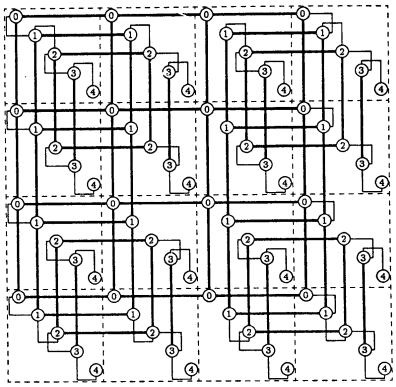


# Channel Width vs. $Cn^p$ (max Rent parameters)



# What's Different?



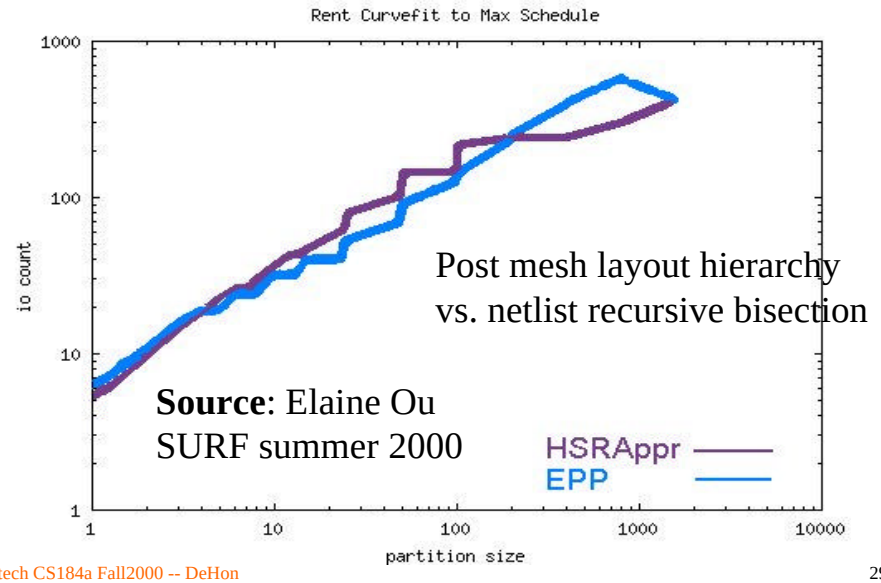




# What's Different?

- Logical and physical closeness
  - with shortcuts, tree has
- Switches in Path
  - Ö N vs. log N
    - depends on how interpret switching nodes
- Mesh connect directly to any channel
- Hierarchical must to climb tree
  - part of how it manages to traverse only log switches

# Rent parameters from a large circuit




Post mesh layout hierarchy
vs. netlist recursive bisection

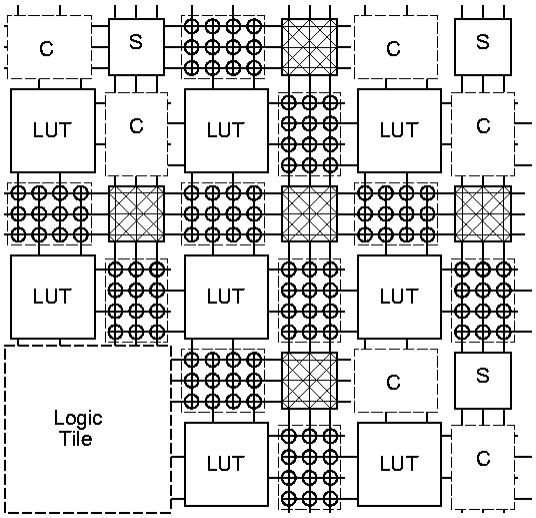
**Source**: Elaine Ou
SURF summer 2000



# Depopulation

# Traditional Mesh Population

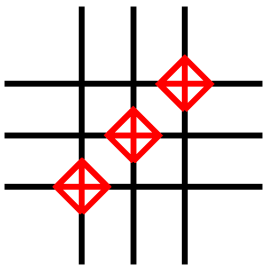
- Switchbox contains only a linear number of switches in channel width
  - 6w vs.
  - $12w^2$






# Diamond Switch

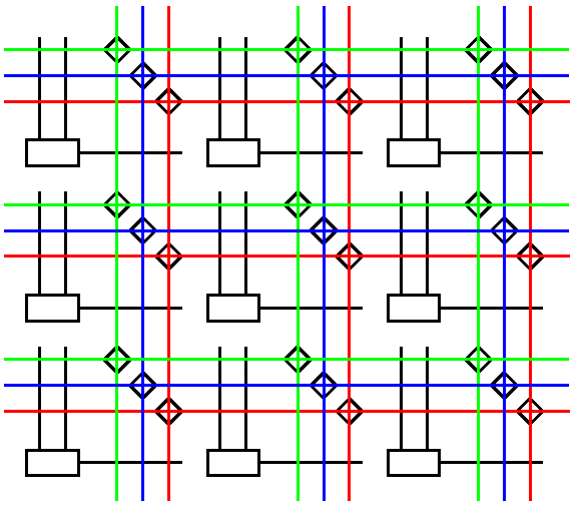
- Typical switchbox pattern:



- Many less switches, but cannot guarantee will be able to use all the wires
  - may need more wires than implied by Rent, since cannot use all wires
  - for mesh: this was already true…now more so

# Domain Structure

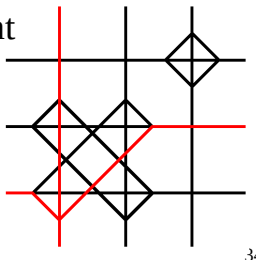
- Once enter network (choose color) can only switch within domain





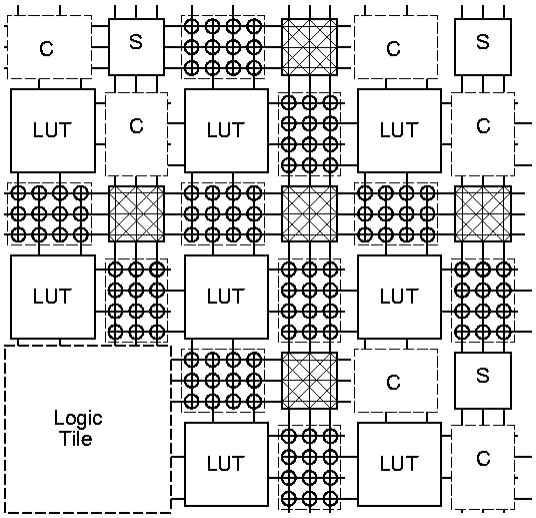
# Universal SwitchBox

- Same number of switches as diamond
- Locally: can guarantee to satisfy any set of requests
  - request = direction through swbox
  - as long as meet channel capacities
  - and order on all channels irrelevant
  - can satisfy
- Not a global property
  - no guarantees between swboxes



# Inter-Switchbox Constraints

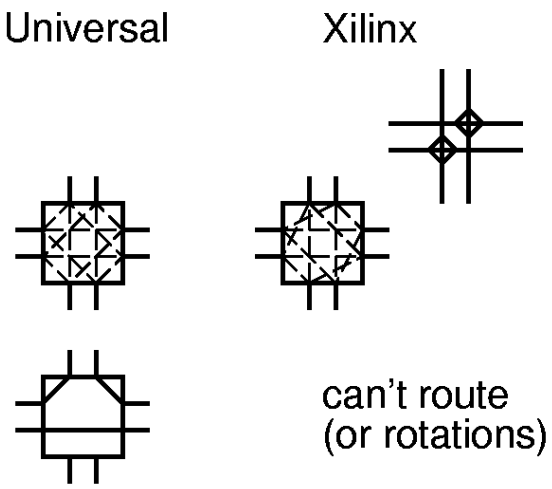
- Channels connect switchboxes
- For valid route, must satisfy all adjacent switchboxes






# Diamond vs. Universal?

- Universal routes strictly more configurations



# Mapping Ratio?

- How bad is it?
- How much wider do channels have to be?

- Mapping Ratio:
  - detail channel width required / global ch width



# Mapping Ratio

- Empirical:
  - Seems plausible, constant in practice
  - anecdotal/published data usually has mapping ratio < 1.5
  - Elaine's data was detail
    - supports CMR model
- Theory/provable:
  - There is no Constant Mapping Ratio
  - can be arbitrarily large!

# Switching Requirements

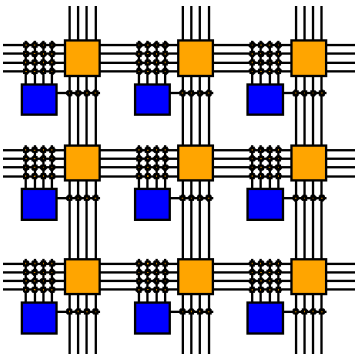
- Linear Population Mesh
- Assuming a constant mapping ratio
- Sw/swbox = 6w
- sw/LUT = (K+6+1)w
- $w\mu N^{p-0.5}$
- $SW/LUT\mu N^{p-0.5}$
- Total Switches $W \mu N^{p+0.5} < N^{2p}$
- Switches grow slower than wires



# Checking Constants: Full Population

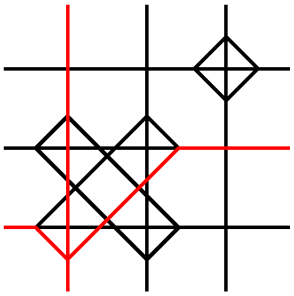
- Wire pitch = $8\lambda$
- switch area = $2500\ \lambda^2$
- wire area: $(8w)^2$
- switch area: $12 \times 2500\ w^2$
- effective wire pitch:
  - $174\ \lambda$

    ~20 times pitch



# Checking Constants

- Wire pitch = 8λ
- switch area = 2500 $\lambda^2$
- wire area: $(8w)^2$
- switch area: 6´2500 w
- crossover
  - w=234 ?
  - (practice smaller)





# Practical

- Since wires aren't dominating
  - under this cost model
  - when both grow at same asymptote
- Can afford to not use some wires perfectly
  - to reduce switches
- Just showed:
  - would take 20x Mapping Ratio for linear population to take same area as full population

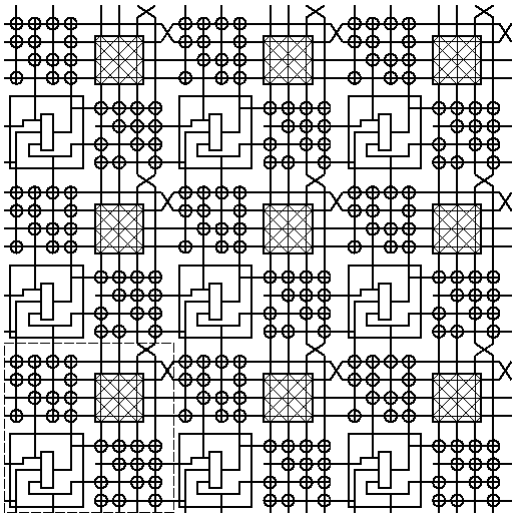
# Routability

- Domain Routing is NP-Complete
  - can reduce coloring problem to domain selection
  - (another reason routers are slow)



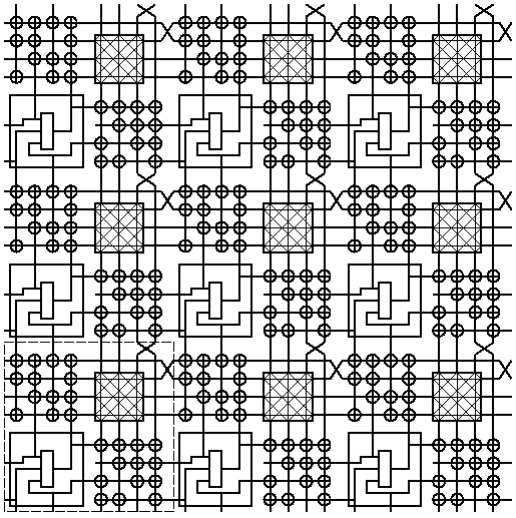
# Segmentation

- To improve speed (decrease delay)
- Allow wires to bypass switchboxes
- Maybe save switches?
- Certainly cost more wire tracks



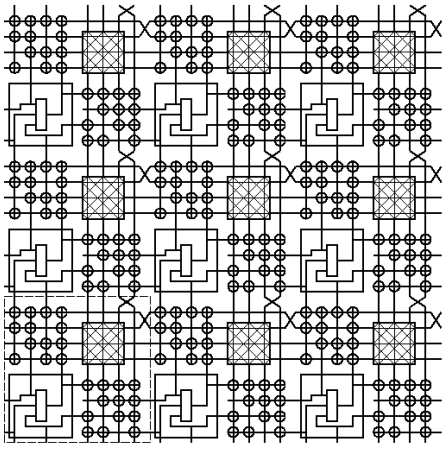
# Segmentation

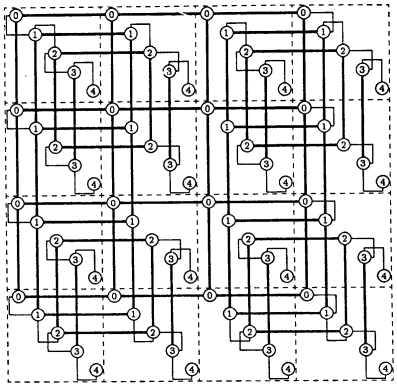
- Reduces switches on path
- May get fragmentation
- Another cause of unusable wires





# Mesh with Hierarchy vs. Fold-and-Squash Tree?



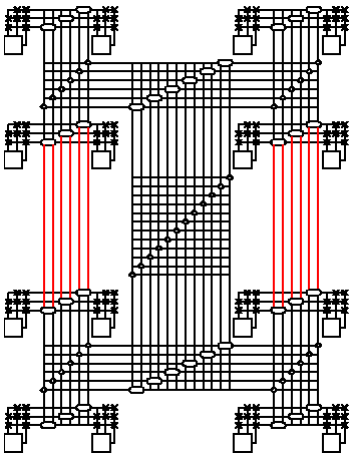


# Depopulation in Tree



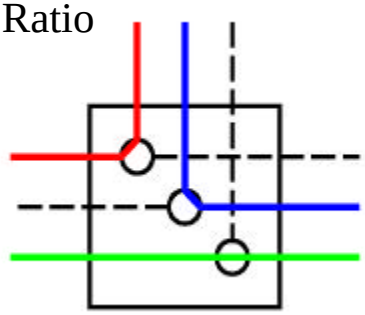
# Linear Population in Tree

- Similar Strategy
- 3-way switch boxes
  - T: 3w (5w w/ short)
  - Pi: 5w (9w w/ short)



# Linear Population

- Will also have a Mapping Ratio
  - at least 1.5 on T stages



- But is it a constant mapping ratio?
  - Have not been able to prove
  - some evidence works in practice

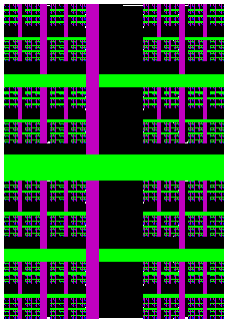


# Switching Requirements Linear Population

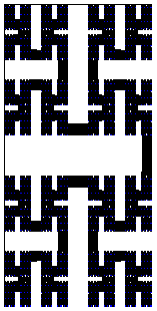
- Key thing to note:
  - as go up the tree
  - half as many switchboxes
  - with (asymptotically) $2^p$ more channels
  - O(w) switches per channel
  - so $2^p/2$ less total switches at each stage
  - …simple geometric regression
- Total number of switches is linear in N
  - compare everything else growing faster than N

# Checking Constants

- 1024 PEs, p=0.67
  - shown to scale



Quadratic/perfect
C=5

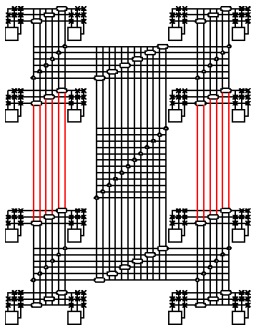


Linear
C=8

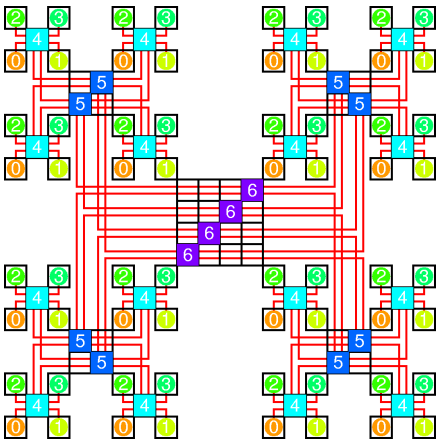
**Again**:
worth wasting some wires to reduce switch growth



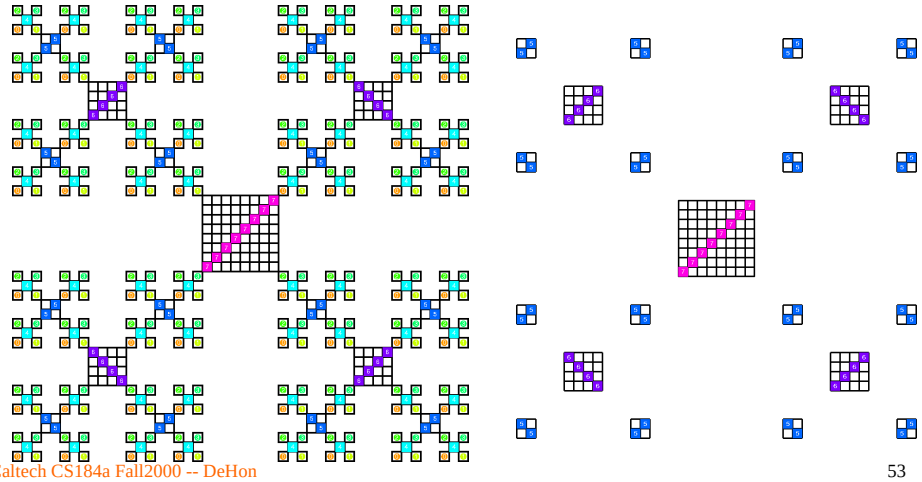
# Fold and Squash Layout



Caveat: may only work conveniently w/ p=0.5

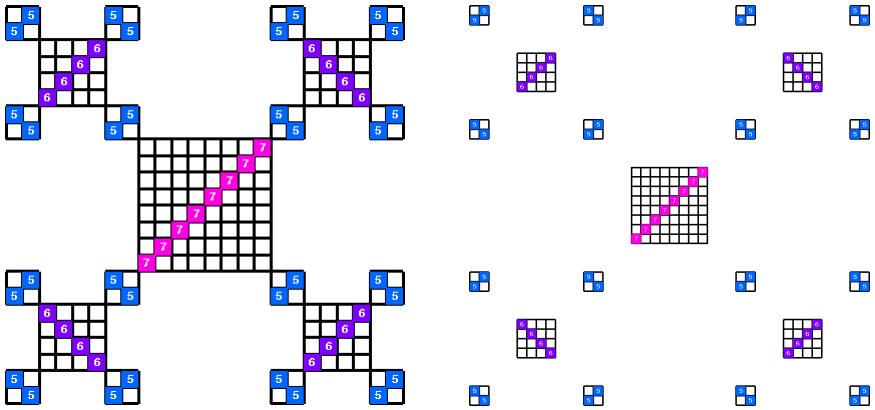


# Fold and Squash Layout

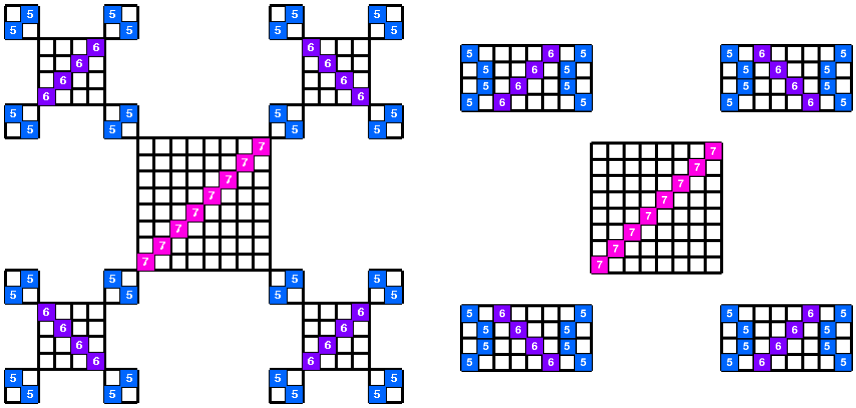




# Folding

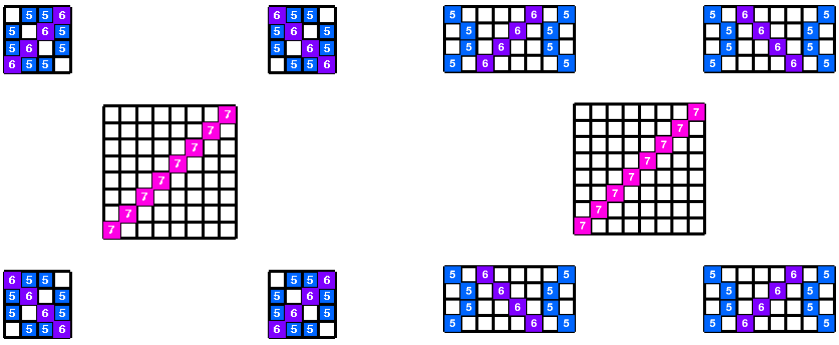


# Folding

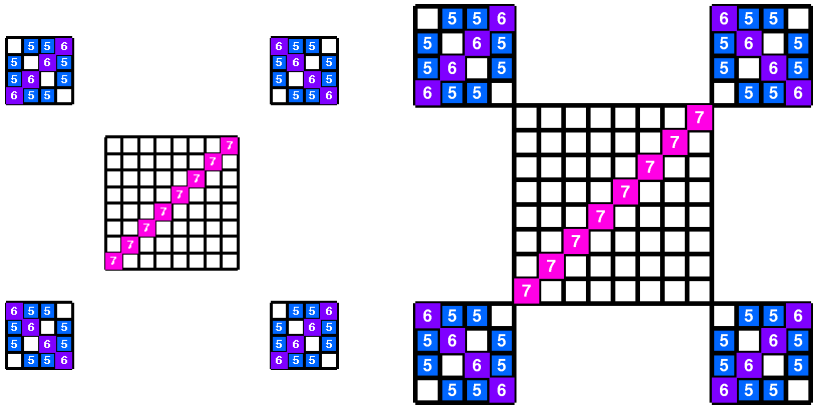




# Folding

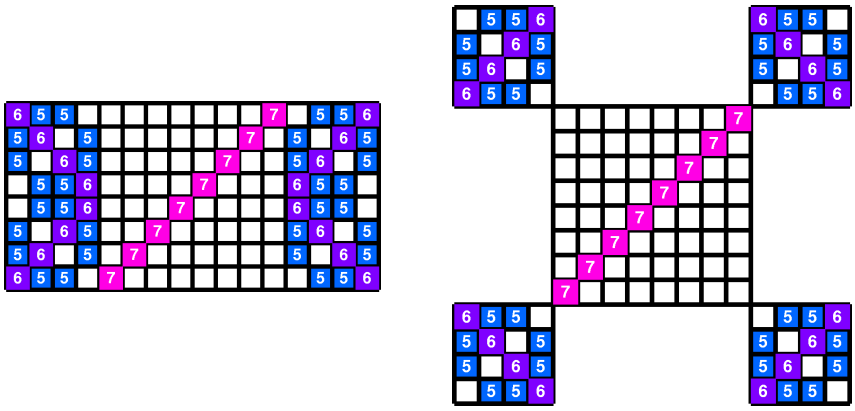


# Folding

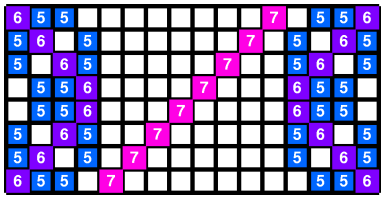


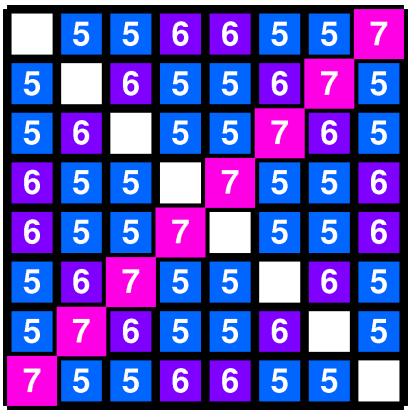


# Folding



# Folding

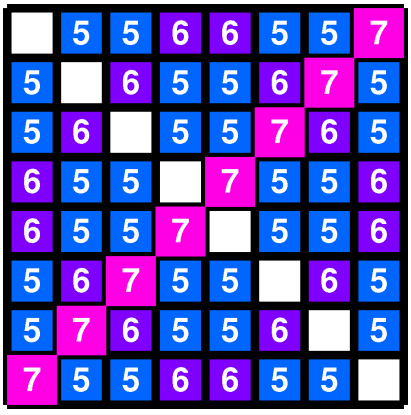






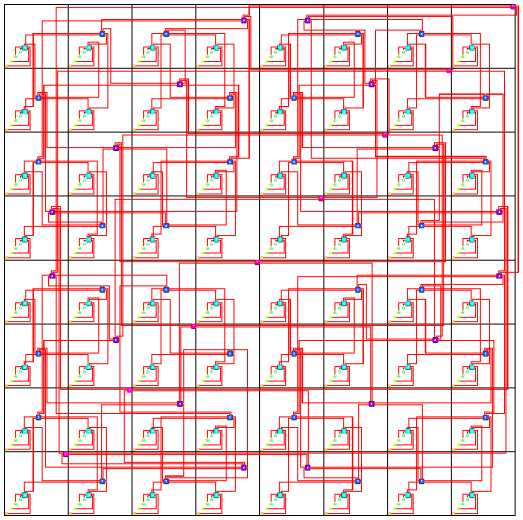
# Folding Invariants

- Lower folds leave both diagonals free
- Current level consumes one, leaving other free



# Compact Folded Layout

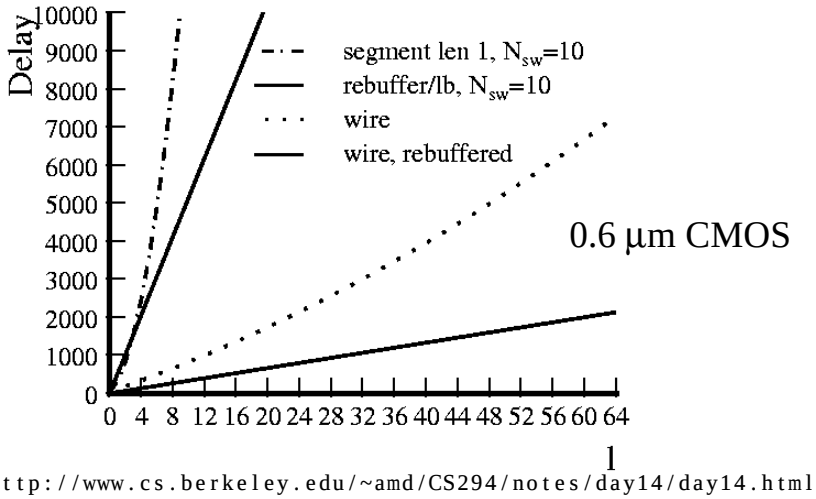
- Can contain switches to constant area
- Wires still grow faster than linear
- Can use extra wire layers to accommodate wire growth
- (whereas switches not helped by additional wire layers)





# Switching and Delay

# Delay through Switching




http://www.cs.berkeley.edu/~amd/CS294/notes/day14/day14.html



# Big Ideas [MSB Ideas]

- Cannot ignore switches
  - area or delay
- Switch population for guaranteed route
  - $O(N^{2p})$
  - like wires, but in CMOS switches larger
- Similarities of Hierarchical and Mesh
- Mesh w grow as $N^{p-0.5}$

# Big Ideas
# [MSB Ideas]

- Switchbox depopulation
  - save considerably on area (delay)
  - will waste wires
  - routing no longer guaranteed
  - routing becomes NP-complete
- Hierarchical/bypass routes
  - can reduce switching delay
  - costs more wires (fragmentation of wires)