# Graphs and Network Flows
# ISE 411

## Lecture 8

Dr. Ted Ralphs

# Why study shortest path problems?

- They arise frequently in practice and as subproblems in higher-level algorithms.
- They are easy to solve efficiently.
- They capture most essential features of the broader class of network flow models.
- SPP is a good starting point for introducing ideas for designing algorithms with good worst-case performance.

# Example: The Kevin Bacon Number

- Start with any actor or actress who has been in a movie and connect him or her to Kevin Bacon in the smallest number of links.
- Two people are linked if they have appeared in a movie together.
- Play on the web! `http://OracleOfBacon.org`.
- Example
  - Alfred Hitchcock and Orson Welles: *Show Business At War (1943)*
  - Orson Welles and Jack Nicholson: *A Safe Place (1971)*
  - Jack Nicholson and Kevin Bacon: *A Few Good Men (1992)*
- How is the shortest path problem related to the Oracle of Kevin Bacon?

# Formal Definition

**Definition 1.** *Given a directed network* $G = (N, A)$ *with an arc length* $c_{ij}$ *associated with each arc* $(i, j) \in A$ *and a distinguished node* $s$*, the* shortest path problem *is to determine a shortest length directed path from node* $s$ *to every node* $i \in N - \{s\}$*.*

- The length of a directed path is the sum of the lengths of arcs in the path.
- $A(i)$ is the arc adjacency list of node $i$
- $C = \max\{c_{ij} : (i, j) \in A\}$

# Variations

- Single-Source: from one node to every other
  - Non-negative arc lengths
  - Arbitrary arc lengths
- All Pairs: from every node to every other
- Maximum Capacity
- Maximum Reliability
- $\cdots$

# Assumptions

- The network is directed.
- All arc lengths are integers.
- There is a directed path from node $s$ to every other node in the network.
- There are no directed cycles with negative length.

# Shortest Paths Tree

Associated with any shortest path problem is a directed out-tree called the *shortest path tree* rooted at node $s$ with the property that the unique path from node $s$ to any node is a shortest path to that node.

**Property 1.** *If the path* $s = i_1 - i_2 - \cdots - i_h = k$ *is a shortest path from node* $s$ *to node* $k$, *then for every* $q = 2, 3, \cdots, h-1$, *the subpath* $s = i_1 - i_2 - \cdots - i_q$ *is a shortest path from the source node to node* $i_q$.

**Property 2.** *Let the vector* $d$ *represent the shortest path distances. Then a directed path* $P$ *from the source node to node* $k$ *is a shortest path if and only if* $d(j) = d(i) + c_{ij}$ *for every arc* $(i, j) \in P$.

From these two properties, we can prove that a shortest paths tree always exists.

# Application (Ahuja et al., 4.13)

- The owner of a home for senior citizens has decided to offer shuttle bus service for her residents to downtown Bethlehem.
- She has interviewed a number of potential drivers whose available hours and wages are shown in the table.
- She needs to ensure that at least one driver is on duty for each hour between 9 A.M. and 5 P.M., and she would like to pay as little as possible.
- How should we solve her scheduling problem?

| Duty Hours | 9 - 1 | 9 - 11 | 12 - 3 | 12 - 5 | 2 - 5 | 1 - 4 | 4 - 5 |
|---|---|---|---|---|---|---|---|
| Cost | 30 | 18 | 21 | 38 | 20 | 22 | 9 |

# Shortest Path Algorithms

- Label Setting (Chapter 4)
  - one label becomes permanent during each iteration
  - acyclic with arbitrary arc lengths OR non-negative arc lengths
- Label Correcting (Chapter 5)
  - all labels are temporary until last iteration
  - more general graphs including negative arc lengths
- Both are iterative; they differ in label update procedure and convergence procedure.

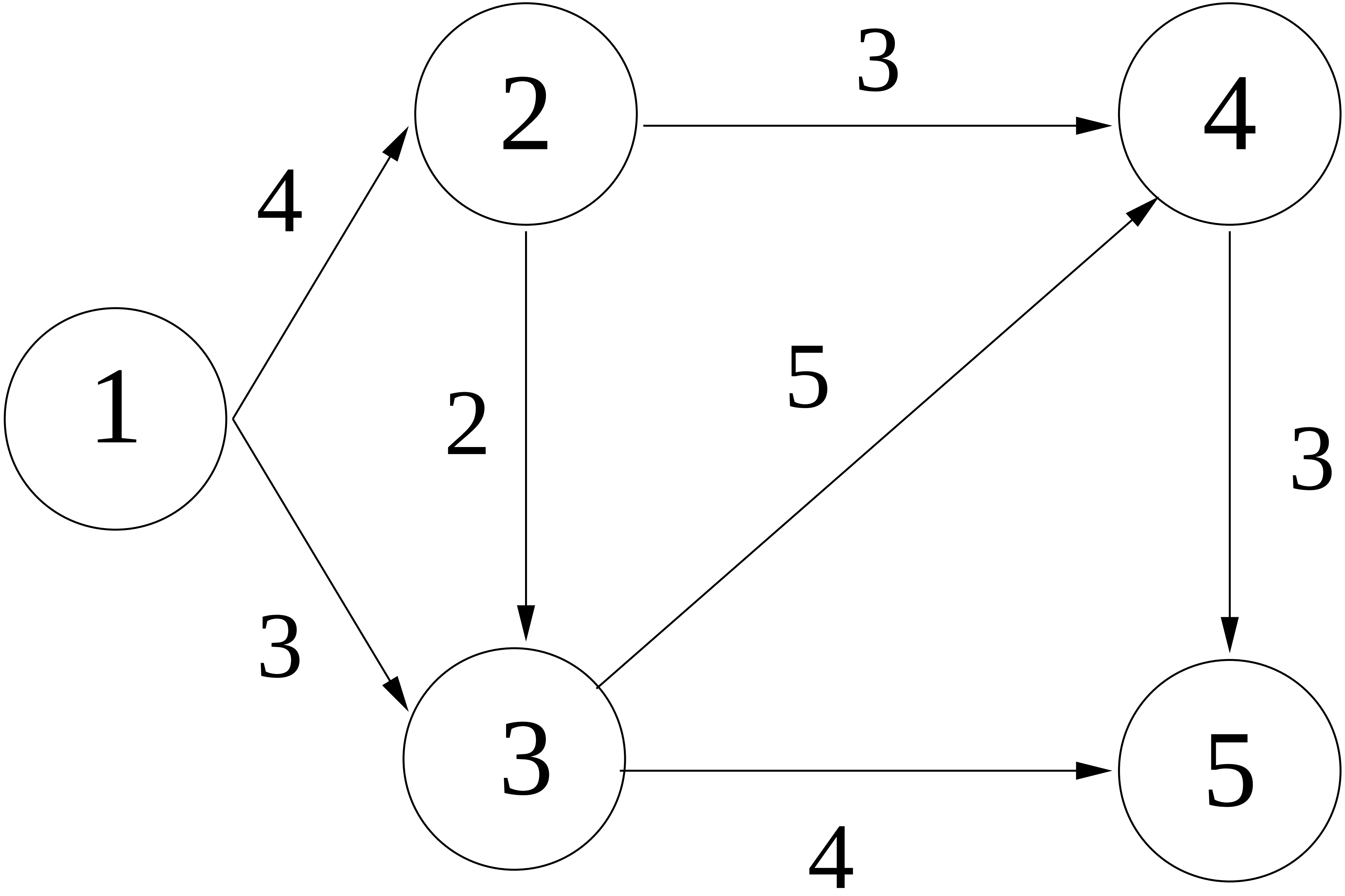
# Solving SPP in Acyclic Networks

Reaching Algorithm

**Input:** An acyclic network $G = (N, A)$ and a vector of arc lengths $c \in \mathbb{Z}^A$
**Output:** $d(i)$ is the length from of a shortest path from node $s$ to node $i$

```
$d(s) \leftarrow 0$
for $i \in N - \{s\}$ do
    $d(i) \leftarrow \infty$
end for
Determine a topological order order[] of the nodes
for $k = 1$ to $n$ do
    $i \leftarrow$ order$[k]$
    for $(i, j) \in A(i)$ do
        if $d(j) > d(i) + c_{ij}$ then
            $d(j) \leftarrow d(i) + c_{ij}$
        end if
    end for
end for
```

# Example of Reaching Algorithm



# Proof of Correctness for Reaching Algorithm

**Claim**: When the algorithm examines a node, its distance label is optimal.

**Proof**

Base Case: When we examine node 1, $d(1) = 0$ is correct. When we examine node 2, $d(2) = d(1) + c_{12}$ is correct because only node 1 can be inbound to node 2 (topological ordering).

Induction: Suppose that the algorithm has examined nodes $1, 2, \cdots, k$ and the distance labels are correct. Show that the distance label for node $k + 1$ is correct.

Let the shortest path from $s$ to $k + 1$ be $s - i_1 - i_2 - \cdots - i_h - i_k + 1$.