Adaptive Server Allocation for Peer-assisted VoD

Konstantin Pussep, Osama Abboud, Florian Gerlach,
Ralf Steinmetz, Thorsten Strufe

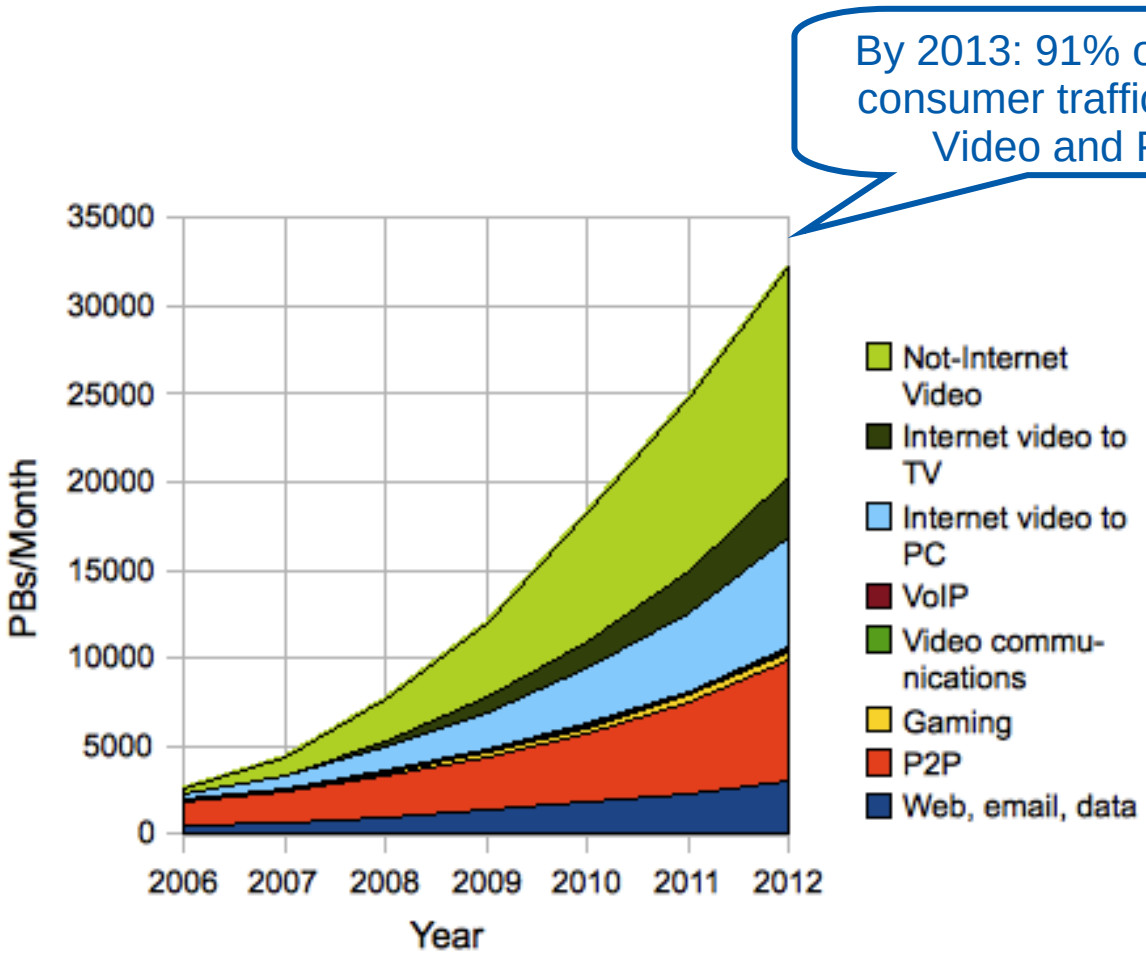
TECHNISCHE UNIVERSITÄT DARMSTADT

KOM - Multimedia Communications Lab
Prof. Dr.-Ing. Ralf Steinmetz (Director)
Dept. of Electrical Engineering and Information Technology
Dept. of Computer Science (adjunct Professor)
TUD – Technische Universität Darmstadt
Rundeturmstr. 10, D-64283 Darmstadt, Germany
Tel.+49 6151 166150, Fax. +49 6151 166152
www.KOM.tu-darmstadt.de

Konstantin Pussep

Konstantin.Pussep@KOM.tu-darmstadt.de
Tel.+49 6151 165188

Motivation



Video delivery dominates IP traffic

Can server farms deliver all this videos?

High costs:

- E.g. YouTube → 1 Mio $/day [Huang2007]

Scalability issues

- Servers must be dimensioned for peak demand

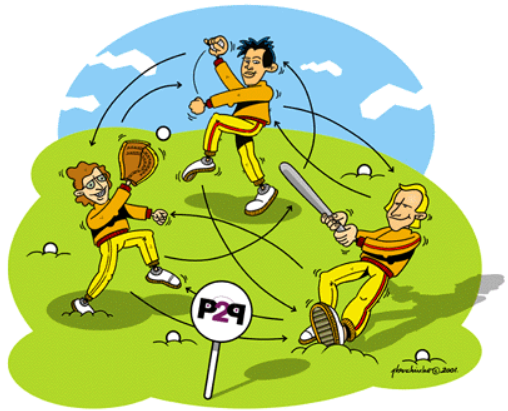
Data source: Cisco visual networking index, 2009

Video-on-Demand over Peer-to-Peer

TECHNISCHE UNIVERSITÄT DARMSTADT

P2P promise

- Self-scalable, resources grow with demand → handle flash crowds
- Cost-efficient → no server costs
- Availability of local replicas → less inter-domain traffic



No service guarantees in pure P2P systems

- Insufficient upload capacity (link asymmetry)
- Unreliability and dynamics of user behavior
- Firewalls, NAT boxes etc.

→Peer-assisted systems

- Servers as backup (service guarantees)
- Peers to offload servers

Challenge: Provide service guarantees at lowest possible server costs

Picture source p2p.weblogsinc.com/

Overview

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation and Problem Statement

Adaptive Allocation Policies

- Modeling
- Global Speed
- Supporter

Evaluation

- Sensitivity analysis
- Comparison

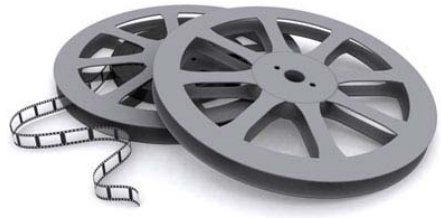
Summary and Next Steps

Scenario and Problem Statement

TECHNISCHE UNIVERSITÄT DARMSTADT

Commercial video distribution

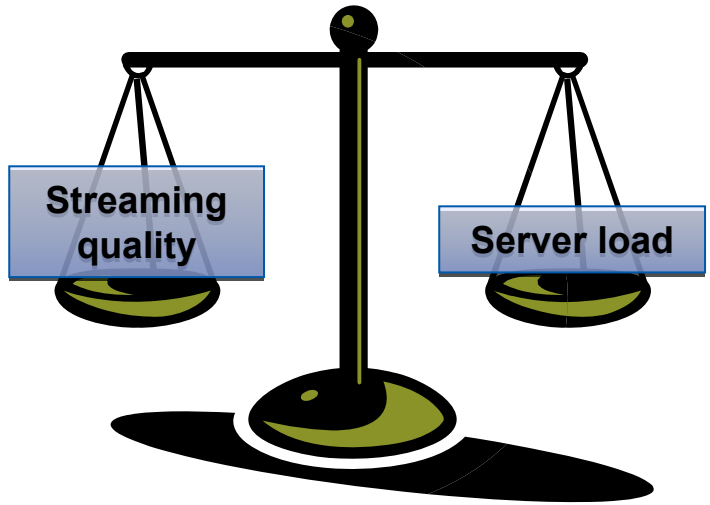
- User-generated content (YouTube etc.)
- Movie trailers (film studios)
- News
- Full-length movies etc.



Content provider applies peer-assisted streaming to reduce distribution costs

Assure high streaming quality

- Startup delay → few seconds
- Stall time → close to zero



How much server bandwidth should be allocated per peer and swarm over time?

Bandwidth Demand-Supply-Model

TECHNISCHE UNIVERSITÄT DARMSTADT

Demand:

$$D_{required} = r \cdot f \cdot |P|$$

Video bitrate

Current downloaders

Prefetching factor 1~2

Supply:

$$U_{total} = \sum_{p \in P} u_p \cdot g + \sum_{s \in S'} u_c$$

Peer contribution

Upload utilization <= 1

Server contribution

Bandwidth matching (U≥D):

$$\sum_{s \in S'} u_s \geq \max\left[\sum_{p \in P} (r \cdot f - u_p \cdot g), 0\right]$$

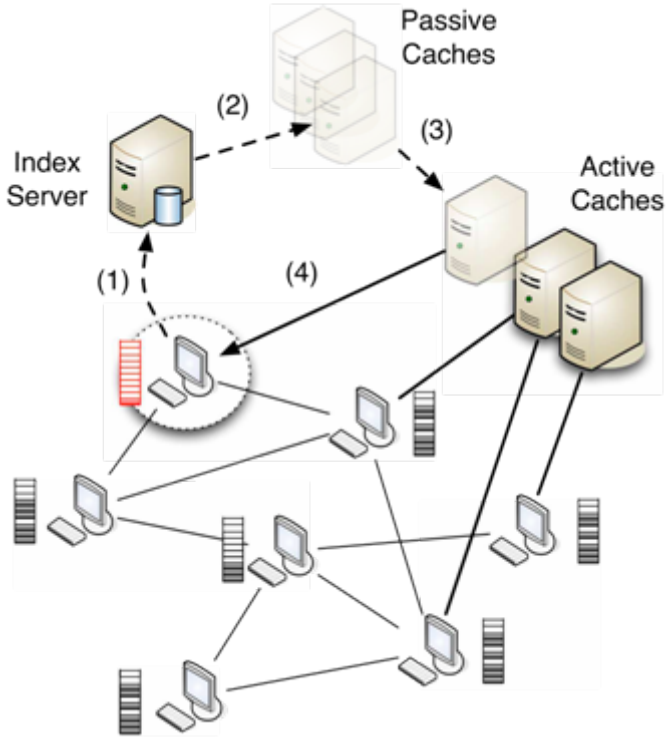
f, g, and u_p are unknown

Adaptive Server Allocation Policies

TECHNISCHE UNIVERSITÄT DARMSTADT

Mechanism

1. Peers report their performance to the index server
2. Index server determines required server contribution and
3. … allocates or disables servers
4. Servers upload to (some) peers to avoid streaming quality degradation



Policy components

- **Monitoring:** Data and frequency
- **Decision metric:** How much resources are needed?
- **Connection management:** How to join the overlay, whom to serve, when to leave?

Global Speed Policy

TECHNISCHE UNIVERSITÄT DARMSTADT

Idea:

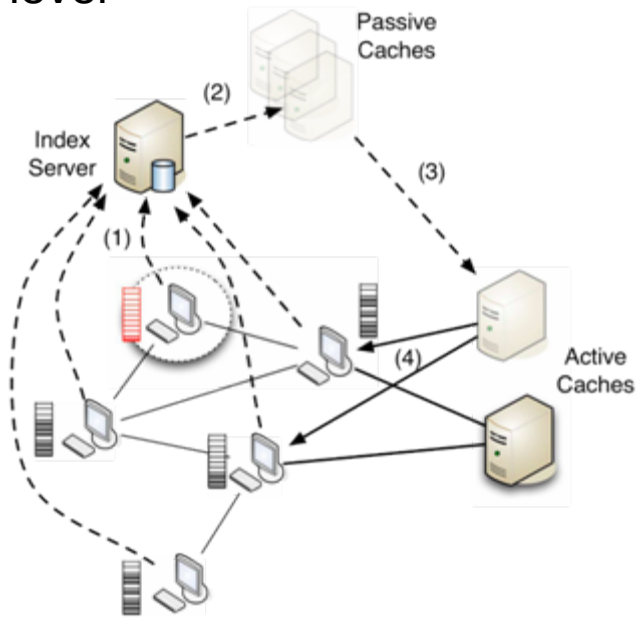
- Peers report their download speed each X seconds
- Total average speed is calculated over last X seconds
- Target speed is video bitrate plus prefetching overhead
- Balance the average download speed at the target level
  - Add or remove server bandwidth

$$d_{average} = \frac{\sum_{p \in P} d_p}{|P|}$$

$$d_{target} = r \cdot f'$$

$$d_{demand} = d_{target} - d_{average}$$

Estimated prefetching factor **f**



Supporter Policy

TECHNISCHE UNIVERSITÄT DARMSTADT

Idea:

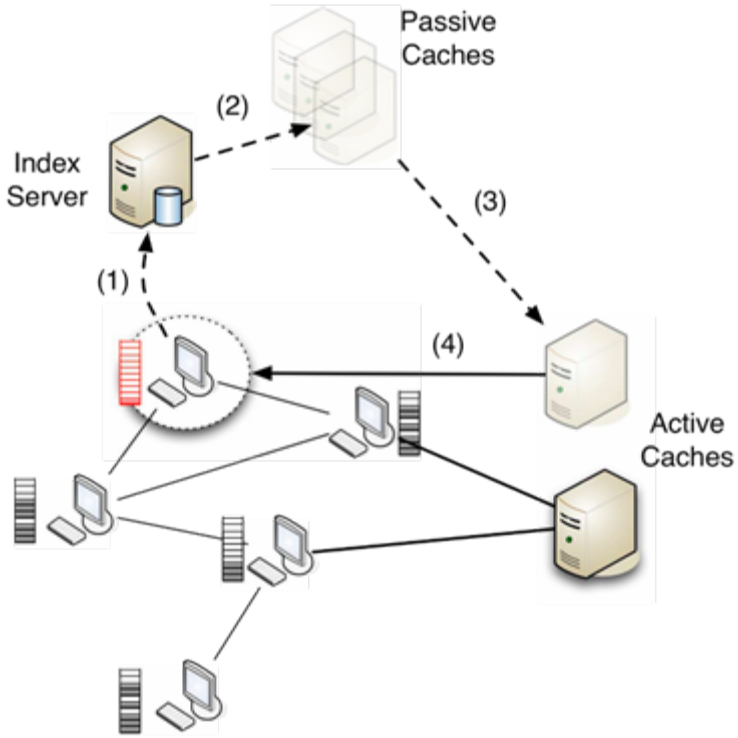
- Keep peers' playout buffer full
  → no stalling, fast startup
- Avoid unnecessary status reports
  - Report only leaking playout buffers
- Avoid bad experience for a minority of peers

If too many peers cannot fill playout buffers for some time

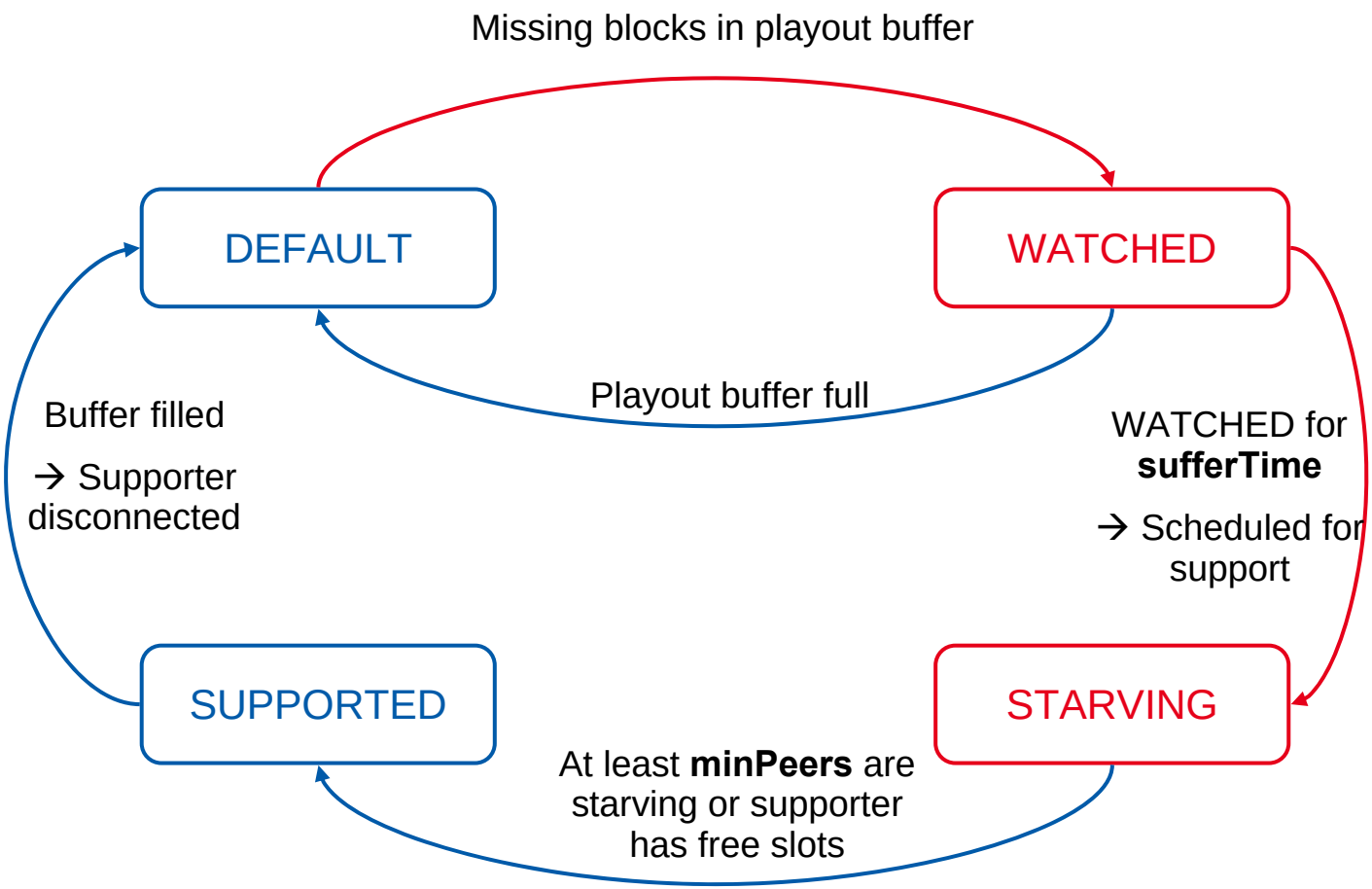
- Allocate servers as supporters

Supporters

- Connect only to suffering peers
- Serve them until they recover



Downloader States



Evaluation

TECHNISCHE UNIVERSITÄT DARMSTADT

BitTorrent simulator by Bharambe et al. [Bharambe2006]

- Highly scalable
- Fair-share underlay model

Additionally implemented:

- Give-to-Get [Mol2008] as underlying streaming protocol
- Adaptive policies: Global Speed and Supporter
- Static policies for comparison

Assess the server contribution and user performance

- Startup and stalling times for fulfilled QoE requirements incl. outliers
  - 50th and 95th percentiles
- Server load (uploaded data)

Basic Scenario

TECHNISCHE UNIVERSITÄT DARMSTADT

Short videos with variable popularity and session durations

- Applicable for UG content (like YouTube), trailers, news …
- Video: bitrate = 512 kbps, duration = 5 minutes
- 10 seconds playout buffers

Server dimensioning

- Up to 10 (virtual) servers
- 2mbps upload capacity

Peer capacities:

- 200 peers, 3 groups (30, 50, and 20% of peers)
- 256, 512, 1024 upload
- 2 mpbs download

Peer behavior

- Exponential arrival rate (6 peers per second)
- Departure time: ~50% video length on average

Global Speed Policy Performance

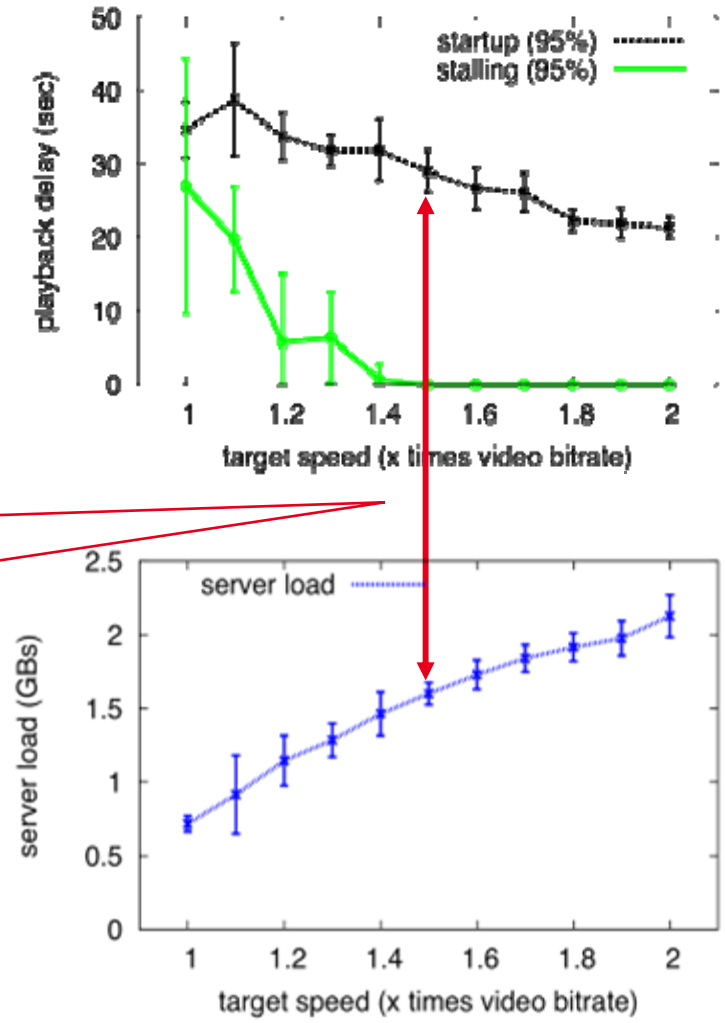
TECHNISCHE UNIVERSITÄT DARMSTADT

How to configure the target speed?

- Too low → bad user experience
- Too high → unnecessary server load

Operating point:

target speed = 1.5*video bitrate

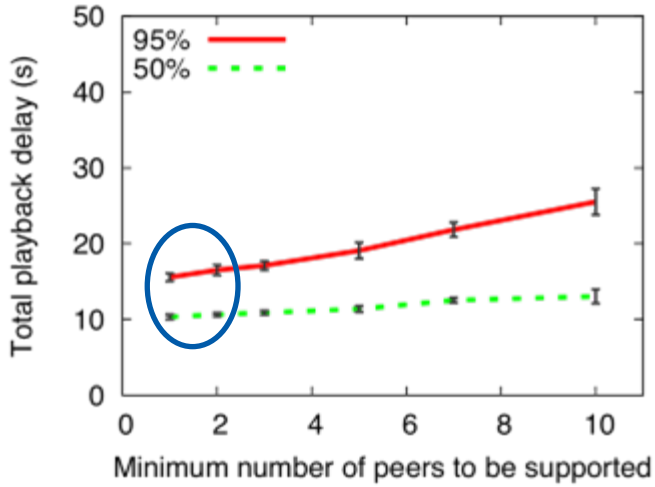


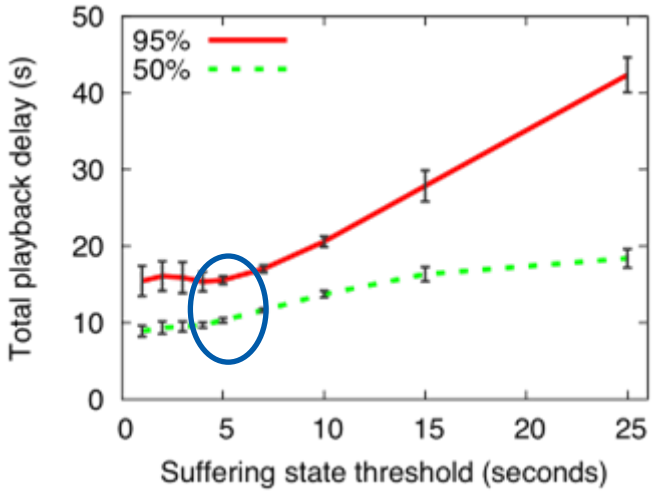
Supporter Policy Performance

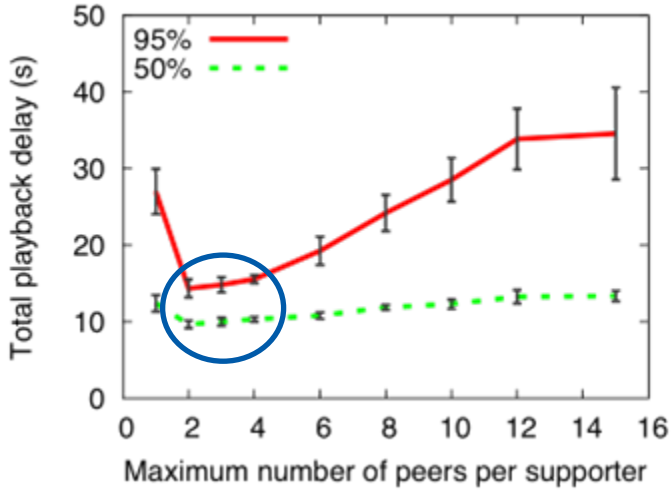
TECHNISCHE UNIVERSITÄT DARMSTADT

Sensitivity analysis of relevant parameters

- minPeers: nr of suffering peers to allocate new servers
- maxPeers: to take care per supporter
- sufferTime: when a peer really needs help







Policy Comparison (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Can adaptive policies compete with perfect allocation?

Static server allocation

- Popularity-based, predicted or manual
- Variable setups

Global speed

- Target speed = 1.5*bitrate

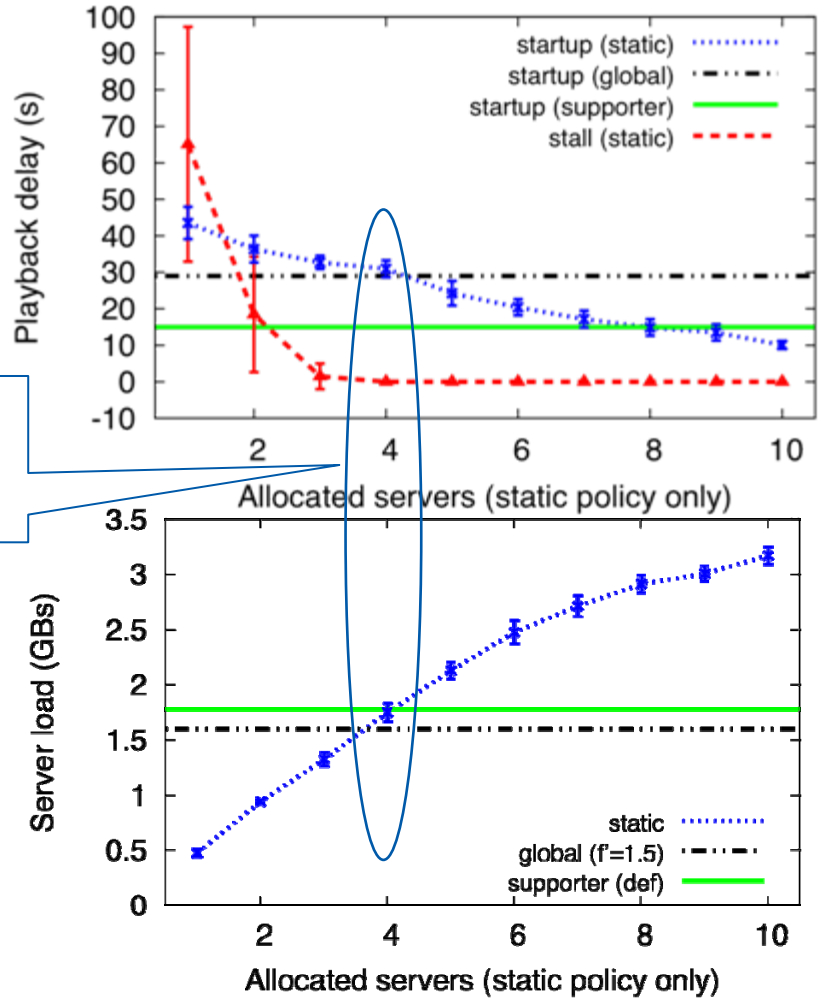
Supporter

- Default configuration

Comparable performance

- For the best static and global speed policies

Supporter policy is more efficient (startup)



Policy Comparison (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Policy	Server load	Stalling (95%)	Startup (95%)	Startup (50%)
Static (best)	1,75 GB	0s	31.0s	12,2s
Global	1,60 GB	0s	29.0s	10,2s
Supporter	1,78 GB	0s	15,5s	10,3s

Observations

- Adaptive policies allow to meet streaming quality requirements
- Median performance similar to best static allocation
- Focusing on starving peers eliminates most of the outliers (→ Supporter policy)

Summary

TECHNISCHE UNIVERSITÄT DARMSTADT

Server allocation policies for peer-assisted VoD

- Guaranteed user performance
- Maximized peer contribution → minimized server load

Proposed policies

- Global Speed → focus on average swarm performance
- Supporter → focus on playout buffers and outliers

Parameter study and comparison with static policies

- Adaptive allocation compete with perfect prediction or complement them
- Supporter policy is more efficient in outlier elimination

Next Steps

TECHNISCHE UNIVERSITÄT DARMSTADT

Additional evaluations

- Large videos, more peers, diurnal traffic pattern
- Overhead measurements

Prototype implementation based on the Tribler client:

- Performance signaling
- Connection management
- Upload policy of the server

Additional mechanisms

- Server allocation among swarms
- Rate allocation for one “super”- server instead of many smaller (complementary)

Thank You! Questions?

TECHNISCHE UNIVERSITÄT DARMSTADT

Department of Electrical Engineering
and Information Technology
Multimedia Communications Lab - KOM

TECHNISCHE UNIVERSITÄT DARMSTADT

Dipl.-Inform. Konstantin Pussep

Konstantin.Pussep@KOM.tu-darmstadt.de
Rundeturmstr. 10
64283 Darmstadt
Germany

Phone +49 (0) 6151/165188
Fax +49 (0) 6151/166152
www.kom.tu-darmstadt.de