

CS137: Electronic Design Automation

Day 8: January 30, 2002
Placement
(Intro, Constructive)



CALTECH CS137 Winter2002 -- DeHon

Placement

- **Problem:** Pick locations for all building blocks
 - minimizing energy, delay, area
 - really:
 - minimize wire length
 - minimize channel density

CALTECH CS137 Winter2002 -- DeHon

Bad Placement

- How bad can it be?
 - Area
 - delay
 - energy

CALTECH CS137 Winter2002 -- DeHon

Bad: Area

- All wires cross bisection
- $O(N^2)$ area
- good: $O(N)$

CALTECH CS137 Winter2002 -- DeHon

Bad: Delay

- All critical path wires cross chip
- Delay = $O(|\text{PATH}| * 2 * L_{\text{side}})$
 - [and L_{side} as $O(N)$]
- good: $O(|\text{PATH}| * L_{\text{cell}})$
- compare 100ps gates to many nanoseconds

CALTECH CS137 Winter2002 -- DeHon

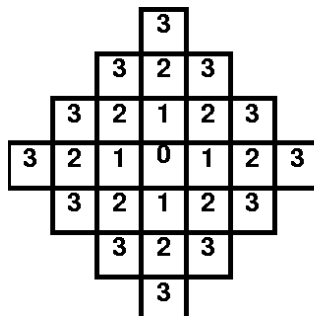
Bad: Energy

- All wires cross chip:
 - $O(L_{\text{side}})$ long $\rightarrow O(L_{\text{side}})$ capacitance per wire
 - $\times O(N)$ wires $\rightarrow O(N^2)$ capacitance
- Good:
 - $O(1)$ long wires $\rightarrow O(N)$ capacitance

CALTECH CS137 Winter2002 -- DeHon

Distance

- Can we place everything close?



CALTECH CS137 Winter2002 -- DeHon

“Closeness”

- Try placing “everything” close

Manhattan Distance	Places	Transitive Fanin
1	4	4
2	8	16
3	12	64
i	i	i
n	4^n	4^n



CALTECH CS137

Problem Characteristics

- Familiar
 - NP Complete
 - local, greedy not work
 - greedy gets stuck in local minima

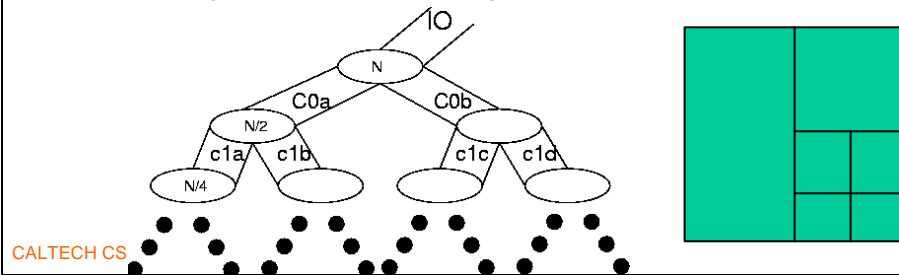
CALTECH CS137 Winter2002 -- DeHon

Constructive Placement

CALTECH CS137 Winter2002 -- DeHon

Basic Idea

- Partition (bisect) to define halves of chip
 - minimize wire crossing
- Recurse to refine
- When get down to single component, done



Adequate?

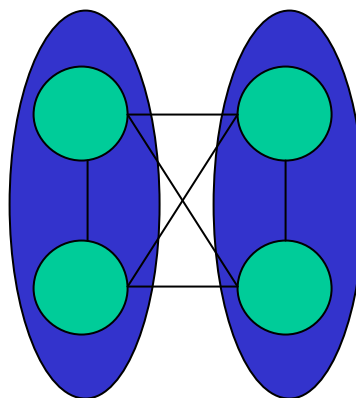
- Does recursive bisection capture the primary constraints of two-dimensional placement?

Problems

- Greedy, top-down cuts
 - maybe better pay cost early?
- Two-dimensional problem
 - (often) no real cost difference between H and V cuts
- Interaction between subtrees
 - not modeled by recursive bisect

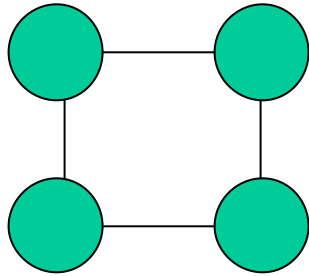
CALTECH CS137 Winter2002 -- DeHon

Interaction

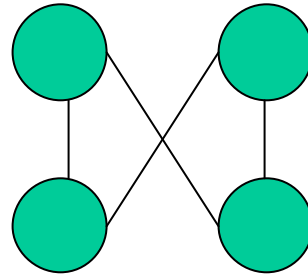


CALTECH CS137 Winter2002 -- DeHon

Example



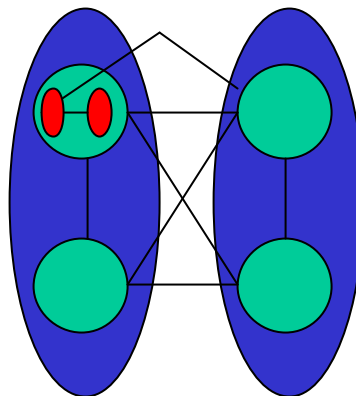
Ideal split
(not typical)



“Equivalent” split
ignoring external constraints
Practically -- makes all H
cuts also be V cuts

CALTECH CS137 Winter2002 -- DeHon

Interaction



CALTECH CS137 Winter2002 -- DeHon

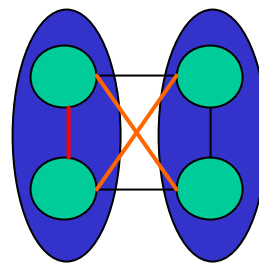
Problem

- Need to keep track of where things are
 - outside of current partition
 - include costs induced by above
- Don't necessarily know where things are
 - still solving problem

CALTECH CS137 Winter2002 -- DeHon

Improvement: Ordered

- Order operations
- Keep track of existing solution
- Use to constrain or pass costs to next subproblem
- Flow cut
 - use existing in src/sink



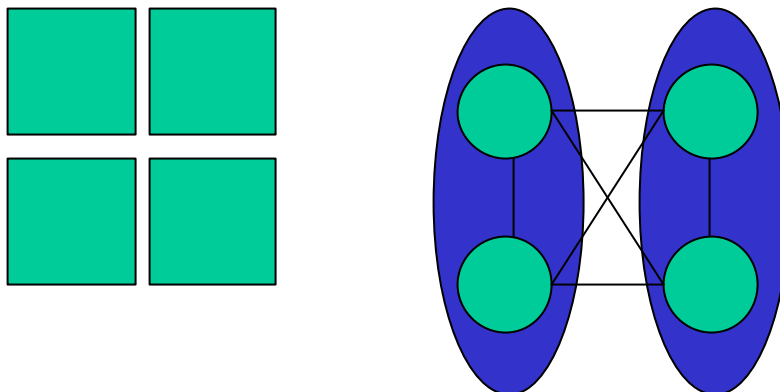
CALTECH CS137 Winter2002 -- DeHon

Improvement: Constrain

- Partition once
- Constrain movement within existing partitions
- Account for both H and V crossings
- Partition next
 - (simultaneously work parallel problems)
 - easy modification to FM

CALTECH CS137 Winter2002 -- DeHon

Constrain Partition



CALTECH CS137 Winter2002 -- DeHon

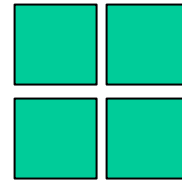
Improvement: Quadrisection

- Solve more of problem at once
- Quadrisection:
 - partition into 4 bins simultaneously
 - keep track of costs all around

CALTECH CS137 Winter2002 -- DeHon

Quadrisection

- Modify FM to work on multiple buckets
- k-way has:
 - $k(k-1)$ buckets
 - from*to
 - quad → 12
- reformulate gains
- update still $O(1)$



CALTECH CS137 Winter2002 -- DeHon

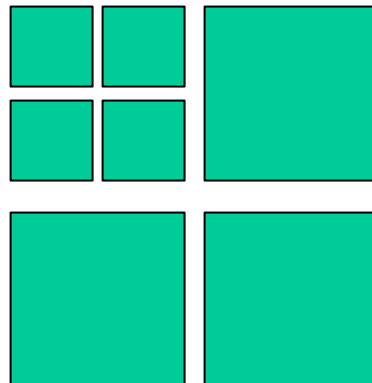
Quadrisect

- Cases (15):
 - (1 partition) x 4
 - (2 part) x 6 = (4 choose 2)
 - (3 part) x 4 = (4 choose 3)
 - (4 part) x 1

CALTECH CS137 Winter2002 -- DeHon

Recurse

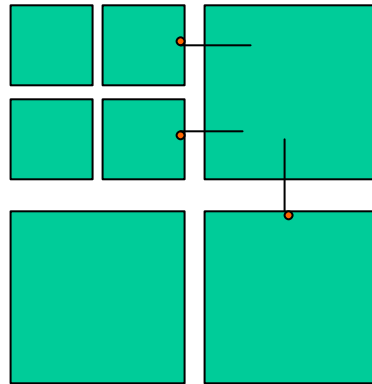
- Keep outside constraints
 - (cost effects)
- Don't know detail place
- Model as at center of unrefined region



CALTECH CS137 Winter2002 -- DeHon

Option: Terminal Propagation

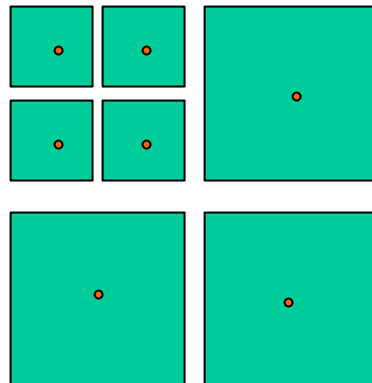
- Abstract inputs as terminals
- Partition based upon
- Represent cost effects on placement/refinement decisions



CALTECH CS137 Winter2002 -- DeHon

Option: Refine

- Keep refined placement
- Use in cost estimates



CALTECH CS137 Winter2002 -- DeHon

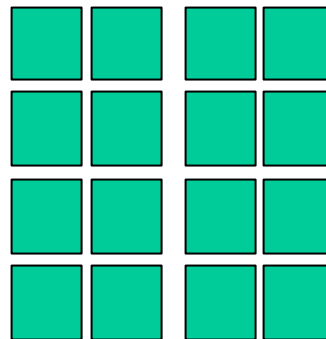
Problem

- Still have ordering problem
- Earlier subproblems solved with weak constraints from later
 - (cruder placement estimates)
- Solved previous case by flattening
 - ...but in extreme give up divide and conquer

CALTECH CS137 Winter2002 -- DeHon

Iterate

- After solve later problems
- Relax solution
- Solve earlier problems again with refined placements (cost estimates)
- Repeat until converge



CALTECH CS137 Winter2002 -- DeHon

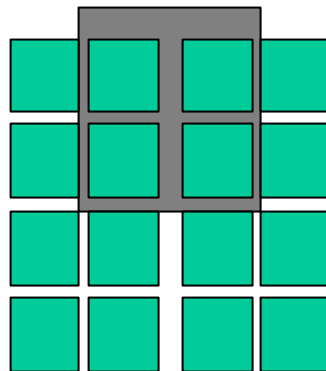
Iteration/Cycling

- General technique to deal with phase-ordering problem
 - what order do we perform transformations, make decisions?
 - How get accurate information to everyone
- Still basically greedy

CALTECH CS137 Winter2002 -- DeHon

Refinement

- Relax using overlapping windows
- Deal with edging effects
- Khang etc. claim 10-15% improve
 - cycle
 - overlap



CALTECH CS137 Winter2002 -- DeHon

Possible Refinement

- Allow unbalanced cuts
 - most things still work
 - just distort refinement grip
 - allowing unbalance using FM quadrisection looks a bit tricky
 - gives another 5-10% improvement

CALTECH CS137 Winter2002 -- DeHon

Runtime

- Each gain update still $O(1)$
 - (bigger constants)
 - so, FM partition pass still $O(N)$
- $O(1)$ iterations expected
- assume $O(1)$ overlaps exploited
- $O(\log(N))$ levels

- Total: $O(N \log(N))$
 - very fast compared to typical annealing

CALTECH CS137 Winter2002 -- DeHon

Uses

- Good by self
- Starting point for simulated annealing
 - speed convergence
- With synthesis (both high level and logic)
 - get a quick estimate of physical effects
 - (play role in estimation/refinement at larger level)
- Early/fast placement
 - before willing to spend time looking for best
- For fast placement where time matters
 - FPGAs, online placement?

CALTECH CS137 Winter2002 -- DeHon

Summary

- Partition to minimize cut size
- Additional constraints to do well
- Quadrisection
- Keep track of estimated placement
- Relax/iterate/Refine

CALTECH CS137 Winter2002 -- DeHon

Big Ideas:

- Divide-and-conquer
- Successive Refinement
- Phase ordering: estimate/relax/iterate