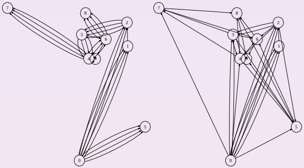
The performance of locality-aware topologies for peer-to-peer live streaming



R. G. Clegg (richard@richardclegg.org), D. Griffin, R. Landa, E. Mykoniati, M. Rio,
Dept. of Electronic and Electrical Engineering, UCL

UK Performance Engineering Workshop, Imperial College 2008

(Prepared using LaTeX and beamer.)

Problem area

Motivation

- Current research interest in peer-to-peer live streaming.
- Peer actions must be largely distributed.
- Want low start-up and end-to-end delay.
- Network co-ordinates give a distributed delay estimation tool.
- Given delay info, how should peers choose partners?

- Preliminary research: simple, low parameter simulation of overlay network.
- Easily measured approximate metrics no dependence on exact peer details.
- Experiment with number of peers and assumptions about the network.

Peer-to-peer live streaming

Terminology

- Stream – the data to be sent live.
- Peercaster – peer from which the stream is originating.
- Delay space – a space (not metric) where the distance between peers is their delay.
- Churn – the turn over (leaving and joining) of peers.

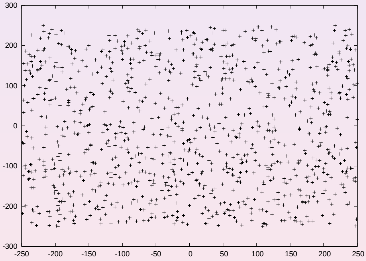
Given a distribution of peers on a delay space.

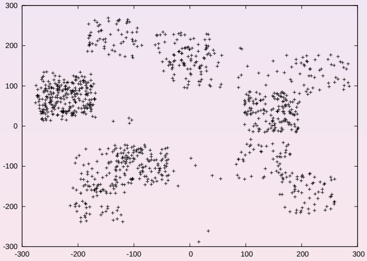
- Want good end-to-end (peercaster to peer) delay, not throughput.
- Want good reliability even in high churn.
- Want "fair" load on peers.
- Want to explore a reasonable amount of the simulation parameter space.

Delay space

Delay estimate here distance in 2D Euclidean space.

1. Flat peer distribution $\mathcal{N}_F$ – (x_i, y_i) with x_i and y_i chosen from flat distribution $(-D, D)$.
2. Loosely clustered peer distribution $\mathcal{N}_L$ – $(x_i, y_i) = (X + \varepsilon^i_X, Y + \varepsilon^i_Y)$, plus a small probability of moving flatly distributed (X, Y).
3. Tightly clustered peer distribution $\mathcal{N}_T$ – as above with smaller variance on ε.





Experiment details

- Distribute $N+1$ peers $(0,\dots,N)$ in the delay space and pick subset $n \leq N+1$ for experiment.
- The stream has fixed bandwidth B.
- Peer 0 is the peercaster and has capacity u_i. (Only one value tried.)
- Peers $i > 0$ randomly allocated some upload capacity u_i from a distribution. (Only one distribution tried.)
- Peers join in order (for local topologies) and attempt to download M streams of bandwidth B/M. (Only one value for B and M tried here.)
- Distribution must be such that $\overline{u_i} > B$ and $\sum_{i=1}^{j} u_{i-1} \geq Bj$ for all $j \leq n$.

Experiment details (2)

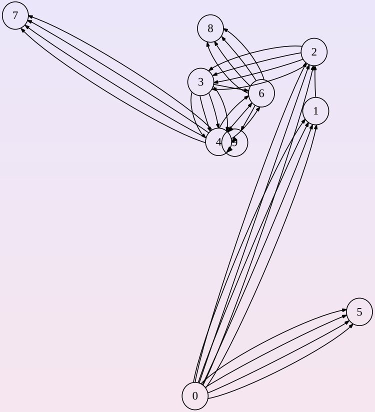
- As every peer joins it chooses M peers (not necessarily distinct) with spare capacity to upload from.
- Algorithm for choosing peers is a strategy for creating a topology.
- After n peers join, then output the resultant overlay network and measure metrics on it.
- Vary n, the peer distribution and the topology creation strategy.
- Repeat each experiment ten times to create a mean and a 95% confidence interval.
- In the full paper global topology strategies are also described (all peers present before peer selection).

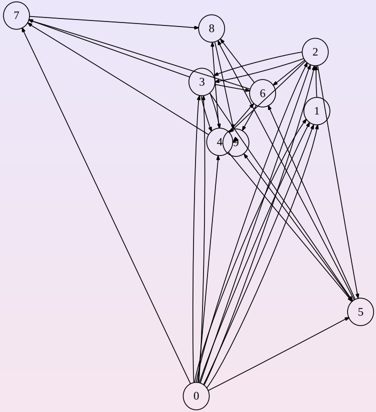
Topologies investigated

Strategies referred to as *local fixed topologies* $\mathcal{TLF}$ (for peer i nodes with spare upload selected from $j < i$).

- Local random $\mathcal{TLF}_R$ – M random peers selected.
- Local closest first $\mathcal{TLF}_{C1}$ – M peer(s) with least delay to this peer.
- Local closest with diversity $\mathcal{TLF}_{C2}$ – as above but M distinct peers if possible.
- Local minimum delay first $\mathcal{TLF}_{D1}$ – M peer(s) with least delay to peercaster.
- Local minimum delay with diversity $\mathcal{TLF}_{D2}$ – as above but M distinct peers impossible
- Local small world $\mathcal{TLF}_S$ – This topology has $M - 1$ using $\mathcal{TLF}_{C2}$ and one peer using $\mathcal{TLF}_R$.

Ten nodes connected with $\mathcal{TLF}_{C1}$ and $\mathcal{TLF}_{C2}$





Metrics used – delay and fairness

Let $D_i(j)$ be the delay from peer i using first hop on connection j and then shortest delay path. Let O_i be the outgoing bandwidth used from node i.

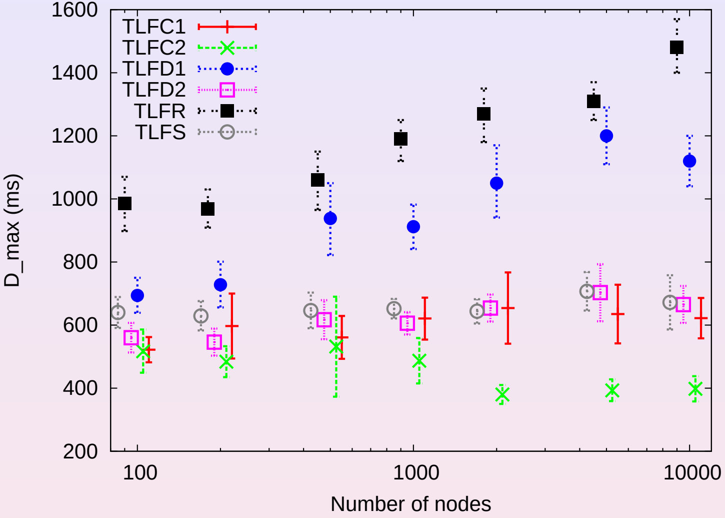
- Mean minimum delay $\mathbf{D}_{\text{min}}$ – $\sum_{i=1}^{N} \max_j D_i(j)/N$, this is the mean of the minimum delay to the peercaster.
- Mean maximum delay $\mathbf{D}_{\text{max}}$ – $\sum_{i=1}^{N} \max_j D_i(j)/N$, this is the mean of the maximum (shortest path) delay to the peercaster.
- Bandwidth variance $\mathcal{B}_v$ – $\text{var}(O_i)$ (for nodes with $u_i > 0$) – reported in full paper.

Metrics used – vulnerability

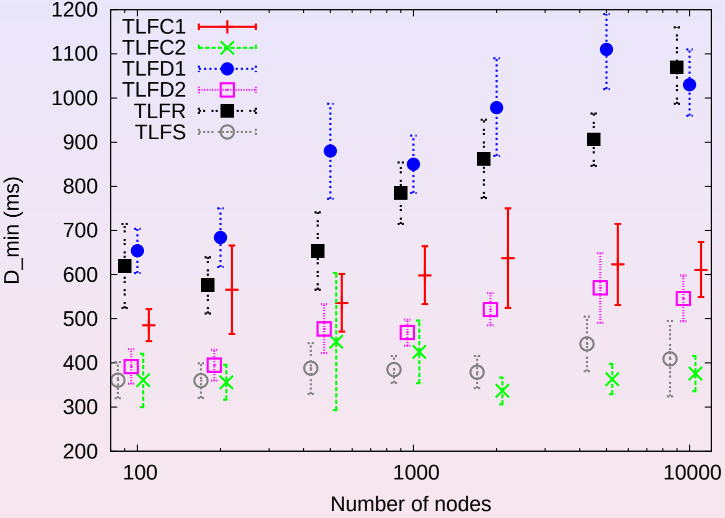
Let V_i (node vulnerability) be the maximum number of paths along $D_i(j)$ from i cut by the removal of one other node. Let S_i (system vulnerability) be the number of paths along $D_k(j)$ cut by the removal of node i.

- Maximum system vulnerability **S** – $\max_i S_i/NM$ – this is the proportion of routes which could potentially be damaged by the removal of a single node. (Related to max Betweenness-Centrality).
- Mean node vulnerability **V** – $\sum_{i=1}^{N} V_i/NM$ – this is the mean proportion of its connections which each node could potentially lose by the removal of a single node.

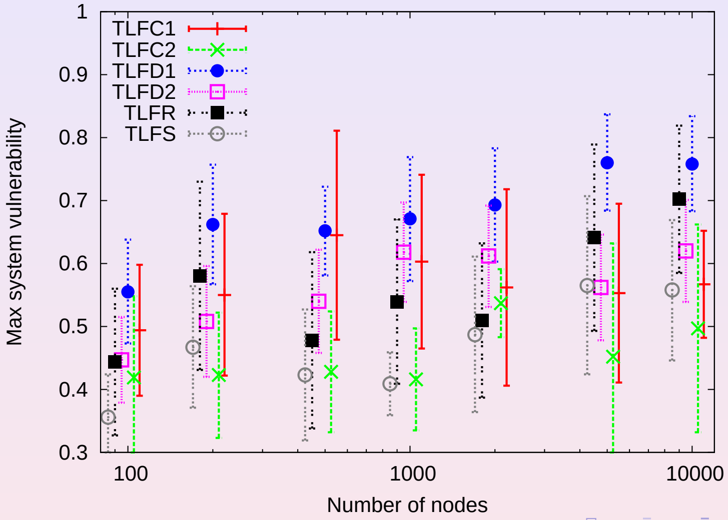
Results for $\mathbf{D}_{\max}$ on $\mathcal{N}_F$



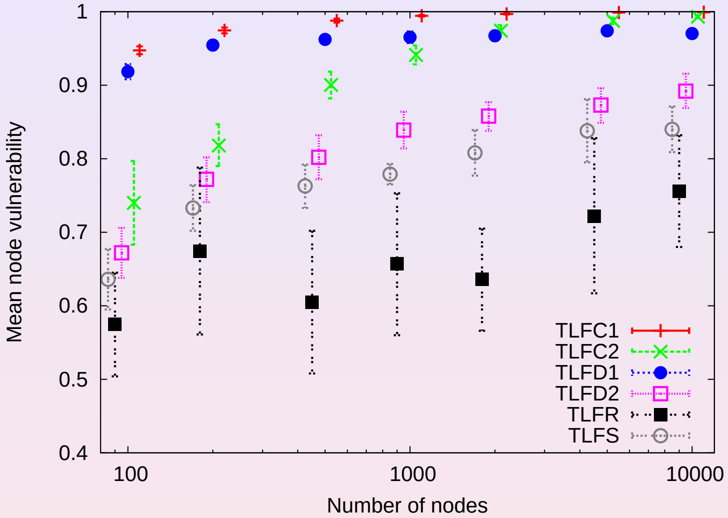
Results for $\mathbf{D}_{\text{min}}$ on $\mathcal{N}_F$



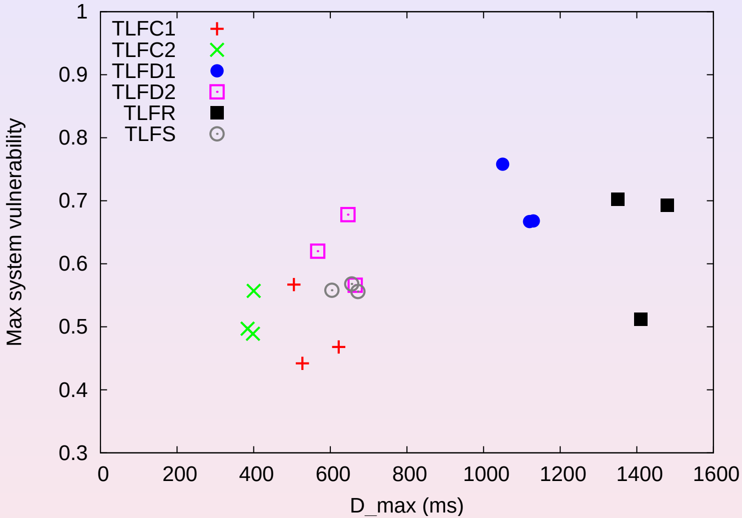
Results for **S** (system vulnerability) on $\mathcal{N}_L$ (loosely clustered)



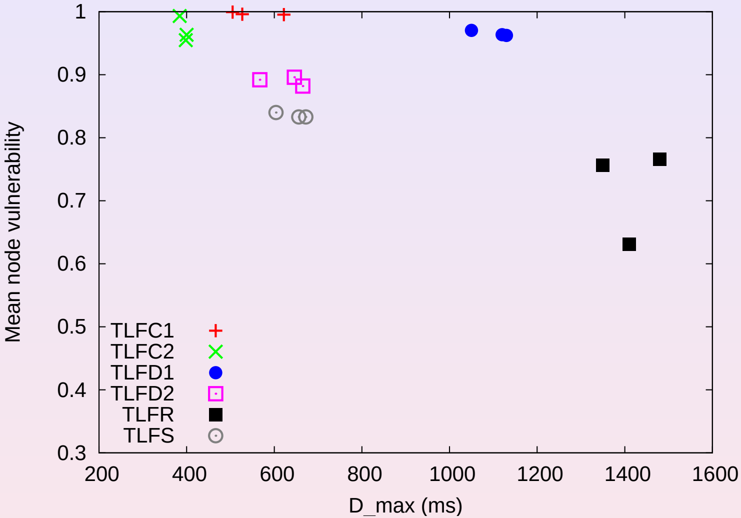
Results for **V** (node vulnerability) on $\mathcal{N}_L$ (loosely clustered)



Results for **S** (system vulnerability) versus $\mathbf{D}_{max}$ all topologies $n = 10,000$



Results for **S** (node vulnerability) versus $\mathbf{D}_{max}$ all topologies $n = 10,000$



Conclusions

- The particular distribution of nodes seemed of little importance.
- The vulnerability measures had a high variance across runs.
- Topology strategies emphasising diversity performed better in most tests.
- Delay and vulnerability measures seem to scale well with size for the best policies.
- Strategies which aggressively minimise delay to peercaster locally do not minimise global delay.
- See full paper for further details `www.richardclegg.org/pubs`.

Further work

- Explore effects of node upload distribution and M parameter.
- Improve metrics, better delay and (possibly) more robust vulnerability estimates (it may be vulnerability is variable).
- Explore $\mathcal{TLR}$ – re-evaluating local topologies which begin as $\mathcal{TLF}$ but are allowed to change connections later.
- Add mathematical analysis – can the local and global problems be formulated as proper optimisation problems.
- Compare results with a more realistic simulation.
- Other suggestions (particularly related to mathematical rigour) welcomed.