Design and Evaluation of a New and Effective Fairness Scheme for Multicasting in Internet-Scale Distributed Systems

Yijun Lu and Hong Jiang
Department of Computer Science and Engineering, University of Nebraska-Lincoln
{yijlu, jiang}@cse.unl.edu

What do we mean by *Fair Overlay Multicasting*

- *Overlay* multicasting vs. IP mutlicasting
- Multicasting *forest* vs. multicasting tree
- Each node contributes outgoing bandwidth *proportional* to its total capacity

Why do we need to enforce proportional contributions?

- To support multiple multicasting sessions.

An example:

- Assumption: Two nodes, *A* and *B*, are both going to multicast a movie and each multicast will span the whole network.
- Scenario 1 (only satisfy a node's network constraint)
  Another node *C* may contribute 90% of its outoing bandwidth for *A*'s multicasting session. Thus it can not support the multicasting session for *B*.
- Scenario 2 (enfoce proportional contribution)
  In this case, suppose *C* only contributes 20% of its outgoing bandwidth for *A*'s multicasting session. Thus it has enough bandwidth to support *B*'s multicasting session.

The benifit:

- Support multiple multicasting sessions, thus provide a fair sharing environment, for performance sake.
  (1) It makes feasible to support multiple sessions when feasibility is required, such as in critical data delievery in data Grid.
  (2) It imporves QoS, in terms of low packet dropping rate, when feasibility is not required such as in real-time multimedia streaming, as it induces less hot spot.

Is it possible to do that?

- This is only necessary when there is excess outgoing bandwidth exist.
- We believe that it will be the case, especailly consider the proliferation of high-speed wireless connections.

Design of FairOM (Fair Overlay Multicast)

(1) Basic Idea

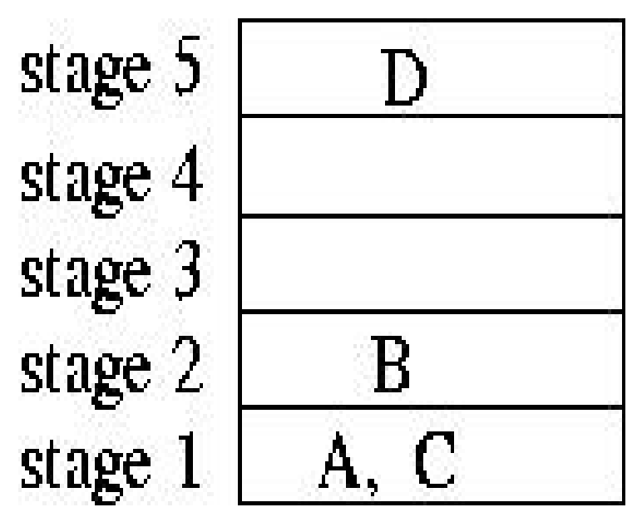
- Build a multicast forest in two phases.
- The first phase:
  use established neighborhood information to build an initial mutlicast forest that may not be complete.
- The second phase:
  a node contacts the source to ask for any missing stripes and finally makes the forest complete.
- One principle: enforcing proportional contribution using "staged spare capcity group".

(2) Establishment of neighborhood

- Each node has at least one bootstrap neighbor.
- Check its neighbor's neighbor list, if the neighbor's neighbor does not appear in this node's neighbor list, act as follows:
  * if its neighbor list is not full, put the neighbor's neighbor in.
  * Othewise, if the routing latency between this new node and itself is smaller than those between the current neighbors and itself, put the node in with probility 0.8 (current setting) to prevent hot spot.
- Run this procedure several round to establish neighborhood

(3) Staged spare capacity group

- Suppose *A* has a total outgoing bandwidth of 20 (it can forward 20 stripes of data).
- *A* has contributed 3 stripes, then its current contribution is 15% (3/20).
- *B* is put into stage 2 as its contribution is in the range (20%, 40%].
- *C* and *D* are following the same creteria. Here, the range of each stage is 20%.



(4) Initial forest construction

- Purpose: provide a skeleton on which the second phase can improve. It is by no means to build a complete forest.
- Procedure:
  (1) The source sends all stipes out and they are fowarded to different neighbors to achieve path diversity.
  (2) For each peer that receives a stripe, it forwards the stripe to as many neighbors as it can within the predefined quota.
  (3) If a peer receives multiple transmissions of the same stripe, it picks one and rejects others. For example, it can alwasy picks the parent that notices it first.
- Then, a multicast relationship between a parent and child can be established.

(5) Making the forest complete

- After the forest constrcution process starts, each node checks with those nodes that treat it as a neighbor.
- If all nodes it contacts have already gotten some stripes and did not choose it as a child in the initial forest construction, it will seek help from the source.
  Moreover, it contacts the source anyway if a deadline has passed.

- In the message it sends to the source, the node indicates the number of times it has requested for spare capacity.
- Based on the number of tries, the source only looks for adoption for this node up to a certain level of the "staged spare capacity group".
  For exampke, for a first time try, the source only looks at stage 1. And for a second time try, the source looks at both stage 1 and 2 (up to stage 2).

- In each round of try, if a parent is found, the parent will receive the adiption request of a potential child from the source, then parent will then send a request to the ptotential child which needs to be adopted. A multicasting relationship thus can be established between the parent and child.
- If the source cannot find a parent in this round, the node with missing stripes waits a predefined period of time before it starts the next round of request.

- By following this protocol, the source relaxes the quota gradually and finally builds a complete forest in which every peer is in all trees.

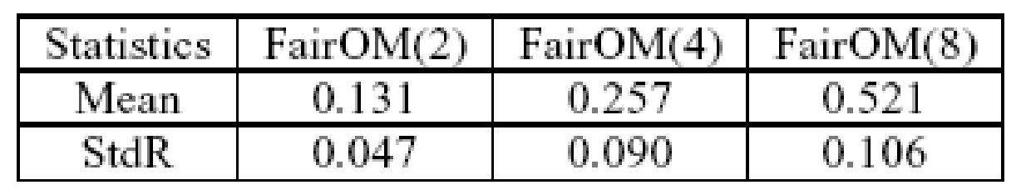
(6) Considering delay information

- Modify step 3 in the initial forest construction process.
- What do we mean by *delay*?
  The time difference between when the source starts multicasting a packet and this node receives the packet.
- When a node receives mutliple transmissions of the same stripe, it picks the one with the smallest dealy and drop others.
- Because the dropping process is based on delay, it will not create cycles.

Evaluation I: Effectiveness of enforcing proportional contribution

- Use *std* (standard deviation) as the measurement
- We run simulation with three configurations with number of stripes of 2, 4, and 8.
- In all the simulations, the algorithm satisfies the requirement to build a complete forest and satisfies all nodes' bandwidth constraint.
- Mean value and *StdR* are shown as follows. It shows that FairOM's effectiveness in this aspect.

Mean and Std of contribution ratios



Statistics	FairOM(2)	FairOM(4)	FairOM(8)
Mean	0.131	0.257	0.521
StdR	0.047	0.090	0.106

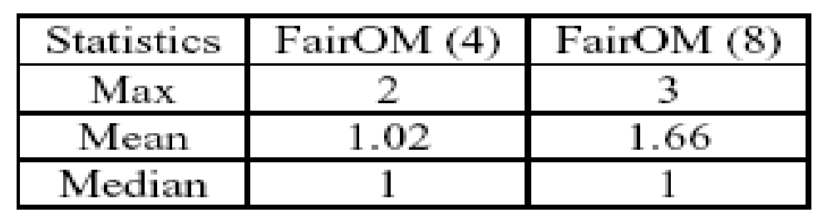
Evaluation II: Path diversity

- What is path diversity?
  It refers to the diversity between the paths from each node to the multicast source. Ideally, the paths should be disjoint with each other so that one node's failure only causes the loss of one stripe for the receiver.

- FairOM uses randomization and enforced delay between quota relaxation requests to achieve it.

- In this simulation, we randomly fail one node, and the result is summaried in the following table. It indicates that FairOM achieves path diversity.

Maximum, mean and median number of stripes lost when a single node fails



Statistics	FairOM (4)	FairOM (8)
Max	2	3
Mean	1.02	1.66
Median	1	1

Discussion

- Discrete stage vs. continuous stage?
- Distributed algorithm for forest construction?
- FairOM assume the trustworthy among participants. How can we relax this requirement?

Future work

- Deploy the prototype on Planet-Lab
- Explore techniques to solve the trustworthy problem.

References

[1] M. Castro, P. Druschel, A.-M. Kermarrec. et. al. SplitStream: High-bandwidth multicast in cooperate environment. *In Proc. of the SOSP*, Bolton Landing, New York, USA, October 2003.