CS557: Hierarchical Routing

Christos Papadopoulos
Hierarchies

• What?
  – Logical structure overlaid on collections of nodes
  – Example: countries, states, cities, neighborhoods, etc.

• Why?
  – Together with information abstraction, the only known solution to scaling issues
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
Areas

• Technique for hierarchically addressing nodes in a network
• Divide network into areas
  – Areas can overlap
  – Areas can have nested sub-areas
  – Constraint:
    • there must exist at least one path between each pair of sub-areas in an area that does not exit the area
    • other areas can have one entry for entire area
Addressing

• Address areas hierarchically
  – sequentially number top-level areas
  – sub-areas of area are labeled relative to the parent area
  – nodes are numbered relative to the smallest containing area
    • nodes can have multiple addresses (when?)
The Area Hierarchy
Routing

• Within a sub-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 border router

• Can result in sub-optimal paths
Path Sub-optimality

1.2.2 \rightarrow 3.2.1
3 hop red path
v.s.
2 hop green path
Landmark Hierarchy

• Idea: know lots of detail about things nearby and less about things far away
• Not defined by arbitrary boundaries
  – thus, not well suited to the real world that does have administrative boundaries
• Self-configuring hierarchy (unlike areas)
A Landmark

Router 1 is a landmark of radius 2
Landmark Overview

- Landmark routers have “height” which determines how far away they can be seen (visibility)
- Routers within radius \( n \) can see a landmark router \( LM(n) \)
- *See* means that those routers have \( LM(n) \)’s address and know next hop to reach it
  - Router \( x \) as an entry for router \( y \) if \( x \) is within radius of \( y \)
- Distance vector style routing with simple metric
- Routing table: Landmark (\( LM2(d) \)), Level(2), Next hop
LM Hierarchy Definition

• Each LM (Li) associated with level (i) and radius (ri)
• Every node is an L0 landmark
• Recursion: some Li are also Li+1
  – Every Li is seen by at least one Li+1
• Terminating state when all level j LMs see entire network
Landmark Levels

Router 1 is a landmark of radius 2 and belongs to level 0
Radius of $LM(0) = 2$
LM Addresses

- LM(2).LM(1).LM(0) (x.a.b and y.a.b)
- LM level maps to radius (part of configuration), e.g.:
  - LM level 0: radius 2
  - LM level 1: radius 4
  - LM level 2: radius 8
- If destination is more than two hops away, node will not have complete routing information
  - ..thus refer to LM(1) portion of address
  - ..if not, then refer to LM(2).. etc.
  (c forwards to y in y.a.b)
LM Routing

- LM does not imply hierarchical forwarding
- It is NOT a source route
- En route to LM(1) may encounter router that is within LM(0) radius of destination address (like longest match)
- Paths may be asymmetric
LM Self-configuration

• Bottom-up hierarchy construction algorithm
  – goal to bound number of children
• Every router is L0 landmark
• All routers advertise themselves at distance r0
• For each level i:
  – All Li landmarks run election to self-promote one or more Li to Li+1
• Dynamic algorithm to adapt to topology changes---Efficient hierarchy
Landmark Routing: Basic Idea

- Not shortest path
- Packet does not necessarily follow specified landmarks

Source wants to reach $LM0[a]$, whose address is $c.b.a$:
- Source can see $LM2[c]$, so sends packet towards $c$
- Entering $LM1[b]$ area, first router diverts packet to $b$
- Entering $LM0[a]$ area, packet delivered to $a$
Landmark Routing: Example
Routing Table for Router “g”

<table>
<thead>
<tr>
<th>Landmark</th>
<th>Level</th>
<th>Next hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM2[d]</td>
<td>2</td>
<td>f</td>
</tr>
<tr>
<td>LM1[i]</td>
<td>1</td>
<td>k</td>
</tr>
<tr>
<td>LM0[e]</td>
<td>0</td>
<td>f</td>
</tr>
<tr>
<td>LM0[k]</td>
<td>0</td>
<td>k</td>
</tr>
<tr>
<td>LM0[f]</td>
<td>0</td>
<td>f</td>
</tr>
</tbody>
</table>

r0 = 2, r1 = 4, r2 = 8 hops

How to go from d.i.g to d.n.t?
How does path length compare to shortest path?
Recap

• Strongest point: self configuration
• No administrative bounds, thus not suitable for Internet
• No policy routing
• Variable addresses
• Not really used at this point but the ideas keep coming up..