The FS Paradigm and Its Practical Validation: PLM

Arnaud Legout
Eurecom
Arnaud.Legout@eurecom.fr
Outline

- Motivation.
- The FS paradigm:
  - Definition of congestion.
  - Properties of an ideal congestion control protocol.
  - Introduction of the FS paradigm.
  - Application of the FS paradigm.
- The PLM multicast congestion control protocol:
  - Introduction of the PP bandwidth inference mechanism.
  - Presentation of PLM.
  - Simulations on basic scenarios.
  - Simulations on realistic scenarios.
- Conclusion.
Motivation

- Congestion control is a **central** problem in networks:
  - No congestion control = congestion collapse.
  - Basic but efficient way to achieve quality of service.

- Congestion control is a **complex** problem in networks:
  - Distributed algorithm that must **optimize** the resource allocation of the network.

- Jacobson and Karels solution for TCP:
  - Collaborative protocol (linear increase, multiplicative decrease).

- The **TCP-Friendly paradigm** for applications that can not use TCP:
  - Preserve the collaborative assumption.
  - Based of TCP long-term behavior.
Motivation

- Collaborative assumption for CC:
  - Strength:
    - Does not require any network support to achieve fairness, efficiency, and stability.
  - Weaknesses:
    - Requires collaboration of all end users, cannot be longer assumed:
      - New applications perform better with non TCP-friendly protocol.
      - Very constraining when devising new CC protocols.
- TCP-friendly well suited for short term, NOT for long term.
- Beyond TCP-friendliness:
  - Very controversial.
  - Essential to significantly improve congestion control protocols.
Motivation

- The network support can help CC:
  - Wide range of network support: from buffer management to active networking.
  - We still want a best effort network.
  - We have to respect the End-to-End argument.
- Network support can simply be a Fair Scheduler (FS), two main contributions:
  - Keshav: Fair Queuing (FQ) + Packet Pair.
  - Shenker: Game theoretic study of CC.
- Two very promising results, but still no paradigm for the Internet:
  - Offer an great alternative to the TCP-friendly paradigm.
The FS paradigm
Definition of Congestion

- Congestion related to:
  - User satisfaction.
  - Network performance.
  - Such a definition introduced by Keshav.

- Definition of congestion:
  - Congestion: decrease of satisfaction due to a modification of the performance (bandwidth, delay, jitter, etc.) of the connection.

- A CC protocol must avoid congestion.
Properties of an Ideal CC Protocol

- We assume **selfish** users.
- Abstract formulation: the properties remain very general.
- **Nash equilibrium, Pareto optimality.**
- Properties of an ideal CC protocol:
  - Stability: Existence and uniqueness of Nash equilibrium.
  - Efficiency: Fast convergence toward Pareto optimality.
  - Fairness: Max-min fairness.
  - Robustness: Against malicious, misbehaving, and greedy users.
  - Scalability: With bandwidths heterogeneity, receivers, etc.
  - Feasibility: Technical requirements (Hardware, Software, Easy to evaluate,…).
- How can we devise such a CC protocol?
**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 (sufficient condition).
  - 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

- The study of FS paradigm is formal:
  - Need a pragmatic validation.
- Multicast dissemination of audio/video (multimedia) content is challenging for CC:
  - No satisfactory congestion control protocols for multicast delivery.
  - RLM and RLC exhibit fundamental pathological behaviors, but use interesting architectural choices (receiver-driven cumulative layer multicast protocol).
  - We applied the FS paradigm to design a new receiver-driven cumulative layered multicast congestion control protocol.
Application of the FS-paradigm

- 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.
- Bandwidth inference mechanism based on explicit available bandwidth notification (Packet Pairs).
  - Has none of the congestion signal drawbacks.
- PLM is a pragmatic test of the validity of the FS paradigm.
Packet pair Layered Multicast (PLM)
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.
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, no specific mechanism to improve one of the properties of an ideal CC 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).
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. \(C=1\)s, 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.
PLM validate the FS paradigm

- PLM is:
  - Stable, fast convergence, no pathological oscillations.
  - Efficient, track the available bandwidth without loss induced.
  - Fair with TCP and PLM.
  - Robust against other congestion control protocols.
  - Scalable due to the receiver-driven cumulative layered solution.
  - Feasible due to its simplicity, in the ns distribution.
Final Conclusion

- We have introduced the FS-paradigm for the design of CC protocols. Appealing properties, but need a pragmatic validation.

- 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.

- FS paradigm + PLM = original and comprehensive study of the CC problem.

Thanks!

9/22/00