Network Layer:
Routing

CS 356 – University of Texas at Austin

Dr. David A. Bryan
Interplay between routing and forwarding

routing algorithm determines end-end-path through network

forwarding table determines local forwarding at this router

<table>
<thead>
<tr>
<th>header value</th>
<th>output link</th>
</tr>
</thead>
<tbody>
<tr>
<td>0100</td>
<td>3</td>
</tr>
<tr>
<td>0101</td>
<td>2</td>
</tr>
<tr>
<td>0111</td>
<td>2</td>
</tr>
<tr>
<td>1001</td>
<td>1</td>
</tr>
</tbody>
</table>

value in arriving packet’s header

1  2  3
Underlying Algorithms

• Looking at BRIEFLY. Should be familiar from (possibly multiple) earlier classes.

• Some excellent videos (University of Washington) on Dijkstra's algorithm, LS routing, and DV routing:
  – Djikstra's Algorithm:
    https://www.youtube.com/watch?v=elUoBUwIlpQ
  – DV Routing:
    https://www.youtube.com/watch?v=ylzAefKENXY
  – How to flood Updates:
    https://www.youtube.com/watch?v=eziOAAdaun_s
  – LS Routing:
    https://www.youtube.com/watch?v=2_0AwfQWKUk

• See also pp. 363-383 of *Kurose and Ross*, Section 4.5.1 and 4.5.2

• Will do one HW with a very simple DV example to be sure you go work it 😊
Graph abstraction

graph: $G = (N,E)$

$N = \text{set of routers} = \{ u, v, w, x, y, z \}$

$E = \text{set of links} = \{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

(Suspend disbelief for a bit…yes, we can't really do these algorithms over the actual $N$ and $E$ of the Internet…too big, but for now assume we do…)

aside: graph abstraction is useful in other network contexts, e.g., P2P, where $N$ is set of peers and $E$ is set of TCP connections
Graph abstraction: costs

c(x, x') = cost of link (x, x')
e.g., c(w, z) = 5

cost could always be 1, or inversely related to bandwidth, or inversely related to congestion

cost of path (x_1, x_2, x_3, ..., x_p) = c(x_1, x_2) + c(x_2, x_3) + ... + c(x_{p-1}, x_p)

**key question:** what is the least-cost path between u and z?

**routing algorithm:** algorithm that finds that least cost path
Routing algorithm classification

**Q: global or decentralized information?**

**global:**
• all routers have complete topology, link cost info
• “link state” algorithms

**decentralized:**
• router knows physically-connected neighbors, link costs to neighbors
• iterative process of computation, exchange of info with neighbors
• “distance vector” algorithms

**Q: static or dynamic?**

**static:**
• routes change slowly over time

**dynamic:**
• routes change more quickly
  – periodic update
  – in response to link cost changes
Dijkstra's algorithm

• net topology, link costs known to all nodes
  – accomplished via “link state broadcast”
  – all nodes have same info
• iterative: after k iterations, know least cost path to k destinations

• Should have seen this in (probably more than one) class already
• Result is tree with least-cost paths
• This tree is used to make a forwarding table from source node to route traffic
Dijkstra's algorithm: example result

resulting shortest-path tree from u:

resulting forwarding table in u:

<table>
<thead>
<tr>
<th>destination</th>
<th>link</th>
</tr>
</thead>
<tbody>
<tr>
<td>v</td>
<td>(u,v)</td>
</tr>
<tr>
<td>x</td>
<td>(u,x)</td>
</tr>
<tr>
<td>y</td>
<td>(u,x)</td>
</tr>
<tr>
<td>w</td>
<td>(u,x)</td>
</tr>
<tr>
<td>z</td>
<td>(u,x)</td>
</tr>
</tbody>
</table>

Again, if you don't remember Dijkstra's algorithm, take a look at the pseudocode on page 367 and this video from University of Washington:

https://www.youtube.com/watch?v=elUoB_UwIlpQ

Convince yourself you understand it and can perform it for small networks!
Distance Vector algorithm

- Link State Algorithms required us to exchange information about the entire network topology, then determine best path. This isn't possible for a large ISP's network, much less entire Internet.
- Distance Vector Algorithms instead allow us to obtain best paths without knowing about the topology of the broader network.
- Still keep a table of cost to each node in N, but we never really know about the structure to get there...
- In real world we will need to reduce N...
Distance vector algorithm

Bellman-Ford equation (dynamic programming)

let

\[ d_x(y) := \text{cost of least-cost path from } x \text{ to } y \]

then

\[ d_x(y) = \min_v \{ c(x,v) + d_v(y) \} \]

- cost from neighbor \( v \) to destination \( y \)
- cost to neighbor \( v \)
- \( \min \) taken over all neighbors \( v \) of \( x \)
Bellman-Ford example

For calculating cost from $u$ to $z$, we can use path through $w$, $v$, or $x$

clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation says:

$$d_u(z) = \min \{ c(u,v) + d_v(z),
             c(u,x) + d_x(z),
             c(u,w) + d_w(z) \}$$

$$= \min \{2 + 5, 1 + 3, 5 + 3\} = 4$$

- Node achieving minimum is next hop in shortest path, so used in forwarding table
- For the moment assume each node knows their own costs
Distance vector algorithm

• \( D_x(y) \) = estimate of least cost from \( x \) to \( y \)
  – \( x \) maintains distance vector \( D_x = [D_x(y): y \in N] \)

• node \( x \):
  – knows cost to each neighbor \( v \): \( c(x,v) \)
  – maintains its neighbors’ distance vectors. For each neighbor \( v \), \( x \) maintains \( D_v = [D_v(y): y \in N] \)
key idea:

- from time-to-time, each node sends its own distance vector estimate to neighbors
- when x receives new DV estimate from neighbor, it updates its own DV using B-F equation:

\[ D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \text{ for each node } y \in N \]

- The estimate \( D_x(y) \) converge to the actual least cost \( d_x(y) \) (under a few reasonable conditions we won't discuss here)
Distance vector algorithm

iterative, asynchronous:
  each local iteration caused by:
  • local link cost change
  • DV update message from neighbor

distributed:
  • each node notifies neighbors only when its DV changes
    – neighbors then notify their neighbors if necessary

each node:

  wait for (change in local link cost or msg from neighbor)

  compute estimates

  if DV to any dest has changed, notify neighbors
Example of DV Algorithm

• Should look at worked example from book.
• Very simple one in your home work.
• Can see a worked example in a video here: https://www.youtube.com/watch?v=ylzAefKE NXY
Hierarchical routing

our routing study thus far - idealization

• all routers identical
• network “flat”

… not true in practice

scale: with millions of destinations:

• can't store all dest's in routing tables!
• routing table exchange would swamp links!

administrative autonomy

• internet = network of networks
• each network admin may want to control routing in its own network
Hierarchical routing

- aggregate routers into regions, “autonomous systems” (AS)
- routers in same AS run same routing protocol
  - “intra-AS” routing protocol
  - routers in different AS can run different intra-AS routing protocol
  - Administrators can define own mechanism internally
  - (if you are directly connected on the same switch don't even need to use the routers)

**gateway router:**
- at “edge” of its own AS
- has link to router in another AS
Reminder: Connected by switch...

Q: how are interfaces actually connected?
A: we’ll learn about that in chapter 5, 6.

A: wired Ethernet interfaces connected by Ethernet switches

For now: don’t need to worry about how one interface is connected to another (with no intervening router)

Router is then connected to other routers

If in same organization (controlled by same group) then same AS...
Interconnected ASes

- Forwarding table configured by both intra- and inter-AS routing algorithm
  - *intra*-AS sets entries for internal dests
  - *inter*-AS & *intra*-AS sets entries for external dests
Inter-AS tasks

• suppose a router in AS1 receives datagram destined outside of AS1:
  ▪ router should forward packet to gateway router, but which one?

AS1 must:
1. learn which destinations are reachable through AS2, which through AS3
2. propagate this reachability info to all routers in AS1

job of inter-AS routing!
Example: setting forwarding table in router 1d

- suppose AS1 learns (via inter-AS protocol) that subnet $x$ reachable via AS3 (gateway 1c), but not via AS2
  - **inter-AS** protocol propagates reachability info to all internal routers
- router 1d determines from intra-AS routing info that its interface $I$ is on the least cost path to 1c
  - installs forwarding table entry $(x, I)$ (used intra- and inter-AS routing to make final decision)
Example: choosing among multiple ASes

- now suppose AS1 learns from inter-AS protocol that subnet $x$ is reachable from AS3 and from AS2.
- to configure forwarding table, router 1d must determine which gateway it should forward packets towards for dest $x$
  - this is also job of inter-AS routing protocol!
Example: choosing among multiple ASes

• now suppose AS1 learns from **inter**-AS protocol that subnet \( x \) is reachable from AS3 *and* from AS2.
• to configure forwarding table, router 1d must determine towards which gateway it should forward packets for dest \( x \)
  • this is also job of **inter**-AS routing protocol!
• *hot potato routing:* send packet towards lowest cost of two routers.

| learn from inter-AS protocol that subnet \( x \) is reachable via multiple gateways | use routing info from intra-AS protocol to determine costs of least-cost paths to each of the gateways | hot potato routing: choose the gateway that has the smallest least cost | determine from forwarding table the interface \( I \) that leads to least-cost gateway. Enter \((x,I)\) in forwarding table |
Intra-AS tasks

- suppose router in AS1 receives datagram destined *inside* of AS1:

  **AS1 must:**
  1. decide which router (1a, 1b, 1c, 1d) will get the datagram to the final destination host.

  *job of intra-AS routing!*
Protocols for Routing

• Intra-AS routing
• also known as *interior gateway protocols (IGP)*
  – We'll look at RIP and OSPF

• Inter-AS routing
• inter-domain routing protocols
  – We'll look at BGP
Intra-AS Routing Protocols

• also known as *interior gateway protocols (IGP)*

• most common intra-AS routing protocols:
  – RIP: Routing Information Protocol (uses DV)
  – OSPF: Open Shortest Path First (uses LS)
  – Some others:
    • IGRP: Interior Gateway Routing Protocol (Cisco proprietary)
    • IS-IS: ISO protocol very similar to OSPF
RIP (Routing Information Protocol)

- included in BSD-UNIX distribution in 1982
- distance vector algorithm
  - distance metric: # hops (max = 15 hops), each link has cost 1
  - DVs exchanged with neighbors every 30 sec in response message (aka advertisement)
  - each advertisement: list of up to 25 destination subnets (in IP addressing sense)
  - Clearly only works up to certain sized networks – this is Intra-AS!

from router A to destination subnets:

<table>
<thead>
<tr>
<th>subnet</th>
<th>hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>1</td>
</tr>
<tr>
<td>v</td>
<td>2</td>
</tr>
<tr>
<td>w</td>
<td>2</td>
</tr>
<tr>
<td>x</td>
<td>3</td>
</tr>
<tr>
<td>y</td>
<td>3</td>
</tr>
<tr>
<td>z</td>
<td>2</td>
</tr>
</tbody>
</table>
RIP: example

routing table in router D

<table>
<thead>
<tr>
<th>destination subnet</th>
<th>next router</th>
<th># hops to dest</th>
</tr>
</thead>
<tbody>
<tr>
<td>w</td>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td>y</td>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>z</td>
<td>B</td>
<td>7</td>
</tr>
<tr>
<td>x</td>
<td>--</td>
<td>1</td>
</tr>
<tr>
<td>....</td>
<td>....</td>
<td>....</td>
</tr>
</tbody>
</table>
RIP: example

Routing table in router D

<table>
<thead>
<tr>
<th>destination subnet</th>
<th>next router</th>
<th># hops to dest</th>
</tr>
</thead>
<tbody>
<tr>
<td>w</td>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td>y</td>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>z</td>
<td>A</td>
<td>7</td>
</tr>
<tr>
<td>x</td>
<td>--</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

A-to-D advertisement

```
<table>
<thead>
<tr>
<th>dest</th>
<th>next hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>x</td>
<td>1</td>
</tr>
<tr>
<td>z</td>
<td>4</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
```
OSPF (Open Shortest Path First)

• "open": publicly available (IETF standard)
• uses link state algorithm
  – LS packet dissemination
  – topology map at each node
  – route computation using Dijkstra’s algorithm
• OSPF advertisement carries one entry per neighbor
• advertisements flooded to entire AS
  – carried in OSPF messages directly over IP (rather than TCP or UDP)
Hierarchical OSPF

• For very large networks, can use OSPF hierarchically
  • two-level hierarchy: local area, backbone.
    – link-state advertisements only in area
    – each node has detailed area topology; only know direction (shortest path) to nets in other areas.
  • area border routers: “summarize” distances to nets in own area, advertise to other Area Border routers.
  • backbone routers: run OSPF routing limited to backbone.
  • boundary routers: connect to other AS’ s.
Hierarchical OSPF

- backbone router
- boundary router
- area border routers
- internal routers

areas:
- area 1
- area 2
- area 3
Internet inter-AS routing: BGP

• **BGP (Border Gateway Protocol):** the de facto inter-domain routing protocol
  – “glue that holds the Internet together”

• **BGP provides each AS a means to:**
  – **eBGP:** obtain subnet reachability information from neighboring ASs.
  – **iBGP:** propagate reachability information to all AS-internal routers.
  – determine “good” routes to other networks based on reachability information and policy.

• allows subnet to advertise its existence to rest of Internet: “**I am here**”
BGP basics

- **BGP session**: two BGP routers (“peers”) exchange BGP messages:
  - advertising *paths* to different destination network prefixes (“path vector” protocol)
  - exchanged over semi-permanent TCP connections

- when AS3 advertises a prefix to AS1:
  - AS3 *promises* it will forward datagrams towards that prefix
  - AS3 can aggregate prefixes in its advertisement
BGP basics: distributing path information

- using eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
  - 1c can then use iBGP do distribute new prefix info to all routers in AS1
  - 1b can then re-advertise new reachability info to AS2 over 1b-to-2a eBGP session
- when router learns of new prefix, it creates entry for prefix in its forwarding table.
Path attributes and BGP routes

• advertised prefix includes BGP attributes
  – prefix + attributes = “route”

• two important attributes:
  – **AS-PATH**: contains ASs through which prefix advertisement has passed: e.g., AS 67, AS 17
  – **NEXT-HOP**: indicates specific internal-AS router to next-hop AS. (may be multiple links from current AS to next-hop-AS)

• gateway router receiving route advertisement uses **import policy** to accept/decline
  – e.g., never route through AS x
  – **policy-based** routing
BGP route selection

- router may learn about more than 1 route to destination AS, selects route based on:
  1. local preference value attribute: policy decision (cost)
  2. shortest AS-PATH
  3. closest NEXT-HOP router: hot potato routing
  4. additional criteria
BGP messages

- BGP messages exchanged between peers over TCP connection
- BGP messages:
  - **OPEN**: opens TCP connection to peer and authenticates sender
  - **UPDATE**: advertises new path (or withdraws old)
  - **KEEPALIVE**: keeps connection alive in absence of UPDATES; also ACKs OPEN request
  - **NOTIFICATION**: reports errors in previous msg; also used to close connection
BGP routing policy

- A, B, C are provider networks
- X, W, Y are customer (of provider networks)
- X is dual-homed: attached to two networks
  - X does not want to route from B via X to C
  - .. so X will not advertise to B a route to C
BGP routing policy (2)

- No way! B gets no “revenue” for routing CBAW since neither W nor C are B’s customers
- B wants to force C to route to w via A
- B wants to route *only* to/from its customers!
Why different Intra-, Inter-AS routing?

**policy:**
- inter-AS: admin wants control over how its traffic routed, who routes through its net.
- intra-AS: single admin, so no policy decisions needed

**scale:**
- hierarchical routing saves table size, reduced update traffic

**performance:**
- intra-AS: can focus on performance
- inter-AS: policy may dominate over performance