CliqueStream

Amir H. Payberah
amir@sics.se

What is CliqeStream?

- An Efficient and Fault-resilient Live Streaming Network on a Clustered Peer-to-peer Overlay.

Reminder

- Node discovery
- Data delivery

Finding supplying peers \ Data Delivery	Push method (Single tree)	Push method (Multiple trees)	Pull method	Push-Pull method
Centralized method	DirectStream (2006)			Prime (2007) mTreeBone (2007)
Hierarchical method	ZigZag (2003)			mTreeBone (2007)
DHT-based method	SAAR (2007)	SAAR (2007) SplitStream (2003)	SAAR (2007)	Pulsar (2007) mTreeBone (2007)
Controlled flooding method			GnuStream (2003)	
Gossip-based method		Orchard (2006) ChunkySpread (2006)	CoolStreaming (2005) PULSE (2006) ChainSaw (2005) PPLive (2004)	Bullet (2003)

Motivation

- In most of the current solutions, the node's neighbours are selected randomly.
- It is possible that distant nodes in the physical network selected as neighbours.
- Two main problems:
  - Data travels unnecessary distances before reaching the destination.
  - Two nodes of very close proximity may receive data through completely disjoint paths from the source.

Contribution

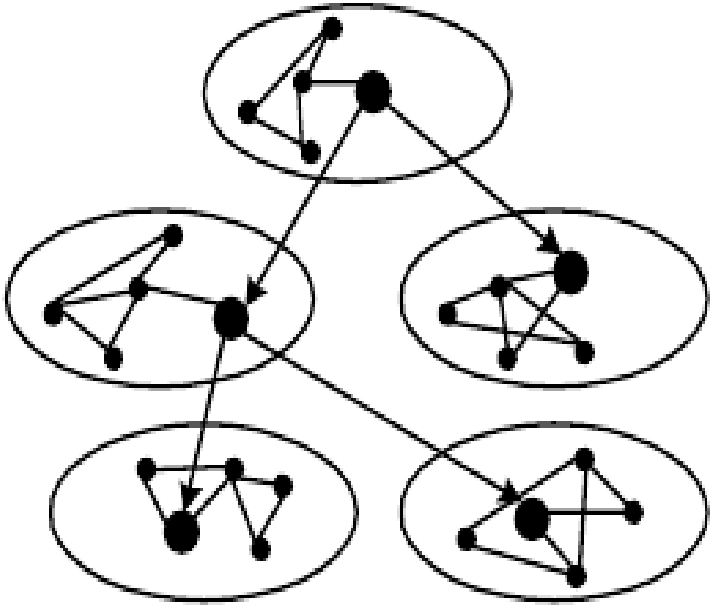
- Consider the proximity of peers to select neighbour set.

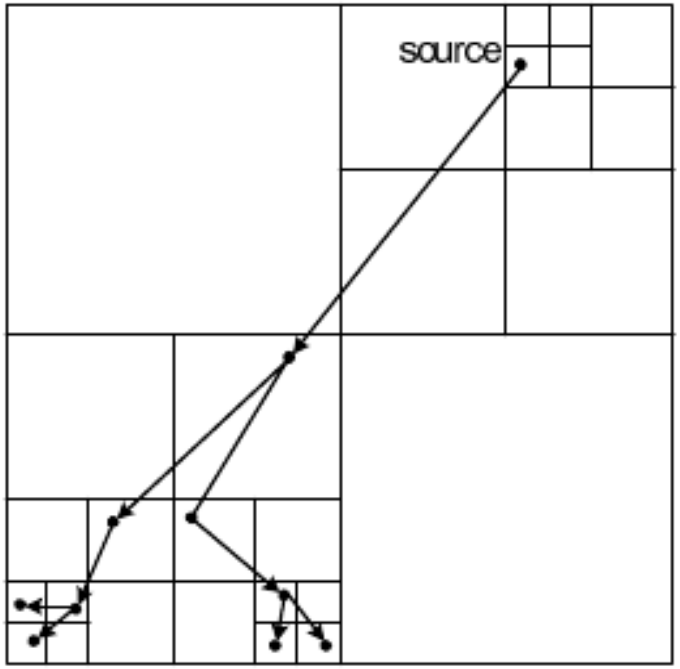
- Use eQuus to build media streaming overlays.

Core Idea

- Uses push-pull method for data delivery.
- The higher capacity and more stable nodes, called super nodes, are organized in tree structure to carry the content traffic.
- Less stable nodes create localized meshes around each super node and pull the content.

Core Idea





Definition

- eQuus
- Super node

eQuus

- It is a DHT that consists cluster of nodes, named clique, instead of individual nodes.
- IDs are assigned to each clique instead of nodes.
- The nodes in the same clique are close to each other based on proximity metrics, e.g. latency.

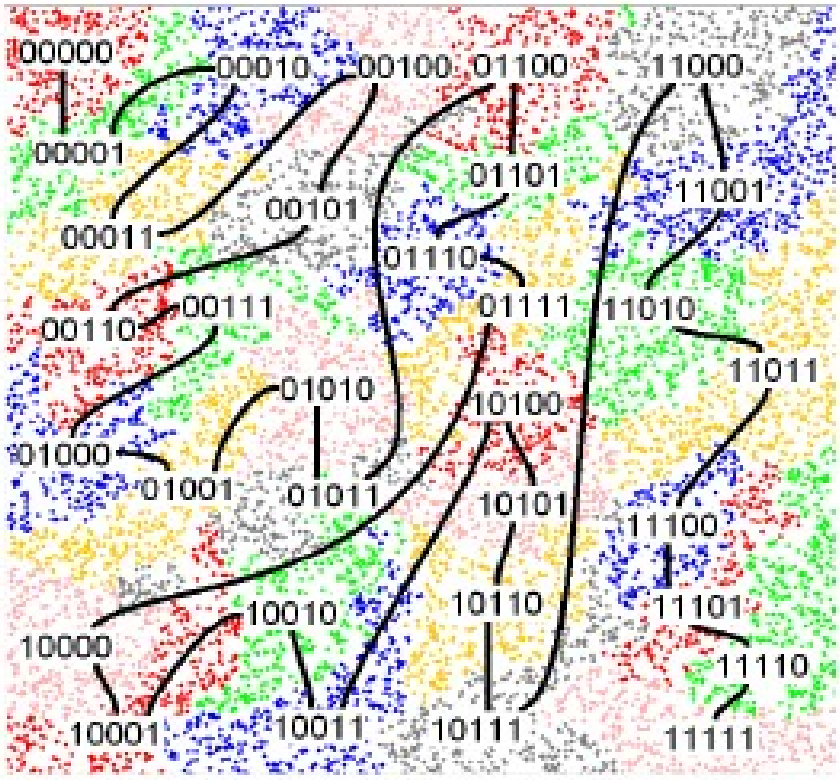
eQuus

- The distribution of IDs among cliques are not random.
- Two cliques with numerically close IDs are close to each other in the proximity space.

eQuus

- All nodes in one clique share the same routing table.
- The routing table is the same as in Pastry, but each entry represent an entire clique not a node, and for each clique ID, address of k random nodes of that particular clique is stored.

eQuus


00000
00010
00100
01100
11000
00001
01101
11001
00011
00101
01110
00110
00111
01111
11010
11011
01010
10100
01000
01001
01011
10101
11100
10110
11101
10010
10000
11110
10001
10011
10111
11111

Super Node

- Super nodes are more stable and more capacity nodes in each clique.
- Adding super nodes into eQuus changes the original routing.
  - In this construction, the stream is routed between two cliques through only one link.

Solution

- Each stable node maintains a channelList.
  - It maps the channel name to channelInfo.
- ChannelList includes all channels received or relayed by one of the node in clique.
- ChannelInfo stores the data to maintain the structure of tree.

Join Procedure

- A new node sends a join request to one of the stable nodes in its clique.
- Two cases:
  - The stable node has information of channel in its channelList.
  - It Does not have it.

If yes ...

- Super node forwards the request to the relaying node.
- The relaying stable node maintains a recipientList.
  - The nodes in the same clique that are receiving the channel.
- The relaying stable node adds the requesting node to the list and returns a random subset of the recipientList to the requesting node.
- Receiving the reply, the requesting node can now request those nodes for their current bufferMap download stream segments.
- In turn, those nodes also know the presence of the new node in recipientList and may include it in their partnerList.

Otherwise ...

- Super node sends a remote join request to the source.
- The source sends a message trough eQuus routing substrate.
- This message travels through nodes in several other cliques before reaching the joining clique.
- While travelling through the cliques, the streaming tree is created or extended.

Leave Procedure

- Non-stable node
  - Sends leave message to all its mesh neighbours.
  - The relay node updates the recipientList.
  - Other neighbours update their neighbourhood table.
- Stable node
  - It initiates a relay election protocol among the other stable nodes in the clique.
  - The stable node with highest available bandwidth is selected.
  - Then the leaving node initiates the handOver protocol to transfer the relaying role for the channel it was relaying.

Failure Recovery

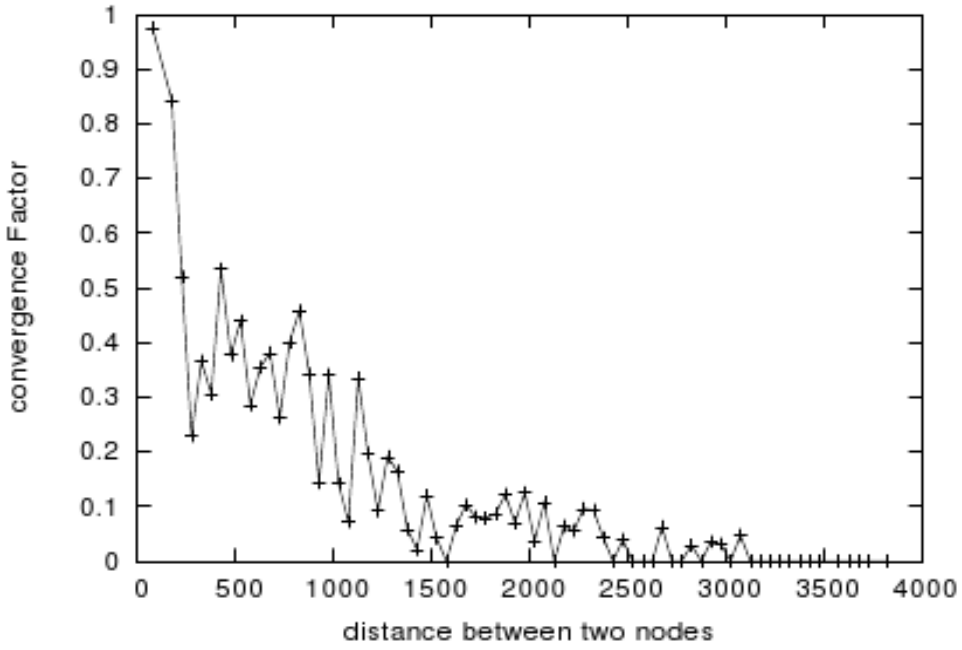
- Non-stable node
  - It is detected by its mesh neighbours.
- Stable node
  - The children of stable node in dissemination tree or its backup node detects its failure.
  - The backup node retains a replica of the channelInfo.
  - A handOver message is sent to the parent.
  - The failure is recovered completely locally.

Evaluation

- Two set of experiments:
  - The commonality of two paths.
  - The property of trees created over eQuus is compared to other type of trees.

First Set of Experiments

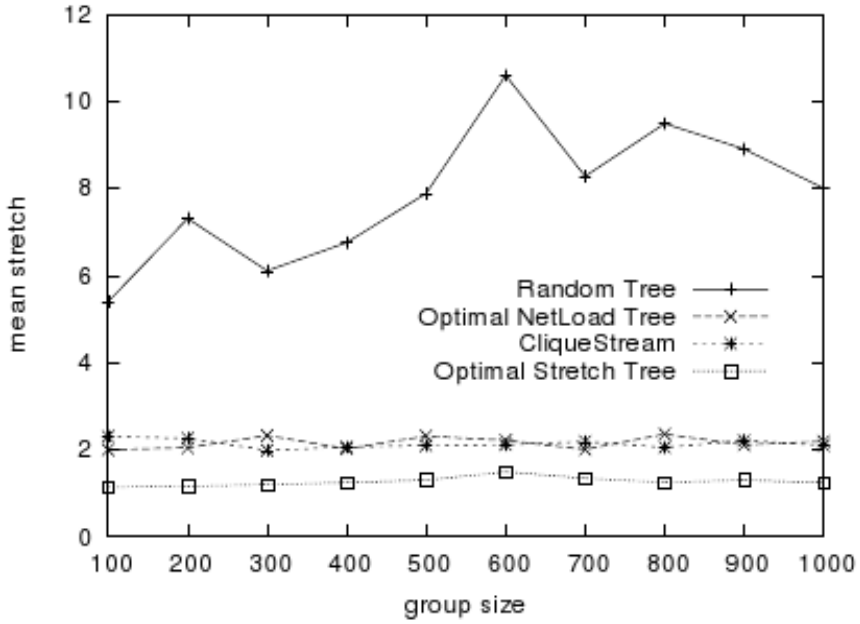
- Convergence metric:
  - The fraction of path that is common in both routing path.
  - It will be 0 for two completely disjoint node and 1 for completely shared.

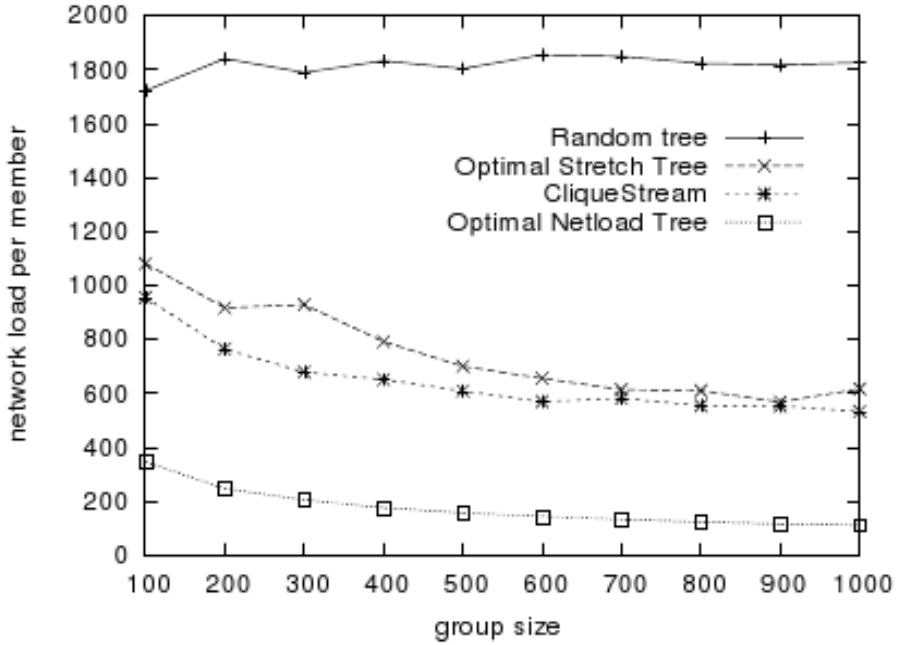


Second Set of Experiments

- Three type of trees:
  - Random tree
  - Optimal netload tree
    - It is constructed by connecting each new node to the node that has shortest distance from new node.
  - Optimal stretch tree
    - It is constructed by connecting new node as close as root.

Second Set of Experiments





Conclusion

- Features of a clustered distributed hash table overlay.
- Good locality properties such as low stretch and low communication.
- Localized failure recovery mechanism.
- Backup relay nodes are used for fast recovery.