Global Routing

Basic Idea

- The routing problem is typically solved using a two-step approach:
  - Global Routing
    - Define the routing regions.
    - Generate a tentative route for each net.
    - Each net is assigned to a set of routing regions.
    - Does not specify the actual layout of wires.
  - Detailed Routing
    - For each routing region, each net passing through that region is assigned particular routing tracks.
    - Actual layout of wires gets fixed.
    - Associated subproblems: channel routing and switchbox routing.

Routing Regions

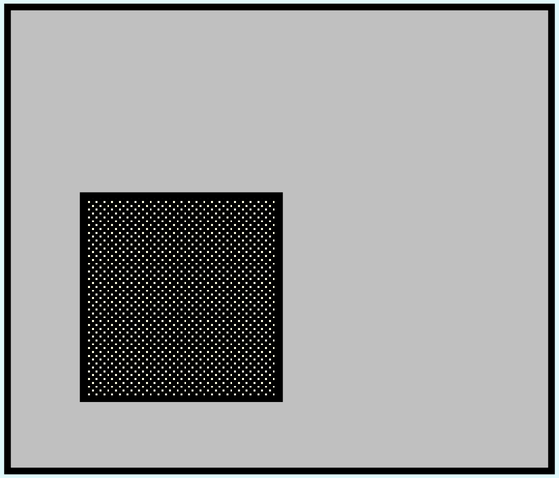
- Regions through which interconnecting wires are laid out.
- How to define these regions?
  - Partition the routing area into a set of non-intersecting rectangular regions.
  - Types of routing regions:
    - Horizontal channel: parallel to the x-axis with pins at their top and bottom boundaries.
    - Vertical channel: parallel to the y-axis with pins at their left and right boundaries.
    - Switchbox: rectangular regions with pins on all four sides.

- **Points to note:**
  - Identification of routing regions is a crucial first step to global routing.
  - Routing regions often do not have pre-fixed capacities.
  - The order in which the routing regions are considered during detailed routing plays a vital part in determining overall routing quality.

Types of Channel Junctions

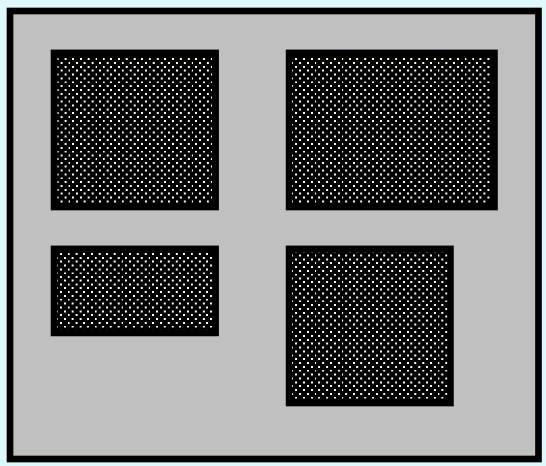
- Three types of channel junctions may occur:
  - **L-type**:
    - Occurs at the corners of the layout surface.
    - Ordering is not important during detailed routing.
    - Can be routed using channel routers.
  - **T-type**:
    - The leg of the “T” must be routed before the shoulder.
    - Can be routed using channel routers.
  - **+-type**:
    - More complex and requires switchbox routers.
    - Advantageous to convert +-junctions to T-junctions.

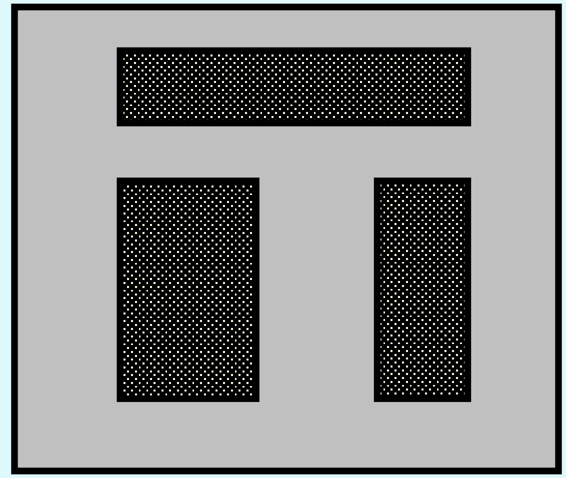
Illustrations



L Type

+ Type



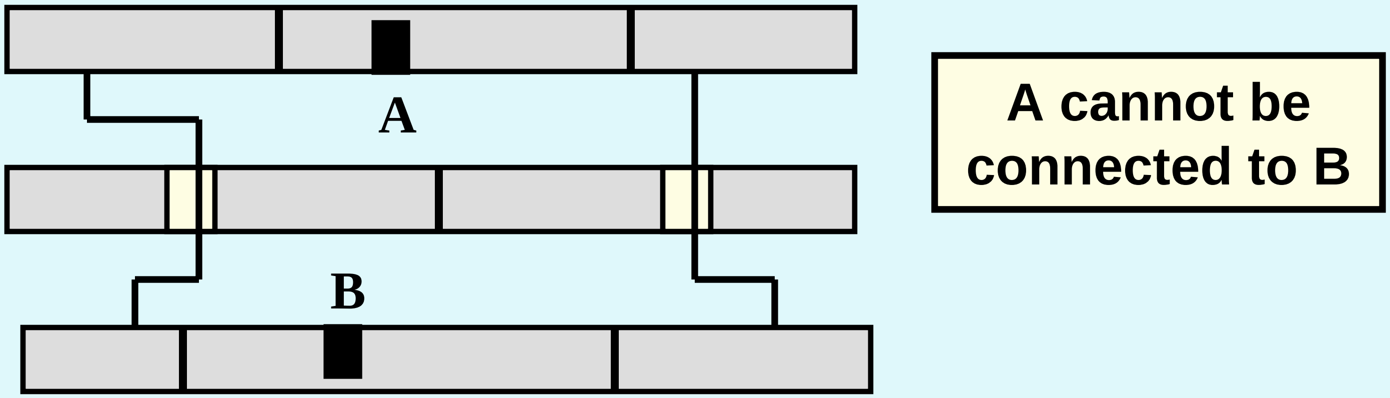


T Type

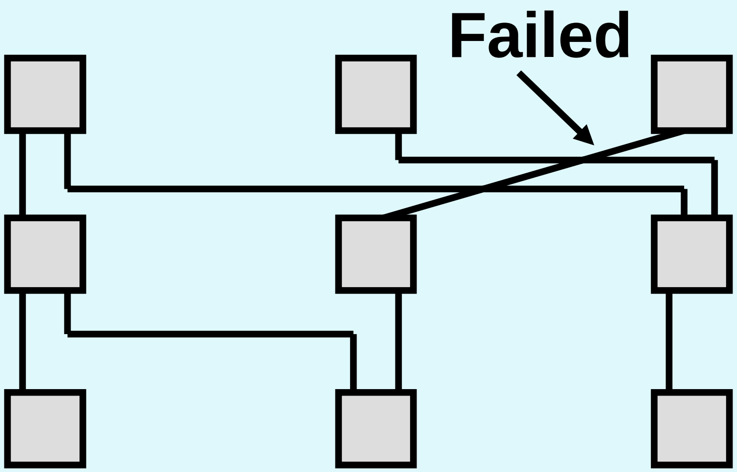
Design Style Specific Issues

- **Full Custom**
  - The problem formulation is similar to the general formulation as discussed.
    - All the types of routing regions and channels junctions can occur.
  - Since channels can be expanded, some violation of capacity constraints are allowed.
  - Major violation in constraints are, however, not allowed.
    - May need significant changes in placement.

- **Standard Cell**
  - At the end of the placement phase
    - Location of each cell in a row is fixed.
    - Capacity and location of each feed-through is fixed.
    - Feed-throughs have predetermined capacity.
  - Only horizontal channels exist.
    - Channel heights are not fixed.
  - Insufficient feed-throughs may lead to failure.
  - Over-the-cell routing can reduce channel height, and change the global routing problem.


A
B
A cannot be connected to B

- **Gate Array**
  - The size and location of cells are fixed.
  - Routing channels & their capacities are also fixed.
  - Primary objective of global routing is to guarantee routability.
  - Secondary objective may be to minimize critical path delay.



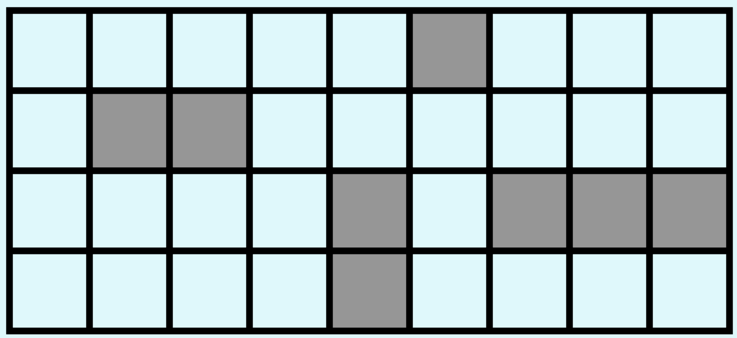
Graph Models used in Global Routing

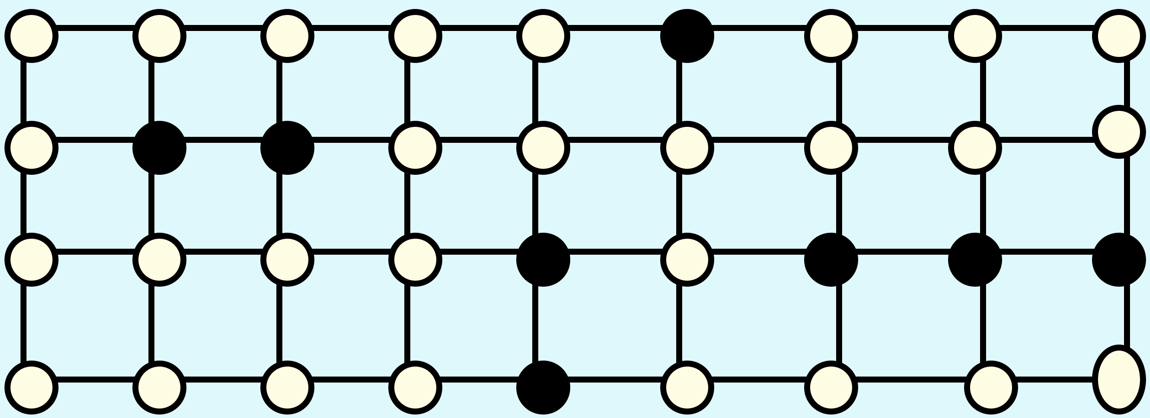
- **Global routing is typically studied as a graph problem.**
  - **Routing regions and their relationships modeled as graphs.**
- **Three important graph models:**
  1. **Grid Graph Model**
     - Most suitable for area routing
  2. **Checker Board Model**
  3. **Channel Intersection Graph Model**
     - Most suitable for global routing

Grid Graph Model

- A layout is considered to be a collection of unit side square cells (grid).
- Define a graph:
  - Each cell c_i is represented as a vertex v_i.
  - Two vertices v_i and v_j are joined by an edge if the corresponding cells c_i and c_j are adjacent.
  - A terminal in cell c_i is assigned to the corresponding vertex v_i.
  - The occupied cells are represented as filled circles, whereas the others as clear circles.
  - The capacity and length of each edge is set to one.
- Given a 2-terminal net, the routing problem is to find a path between the corresponding vertices in the grid graph.

Grid Graph Model :: Illustration

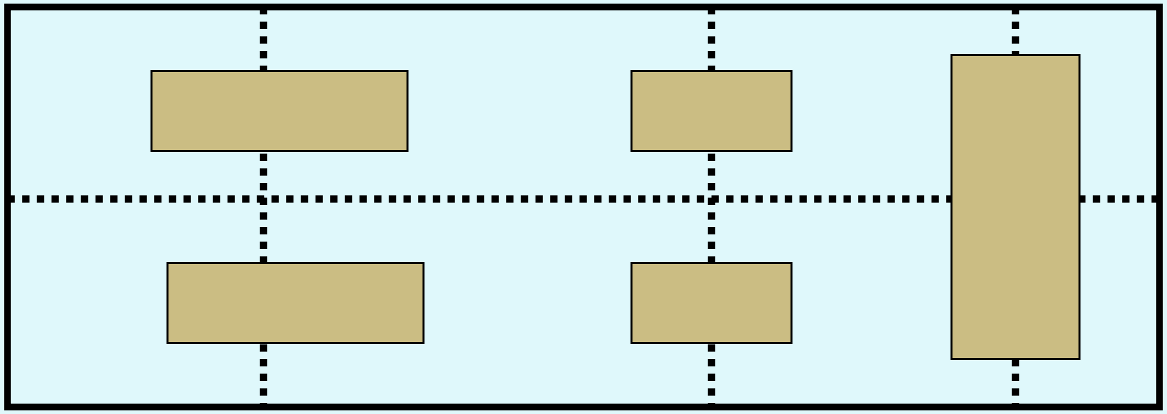


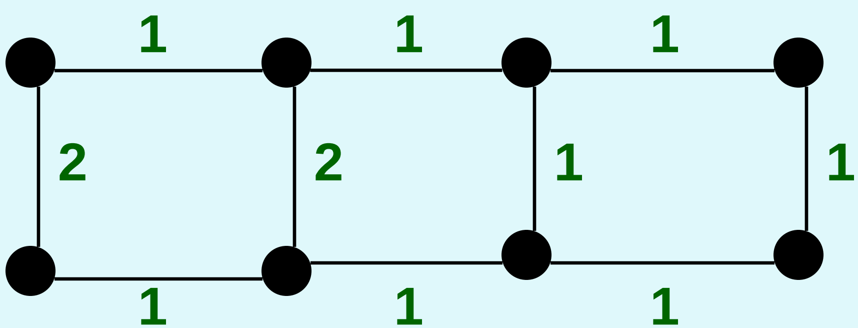


Checker Board Model

- More general than the grid graph model.
- Approximates the layout as a coarse grid.
- Checker board graph is generated in a manner similar to the grid graph.
- The edge capacities are computed based on the actual area available for routing on the cell boundary.
  - The partially blocked edges have unit capacity.
  - The unblocked edges have a capacity of 2.
- Given the cell numbers of all terminals of a net, the global routing problem is to find a path in the coarse grid graph.

Checker Board Model :: Illustration

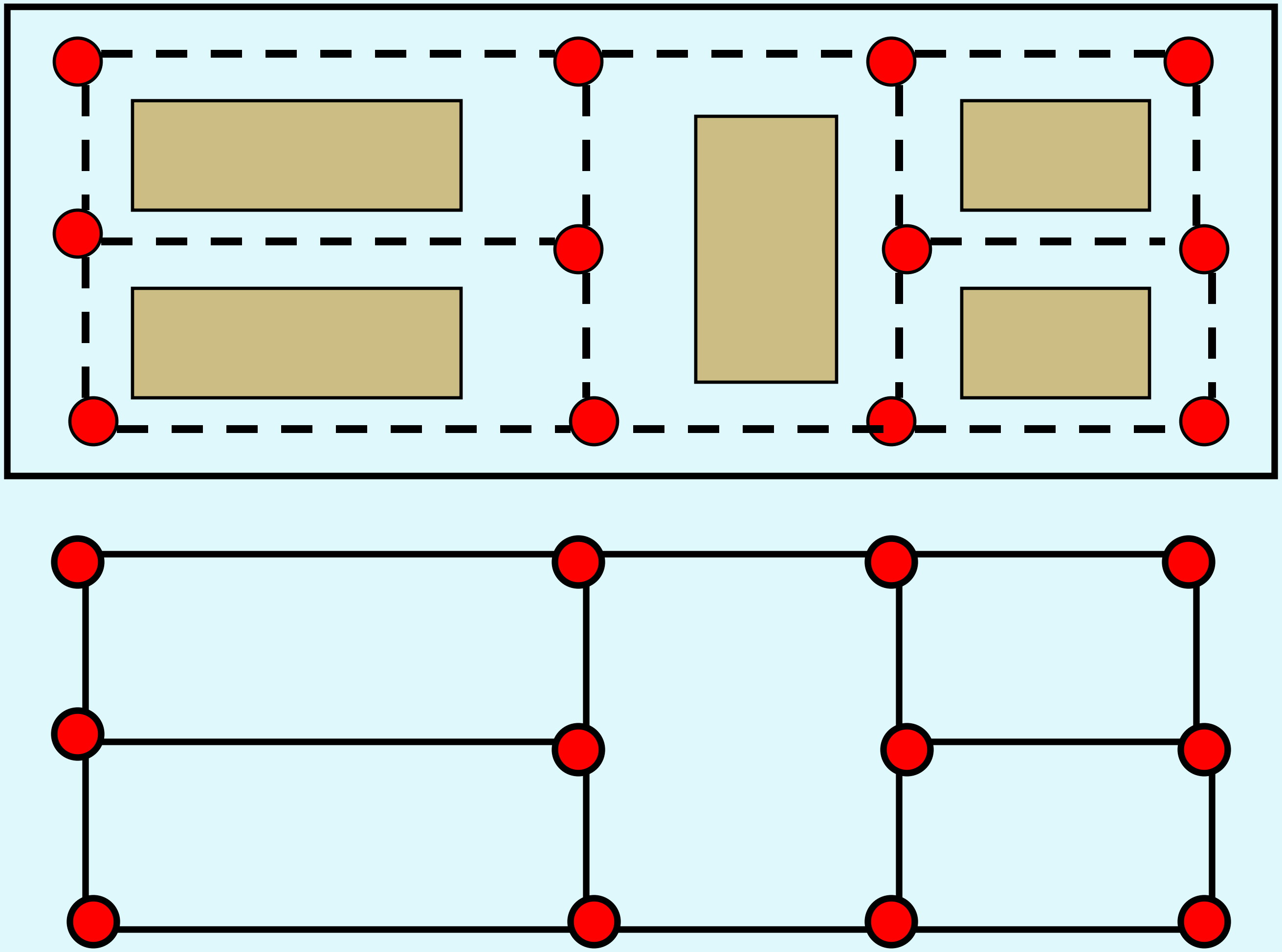




Channel Intersection Graph

- **Most general and accurate model for global routing.**
- **Define a graph:**
  - **Each vertex v_i represents a channel intersection CI_i.**
  - **Channels are represented as edges.**
  - **Two vertices v_i and v_j are connected by an edge if there exists a channel between CI_i and CI_j.**
  - **Edge weight represents channel capacity.**

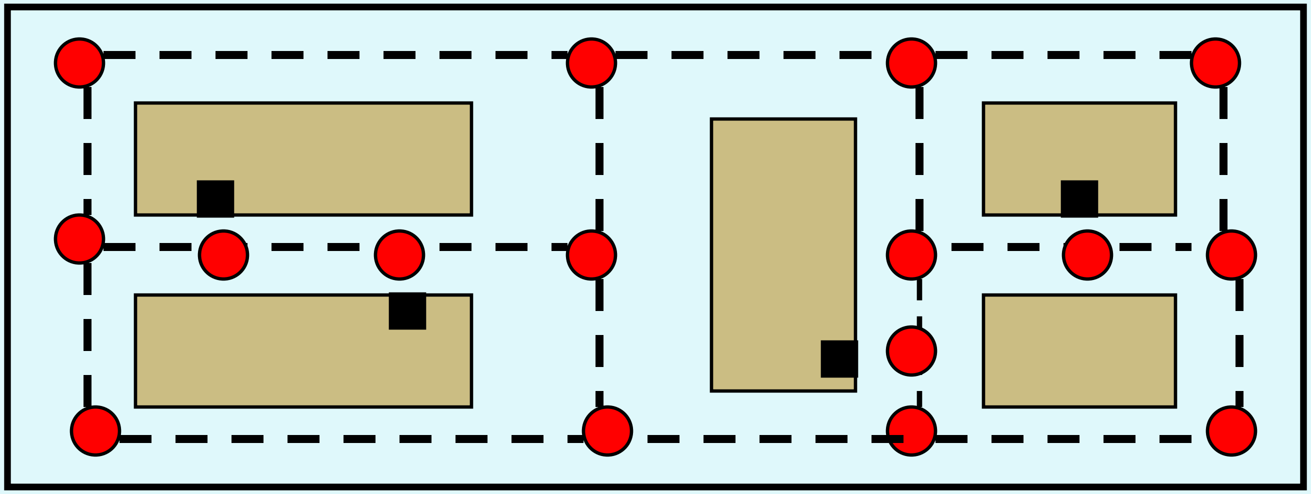
Illustration

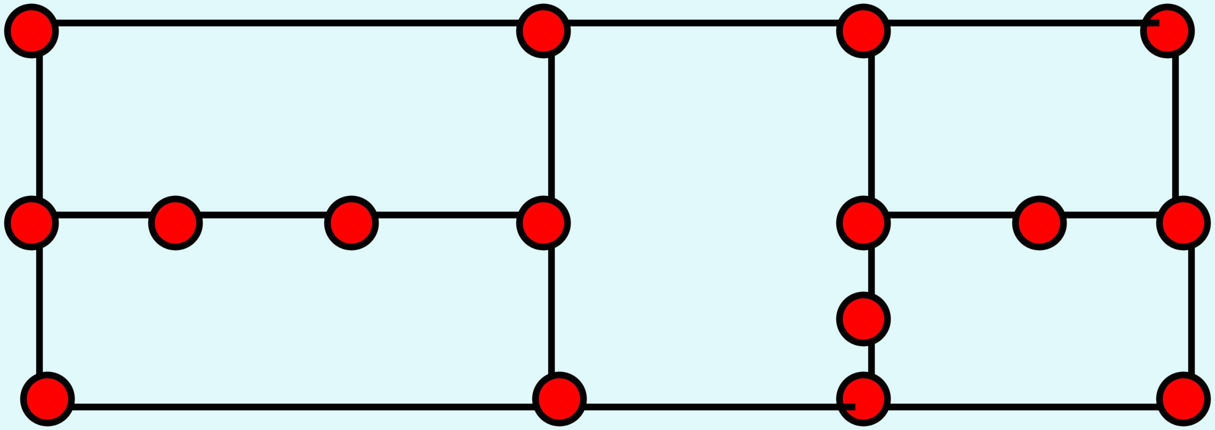


Extended Channel Intersection Graph

- **Extension of the channel intersection graph.**
  - **Includes the pins as vertices so that the connections between the pins can be considered.**
- **The global routing problem is simply to find a path in the channel intersection graph.**
  - **The capacities of the edges must not be violated.**
  - **For 2-terminal nets, we can consider the nets sequentially.**
  - **For multi-terminal nets, we can have an approximation to minimum Steiner tree.**

Illustration



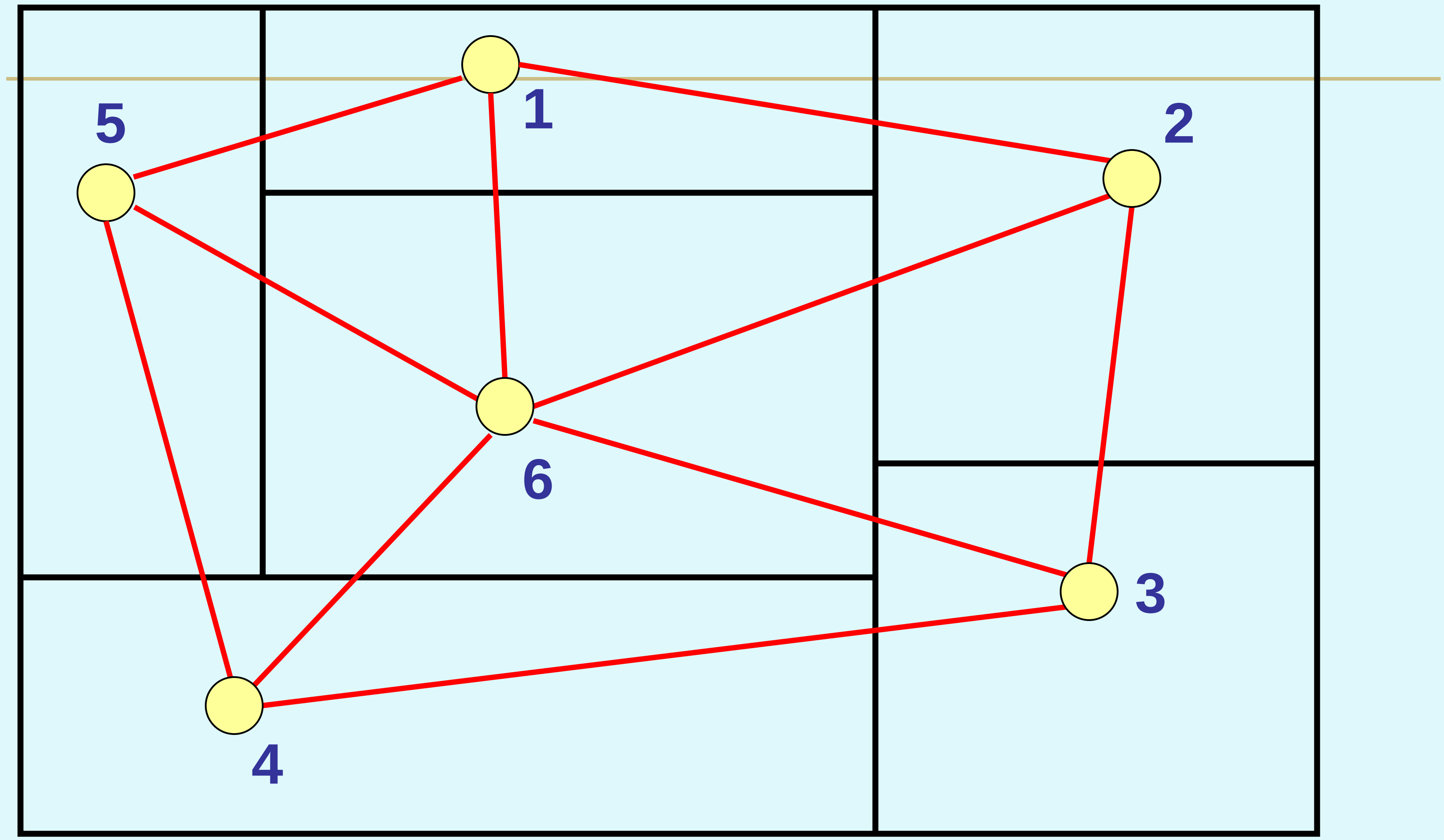


Approaches to Global Routing

- **What does a global router do?**
  - **It decomposes a large routing problem into small and manageable sub-problems**
    - **Called detailed routing**
  - **This is done by finding a rough path for each net**
    - **Sequences of sub-regions it passes through**

When Floorplan is Given

- The dual graph of the floorplan (shown in red) is used for global routing.
- Each edge is assigned with:
  - A weight w_{ij} representing the capacity of the boundary.
  - A value L_{ij} representing the edge length.
- Global routing of a two-terminal net
  - Terminals in rectangles r_1 and r_2.
  - Path connecting vertices v_1 and v_2 in G.


1
2
5
6
3
4

When Placement is Given

- The routing region is partitioned into simpler regions.
  - Typically rectangular in shape.
- A routing graph can be defined.
  - Vertices represent regions, and correspond to channels.
  - Edges represent adjacency between channels.
- Global routing of a two-terminal net
  - Terminals in regions r_1 and r_2.
  - Path connecting vertices v_1 and v_2 in G.

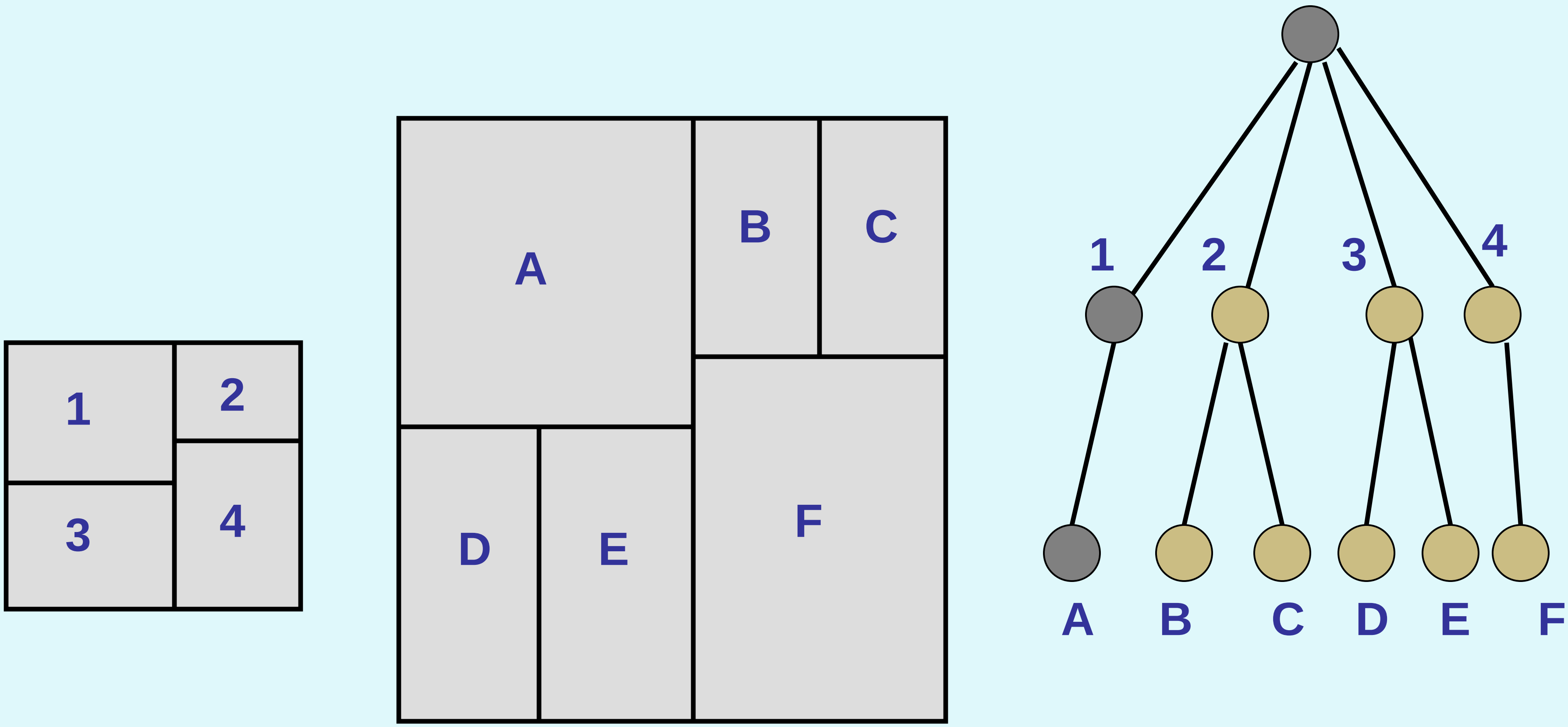
Sequential Approaches

- **Nets are routed sequentially, one at a time.**
  - **First an ordering of the nets is obtained based on:**
    - **Number of terminals**
    - **Bounding box length**
    - **Criticality**
  - **Each net is then routed as dictated by the ordering.**
- **Most of these techniques use variations of maze running or line search methods.**
- **Very efficient at finding routes for nets as they employ well-known shortest path algorithms.**
  - **Rip up and reroute heuristic in case of conflict.**

Hierarchical Approaches

- Use the hierarchy of the routing graph to decompose a large routing problem into sub-problems of manageable size.
  - The sub-problems are solved independently.
  - Sub-solutions are combined to get the total solution.
- A cut tree is defined on the routing graph.
  - Each interior node represents a primitive global routing problem.
  - Each problem is solved optimally by translating it into an integer programming problem.
  - The solutions are finally combined.

Hierarchical Approach :: Illustration



Hierarchical Routing :: Top-Down Approach

- Let the root of the cut tree T be at level '1', and the leaves of T at level 'h'.
  - 'h' is the height of T.
- The top-down approach traverses T from top to down, level by level.
  - I_i denotes the routing problem instance at level i.
- The solutions to all the problem instances are obtained using an integer programming formulation.

Algorithm

```
procedure  Hier_Top_Down
begin
   Compute solution R_i of the routing problem I_1;
   for i=2 to h do
   begin
      for all nodes n at level i-1 do
         Compute solution R_n of the routing problem I_n;
         Combine all solutions R_n for all nodes n, and
                         R_{i-1} into solution R_i;
   end;
end.
```

Hierarchical Routing :: Bottom-up Approach

- **In the first phase, the routing problem associated with each branch in T is solved by IP.**
- **The partial routings are then combined by processing internal tree nodes in a bottom-up manner.**
- **Main disadvantage of this approach:**
  - **A global picture is obtained only in the later stages of the process.**

Algorithm

```
procedure  Hier_Bottom_Down
begin
   Compute solution R_h of the level-h abstraction of
          the problem;
   for i=h to 1 do
     begin
        for all nodes n at level i-1 do
           Compute solution R_n of the routing problem
                I_n by combining the solution to the
                children of node n;
      end;
end;
```

Integer Linear Programming Approach

- The problem of concurrently routing the nets is computationally hard.
  - The only known technique uses integer programming.
- Global routing problem can be formulated as a 0/1 integer program.
- The layout is modeled as a grid graph.
  - N vertices: each vertex represents a grid cell.
  - M edges: an edge connects vertices i and j if the grid cells i and j are adjacent.
  - The edge weight represents the capacity of the boundary.

Contd.

- For each net i, we identify the different ways of routing the net.
  - Suppose that there are n_i possible Steiner trees $t^i_1, t^i_2, \ldots, t^i_{ni}$ to route the net.
  - For each tree t^i_j, we associate a variable x_{ij}:

$$x_{ij} = 1, \text{ if net i is routed using tree } t^i_j$$
$$= 0, \text{ otherwise.}$$

  - Only one tree must be selected for each net:

$$\sum_{j=1}^{n_i} x_{ij} = 1$$

Contd.

- For a grid graph with M edges and $T = \Sigma n_i$ trees, we can represent the routing trees as a 0-1 matrix $A_{MxT} = [a_{ip}]$.

  a_{ip} = 1, if edge i belongs to tree p

  = 0, otherwise.

- Capacity of each arc (boundary) must not be exceeded:

$$\sum_{k=1}^{N} \sum_{l=1}^{n_k} a_{ip}\, x_{lk} \leq c_i$$

- If each tree t^j_i is assigned a cost g_{ij}, then a possible objective function to minimize is:

$$F = \sum_{i=1}^{N} \sum_{j=1}^{n_k} g_{ij}\, x_{ij}$$

Contd.

- **0-1 integer programming formulation:**

Minimize

$$\sum_{i=1}^{N} \sum_{j=1}^{n_k} g_{ij}\, x_{ij}$$

Subject to:

$$\sum_{j=1}^{n_i} x_{ij} = 1, \qquad 1 \le i \le N$$

$$\sum_{k=1}^{N} \sum_{l=1}^{n_k} a_{ip}\, x_{lk} \le c_i\,, \qquad 1 \le i \le M$$

$$x_{kj} = 0,1 \qquad 1 \le k \le N,\ 1 \le j \le n_k$$

Contd.

- General formulation.
- Looks at the problem globally.
- Not feasible for large input sizes.
  - Time complexity increases exponentially with problem size.

Performance Driven Routing

- Advent of deep sub-micron technology
  - Interconnect delay constitutes a significant part of the total net delay.
  - Reduction in feature sizes has resulted in increased wire resistance.
  - Increased proximity between the devices and interconnections results in increased cross-talk noise.
- Routers should model the cross-talk noise between adjacent nets.
- For routing high-performance circuits, techniques adopted:
  - Buffer insertion
  - Wire sizing
  - High-performance topology generation