

# CS184a: Computer Architecture (Structure and Organization)

Day 13: February 6, 2003  
Interconnect 3: Richness



Caltech CS184 Winter2003 -- DeHon

## Last Time

- Rent's Rule
  - And it's implications
- Superlinear growth rate of interconnect  
 $p > 0.5$ 
  - Area growth  $\Omega(N^{2p})$

Caltech CS184 Winter2003 -- DeHon

# Today

- How rich should interconnect be
  - specifics of understanding interconnect
  - methodology for attacking these kinds of questions

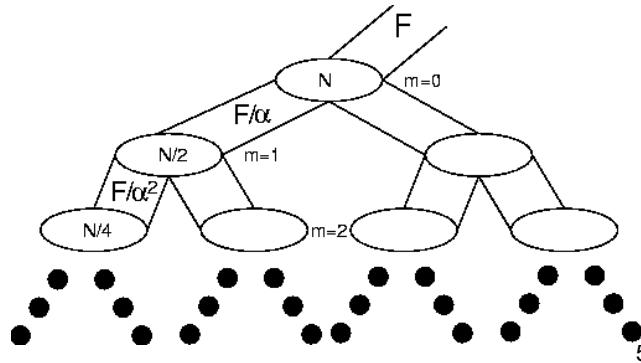
# Now What?

- There is structure (locality)
- Rent characterizes locality
  
- How rich should interconnect be?
  - Allow full utilization?
  - Most area efficient?
  - Model requirements and area impact

# Step 1: Build Architecture Model

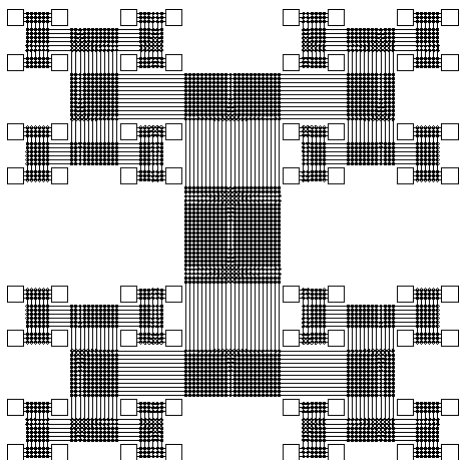
- Assume geometric growth
- Pick parameters: Build architecture can tune

- $F, C$
- $\alpha, p$



Caltech CS184 Winter2003 -- DeHon

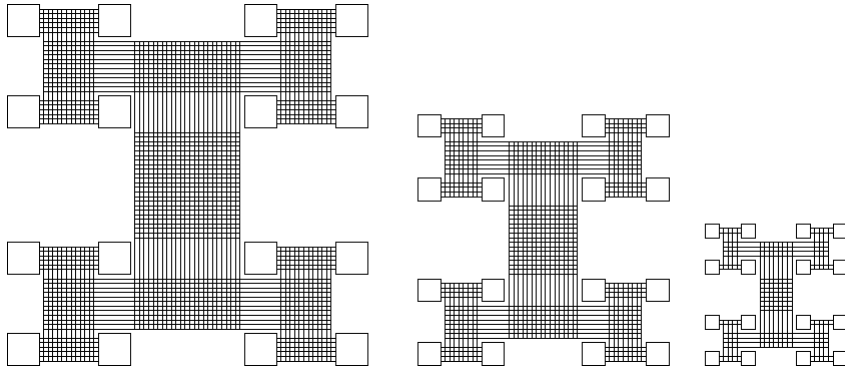
# Tree of Meshes



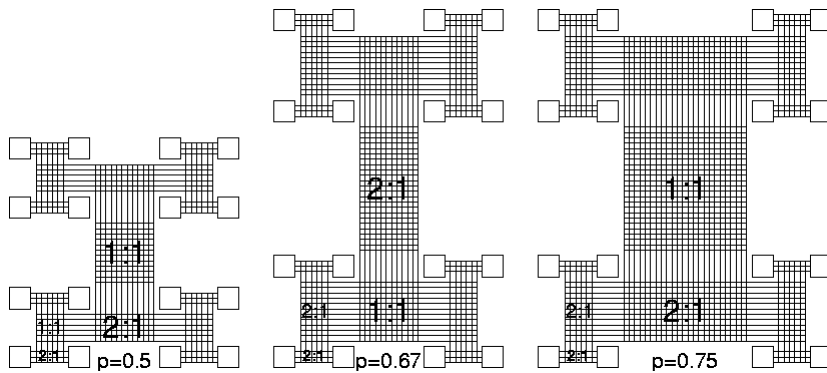
- Nature model is hierarchical
- Restricted internal bandwidth
- Can match to model

Caltech CS184 Winter2003 -- DeHon

# Parameterize C



# Parameterize Growth



$$(2 \ 1)^* \Rightarrow \alpha = \sqrt{2}$$

$$(2 \ 2 \ 2 \ 1)^* \Rightarrow \alpha = 2^{(3/4)}$$

$$(2 \ 2 \ 1)^* \Rightarrow \alpha = (2*2)^{(1/3)} = 2^{(2/3)}$$

## Step 2: Area Model

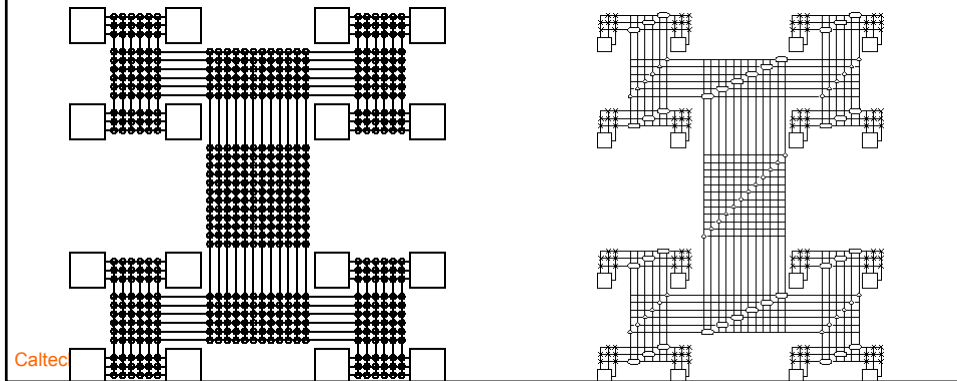
- Need to know effect of architecture parameters on area (costs)
  - focus on dominant components
    - wires
    - switches
    - logic blocks(?)

## Area Parameters

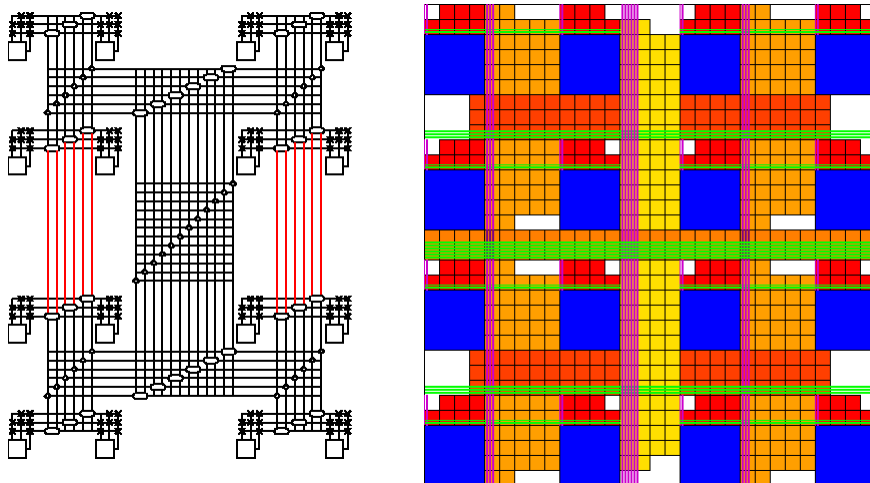
- $A_{\text{logic}} = 40K\lambda^2$
- $A_{\text{sw}} = 2.5K\lambda^2$
- Wire Pitch =  $8\lambda$

# Switchbox Population

- Full population is excessive (next lecture)
- **Hypothesis:** linear population adequate
  - still to be (dis)proven

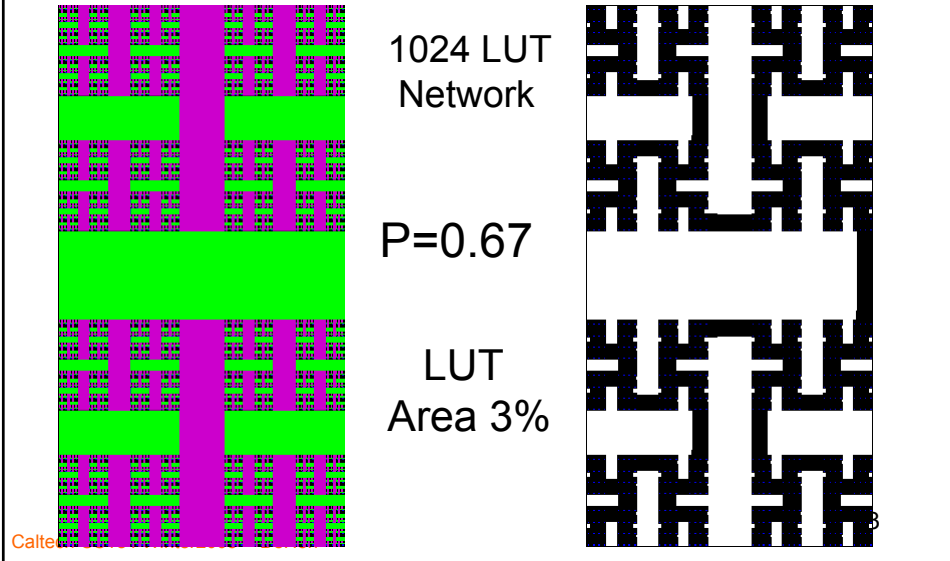


# “Cartoon” VLSI Area Model

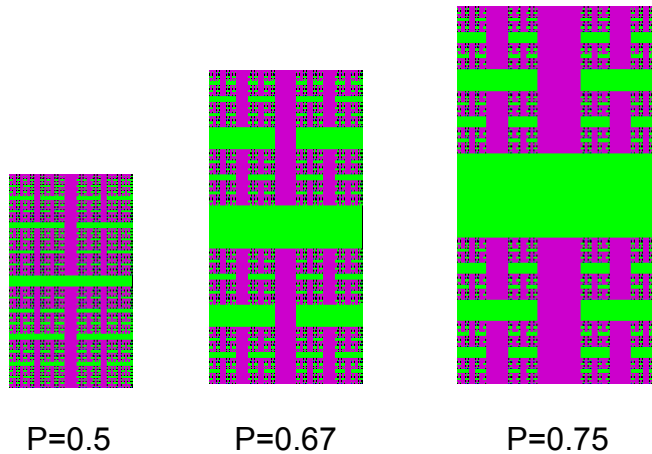


(Example artificially small for clarity)

# Larger "Cartoon"

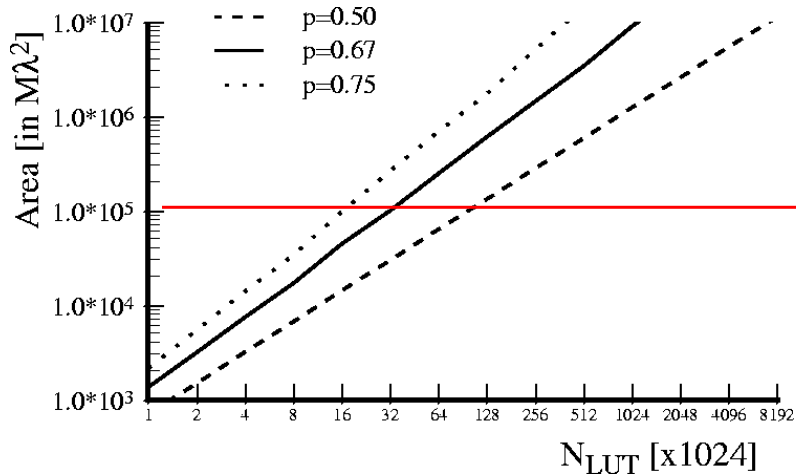


## Effects of $P$ ( $\alpha$ ) on Area



1024 LUT Area Comparison

## Effects of P on Capacity



Caltech CS184 Winter2003 -- DeHon

15

## Step 3: Characterize Application Requirements

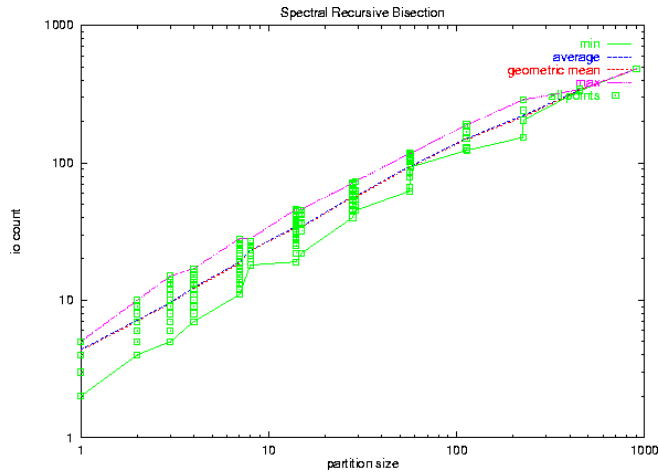
- Identify representative applications.
  - Today: IWLS93 logic benchmarks
- How much structure there?
- How much variation among applications?

Caltech CS184 Winter2003 -- DeHon

16



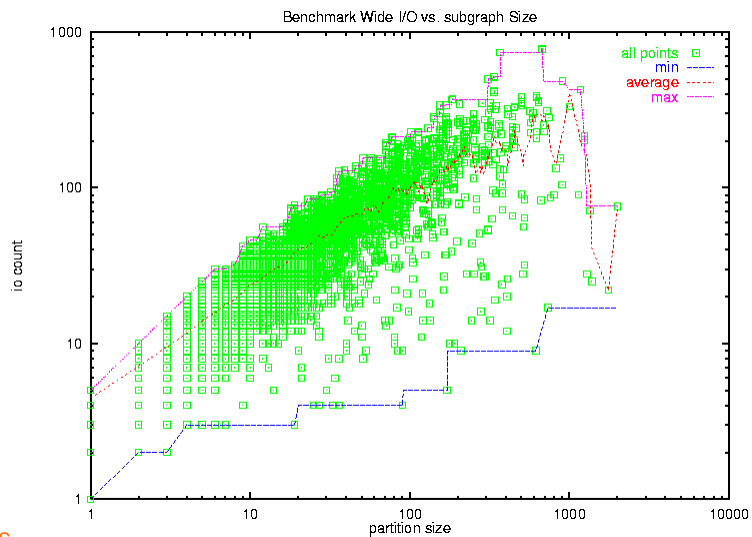
# Application Requirements



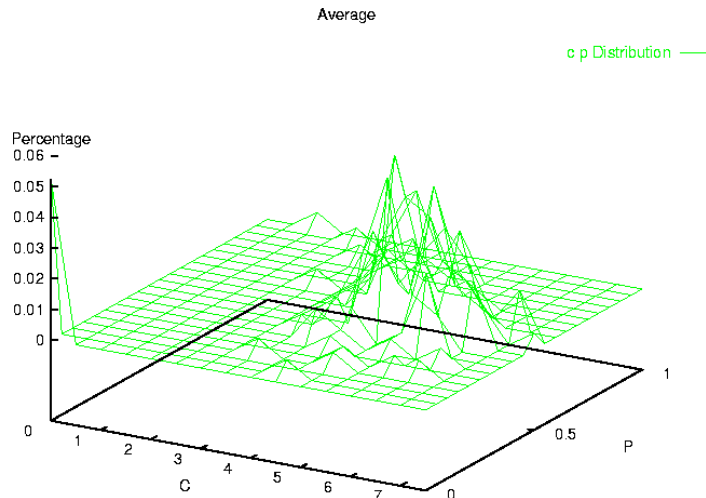
Max:  $C=7, P=0.68$

Avg:  $C=5, P=0.72$

# Benchmark Wide



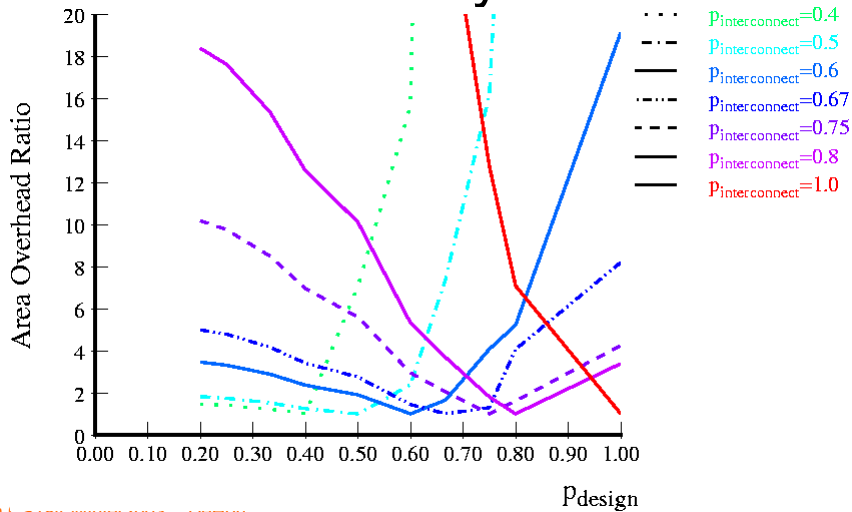
# Benchmark Parameters



# Complication

- Interconnect requirements vary among applications
- Interconnect richness has large effect on area
- What is effect of architecture/application mismatch?
  - Interconnect too rich?
  - Interconnect too poor?

## Interconnect Mismatch in Theory



Caltech CS184 Winter2003 -- DeHon

## Step 4: Assess Resource Impact

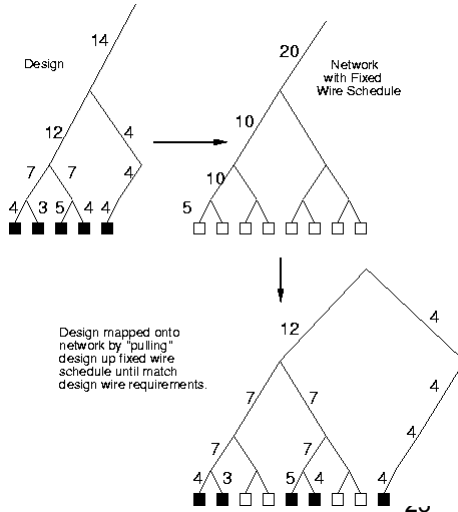
- Map designs to parameterized architecture
- Identify architectural resource required

**Compare:** mapping to k-LUTs; LUT count vs. k.

Caltech CS184 Winter2003 -- DeHon

# Mapping to Fixed Wire Schedule

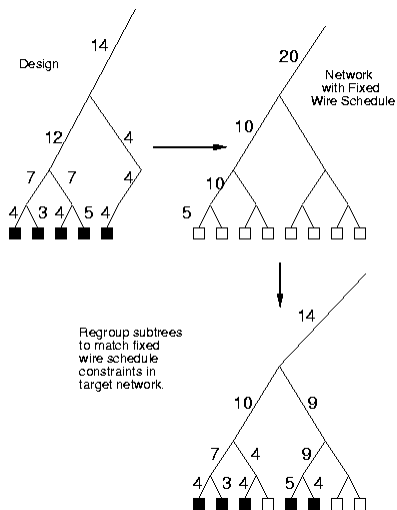
- Easy if need less wires than Net
- If need more wires than net, must depopulate to meet interconnect limitations.



Caltech CS184 Winter2003 -- DeHon

# Mapping to Fixed-WS

- Better results if "reassociate" rather than keeping original subtrees.



Caltech CS184 Winter2003 -- DeHon

## Observation

- Don't really want a "bisection" of LUTs
  - subtree filled to capacity by either of
    - LUTs
    - root bandwidth
  - May be profitable to cut at some place other than midpoint
    - not require "balance" condition
  - "Bisection" should account for both LUT and wiring limitations

## Challenge

- Not know where to cut design into
  - not knowing when wires will limit subtree capacity

## Brute Force Solution

- Explore all cuts
  - start with all LUTs in group
  - consider “all” balances
  - try cut
  - recurse

## Brute Force

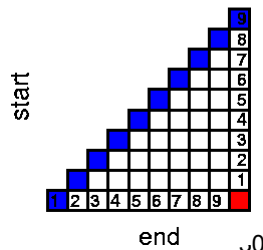
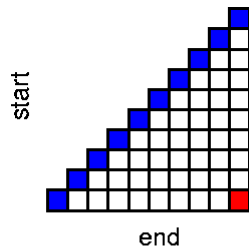
- Too expensive
- Exponential work
  
- ...viable if solving same subproblems

# Simplification

- Single linear ordering
- Partitions = pick split point on ordering
- Reduce to finding cost of [start,end] ranges (subtrees) within linear ordering
- Only  $n^2$  such subproblems
- Can solve with dynamic programming

# Dynamic Programming

- Start with base set of size 1
- Compute all splits of size  $n$ , from solutions to all problems of size  $n-1$  or smaller
- Done when compute where to split  $0, N-1$



# Dynamic Programming

- Just one possible “heuristic” solution to this problem
  - not optimal
  - dependent on ordering
  - sacrifices ability to reorder on splits to avoid exponential problem size
- Opportunity to find a better solution here...

31

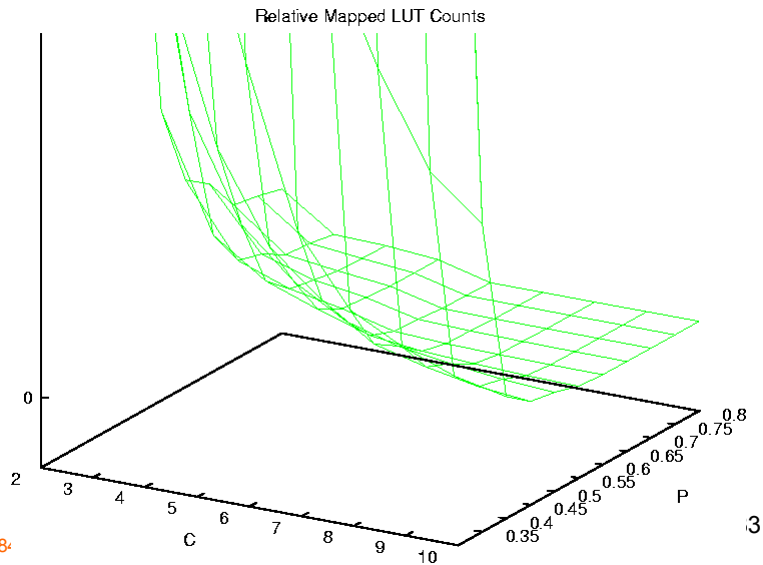
# Ordering LUTs

- Another problem
  - lay out gates in 1D line
  - minimize sum of squared wire length
    - tend to cluster connected gates together
  - Is solvable mathematically for optimal
    - Eigenvector of connectivity matrix
- Use this 1D ordering for our linear ordering

32



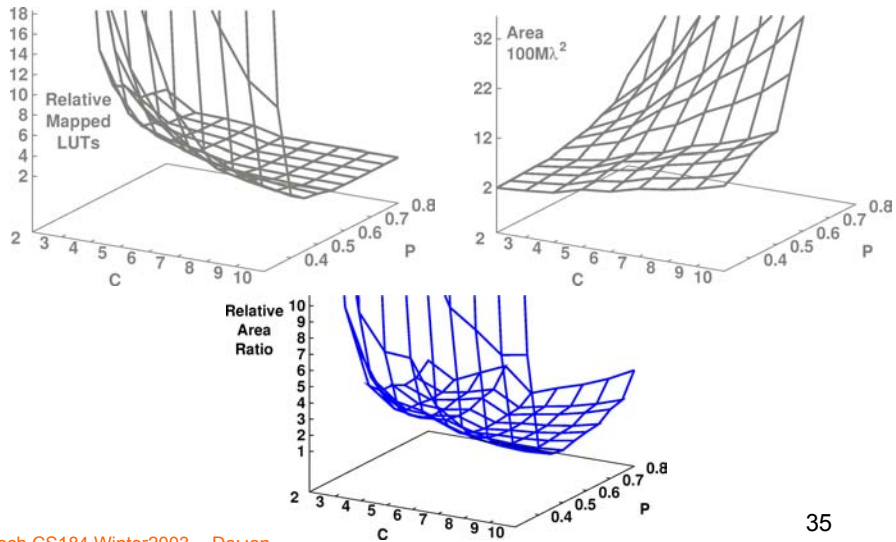
# Mapping Results



## Step 5: Apply Area Model

- Assess impact of resource results

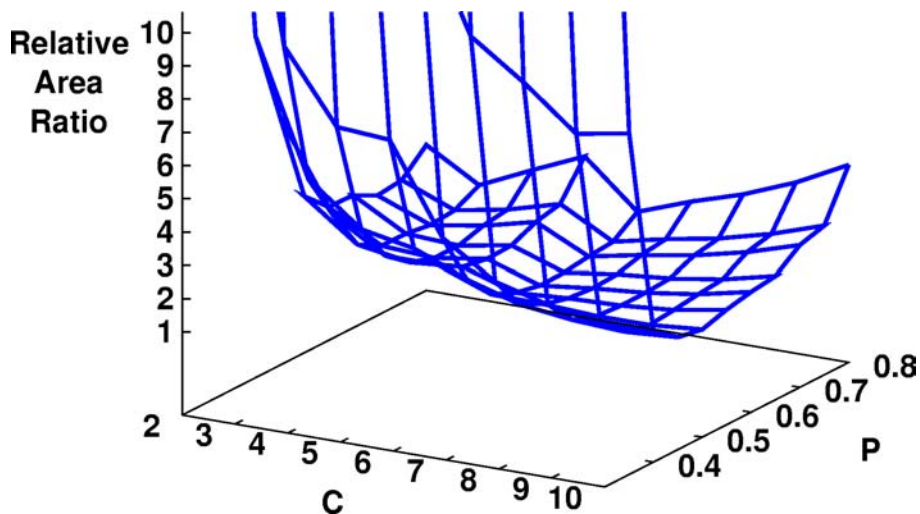
# Resources $\times$ Area Model $\Rightarrow$ Area



Caltech CS184 Winter2003 -- DeHon

35

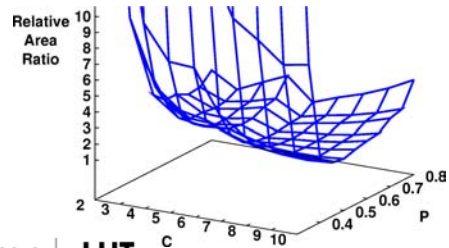
# Net Area



Caltech CS184 Winter2003 -- DeHon

36

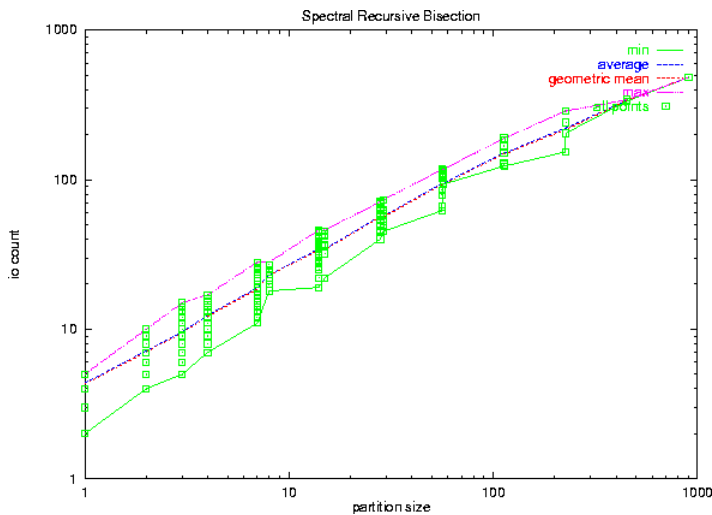
# Picking Network Design Point



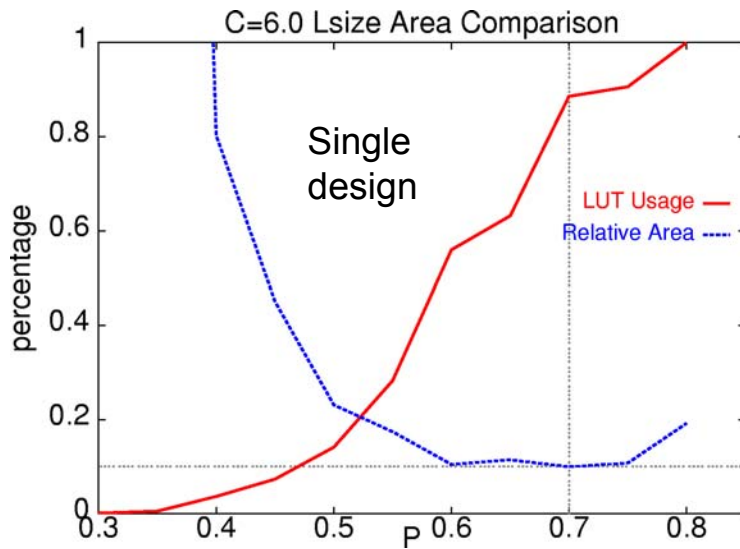
Minimize Objective	params		Sigma rel area	LUT Util.
	C	P		
relative area	6	0.6	1.23	0.87
area with full util	10	0.75	2.98	1.00

Don't optimize for 100% compute util. (100% yield)  
 ...also don't optimize for highest peak.

# What about a single design?



# LUT Utilization predict Area?



Caltech CS184 Winter2003 -- DeHon

39

## Methodology

1. Architecture model (parameterized)
2. Cost model
3. Important task characteristics
4. Mapping Algorithm
  - Map to determine resources
5. Apply cost model
6. Digest results
  - find optimum (multiple?)
  - understand conflicts (avoidable?)

Caltech CS184 Winter2003 -- DeHon

40

## Big Ideas [MSB Ideas]

- Interconnect area dominates logic area
- Interconnect requirements vary
  - among designs
  - within a single design
- To minimize area
  - focus on using dominant resource (interconnect)
  - may underuse non-dominant resources (LUTs)

41

Caltech CS184 Winter2003 -- DeHon

## Big Ideas [MSB Ideas]

- Two different resources here
  - compute, interconnect
- Balance of resources required varies among designs (even within designs)
- Cannot expect full utilization of every resource
- Most area-efficient designs may *waste* some compute resources (cheaper resource)

42

Caltech CS184 Winter2003 -- DeHon