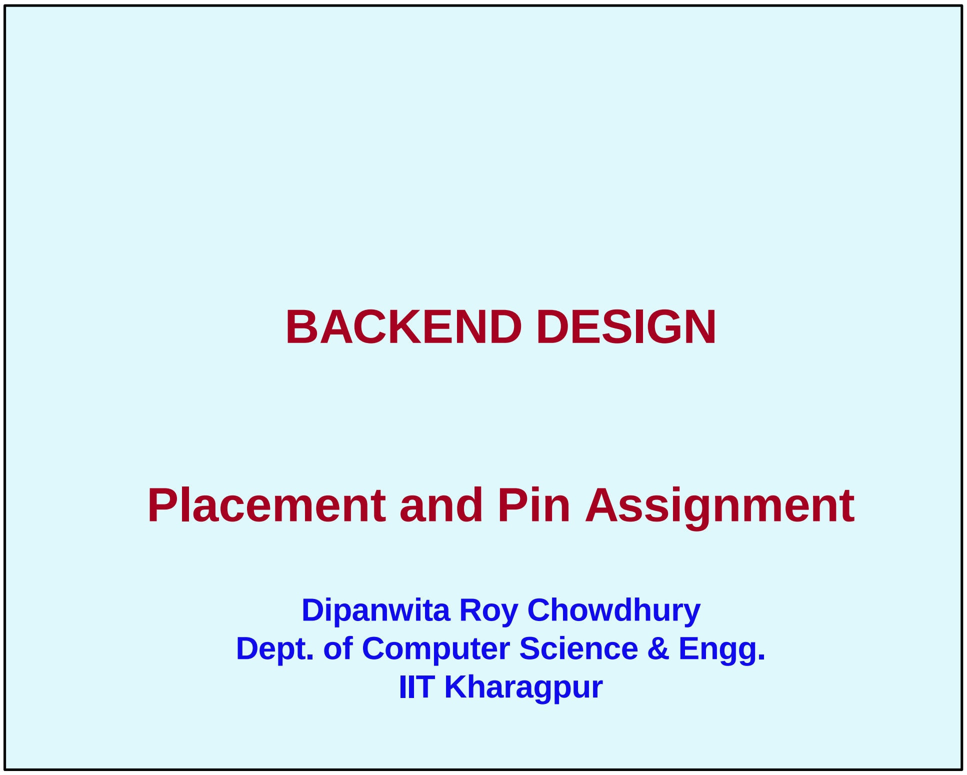
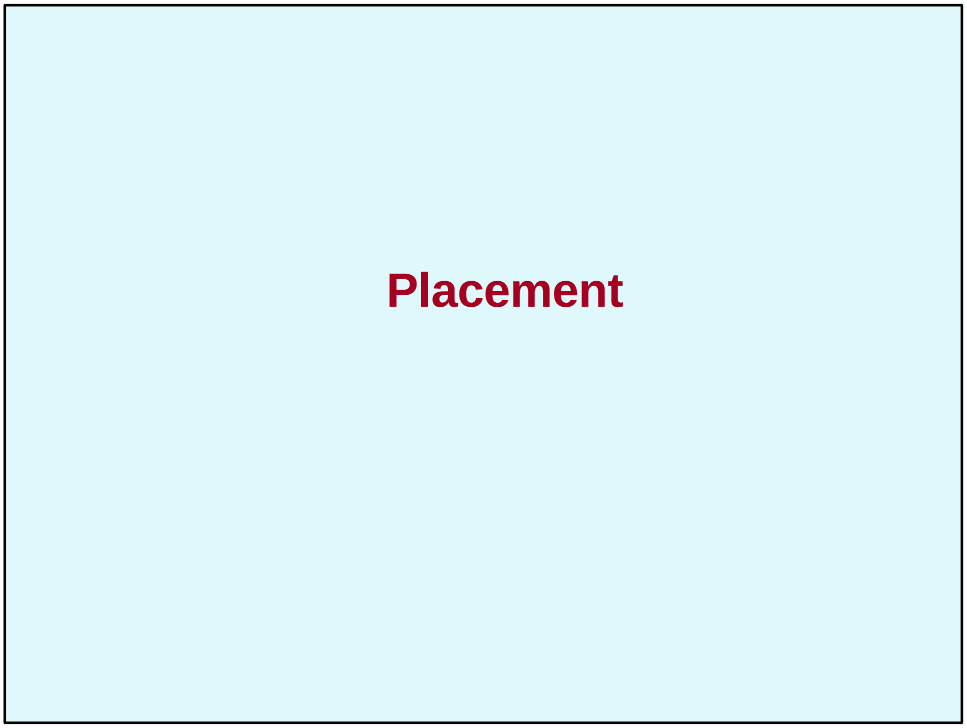


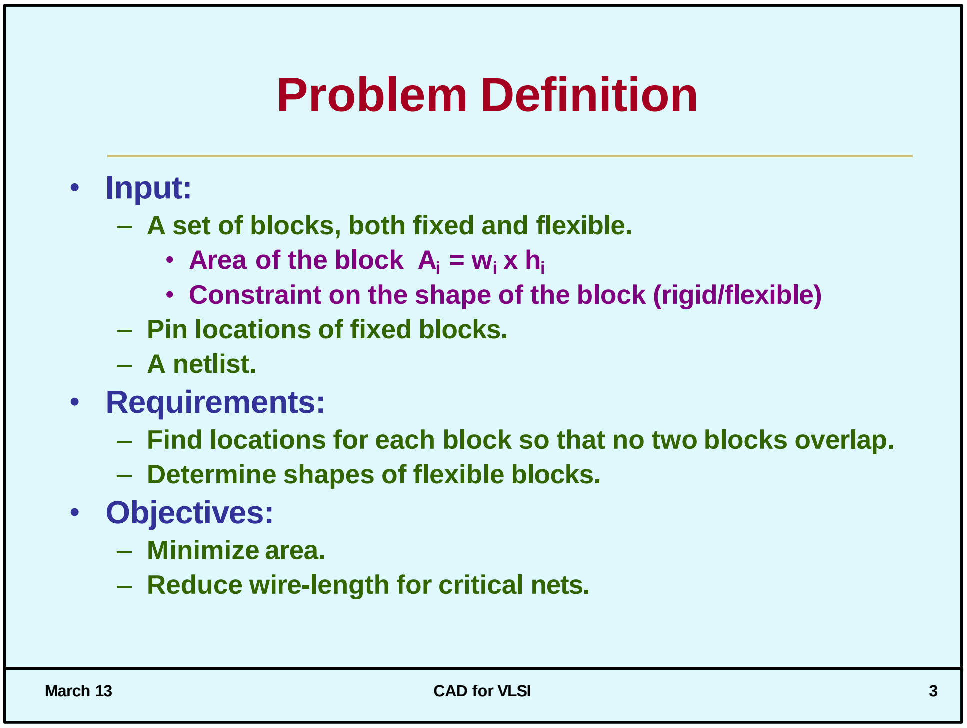
BACKEND DESIGN

Placement and Pin Assignment

Dipanwita Roy Chowdhury
Dept. of Computer Science & Engg.
IIT Kharagpur

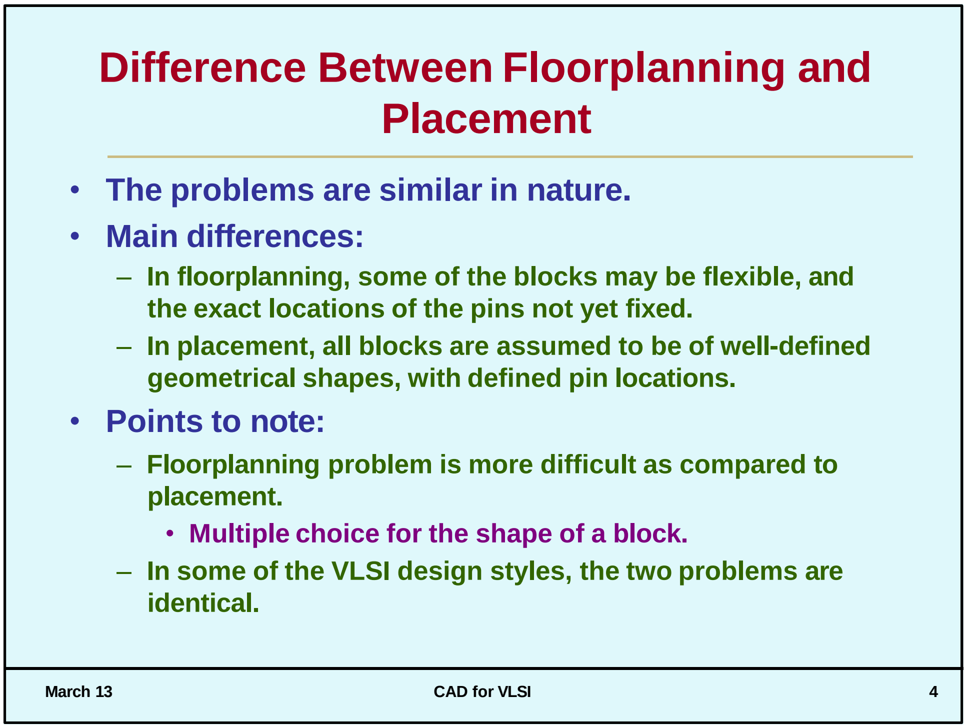


Placement



Problem Definition

- **Input:**
  - A set of blocks, both fixed and flexible.
    - Area of the block $A_i = w_i \times h_i$
    - Constraint on the shape of the block (rigid/flexible)
  - Pin locations of fixed blocks.
  - A netlist.
- **Requirements:**
  - Find locations for each block so that no two blocks overlap.
  - Determine shapes of flexible blocks.
- **Objectives:**
  - Minimize area.
  - Reduce wire-length for critical nets.



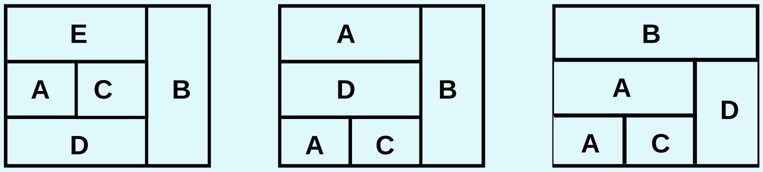
Difference Between Floorplanning and Placement

- The problems are similar in nature.
- Main differences:
  - In floorplanning, some of the blocks may be flexible, and the exact locations of the pins not yet fixed.
  - In placement, all blocks are assumed to be of well-defined geometrical shapes, with defined pin locations.
- Points to note:
  - Floorplanning problem is more difficult as compared to placement.
    - Multiple choice for the shape of a block.
  - In some of the VLSI design styles, the two problems are identical.

An Example for Rigid Blocks

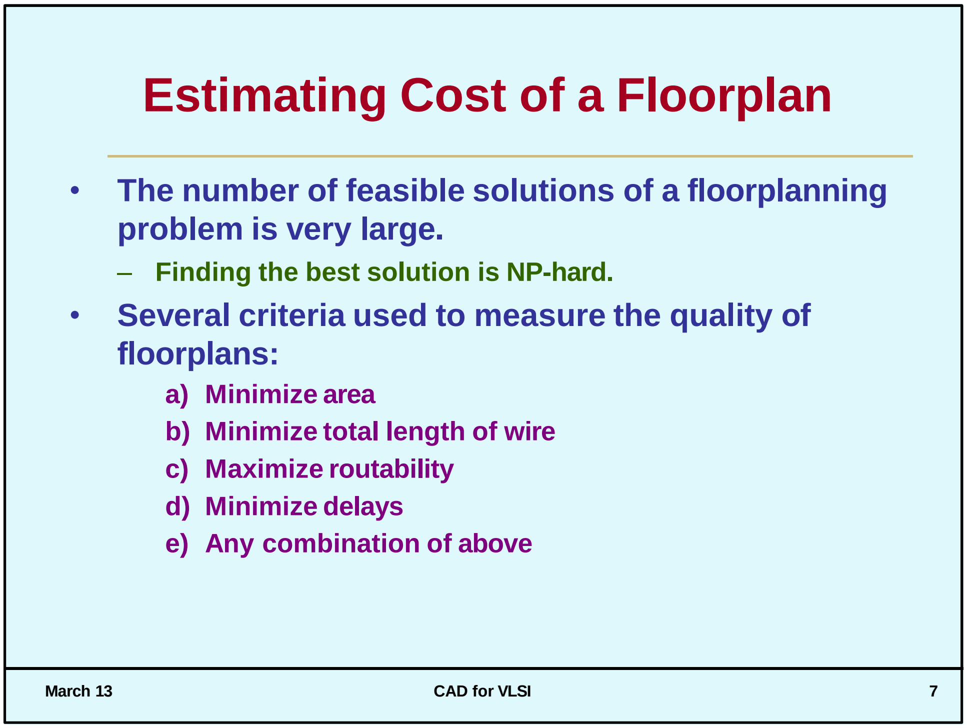
Module	Width	Height
A	1	1
B	1	3
C	1	1
D	1	2
E	2	1

Some of the Feasible Floorplans



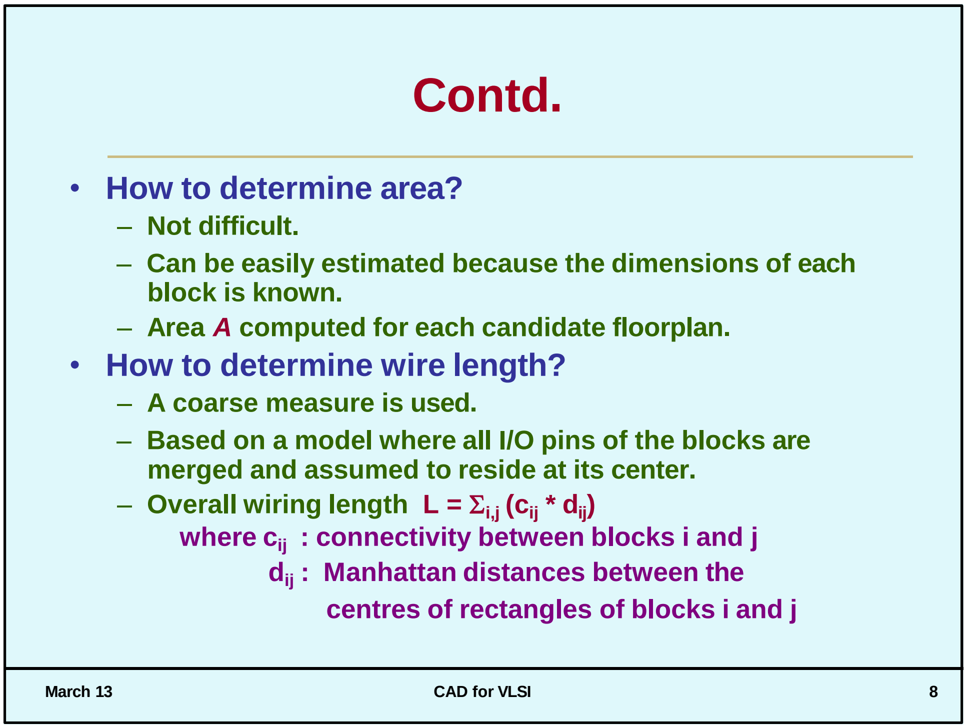
Design Style Specific Issues

- **Full Custom**
  - All the steps required for general cells.
- **Standard Cell**
  - Dimensions of all cells are fixed.
  - Floorplanning problem is simply the placement problem.
  - For large netlists, two steps:
    - First do global partitioning.
    - Placement for individual regions next.
- **Gate Array**
  - Floorplanning problem same as placement problem.



Estimating Cost of a Floorplan

- The number of feasible solutions of a floorplanning problem is very large.
  - Finding the best solution is NP-hard.
- Several criteria used to measure the quality of floorplans:
  - a) Minimize area
  - b) Minimize total length of wire
  - c) Maximize routability
  - d) Minimize delays
  - e) Any combination of above

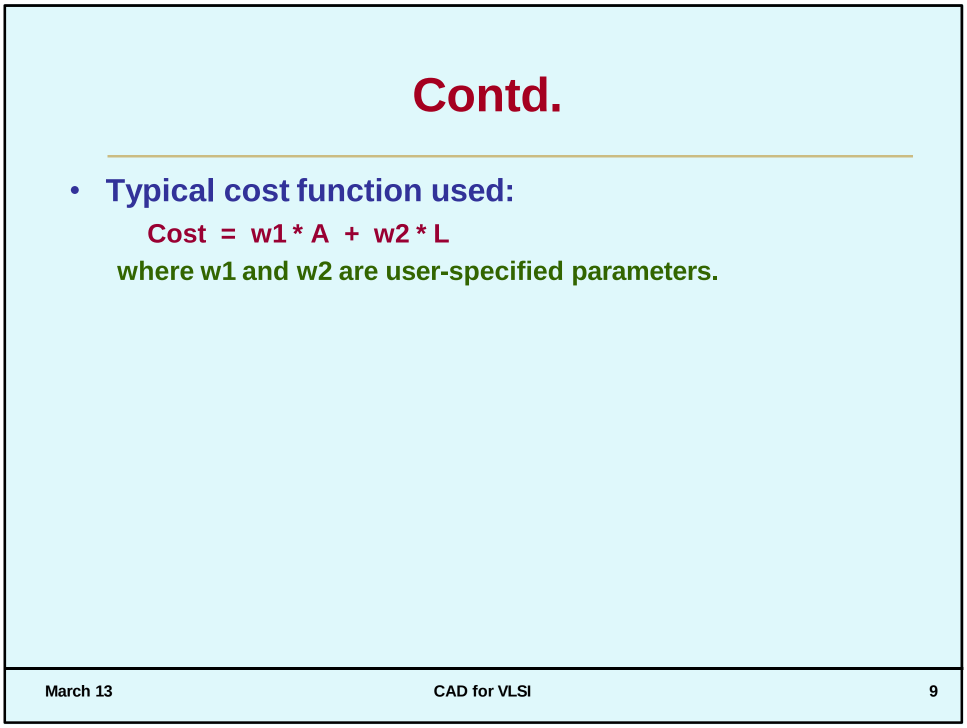


Contd.

- How to determine area?
  - Not difficult.
  - Can be easily estimated because the dimensions of each block is known.
  - Area *A* computed for each candidate floorplan.
- How to determine wire length?
  - A coarse measure is used.
  - Based on a model where all I/O pins of the blocks are merged and assumed to reside at its center.
  - Overall wiring length $L = \Sigma_{i,j} (c_{ij} * d_{ij})$

    where c_{ij} : connectivity between blocks i and j

    d_{ij} : Manhattan distances between the centres of rectangles of blocks i and j

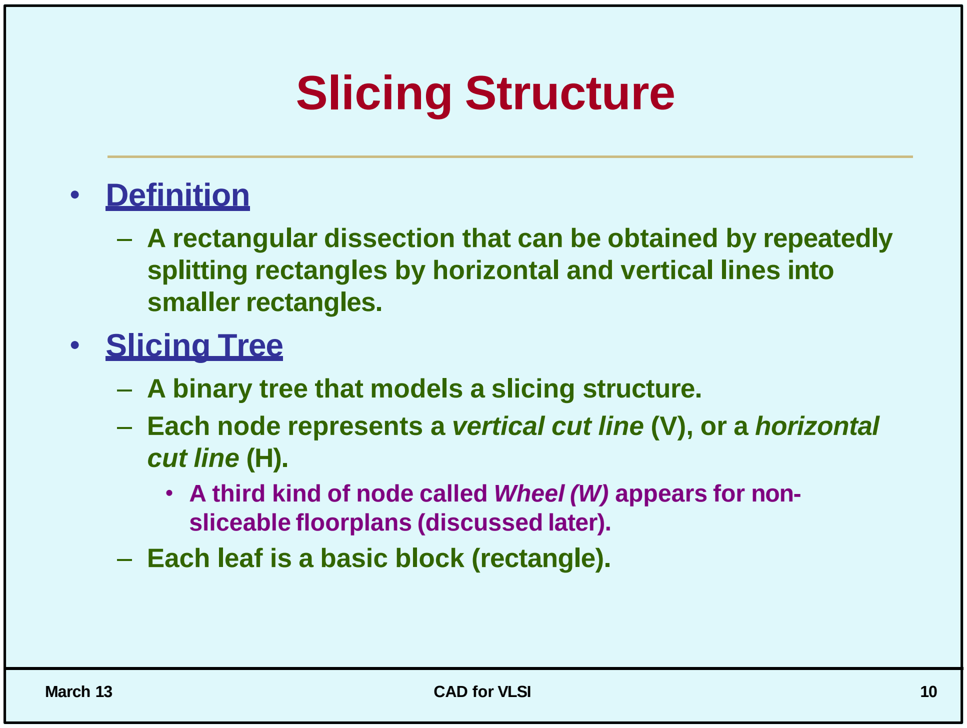


Contd.

- Typical cost function used:

  Cost = w1 * A + w2 * L

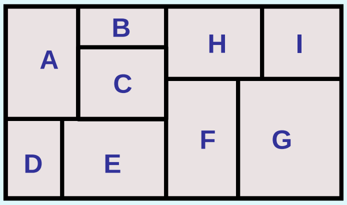
  where w1 and w2 are user-specified parameters.

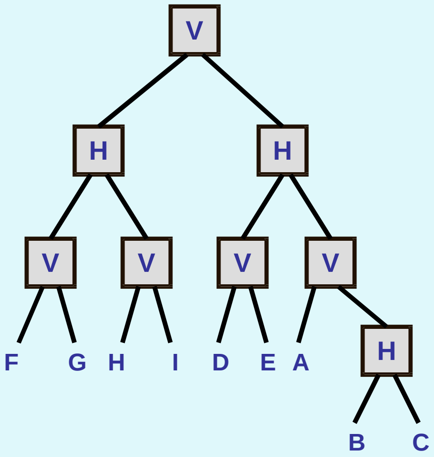


Slicing Structure

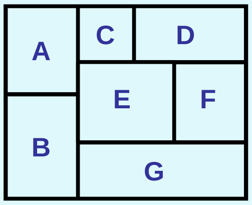
- **Definition**
  - A rectangular dissection that can be obtained by repeatedly splitting rectangles by horizontal and vertical lines into smaller rectangles.
- **Slicing Tree**
  - A binary tree that models a slicing structure.
  - Each node represents a *vertical cut line* (V), or a *horizontal cut line* (H).
    - A third kind of node called *Wheel (W)* appears for non-sliceable floorplans (discussed later).
  - Each leaf is a basic block (rectangle).

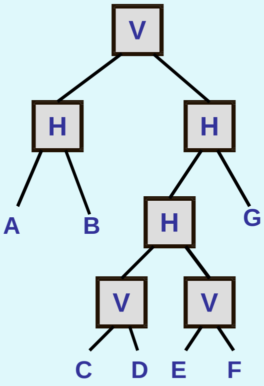
A Slicing Floorplan and its Slicing Tree

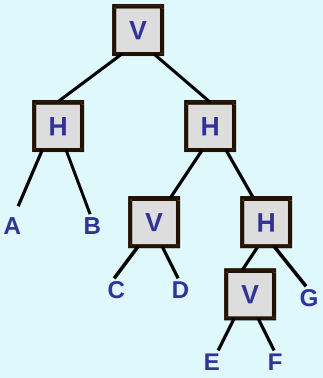




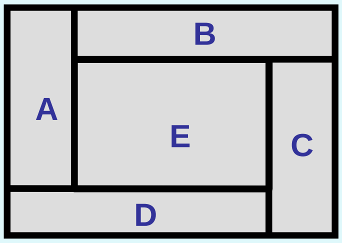
Slicing Tree is not Unique



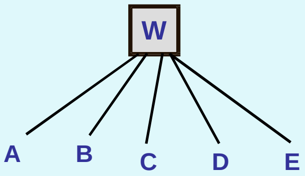




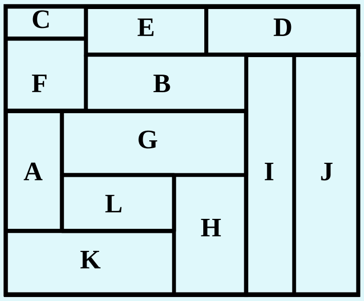
A Non-Slicing Floorplan



Also called "WHEEL"



A Hierarchical Floorplan



The dissection tree will contain "wheel".

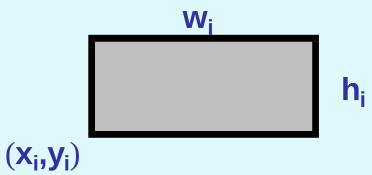
Floorplanning Algorithms

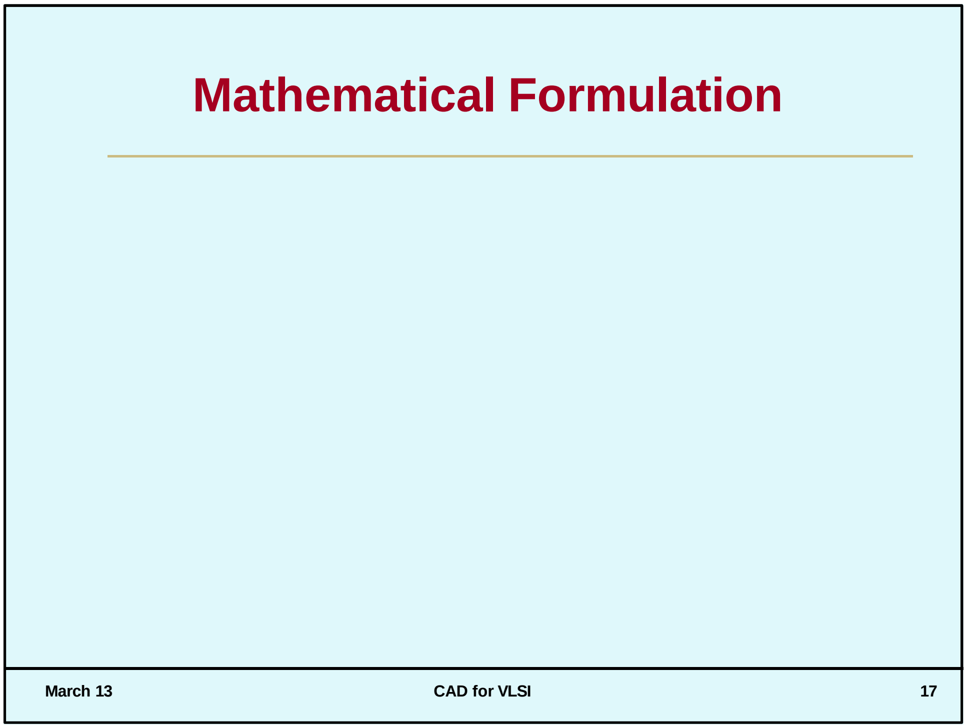
- Several broad classes of algorithms:
  - Integer programming based
  - Rectangular dual graph based
  - Hierarchical tree based
  - Simulated annealing based
  - Other variations

Integer Linear Programming Formulation

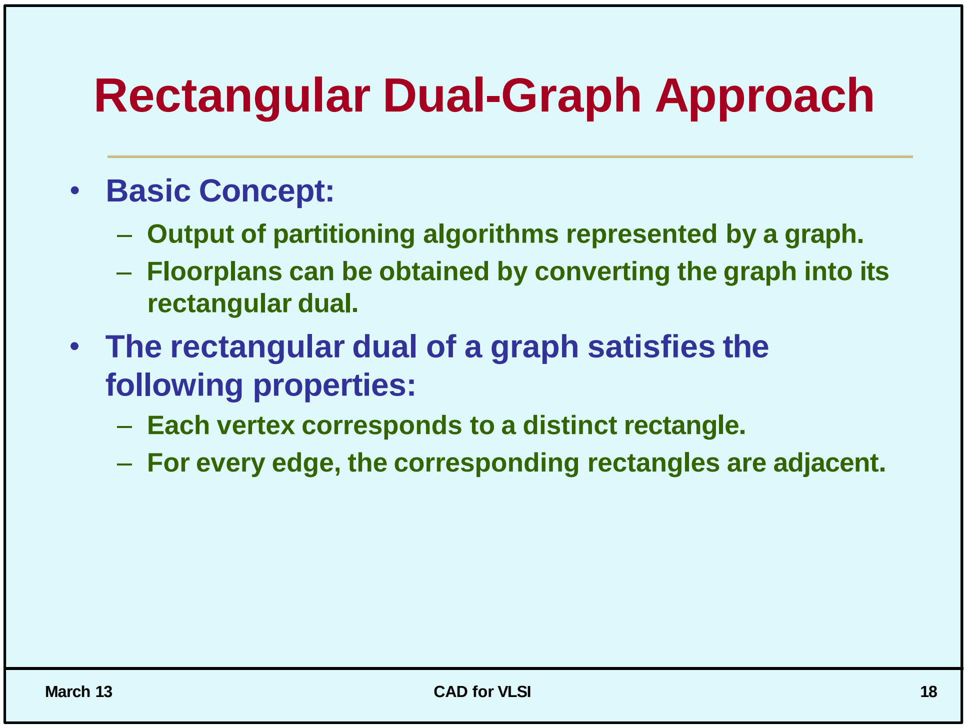
- The problem is modeled as a set of linear equations using 0/1 integer variables.
- Given:
  - Set of *n* blocks S = {B_1, B_2, …,B_n} which are rigid and have fixed orientation.
  - 4-tuple associated with each block

    (x_i, y_i, w_i, h_i)



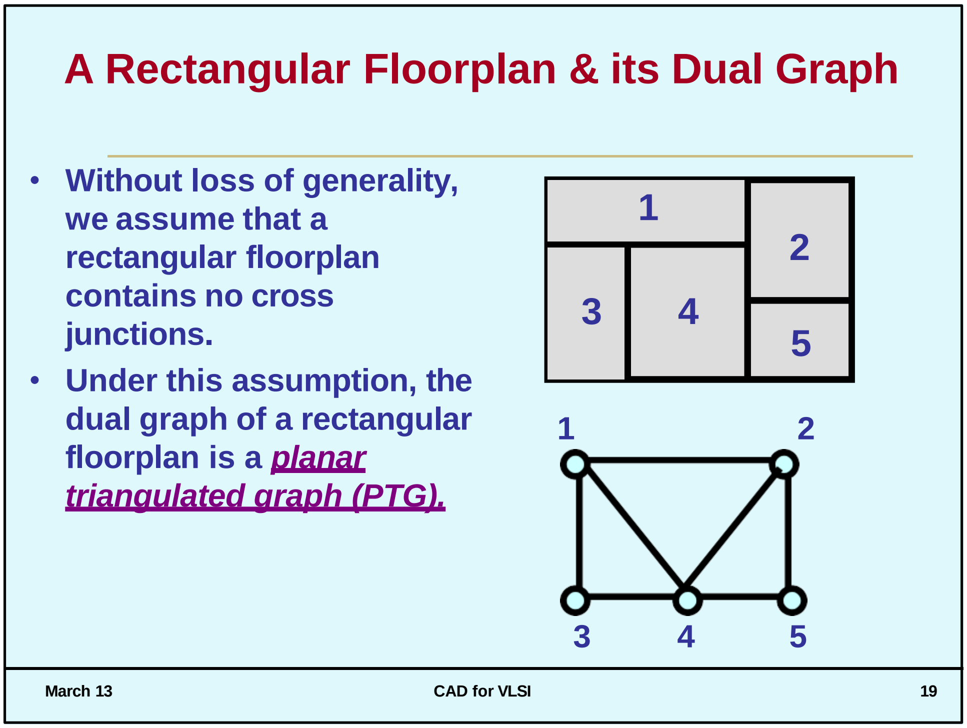


Mathematical Formulation



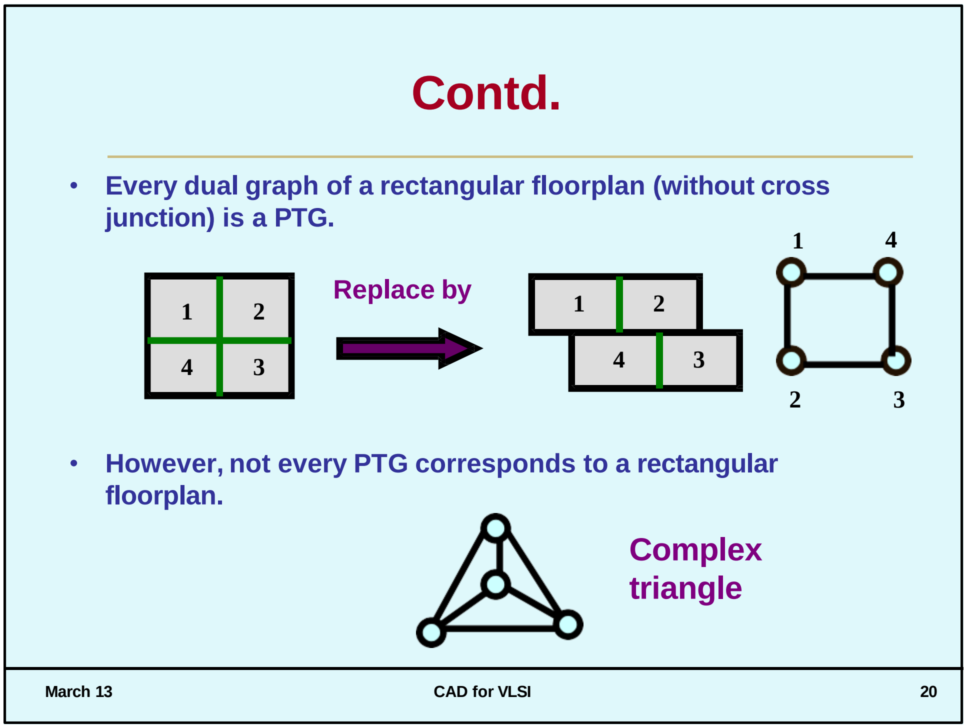
Rectangular Dual-Graph Approach

- Basic Concept:
  - Output of partitioning algorithms represented by a graph.
  - Floorplans can be obtained by converting the graph into its rectangular dual.
- The rectangular dual of a graph satisfies the following properties:
  - Each vertex corresponds to a distinct rectangle.
  - For every edge, the corresponding rectangles are adjacent.



A Rectangular Floorplan & its Dual Graph

- Without loss of generality, we assume that a rectangular floorplan contains no cross junctions.
- Under this assumption, the dual graph of a rectangular floorplan is a ***planar triangulated graph (PTG).***



Contd.

- Every dual graph of a rectangular floorplan (without cross junction) is a PTG.

- However, not every PTG corresponds to a rectangular floorplan.

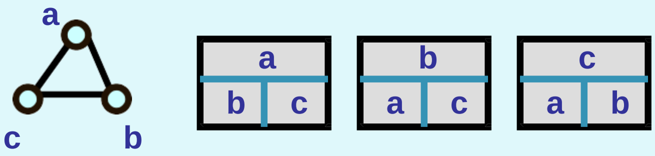
Complex triangle

Drawbacks

- A new approach to floorplanning, in which many sub-problems are still unsolved.
- The main problem concerns the existence of the rectangular dual, i.e. the elimination of complex triangles.
  - Select a minimum set E of edges such that each complex triangle has at least one edge in E.
  - A vertex can be added to each edge of E to eliminate all complex triangles.
  - The weighted complex triangle elimination problem has been shown to be NP-complete.
    - Some heuristics are available.

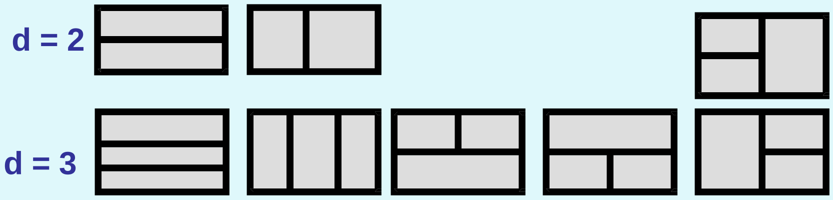
Hierarchical Approach

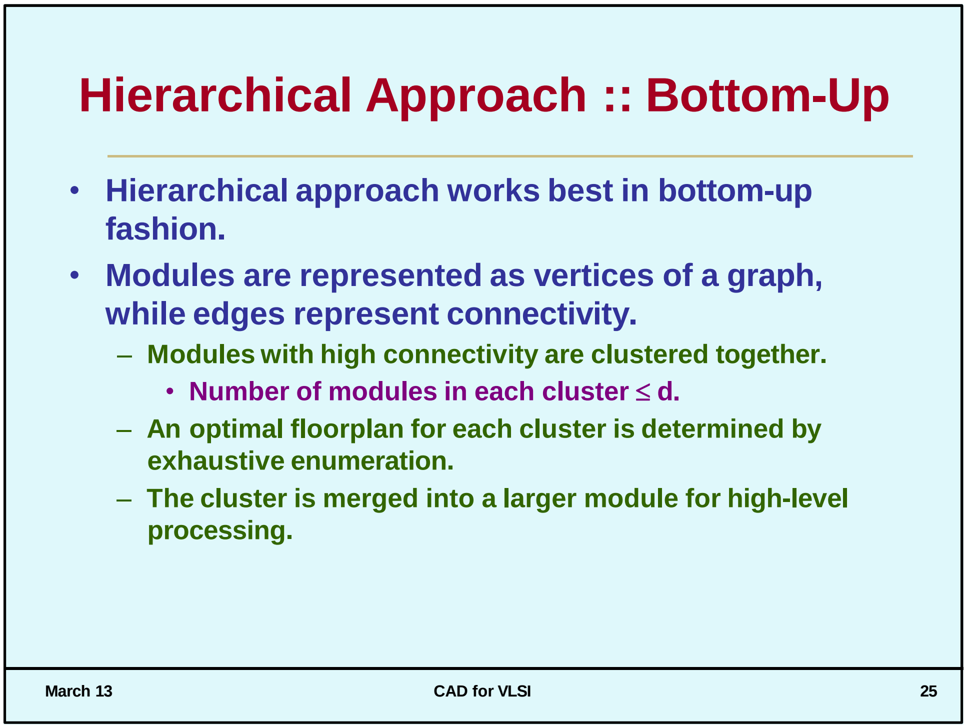
- Widely used approach to floorplanning.
  - Based on a divide-and-conquer paradigm.
  - At each level of the hierarchy, only a small number of rectangles are considered.
- A small graph, and all possible floorplans.



- After an optimal configuration for the three modules has been determined, they are merged into a larger module.
- The vertices ‘a’, ‘b’, ‘c’ are merged into a super vertex at the next level.

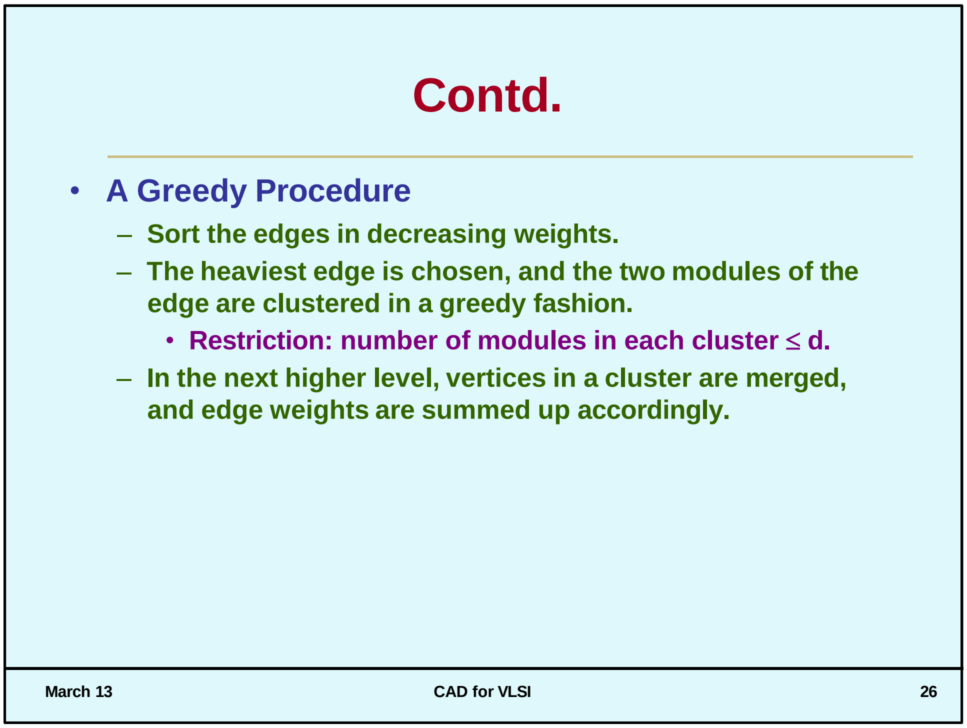
- The number of floorplans increases exponentially with the number of modules ‘d’ considered at each level.
  - ‘d’ is thus limited to a small number (typically d < 6).
- All possible floorplans for:





Hierarchical Approach :: Bottom-Up

- Hierarchical approach works best in bottom-up fashion.
- Modules are represented as vertices of a graph, while edges represent connectivity.
  - Modules with high connectivity are clustered together.
    - Number of modules in each cluster ≤ d.
  - An optimal floorplan for each cluster is determined by exhaustive enumeration.
  - The cluster is merged into a larger module for high-level processing.



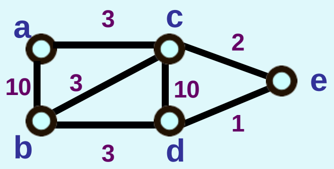
Contd.

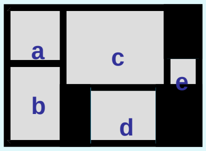
- A Greedy Procedure
  - Sort the edges in decreasing weights.
  - The heaviest edge is chosen, and the two modules of the edge are clustered in a greedy fashion.
    - Restriction: number of modules in each cluster ≤ d.
  - In the next higher level, vertices in a cluster are merged, and edge weights are summed up accordingly.

Contd.

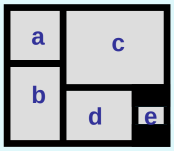
- Problem
  - Some lightweight edges may be chosen at higher levels in the hierarchy, resulting in adjacency of two clusters of highly incompatible areas.
- Possible solution
  - Arbitrarily assign a small cluster to a neighboring cluster when their sizes will be too small for processing at a higher level of the hierarchy.

Example





Greedy clustering

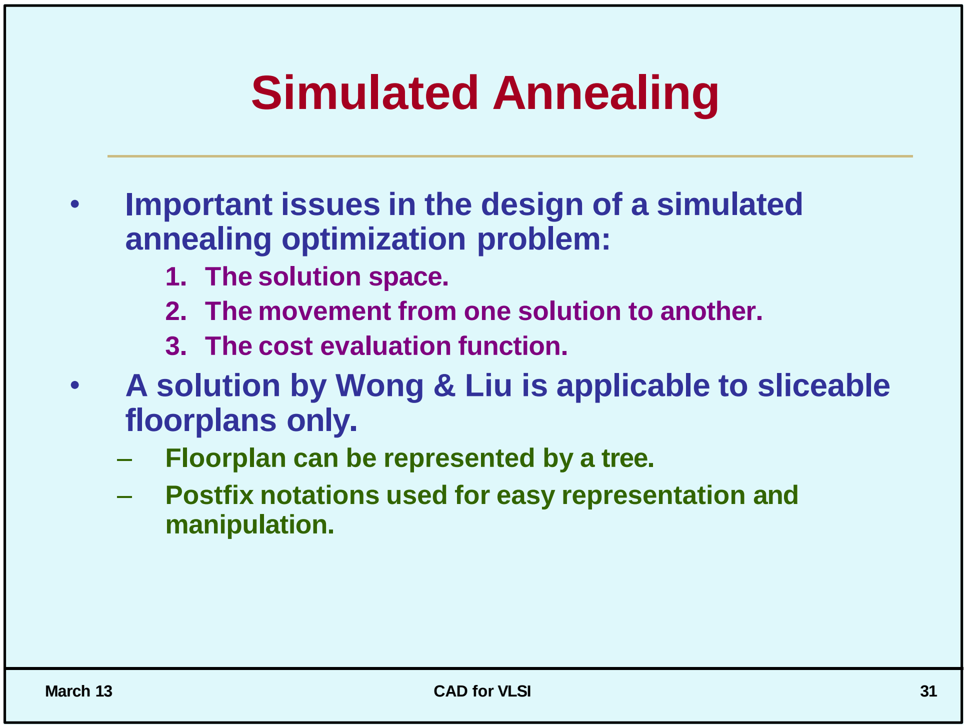


Merging small clusters

Hierarchical Approach :: Top-Down

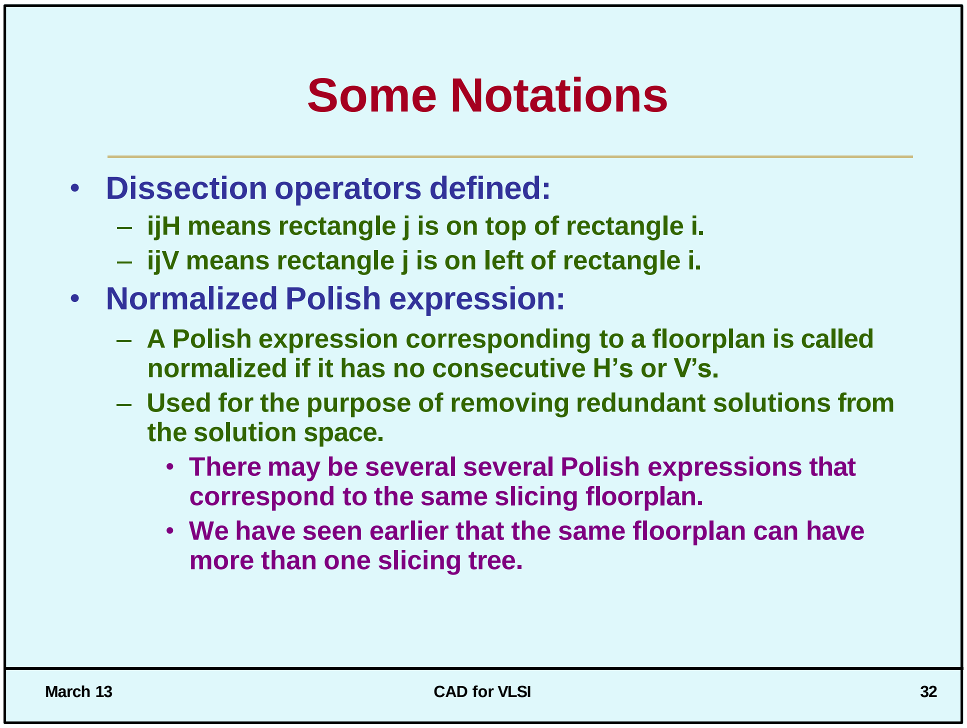
- The fundamental step is the partitioning of modules.
  - Each partition is assigned to a child floorplan.
  - Partitioning is recursively applied to the child floorplans.
- Major issue here is to obtain balanced graph partitioning.
  - k-way partitioning, in general.
- Not very widely used due to the difficulty of obtaining balanced partitions.

- One can combine top-down and bottom-up approaches.
  - Apply bottom-up technique to obtain a set of clusters.
  - Apply top-down approach to these clusters.



Simulated Annealing

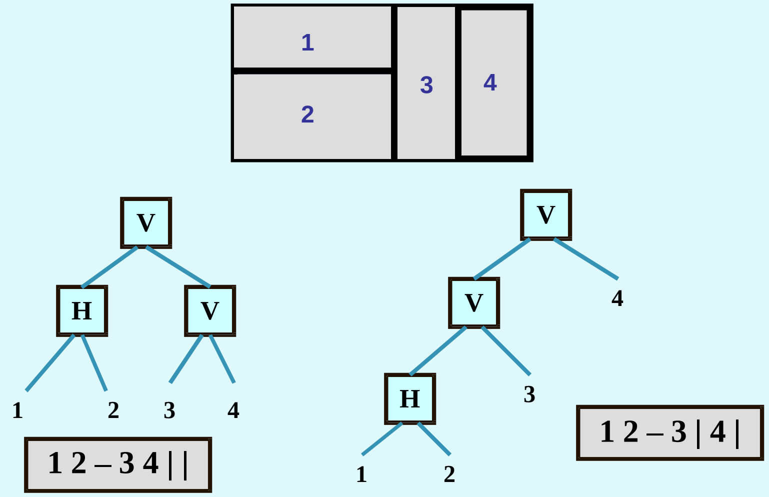
- Important issues in the design of a simulated annealing optimization problem:
  1. The solution space.
  2. The movement from one solution to another.
  3. The cost evaluation function.
- A solution by Wong & Liu is applicable to sliceable floorplans only.
  - Floorplan can be represented by a tree.
  - Postfix notations used for easy representation and manipulation.



Some Notations

- Dissection operators defined:
  - ijH means rectangle j is on top of rectangle i.
  - ijV means rectangle j is on left of rectangle i.
- Normalized Polish expression:
  - A Polish expression corresponding to a floorplan is called normalized if it has no consecutive H's or V's.
  - Used for the purpose of removing redundant solutions from the solution space.
    - There may be several several Polish expressions that correspond to the same slicing floorplan.
    - We have seen earlier that the same floorplan can have more than one slicing tree.

Example: Two tree representations of a floorplan

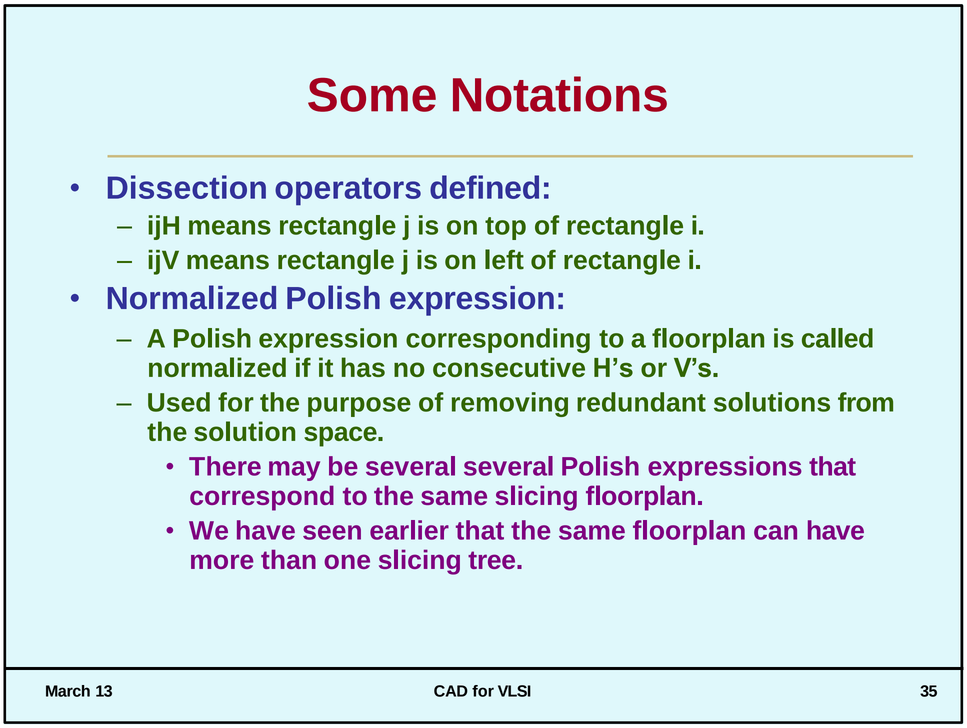


1 2 – 3 4 | |

1 2 – 3 | 4 |

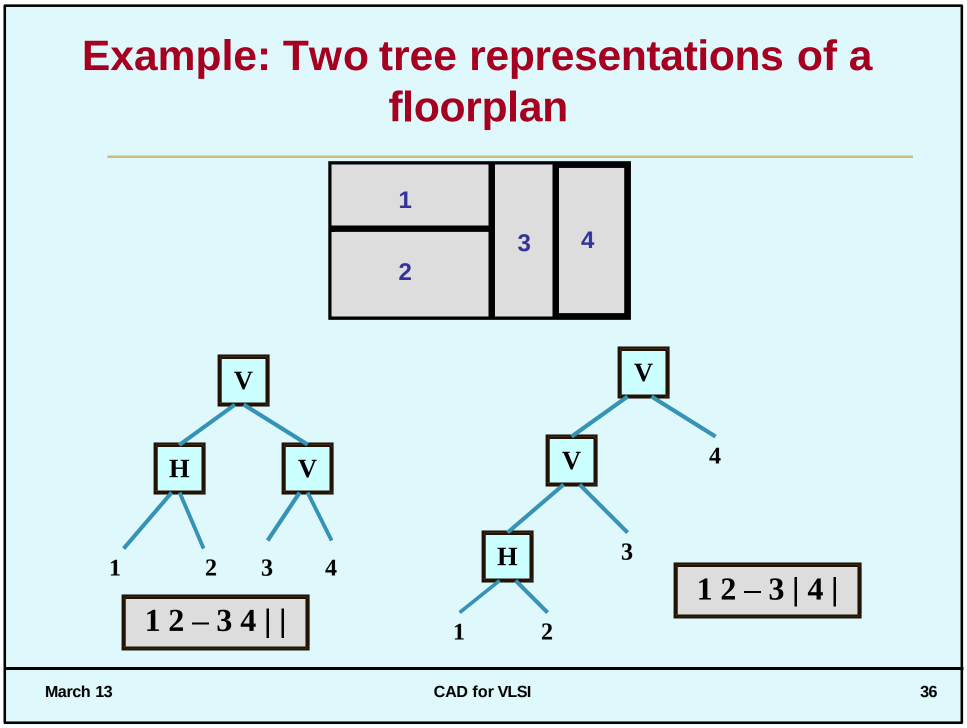
Simulated Annealing

- Important issues in the design of a simulated annealing optimization problem:
  1. The solution space.
  2. The movement from one solution to another.
  3. The cost evaluation function.
- A solution by Wong & Liu is applicable to sliceable floorplans only.
  – Floorplan can be represented by a tree.
  – Postfix notations used for easy representation and manipulation.



Some Notations

- Dissection operators defined:
  - ijH means rectangle j is on top of rectangle i.
  - ijV means rectangle j is on left of rectangle i.
- Normalized Polish expression:
  - A Polish expression corresponding to a floorplan is called normalized if it has no consecutive H’s or V’s.
  - Used for the purpose of removing redundant solutions from the solution space.
    - There may be several several Polish expressions that correspond to the same slicing floorplan.
    - We have seen earlier that the same floorplan can have more than one slicing tree.



Example: Two tree representations of a floorplan

1 2 – 3 4 | |

1 2 – 3 | 4 |

Parameters of the Algorithm

- Solution Perturbations (Move)
  - a) Swap two adjacent operands
  - b) Complement a series of operators between two operands (called a *chain*).

    $\bar{V} = H$ and $\bar{H} = V$
  - c) Swap two adjacent operand and operator.
- We accept a move only it is results in a normalized expression.
  - – Only move “c” may result in a non-normalized solution.
  - – We need to check only when move “c” is applied.

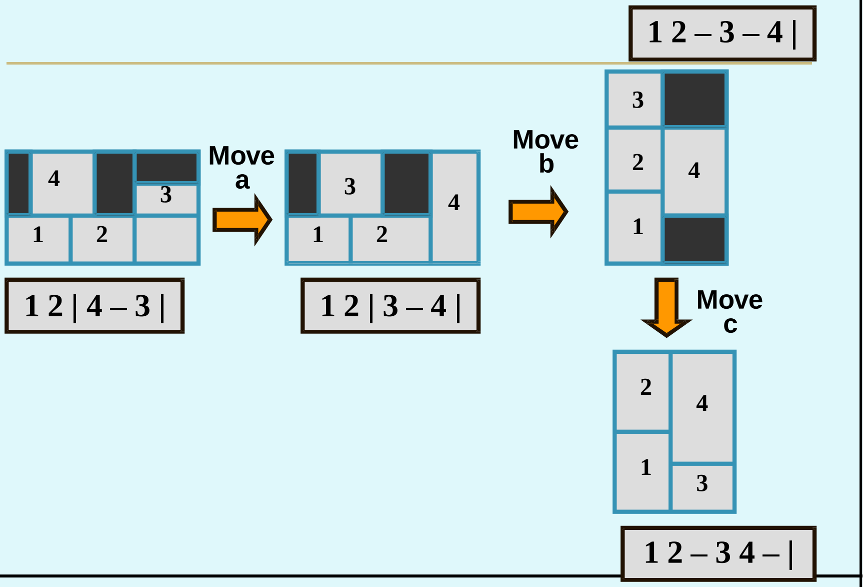
Contd.

- Cost Function:
  - – Typical cost function used:

$$\text{Cost} = w_1 * A + w_2 * L$$

where A is the area of the smallest rectangle
enveloping the given basic rectangles,
L is the overall interconnection length,
w_1 and w_2 are user-specified parameters.

An Example



1 2 | 4 – 3 |

Move a

1 2 | 3 – 4 |

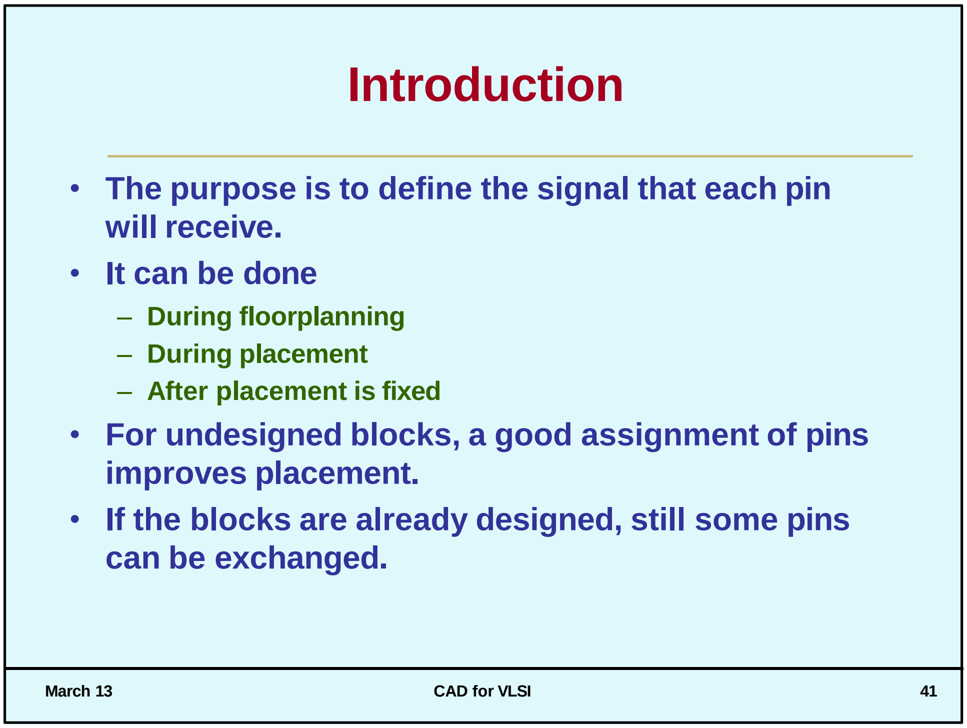
Move b

1 2 – 3 – 4 |

Move c

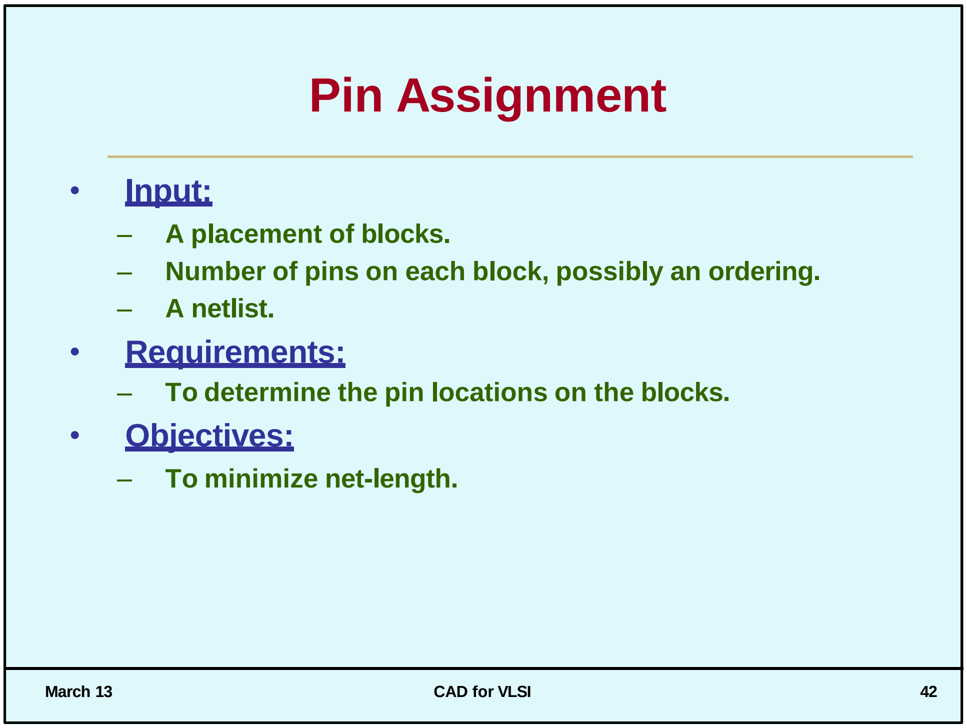
1 2 – 3 4 – |

Pin Assignment



Introduction

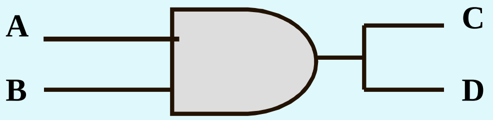
- The purpose is to define the signal that each pin will receive.
- It can be done
  - During floorplanning
  - During placement
  - After placement is fixed
- For undesigned blocks, a good assignment of pins improves placement.
- If the blocks are already designed, still some pins can be exchanged.



Pin Assignment

- Input:
  - A placement of blocks.
  - Number of pins on each block, possibly an ordering.
  - A netlist.
- Requirements:
  - To determine the pin locations on the blocks.
- Objectives:
  - To minimize net-length.

- **Functionally equivalent pins**:
  - Exchanging the signals does not affect the circuit.
- **Equipotential pins**:
  - Both are internally connected and represent the same net.

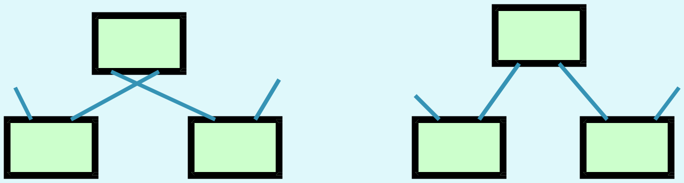


A,B :: functionally equivalent

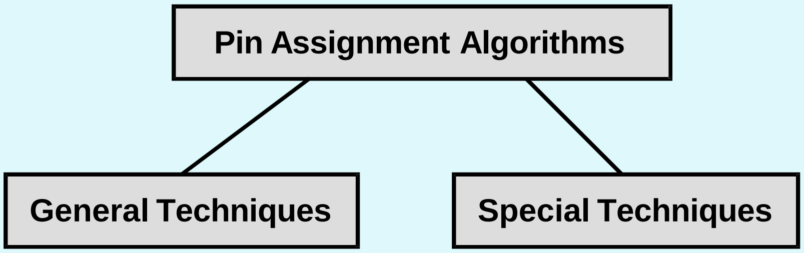
C,D :: equipotential

Problem Formulation

- Purpose is to optimize the assignment of nets within a functionally equivalent (or equipotential) pin groups.
- **Objective**:
  - To reduce congestion or reduce the number of crossovers.



Classification of Algorithms



Pin Assignment Algorithms

General Techniques

Special Techniques

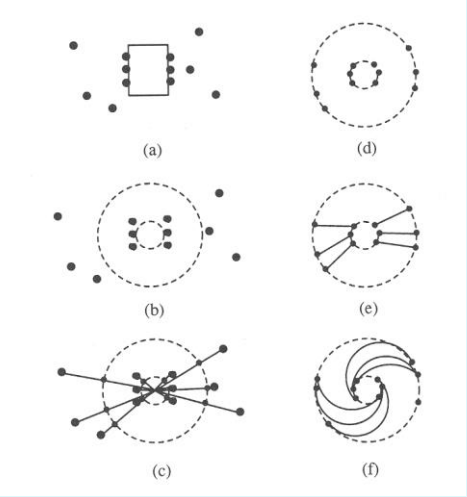
Concentric Circle Mapping

Topological Method

Nine Zone Method

Channel Pin Assignment

Concentric Circle Mapping



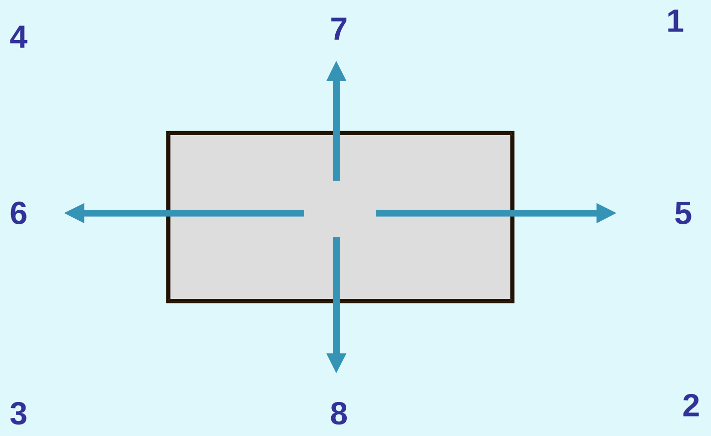
Topological Pin Assignment

- Similar to concentric circle mapping.
- Easier to complete pin assignment.
  - When there is interference from other components and barriers.
  - For nets connected to more than two pins.
- If a net has been assigned to more than two pins, then the pin closest to the center of the primary component is chosen.
- Pins of primary component are mapped onto a circle as before.
- Beginning at the bottom of the circle, and moving clockwise, the pins are assigned to nets.

Nine Zone Method

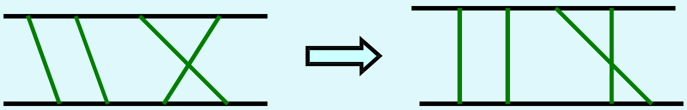
- Based on zones in a Cartesian co-ordinate system.
- The center of the co-ordinate system is located inside a group of interchangeable pins on a component.
  - This component is called *pin class*.
- A net rectangle is defined by each of the nets connected to the pin class.
  - There are nine zones in which this rectangle can be positioned.

The Nine Pin Zones



Channel Pin Assignment

- A significant portion of the chip area is used for channel routing.
  - After the placement phase, the position of terminals on the boundaries of a block are not fixed.
  - They may be moved before routing begins.



- Yang & Wong proposed a dynamic programming formulation to the channel pin assignment problem.

Integrated Approach

- Better understanding of the different stages in physical design automation over the years.
  - Attempts are being made to merge some steps of the design cycle.
  - For example, floorplanning and placement are considered together.
  - Sometimes, placement and routing stages can also be combined together.
- Still a problem of research.