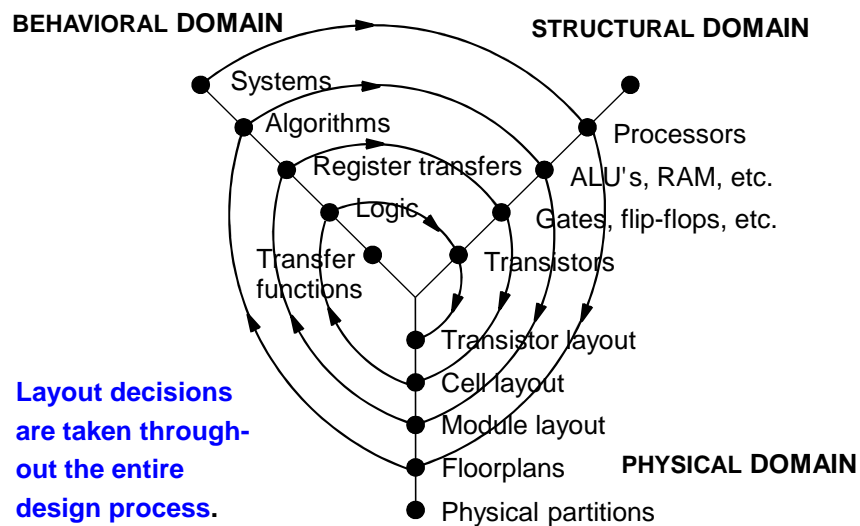


Chapter 8 Floorplanning

Floorplanning based Design Methodology



Floorplanning vs Placement

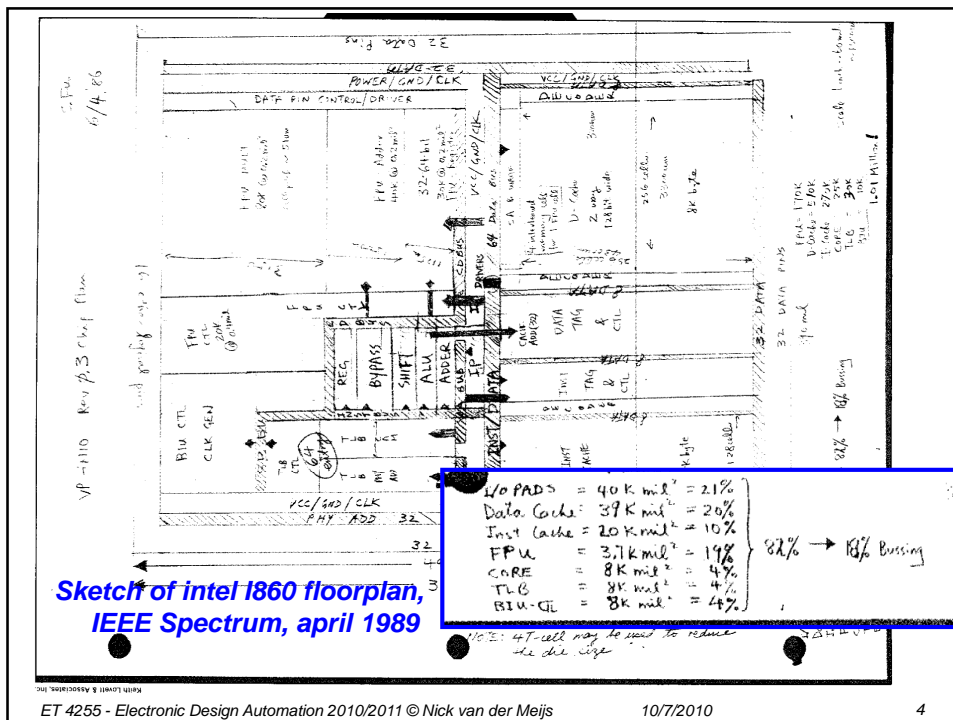
Q: which is more difficult?

Placement

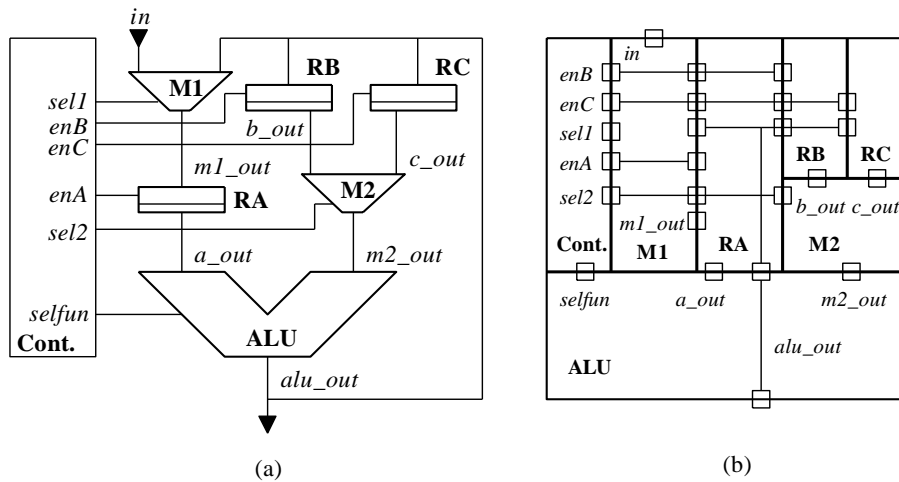
- Assignment of (legal and) optimal module locations
- Given
- Collection of modules with **fixed** dimensions, pin positions, delay, power
 - Netlist
 - Geometric constraints
 - Electrical constraints

Floorplanning

- Assignment of (legal and) optimal module locations
- Given
- Collection of modules with **variable** dimensions, pin positions, delay, power
 - Netlist
 - Geometric constraints
 - Electrical constraints
 - **Estimates** of various properties



Early Layout Decision Example



ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

5

The Floorplan-Based Design Methodology

- An integrated circuit is essentially a **two-dimensional medium**; taking this aspect into account in early stages of the design helps in creating designs of good quality.
- Floorplanning gives early feedback: thinking of layout at early stages may suggest valuable architectural modifications; floorplanning also aids in estimating delay due to wiring.
- Floorplanning fits very well in a **top-down** design strategy, the **stepwise refinement** strategy also propagated in software design.
- Floorplanning assumes, however, **flexibility** in layout design, the existence of cells that can adapt their shapes and terminal locations to the environment.

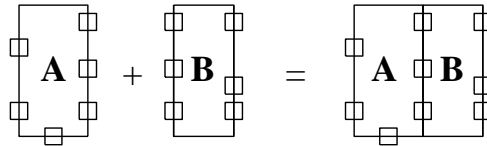
ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

6

Floorplanning Concepts

- **Abutment:** establishing connections between cells by putting them directly next to each other, without the necessity of routing.



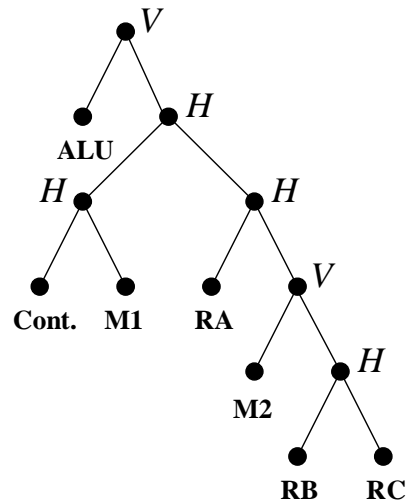
- **Leaf cell:** a cell at the lowest level of the hierarchy; it does not contain any other cell.
 - **Composite cell:** a cell that is composed of either leaf cells or composite cells. The entire IC is the highest-level composite cell.
- Restriction:** all leaf cells and composite cells are supposed to be rectangular.

Floorplan Representation

- Need to represent a floorplan in a datastructure suitable for algorithmic manipulations
- Different algorithms and floorplan restrictions require different representations
- Some representations are natural, some are intricate

Slicing Floorplan + Slicing Tree

- A composite cell's subcells are obtained by a horizontal or vertical **bisection** of the composite cell.
- Slicing floorplans can be represented by a **slicing tree**.
- In a slicing tree, all cells (except for the top-level cell) have a **parent**, and all composite cells have **children**.
- A slicing floorplan is also called a floorplan of **order 2** (allow 2-ary tree representation).

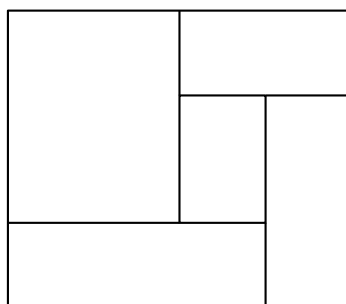


ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

9

Wheel or Spiral Floorplan



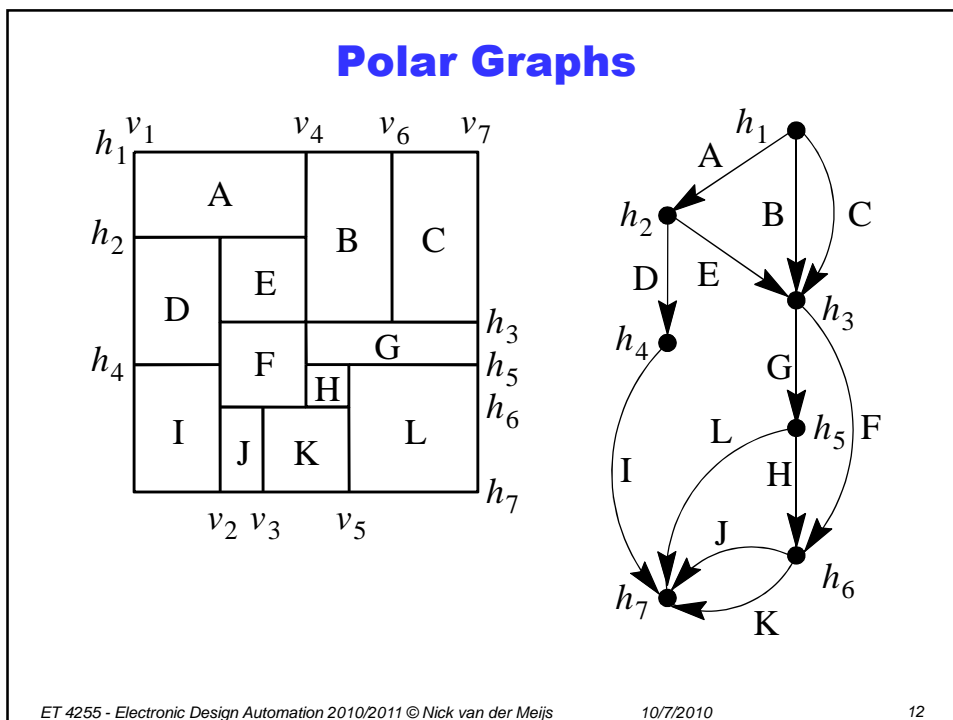
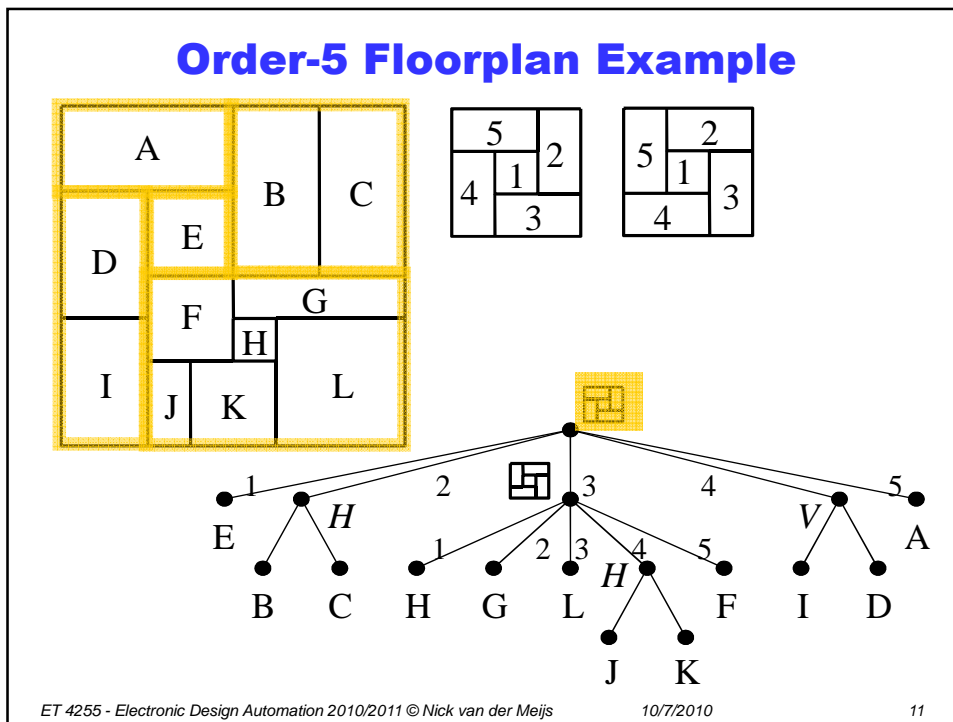
- This floorplan is not slicing!

- Limiting floorplans to those that have the slicing property is reasonable: it certainly facilitates floorplanning algorithms.
- Taking the shape of a wheel floorplan and its mirror image as the basis of operators leads to hierarchical descriptions of **order 5**.

ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

10



Optimization Problems

- **Floorplan generation:** this is the problem of e.g. constructing the slicing tree, given a structural description. It can be done by adapting methods known from placement such as **min-cut partitioning**.
- ☞ ■ **Floorplan sizing:** optimization of the floorplan area taking advantage of the flexibility in the cells.
- **Generation of flexible cells:** the generation of layouts once that the cell shapes have been fixed by the sizing procedure.

Floorplan Sizing

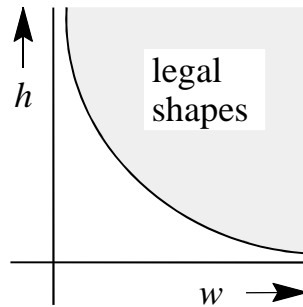
Of slicing floorplans

- This is a restriction, cannot represent all possible configurations, will (probably) miss the best configuration
- However, sizing of slicing floorplans can be done in polynomial time (while the general problem is NP hard)
- Thus, we pay a price in optimality, for which we gain the possibility to get some solution
- Historically, there were pro-slicing and anti-slicing camps

Shape Functions (1)

- Flexible cells imply that cells can assume different aspect ratios.
- One can e.g. assume that all implementations of the cell have the same area A . The relation between the width w and the height h is:

$$hw = A, \text{ or } h = \frac{A}{w}$$



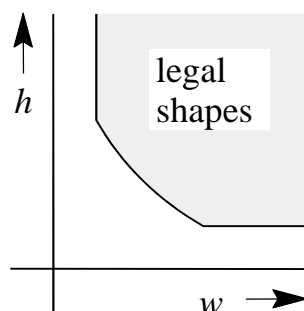
Shape function: set of (w,h) pairs in which the required functionality can be realized

Or: the amount of space need to reserve on the chip

The shape function is a hyperbola.

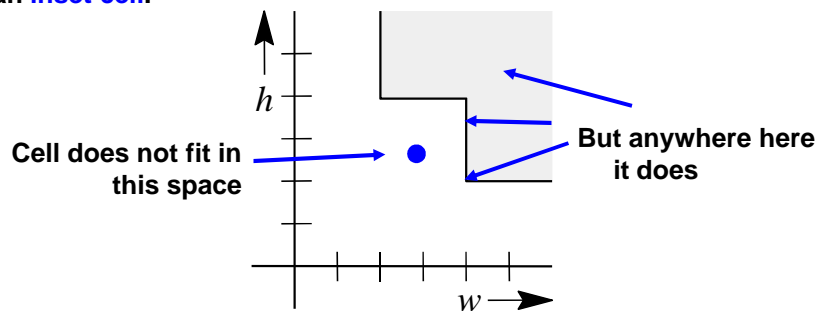
Shape Functions (2)

- Very thin cells are not interesting and often not feasible to design. The shape function is a combination of a hyperbola and two straight lines.



Shape Functions (3)

- Leaf cells are built from discrete transistors: it is not realistic to assume that the shape function follows the hyperbola continuously.
- In an extreme case, a cell is rigid: it can only be rotated and mirrored during floorplanning or placement. Such a cell is called an **inset cell**.



The shape function of a 2 x 4 inset cell.

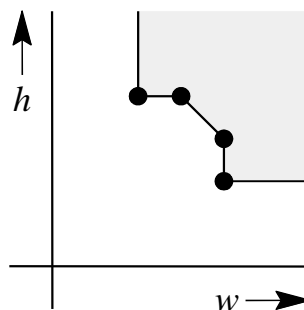
ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

17

Shape Functions (4)

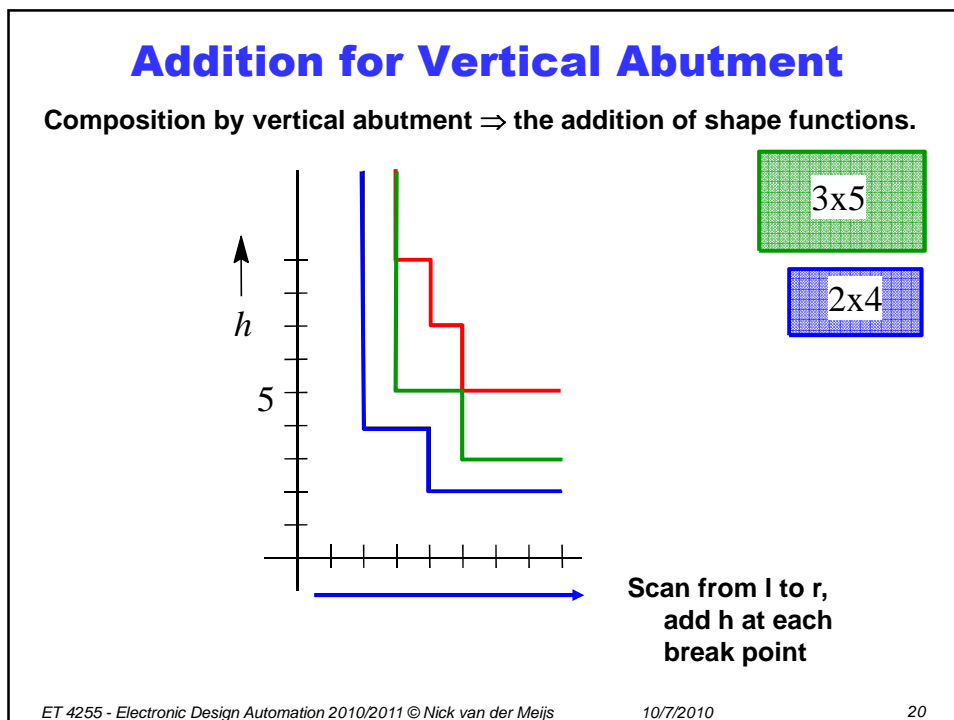
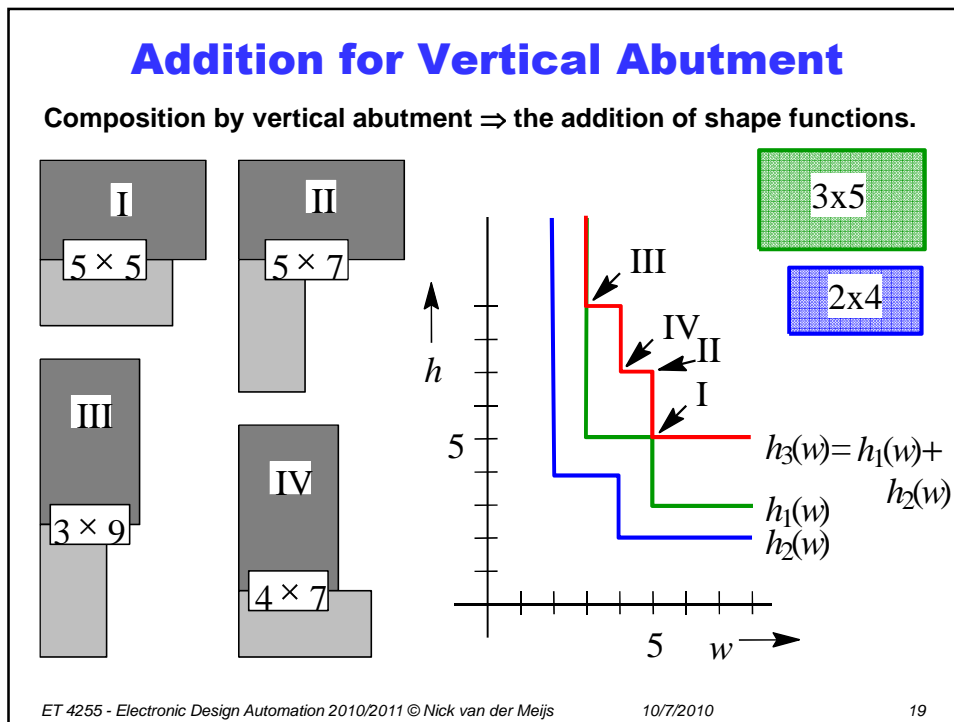
- In general, a **piecewise linear** function can be used to approximate any shape function.
- The points where the function changes its direction, are called the **break points** of the piecewise linear function.



ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

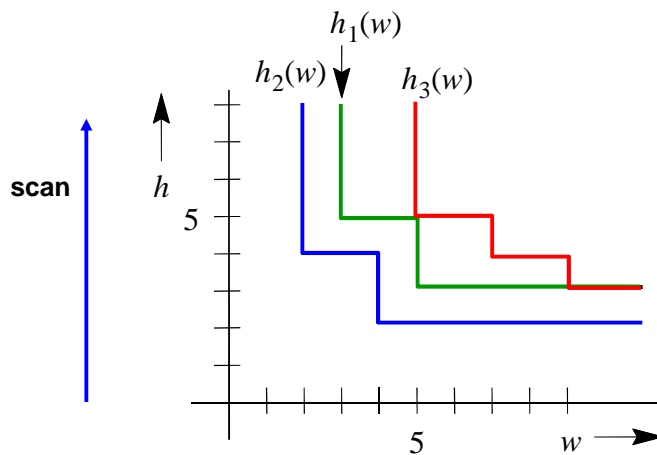
10/7/2010

18



Addition for Horizontal Abutment

- In case of horizontal abutment, the computation of the new shape function can be expressed as $h_3^{-1} = h_1^{-1} + h_2^{-1}$



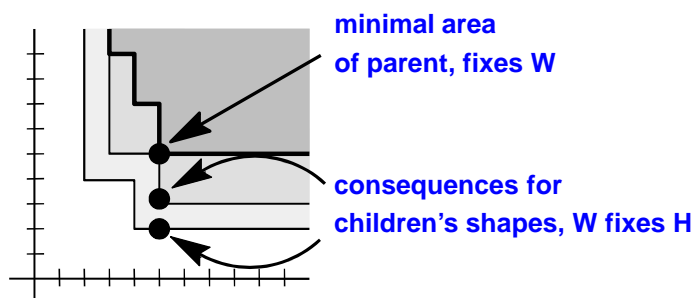
ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

21

Deriving Shapes of Children

- A choice for the minimal shape of the composite cell **recursively** fixes the shapes of its children cells.



ET 4255 - Electronic Design Automation 2010/2011 © Nick van der Meijs

10/7/2010

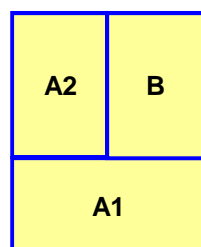
22

Sizing Algorithm for Slicing Floorplans

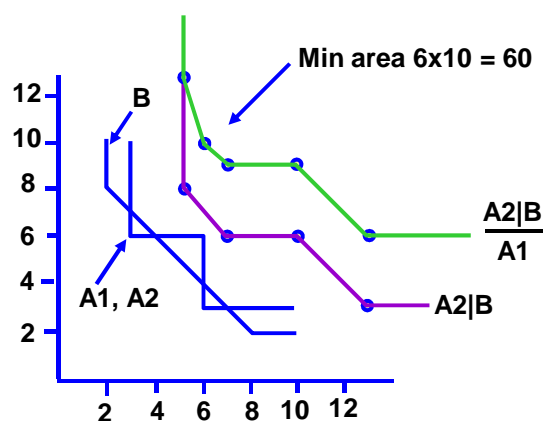
- The shape functions of all leaf cells are given as piecewise linear functions.
- Traverse the slicing tree in order to compute the shape functions of all composite cells (bottom-up composition).
- Choose the desired shape of the top-level cell; as the shape function is piecewise linear only the break points of the function need to be evaluated, when looking e.g. for minimal area.
- Propagate the consequences of the choice down to the leaf cells (top-down propagation).

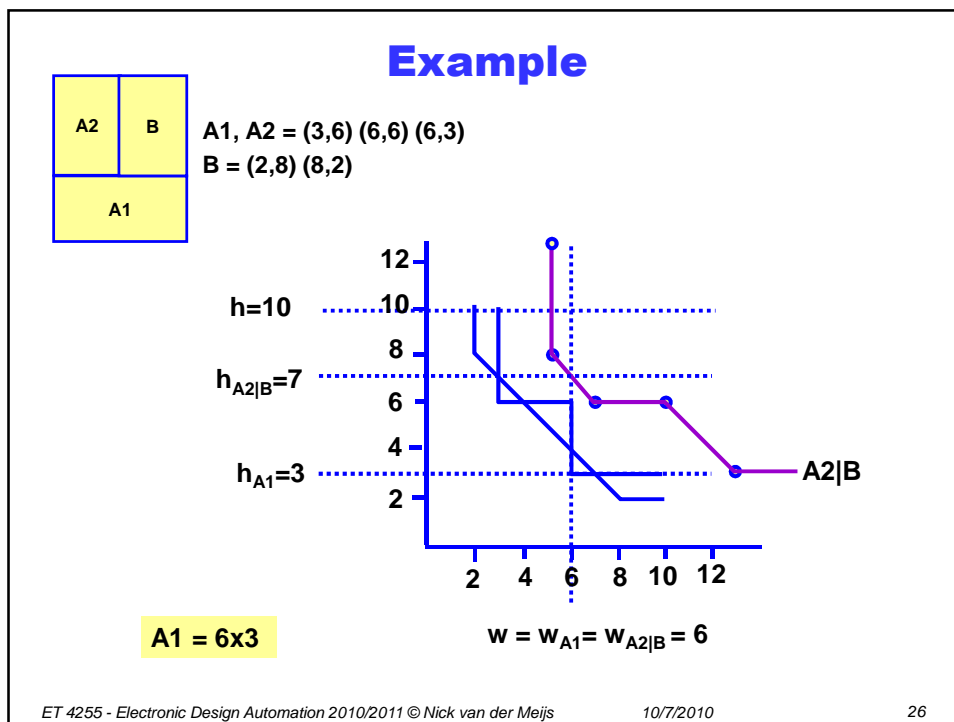
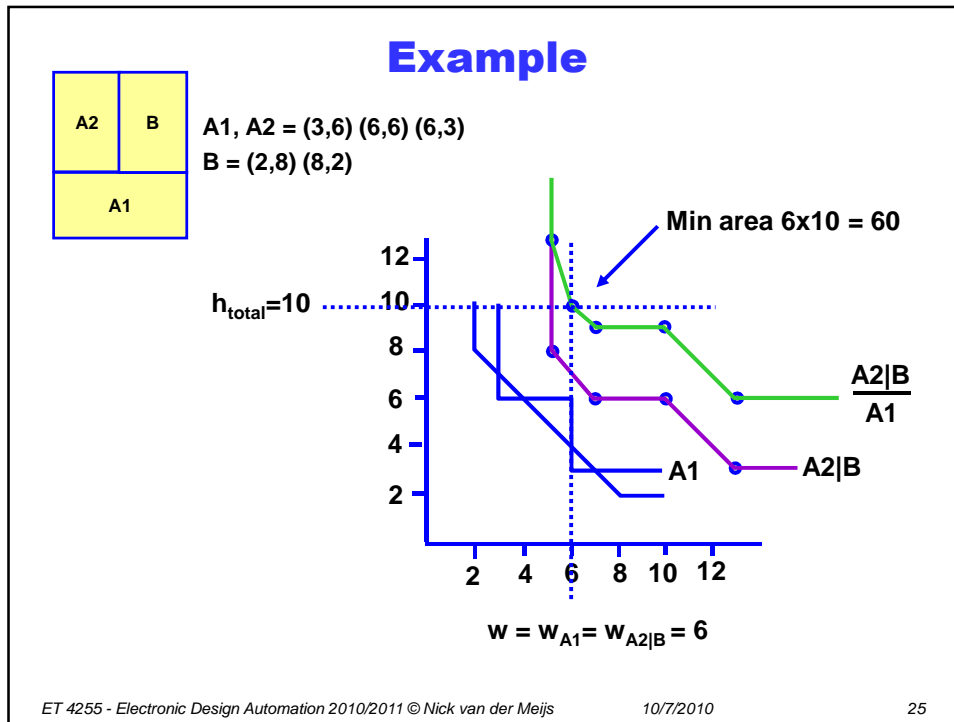
Note: the algorithm operates in polynomial time (but sizing of floorplans of order 5 is NP-complete).

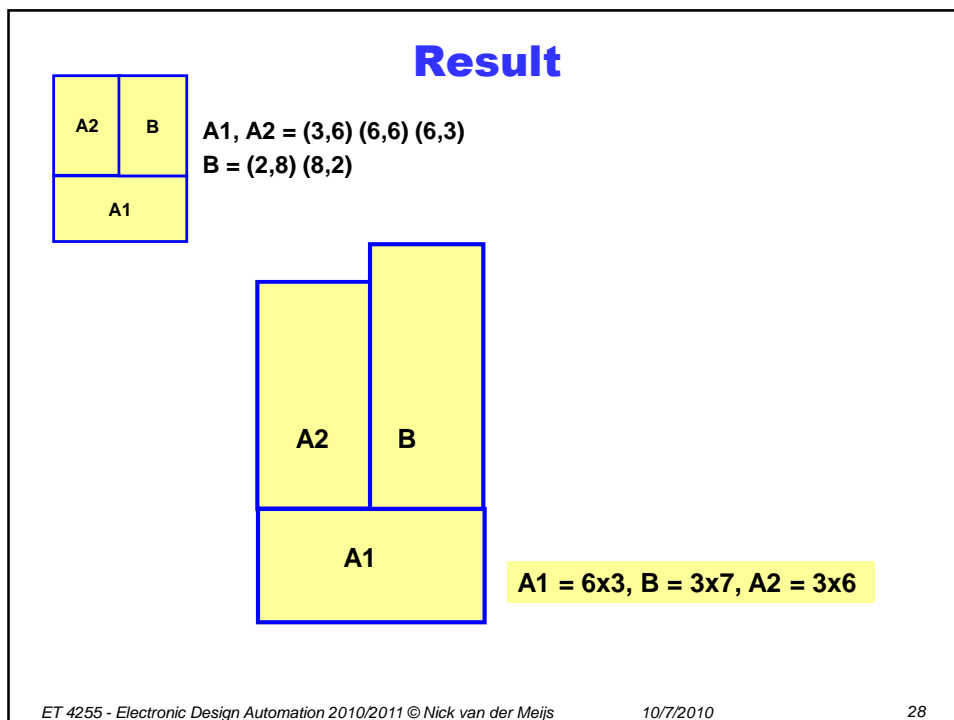
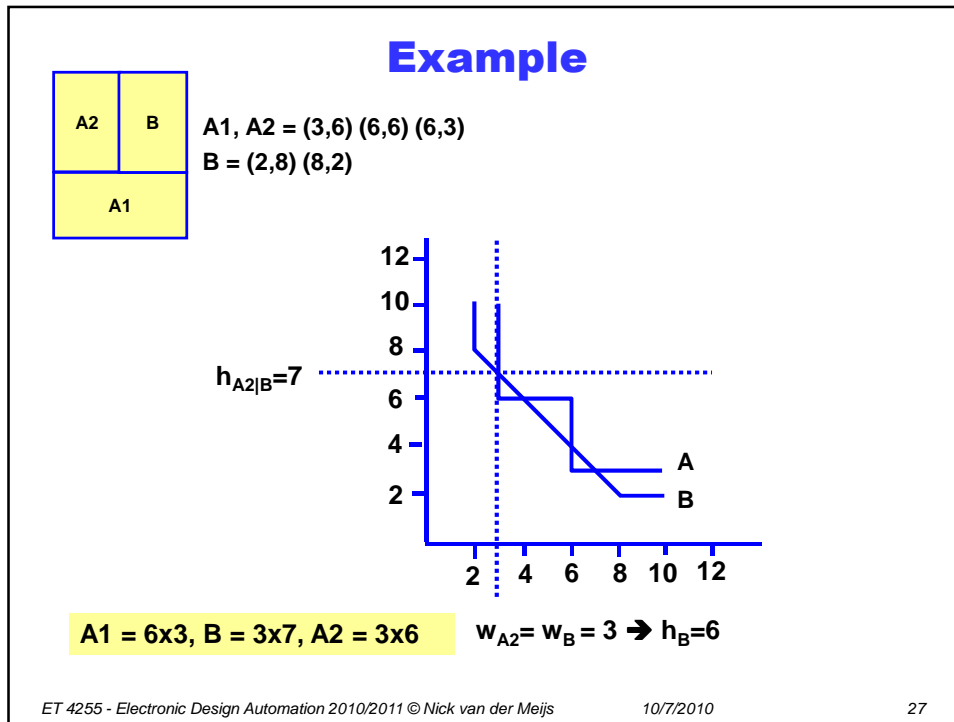
Example



A1, A2 = (3,6) (6,6) (6,3)
B = (2,8) (8,2)







Outlook

- Floorplanning is but a part of *Design Planning*
 - *An ounce of prevention is worth a pound of cure*
 - *Be smart from the start*

- Problems are called 'closure problems'
 - Timing closure, placement closure, power closure, X closure,
 - *Design Closure*
 - Iterations during design might not converge

- Floor Planning goes together with Wire Planning

- See Otten, Design Planning, in EDA for IC System Design, Verification and Testing, Scheffer (Ed.). Online resource at TU Delft Library