Towards Deployable Large Scale End-point-based Multicast Streaming

György Dán, Viktória Fodor
Ilias Chatzidrossos and Gunnar Karlsson

Laboratory for Communication Networks
School of Electrical Engineering
KTH, Royal Institute of Technology
Stockholm, Sweden

Why end-point-based multicast?

- IP level multicast
  - Not widely available
- Content delivery networks
  - Cost increases with number of spectators
  - Difficult to handle sudden traffic surges
  - Dedicated infrastructure required
- End-point-based multicast
  - Peer-to-peer approach
  - Share the costs among the spectators
    - Bandwidth
    - Processing power

Pros and Issues

Pros

- Easy deployment
  - No infrastructure needed in the network
- Low cost per viewer for content provider
  - Viewers forward the content to others
- Scalability
  - Can adapt to variations of user population size

Issues

- Incentives
  - Nodes use their resources to allow others to join
- Data plane
  - Loss propagation
    - Network failure
    - Group dynamics
- Control plane
  - Scalability of overlay construction
    - Group dynamics

- Why are such systems not deployed?
  - Predictability Controllability
    - System performance evaluation

End-point-based overlays

- Control plane
  - Organize nodes into an overlay
    - Handle high join and departure rates
    - Low overhead
    - Scalable
  - Centralized
    - CoopNet, ALMI, ESM
  - Structured p2p
    - SplitStream
  - Unstructured p2p

- Data plane
  - Distribute data among nodes
    - Robustness
    - Efficiency
  - Mesh based
    - TMesh, ScatterCast
  - Tree based
    - Yoid, ALMI, OverCast, SRMS, ESM
  - Multiple tree based
    - SplitStream, CoopNet

- Robustness, efficiency, relatively low delay and scalability at the same time
  - Multiple data paths from the root to nodes
    - Multiple distribution trees
  - Regeneration of data in nodes
    - Block based FEC ($\rightarrow$ PET, MDC)
    - High probability of packet possession

System description

Overlay

- # of distribution trees: t
- FEC(n,k) for error control
  - Lost packets can be reconstructed

Root node

- # of child nodes/tree: *m* (C_{link}/C_{stream})
- Sends packets in round-robin in the trees
- Sets *k* and *n* based on some policy

***N* peer nodes**

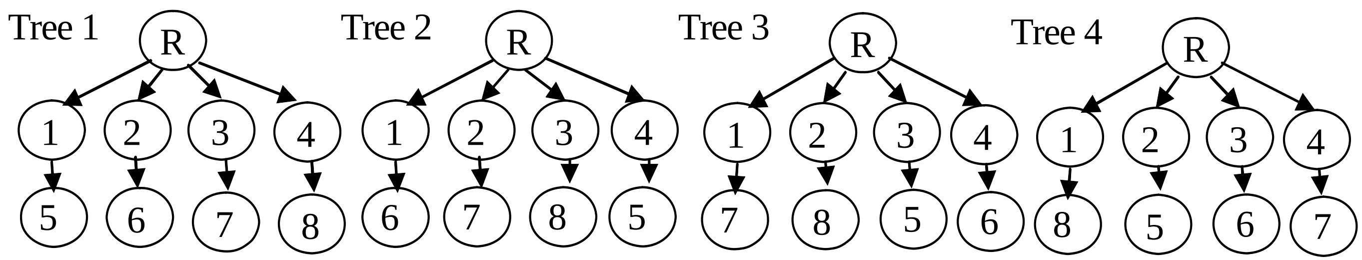
- Output bandwidth = input bandwidth
  - *t* children
- Forward data in *d* trees (fertile)
- Do not forward data in t-d trees (sterile)
- Have a different parent in each tree
- Reconstruct lost packets if possible

- # of layers: O(logN) if d<t
- Arriving nodes are handled centrally or in a distributed way

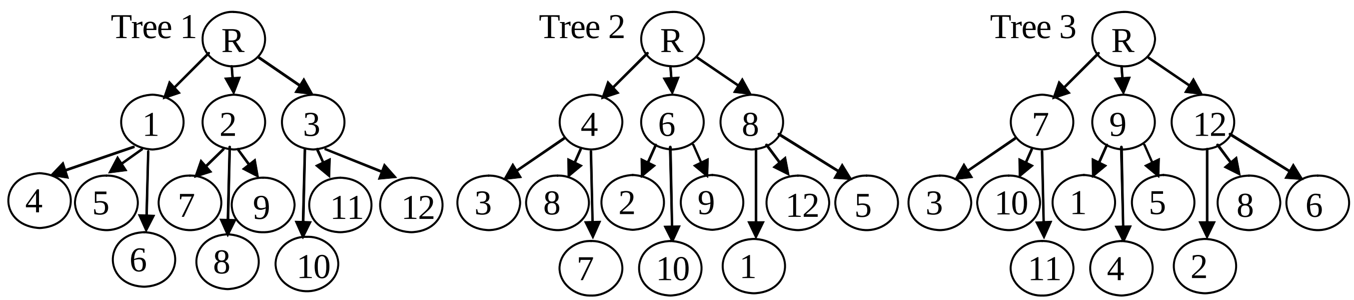
- Examples:
  - Case *d=1*: was considered in SplitStream and CoopNet
  - Case *d=t*: was considered in CoopNet
  - Case 1<d<t was not considered before

Some examples

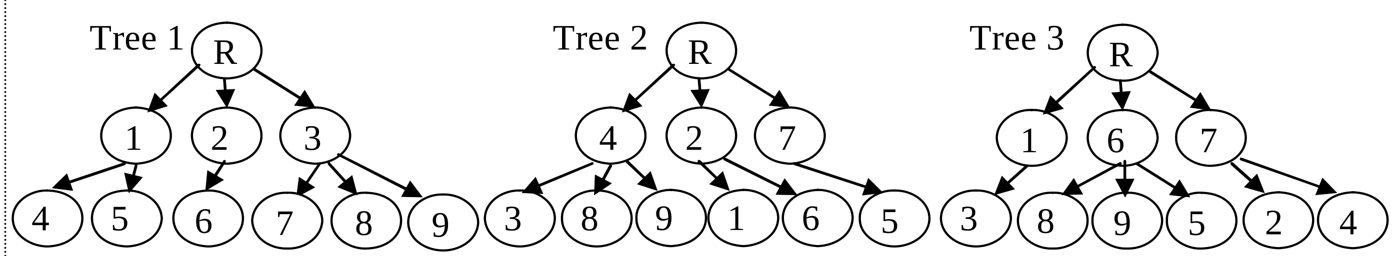
- d=t:
  - N=8
  - t=4
  - m=4
  - L=2



- d=1:
  - N=12
  - t=3
  - m=3
  - L=2



- d=2 (1<d<t):
  - N=9
  - t=3
  - m=3
  - L=2



System performance

Analytical model to understand the system's behavior

Sources of impairment

- Network failures
  - Packet losses with probability p between peer nodes
    - Loss propagation
- Group dynamics
  - Interruption of data flow – packet loss
    - Loss propagation
  - Overlay maintenance

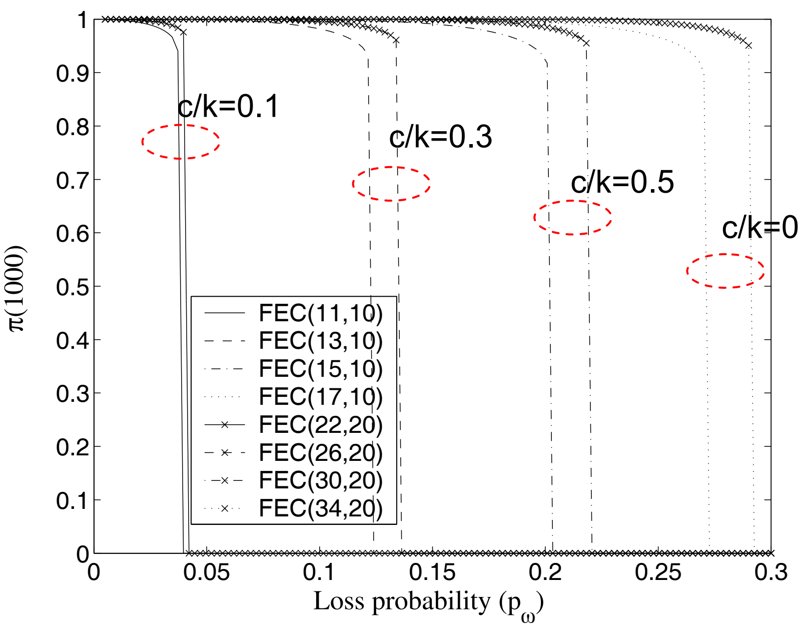
Performance measures:

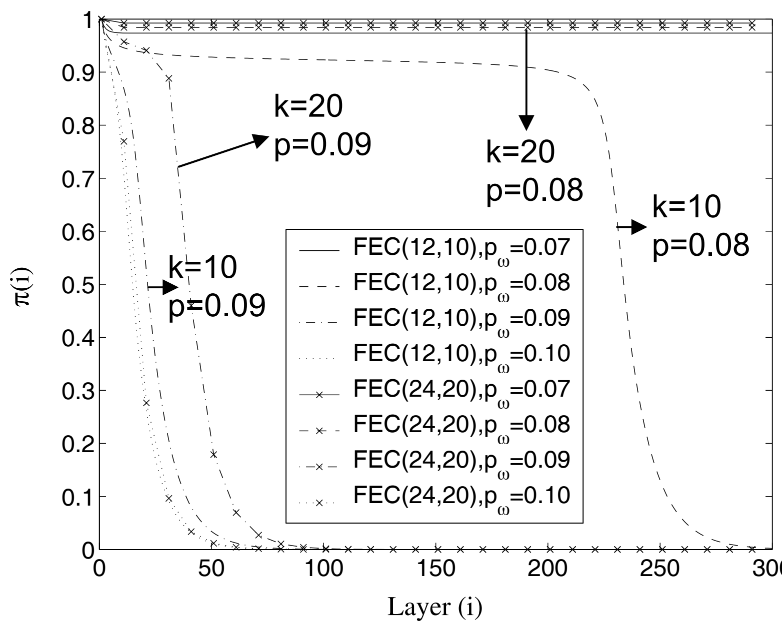
- Probability of packet possession: $\pi(i)$
- Probability of blocking:
  - Arriving node cannot join the overlay due to lack of resources
- Probability of reconnection failure:
  - Node in the overlay cannot reconnect to the overlay after departure of another node

Mathematical model (d=t)-static

- m≥t (different parent in each tree)
- Initial condition: $\pi(0) = 1$
- Recurrence equation for $\pi(i)$

$$\pi(i+1) = R(\pi(i), p) = \pi(i)(1-p) + \sum_{j=k}^{n-1} \binom{n-1}{j} (\pi(i)(1-p))^j ((1-\pi(i))(1-p))^{n-j}$$





π(i) high if *$p<p_{max}(n,k)$ - non-graceful degradation for $p>p_{max}(n,k)$*

Mathematical model (d=t)-static

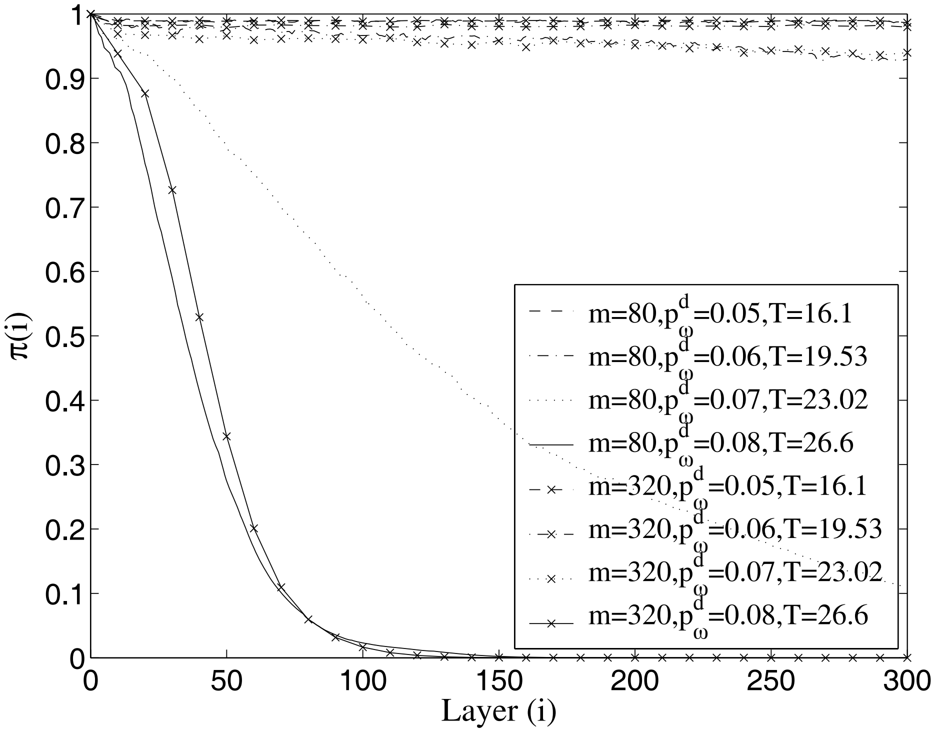
- Correlated losses
  - Output link
    - Does not affect the performance while n<t
    - Node departures can be thought of as bursty losses at the output link: dynamic case ~ static model
  - Input link
    - Can be modeled (e.g. using Gilbert model)
    - Correlations decrease the value of p_{max}
- Non-homogeneous losses (Distribution of losses: Q)

$$\pi(i+1) = \int R(\pi(i), p)\,dQ$$

  - Decreases performance depending on the variance of Q
- Malicious layers (e.g. DDOS)
  - High loss experienced in a particular layer
    - Recovery from losses in the lower layers

Mathematical model (d=t)-dynamic

- Arrival process: Poisson (λ)
- Holding time distribution: Log-normal (mean 1/μ)
- # of departing nodes per time unit: N_d
- Mean time to find new parent: T
  - Modeled by switching off nodes
- Packet loss due to departures: $p_{\omega}^{d} = \frac{N_d}{N} T$
- For high *m* the approximation is accurate
  - Number of active nodes per layer υ follows binomial (m, μ/(1+ μT)) distribution (Engset system)
  - Coefficient of variation: $CoV(\nu) \sim m^{-0.5}$
- Non-graceful degradation as in the static case
- Main drawback of the overlay:
  - *#of layers O(N)*
  - *High delay*

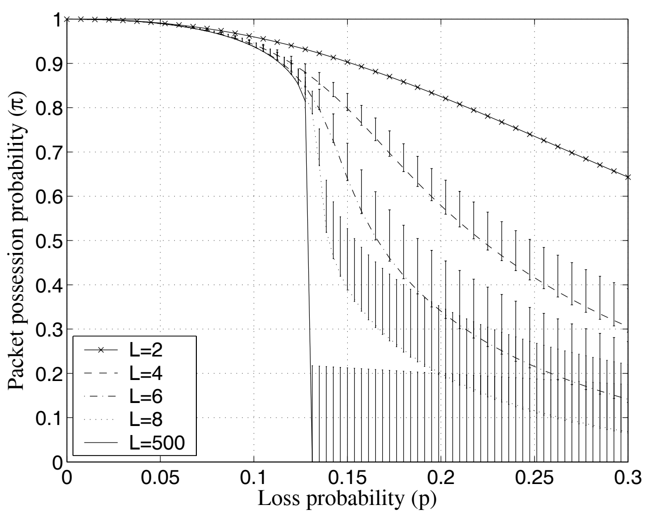


Mathematical model (d=1)-static

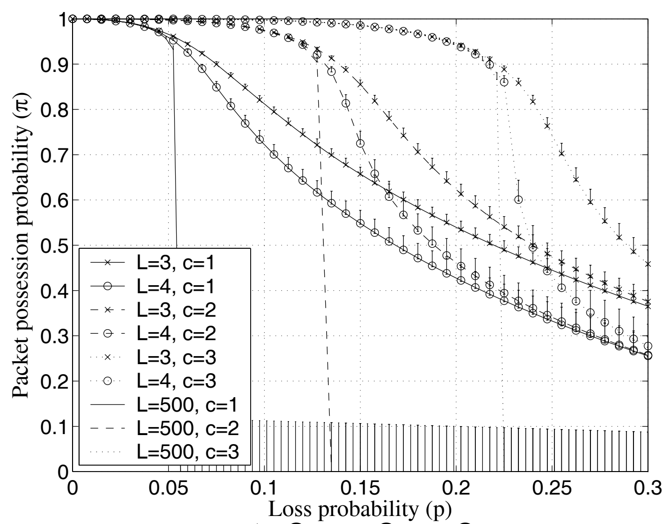
- m≥t-1 for feasibility
- Recurrence equation for: $\pi_f(i)$
  - Probability of packet possession in fertile tree

$$\pi_f(i+1) = (1-p)\pi_f(i) + (1-(1-p)\pi_f(i))\sum_{j=k}^{n-1}\binom{n-1}{j}((1-p)\pi_f(L-1))^j(1-(1-p)\pi_f(L-1))^{n-1-j}$$

- For $\pi(i)$: $$\pi(i+1) = \frac{1}{n}(1-p)\pi_f(i)\sum_{j=1}^{n}\tau(j)\binom{n-1}{j-1}((1-p)\pi_f(L-1)^{j-1}(1-(1-p)\pi_f(L-1))^{n-1-(j-1)}$$
$$+\frac{1}{n}(1-(1-p)\pi_f(i))\sum_{j=0}^{n-1}\tau(j)\binom{n-1}{j}((1-p)\pi_f(L-1)^j(1-(1-p)\pi_f(L-1))^{n-1-j}$$



t=4,m=4,n=4,c=1

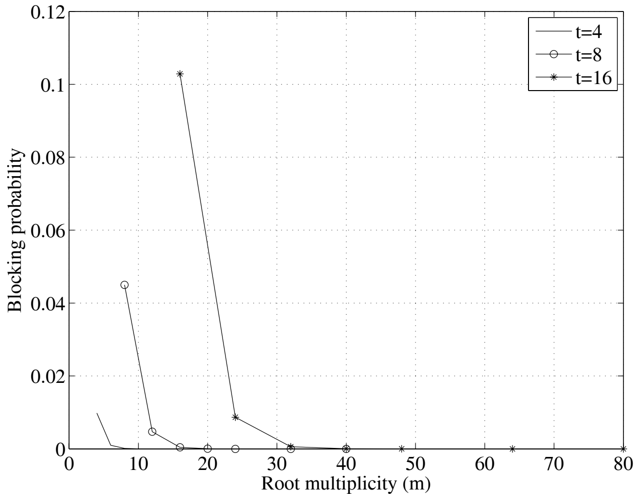


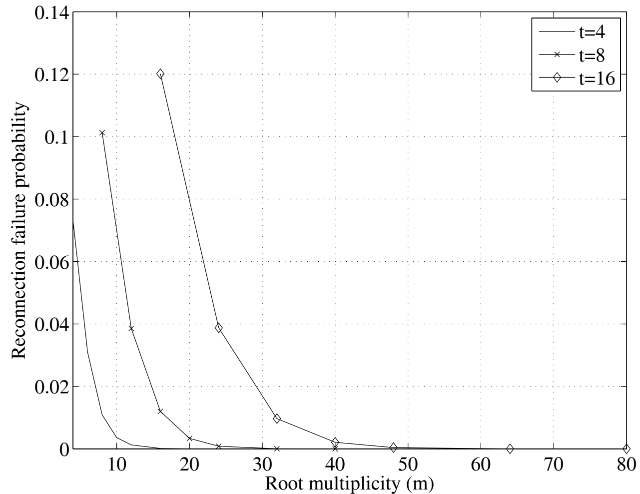
t=8,m=8,n=8

- where $\tau(j) = \begin{cases} j & \text{if } j < k \\ n & \text{if } j \ge k \end{cases}$
- $\pi(i)$ high if $p<p_{max}(n,k)$ (like for d=t)
- $\pi \to 0$ if $p>p_{max}(n,k)$
- Non-graceful degradation if L high

Mathematical model (d=1)-dynamic

- Arrival process: Poisson (λ)
- Holding time distribution: Log-normal (mean *1/μ*)
- Number of fertile nodes per tree can become unbalanced due to departures, and has to be handled by
  - Intervention: reallocation of fertile nodes – *problematic if λ,μ are high*
  - **Failed reconnections & blocking**: retry after τ seconds in hope that balance will be restored by arrivals and departures - *scalable*
- Probability of blocking and failed reconnections (approximate Markovian model of spare capacity in the trees)





- Blocking and reconnection failure
  - High if m~t
  - Decrease as N increases
  - Decrease as τ increases

Generalized overlay (1<d<t)-static

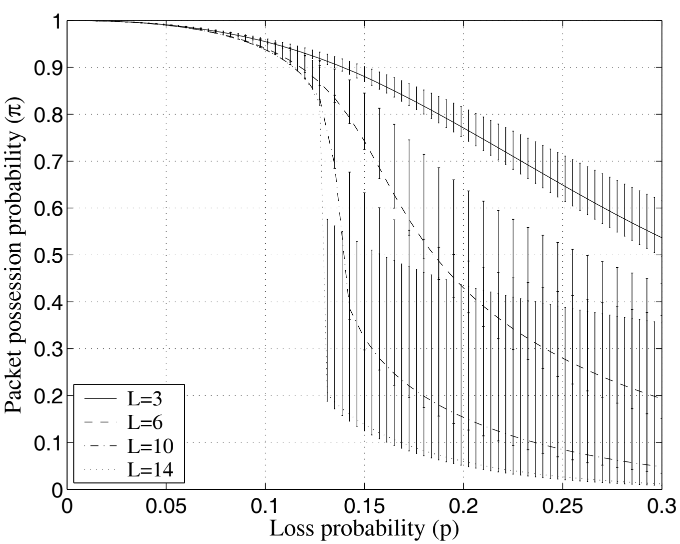
- Feasible for m<t-1
- Recurrence equation for: $\pi_f(i)$
  - Probability of packet possession in fertile tree $\pi_{fa}(i+1) = (1-p)\pi_f(i)$

$$\pi_f(i+1) = \pi_{fa}(i+1) + (1-\pi_{fa}(i+1))$$

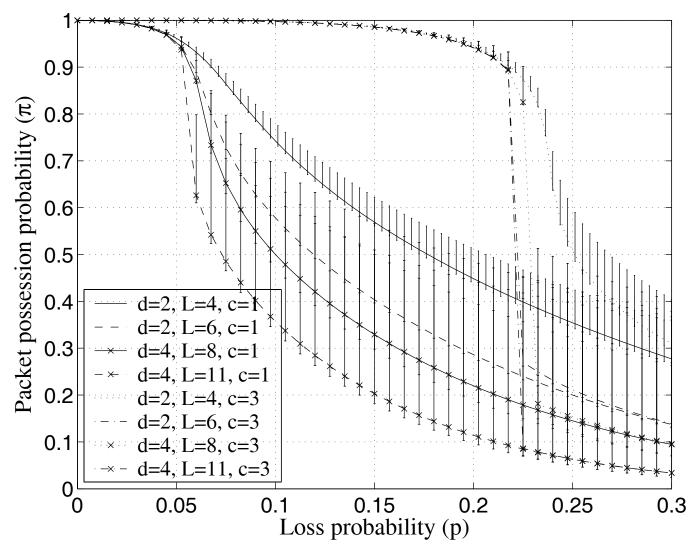
$$\sum_{j=k}^{n-1} \sum_{z=\max(0,j-n+d)}^{\min(j,d-1)} \binom{d-1}{z} \pi_{fa}(i+1)^z (1-\pi_{fa}(i+1))^{d-1-z} \binom{n-d}{j-z} \pi_{fa}(L)^{j-z} (1-\pi_{fa}(L))^{n-d-j+z}$$

- For $\pi(i)$:

$$\pi(i+1) = \frac{1}{n} \sum_{j=0}^{n-d} \sum_{z=0}^{d} \tau(j+z) \binom{d}{z} \pi_{fa}(i+1)^z (1-\pi_{fa}(i+1))^{d-z} \binom{n-d}{j} \pi_{fa}(L)^j (1-\pi_{fa}(L))^{n-d-j}$$



t=4,m=4,n=4,c=1

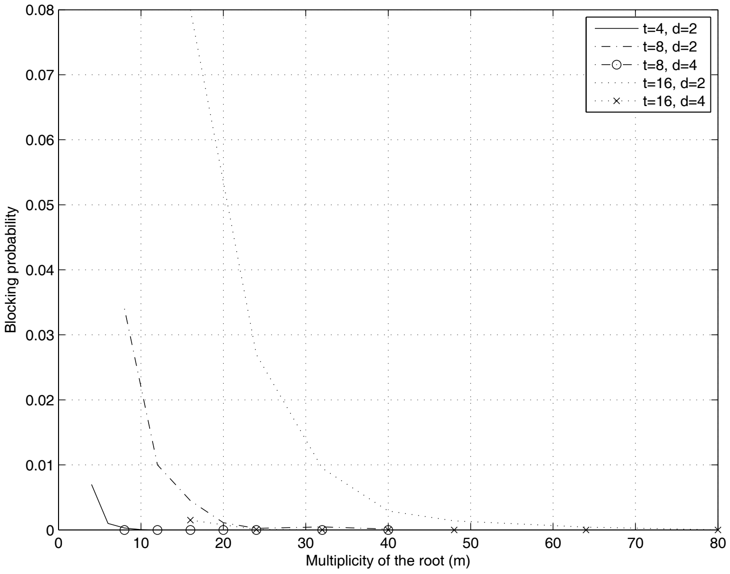


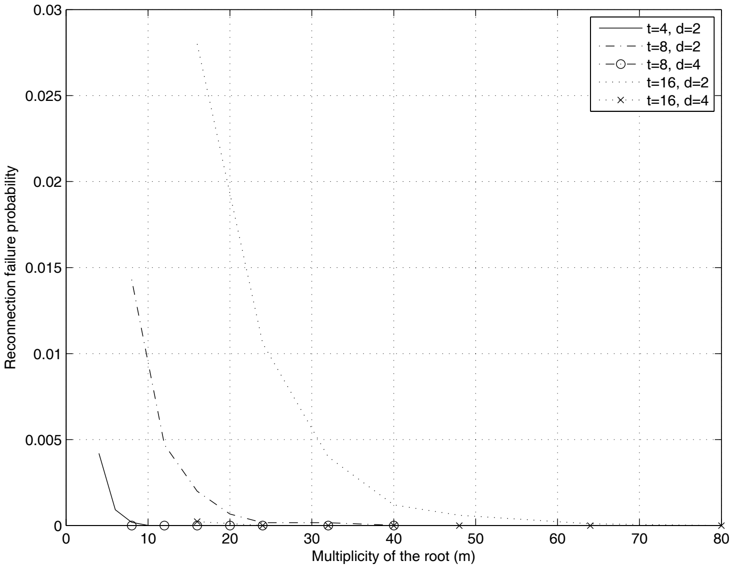
t=8,m=8,n=8

- where $\tau(j) = \begin{cases} j & \text{if } j < k \\ n & \text{if } j \ge k \end{cases}$
- $\pi(i)$ high if $p<p_{max}(n,k)$ (like for d=t)
- $\pi \rightarrow 0$ if $p>p_{max}(n,k)$
- **Similar results to d=1!**
- Effects of higher L

Generalized overlay (1<d<t)-dynamic

- Effects of increasing *d*
  - Increases the number of layers and mean number of children rooted at an arbitrary node (still *O(logN)*)
  - Decreases blocking and reconnection failure
- Probability of blocking and failed reconnections
  - Changes inverse proportional to *d*
  - Similar behavior as for *d=1* but **significantly lower**





- What is the optimal value for d?

Dynamic environment

- How to adapt to the changes of the departure rate and the loss probability?

  Domino effect: Low packet reception probability increases the departure rate $\rightarrow$ further decrease of π

  - Feed-forward
    - Robust control considering a set of possible operating conditions ($p \in [0, p_{\omega}^{max}]$)
    - Set redundancy for stable operation at p_{ω}^{max}
      - This ensures stable operation for all $p < p_{\omega}^{max}$
    - No measurement and estimation needed in the root
    - Sub-optimal performance if losses are low
  - Feedback-based
  - Incremental redundancy

Dynamic environment

- Feedback-based mechanism
  - Measure packet reception probability (π_a)
    - Aggregation tree
      - Measurement involves only trees where the node is sterile
      - Measured value is sent to the parent node in one of the fertile trees
    - Estimation of the packet loss probability at the root
      - e.g.: $p=1-\pi_a$
  - Possible feedback rules:
    - Fuzzy control based on human knowledge
    - Based on equations for the evolution of π_a and π
      - Minimize for the worst case in the $1-\alpha$ confidence interval of the estimate (min-max-α)
      - Model the evolution of π_a

Dynamic environment

- Incremental redundancy
  - Distributed solution
  - Root creates $k+r$ trees
    - r trees are for redundancy only
      - LDPC codes
      - Raptor codes
  - Nodes subscribe to $k+\rho$ trees ($\rho \leq r$)
  - Choice of ρ depends on the packet reception probability that individual nodes experience
  - Nodes with high bandwidth
    - Can reach higher packet reception probability
    - Serve as reconstruction points for the stream
  - Issues
    - How to maintain capacity balanced in each tree?

Conclusions and discussion

- Analytical model of a robust p2p multicast overlay
  - Packet reception shows non-graceful degradation
  - Factors that influence the cost of the overlay maintenance – reconnection failures
- Proposed general overlay
  - Shows good properties
  - Choice of optimal d
    - Future work based on analytical models
- Issues regarding deployment
  - How to set the FEC parameters
    - Feedback vs. feedforward vs. decentralized
  - How to maintain the overlay
    - Centralized
    - Distributed – structured/non-structured