Lcast: LISP-based Single-Source Inter-Domain Multicast

LISP WG, IETF-83, Paris

Florin Coras, Albert Cabellos and Jordi Domingo
Universitat Politècnica de Catalunya

Fabio Maino and Dino Farinacci
Cisco Systems

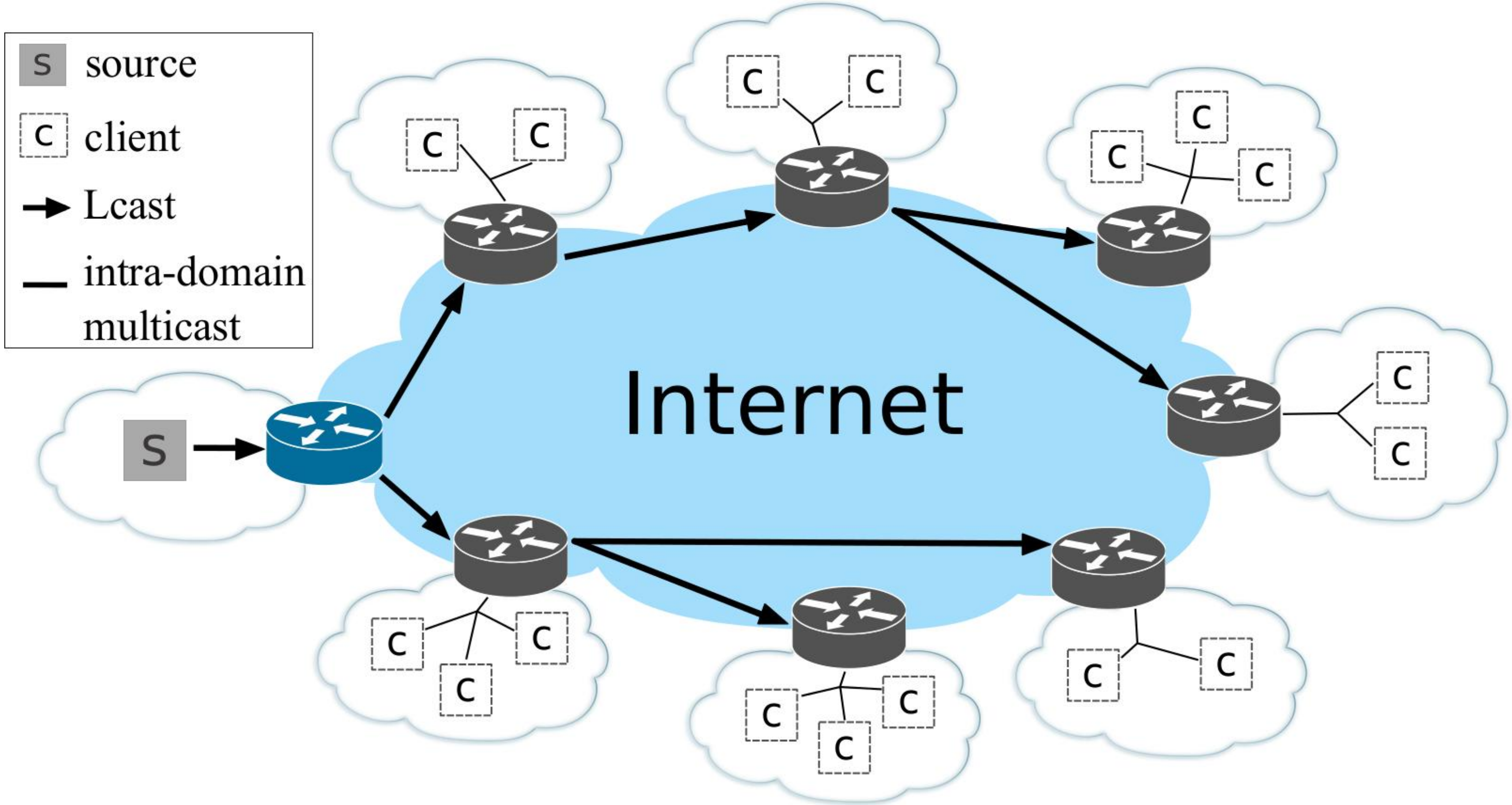
Goal

- Build a network-layer single-source multicast framework
- Combine the efficiency of IP-multicast with flexibility of Application Layer Multicast (ALM)
  - Easy to deploy
  - Scalable
  - Configurable performance based on operational requirements

Inter-Domain Multicast

- Three sub-problems
  - Intra-domain multicast
  - **Inter-domain multicast**
  - **Interface between the two**

Lcast


S source
C client
Lcast
intra-domain multicast
Internet
S
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C

Architecture Overview

- Interfaces with end-hosts by means of intra-domain multicast
  - ITRs convert intra-domain multicast subscriptions to Lcast subscriptions
- Data plane:
  - Over the Internet's core xTR overlay
  - xTRs unicast replicate content
- Control plane functions centralized in the MS of the source domain
  - Group management
  - Distribution tree optimization
    - Centralized performance control

Group Management

- Join
  - Initiated by end-hosts and intercepted by xTRs
  - xTR subscription requests are routed by the mapping system to the source domain's MS
  - For a joining xTR, the MS finds an overlay parent and requests that it replicates content to the newcomer
- Leave
  - A member leaving the overlay announces its intentions to the MS (graceful leave)
  - MS acknowledges the leave but only after it finds new parents for the leaving node's children

Distribution Tree Optimization

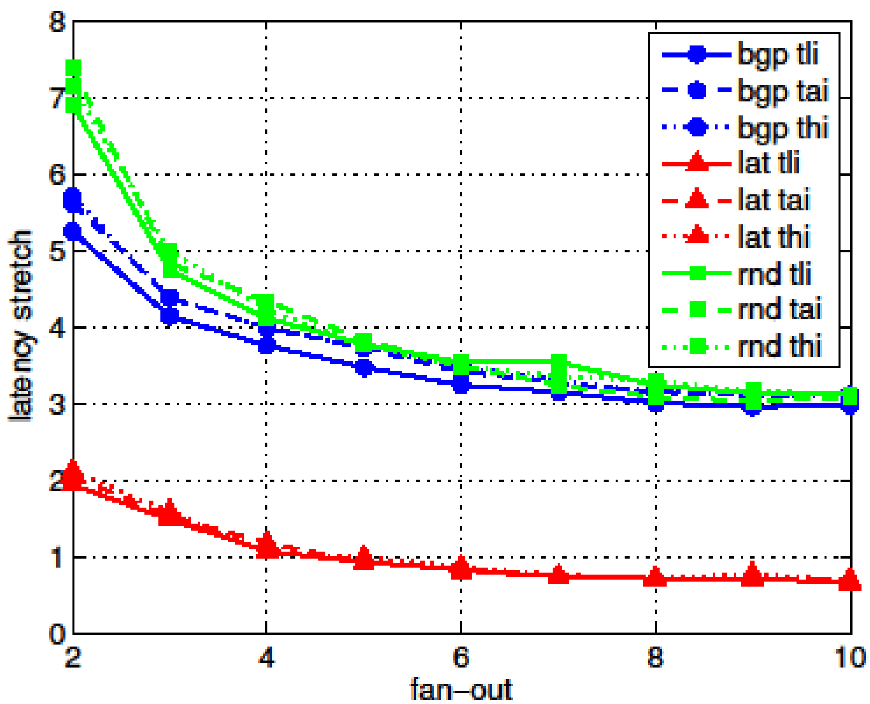
- Initiated by the MS
- Optimization algorithm
  - Degree-bounded spanning tree
  - Minimum average end-host *distance* to root
- Topology discovery
  - **BGP**: MS makes use of the BGP RIB in the xTR of the source domain to infer the number of AS hops between members
  - **Latency**: MS requests nodes to measure their latencies to a subset of peers according to an heuristic

Evaluation

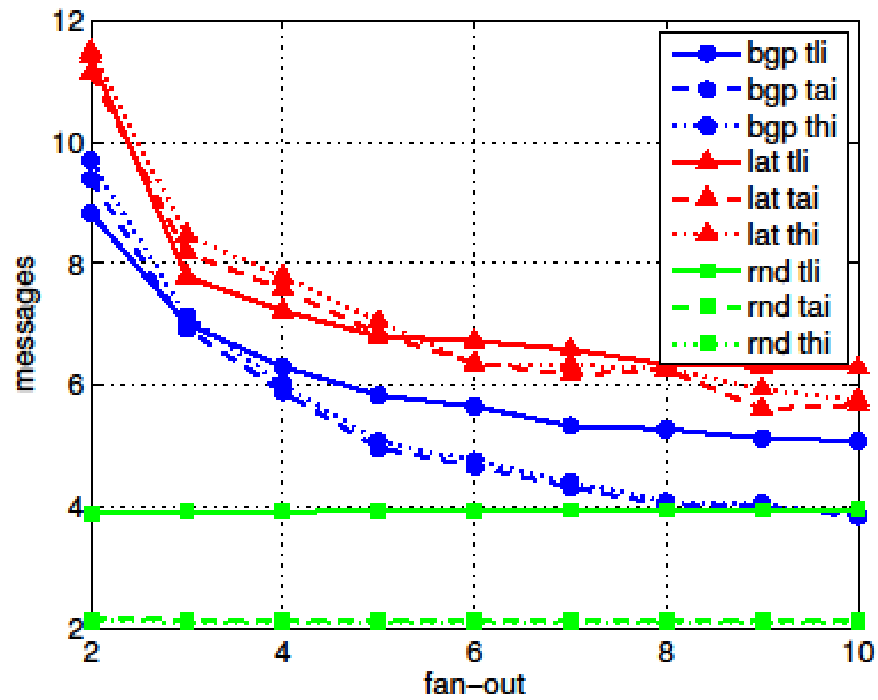
- Internet-scale simulator
- We generate an Internet-like AS topology
  - We aggregated topology information from: CAIDA, RouteViews, RIPE, iPlane
  - Latency information from iPlane
- Generate traces of client arrivals and departures
  - End-host distribution in Ases
    - Passive capture of P2P TV traffic
    - 146k unique IPs in 3.8k ASes
  - Model client churn

P2P TV Traces Capture Points

Results

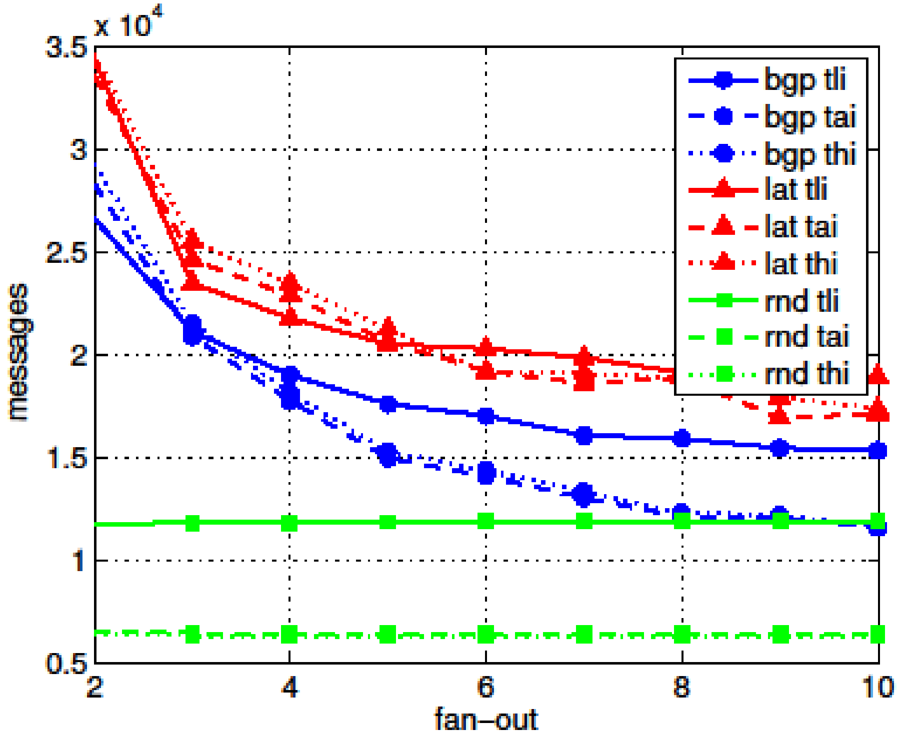


Latency stretch

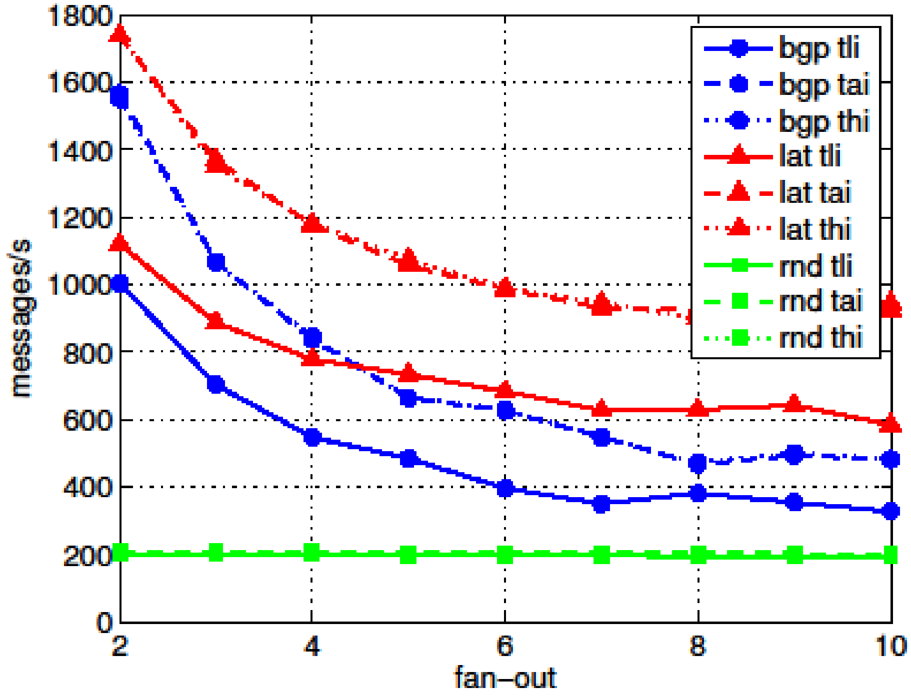


Av number of messages/xTR

Results



Number of messages/MS



Peak messages/s for the MS

Results Discussion (1)

- Control overhead is easily manageable
  - For 3k overlay members
  - Even when using active topology discovery
- Client churn
  - Slightly influences performance
  - Increases management overhead
- Fan-out
  - Values larger than 6 offer limited benefits

Results Discussion (2)

- The latency-based optimization strategy
  - Offers unicast-like average latency and tight bounds
  - Larger, but manageable, control overhead
- BGP and random behave similarly
  - AS hops don’t do a good job at estimating latency
  - Radom could be used as backup optimization strategy

Draft?

- Lcast architecture draft?

Backup Slides

Lcast Constraints and Assumptions

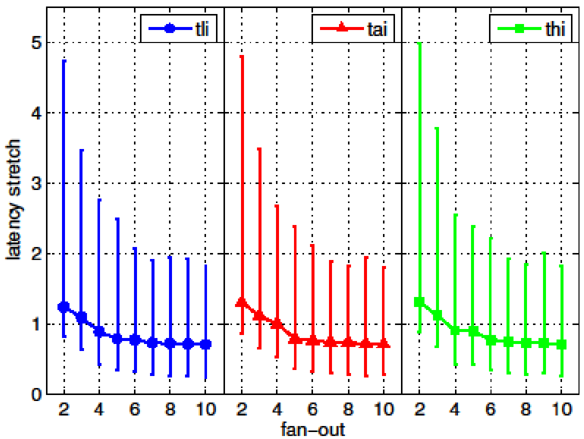
- No changes to host stacks
- Centralized group management
- Member (Router)
  - Constrained fan-out
  - Graceful leave
  - Connectivity robustness
- IGP stability

Distribution Tree Optimization Algorithm

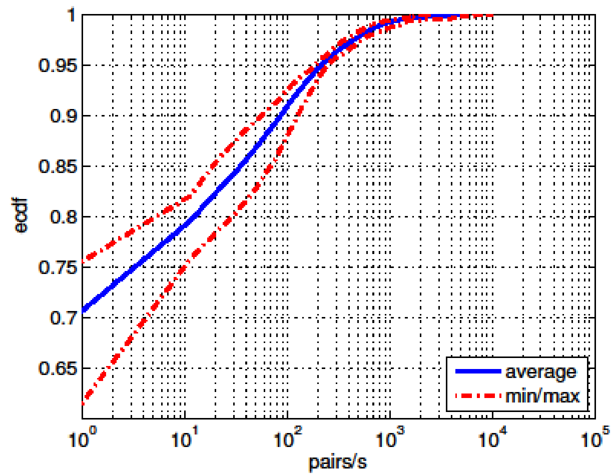
- Minimum Average Distance Degree-Bounded Spanning Tree (MADDBST)
  - For a graph where vertices have weight, it constructs a degree-constrained spanning tree with the lowest average distance/weight to the root.
- When weight is clients it converts to a degree-constrained ST with the lowest average distance/client to the root.
- NP-complete
  - Solved with an heuristic that incrementally grows a spanning tree starting at the root node while minimizing the distance/client of every member xTR

Results (3)

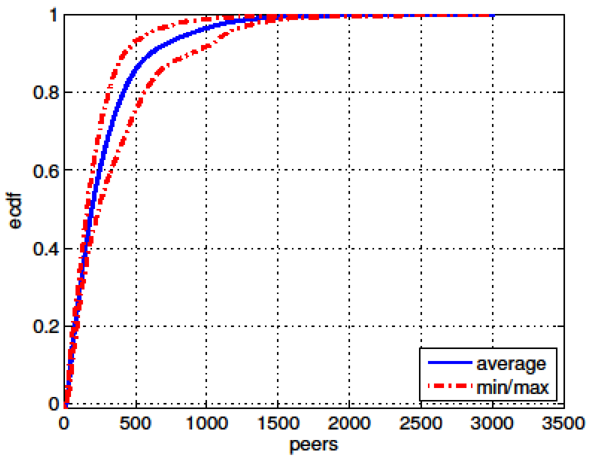
Results for latency topology discovery



Latency stretch with 95% confidence interval



ECDF peer pairs measured/s



ECDF peers measured/member