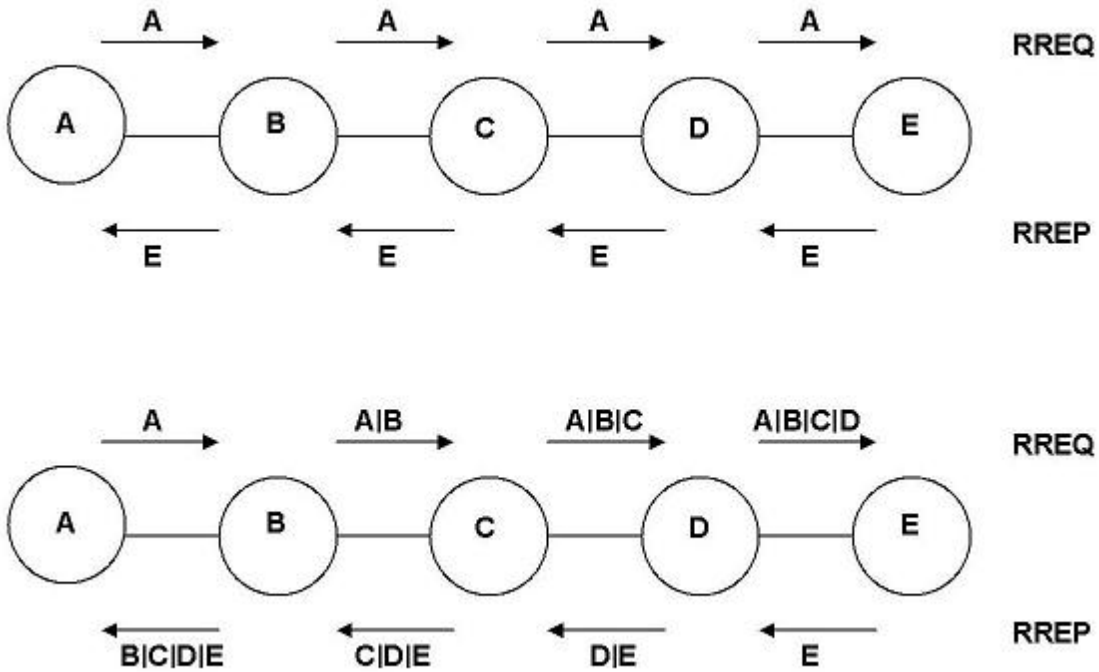
Better Plumbing for Reduced Flooding

Charles E. Perkins

(with Karim Seada, Cedric Westphal, and Mahesh Sivakumar)

IETF 68

Path accumulation, schematic



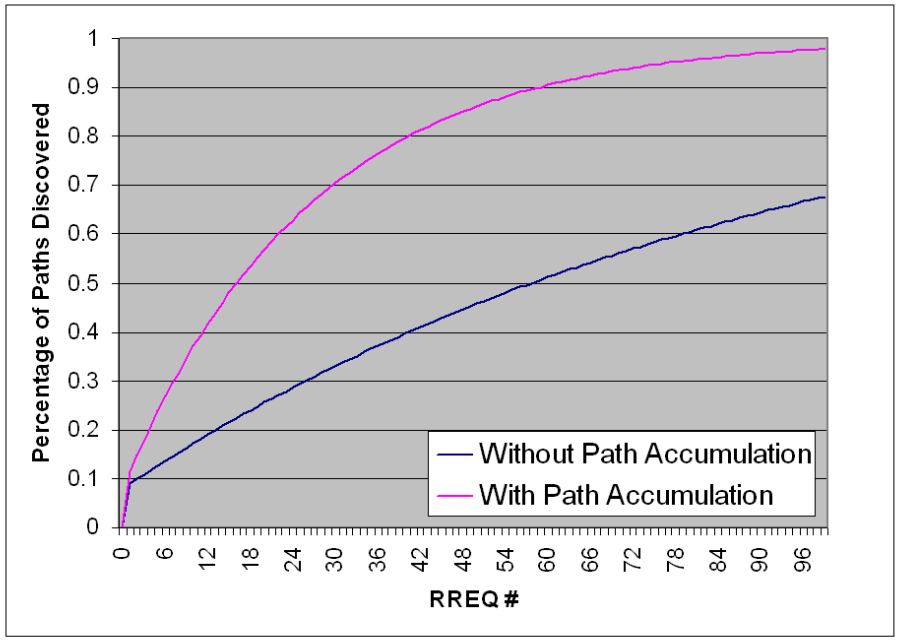
- Equip RREQ and RREP with more topology data
- Longer routes allow acquisition of more data

Path accumulation

- DYMO specifies an extension for this purpose
- Results show improvements in some scenarios, sometimes no change, and sometimes slight deterioration
- When basic signaling gives very high PDR, then path accumulation will not improve it
- Reducing RREQ will allow higher node density without producing congestion

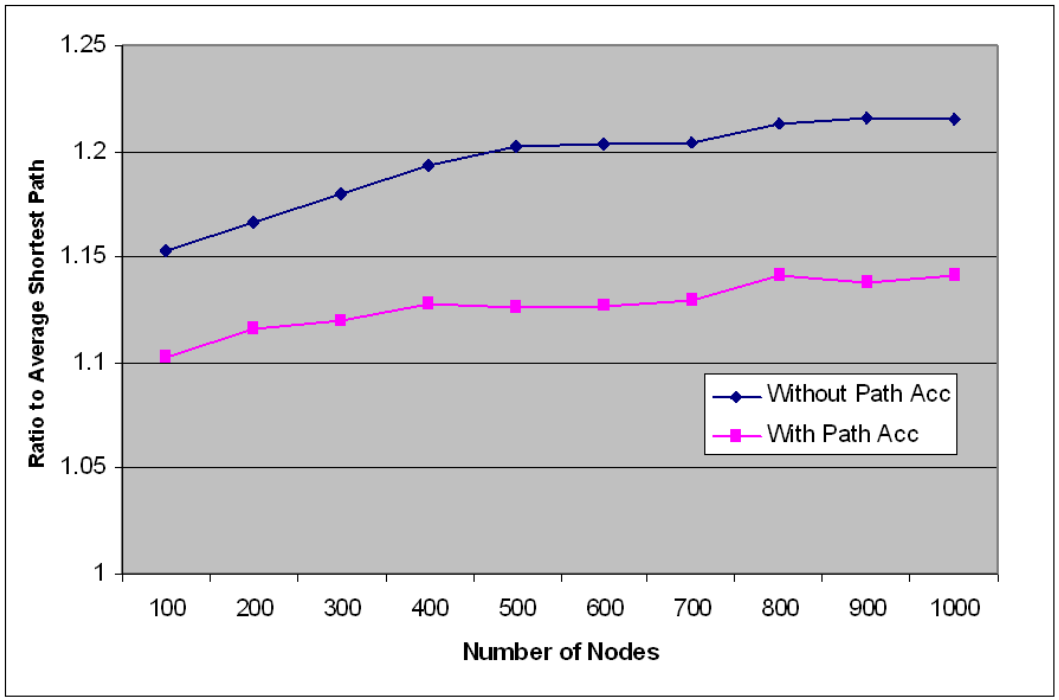
Pre-empting Route Discovery (analytical result)

With path accumulation, topology information is discovered much more quickly



We now believe this graph overstates the improvement, but there is definitely substantial improvement anyway

Route length reduced



- Another benefit from intermediate node RREP

Overall performance of Path Accumulation

- Path accumulation definitely reduces the number of RREQs
- However, it also increases the packet size
- And, the benefit is reduced if newly discovered routes are not used before being purged from the routing cache
- Needed: avoid replaying redundant updates
- Packet size is often a burden that negates some of the benefit of path accumulation
  - Heed this as a warning against packet bloat!!

Recent results for SMURF

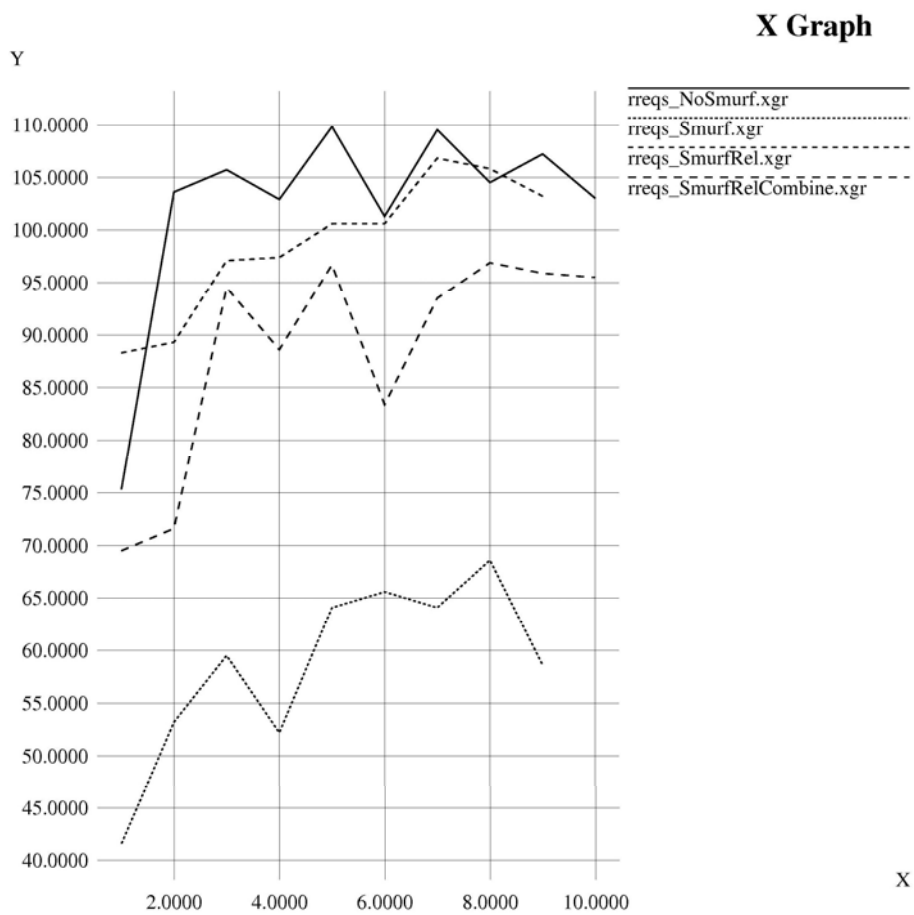
- GOAL: a CDS backbone for DYMO
- Simplified Multicast Routing and Forwarding (SMURF)
  - A modular flooding component for any protocol
- Shows increased PDR under recent tests
- Has a component for reliable flooding
  - BUT – making broadcasts reliable increases congestion

Simulations

- AODV routing protocol (we had the code)
- ns2, 802.11b with CTS/RTS MAC, two-ray ground propagation
- Static networks, random uniform distribution, 100 to 1000 nodes
- Traffic model: Each node sends 1 packet to a random destination

Greatly reduced # of RREQs

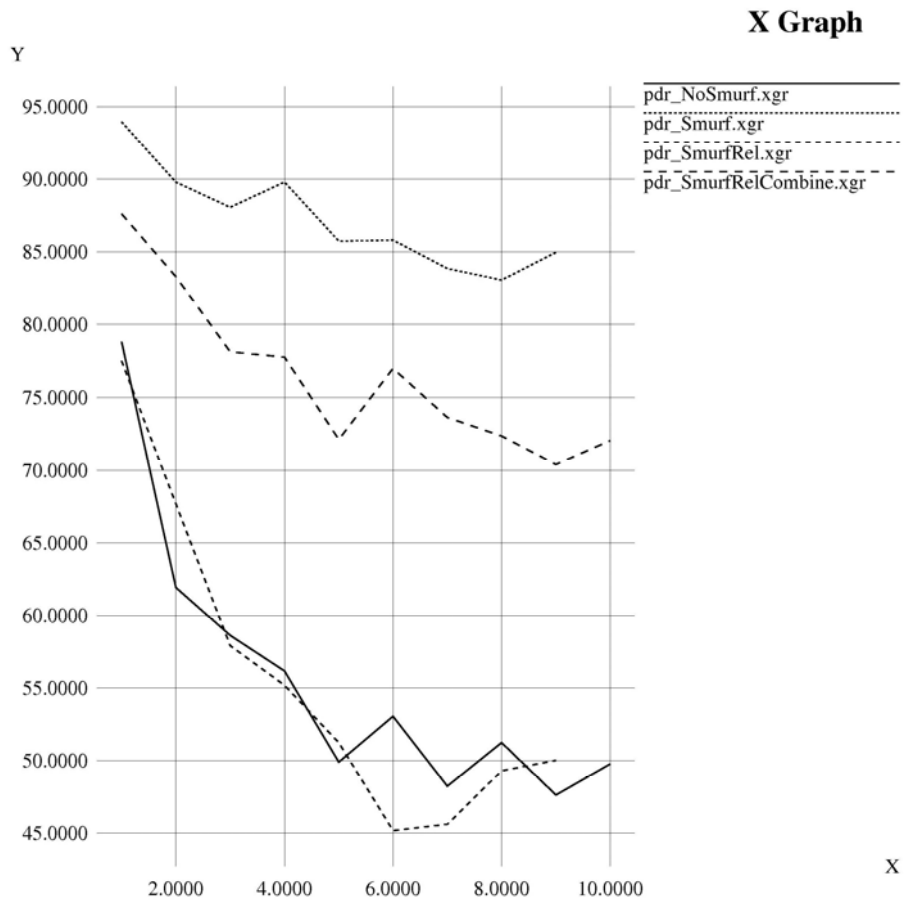
- SMURF backbone does its job very well!
- Reliable SMURF causes *almost as many* RREQs as regular AODV!
- X-axis: 0→1000 nodes
- Y-axis: # of RREQs in 1000s



Greatly improved PDR

SMURF without any reliability signaling enables AODV to perform quite a bit better than base AODV.

- X-axis: from 0→1000
- Y-axis: Packet Delivery Ratio
- Why does reliability hurt?!
  - Culprit seems to be additional signaling overhead



Proactive vs. Reactive

- Which is better:
  - maintain routes for all destination at every node?
  - to acquire a route only when needed?
- Cannot answer without more information:
  - What are the expected traffic patterns?
  - What is the allowable application latency?
  - What is the subnet structure, if any?

Proactive vs. Reactive (pg. 2)

- Generally, sparse traffic favors reactive
  - OLSR expected to be better for dense patterns
  - AODV, DYMO expected to be better for sparse communications
    - most proactive information is wasted in that case
- Where is the crossover? -- in other words, when does the traffic pattern begin to favor proactive? [measured against percentage of N^2 possible communications]

What we started to do…

- 100% == traffic pattern is when there are N * (N-1) traffic flows
- Measured the PDR vs. percentage of 100% traffic flows, OLSR & AODV over backbone
- As expected, sparser traffic patterns favor reactive protocols
- There are exceptions!
  - e.g. routes to an Internet Gateway should be maintained proactively

very preliminary results

- For 40 nodes, the PDR crossover is around 30%
  - → OLSR if favored if each node will maintain traffic at all times with 12 other nodes
- For 50 nodes, the crossover is around 20%
  - → OLSR if favored if all 50 nodes will maintain traffic at all times with 10 other nodes
- For 60 nodes, the crossover seems to be around 7-10%
  - i.e, all 60 nodes must be communicating with 4-6 partners at all times before OLSR becomes effective…

Future Directions

- A more accurate analytical model for PA
  - Plus, try out a half-dozen ideas for debloating
- Extending the PA analytical model to include mobility
- Verifying/sharpening all of the results reported here
  - Does network density affect proactive vs. reactive??
- Why doesn’t reliability help??!
- Less overhead of backbone flooding,
  - Needed? bundling with neighborhood discovery
- Finalize comparison of OLSR with DYMO
  - Determine value of distinguished node routing
  - Create unified/adaptive routing protocol
- Compare against recent “chordal” algorithms

Needed soon

- Make sure DYMO specification works well with a reduced relay set algorithm (e.g., SMF, or possibly revamped SMURF document)
- Sharpen applicability statement for current standards track protocol documents
- Carefully analyze current specifications for packet bloat
- Improve utilization of path accumulation
- (?) Improve modularity and signal bundling