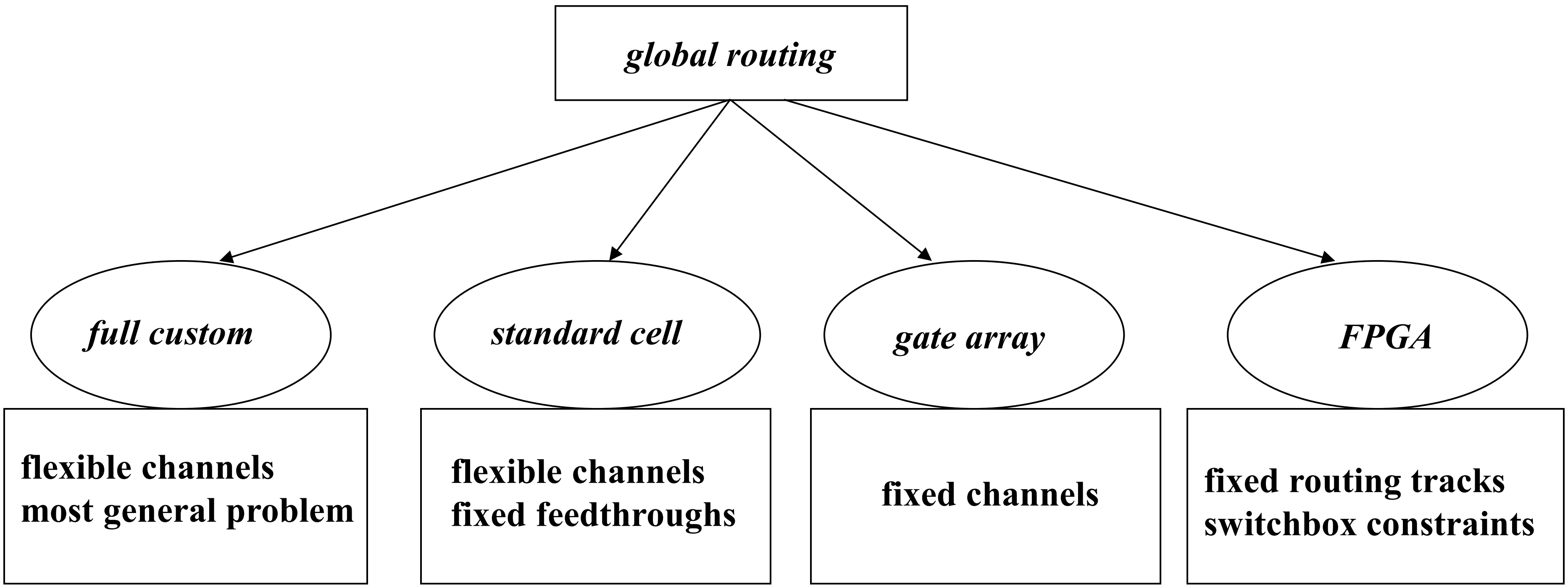
Global Routing Problem

- Given a netlist $N=\{N_1,N_2,\ldots,N_n\}$, a routing graph $G=(V,E)$, find a Steiner tree T_i for each net N_i, $1 \le i \le n$, such that $U(e_j) \le c(e_j)$, $\forall e_j \in E$ and $\sum_{i=1}^{n} L(T_i)$ is minimized,

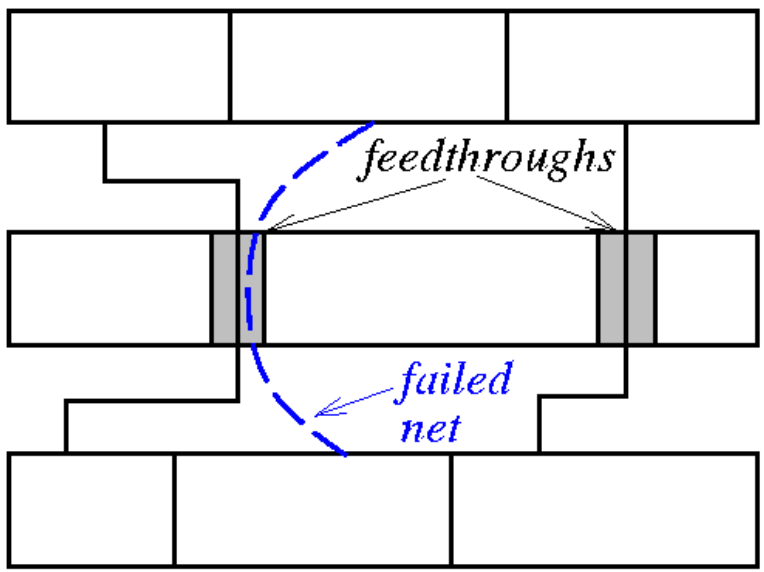
  where
  - $c(e_j)$: capacity of edge e_j;
  - $x_{ij}=1$ if e_j is in T_i; $x_{ij}=0$ otherwise;
  - $U(e_j)= \sum_{i=1}^{n} x_{ij}$: # of wires that pass through the channel corresponding to edge e_j;
  - $L(T_i)$: total wirelength of Steiner tree T_i.
- For high-performance, the maximum wirelength $(\max_{i=1}^{n} L(T_i))$ is minimized (or the longest path between two points in T_i is minimized).

Global Routing in Different Design Styles



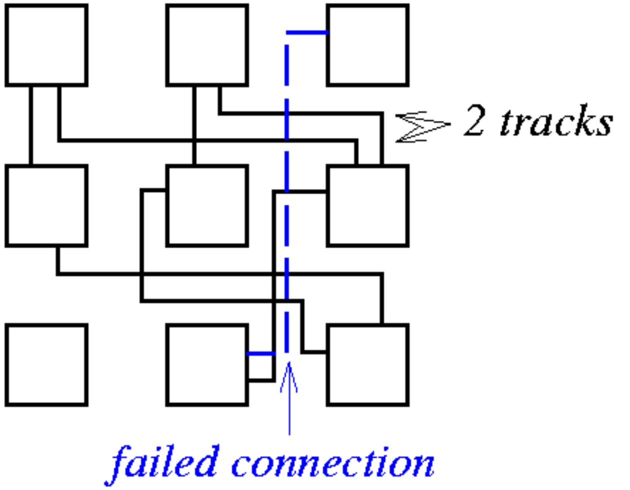
Global Routing in Standard Cell

- Objective
  - Minimize total channel height.
  - Assignment of **feedthroughs**.
- For high performance,
  - Minimize the maximum wire length.
  - Minimize the maximum path length.



Global Routing in Gate Array

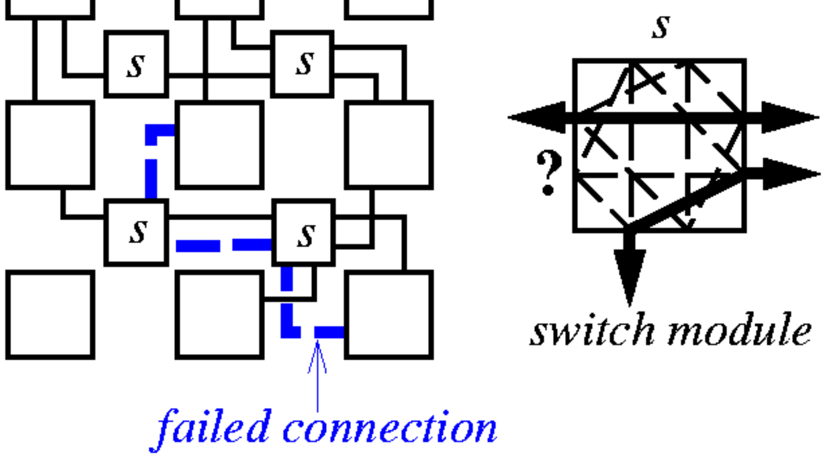
- Objective
  - **Guarantee 100% routability**.
- For high performance,
  - Minimize the maximum wire length.
  - Minimize the maximum path length.



Each channel has a capacity of 2 tracks.

Global Routing in FPGA

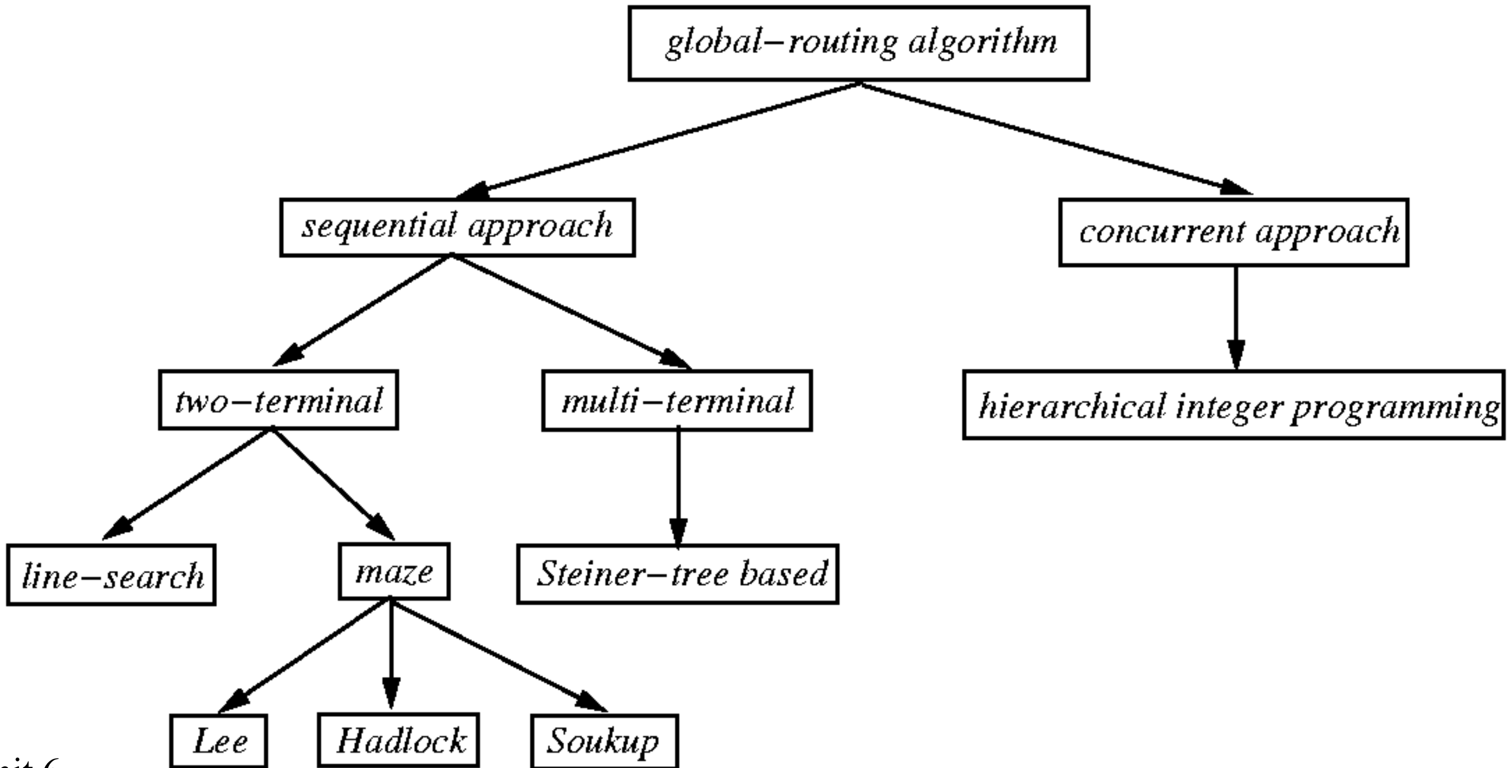
- Objective
  - Guarantee 100% routability.
  - Consider **switch-module architectural constraints**.
- For performance-driven routing,
  - **Minimize # of switches used**.
  - Minimize the maximum wire length.
  - Minimize the maximum path length.



Each channel has a capacity of 2 tracks.

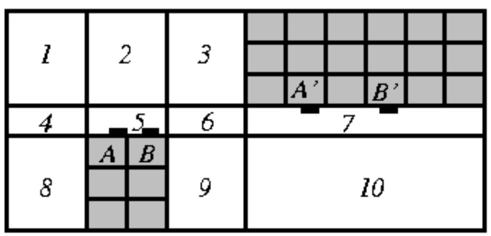
Classification of Global Routing Algorithm

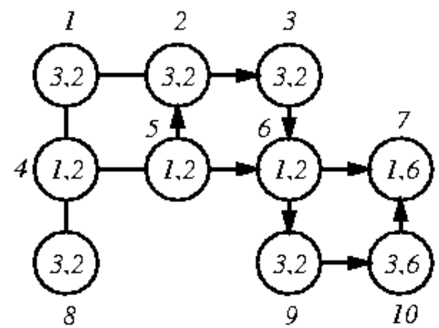
- **Sequential approach**: Assigns priority to nets; routes one net at a time based on its priority (net ordering?).
- **Concurrent approach**: All nets are considered at the same time (complexity?)

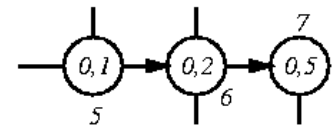


Global Routing: Maze Routing

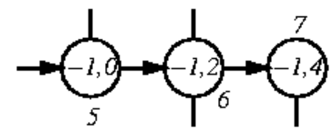
- Routing channels may be modeled by a weighted undirected graph called **channel connectivity graph**.
- Node $\leftrightarrow$ channel; edge $\leftrightarrow$ two adjacent channels; capacity: (*width*, *length*)



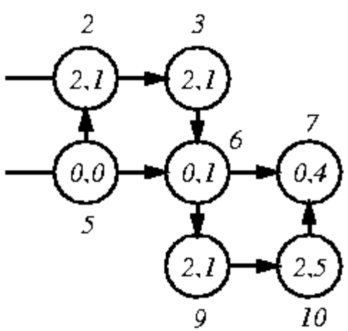


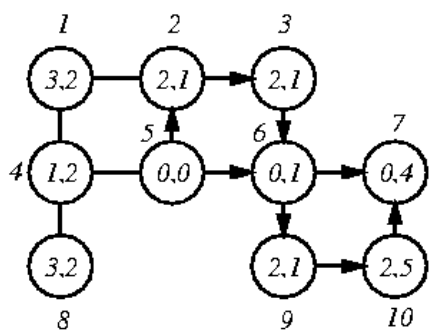


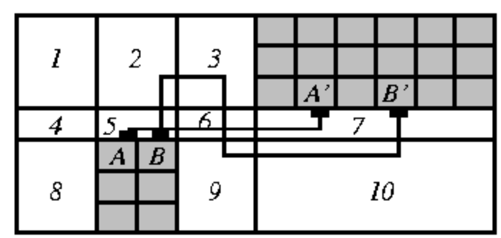
route A–A' via 5–6–7



route B–B' via 5–6–7







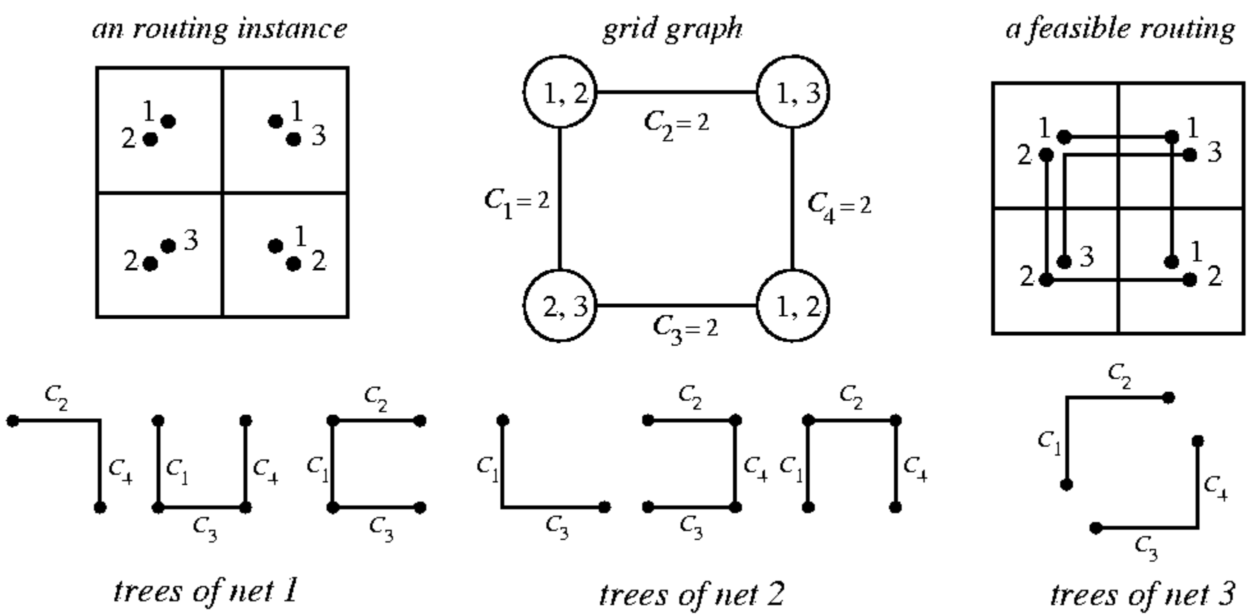
route B–B' via 5–2–3–6–9–10–7

updated channel graph

maze routing for nets A and B

Global Routing by Integer Programming

- Suppose that for each net i, there are n_i possible trees $t_1^i, t_2^i, \ldots, t_{n_i}^i$ to route the net.
- Constraint I: For each net i, only one tree t_j^i will be selected.
- Constraint II: The capacity of each cell boundary c_i is not exceeded.
- Minimize the total tree cost.
- **Question:** Feasible for practical problem sizes?
  - **Key**: hierarchical approach!



An Integer-Programming Example

Boundary	t_1^1	t_2^1	t_3^1	t_1^2	t_2^2	t_3^2	t_1^3	t_2^3
B1	0	1	1	1	0	1	1	0
B2	1	0	1	0	1	1	1	0
B3	0	1	1	1	1	0	0	1
B4	1	1	0	0	1	1	0	1

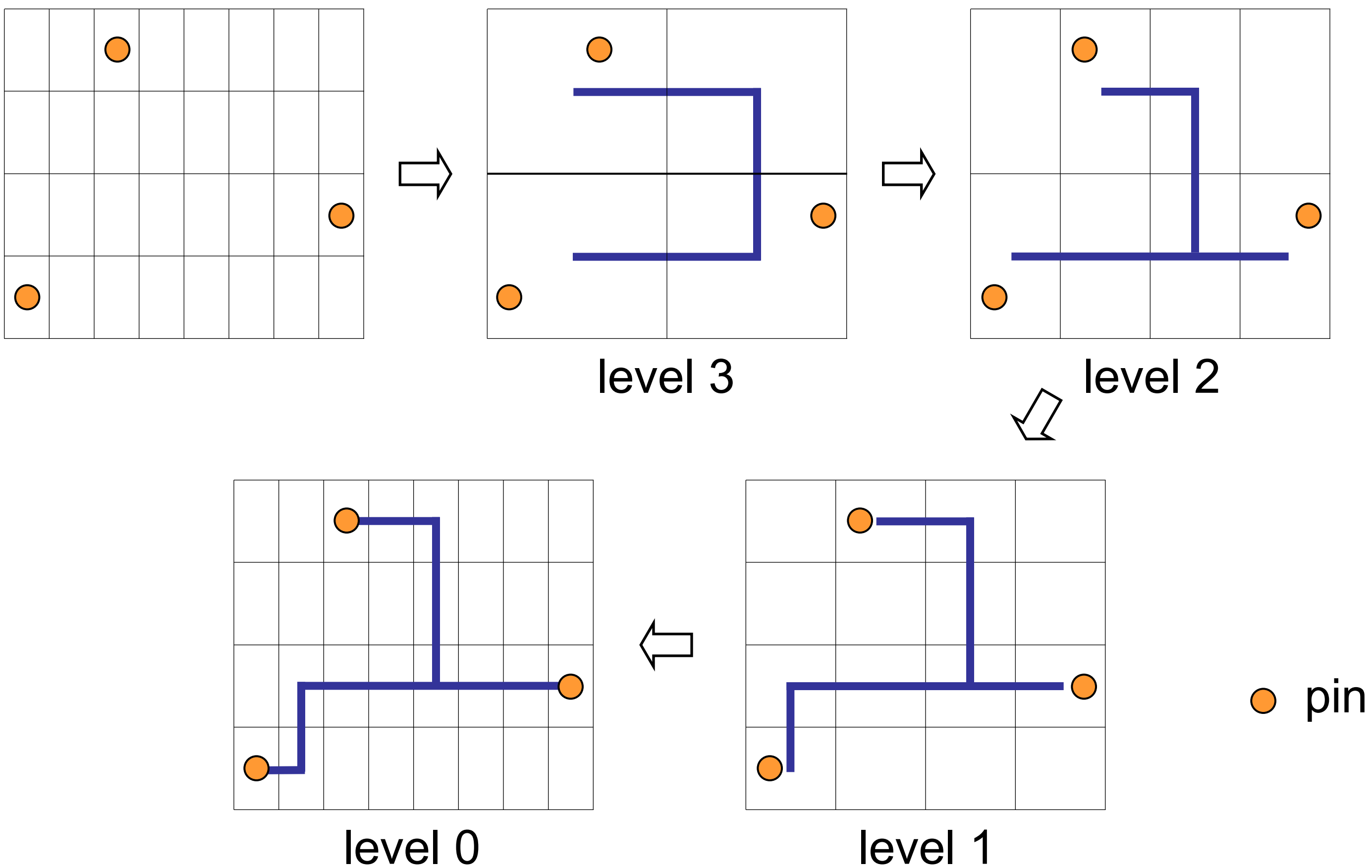
- $g_{i,j}$: cost of tree t_j^i $\Rightarrow$ $g_{1,1} = 2$, $g_{1,2} = 3$, $g_{1,3} = 3$, $g_{2,1} = 2$, $g_{2,2} = 3$, $g_{2,3} = 3$, $g_{3,1} = 2$, $g_{3,2} = 2$.

Minimize $2x_{1,1} + 3x_{1,2} + 3x_{1,3} + 2x_{2,1} + 3x_{2,2} + 3x_{2,3} + 2x_{3,1} + 2x_{3,2}$
subject to

$$\begin{aligned}
x_{1,1} + x_{1,2} + x_{1,3} &= 1 \quad (Constraint\ I : t^1) \\
x_{2,1} + x_{2,2} + x_{2,3} &= 1 \quad (Constraint\ I : t^2) \\
x_{3,1} + x_{3,2} &= 1 \quad (Constraint\ I : t^3) \\
x_{1,2} + x_{1,3} + x_{2,1} + x_{2,3} + x_{3,1} &\leq 2 \quad (Constraint\ II : B1) \\
x_{1,1} + x_{1,3} + x_{2,2} + x_{2,3} + x_{3,1} &\leq 2 \quad (Constraint\ II : B2) \\
x_{1,2} + x_{1,3} + x_{2,1} + x_{2,2} + x_{3,2} &\leq 2 \quad (Constraint\ II : B3) \\
x_{1,1} + x_{1,2} + x_{2,2} + x_{2,3} + x_{3,2} &\leq 2 \quad (Constraint\ II : B4) \\
x_{i,j} &= 0, 1, 1 \leq i, j \leq 3
\end{aligned}$$

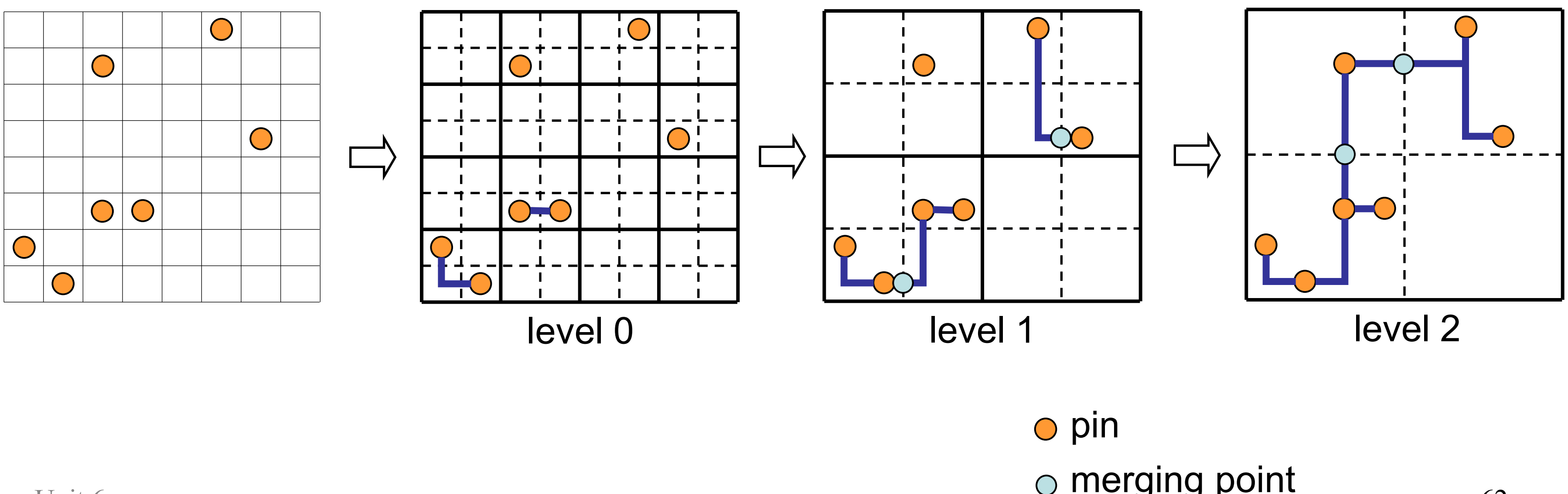
Top-down Hierarchical Global Routing

- Recursively divides routing regions into successively smaller **super cells**, and nets at each hierarchical level are routed sequentially or concurrently.



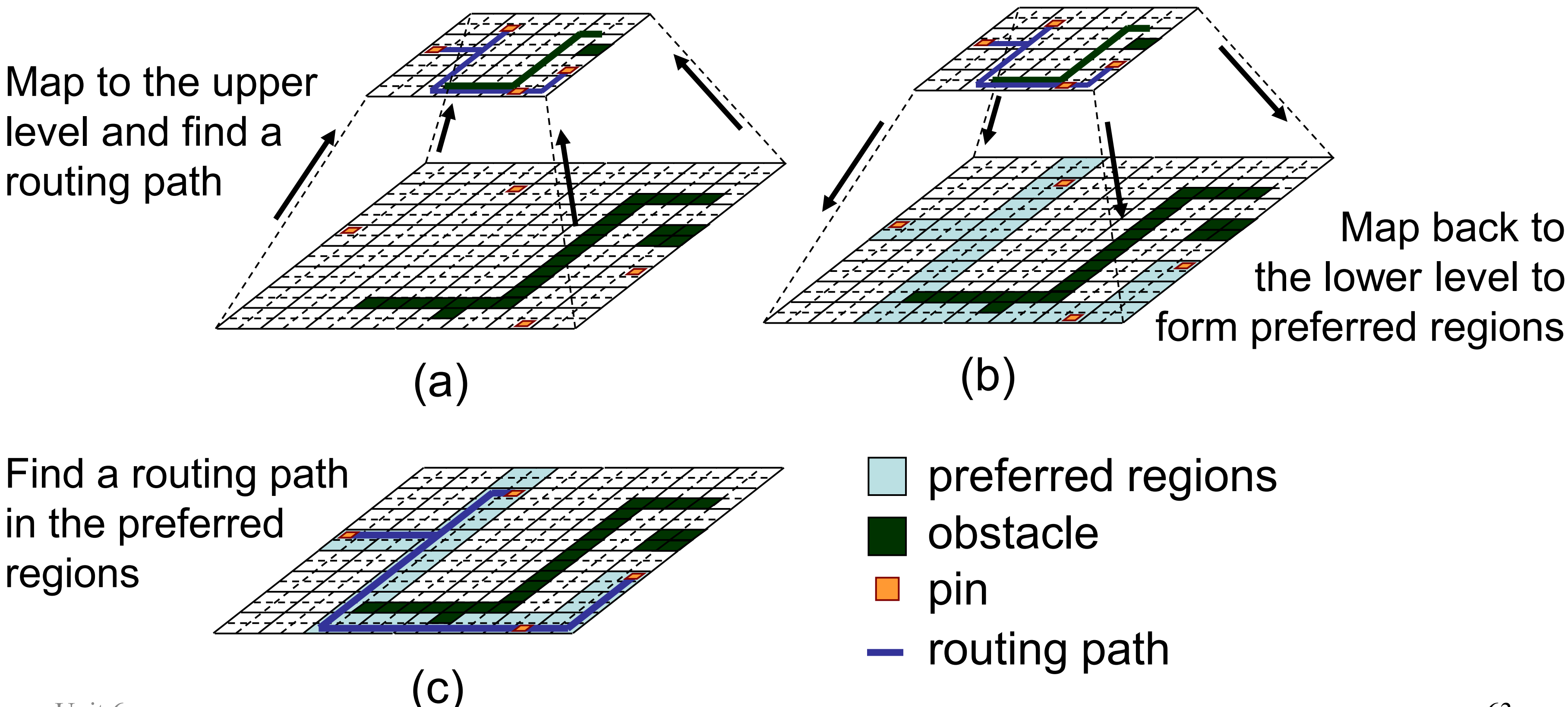
Bottom-up Hierarchical Global Routing

- At each hierarchical level, routing is restrained within each super cell individually.
- When the routing at the current level is finished, every four super cells are merged to form a new larger super cell at the next higher level.



Hybrid Hierarchical Global Routing

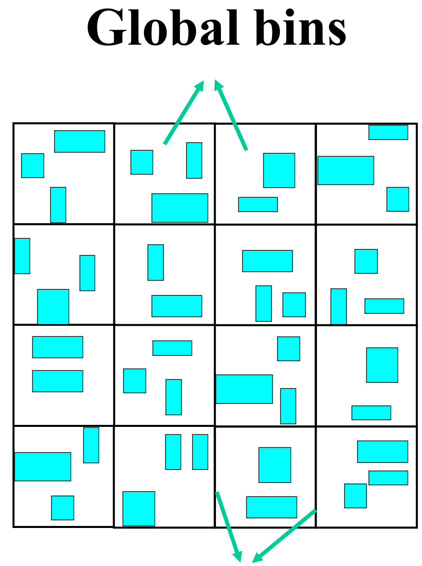
- (1) neighboring propagation, (2) preference partitioning, and (3) bounded routing.

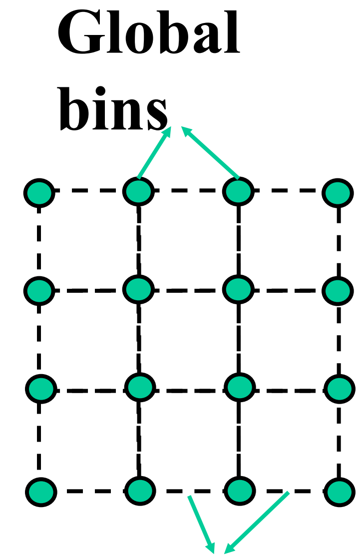


Global Routing Problem Revisited

- supply $s(e)$: the number of available routing tracks passing through edge e

- demand $d(e)$: the number of wires that utilize edge e





- $overflow(e) = \begin{cases} d(e) - s(e) & if\, d(e) > s(e) \\ 0 & otherwise \end{cases}$

Global Routing Problem Revisited

- Input: a set of nets to be routed over a grid graph
- Output: Steiner tree topologies for all nets such that the total overflow, the maximum overflow and the total wirelength are minimized