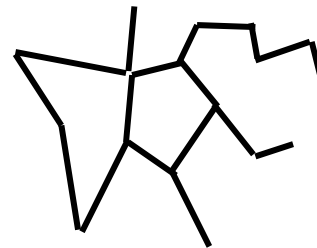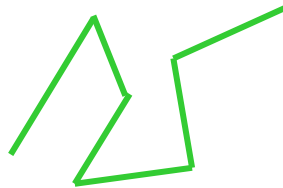# Curve - Graph

Shape Matching for Curves and Graphs

Free Space Diagram

- For each edge (i,j) in G let $FD_{i,j} = FD(\alpha, (i,j))$
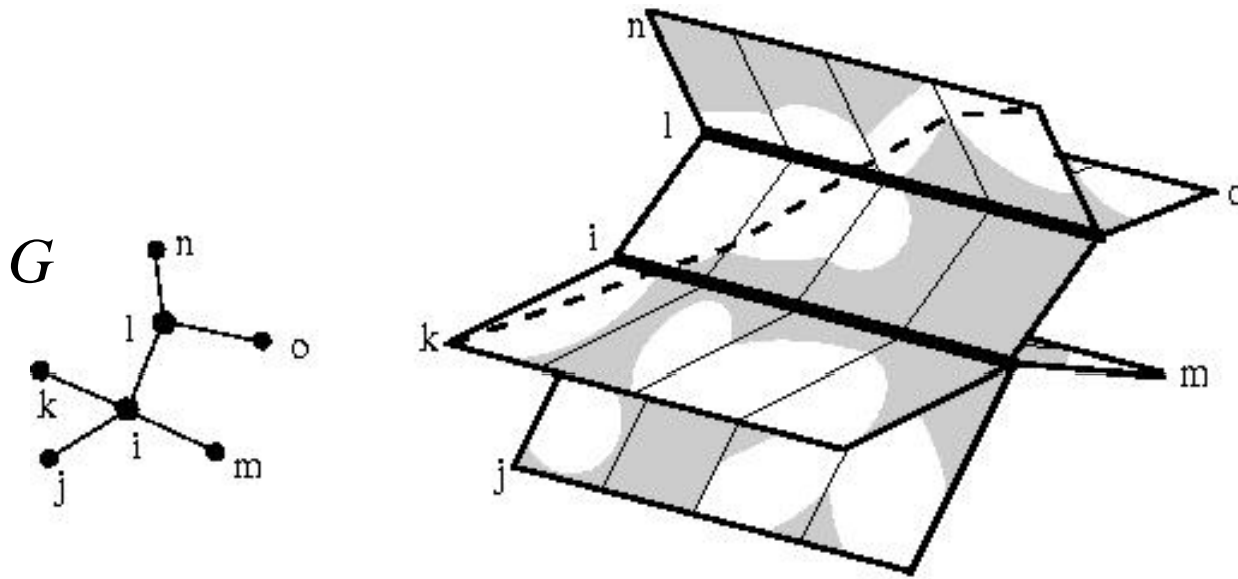
Shape Matching for Curves and Graphs
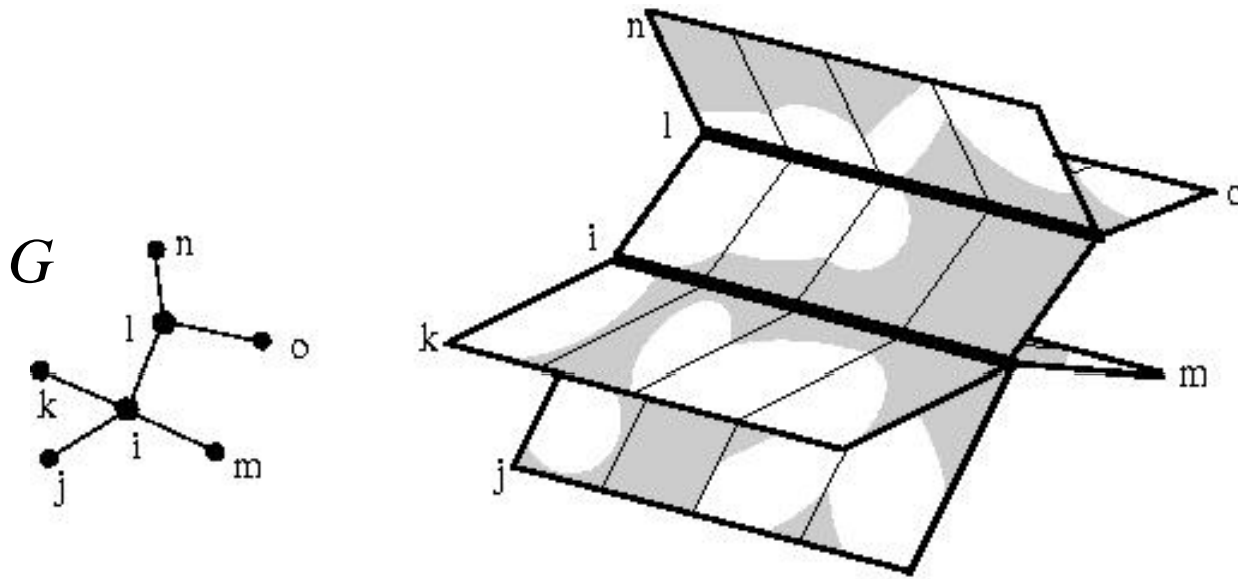
Free Space Surface

- Glue free space diagrams $FD_{i,j}$ together according to adjacency information in G
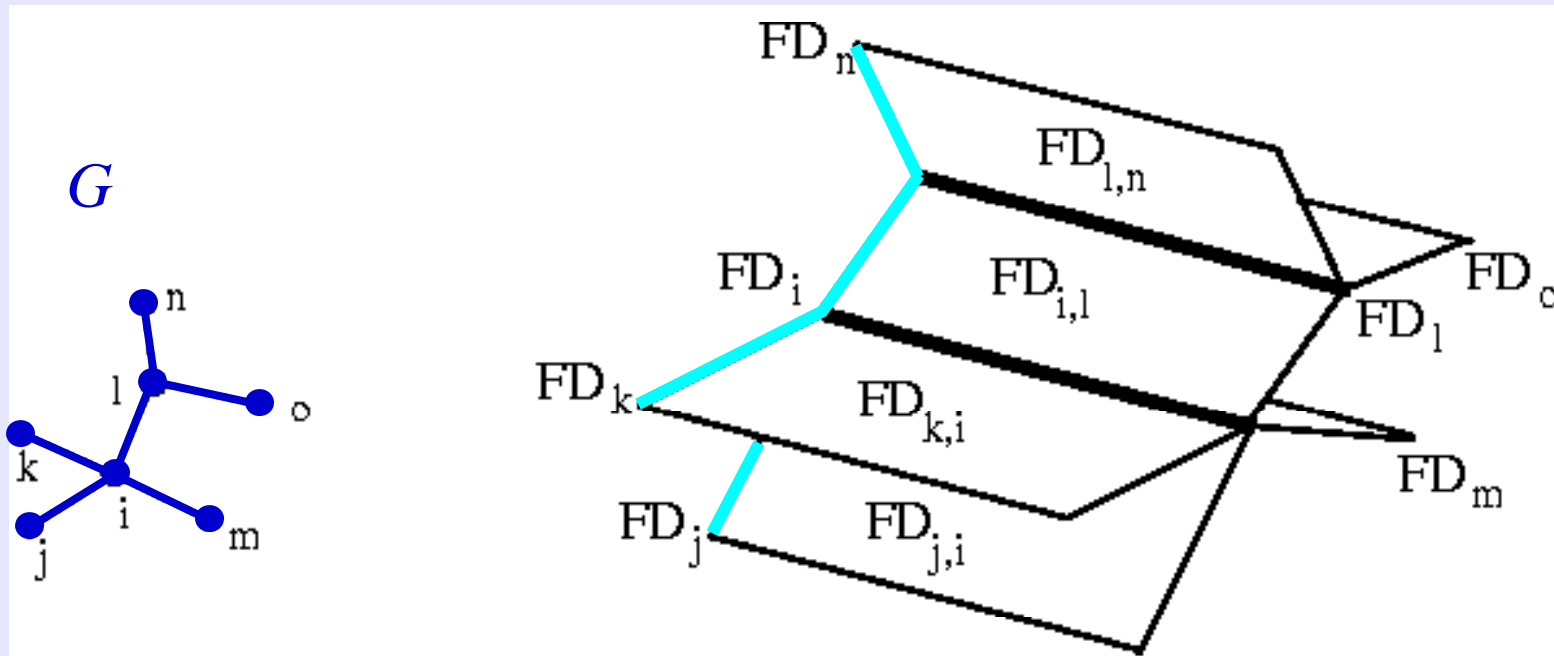
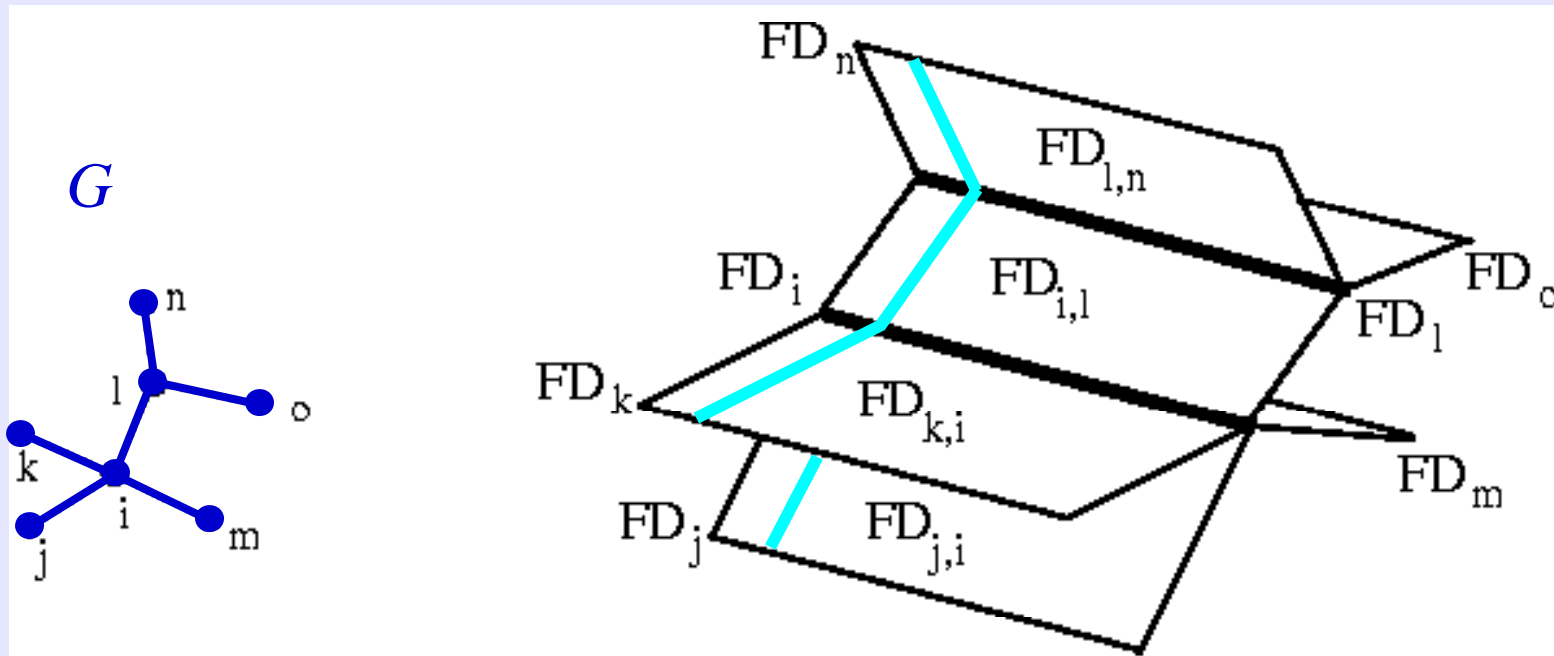 ➡ **Free space surface of α and G**

Free Space Surface

- **TASK:** Find **monotone path** in free space surface
  - starting in some lower left corner, and
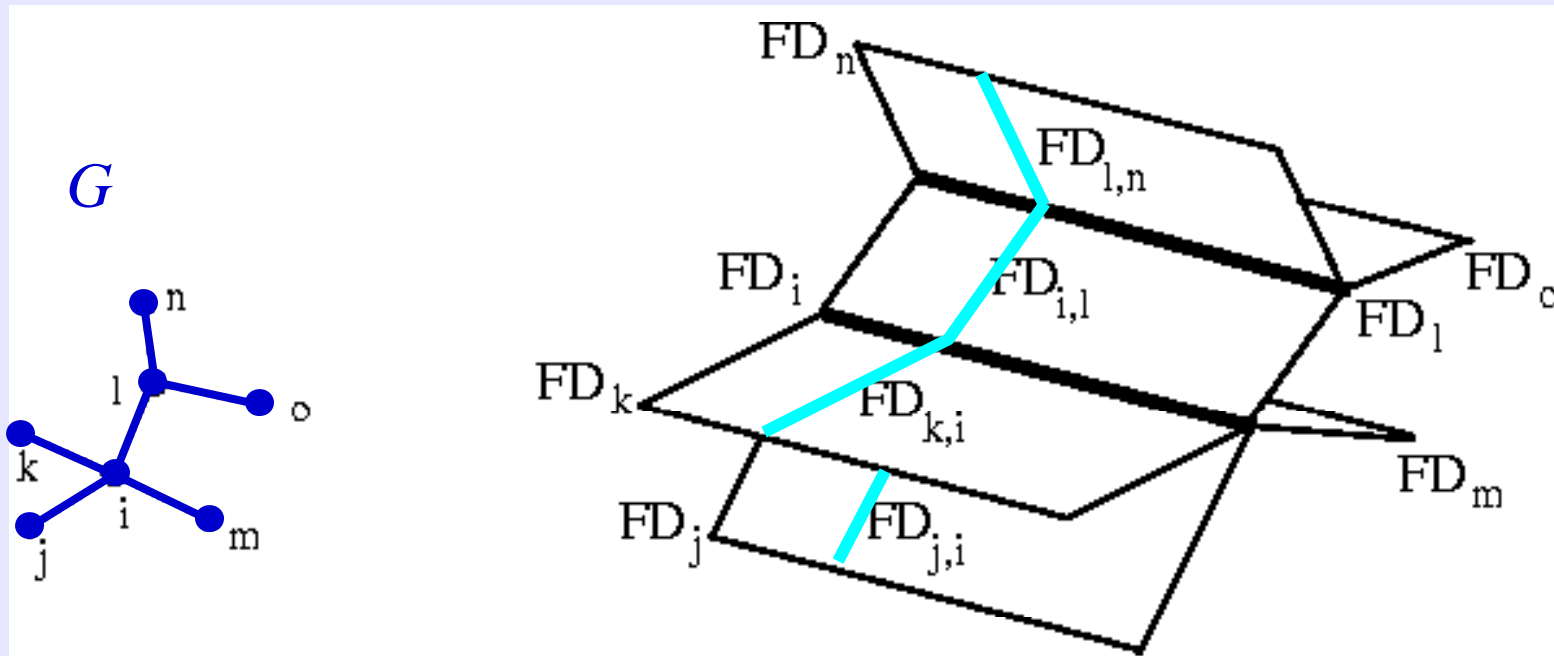 - ending in some upper right corner

Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right

Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right

Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right
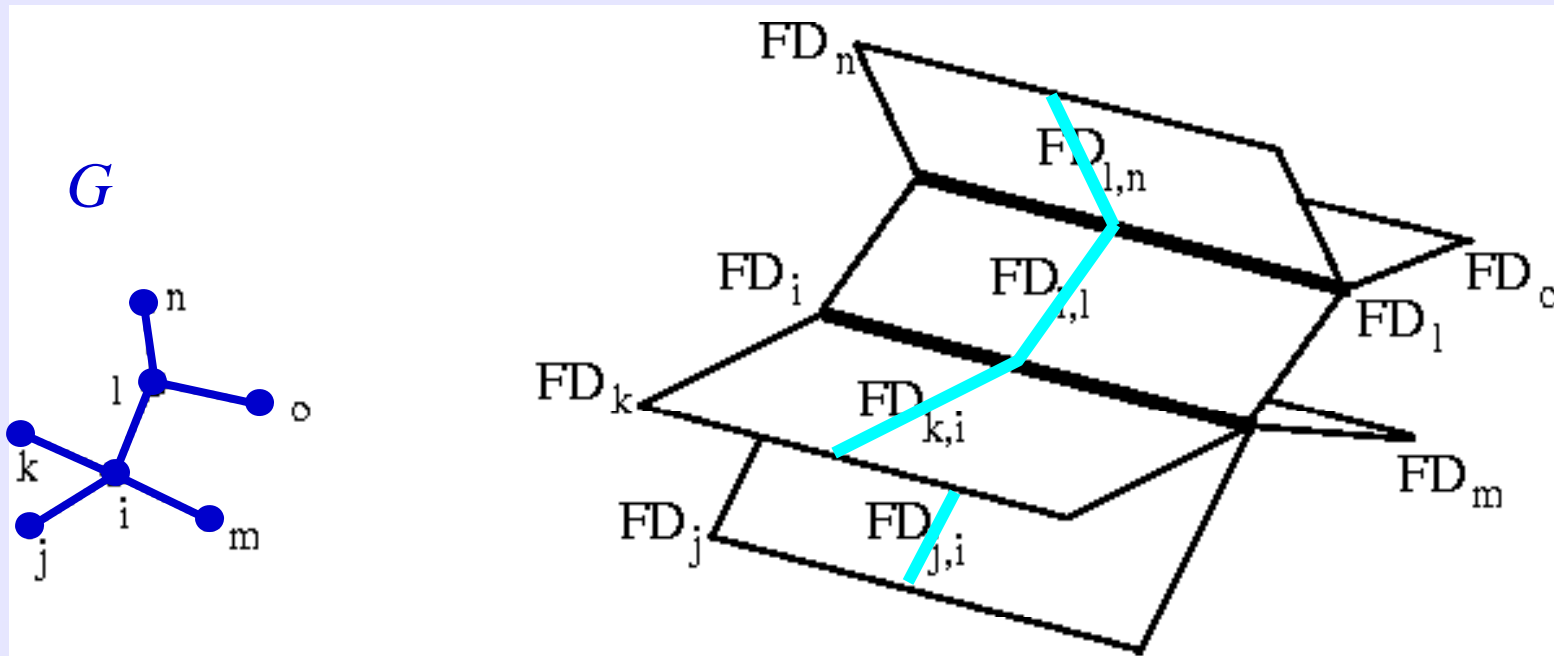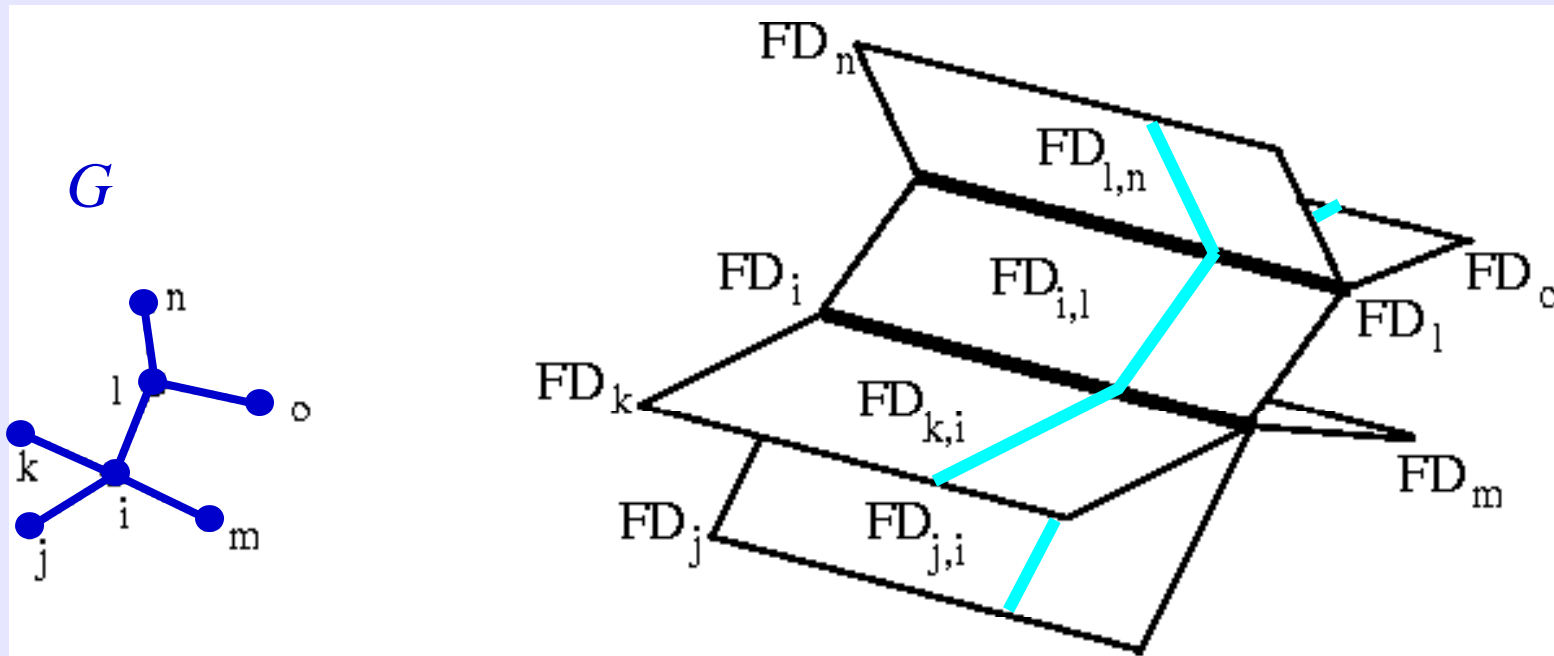
Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right
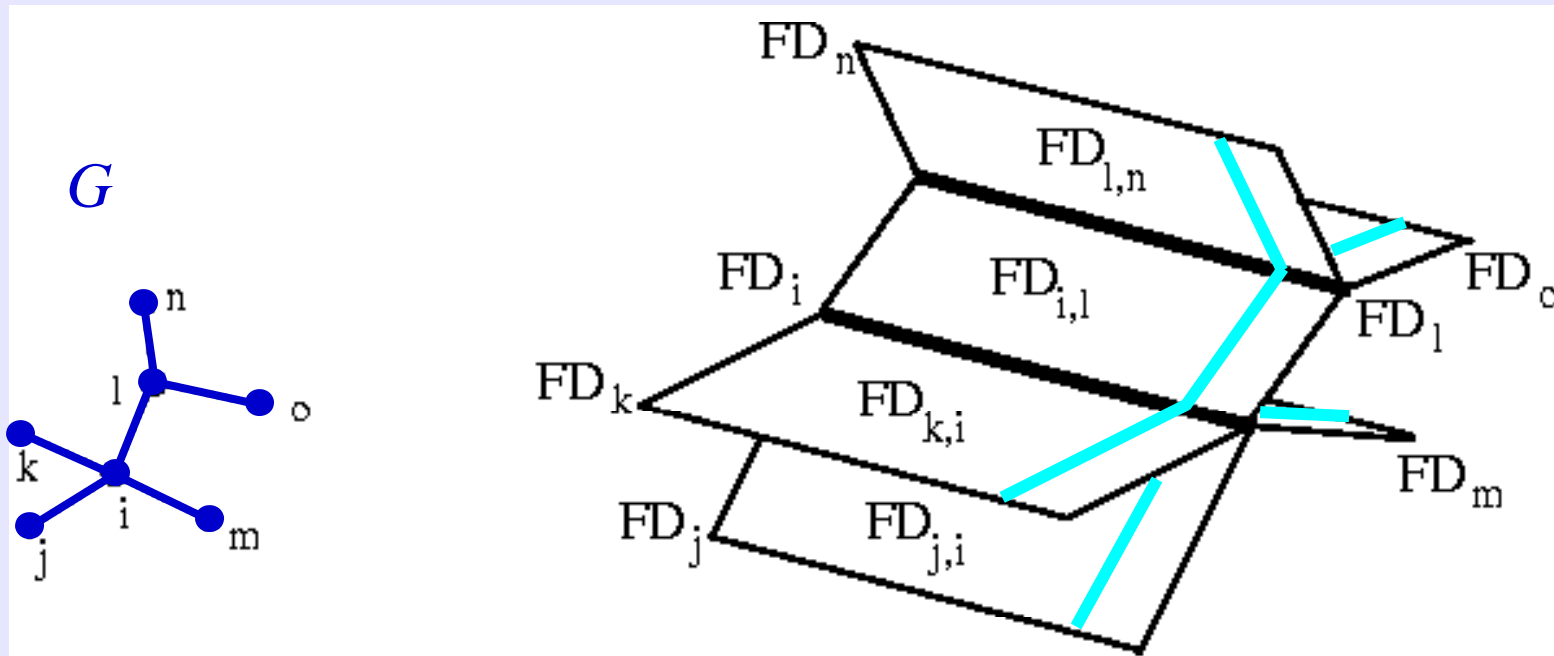
Sweep

- Sweep all $FD_{i,j}$ at once with a <mark>sweepline</mark> from left to right
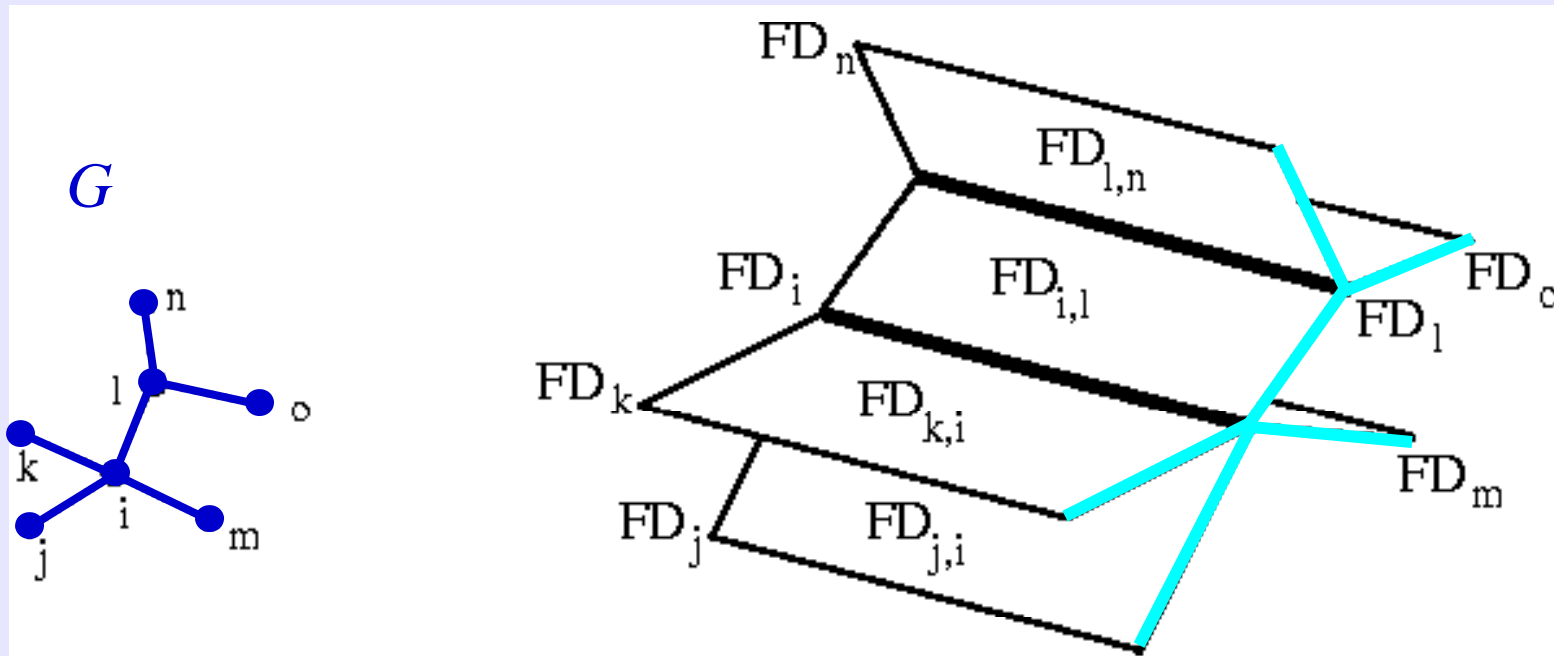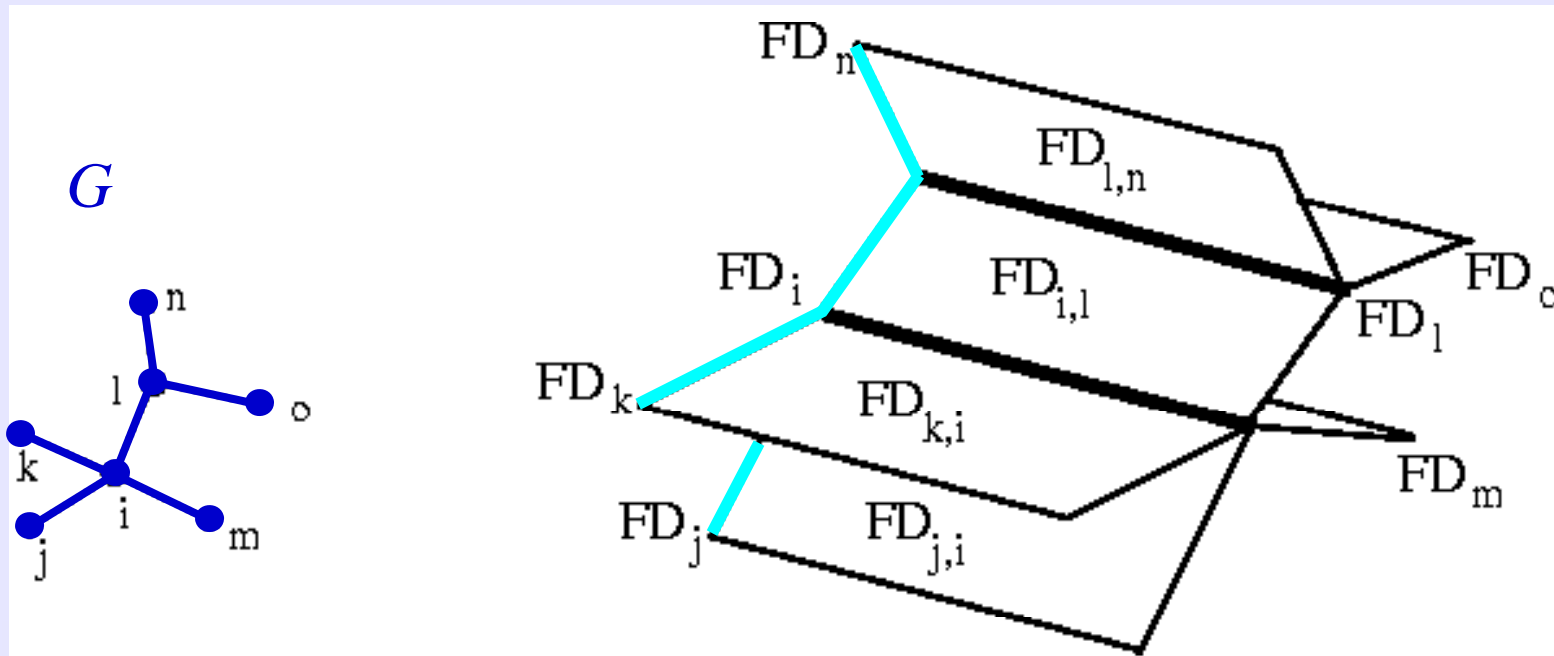
Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right

Sweep

- Sweep all $FD_{i,j}$ at once with a sweepline from left to right

Maintaining reachable points

- **During sweep:**

   Maintain `points` on free space surface, up to the sweepline, that can be reached by some monotone path from some lower left corner

Maintaining reachable points

- **During sweep:**

  Maintain points on free space surface, up to the sweepline, that can be reached by some monotone path from some lower left corner
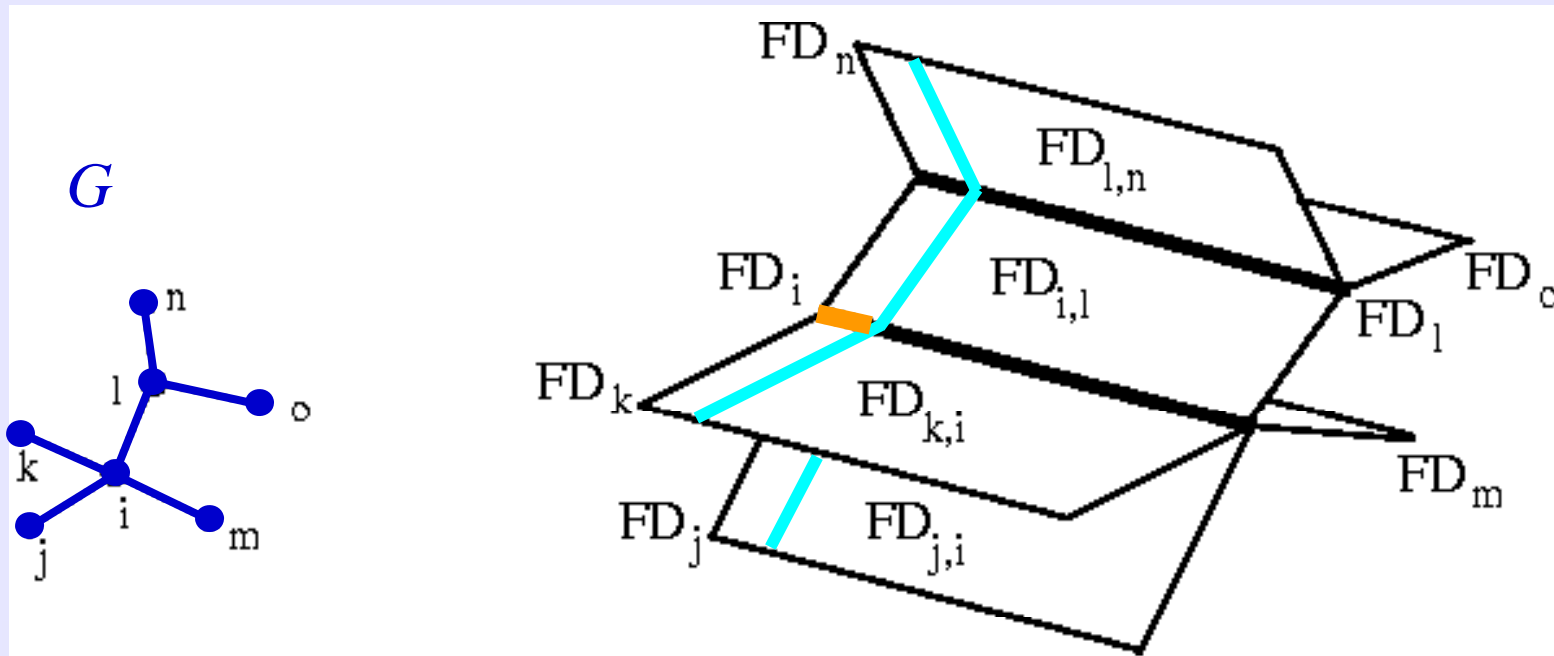
Maintaining reachable points

- **During sweep:**

   Maintain **points** on free space surface, up to the sweepline, that can be reached by some monotone path from some lower left corner
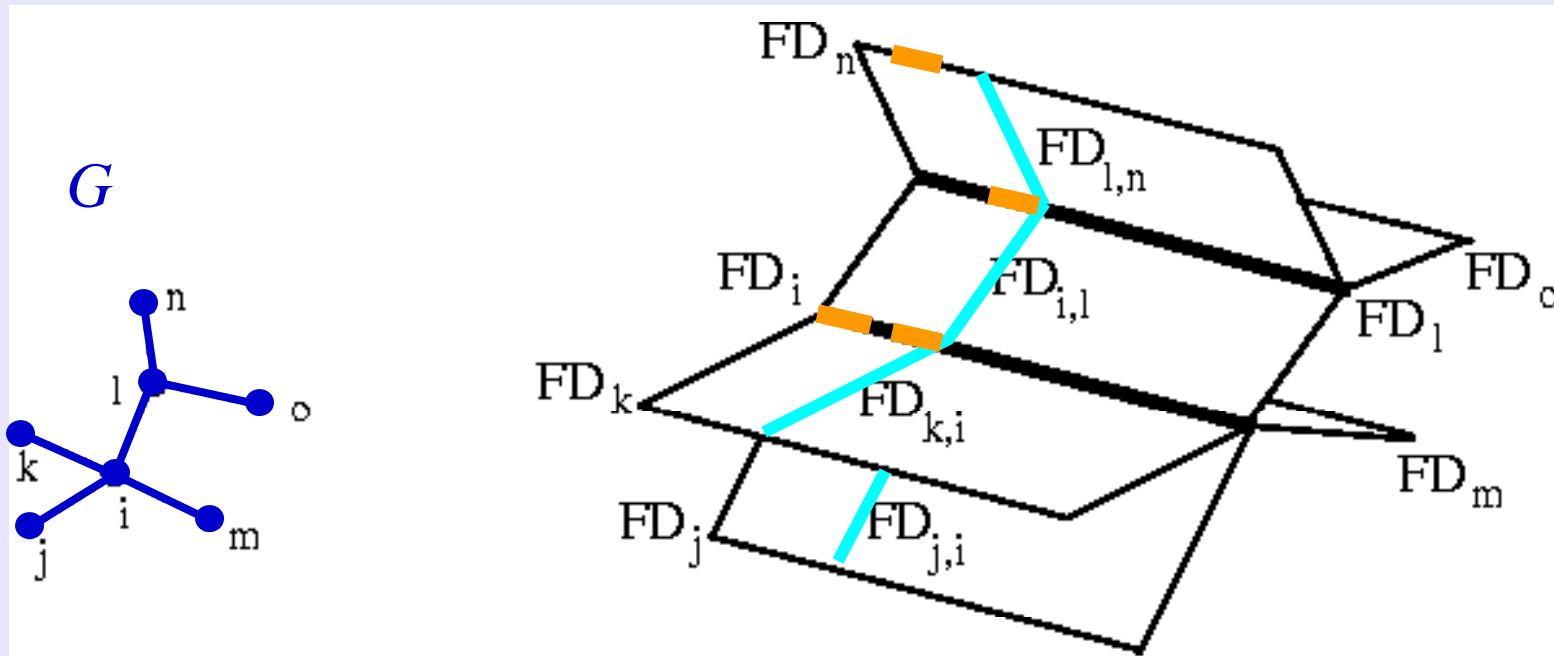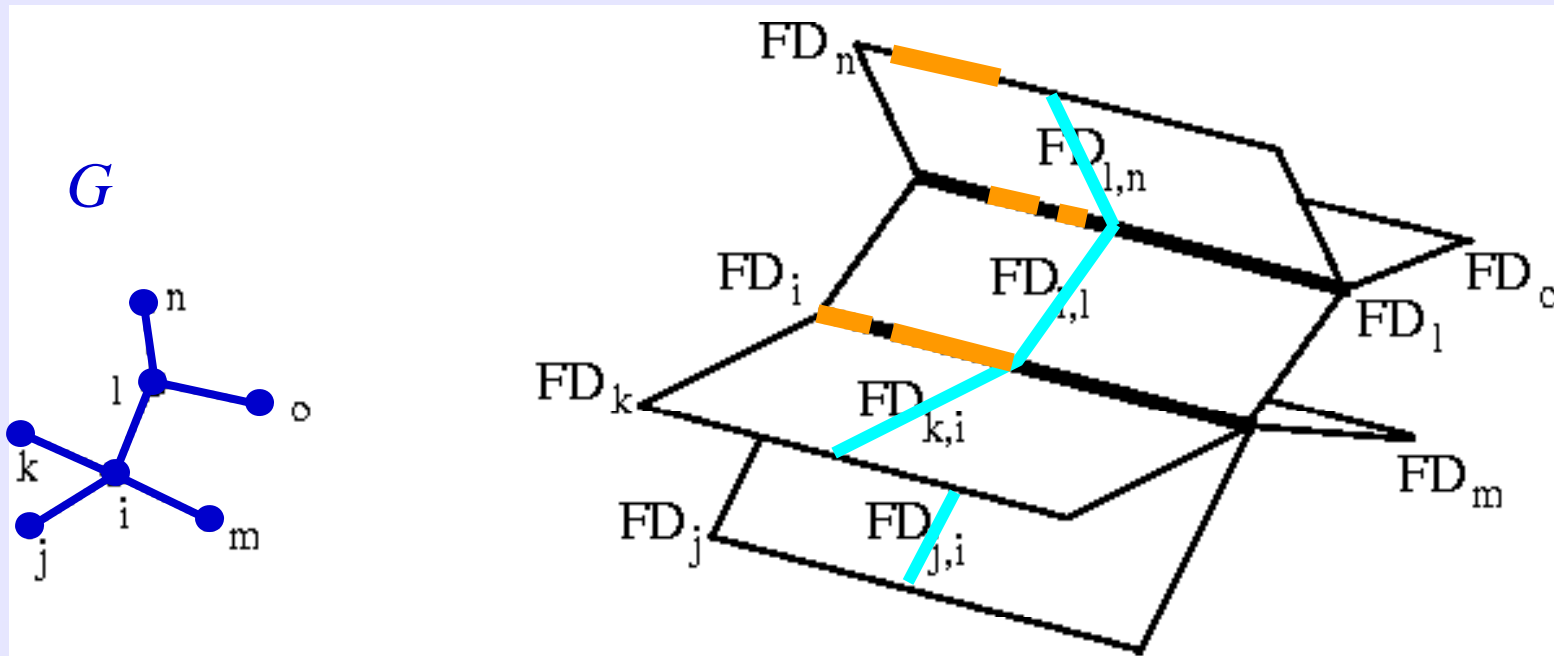
Maintaining reachable points

- **During sweep:**

  Maintain `points` on free space surface, up to the sweepline, that can be reached by some monotone path from some lower left corner
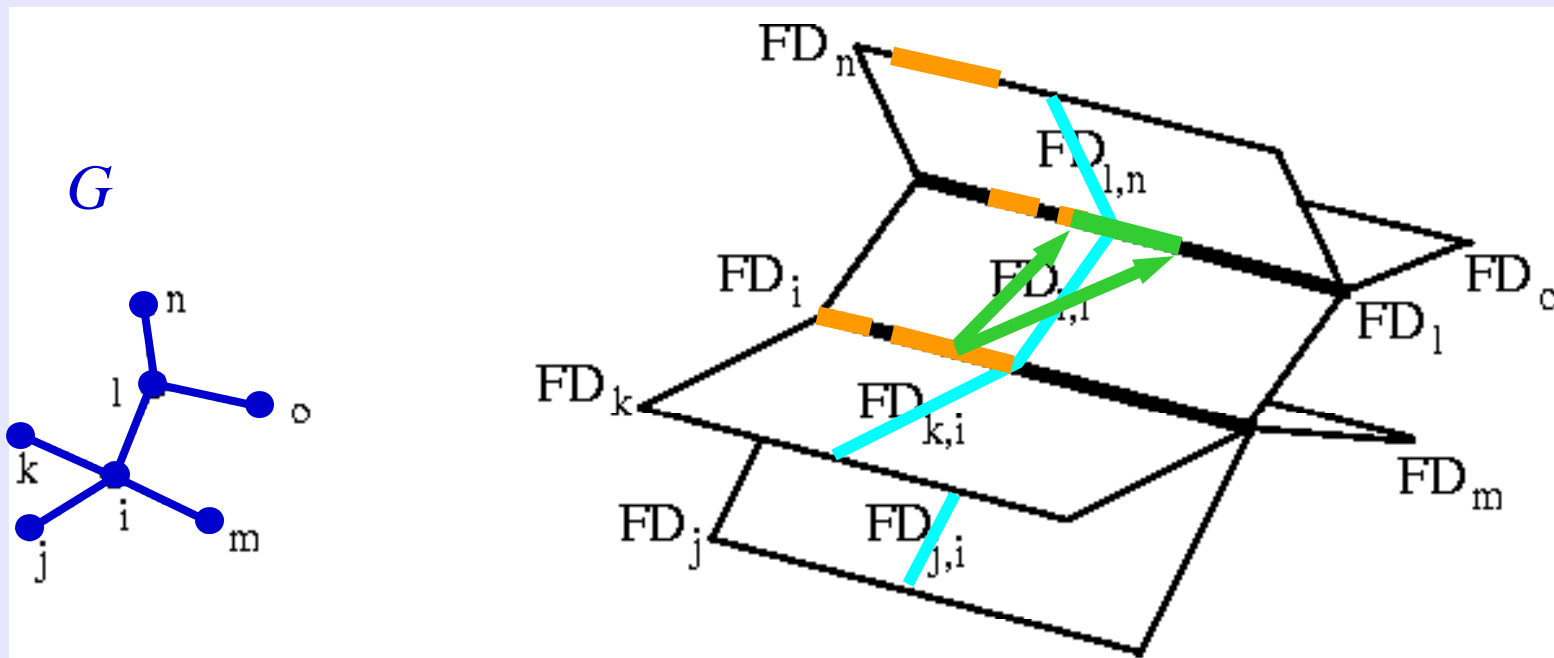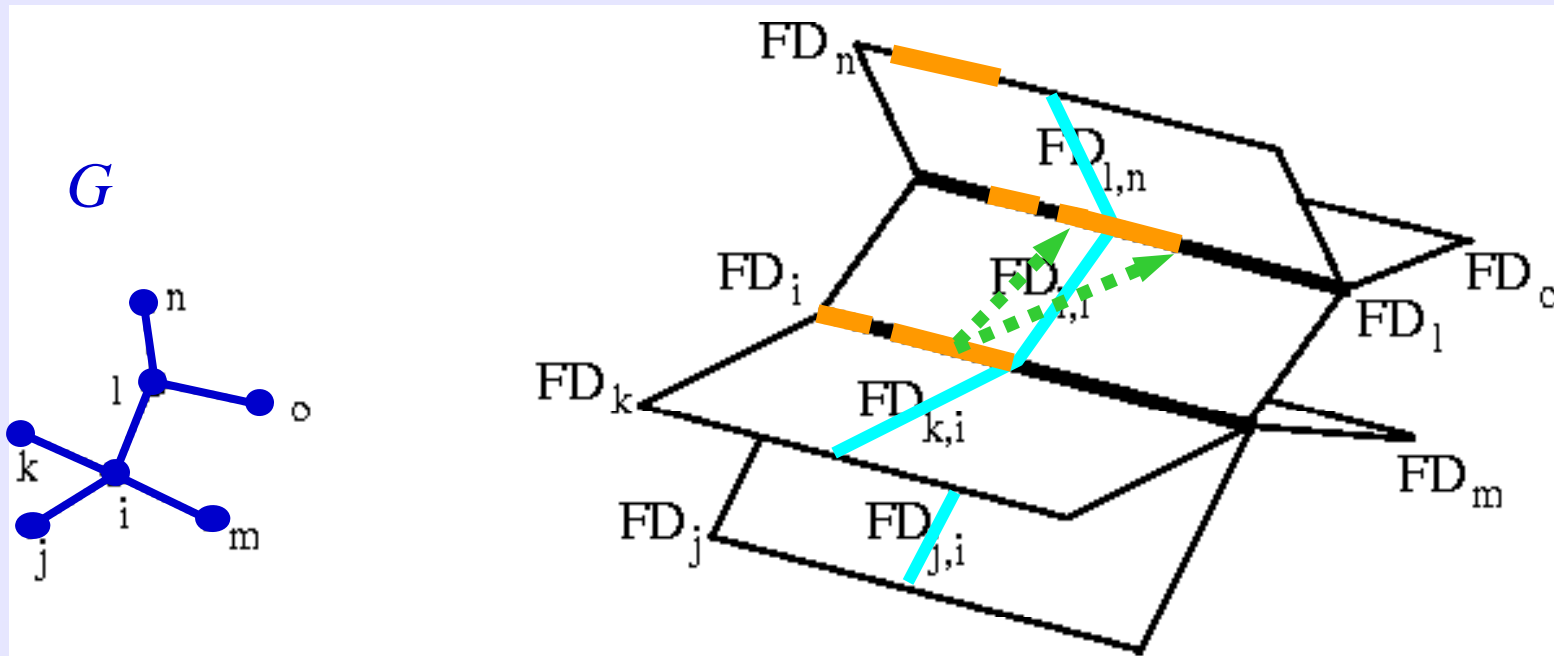
Update Dijkstra Style

- **During sweep:**
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface

Update Dijkstra Style

- **During sweep:**
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface

Update Dijkstra Style

- **During sweep:**
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface
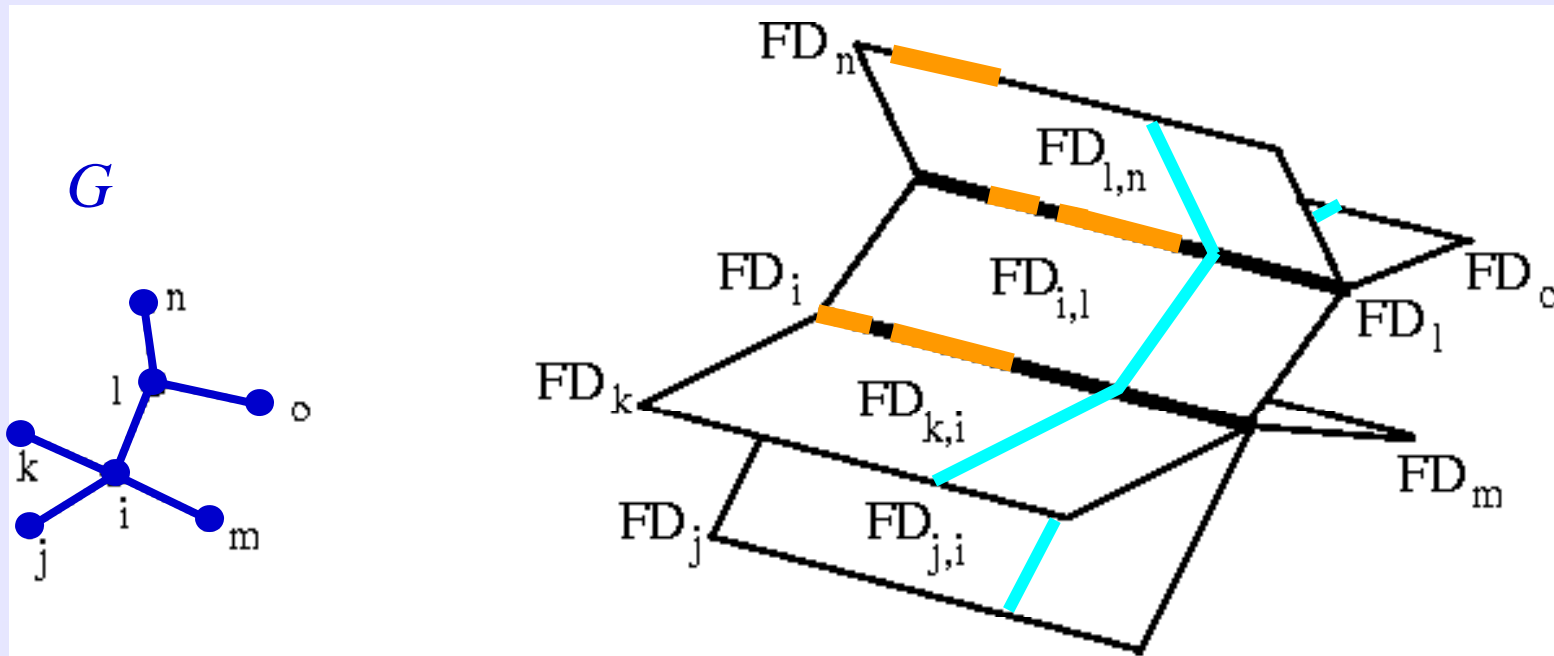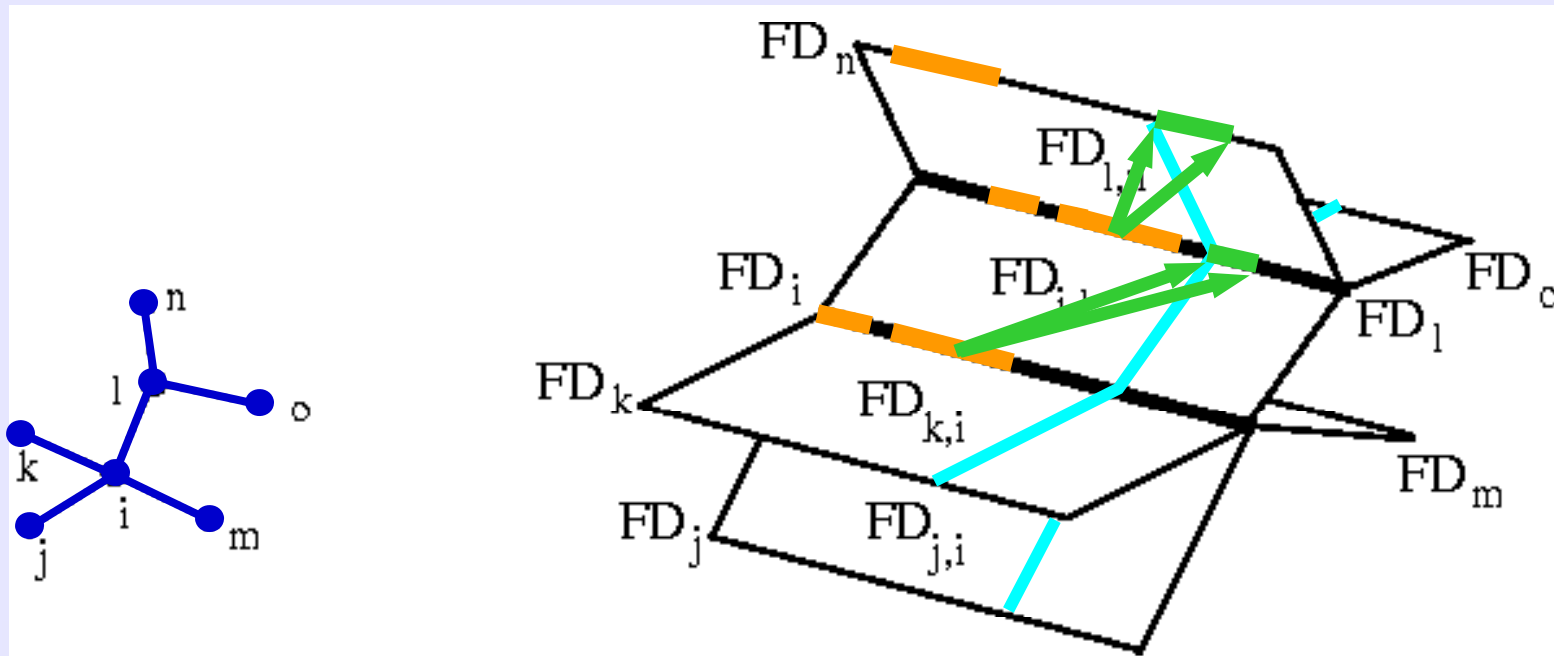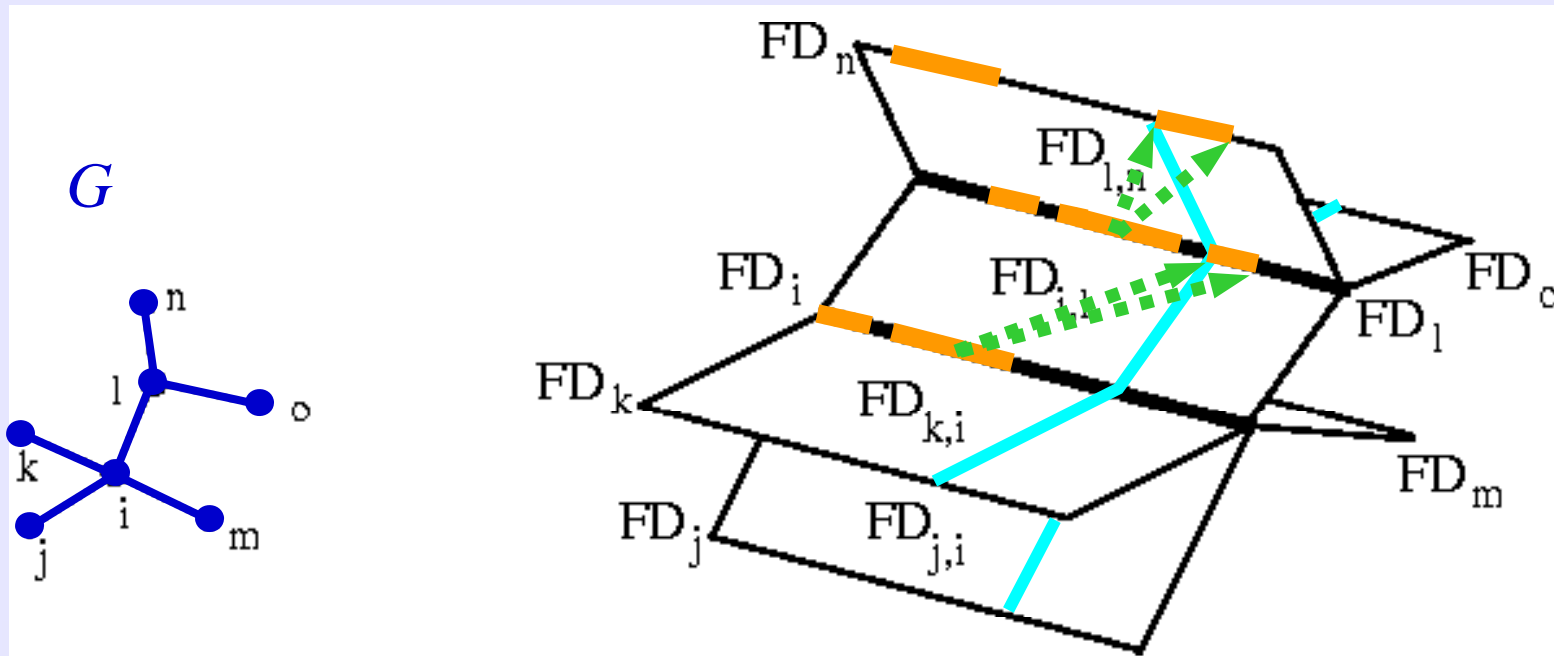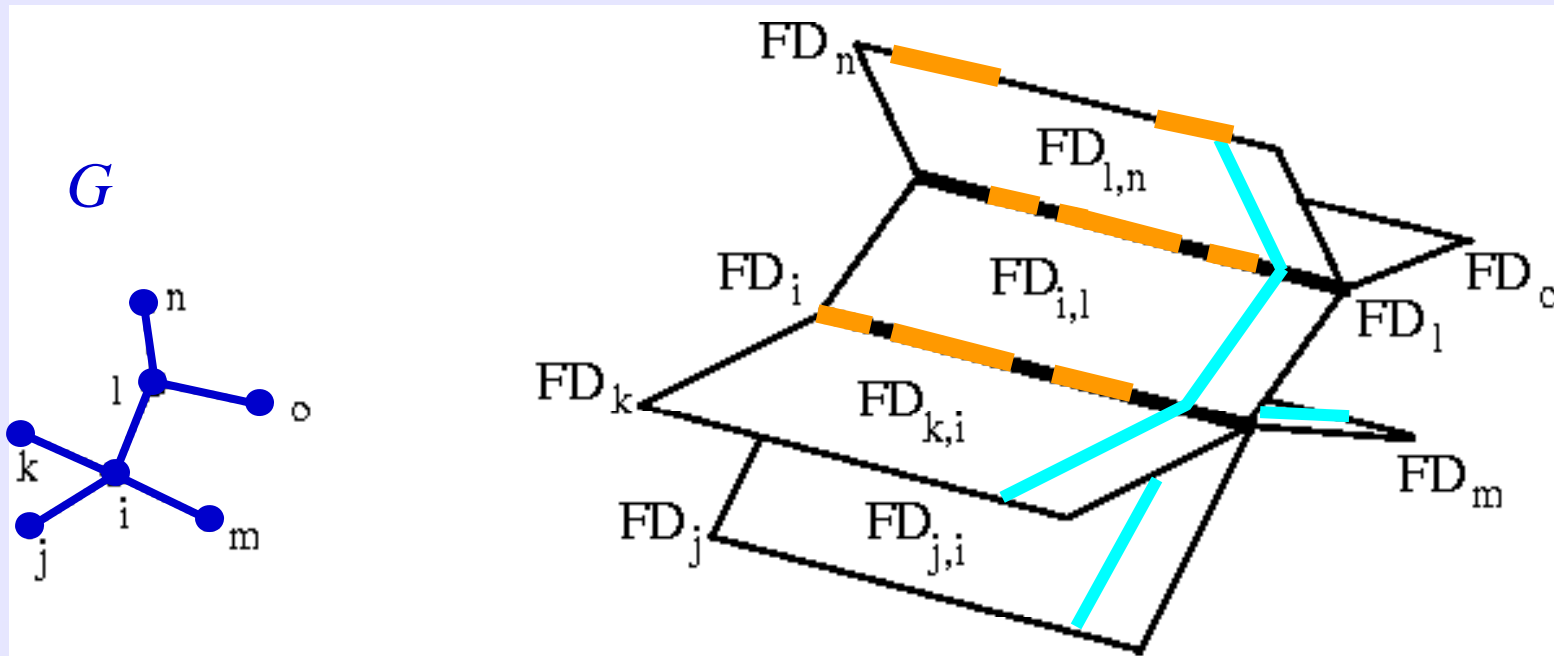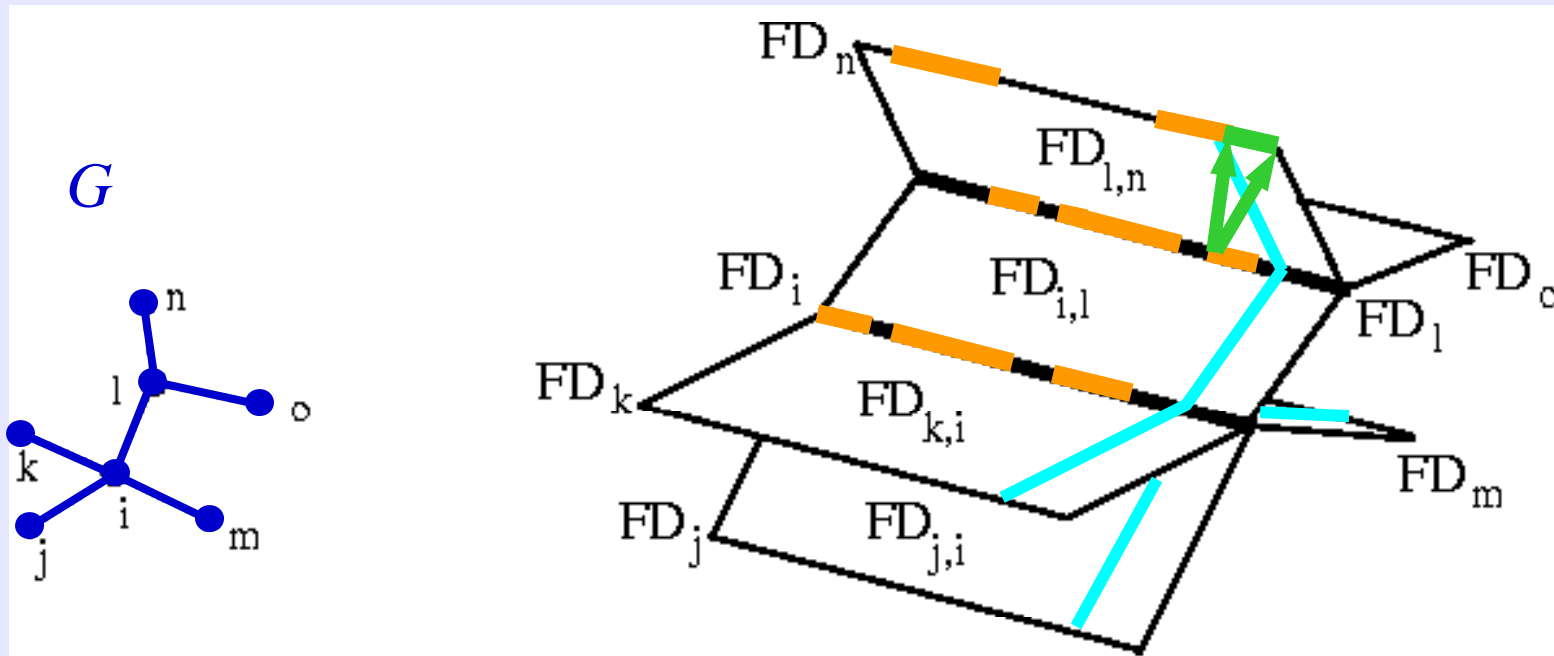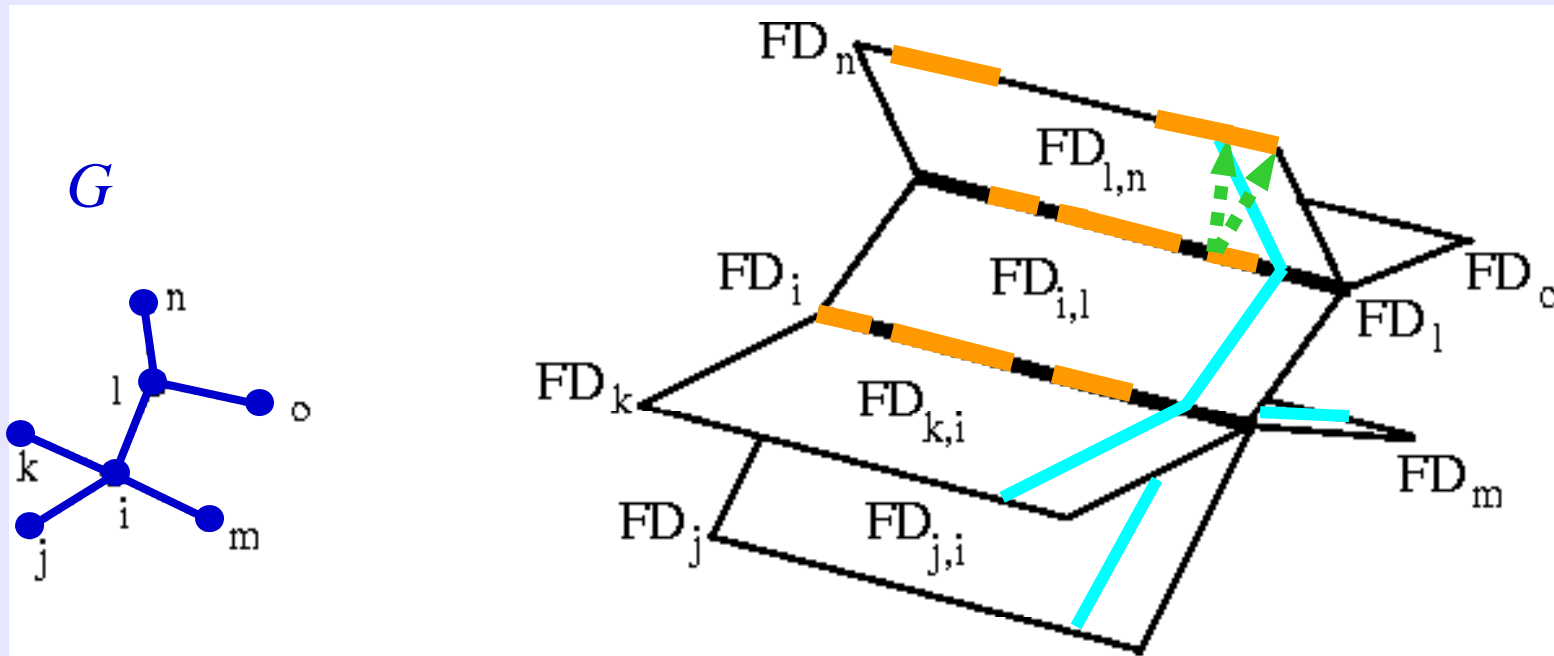
Update Dijkstra Style

- ## During sweep:
  - Update reachable points Dijkstra style
  - Use data structure which supports reachability queries on the free space surface

Update Dijkstra Style

- **During sweep:**
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface

Update Dijkstra Style

- ## During sweep:
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface

Update Dijkstra Style

- ## During sweep:
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface

Update Dijkstra Style
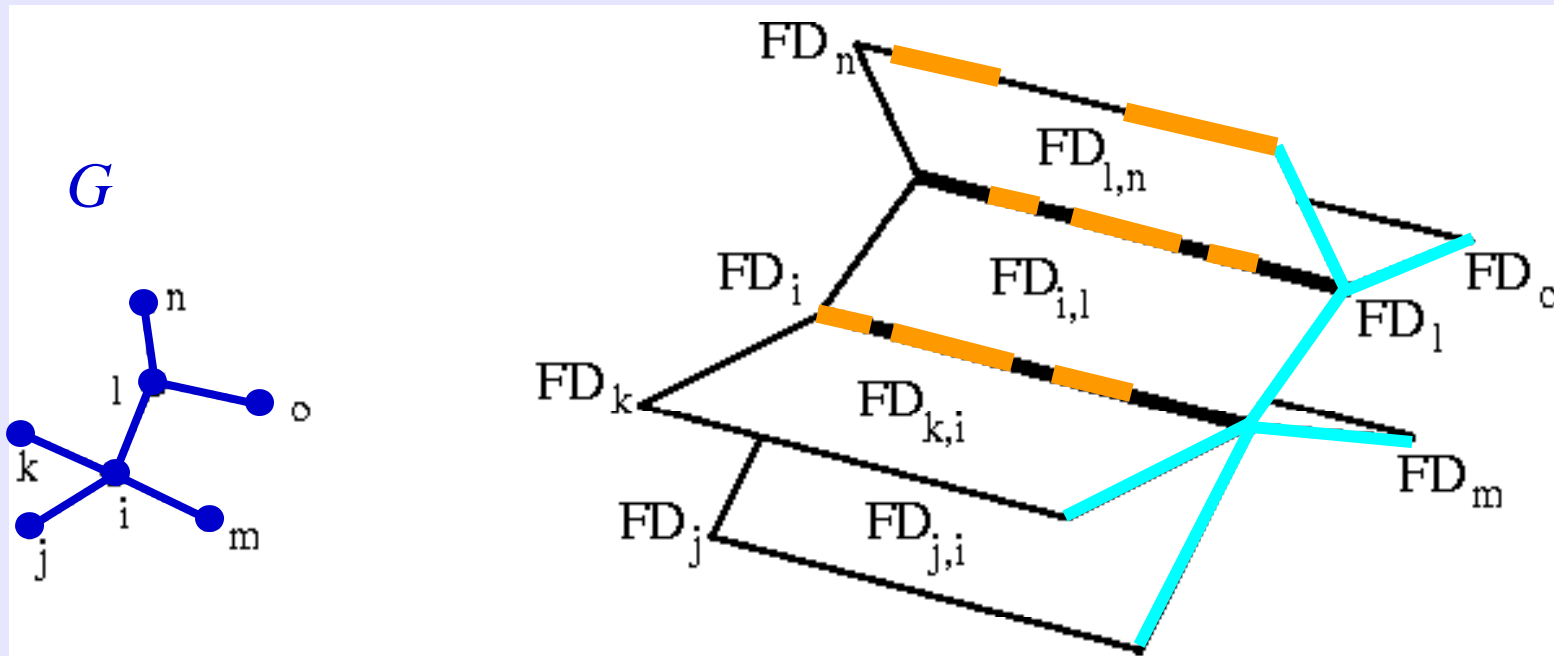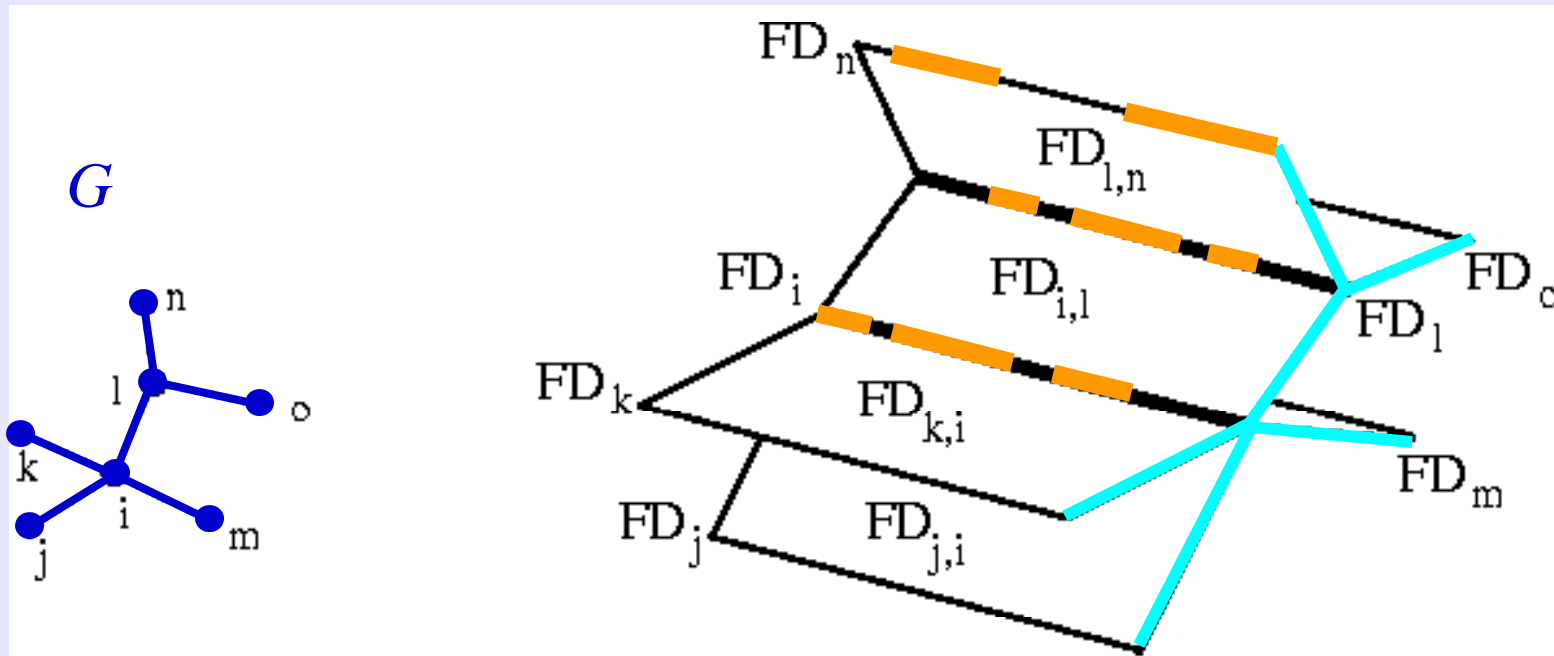

- ## During sweep:
  - **Update reachable points** Dijkstra style
  - Use data structure which supports **reachability queries** on the free space surface
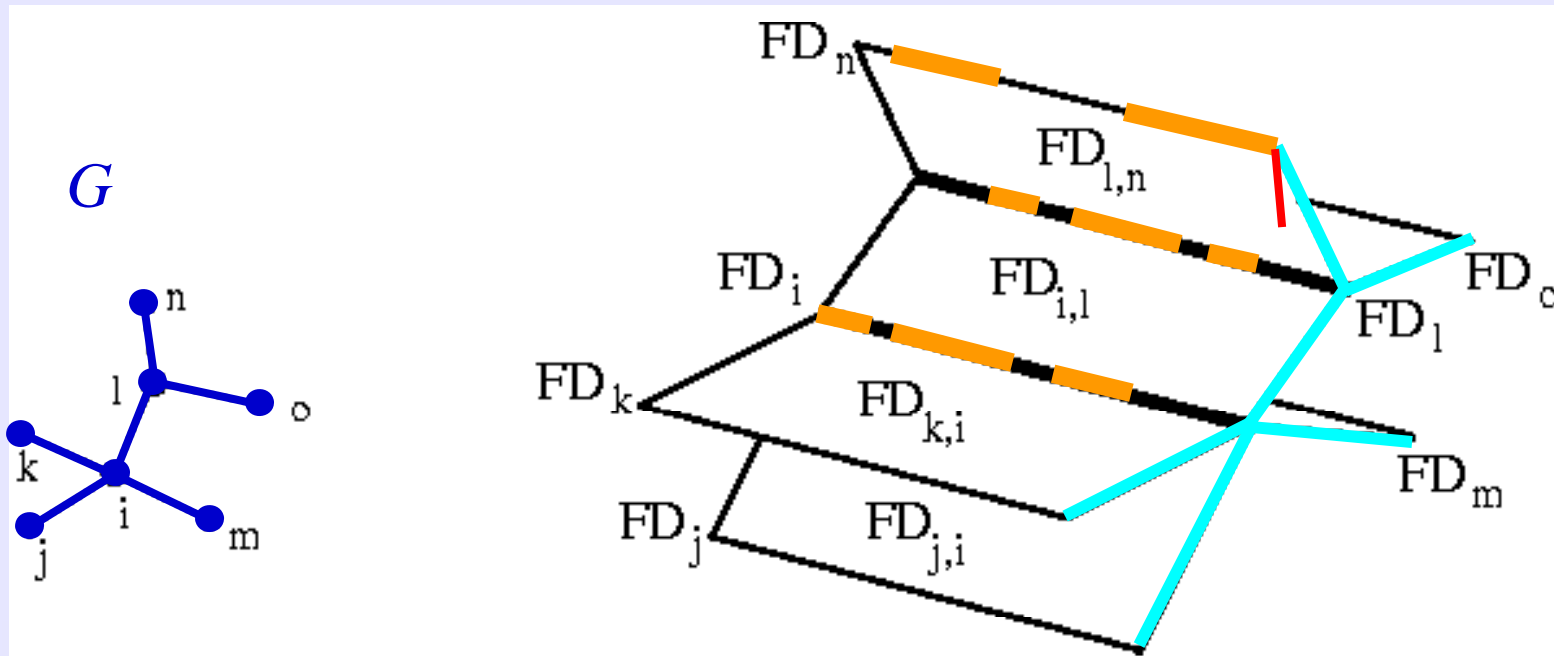
Update Dijkstra Style

- **During sweep:**
  - **Update reachable points** Dijkstra style
  - Use data structure which supports reachability queries on the free space surface
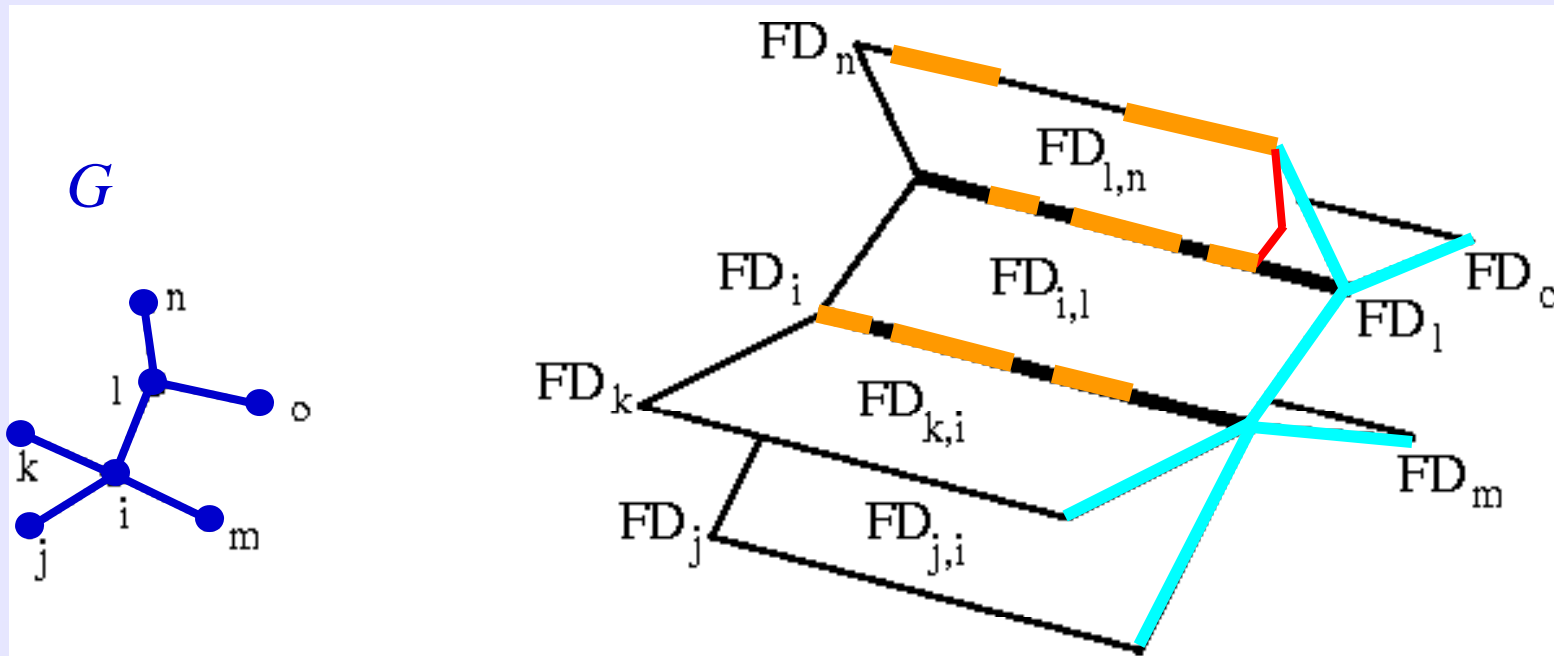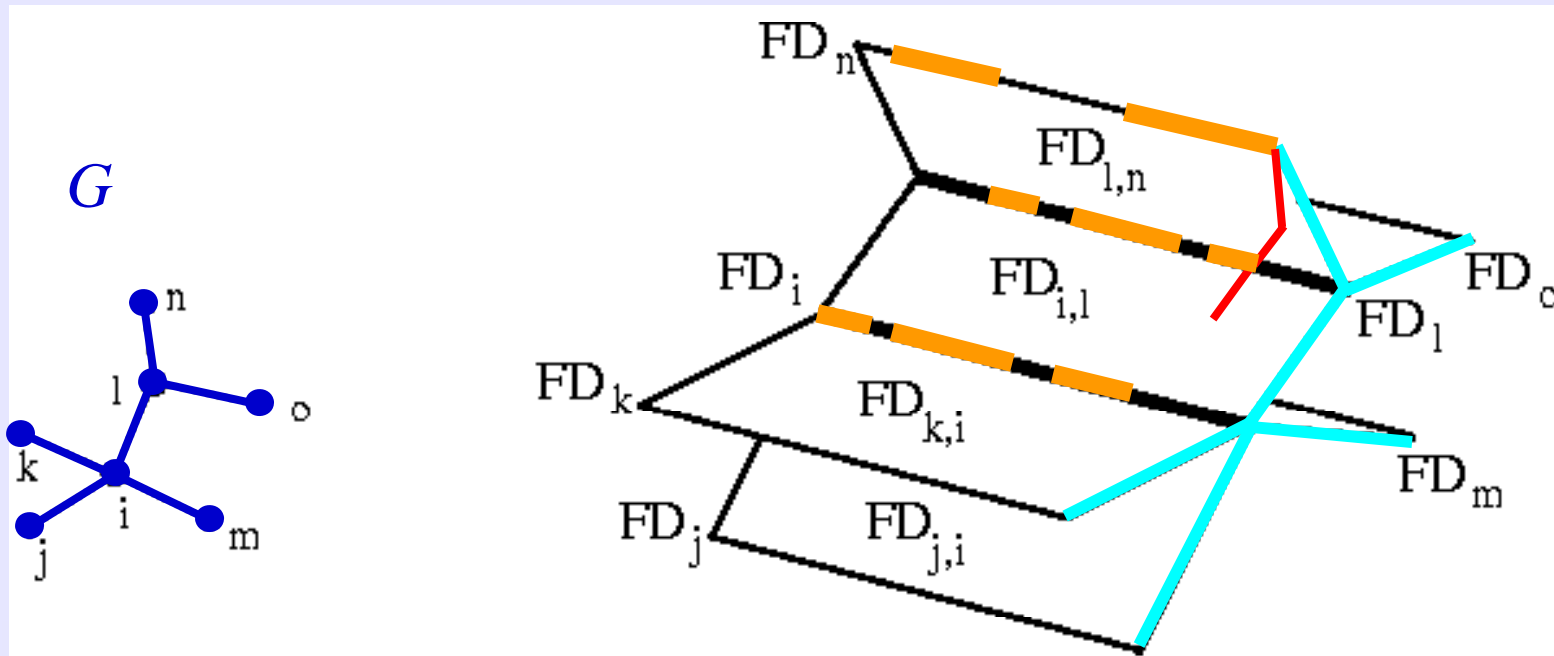
Backtracking

- ## After sweep:
    - Construct a **monotone path** by backtracking

Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

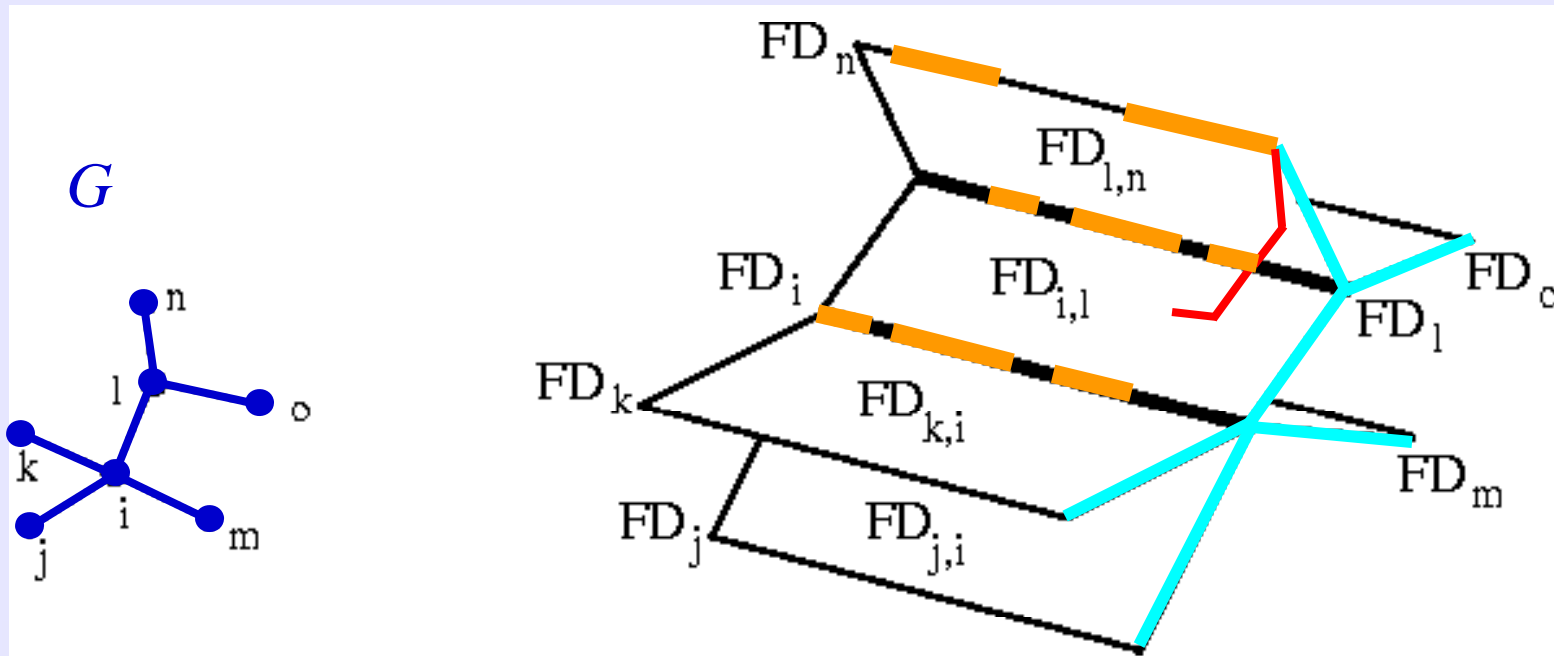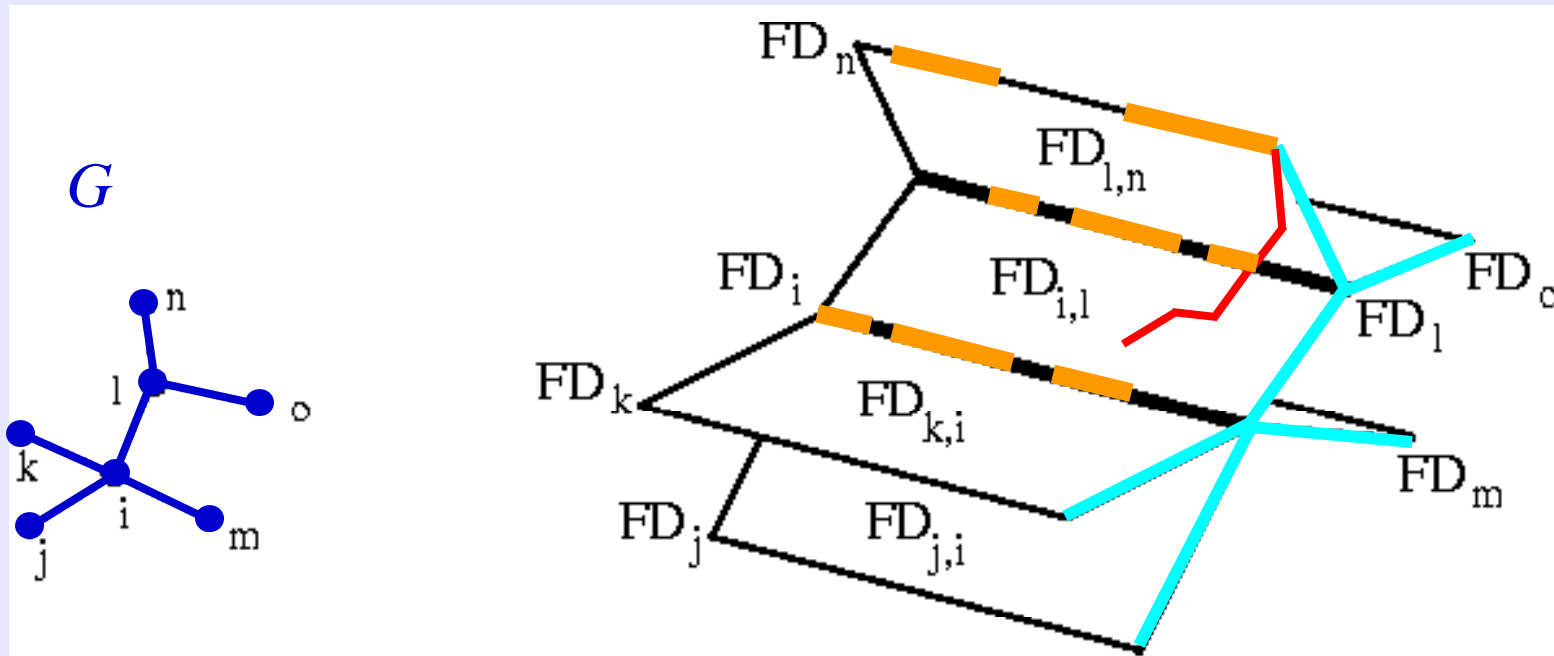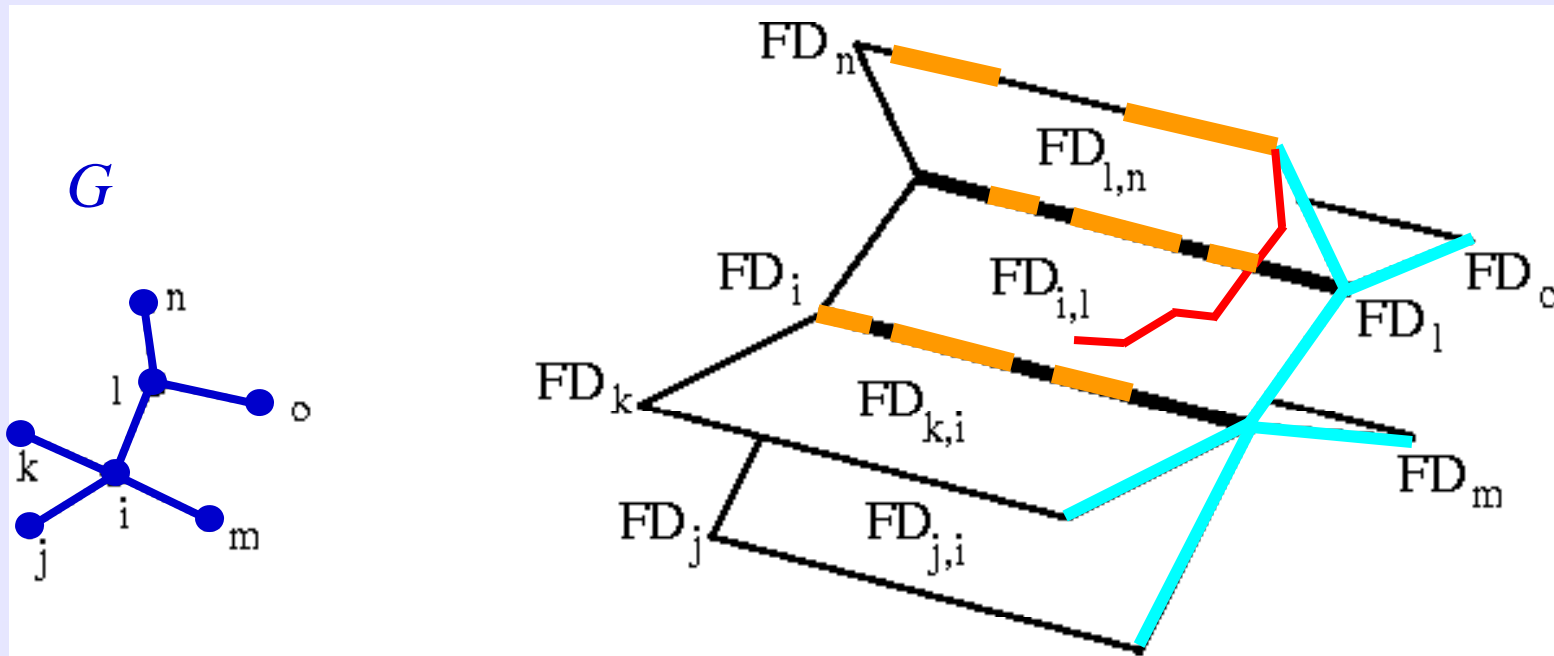- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- **After sweep:**
  - Construct a **monotone path** by backtracking
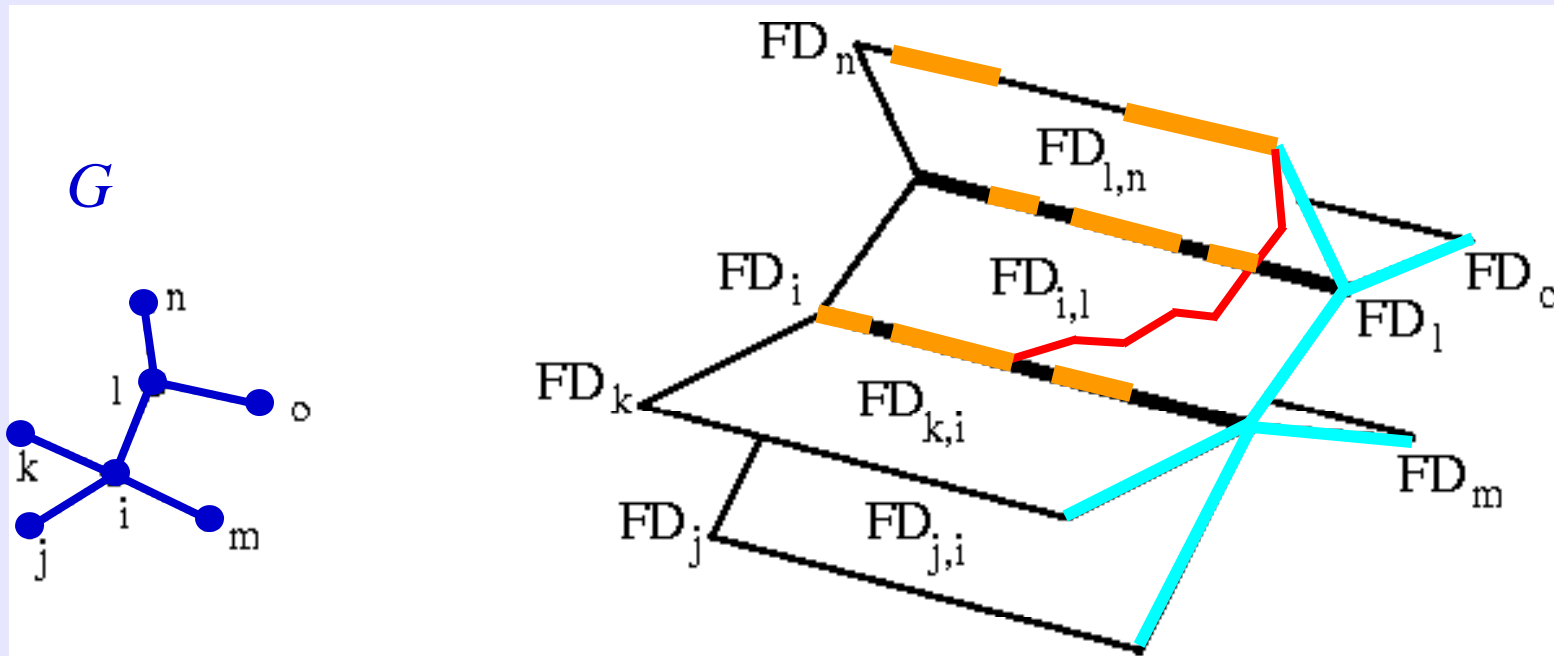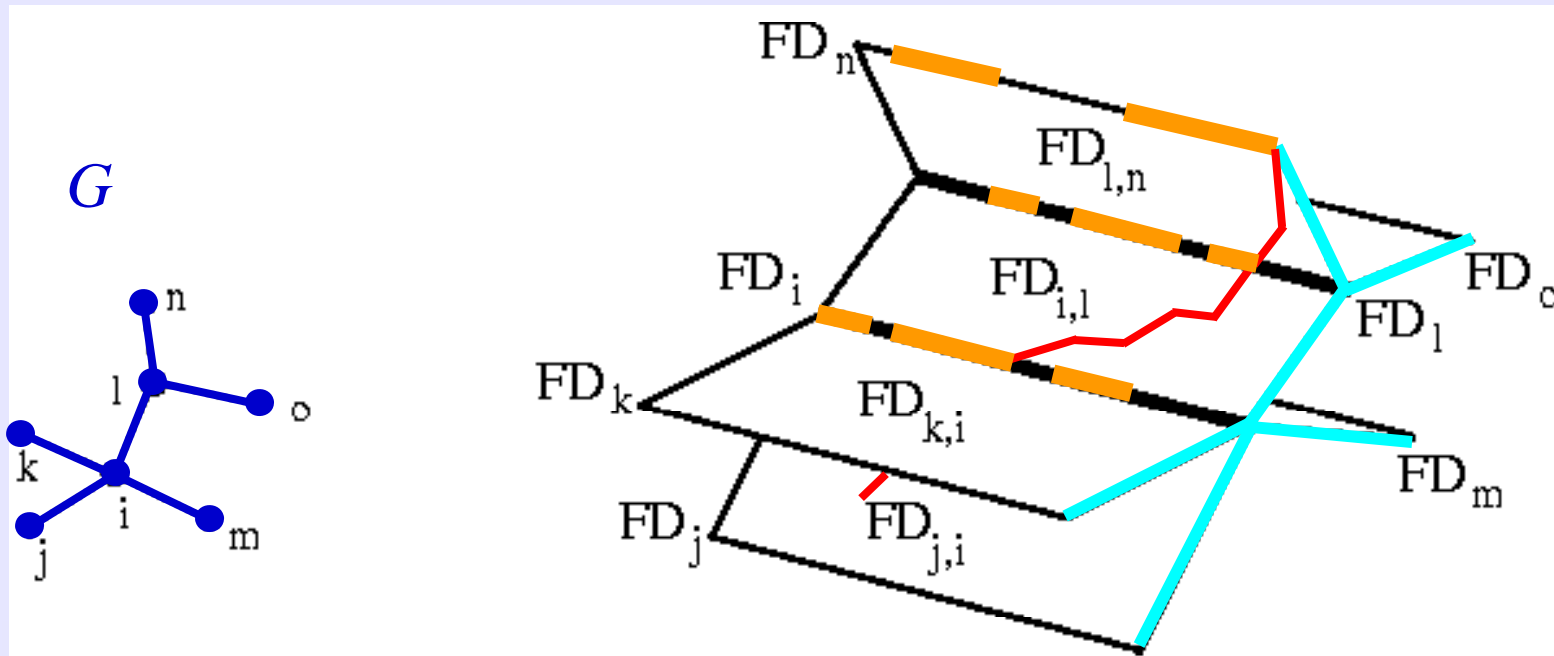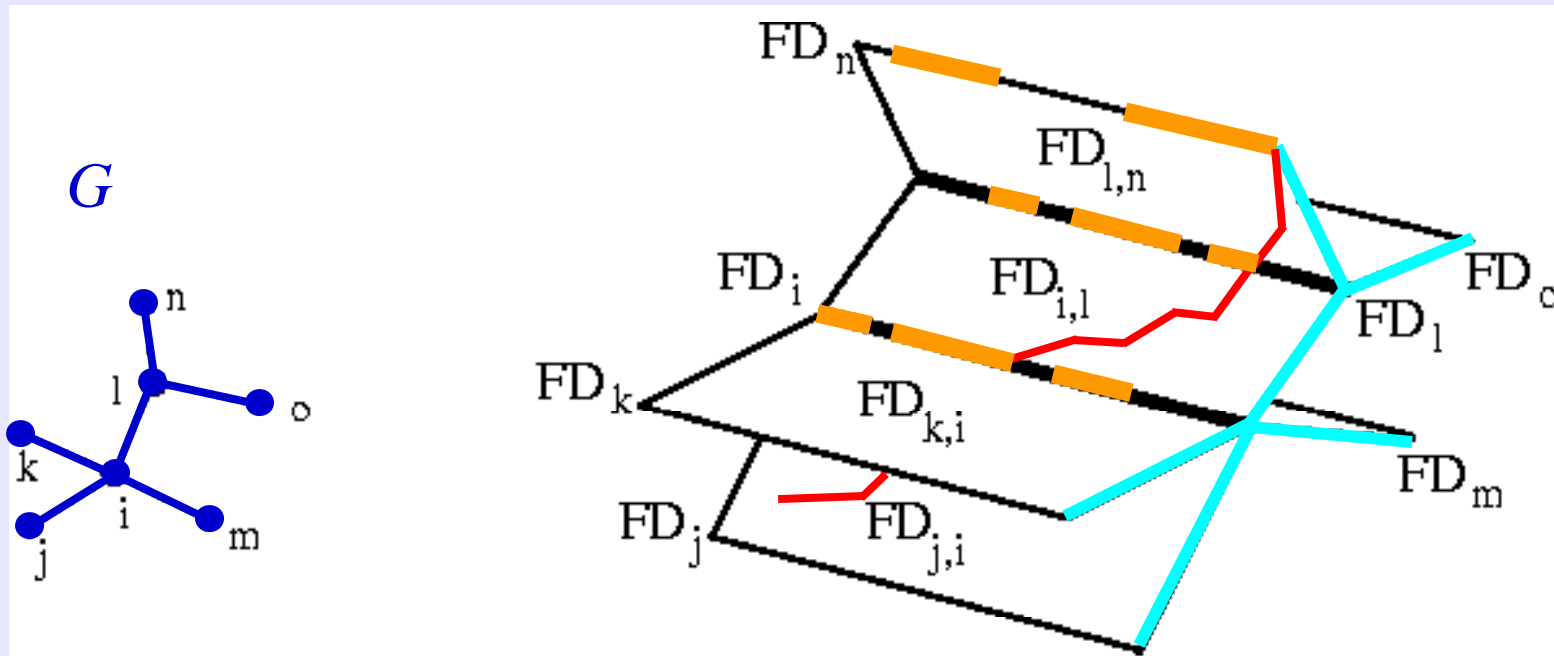
Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- ## After sweep:
  - Construct a **monotone path** by backtracking

Backtracking

- **After sweep:**
  - Construct a **monotone path** by backtracking

Backtracking

- **After sweep:**
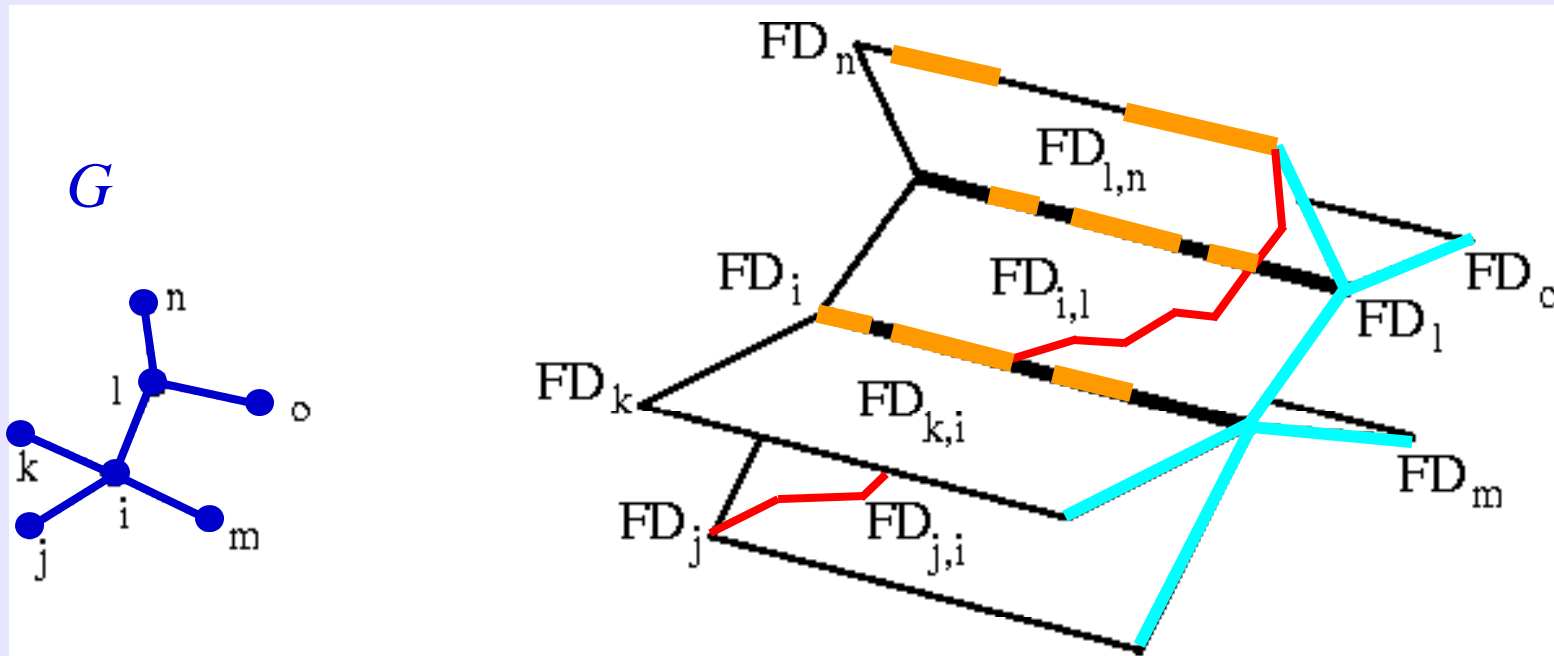 - Construct a **monotone path** by backtracking