

CSE 461: Introduction to Computer Communications Networks Autumn 2006

Module 7 Routing Overview

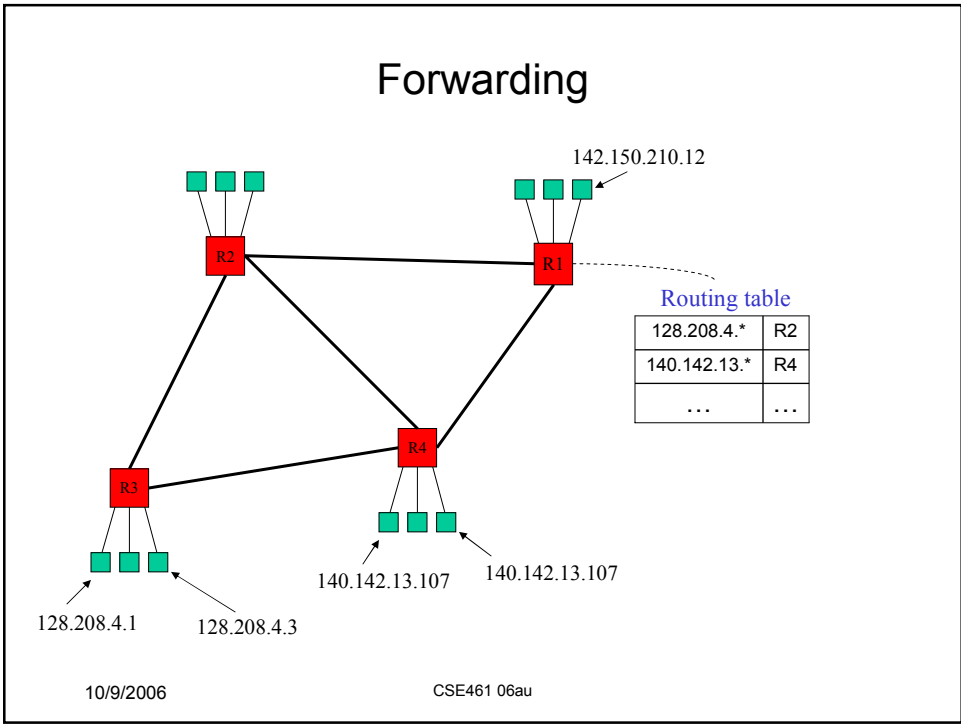
John Zahorjan
zahorjan@cs.washington.edu
534 Allen Center

This Module

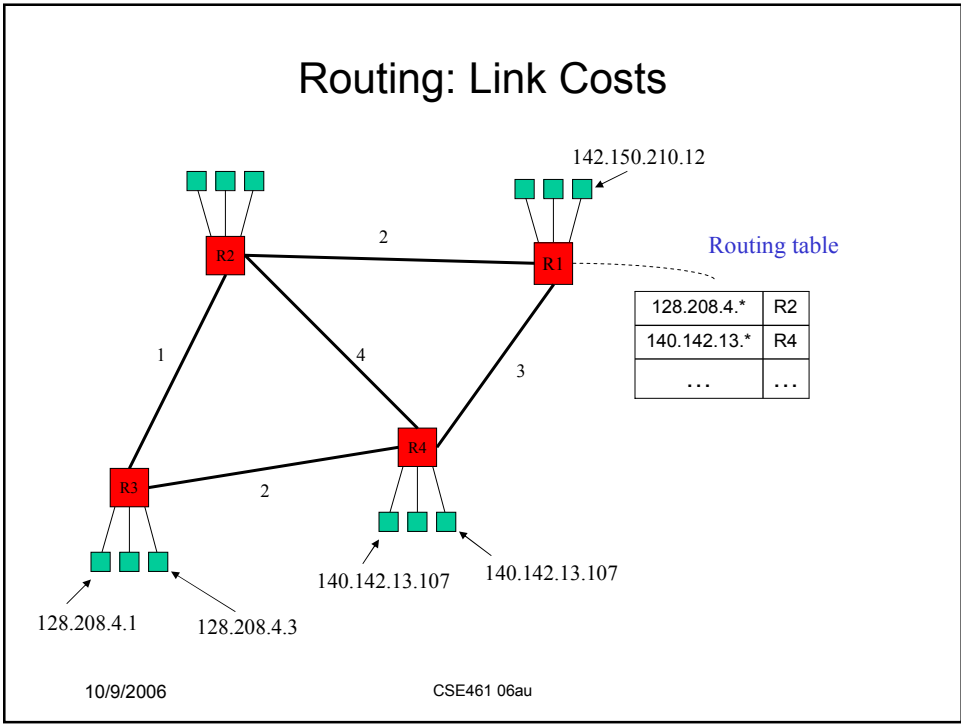
- Review of forwarding
- Overview of approaches
 - Distance Vector Routing
 - Link State Routing

Application
Presentation
Session
Transport
Network
Data Link
Physical

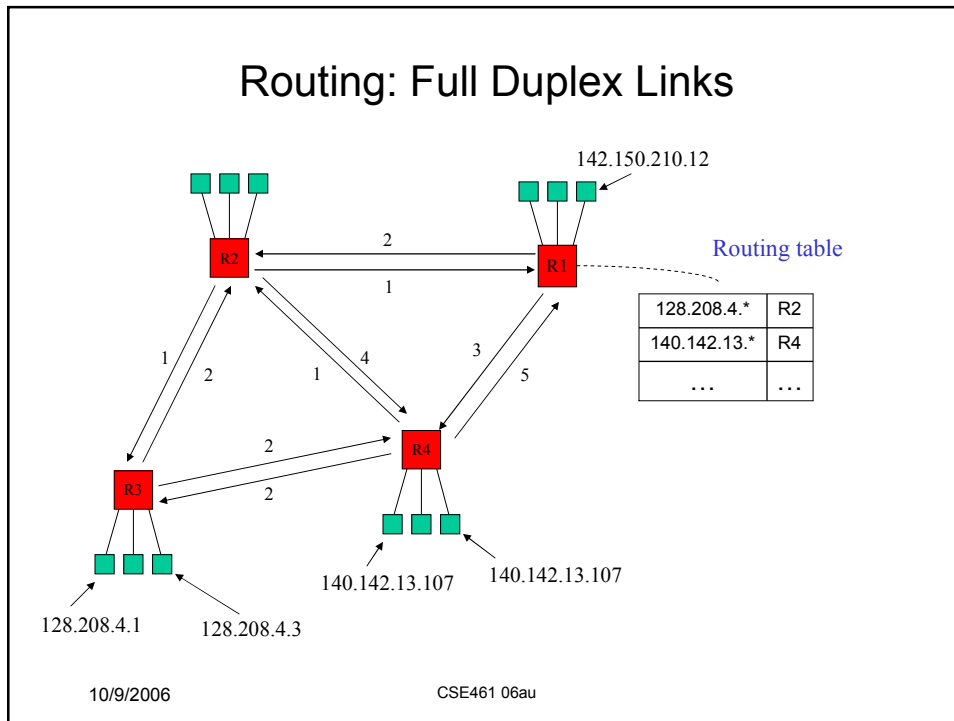
Forwarding



Routing: Link Costs



Routing: Full Duplex Links



Routing as a Shortest Path Problem

- Routing table entries: [destination network, next hop router]
- To decide which router is on the next hop, want to find the shortest path from the router to the destination network's router
- We'll first look at sequential solutions, then distributed
 - "Sequential": full network topology information is available
 - "Distributed": must distribute information and perform computation on each router
- We'll first look at the single-destination / all-sources problem, then all-destinations / all-sources
- One thing to look for:
 - each router obtains a consistent view
 - am forwarding on shortest path
 - shortest paths don't have loops!

10/9/2006

CSE461 06au

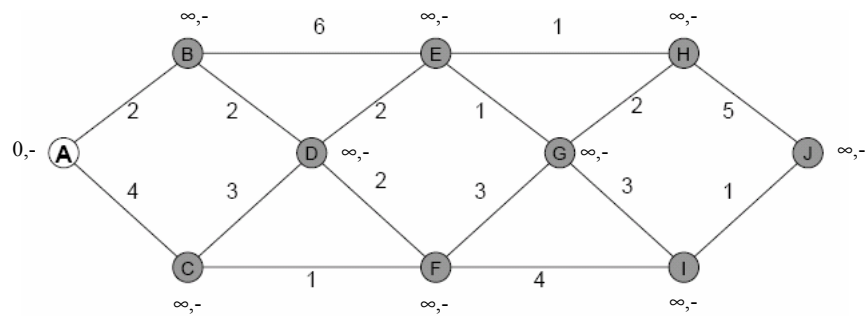
First Approach: Iterative

- Bellman-Ford Algorithm
- Iterative:
 - At each step, update [cost, next hop] for every router based on [cost] at neighbors
 - Starting conditions:
 - [0, -] at destination
 - [∞ , -] at every other router
- Running time: $O(VE)$
 - V: number of vertices (routers)
 - E: number of edges (links)

10/9/2006

CSE461 06au

Bellman-Ford Example

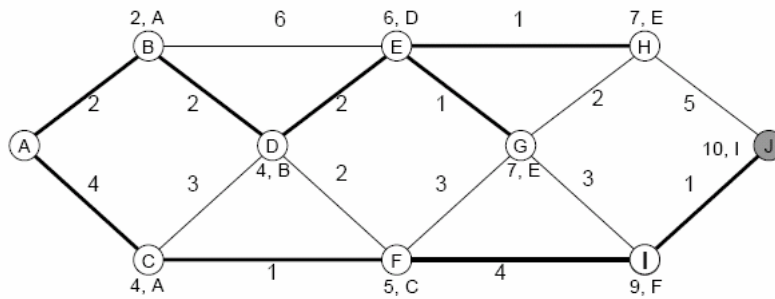


How long can it take to converge?

10/9/2006

CSE461 06au

Result



Note: The result is a spanning tree rooted at the destination

10/9/2006

CSE461 06au

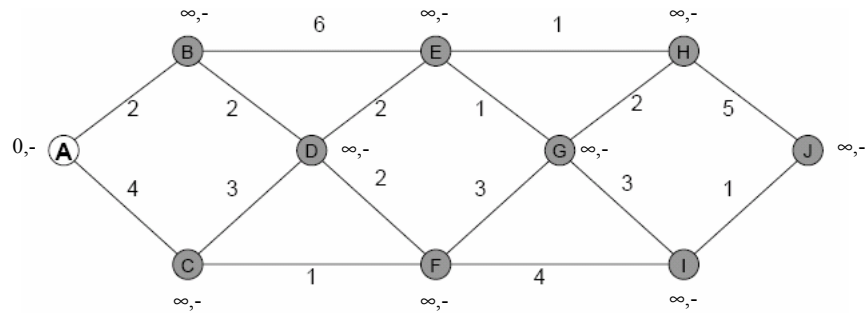
Second Approach: Greedy

- [Dijkstra's Algorithm](#)
- Greedy:
 - Build the spanning tree by adding routers to the current spanning tree one at a time
 - Choose next the as-yet-unadded router whose distance to the destination is minimal
 - Starting conditions:
 - $[0, -]$ at destination
 - $[\infty, -]$ at every other router
 - Spanning tree is the destination router alone
- Running time: $O(E \log V)$

10/9/2006

CSE461 06au

Dijkstra Example



How do we know this works?

10/9/2006

CSE461 06au

Moving to the Internet

- Routing table reflects spanning tree from source to every destination
 - Not really a big change
 - Bellman-Ford: every destination is engaged in the procedure
 - Dijkstra: make the source the root, rather than the destination
- Have to distribute information
 - Bellman-Ford: neighbor information about current costs to each destination
 - Dijkstra: full topology/cost information
- The process is on-going
 - Not all routers boot at once
- Router/link failures can occur
 - Link cost data isn't static

10/9/2006

CSE461 06au