UNIVERSITY OF STIRLING

DEPARTMENT OF COMPUTING SCIENCE AND MATHEMATICS

Multi-Destination Routing and the Design of Peer-to-Peer Overlays

Authors

John Buford

Panasonic Princeton Lab, USA.

Alan Brown, Mario Kolberg

University of Stirling, Scotland.

Panasonic

ideas for life

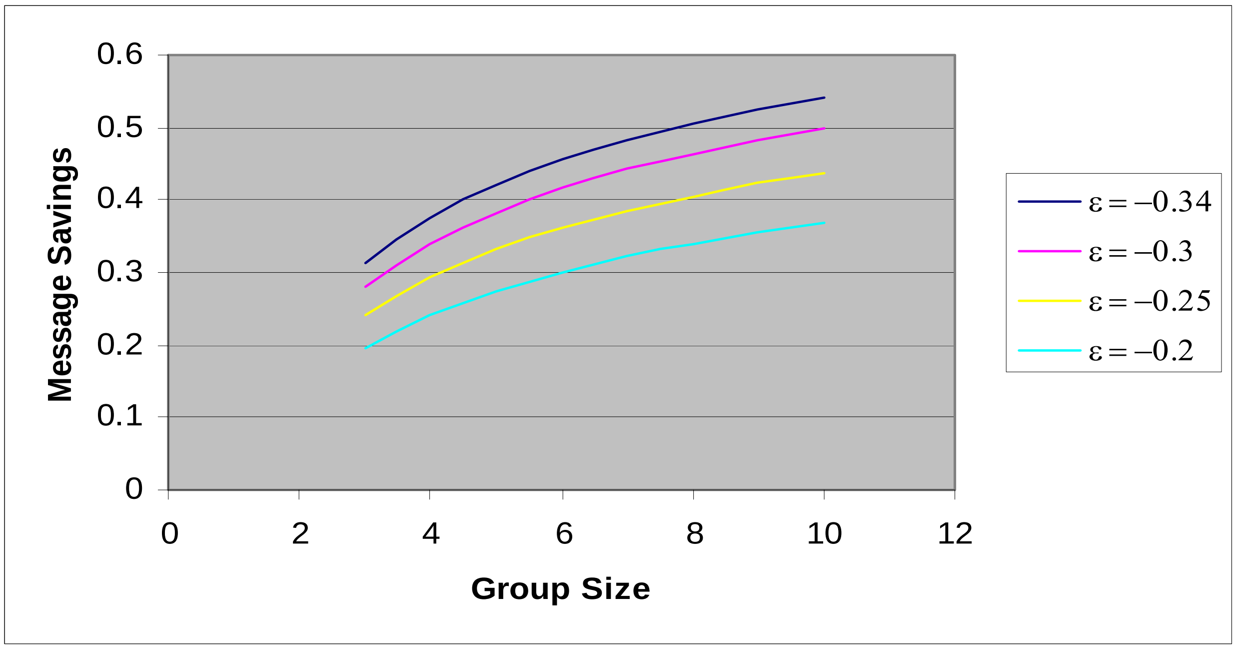
Problem Statement / Motivation

1. Overlay Networks provide widely available end-to-end network services that would be difficult to deploy in physical networks.
2. O(1)-hop overlays have better latency characteristic than multi-hop overlays, but require more maintenance traffic.
3. Distributed Hash Tables (DHTs) are the basic indexing mechanism for large scale decentralized peer-to-peer systems.
   - How to obtain best performance in a large-scale wide area context for DHT operations is an important question.

Can we use multicast as a performance enhancement?

Why would we want to multicast?

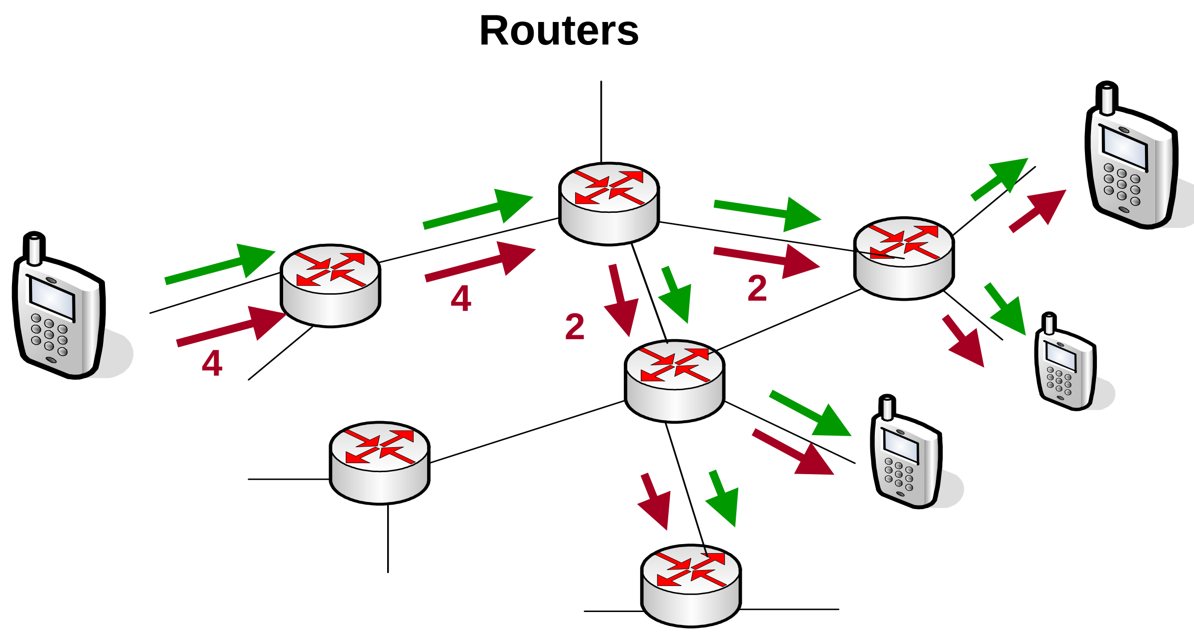
- Chuang-Sirbu multicast scaling law says message savings are related to group size: $1 - m^{-\varepsilon}$, $-0.34 < \varepsilon < -.2$
- 5-way saves 28% to 42%, 10-way saves 37% to 54%



Why hasn't this been used already?

- All multicast protocols in Internet use host group model
  - Each group has unique group address
  - Each IP multicast router maintains routing state for each group
  - PIM-DM, PIM-SM, PIM-SSM, DVRMP, CBT
- Routers maintain per group state
  - Scales well for extremely large multicast groups
  - Scales poorly for large numbers of groups
  - Requires time to create group state throughout the network
  - **Doesn't fit peer-to-peer overlay characteristics**
- We propose to use multi-destination routing model
  - No state in routers, no time to create group needed
  - Scales well for large numbers of small groups
  - Group size is limited to about 50
  - **Fits most cases of interest in parallelizing peer-to-peer overlays**

Multi-Destination Routing



Unicast packets

Multicast packets

Example implementation : XCAST (Explicit multi-unicast)

Panasonic
ideas for life

Criteria for parallelization

- Criteria for determining whether overlay messages can be parallelized using multicast.
  - maximum group size
  - number of groups
  - time to create a new multicast group
  - group formation rate
  - temporal locality of messages
  - overlap of message content
- Multi-destination routing can be used in several categories of overlays for various overlay operations
  - DHT operations (Kademlia, EpiChord), overlay maintenance (EDRA*), replication (Beehive), and measurement (Meridian).
- Multicast savings for two overlay algorithms based on simulation results (EpiChord, EDRA*) are described in this paper.

When is multicast suitable for implementing overlay operations?

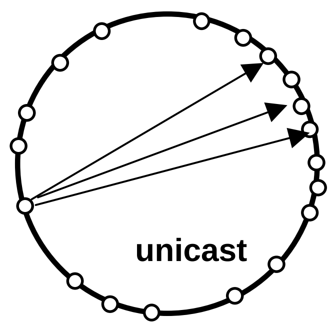
- Scalability of the multicast mechanism for number of groups and group size meets the scalability requirements of the overlay.
  - If C is the capacity of the network to support simultaneous multicast group state for this overlay, then $N_G \leq C$.
  - If v is the maximum group size supported by the network, then $|g_{max}| < v$.
- Overlay's rate r of group formation and group membership change must be sustainable by the multicast mechanism.
- Time to create a new multicast group $t_c < d_{q,}$ the maximum allowed delay time in the peers outbound queue .

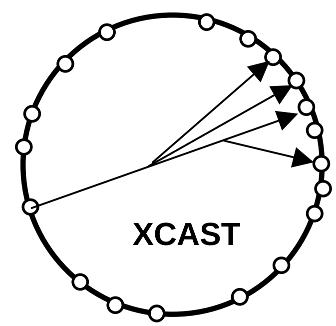
Results / Simulation

- To determine whether multi-destination routing is applicable to a number of different Overlay systems, we either simulated or modelled its application in:
  - EpiChord (simulation).
  - EDRA (simulation).
  - Kademlia (model).
  - Beehive, Meridian and Random Walk (models – see paper).
- Simulations were carried out using a 10,450 node network in the SSFNet simulation environment. Overlay sizes varied from 1k to 9k nodes.

Simulation: EpiChord O(1)-hop overlay

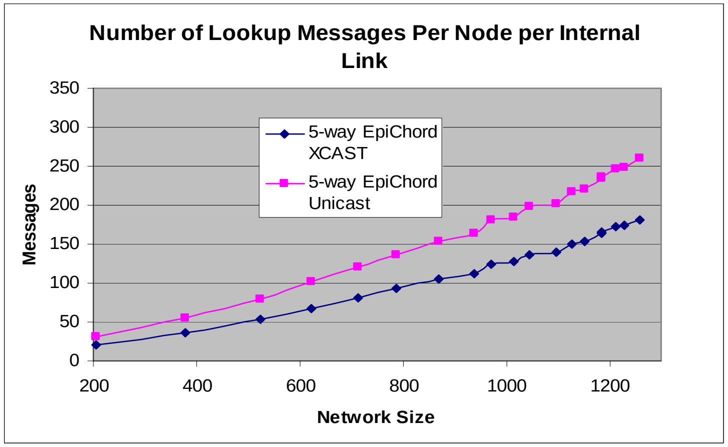
- Routing table is organized in slices
- Slice density is highest in region near peer
- Each slice must have at least 2 entries
- DHT lookups and slice maintenance use parallel unicast requests
    - Failed responses are used iteratively to update routing table and narrow the search

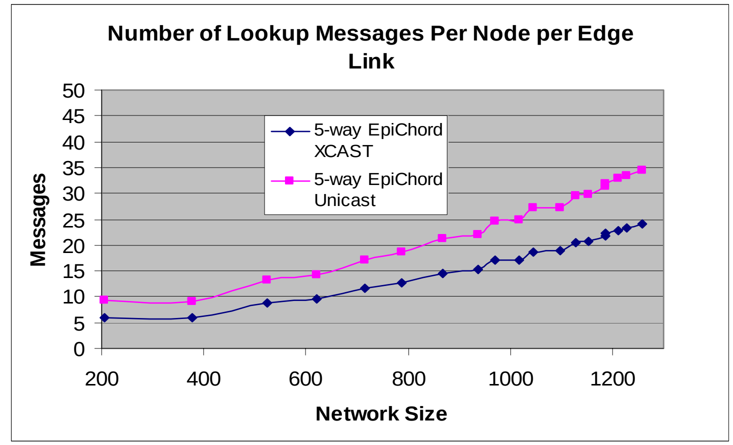




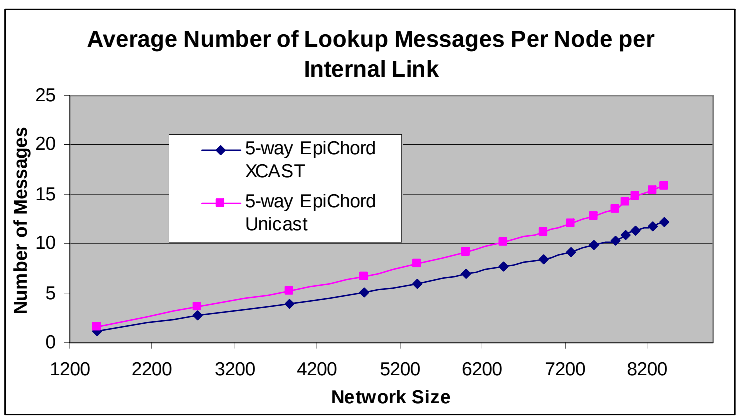
Unicast vs XCAST EpiChord

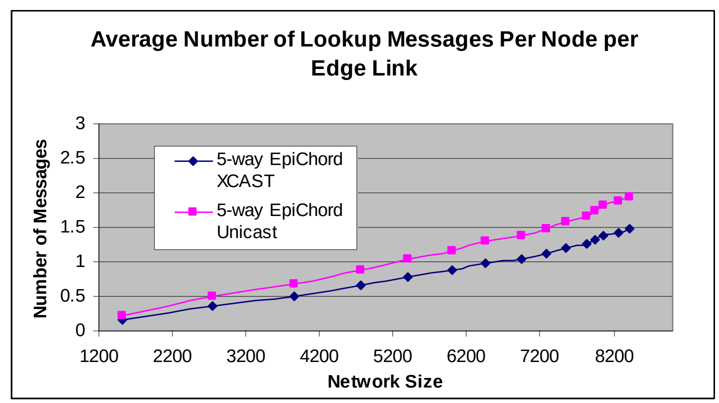
Lookup intensive workload, 1K peers



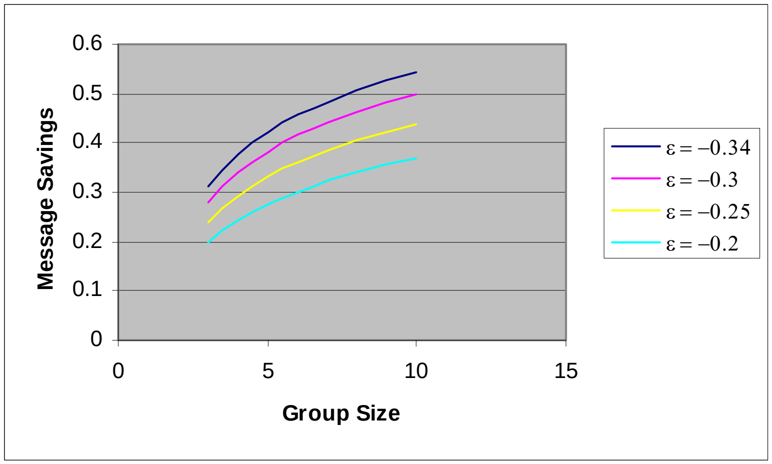


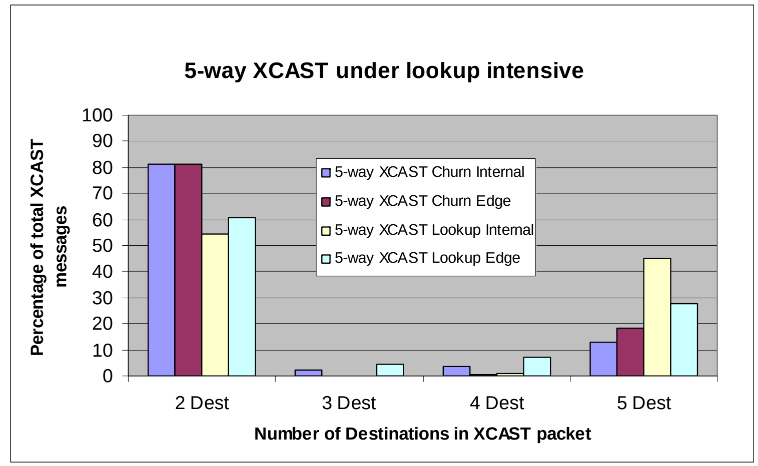
Churn intensive workload, 9K peers





EpiChord Savings vs Chuang-Sirbu Multicast Scaling Law





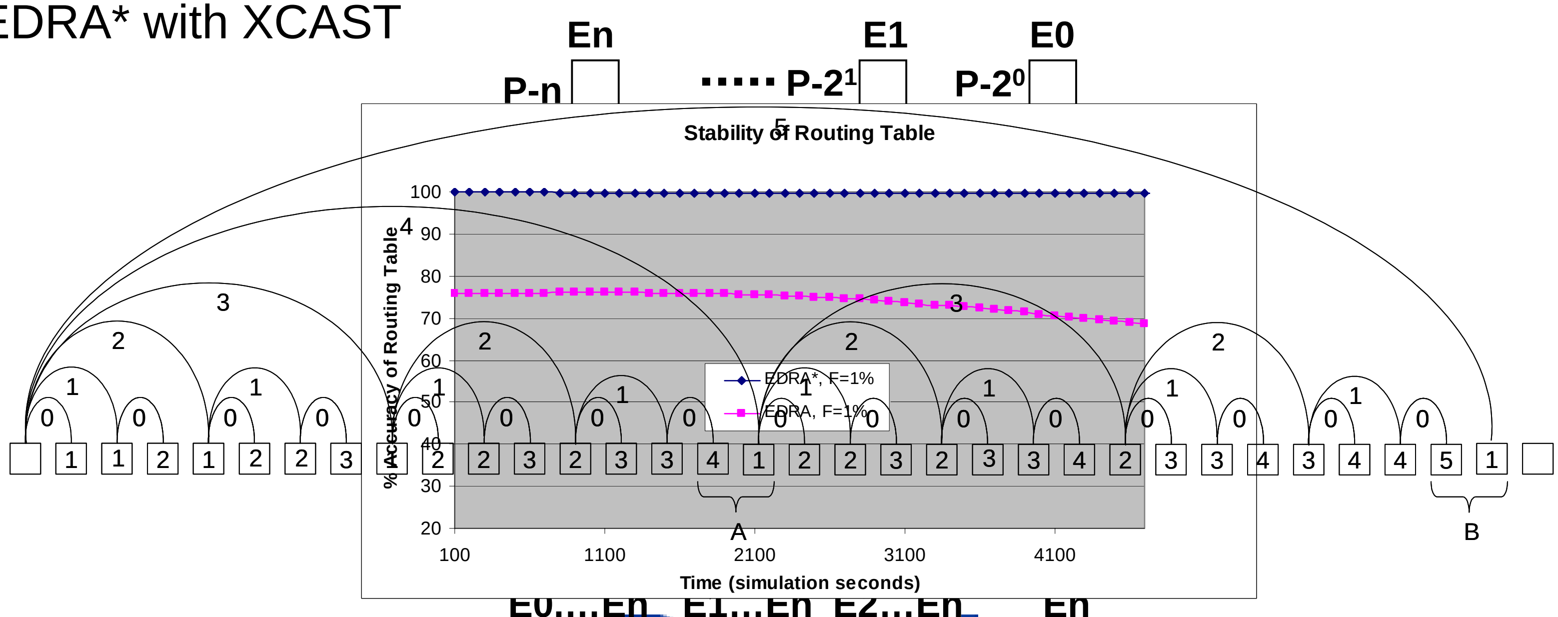
Chuang-Sirbu:
Multicast savings = 1 - m $^{-\varepsilon}$, $-0.34 < \varepsilon < -.2$, **where m is multicast group size**

- 5-way EpiChord actually sends 5-way, 2-way and unicast requests
  - Timeouts cause retries
  - NAKs cause additional queries
- 5-way EpiChord savings is about 30% for both edge and internal link
  - Consistent with Chuang-Sirbu for ε=-.3, based on combination of 2-way, 5-way

- *Validated with Markov model*

Simulation: EDRA (used in D1HT)

- EDRA (Event Detection and Recording Algorithm)
- Each peer collects join and leave events
- Propagates events to (log n) successors
- No peer receives duplicate events
- We fixed 6 problems with published EDRA algorithm and simulated EDRA* with XCAST

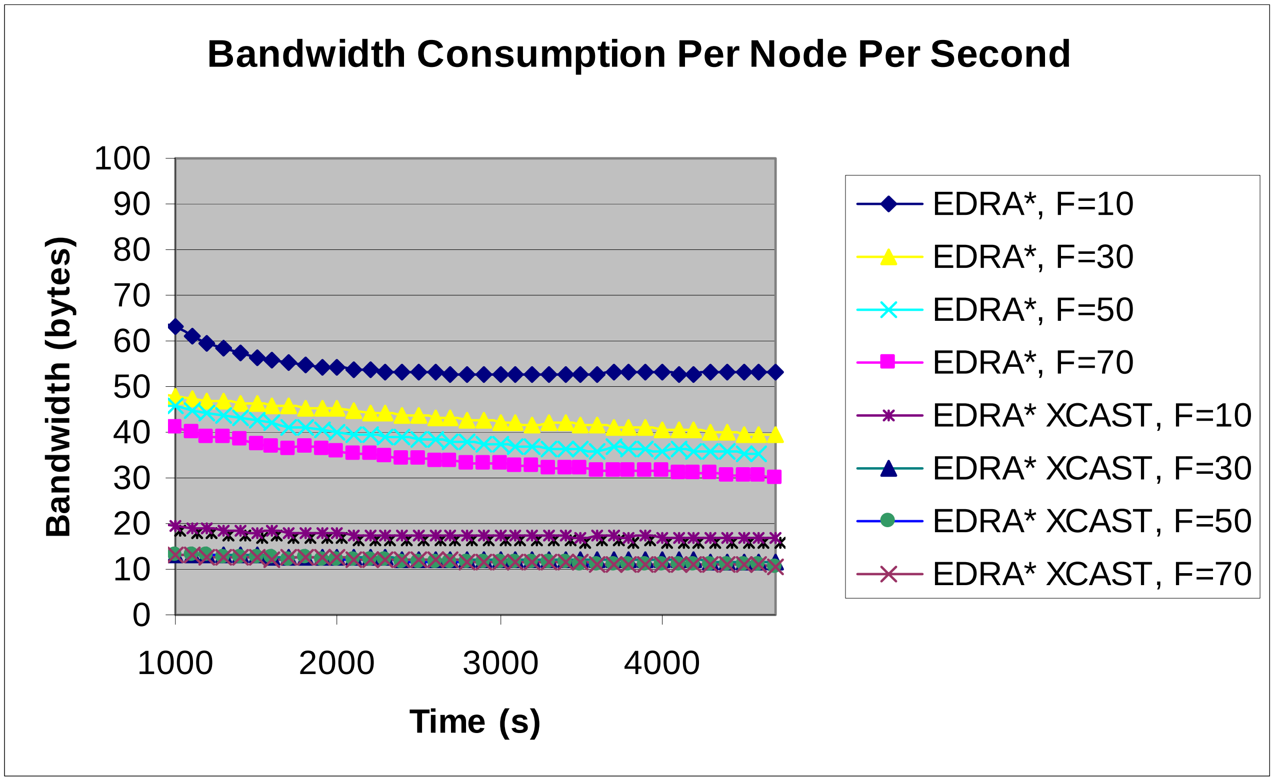


EDRA* - Improvements

EDRA* Technique	Summary
Explicit join interval	Joining node gets events from node which provided copy of routing table
Correct join point	Successor node checks new predecessor is in correct position
Forwarding of un-acknoweldged events	Events are forwarded to successors of peer
Handling of duplicate events	Forwards duplicate events that occur due to routing table errors
Detecting concurrent adjacent events	New nodes contact both successor and predecessor nodes
Event cache propagation	Events are cached and forwarded as routing table changes reach to new nodes in overlay

EDRA* vs EDRA*-XCAST

Overall savings about 33% for n = 1024



Kademlia

- Multi-hop overlay uses XOR as distance metric
- Bi-directional iterative lookups
- Node lookup
  - Sends parallel requests to peers.
  - Responses return closer nodes.
- Peer does at least k/α iterations for a node lookup in a given bucket (list of nodes a peer knows about, ordered by last seen). Each bucket covers a section of the ID space.
  - For k = 20 and α = 3, that is 3-way queries to seven multicast groups
  - With 160 buckets each peer would need at least 160 groups to do queries across its address space.
  - If the multicast queries were α-way, Chuang-Sirbu estimates a 20% to 30% savings.
  - If the queries were k-way, k=20, Chuang-Sirbu estimates a 45% to 64% savings from multicasting Kademlia requests, although responses would be unicasted.

Conclusion

- Parallelizing overlay operation using multi-destination is a generally applicable technique.
- Savings can be easily 30% or more for systems that were not explicitly designed to use multi-destination routing.
- Requires network support.

Questions?

Thank you for your time!