Comsnets 2009

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Handling Very Large Numbers of Messages in Distributed Hash Tables

Fabius Klemm

www.wuala.com

Jean-Yves Le Boudec, Dejan Kostic, Karl Aberer

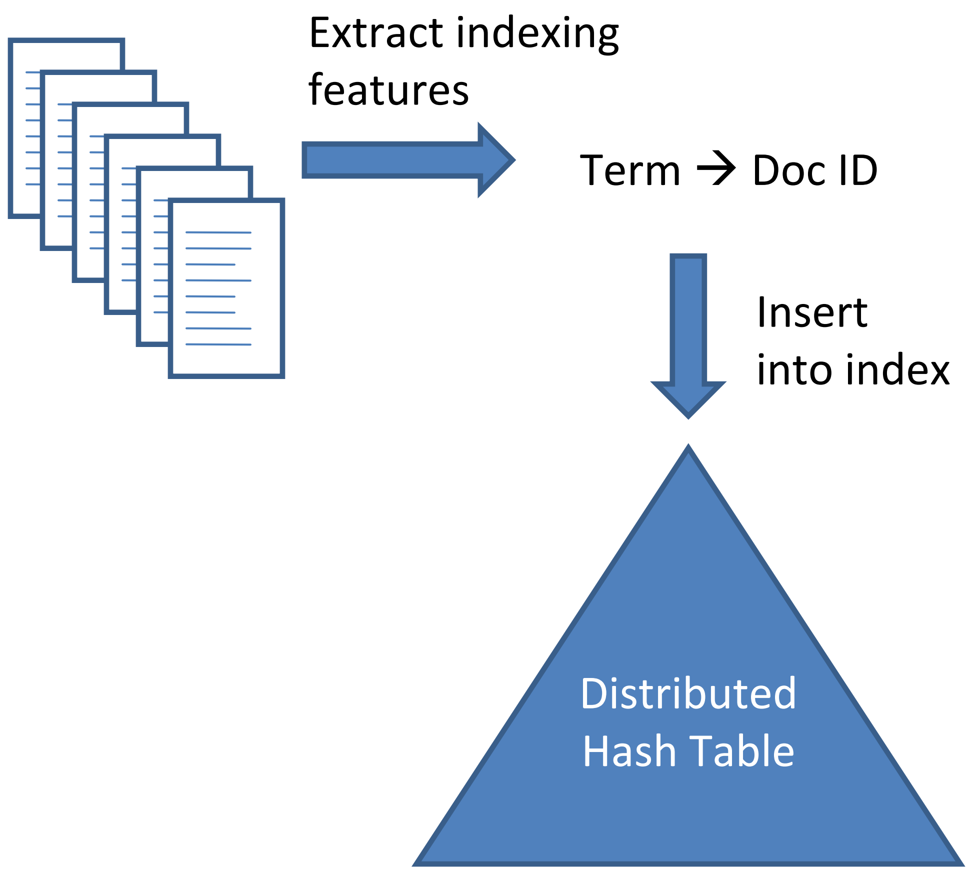
Contribution

- Analysis:
  - A DHT can suffer a congestion collapse
- Emulation + PlanetLab results:
  - Evaluation of different congestion control algorithms for DHTs

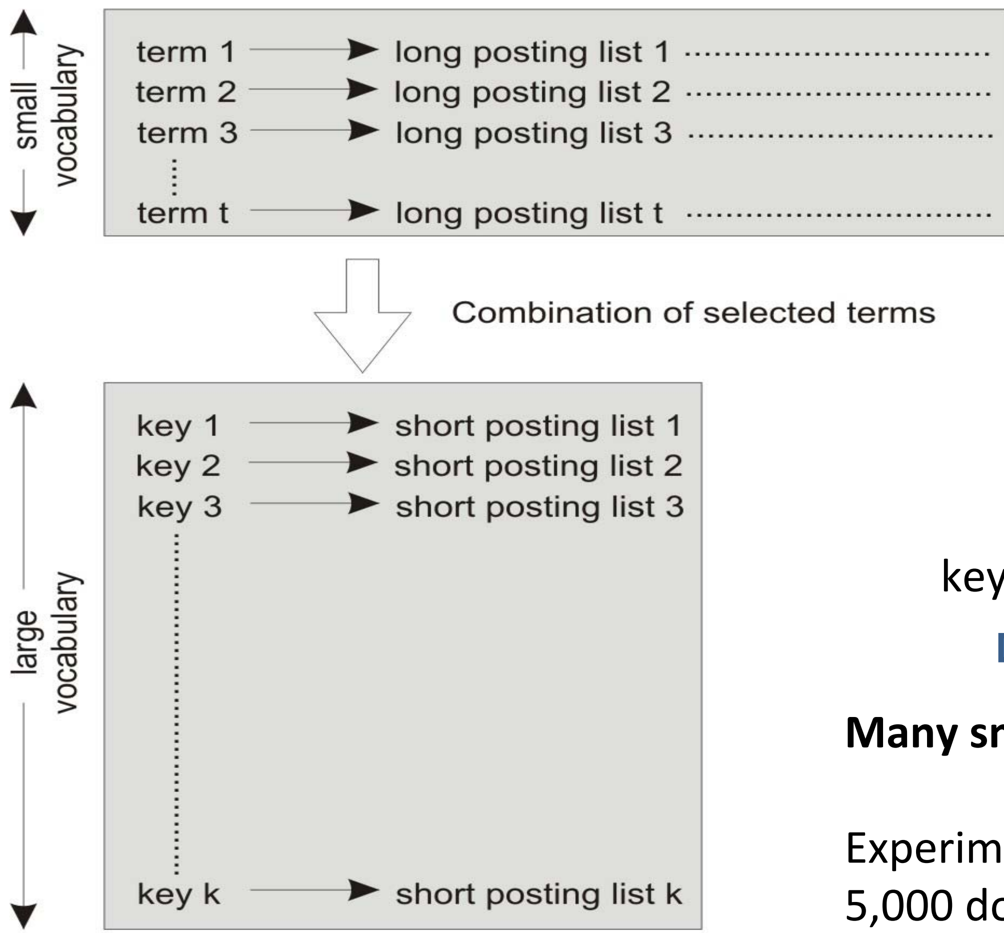
Motivation

- Peer-to-Peer Information Retrieval
- Goal: Build an P2P-IR System for the WWW that scales with millions of peers

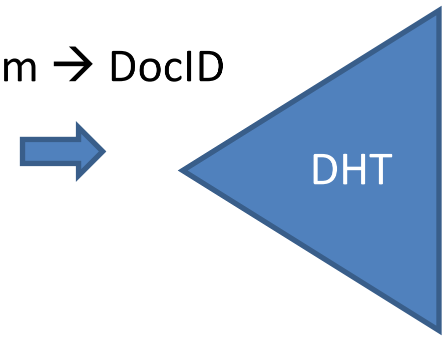
Overview of P2P-IR



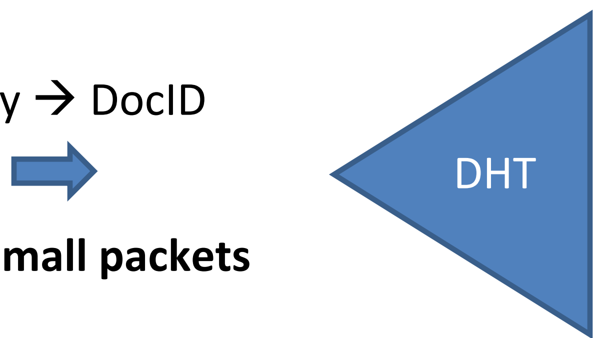
Problem: Long Posting Lists



term → DocID



key → DocID



Many small packets

Experimental results:
5,000 documents →
8,000,000 keys

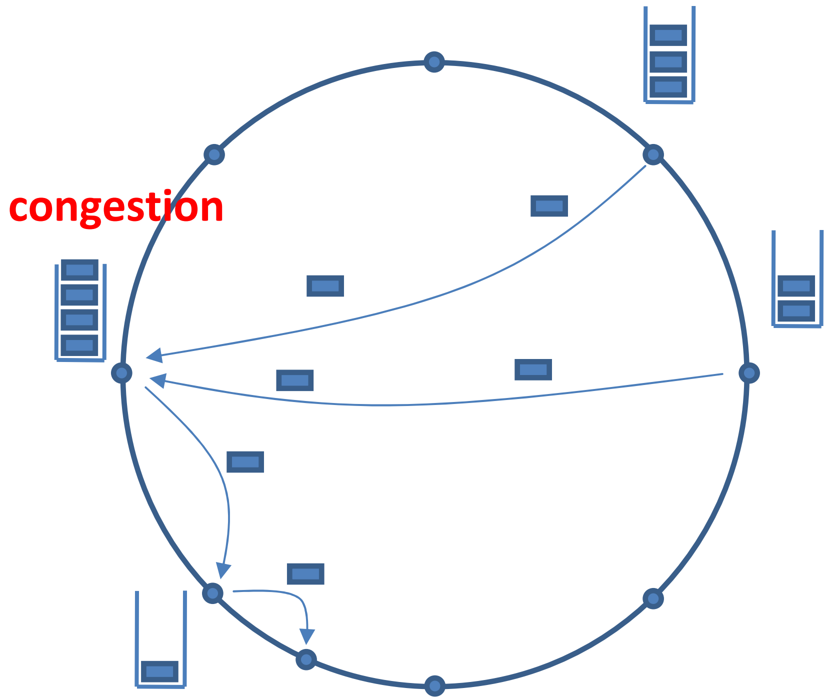
key: set of terms

Congestion Control for DHTs

- Initial DHT design was for moderate loads
  - File sharing, name resolution
- New demands
  - Insert **billions of small packets** into a DHT
  - **High throughput**: Maximize the number of insert operations per peer per time unit
  - **Low latency**

Congestion in DHTs

- What is congestion?
  - A peer receives more packets than it can handle
- Reasons for congestion:
  - **High load**
  - **Heterogeneous peers**
    - Link bandwidth, CPU
  - **Skewed traffic**
- Independent of the routing protocol
- Recursive routing



Congestion Collapse Analysis

- Goal of this analysis:
  - Show what happens in a DHT without Congestion Control

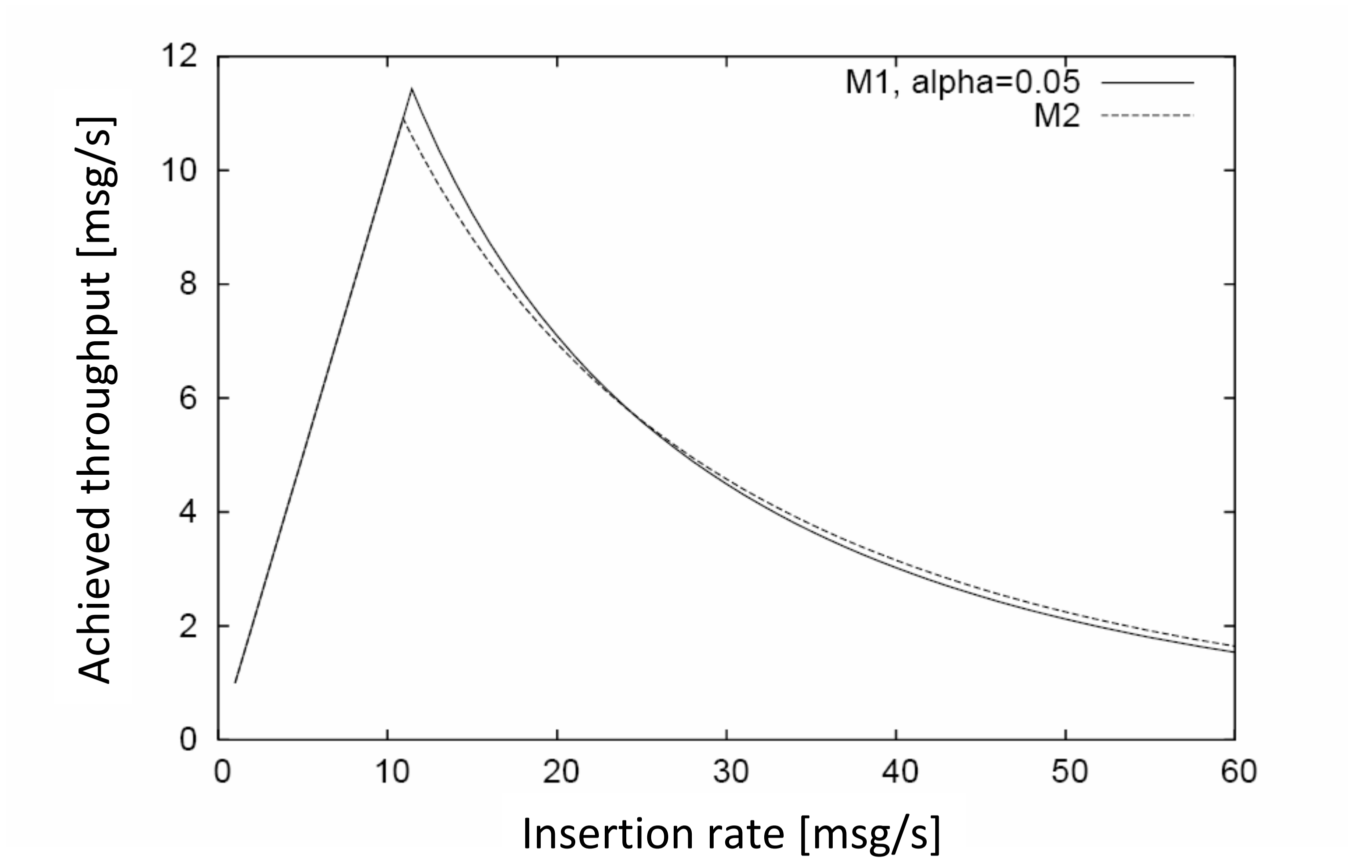
- Given:
  - Given a DHT with n peers
  - Each peer has a capacity c [msg/s]
  - Each peer inserts new requests with a certain rate r [msg/s]

  - If the offered traffic exceeds the available capacity, requests are dropped

  - What is the achieved throughput?

Congestion Collapse

- Example: DHT with 1 million peers



Solutions

Congestion control (a.k.a. rate control)

- Avoid that the DHT is overloaded with packets

Two approaches:

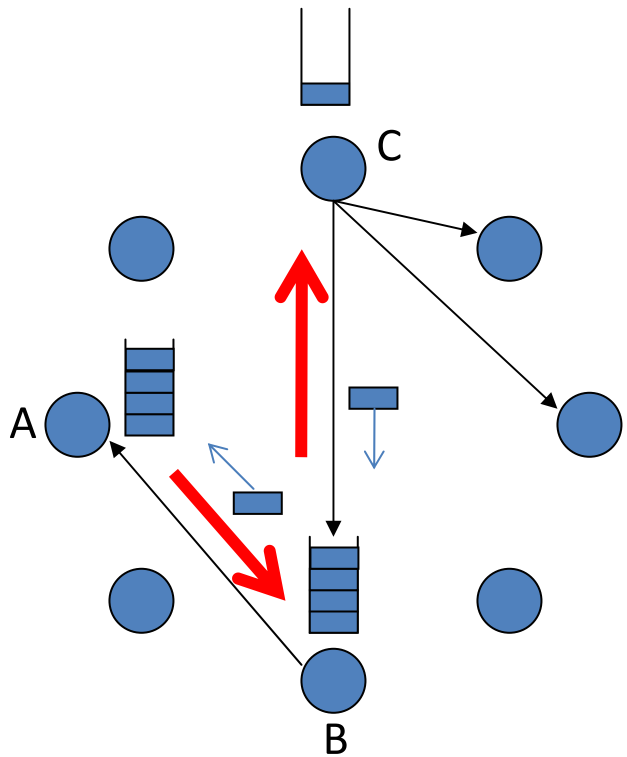
- Hop-by-hop: Back-pressure along relaying peers (local feedback)
- End-to-end: Peers adapt their insertion rates using global feedback from the overlay network

1. Back-Pressure Congestion Control

- Hop-by-hop
- **Block** sending peers when incoming queue is full

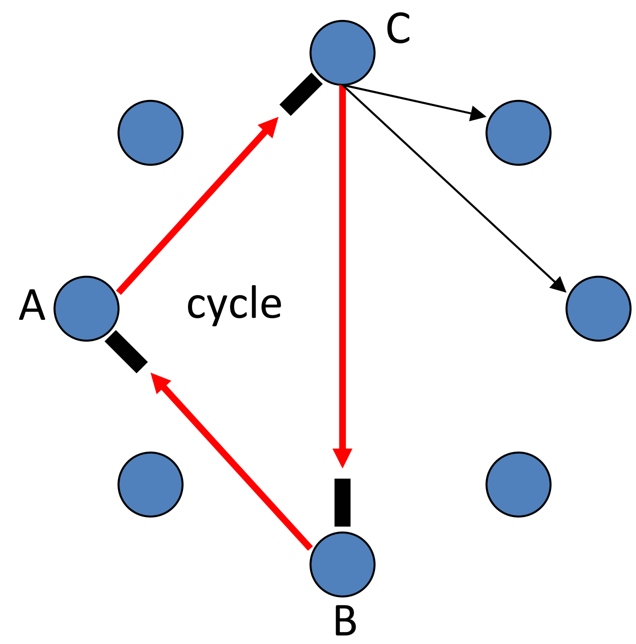
  →No packets are dropped

- Problems?



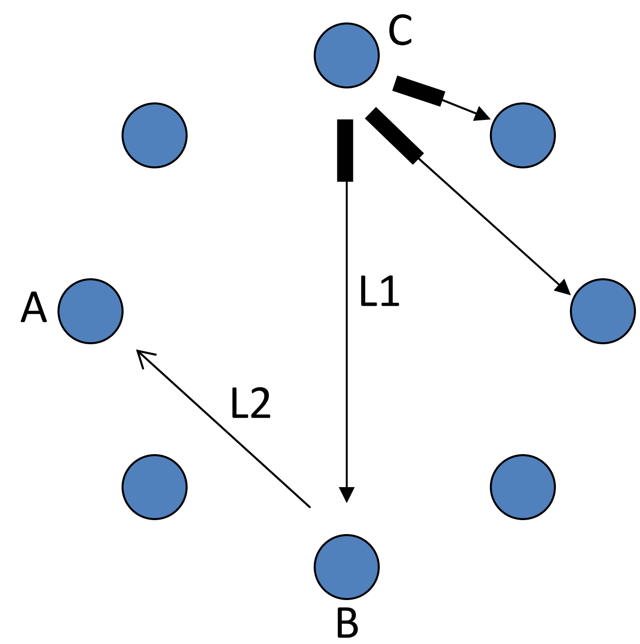
Problems with Back-Pressure

- Risk of deadlocks:
  - All peers are waiting for other peers to receive the next packet
- Possible when there is a **cycle** in the buffer waiting graph
- Remedy?



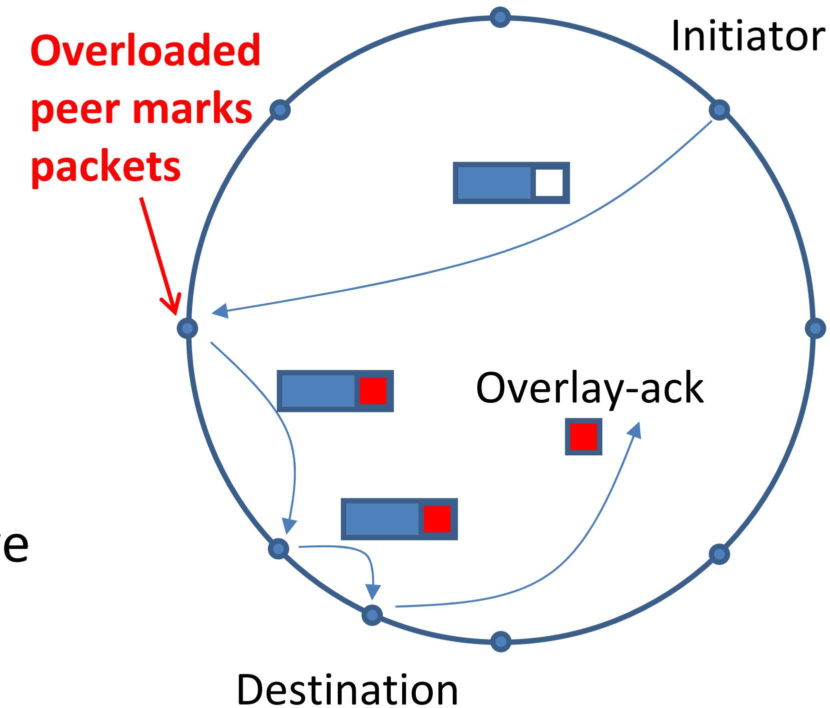
Deadlock-free Back-Pressure for Rings

- One queue per outgoing link: O(log n) queues
- Why deadlock-free?
  - “Hop lengths” **strictly monotonously** decrease when approaching searched ID



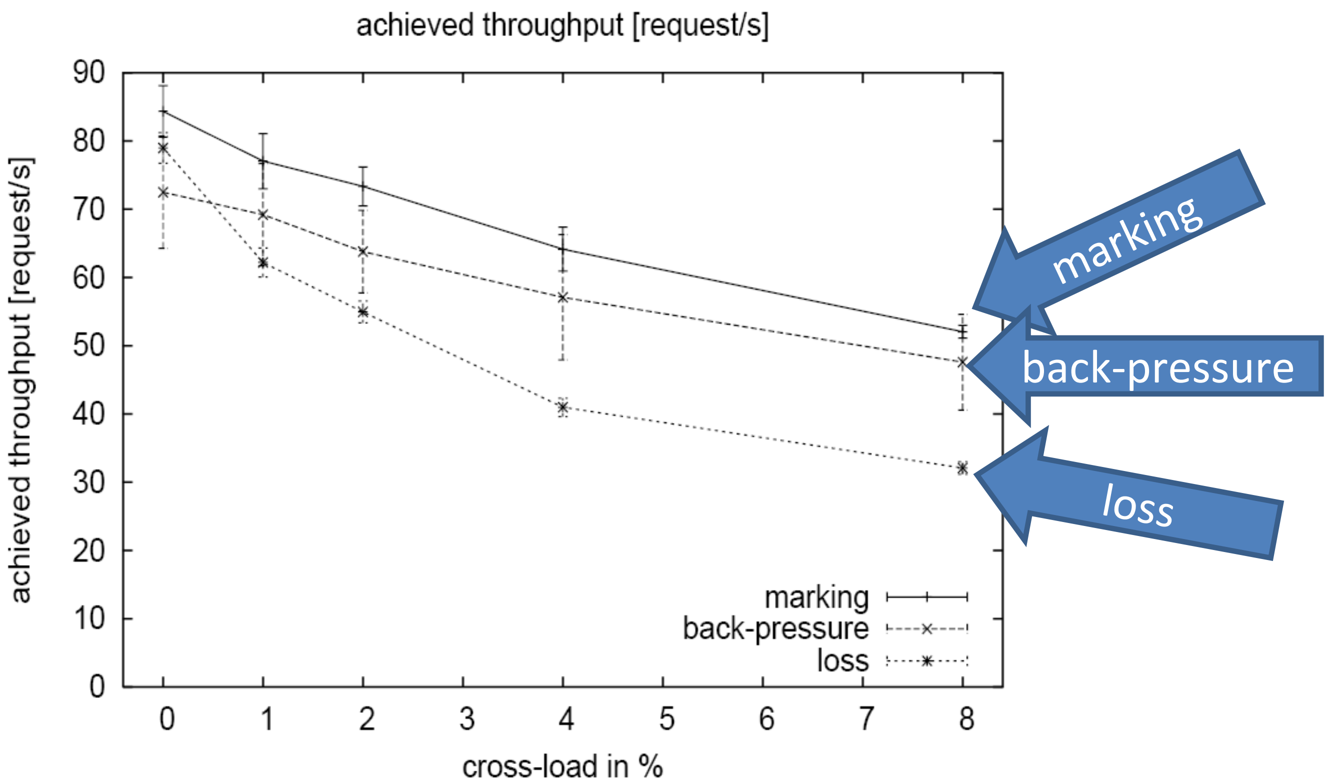
2. End-to-End Congestion Control

- Limit rates at which peers can insert new packets
- Feedback from the DHT
  - Overloaded peers mark packets
  - Mark return with overlay acknowledgement
- Peers increase or decrease insertion rate accordingly
  - Additive increase – multiplicative decrease (AIMD)
  - Destination-independent

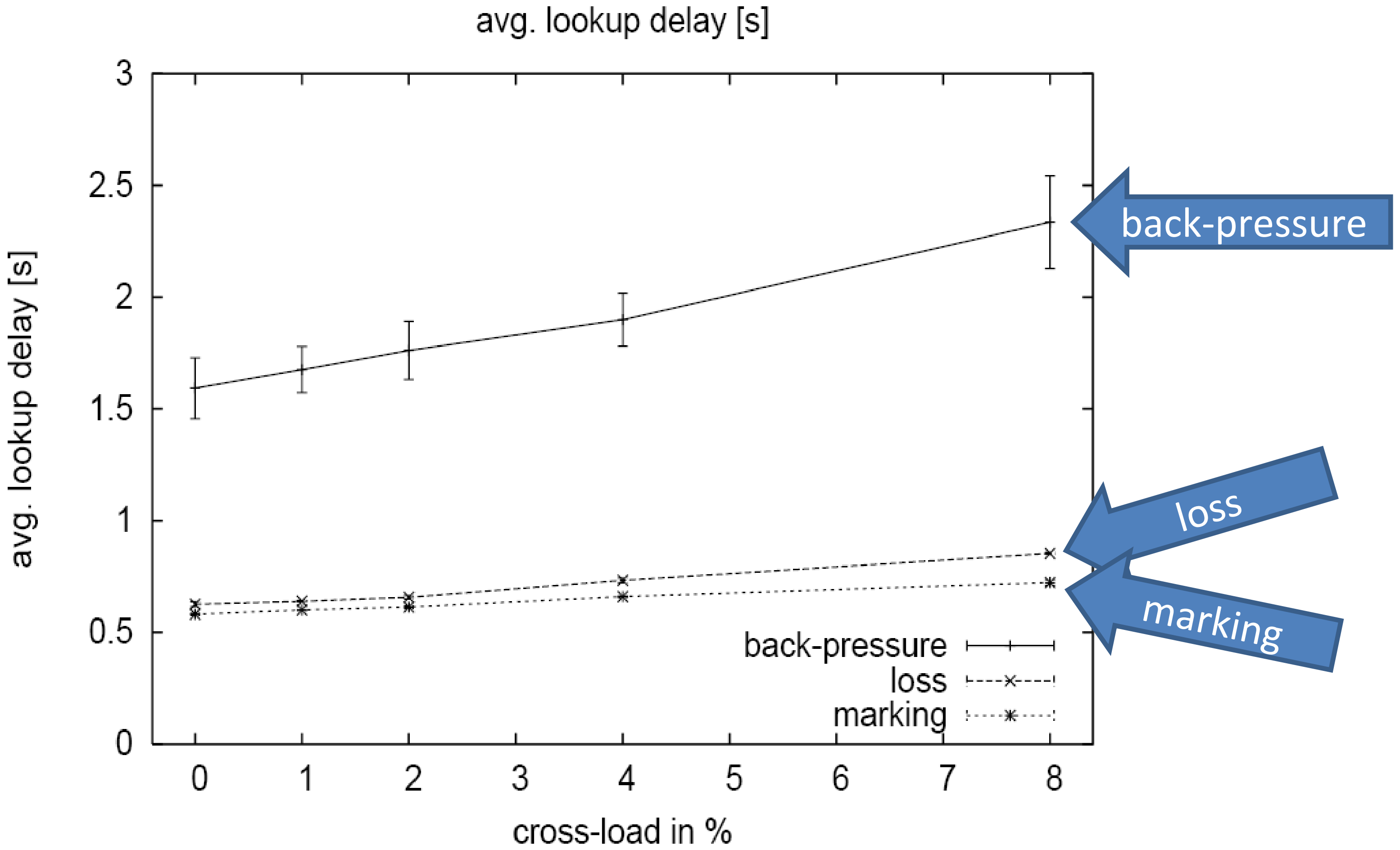


Evaluation - Throughput

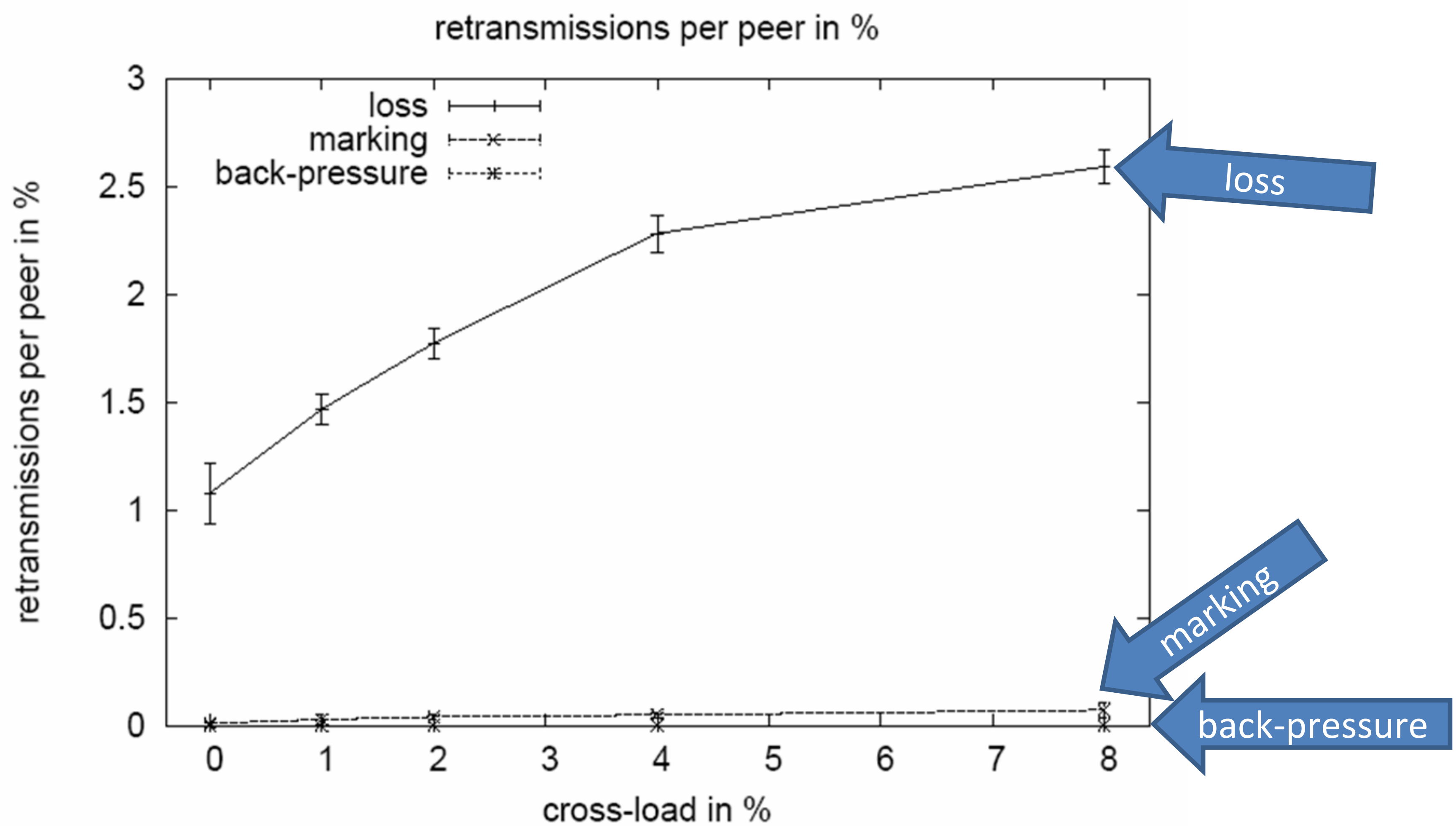
- 128 peers, ModelNet network emulation
  - 21 dual 3.4 GHz Xeon, 2 GB RAM
- Crossload: Peers stop forwarding



Evaluation – Latency



Retransmissions



Summary

- P2P-IR
- Analysis of a congestion collapse
- Marking performs best: low latency, high throughput
- Back-pressure
  - Deadlock-free with separate queues for each link in the routing table
  - Deadlocks possible with failures or malicious peers
    - Not suitable for a distributed, uncontrolled environment

Thank you for your attention!