Epidemic Multicast

(Mike) Yu Cheng

Outline

- Definition, properties of epidemic multicast
- Why is it so special? Advantages over other algorithms
- Take a look at some proposals

Epidemic?

- Outbreak of a contagious disease which spreads rapidly and widely
- SIR/SEIR (Susceptible, Exposed, Infected, Recovered)
- Example: Flu

Epidemic (gossip) Algorithms

- Inspired by the theory of epidemics of infectious disease
- “Infect” as many nodes as possible
- In networking: send the message to as many host as possible
- Have been applied to many problems, e.g., multicast in ad hoc networks, reliable recovery, etc.

Basic concepts of gossip algorithms

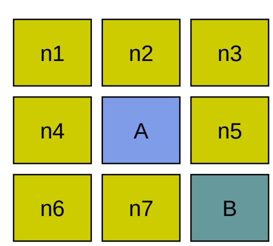
- Nodes/Hosts periodically compare their states and reconcile inconsistencies with others
- Randomly decide when and with whom each participant will gossip
- States can be missing messages or packets, etc.

Basic Concept: A Gossip Round

- A chooses randomly another host to gossip with, say B
- A sends to B information about the messages A has received or missed
- A and B will reconcile the states by exchanging messages

A has messages 1, 2, 4

B has messages 1, 2, 3, 5

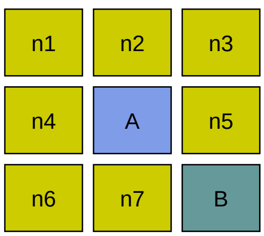


Basic Concept: Different Ways of Communication

- Pulling/ negative
- Pushing/ possible

Example: Current state: A has messages 1, 2, 4, 5

Pushing: " I have messages 1, 2, 4, 5, I will give them to you"



Pulling: " I need message 3! I will take it from you"

Basic Concept: Different Ways of Communication (continue…)

- For Pulling strategy, gossip is triggered only when a host realizes it has lost a message
- For Pushing, gossip must take place periodically.
- A mixed approach is possible

Advantage of using gossip algorithms

- Decentralized. When a node fails, the system will not fail
- Probabilistic. Randomly picking a node to gossip
- Every nodes/hosts share the same amount of load
- Simple to implement
- Scalable

Flat gossiping multicast

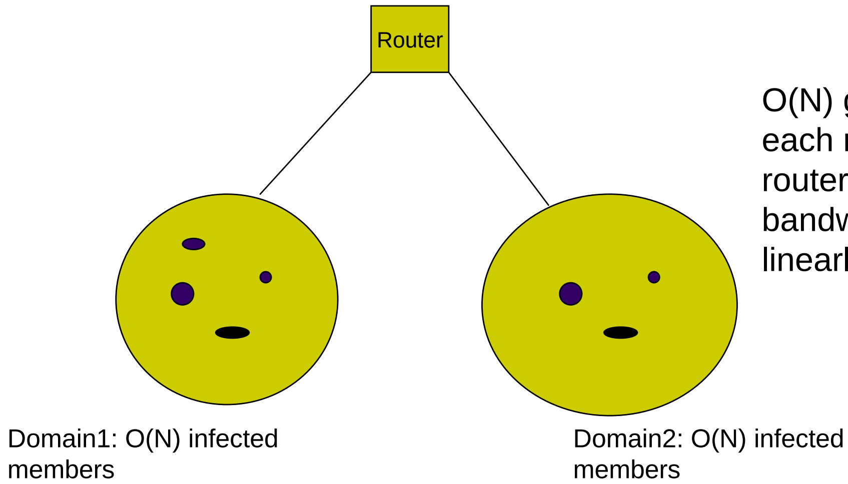
- There are N number of hosts/nodes
- Each group member that receives the multicast gossips about it for log(N) rounds
- During each round, the group member select B other members uniformly at random (flatly) and send them a copy of the multicast message
- A round can be a fixed interval at the member. B can also a be constant

Problem with flat gossiping

- Large networks, for example, wide area networks, or corporate VPN that spans several locations, are structured as a hierarchy of domains.
- Internet is a perfect example as well (AS, class networks, subnets, etc.)
- Flat gossiping generates substantial network traffic into and out of these domains.
- Creates significant network overhead on connection core routers, bridges, and links.

Example:

- In each gossip round:

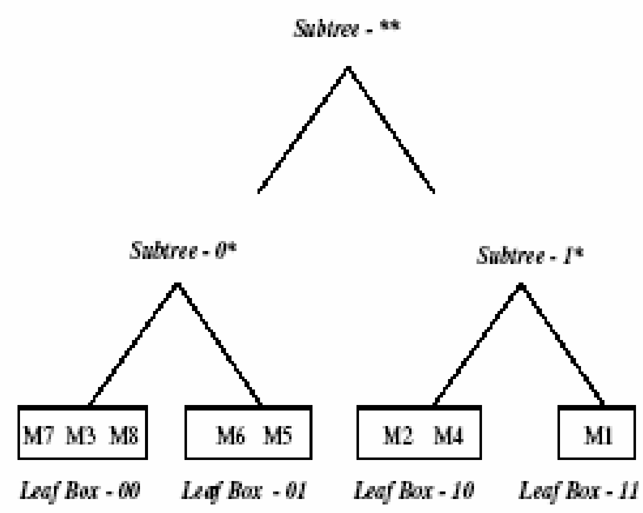


O(N) gossip messages in each round go across the router. Therefore the router bandwidth usage grows linearly with group size

Leaf Box Hierarchy

- Assume group size = N
- The number of leaf box = N/K (K is a constant)
- There is a map function H that maps members to one of N/K leaf boxes
- Each leaf box has a $\text{Log}_k N$ -1 digit address
- Subtrees of height j (0<= j <= $\text{Log}_k N$ -1)
- Leaf boxes whose address match in the most significant ($\text{Log}_k N$ -1-j) digits

Example:

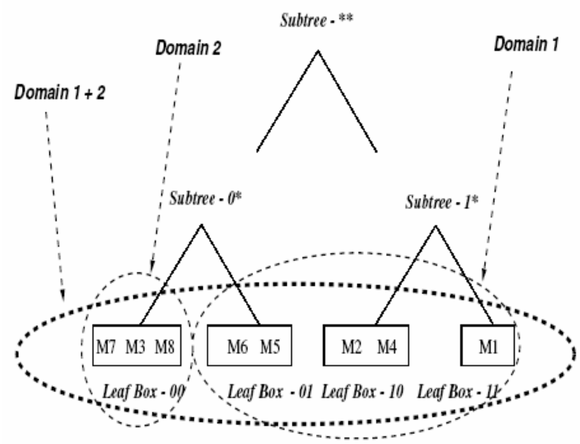


- K = 2
- N = 8
- Each box has (3-1) = 2 digit address
- Subtrees of height (0<= j <=2)

Picture from [1]

Example (Mapping function):

- Contiguous Mapping: Assigns to each network domain a set of leaf boxes that are contiguous in the lexicographic space of leaf box addresses.

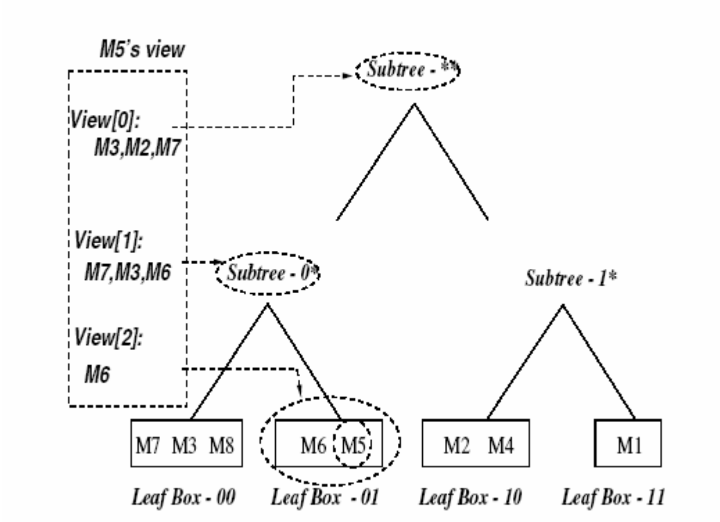


Picture from [1]

Views of a member

- Each member M_i maintains a view that consists of $\log_k N$ subviews.
- There are viewFactor*$\log_k N$ members in each subviews
- Members in the subviews are chosen randomly

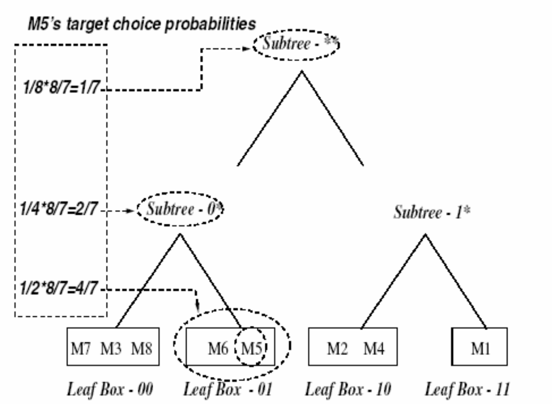
Example (views)



- viewFactor = 1
- K = 2
- N = 8
- # of subviews = 3
- View member = 3 *1 = 3

Picture from [1]

Example (target choice probabilities):



- A subtree of height l has probability of picking:

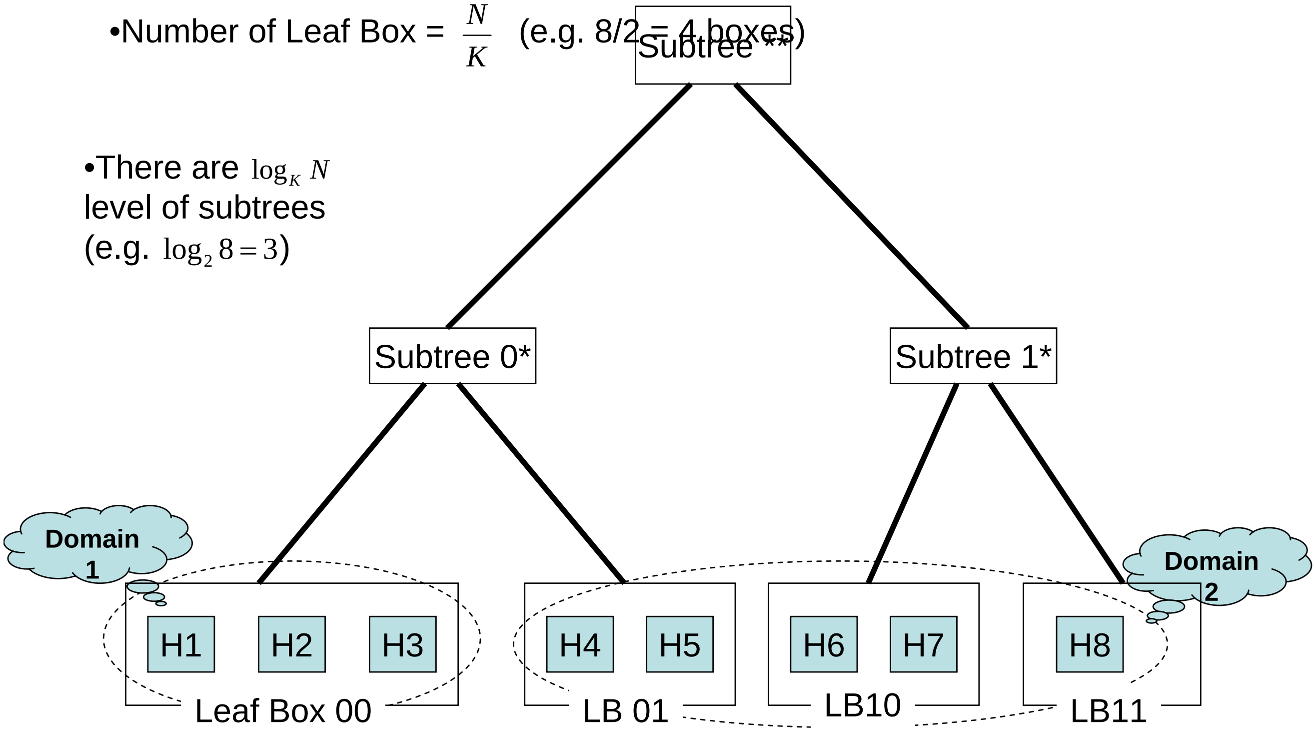
$$p_l = \frac{1}{K^{l+1}} * p(N,K)$$
$$p(N,K) = \left[\sum_{j=0}^{\log_K(N)-1} \frac{1}{K^{l+1}}\right]^{-1}$$

- Prefer target near the leaf box hierarchy

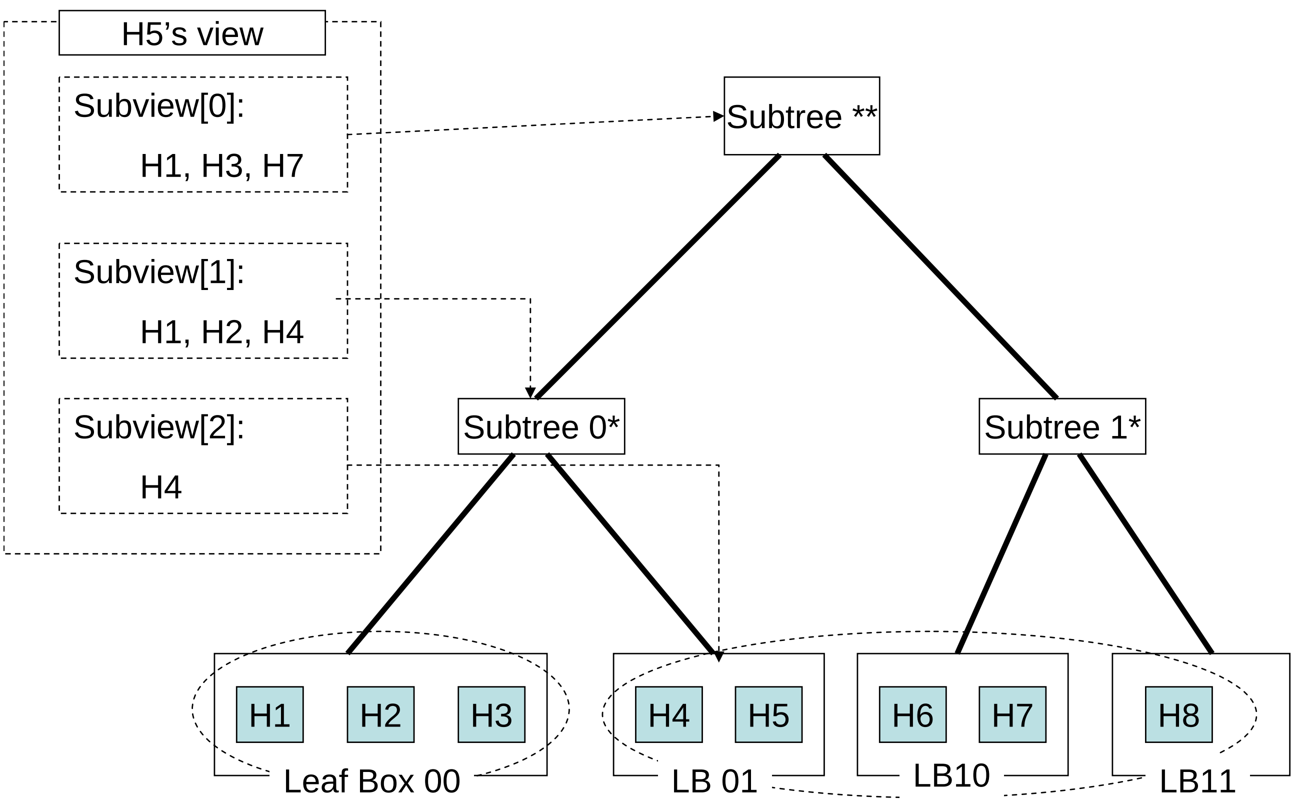
Picture from [1]

- Constant K (e.g. 2)
- Group Size N (e.g. 8)
- Number of Leaf Box = $\frac{N}{K}$ (e.g. 8/2 = 4 boxes)

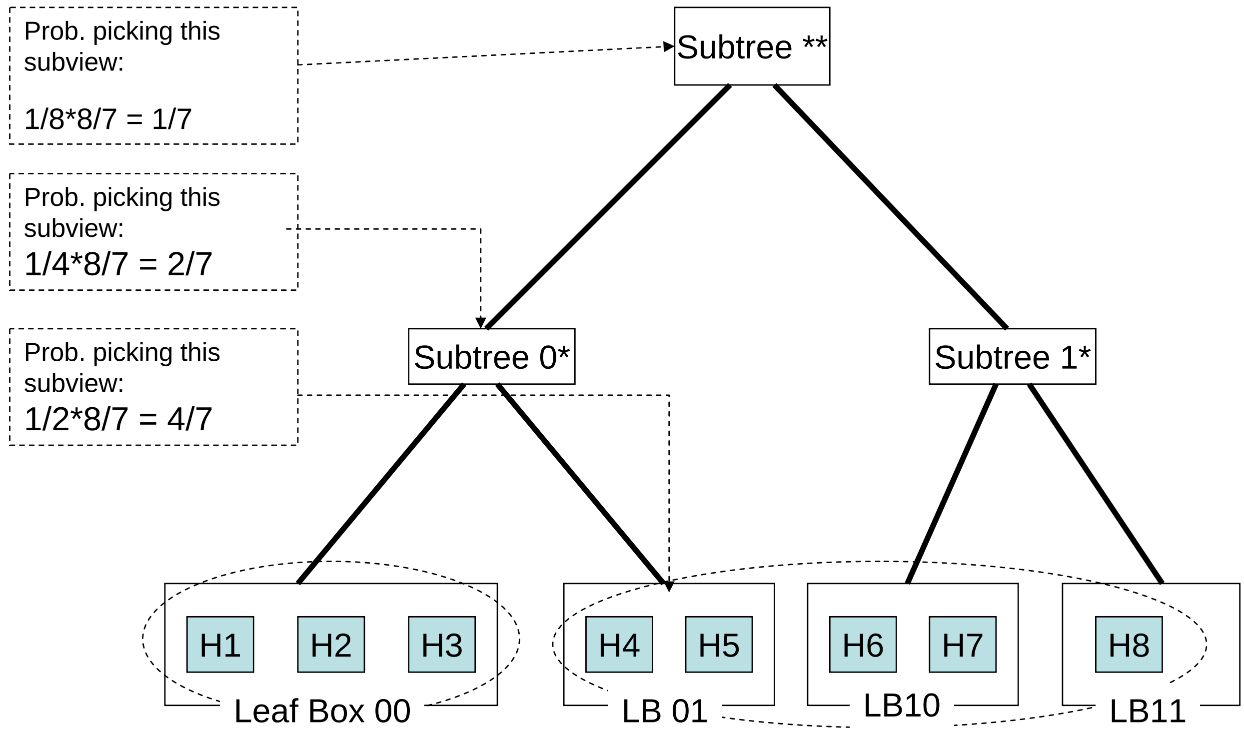
- There are $\log_K N$ level of subtrees (e.g. $\log_2 8 = 3$)



- Each host has a view (member list)
- There are $\log_K N$ subviews (e.g. $\log_2 8 = 3$)
- Number of hosts in each subview = viewFactor * $\log_K N$ (e.g. $1 * \log_2 8 = 3$)

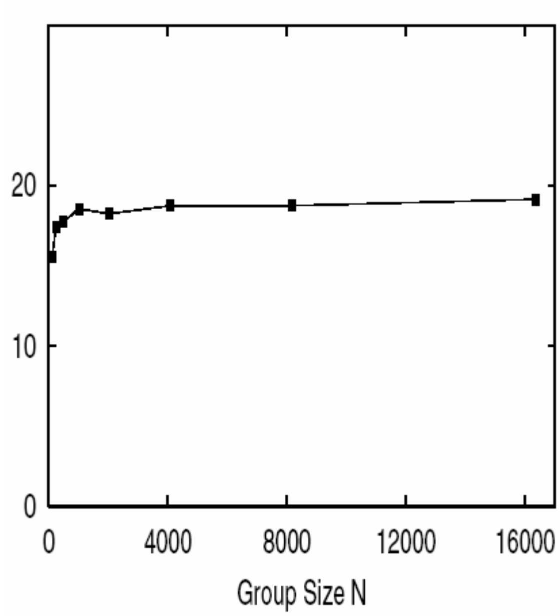


- Select a subview with probability $P = \frac{1}{K^{l+1}} * p(N,K)$ $p(N,K) = \left(\sum_{j=0}^{\log_k(N)-1} \frac{1}{K^{j+1}}\right)^{-1}$
- Pick a host uniformly at random from that subview



Result:

Avg. Domain boundary load/Gossip per round

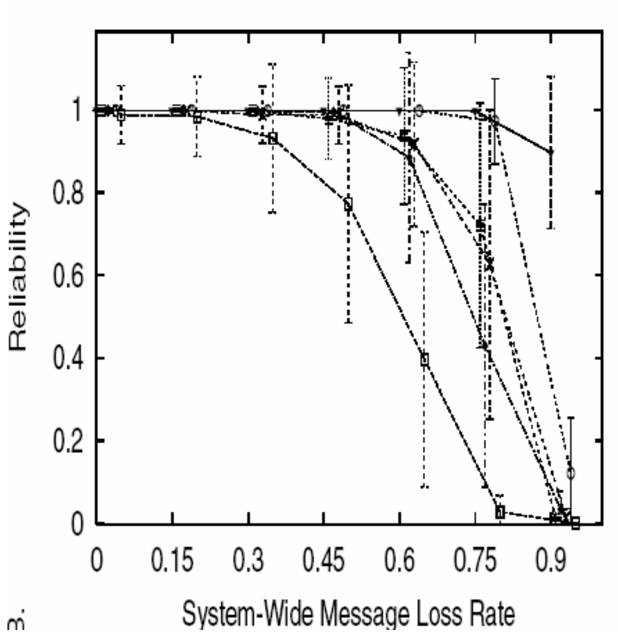


Picture from [1]

Result:

•Flat: Dropping starts at 0.75

•LBH: Dropping starts at 0.3



Picture from [1]

Reference:

- [1] Indranil Gupta Efficient Epidemic-style Protocols for Reliable and Scalable Multicast IEEE Symposium on Reliable Distributed Systems 2002
- [2] Werner Vogels, Ken , Birman, et. al. Using Epidemic Techniques for building Ultra-Scalable Reliable Communication System