Enabling Bandwidth Adaptation in Overlay Networks

Simon Schubert (EPFL) – Frank Uyeda (UCSD) – Dejan Kostić (EPFL) – Antony Rowstron (MSR)

Challenges in Streaming Overlays

- Host-based overlays are popular for real-time media streaming
- A significant number of hosts are behind asymmetric links
- Aggregate outbound bandwidth << aggregate inbound bandwidth
- Users often underestimate the available outbound bandwidth
- Challenge: manage the available bandwidth resources in the overlay

The black node successfully forwards data to its children, the blue and green nodes. However, because black overestimates its outgoing bandwidth, it accepts red at 9 ms. Not only red receives a reduced data rate, but blue and green suffer as well. This in turn propagates to their children.

Bandwidth Adaptation with GateKeeper

- Each overlay node runs an instance of GateKeeper
- Overlay exposes information about peers and stream rate to GateKeeper
- GateKeeper maintains and exposes local and global bandwidth info to the overlay
- This enables the overlay to adapt to the underlying bandwidth availability.

GateKeeper Benefits

- Not engineered into an overlay, but works in tandem
- Provides mechanism that can easily support adaptation policies
- Enables policies to be combined in different ways

Possible Policies enabled by GateKeeper

- Dynamic stream rate scaling
- Accommodates as many users as possible.
- Admission control
- Prevents significant performance degradation for a large fraction of users.
- Dynamic contribution control
- Saves cost & energy and can balance the contribution skew.

Local Probing and Interaction with TCP

- GateKeeper correctly estimates available outbound bandwidth
- Given the information from GateKeeper, the overlay decides to back off in the presence of TCP

Admission Control

When outbound bandwidth is scarce, nodes use the histograms to explicitly reject new joins.

Configuration:
- Nodes receive global capacity histograms
- When < 5% available capacity, nodes explicitly reject peers
- At 30%, 50% of contributing nodes start a TCP upload

Results:
- Nodes rejected when capacity is low
- Excellent performance for admitted nodes

Dynamic Stream Rate Scaling

Using the histograms, the source dynamically scales the bitrate so that there are enough free slots for all nodes to join.

Configuration:
- The source tries to maintain 20% spare capacity
- The source can scale the stream rate in 5 packet steps

Results:
- Source scales down stream rate when capacity is low
- All nodes are able to join the overlay
- All nodes receive the stream

Dynamic Contribution Control

Per-node contribution depends on the spare capacity in the system and its own contribution relative to other nodes.

Configuration:
- 49 contributing nodes configured with 1Mbps in/out bandwidth
- Initial max. contribution level set to 2 children for all nodes
- When < 10% spare capacity, nodes probabilistically increase their max. contribution level if they can

Results:
- Big contribution skew
- All nodes are able to join the overlay
- All nodes successfully receive the stream

Experimental Evaluation

Performance of a streaming overlay when a fraction of the nodes under- or overestimate their outbound bandwidth. For a large fraction of nodes the receive rate is seriously impaired.

Results:
- When < 5% available capacity, nodes explicitly reject joiners
- Nodes receive global capacity histograms
- These histograms are convergasted up a control-channel tree and then multicasted to each node.

Throughput (KBps)

<table>
<thead>
<tr>
<th>Time (seconds)</th>
<th>0</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Admission Control

Configuration:
- Nodes receive global capacity histograms
- When < 5% available capacity, nodes explicitly reject peers
- At 30%, 50% of contributing nodes start a TCP upload

Results:
- Nodes rejected when capacity is low
- Excellent performance for admitted nodes

Dynamic Stream Rate Scaling

Configuration:
- The source tries to maintain 20% spare capacity
- The source can scale the stream rate in 5 packet steps

Results:
- Source scales down stream rate when capacity is low
- All nodes are able to join the overlay
- All nodes receive the stream

Dynamic Contribution Control

Configuration:
- 49 contributing nodes configured with 1Mbps in/out bandwidth
- Initial max. contribution level set to 2 children for all nodes
- When < 10% spare capacity, nodes probabilistically increase their max. contribution level if they can

Results:
- Big contribution skew
- All nodes are able to join the overlay
- All nodes successfully receive the stream