Wireless Meshing with the One Laptop Per Child

cozybit Inc.

Javier Cardona

April 13th, 2007
1. Mesh Routing
2. The OLPC Mesh
3. Internetworking
4. Application Development
5. Debugging
6. Mesh In Action
Outline

1. Mesh Routing
   Definition
   Advantages/Disadvantages

2. The OLPC Mesh

3. Internetworking

4. Application Development

5. Debugging

6. Mesh In Action
What is Mesh Routing?

- *Mesh routing*: routing protocols that work in partially connected mesh topologies.
- Not to be confused with 802.11 Ad-Hoc networks, which require a full mesh topology.
What is Mesh Routing?

- *Mesh routing*: routing protocols that work in partially connected mesh topologies.
- Not to be confused with 802.11 Ad-Hoc networks, which require a full mesh topology.
Advantages

- Power efficiency (some geometries better than others)
Advantages

- Power efficiency (some geometries better than others)
Advantages

- Power efficiency (some geometries better than others)
Advantages

- Power efficiency (some geometries better than others)
Advantages

- Power efficiency (some geometries better than others)
Advantages

• Power efficiency (some geometries better than others)
Advantages

- Self-configured
- Self-healing
- Extended range
Disadvantages

- Higher delays.
- Less secure.
- Additional complexity in nodes, as each node becomes a router.
1. Mesh Routing

2. The OLPC Mesh
   - 802.11s
   - Path Selection
   - Limited Broadcast and Multicast Propagation

3. Internetworking

4. Application Development

5. Debugging

6. Mesh In Action
Path Selection

- 802.11s still in draft stage. Draft 1.02 published last week.
- OLPC mesh based on Draft 0.01
- Only a subset of it has been implemented: path selection, broadcast and forwarding.
- Hybrid Wireless Mesh Protocol is the mandatory path selection protocol of 802.11s. On demand, table based routing.

<table>
<thead>
<tr>
<th>next hop</th>
<th>destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{MAC}_B$</td>
<td>$\text{MAC}_A$</td>
</tr>
<tr>
<td>$\text{MAC}_E$</td>
<td>$\text{MAC}_E$</td>
</tr>
<tr>
<td>$\text{MAC}_A$</td>
<td>$\text{MAC}_C$</td>
</tr>
</tbody>
</table>
Path Requests

- On demand routing: Paths are discovered when needed.
- PREQs are sent as broadcast and propagated into the mesh.
- Intermediate nodes forward only the best PREQs.
- Intermediate nodes maintain routing tables.
Path Requests

- On demand routing: Paths are discovered when needed.
- PREQs are sent as broadcast and propagated into the mesh.
- Intermediate nodes forward only the best PREQs.
- Intermediate nodes maintain routing tables.
Path Replies

- PREPs come back as unicast through the route with the best metric.
- Currently the cost of a link is proportional to the transmission power needed to successfully send a frame.
- Soon the metric will also take into account the state of the battery of the intermediate nodes.
Path Replies

- PREPs come back as unicast through the route with the lowest forward cost.
- Currently the cost of a link is proportional to the transmission power needed to successfully send a frame.
- Soon the metric will also take into account the state of the battery of the intermediate nodes.
Limited Broadcast and Multicast Propagation

- Intermediate nodes will re-transmit broadcast/multicast frames only once.
- Each mesh frame is identified by end-to-end sequence number generated at the originating node.
- Nodes maintain a Recently Broadcast Table indexed by \(< sa, e2eseq >\)
Outline

1. Mesh Routing
2. The OLPC Mesh
3. Internetworking
   - Mesh Portals (MPP)
   - L3 Routing
   - MPP Discovery
   - Gateway Configuration
4. Application Development
5. Debugging
6. Mesh In Action
Mesh Portals (MPP)

- Mesh Points that associate with an Access Point become Mesh Portals (MPPs)
- MPPs route traffic in and out of the mesh.
- L2 bridging not possible because of the 3-address problem.
The 3-Address Problem

Addr1: A? B?
Addr2: X
Addr3: AP (BSSID)
L3 Routing

- Mesh network is an IP subnet.
- Check destination IP to determine if traffic is out-mesh.
- MPs must find a Mesh Portal (MPP) to communicate outside the mesh.
MPP Discovery

- MP will send an MPP request to **ANY\_MPP\_MAC\_ADDRESS**, an *anycast address* claimed by all the MPPs in the mesh.
- Path discovery mechanism will result in best MPP chosen.
MP Configuration

- MPPs listen for configuration requests from MPs.
- MPPs will respond with gateway/DNS information.
Outline

1. Mesh Routing
2. The OLPC Mesh
3. Internetworking
4. Application Development
   - Mesh Virtual Interface
   - TTL
   - Round Trip Time
   - Throughput
5. Debugging
6. Mesh In Action
Network Interfaces

- Two network interfaces, one PHY.
- Simultaneous operation in mesh and infra/ad-hoc modes
- eth0 for infra/ad-hoc traffic
- msh0 for mesh traffic
Network Interfaces

eth0  
Link encap:Ethernet  HWaddr 00:17:C4:02:2F:07  
inet addr:98.85.46.99  Bcast:98.85.46.255  Mask:255.255.255.0  
inet6 addr: 2001:4830:2446:ff00:217:c4ff:fe02:2f07/64 Scope:Global  
inet6 addr: fe80::217:c4ff:fe02:2f07/64 Scope:Link  
UP  BROADCAST  RUNNING  MTU:1500  Metric:1  
RX packets:231534 errors:0 dropped:0 overruns:0 frame:0  
TX packets:14225 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:36946391 (35.2 MiB)  TX bytes:5025436 (4.7 MiB)

msh0  
Link encap:Ethernet  HWaddr 00:17:C4:02:2F:07  
inet addr:10.2.47.7  Bcast:10.255.255.255  Mask:255.0.0.0  
inet6 addr: fe80::217:c4ff:fe02:2f07/64 Scope:Link  
UP  BROADCAST  RUNNING  MULTICAST  MTU:1500  Metric:1  
RX packets:231534 errors:0 dropped:0 overruns:0 frame:0  
TX packets:14225 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:36946391 (35.2 MiB)  TX bytes:5025436 (4.7 MiB)
Your Audience May Vary
Your Audience May Vary

Mesh TTL = 1
Your Audience May Vary

Mesh TTL = 2
Your Audience May Vary

Mesh TTL = 3
Your RTT May Vary
Your RTT May Vary
Your RTT May Vary

RTT over a 22 node Mesh

RTT Mean Deviation (ms) vs. Average RTT (ms)
Your Thoughput May Vary
Your Throughput May Vary

Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)

[ 4] local 10.2.47.7 port 5001 connected with 10.0.202.163 port 43916
[ 4] 0.0-13.5 sec 4.42 MBytes 2.75 Mbits/sec
[ 4] local 10.2.47.7 port 5001 connected with 10.2.46.254 port 38633
[ 4] 0.0-12.7 sec 1.46 MBytes 966 Kbits/sec
[ 4] local 10.2.47.7 port 5001 connected with 10.2.47.175 port 42537
[ 4] 0.0-12.9 sec 944 KBytes 598 Kbits/sec
Outline

1. Mesh Routing
2. The OLPC Mesh
3. Internetworking
4. Application Development
5. Debugging
   - Inspecting the Forwarding Table
   - Path Lookup
   - Forcing Topologies
   - Statistics
   - Sniffing Traffic
6. Mesh In Action
Inspecting the Forwarding Table

$ iwpriv msh0 fwt_list <entry-id>

00:17:c4:00:cc:92 00:17:c4:00:ce:fc 1 26 1 12 4871 5207 2 0 \\
78213060 0 61 00:00:00:00:00:00:00

- **DA** Destination Address
- **RA** Receiver Address
Inspecting the Forwarding Table

$ iwpriv msh0 fwt_list <entry-id>

00:17:c4:00:cc:92 00:17:c4:00:ce:fc 1 26 1 12 4871 5207 2 0 \ 78213060 0 61 00:00:00:00:00:00:00:00

- metric
Inspecting the Forwarding Table

$ iwpriv msh0 fwt_list <entry-id>

00:17:c4:00:cc:92 00:17:c4:00:ce:fc 1 26 1 12 4871 5207 2 0 \ 78213060 0 61 00:00:00:00:00:00:00

- direct or reverse
Path Lookup

$ iwpriv msh0 fwt_lookup <mac-add>

$ iwpriv msh0 fwt_lookup 00:17:c4:02:2e:fe
   00:17:c4:02:2e:fe 00:17:c4:00:ca:a3 1 90 1 0 37753 74010 3 0 \  
227997200 0 71 00:00:00:00:00:00
Forcing Topologies

The Blinding Table (BT) is a useful tool for debugging:

- Simulate nodes being out of range.
- Purposely break routes.
- Force specific routes to be created.
- Testing application performance over multiple hops.
Forcing Topologies

Add a MAC address to the BT:

$ iwpriv msh0 bt_add 00:17:c4:00:cc:92

Inspect the BT:

$ iwpriv msh0 bt_list <entry-id>

Reset (clear) the BT:

$ iwpriv msh0 bt_reset
Forcing Topologies

Two modes of operation:

- **Whitelist**
  
  `$ iwpriv msh0 bt_set_invert 1`

- **Blacklist (default)**

  `$ iwpriv msh0 bt_set_invert 0`
Use ethtool to see some useful statistics about packets transmitted and dropped:

```bash
$ ethtool -S msh0
NIC statistics:
  drop_duplicate_bcast: 81006
  drop_ttl_zero: 10366
  drop_no_fwd_route: 0
  drop_no_buffers: 6
  fwded_unicast_cnt: 1127
  fwded_bcast_cnt: 20599
  drop_blind_table: 986595
  tx_failed_cnt: 0
```
Sniffing Traffic

Sometimes you need to bring out the big guns:
Sniffing Traffic

We have patches available for parsing mesh traffic.
Outline

1. Mesh Routing
2. The OLPC Mesh
3. Internetworking
4. Application Development
5. Debugging
6. Mesh In Action
   - Mesh Maps
   - Extended Range Demo
lsmeshd + mesh potato

Goals:
- Find nodes in the mesh
- Generate 'maps' of the mesh

Approach:
- Find one-hop neighbors:
  - Set mesh TTL to 1.
  - Broadcast a request.
  - Receive unicast responses.
- Ask them for their neighbors:
  - Reset mesh TTL.
  - Send unicast requests to neighbors.
  - Receive neighbor lists from neighbors.
lsmeshd + mesh potato
Extended Range
Extended Range
Extended Range
Extended Range
Pointers

- OLPC git (see drivers/net/wireless/libertas/README)
  git://dev.laptop.org/olpc-2.6
- Mesh Portal Utils
  http://www.cozybit.com/projects/mpp-utils
- lsmeshd + meshpotato
  http://www.cozybit.com/projects/lsmesh
Thank You!

Thank you and Happy Meshing!