Distance Vector Routing: overview

Iterative, asynchronous:
- each local iteration caused by:
  - local link cost change
  - message from neighbor: its least cost path change from neighbor

Distributed:
- each node notifies neighbors only when its least cost path to any destination changes
  - neighbors then notify their neighbors if necessary

Each node:
1. wait for (change in local link cost of msg from neighbor)
2. recompute distance table
3. if least cost path to any dest has changed, notify neighbors
Distance Vector Algorithm:

At all nodes, X:

1. Initialization:
2. for all adjacent nodes v:
3. \( D^{X(*)}(v) = \infty \) /* the * operator means "for all rows" */
4. \( D^{X}(v,v) = c(X,v) \)
5. for all destinations, y
6. send \( \min_w D^{X}(y,w) \) to each neighbor /* w over all X's neighbors */
Distance Vector Algorithm (cont.):

8    loop
9      wait (until I see a link cost change to neighbor V
10     or until I receive update from neighbor V)
11
12    if (c(X,V) changes by d)
13       /* change cost to all dest's via neighbor v by d */
14       /* note: d could be positive or negative */
15       for all destinations y: D^X(y,V) = D^X(y,V) + d
16
17    else if (update received from V wrt destination Y)
18       /* shortest path from V to some Y has changed */
19       /* V has sent a new value for its min_w DV(Y,w) */
20       /* call this received new value is "newval" */
21       for the single destination y: D^X(Y,V) = c(X,V) + newval
22
23    if we have a new min_w D^X(Y,w) for any destination Y
24       send new value of min_w D^X(Y,w) to all neighbors
25
26    forever
Distance Vector Algorithm: example
Distance Vector Algorithm: example

\[ D^X(Y,Z) = c(X,Z) + \min_w \{ D^Z(Y,w) \} = 7 + 1 = 8 \]

\[ D^X(Z,Y) = c(X,Y) + \min_w \{ D^Y(Z,w) \} = 2 + 1 = 3 \]
Distance Vector: link cost changes

Link cost changes:
- node detects local link cost change
- updates distance table (line 15)
- if cost change in least cost path, notify neighbors (lines 23,24)

"good news travels fast"
Distance Vector: link cost changes

Link cost changes:
- good news travels fast
- bad news travels slow - “count to infinity” problem!

algorithm continues on!
Distance Vector: poisoned reverse

If Z routes through Y to get to X:

- Z tells Y its (Z's) distance to X is infinite (so Y won't route to X via Z)
- will this completely solve count to infinity problem?

![Diagram of distance vector algorithm]

algorithm terminates
Comparison of LS and DV algorithms

Message complexity
- **LS**: with n nodes, E links, O(nE) msgs sent each
- **DV**: exchange between neighbors only
  - convergence time varies

Speed of Convergence
- **LS**: $O(n^2)$ algorithm requires $O(nE)$ msgs
  - may have oscillations
- **DV**: convergence time varies
  - may be routing loops
  - count-to-infinity problem

Robustness: what happens if router malfunctions?
- **LS**:
  - node can advertise incorrect *link* cost
  - each node computes only its own table
- **DV**:
  - DV node can advertise incorrect *path* cost
  - each node’s table used by others
    - error propagate thru network
Hierarchical Routing

Our routing study thus far - idealization
- all routers identical
- network “flat”
... not true in practice

scale: with 50 million 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
  - “inter-AS” routing protocol
  - routers in different AS can run different inter-AS routing protocol

- gateway routers
  - special routers in AS
  - run inter-AS routing protocol with all other routers in AS
  - also responsible for routing to destinations outside AS
  - run *intra-AS routing* protocol with other gateway routers
Intra-AS and Inter-AS routing

Gateways:
- perform inter-AS routing amongst themselves
- perform intra-AS routers with other routers in their AS

inter-AS, intra-AS routing in gateway A.c

network layer
link layer
physical layer
Intra-AS and Inter-AS routing

Intra-AS routing within AS A

Inter-AS routing between A and B

Intra-AS routing within AS B

Host h1

Host h2
The Internet Network layer

Host, router network layer functions:

- Routing protocols
  - path selection
  - RIP, OSPF, BGP
- IP protocol
  - addressing conventions
  - datagram format
  - packet handling conventions
- ICMP protocol
  - error reporting
  - router “signaling”
IP Addressing

- **IP address**: 32-bit identifier for host, router *interface*
- **interface**: connection between host, router and physical link
  - router’s typically have multiple interfaces
  - host may have multiple interfaces
  - IP addresses associated with interface, not host, router

```
223.1.1.1 = 11011111 00000001 00000001 00000001
```

```
223.1.1.1
223.1.1.2
223.1.1.3
223.1.1.4
```

```
223.1.2.1
223.1.2.2
223.1.2.9
```

```
223.1.3.1
223.1.3.2
223.1.3.27
```
IP Addressing

- IP address:
  - network part (high order bits)
  - host part (low order bits)

- What's a network? (from IP address perspective)
  - device interfaces with same network part of IP address
  - can physically reach each other without intervening router

Network consisting of 3 IP networks (for IP addresses starting with 223, first 24 bits are network address)
IP Addressing

How to find the networks?

- Detach each interface from router, host
- create “islands of isolated networks

Interconnected system consisting of six networks
# IP Addresses

<table>
<thead>
<tr>
<th>Class</th>
<th>Network</th>
<th>Host</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>host</td>
<td>1.0.0.0 to 127.255.255.255</td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td>network</td>
<td>128.0.0.0 to 191.255.255.255</td>
</tr>
<tr>
<td>C</td>
<td>110</td>
<td>network</td>
<td>192.0.0.0 to 239.255.255.255</td>
</tr>
<tr>
<td>D</td>
<td>1110</td>
<td>multicast address</td>
<td>240.0.0.0 to 247.255.255.255</td>
</tr>
</tbody>
</table>

32 bits
Internet Addresses

- Also called IP addresses
  - Example: 128.171.17.13
  - Really 32-bit strings of ones and zeros
- Fit into source and destination address field of IP headers

IP Packet

32-bit Source and Destination Addresses
Internet Address

- Hierarchical Addressing
  - Two-Parts
    - Network part (organization on the Internet)
    - Local part (host on the network)
  - Three-Parts
    - Network (organization on the Internet)
    - Subnet (suborganization)
    - Host on the subnet
Internet Addresses

- Two-Part
  - Divide Internet address into two parts
  - First part designates the network
  - Second (local) part designates the host on the network
  - Example:

<table>
<thead>
<tr>
<th>Network Part</th>
<th>Local Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.171.17.13</td>
<td></td>
</tr>
</tbody>
</table>
Internet Addresses

- Three-Part
  - Local part is subdivided
  - Subnet part designates the subnet (suborganization)
  - Host part designates the host
  - Example:

<table>
<thead>
<tr>
<th>Network Part</th>
<th>Subnet Part</th>
<th>Host Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.171.17</td>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>
2-Part Internet Address and Routers

- Routers attach to multiple networks
  - Has an internet address on each network
    - 128.171.193.15 on network 128.171
    - 183.287.7.7 on network 183.287

```
Network 128.171
Router A 128.171.93.15
Host 1 128.171.15.12
```
```
Router A 183.287.7.7
Router B 183.287.9.12
Host B 183.287.180.6
```
2-Part Internet Address and Routers

- Host sends IP packet to router
  - Router looks at destination address network part ONLY (183.287)
  - Compares to network parts of its own addresses (128.171, 183.287)
2-Part Internet Address and Routers

- If a network part matches (187.287)
  - The destination host is on that network
  - The router delivers it to the destination host
2-Part Internet Address and Routers

- If no network part matches
  - Destination host is not on one of the router’s networks
  - Passes the IP packet onto another router
3-Part Internet Address and Routers

- Routers attach to multiple *subnets* (not networks)
  - Has an internet address on each subnet
  - Network PLUS subnet part underlined

```
Network/Subnet  128.171.17  
Host 1          128.171.17.12

Router A        128.171.17.15

Router B        128.171.15.12
Host B          128.171.15.6
```
3-Part Internet Address and Routers

Host sends IP packet to router

- Router looks at destination address network plus parts ONLY (128.171.15)
- Compares to network plus subnet parts of its own addresses (128.171.17, 128.171.15)
3-Part Internet Address and Routers

- If a network plus subnet part matches (128.171.15),
  - The destination host is on that subnet
  - The router delivers it
3-Part Internet Address and Routers

- If no network plus subnet part matches
  - Destination host is not on one of the router's subnets
  - Passes the IP packet onto another router
Assigning Two-Part Internet Addresses

- Organization applies to an internet registrar
  - It is given a two-part internet address
  - It assigns the local part to hosts internally
  - Only large organizations and ISPs get two-part addresses

Registrar \(128.171\) Firm

\(128.171.17.13\)

\(128.171.123.130\)
Assigning 3-Part Internet Addresses

- Organization has 2-part internet address
  - Assigns subnet part to suborganization
  - Suborganization assigns host bits to hosts

128.171  128.171.17.13

Registrar  ➔  Firm  ➔  Suborganization

128.171.17.13

Host
ICANN

- Internet Corporation for Assigned Numbers and Names
  - Is being given overall control over name and number assignments
  - Will work through multiple registrars.
Subnet addressing

All traffic to 128.10.0.0

Network 128.10.1.0

Network 128.10.2.0

128.10.1.2

128.10.2.7

128.10.2.8

Rest of the Internet
Subnet masks

- Choosing a subnet addressing scheme is synonymous with choosing how to partition the local portion of an IP address into physical subnet and host part.
- The standard specifies that a site using subnet addressing must choose a 32-bit subnet mask for each network.
- Bits in the subnet mask are set to 1 if the network treats the corresponding bit in the IP address as part of the network address, 0 if it treats the bit as part of the host identifier.
- \[ \begin{align*}
  11111111 & 11111111 & 11111111 & 00000000 & \rightarrow 255.255.255.0
\end{align*} \]
Subnet Masks

- By very definition of the network and host portions, each class network has a natural mask.
  - Class A natural mask 255.0.0.
  - Class B natural mask 255.255.0.0
  - Class C natural mask 255.255.255.0

- Without subnetting, network numbers would be of very limited use. The subnetting technique increases the number of subnetworks and reduces the number of hosts.

- A mask of 255.255 0.0 is applied to a network 10.0.0.0.
  - This increases from a single network 10.0.0.0 to 256 subnetworks ranging from 10.0.0.0 to 10.255.0.0.
  - This however decreases the number of hosts per each subnet from 16777216 to 65536 (ignoring boundaries).
IP Addressing

class

A  0  network  host  1.0.0.0 to 127.255.255.255

B  10  network  host  128.0.0.0 to 191.255.255.255

C  110  network  host  192.0.0.0 to 239.255.255.255

D  1110  multicast address  240.0.0.0 to 247.255.255.255

32 bits
Subnetting

- Detailed in RFC 950

<table>
<thead>
<tr>
<th>Class B</th>
<th>16 bits</th>
<th>8 bits</th>
<th>8 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>network</td>
<td></td>
<td>Subnet</td>
<td>host</td>
</tr>
</tbody>
</table>

- Subnetting splits the host address into two parts
- Size of each part up to the administrator.
- Only 8 bits to subnet in a class C address!
Subnetting

- Network ID. Subnet. Host
- Reduces the size of routing tables.
- One external class B routing table entry instead of 256 class C addresses.
- Changes to subnets does not require external announcements.
Subnetting inside a domain

- Internal routers must be aware of the subnet sizes.
- Subnet Masks identify the size of subnets.

<table>
<thead>
<tr>
<th>Class B</th>
<th>Network</th>
<th>Subnet</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1111 1111</td>
<td>1111 1111</td>
<td>0000 0000</td>
</tr>
</tbody>
</table>

- 32-bit value containing 1s over the network and subnet IDs, and 0s over the host IDs.
- & with any host to determine subnet number.
Subnet Masks

Using the high order bits of your IP address, you can determine your class (A, B, or C).
**Netstat -nr (unix command)**

> netstat -nr

Routing Table:

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Flags</th>
<th>Ref</th>
<th>Use</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.114.48.0</td>
<td>128.114.49.15</td>
<td>U</td>
<td>3</td>
<td>1218</td>
<td>hme0</td>
</tr>
<tr>
<td>224.0.0.0</td>
<td>128.114.49.15</td>
<td>U</td>
<td>3</td>
<td>0</td>
<td>hme0</td>
</tr>
<tr>
<td>default</td>
<td>128.114.48.1</td>
<td>UG</td>
<td>0</td>
<td>7977</td>
<td></td>
</tr>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>UH</td>
<td>0</td>
<td>4182</td>
<td>lo0</td>
</tr>
</tbody>
</table>
Subnet Arithmetic

- Host address 128.114.49.15
- subnet mask: 255.255.248.0

\[
\begin{align*}
10000000.1110010.00110001&11111110.11111111.00000000
\end{align*}
\]

(Class B). 00110 000.00000000

this host is on subnet 6

or sometimes written as 128.114.48.0
Host routing with Subnets

Hosts search routing tables for

1. Matching host address (same LAN)
2. Matching subnet address
3. Matching network address
4. Default route
Address Resolution Protocol (ARP)

- Interface between Link layer and Network Layer.
- Allows hosts to query who owns an IP address on the same LAN.
- Owner responds with hardware address.

- Allows changes to link layer to be independent of IP addressing.

(Example to come)