PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes

Arnaud Legout/Ernst W. Biersack
Eurecom
Arnaud.Legout@eurecom.fr

Outline

- Motivation.
- Introduction and application of the FS-paradigm.
- Introduction of the PP bandwidth inference mechanism.
- Presentation of PLM.
- Simulations on basic scenarios.
- Simulations on realistic scenarios.
- Conclusion.

Motivation

- Audio/video (multimedia) applications take a growing place in the Internet.
- No satisfactory Congestion Control (CC) protocols for multicast delivery:
  - RLM (low convergence time, unstability, unfairness, loss induced, ...).
  - RLC, TCP-like version of RLM (low convergence time, loss induced, ...).
- We previously introduced the FS-paradigm that allows to devise nearly ideal end-to-end CC protocols:
  - We applied the FS-paradigm to design a new layered multicast congestion control protocol.
  - PLM is a practical test of the validity of the FS-paradigm.

Introduction to the FS-paradigm

- FS-paradigm for the design of end-to-end CC protocols:
  - Set of assumptions:
    - Network Part (NP): We assume a Fair Scheduler network.
    - End System Part (ESP): We assume selfish and non-collaborative end users (weak assumption).
  - Nearly ideal end-to-end CC protocols (stability, efficiency, fairness, robustness, scalability, and feasibility).
  - No need for specific mechanisms in the CC protocol to improve one of the properties of an ideal CC protocol.
  - Just address the application needs.
- The FS-paradigm does not give the mechanisms to meet the application needs but considerably simplifies the design of CC protocols.

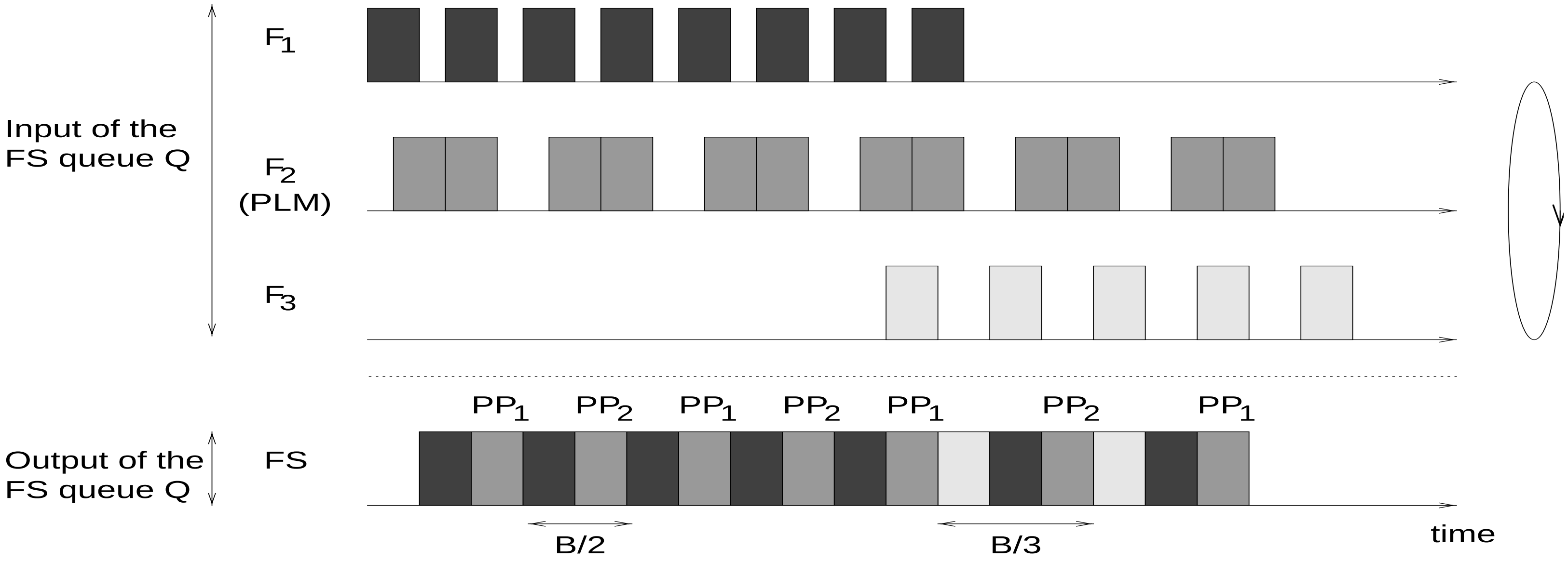
Application of the FS-paradigm

- FS-paradigm allows to focus on the application needs:
  - The (multimedia) application needs -- high throughput, fast convergence, low loss rate -- are not successfully addressed by the previous protocols.
- Previous protocols used bandwidth inference mechanisms based on congestion signal (losses, ECN, etc.):
  - The bottleneck buffer needs to overflow.
  - The congestion signal (missing packet) reaches the receiver far after the queue starts to build (i.e. congestion starts).
  - No information on the available bandwidth.
- PLM bandwidth inference mechanism based on **explicit available bandwidth notification** (Packet Pairs).
  - Has none of the congestion signal drawbacks.

PP Bandwidth Inference

- The Packet Pair (PP) bandwidth inference mechanism first introduced by Keshav (PP is two packets sent back-to-back, PP+FQ = PP spaced out by the available bandwidth):
  - A **sender** based version.
  - Estimates used for a fine grain rate adjustment.
  - Needs complex estimator to filter out noise.
- The PP bandwidth inference mechanism applied to PLM:
  - A **receiver** based version of PP. Less noise (Paxson), no problem due to the reverse path noise and bottleneck.
  - We use PP for a coarse grain adaptation, less sensitive to noise.
- A PP signal of congestion leaves the queue before the queue starts to build and far **before the queue overflows.**

PP Strengths: Example



- Before F3 starts, the PP space gives the available bandwidth B/2.
- **One** FS round after the first packet of F3 was backlogged in Q, a PP leaves Q spaced out by the available bandwidth B/3:
  - This PP was backlogged in Q before F3 started in Q.
  - This PP leaves Q when there is only one packet of F3 backlogged in Q (far before the queue starts to build).

Presentation of PLM

- PLM is a **receiver-driven** cumulative layered multicast congestion control protocol:
  - Data that can be striped in **cumulative** layers (Audio/Video/Data).
  - Multicast capable network.
  - Fair scheduler network.
- PLM source:
  - The source sends each layer on a different multicast group.
    - Same multicast tree for all the multicast groups of the same PLM session.
  - **The source sends on each layer packet by pair (PP).**
- The PPs allow to dynamically infer the **available bandwidth** for each receiver (explicit available bandwidth notifications).

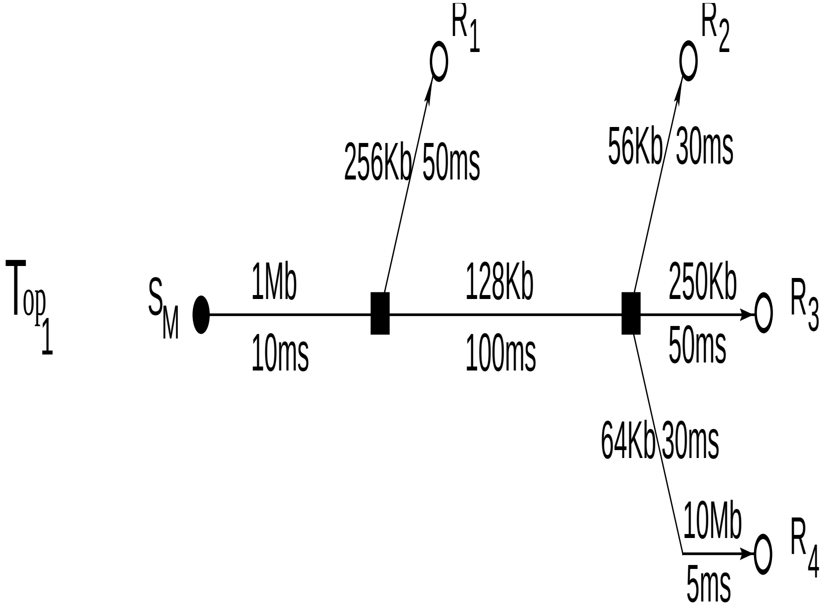
PLM Algorithm

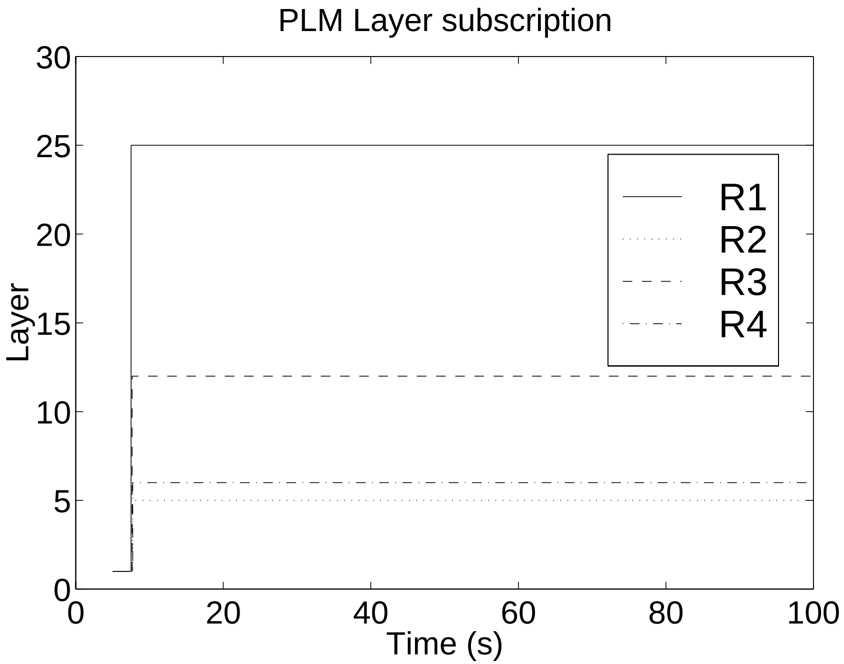
- PLM receiver:
    - Each PP received leads to an estimate of the available bandwidth.
    - We drop layers each time we have an estimate lower than the current layer subscription until the layer subscription is lower than the estimate.
    - We add layers according to the minimum estimate received during a period C (the Check value) if all the estimates received during C are greater than the current layer subscription.
- Note the simplicity of the protocol.
- Does it work?

Simulations: Basic Scenarios

- Simple scenarios, not intended to be realistic.
- Allow to assess the fundamental properties of PLM.

A Single PLM Session: Convergence

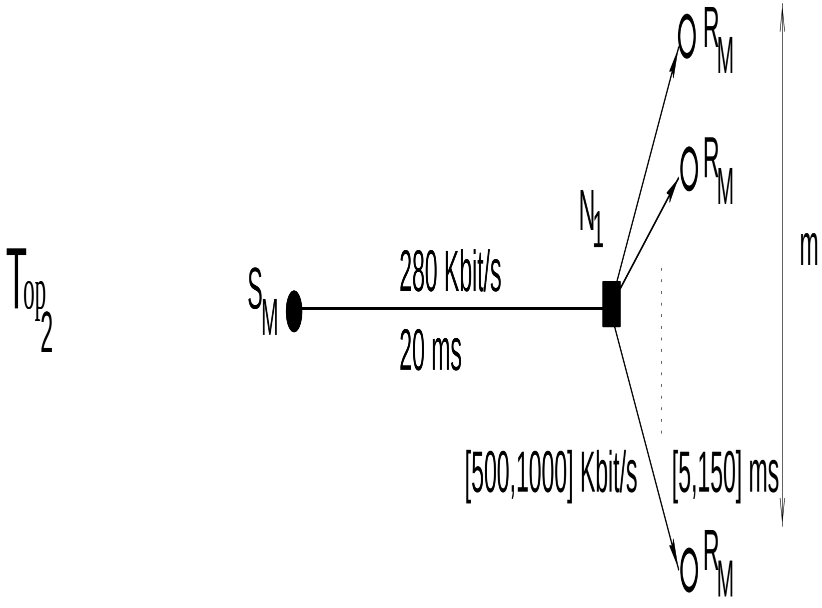


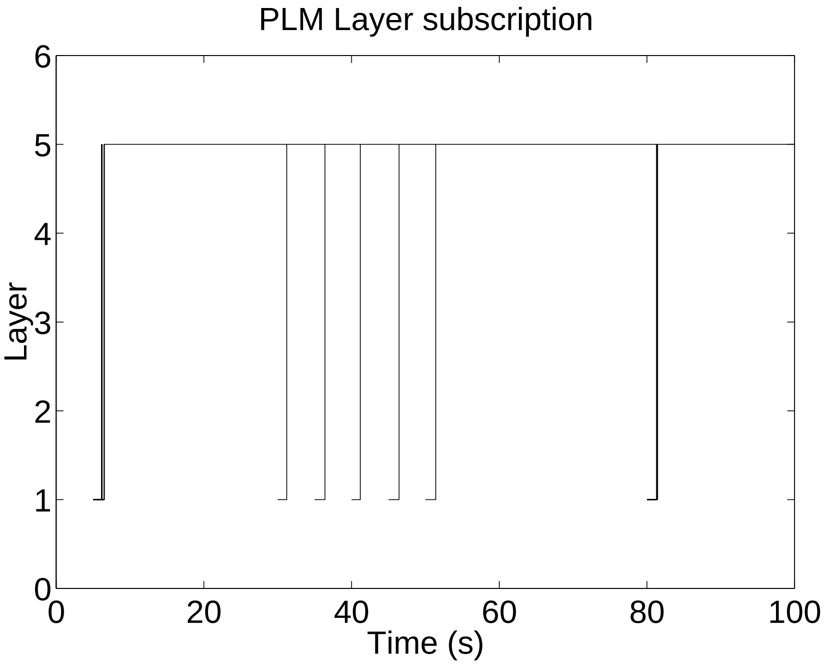


- Evaluation of the speed, stability, and accuracy of the PLM convergence in the context of a large heterogeneity of delay and bandwidth.
- 10 Kbit/s per layer (tough test).

- All the receivers converge to the optimal rate in the order of C=1 second and stay at this rate during the whole simulation. **No loss induced**.

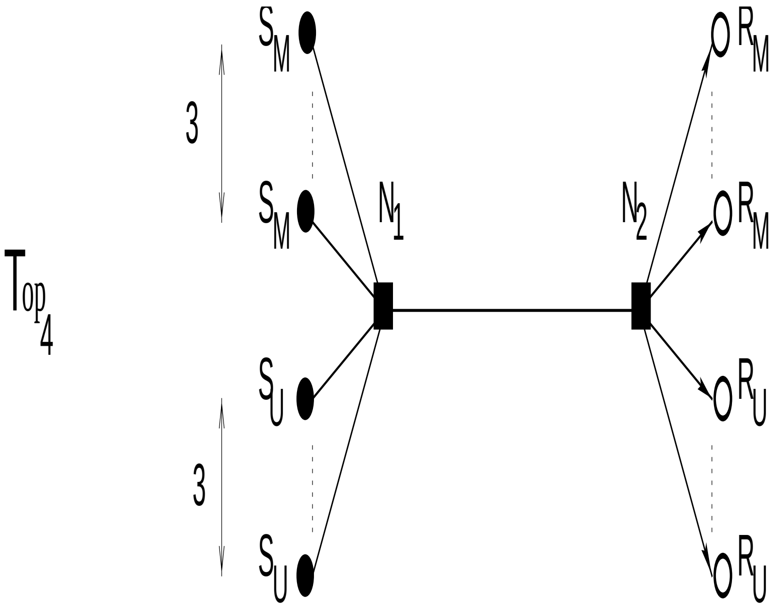
A Single PLM Session: Scalability

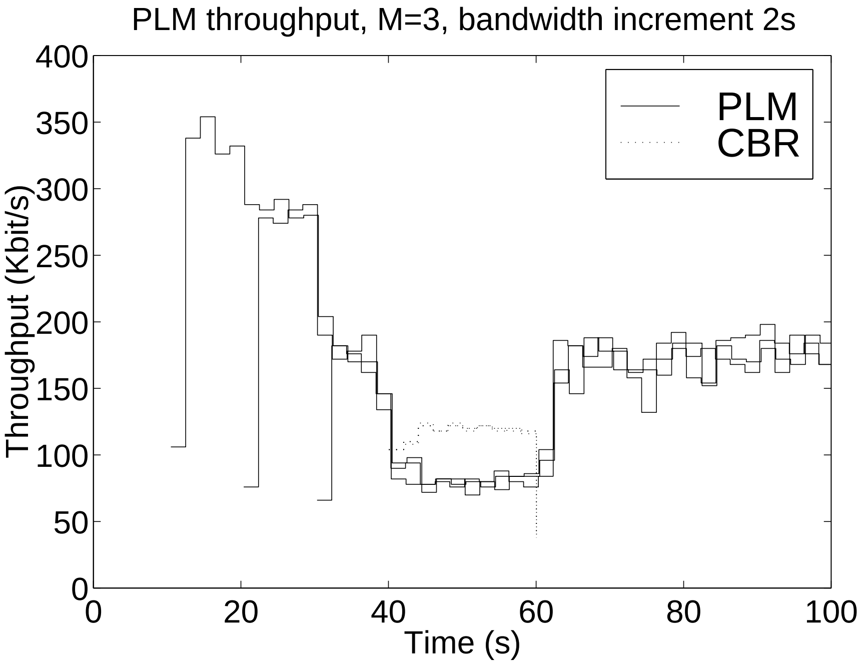




- Evaluation of the PLM scalability with the number of receivers and with late joins.
- 50 Kbit/s per layer.
- 20+5+5 receivers.
- PLM convergence is independent of the number of receivers and of the late joins. **No loss induced**.

Multiple PLM and CBR Sessions



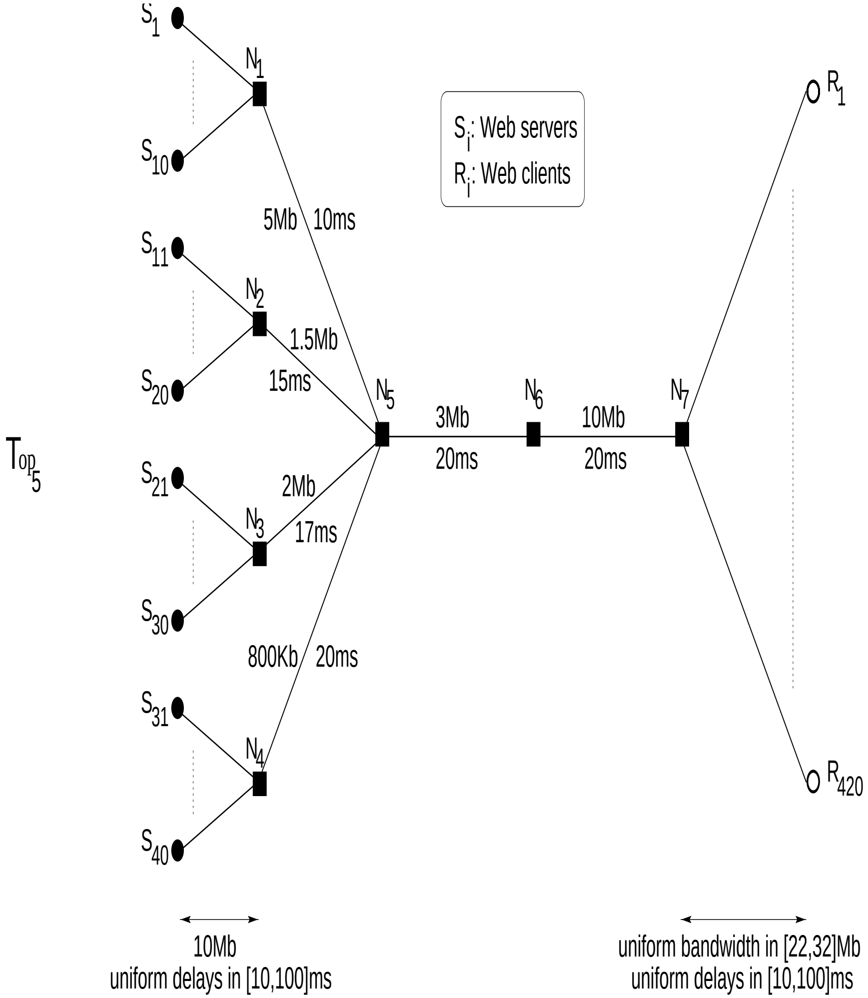


- 3 PLM + 3 CBR. Evaluation of the scalability of PLM with the number of sessions, PLM adaptation to heavy congestion.
- 20 Kbit/s per layer.
- PLM adapt to the available bandwidth in less than a RTT. **No loss induced** even in case of high congestion.

Simulations: Realistic Background Traffic

- PLM performs very well for the basic scenarios:
  - Fast convergence, stability, scalability, fairness, no loss induced.
  - Do these nice properties still hold in a realistic environment?
- Strong evidence of self similar and even multifractal traffic in the Internet.
  - We use the Anja Feldmann’s Sigcomm’99 scenarios and add a PLM session in these scenarios (self similar and multifractal background traffic).

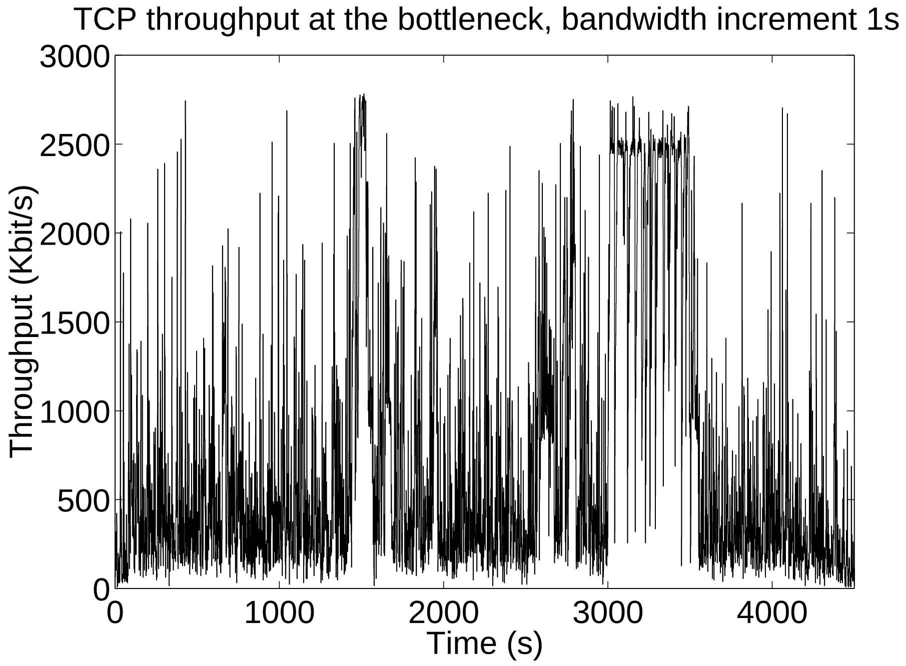
Topology



- Evaluation of PLM with a complex background traffic (Feldmann):
  - S_i are web servers, R_i are web clients.
  - A session is defined for a client. Each session contains 300 pages, each page contains 1 object:
    - 100 sessions: lightly loaded.
    - 400 sessions: heavy loaded.
  - For a given session, a client requests each page on a randomly chosen server.
  - The object size is Pareto distributed.

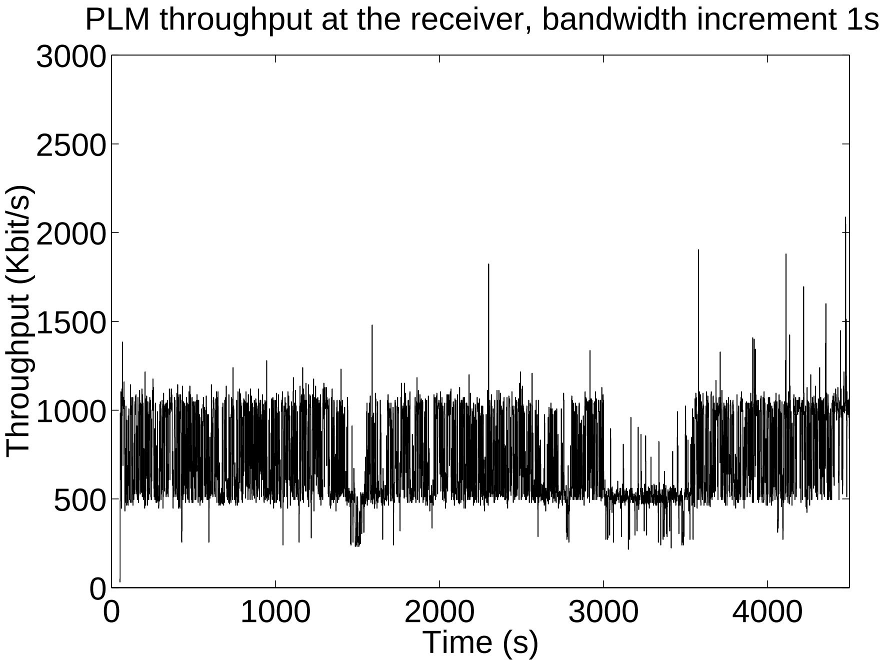
PLM in a Multifractal Environment

- Background traffic



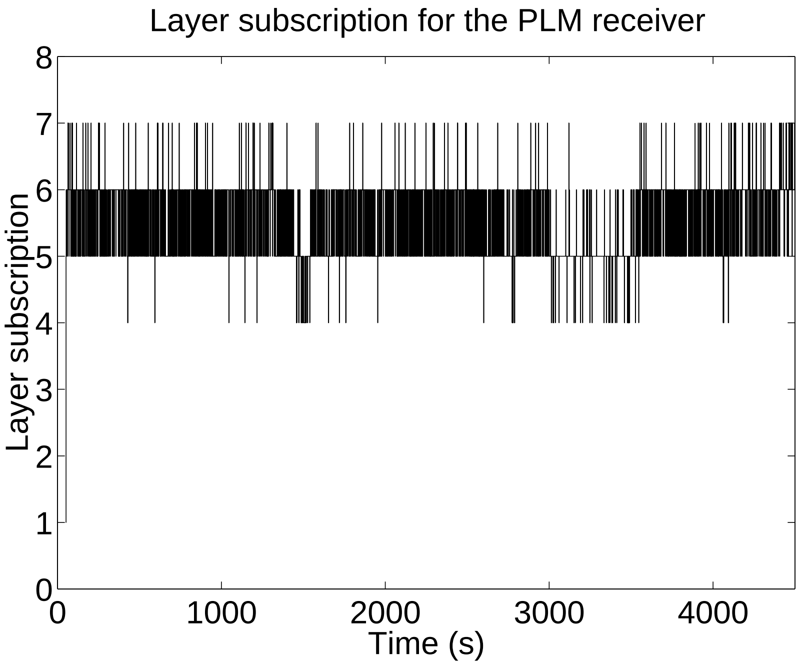
- 100 sessions. Mean throughput for the background traffic at the bottleneck (N_5,N_6): 737 Kbit/s.
- Self similar and multifractal background traffic.

- PLM

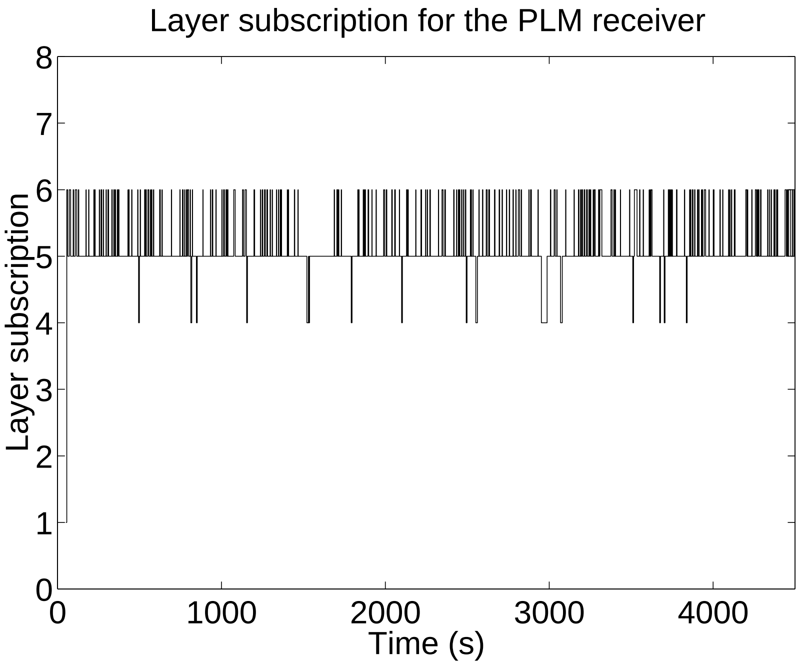


- Mean throughput seen by the PLM receiver. C=1s, exponential layers, 1000 bytes packet size.
- PLM closely follow the background traffic, **no loss**.

PLM in a Multifractal Environment



- Layer subscription for C=1s, 100 sessions, exponential layers, and 1000 bytes PLM packet size.
- 2090 layer changes. This is not a sign of instability.
- PLM mean throughput: 733 Kbit/s.



- Layer subscription for C=5s, 100 sessions, exponential layers, and 1000 bytes PLM packet size.
- 417 layer changes.
- PLM mean throughput: 561 Kbit/s.

Final Conclusion

- We have introduced the FS-paradigm for the design of CC protocol. Its appealing properties lead us to devise a new CC protocol using this paradigm.
- We devised PLM, a new multicast CC protocol for audio/video/data multicast dissemination:
  - PLM bandwidth inference mechanism based on PP.
  - PLM outperforms all the previous multicast layered CC protocols:
    - PLM converges fast to the optimal rate and tracks this rate with no loss induced. PLM is efficient, stable, fair, and simple.
    - PLM still performs well with a realistic background traffic.
- PLM is incontestably a practical validation of the FS-paradigm.

Thanks!