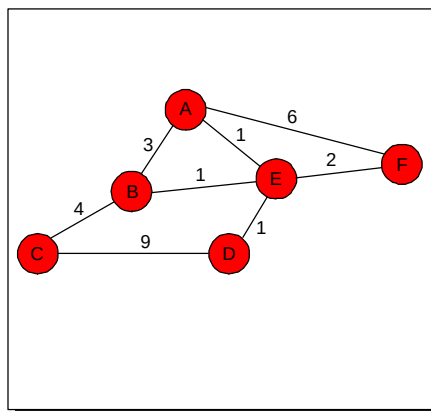


Routing

Arvind Krishnamurthy
Fall 2003

Routing

- Routing algorithms view the network as a graph
- Problem: find lowest cost path between two nodes
- Factors:
 - Static topology
 - Dynamic load
 - Policy
- Two main approaches:
 - Distance-vector protocol
 - Link state protocol



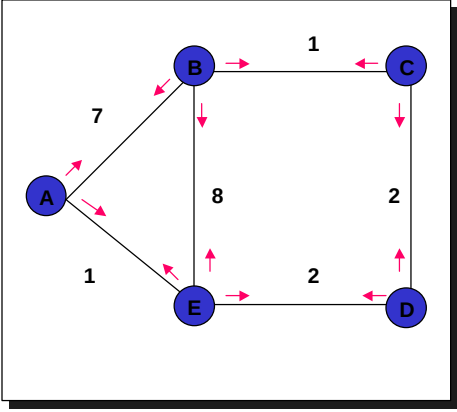


Distributed Bellman-Ford

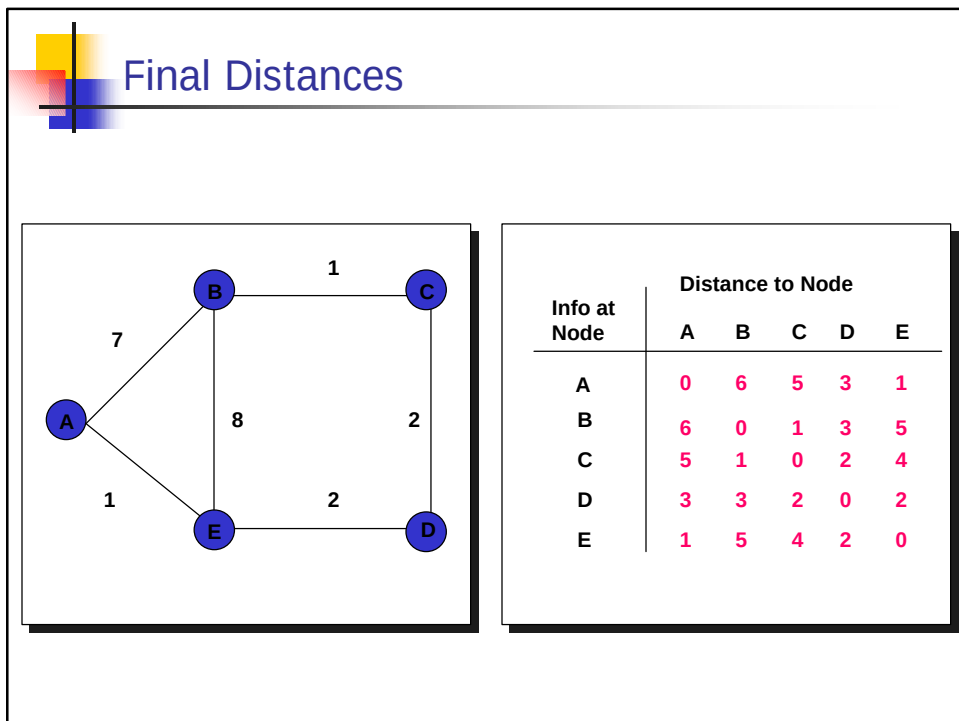
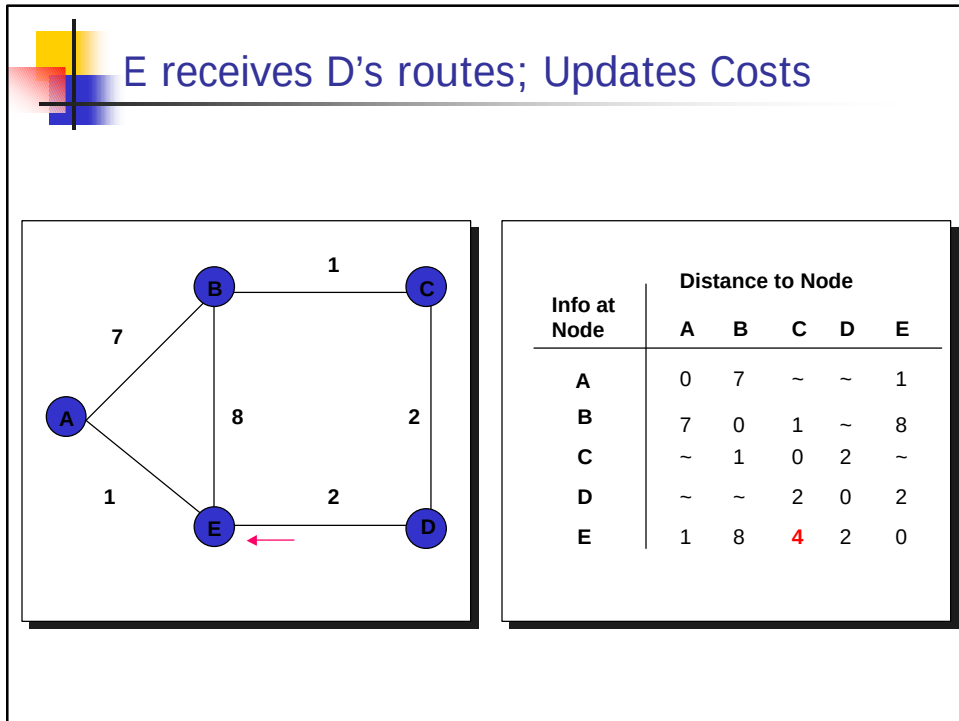
- Start Conditions:
 - Each router starts with a vector of distances to all directly attached networks
- Send step:
 - Each router advertises its current vector to all neighboring routers
- Receive step:
 - For every network X, router finds shortest distance to X
 - Considers current distance to X
 - Then takes into account distance to X from its neighbors
 - Router updates its cost to X
 - After doing this for all X, router goes to send step

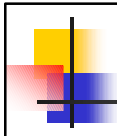


Example - Initial Distances



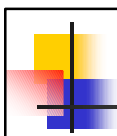
Info at Node	Distance to Node				
	A	B	C	D	E
A	0	7	~	~	1
B	7	0	1	~	8
C	~	1	0	2	~
D	~	~	2	0	2
E	1	8	~	2	0





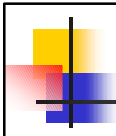
Complexity

- Each message size: $O(n)$
- Number of messages in each step: $O(m)$
 - Each node sends distance vector to all of its neighbors
- Convergence time?



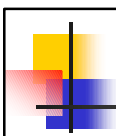
Link State Protocol: Basic Steps

- Start condition
 - Each node assumed to know state of links to its neighbors
- Periodically, each node creates a link state packet containing:
 - Node ID
 - List of neighbors and link cost
 - Sequence number
 - Needed to avoid stale information from flood
 - Time to live (TTL)
 - Node outputs LSP on **all** its links
- **Step 1**
 - Each node broadcasts its state to all other nodes
 - Reliable flooding mechanism
- **Step 2**
 - Each node locally computes shortest paths to all other nodes from global state
 - Dijkstra's shortest path tree (SPT) algorithm



Complexity

- Message size of message from node k :
 - $O(\text{number of edges attached to node } k)$
- Number of messages for each broadcast: $O(m)$
- Total number of messages: $O(nm)$



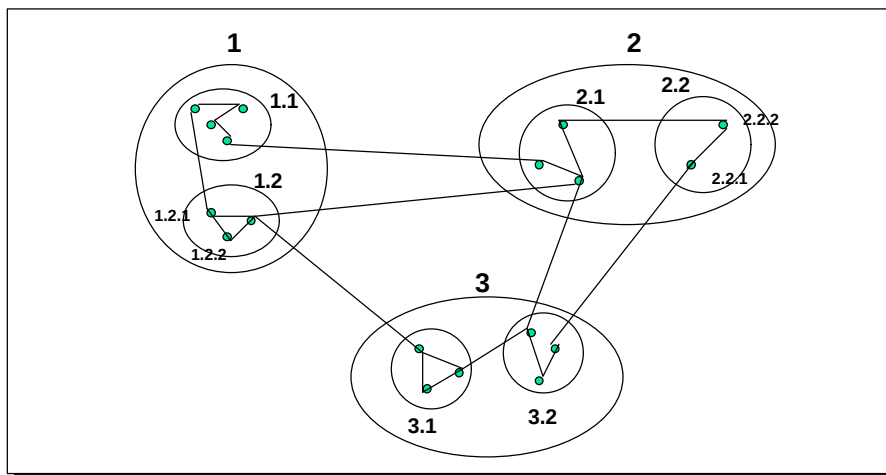
Summary: LS vs. DV

- In DV send everything you know to your neighbors
- In LS send info about your neighbors to everyone
- Msg size: small with LS, potentially large with DV
- Space requirements:
 - LS maintains entire topology
 - DV maintains only neighbor state
- Robustness: DV can have cycles

Routing Hierarchies

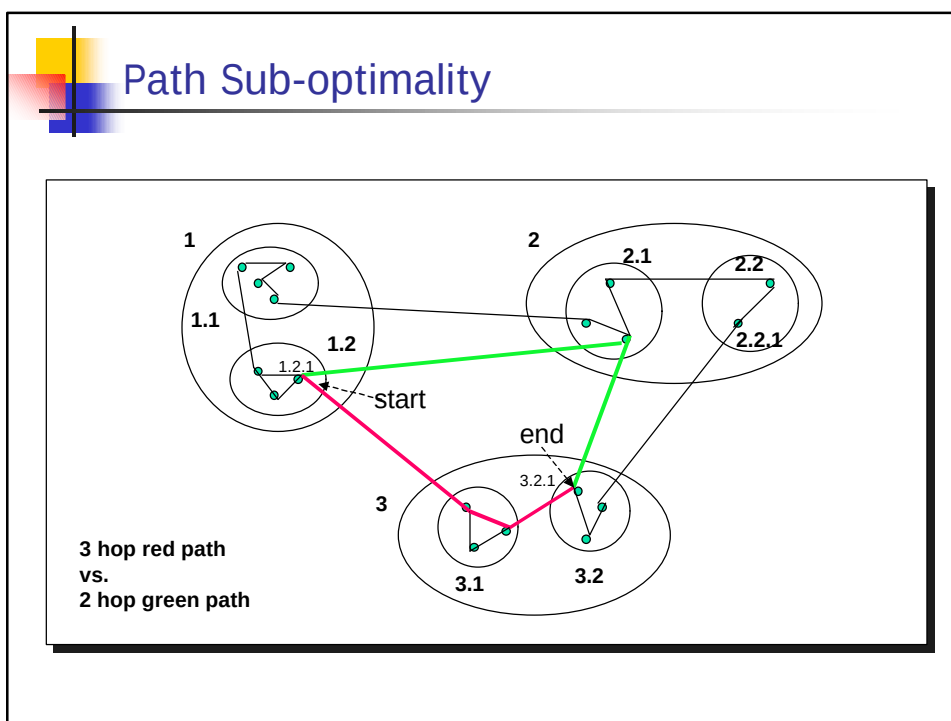
- Flat routing doesn't scale
 - Each node cannot be expected to have routes to every destination (or destination network)
- Key observation
 - Need less information with increasing distance to destination
- Two radically different approaches for routing
 - The area hierarchy
 - The landmark hierarchy
- Area hierarchy:
 - Divide network into areas
 - Areas can have nested sub-areas
 - Constraint: no path between two sub-areas of an area can exit that area
 - Hierarchically address nodes in a network
 - Sequentially number top-level areas
 - Sub-areas of area are labeled relative to that area

The Area Hierarchy



Routing

- Within area
 - Each node has routes to every other node
- Outside area
 - Each node has routes for **other top-level areas only**
 - Inter-area packets are routed to nearest appropriate border router
- Can result in sub-optimal paths





Internet's Area Hierarchy

- What is an Autonomous System (AS)?
 - A set of routers under a single technical administration
 - using an *interior gateway protocol (IGP)* and common metrics to route packets within the AS, and
 - using an *exterior gateway protocol (EGP)* to route packets to other AS's
- Each AS assigned unique ID
- AS's peer at network exchanges