

Adding Capacity-Aware Storage Indirection to Homogeneous Distributed Hash Tables

Philip Wette
Computer Networks Group, University of Paderborn

Kalman Graffi
Technology of Social Networks Group, University of Düsseldorf

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB901
ON - THE - FLY COMPUTING

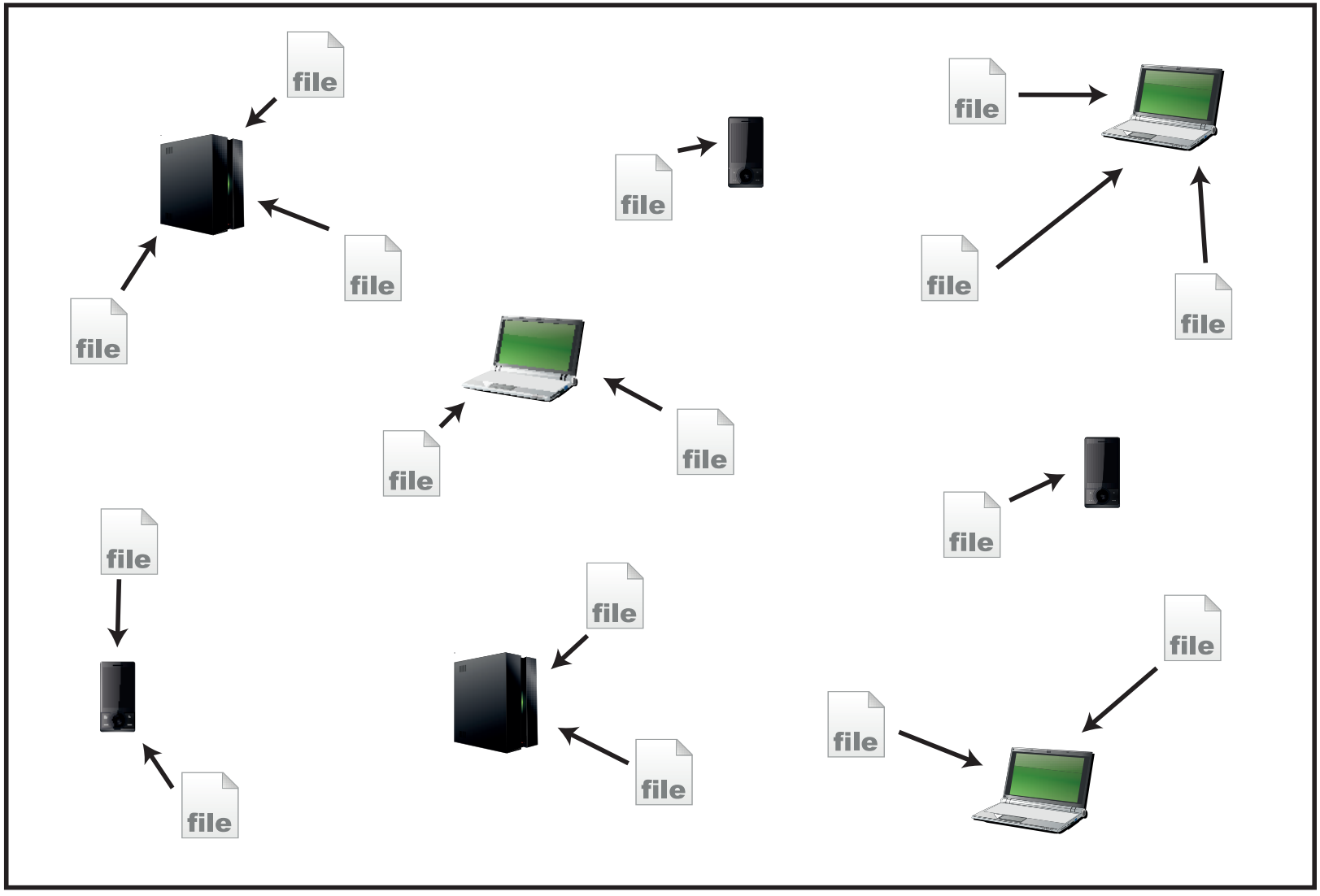
What is a Peer-to-Peer network?

- Loosely coupled group of equally treated computers (peers)
- Peers share their resources
- Service hosted at home computers
- Decentralized system $\Rightarrow$ high availability

In this talk: *Distributed Hash Tables* (DHT) used for file sharing
(P2P Online Social Networks; Distributed filesystem)

Motivation

Prototype of a Distributed Hash Table



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Motivation

Why Loadbalancing?

Why do DHTs need loadbalancing?

- Peers and files are equally distributed in the identifier space
- $\Rightarrow$ in average, each peer serves an equal amount of files

But:

- File sizes are heterogeneous
- File popularities are heterogeneous
- Resources required for hosting a file $\sim$ file size $\cdot$ file popularity

Real World observations:

- A small group of files responsible for largest amount of load
  - In WWW: file popularity is Zipf distributed (heavy tailed)
- A small group of peers own largest amount of resources
  - University networks: 1 Gbps upload capacity

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB901
ON-THE-FLY COMPUTING

Motivation

Why Loadbalancing?

Why do DHTs need loadbalancing?

- Peers and files are equally distributed in the identifier space
- $\Rightarrow$ in average, each peer serves an equal amount of files

But:

- File sizes a
- File popula
- Resources opularity

Idea

Let “strong” peers host “hot” files

Real World observations:

- A small group of files responsible for largest amount of load
  - In WWW: file popularity is Zipf distributed (heavy tailed)
- A small group of peers own largest amount of resources
  - University networks: 1 Gbps upload capacity

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Motivation

Goal

Goal: Creating a DHT that is aware of both heterogeneous peers and files

Considering

- Different capacities of individual peers
- File sizes and file popularities
- Changing file popularities over time

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Motivation

Goal

Goal: Creating a DHT that is aware of both heterogeneous peers and files

Considering

- Different capacities of individual peers
- File sizes and file popularities
- Changing file popularities over time

- Explained at the showcase of Chord, turning it into **Adaptive-Chord**
- Techniques can be applied to *any* DHT

Overview

1 Motivation

2 Adaptive load balancing for DHTs
- Idea
- Load balancing

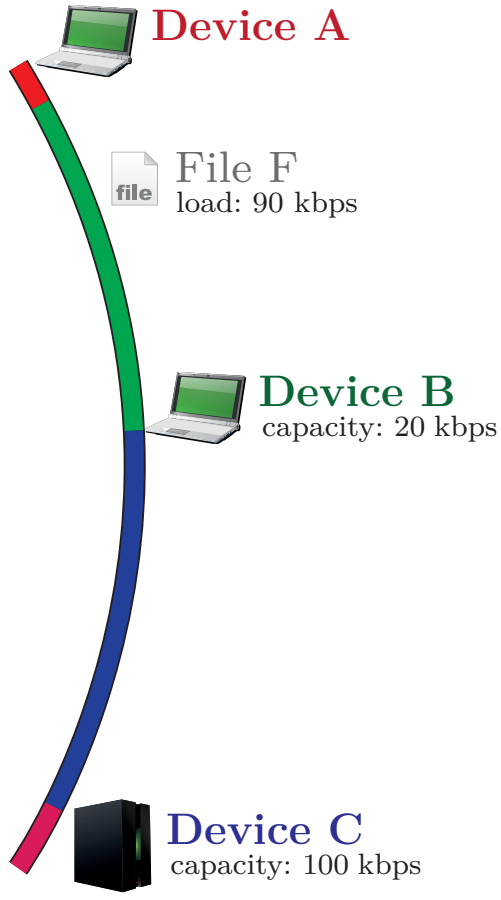
3 Evaluation
- Scenarios
- Results

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Adaptive-Chord

Idea: Decoupling of Responsibility and Supply

- A peer can supply files it is not responsible for: Creating **mirrors** in the network

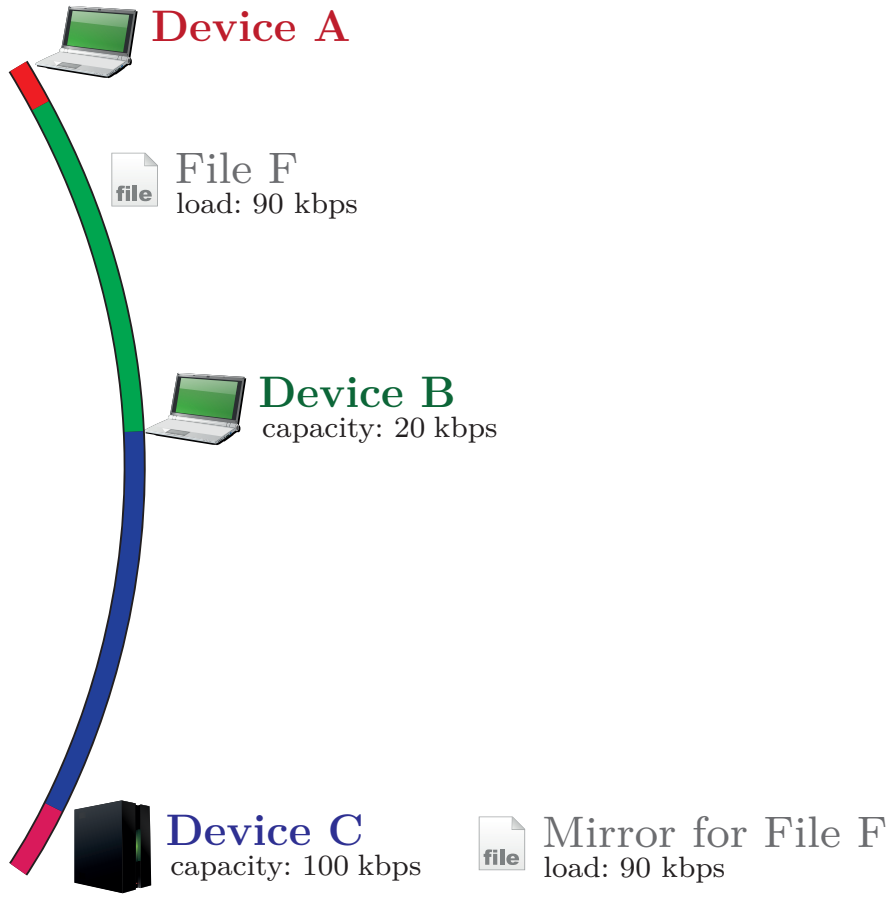


UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON-THE-FLY COMPUTING

Adaptive-Chord

Idea: Decoupling of Responsibility and Supply

- A peer can supply files it is not responsible for: Creating **mirrors** in the network



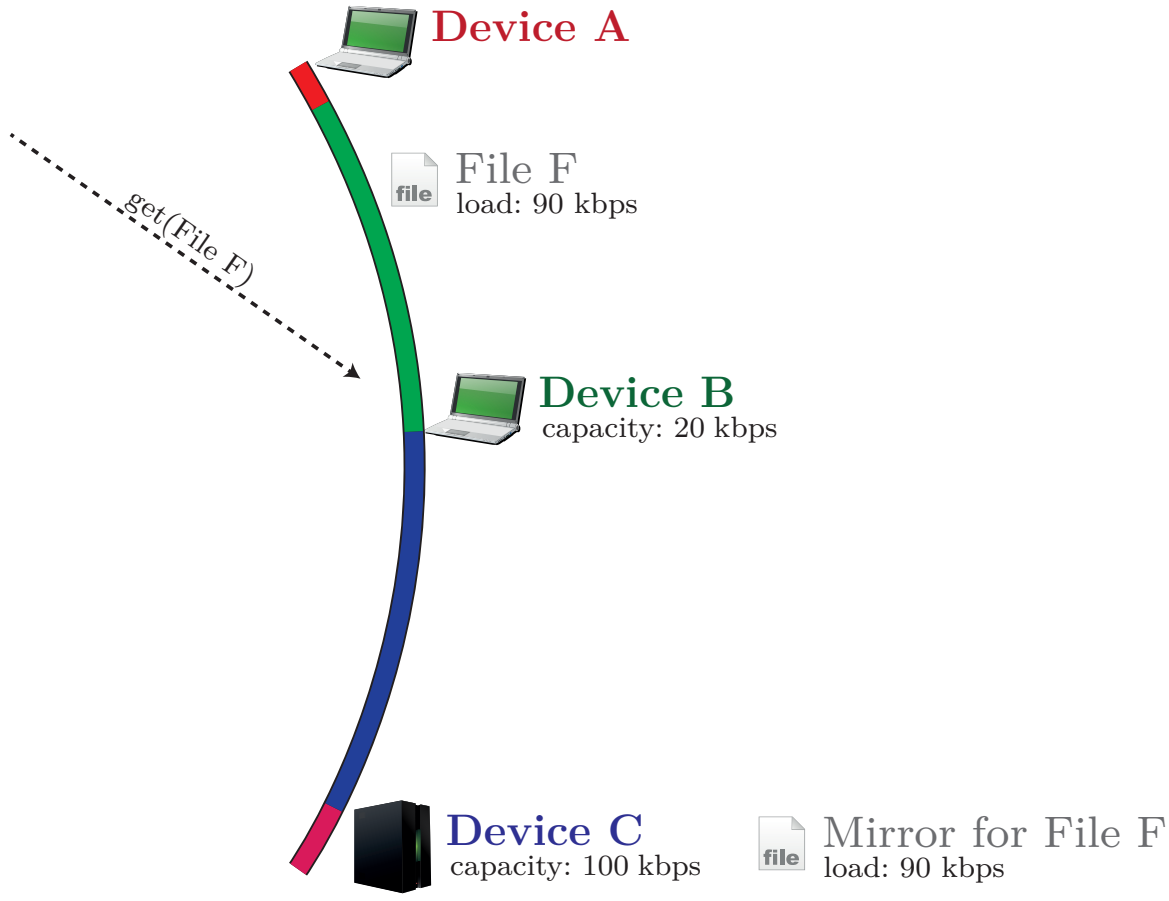
UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON-THE-FLY COMPUTING

Adaptive-Chord

Adaptive load balancing for DHTs

Idea: Decoupling of Responsibility and Supply

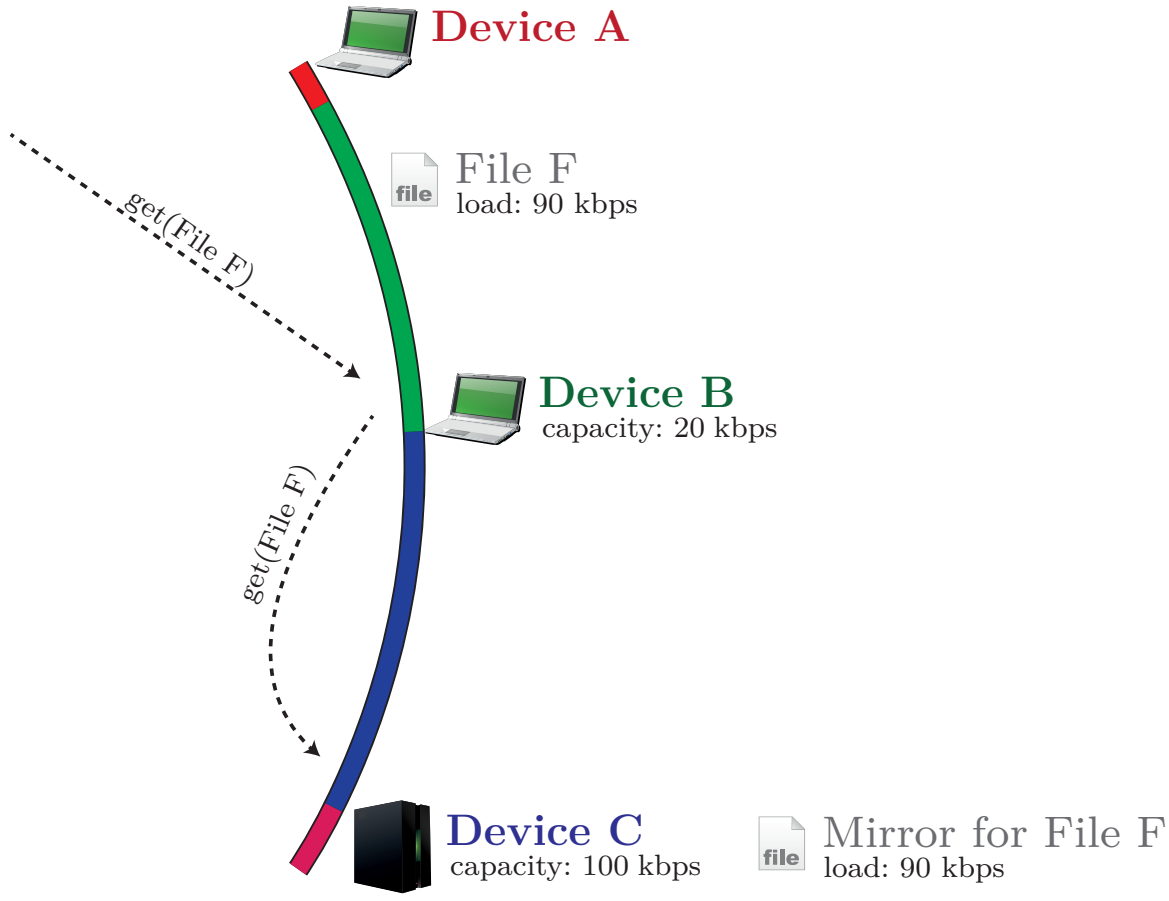
- A peer can supply files it is not responsible for: Creating **mirrors** in the network



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

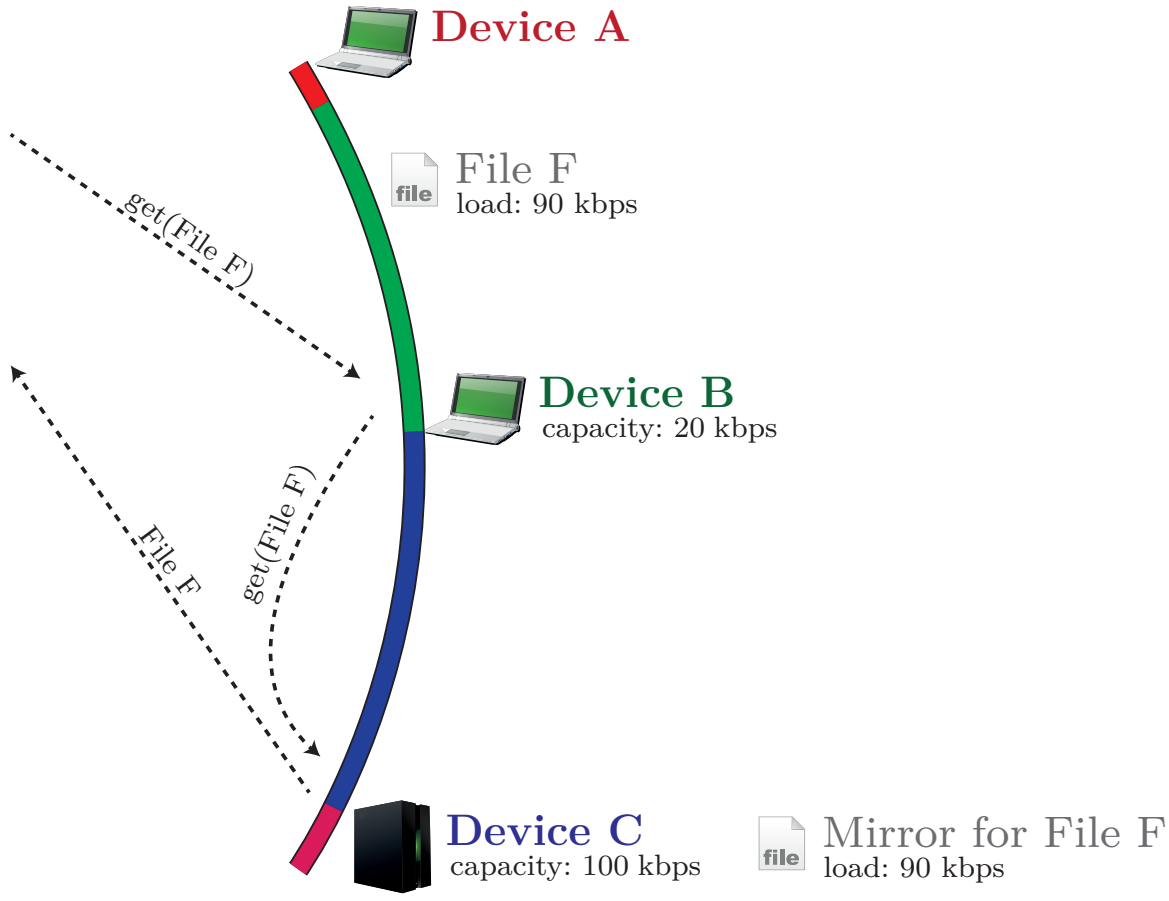
Idea: Decoupling of Responsibility and Supply

- A peer can supply files it is not responsible for: Creating **mirrors** in the network



Idea: Decoupling of Responsibility and Supply

- A peer can supply files it is not responsible for: Creating **mirrors** in the network

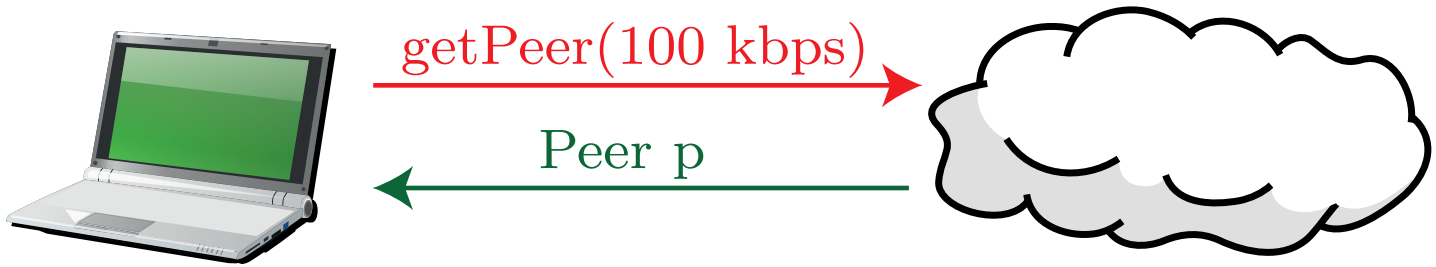


UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON-THE-FLY COMPUTING

Idea: How to find a peer to host a mirror?

Main Problem:

Given a file, how to find a peer capable of hosting a mirror without getting overloaded?

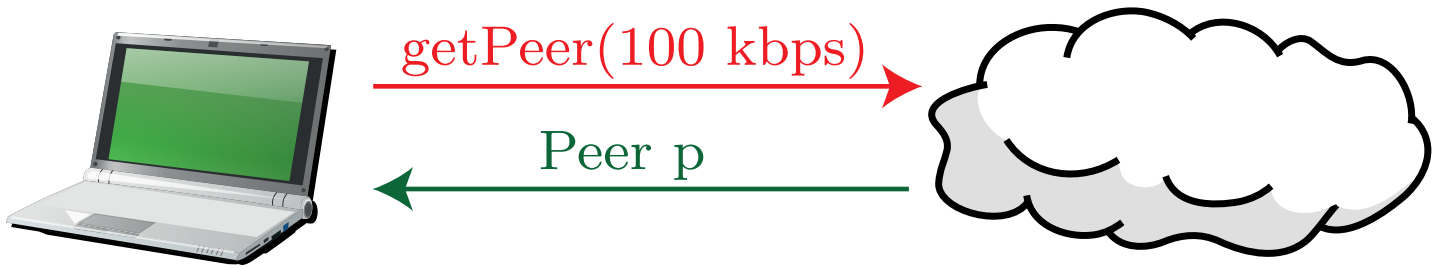


Adaptive-Chord

Idea: How to find a peer to host a mirror?

Main Problem:

Given a file, how to find a peer capable of hosting a mirror without getting overloaded?



Solution:

A *second* P2P to address peers based on their *free resources*: the **Capacity overlay**

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON-THE-FLY COMPUTING

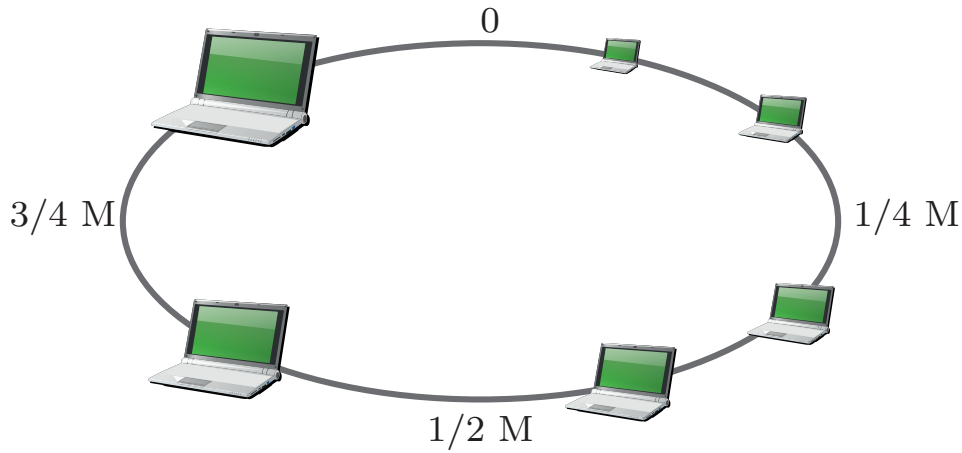
Adaptive-Chord

Capacity overlay

Idea: Capacity overlay

- Quickly find peers based on their free resource count
- Use slightly modified Chord (works with any P2P using 1D keyspace)
- Peer identifier reflects free resource count
- Sorts peers based on free resource count

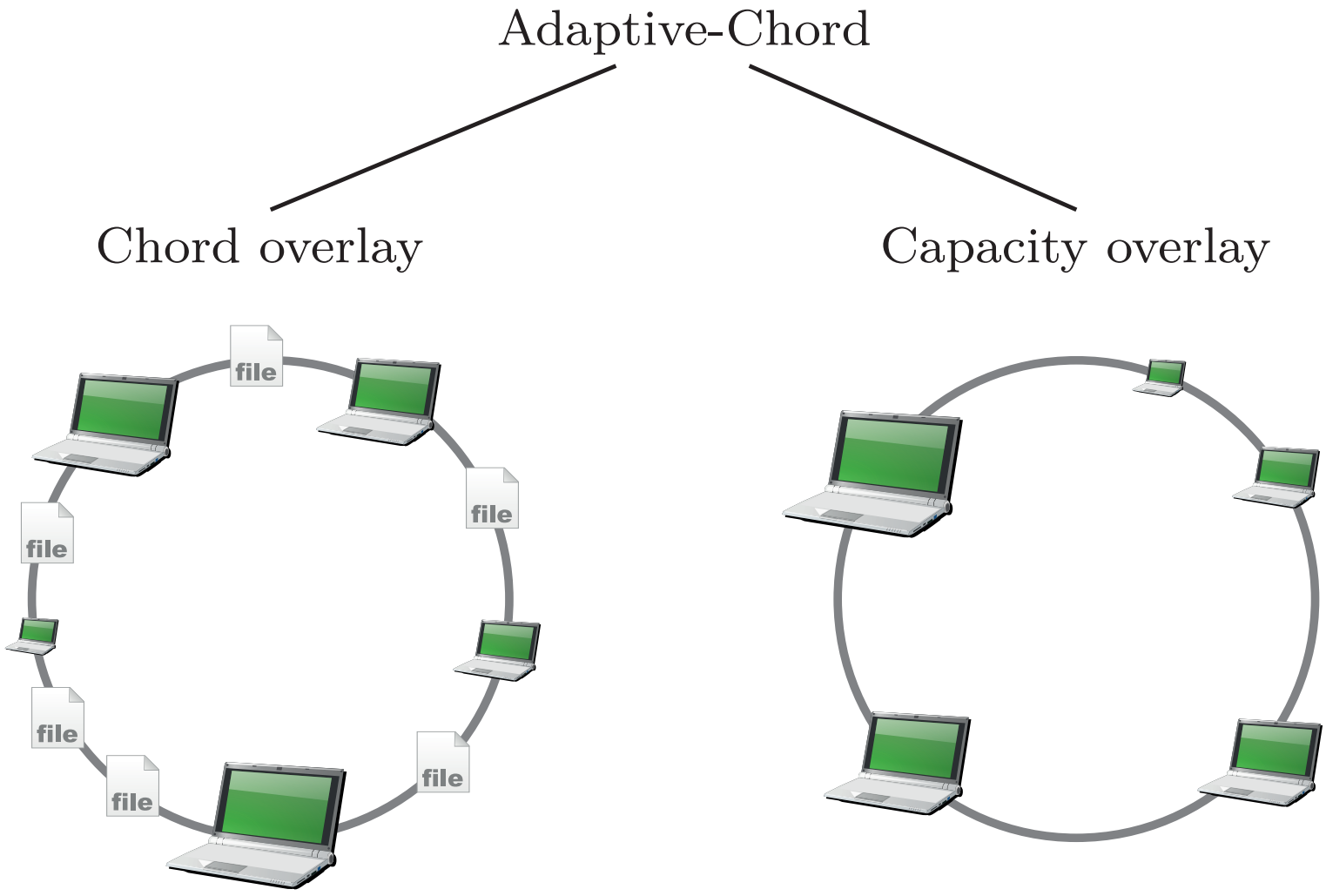
$$ID_K = (2^m - 1) \cdot \min(1, \frac{v}{M})$$



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON-THE-FLY COMPUTING

Adaptive-Chord

System Architecture



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Overview

1 Motivation

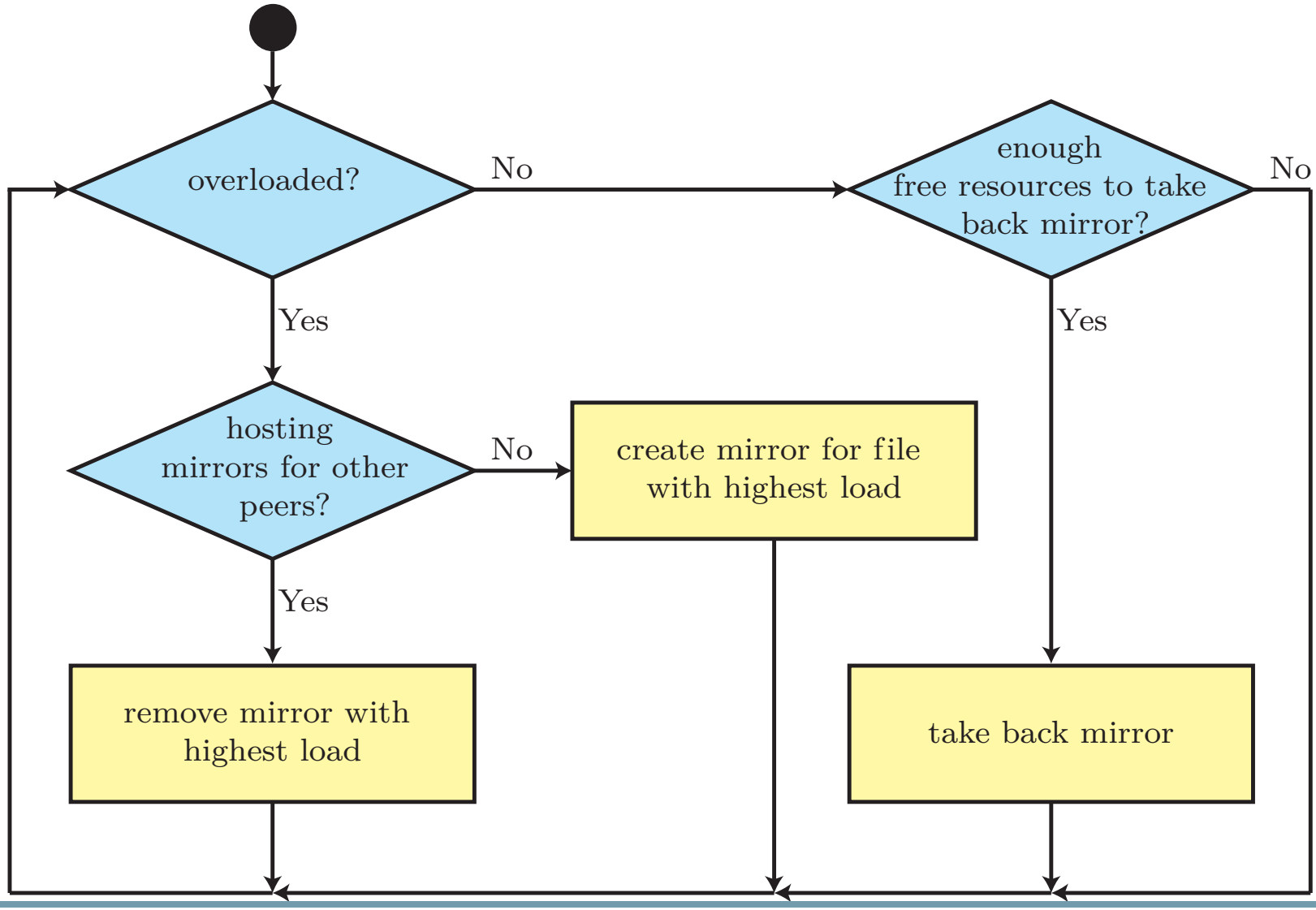
2 Adaptive load balancing for DHTs
- Idea
- Load balancing

3 Evaluation
- Scenarios
- Results

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Adaptive-Chord

Load balancing



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Overview

1 Motivation

2 Adaptive load balancing for DHTs
- Idea
- Load balancing

3 Evaluation
- Scenarios
- Results

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Evaluation

Simulator: PeerfactSim.KOM

PeerfactSim.KOM

- Discrete P2P-Simulator
- Event based
- Layers independently configurable
- Simulation at packet level
- Latency Model: Global Network Positioning ICS
- Bandwidth Model: Last Mile Model
- `http://www.peerfact.org`

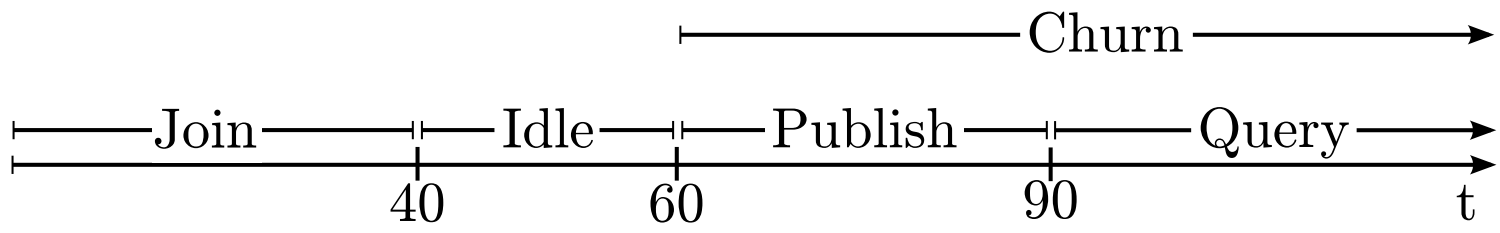
UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Evaluation

Scenarios

Parameters for Simulation

- 1000 Peers simulated with PeerfactSim.KOM
- 24 Hour period
- 10.000 Documents
- Uniformly distributed file sizes 0 – 10 MB
- Zipf distributed file popularities
- with / without Churn
- high / low number of queries (10 min / 30 min inter query time)
- unstable / stable file popularity patterns (10 min / 30 min pop. change)



UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Overview

1 Motivation

2 Adaptive load balancing for DHTs
- Idea
- Load balancing

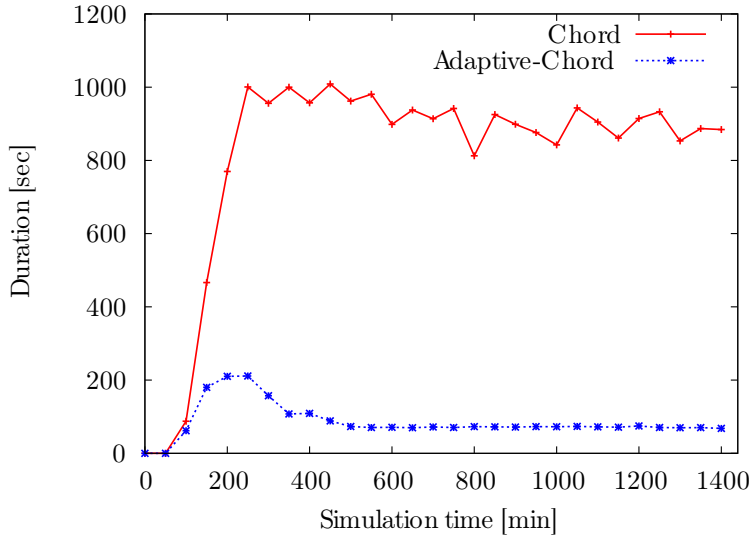
3 Evaluation
- Scenarios
- Results

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Evaluation

Szenario: High load - unstable popularities - no churn

Average download duration:



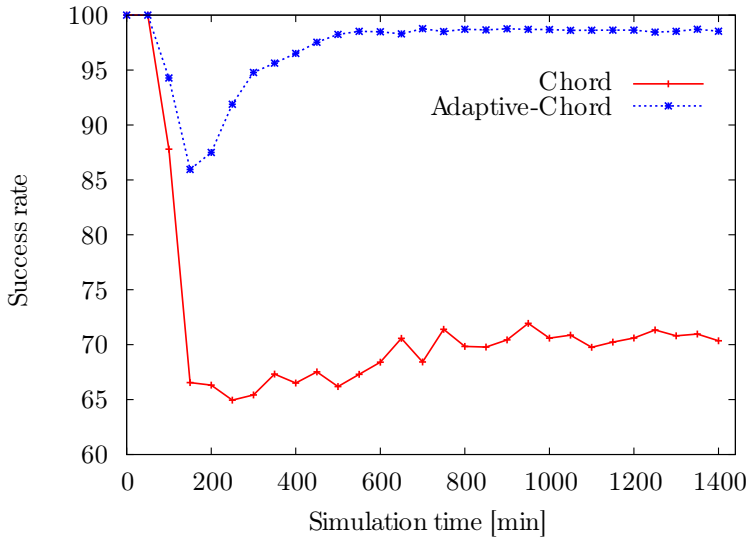
- Average download up to 9 times faster when using Adaptive-Chord

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

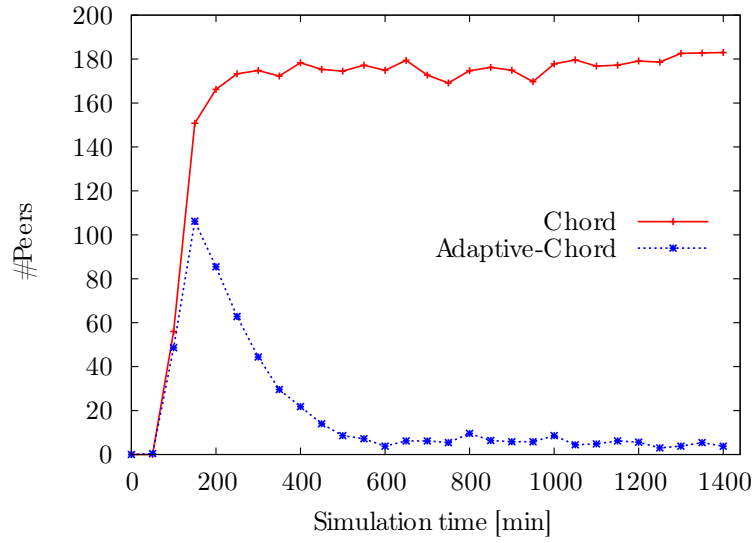
Evaluation

Szenario: High load - unstable popularities - no churn

Success rate:



Overloaded Peers:

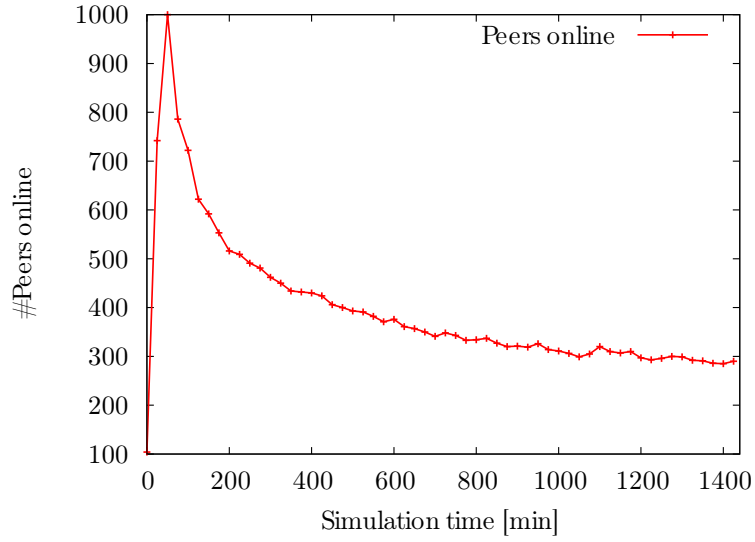


- To model user's impatience, a download is aborted when it's data rate falls below 1 kbps
- Nearly 100% success rate in Adaptive-Chord
- A peer is overloaded when more than 95% of its resources are consumed

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Evaluation
Churn

- Creating more realistic simulation by considering **churn**
- No file replication in Chord: Peer offline = files are no longer available

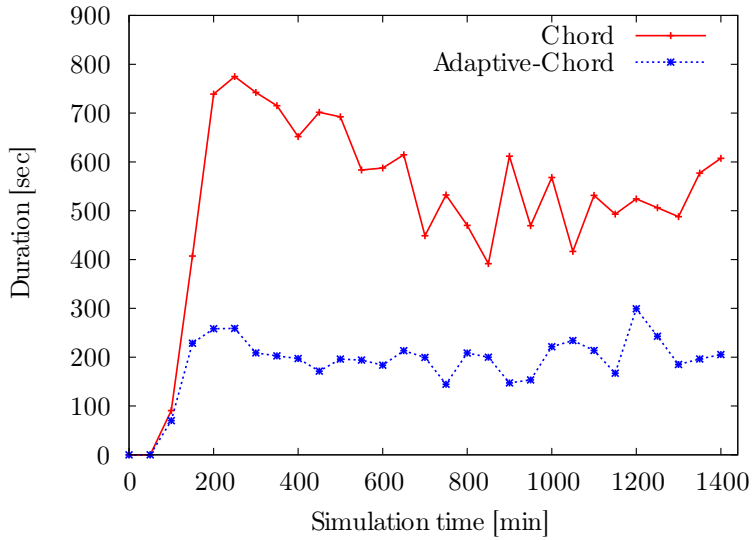


UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Evaluation

Szenario: High load - unstable popularities - churn

Average download duration:



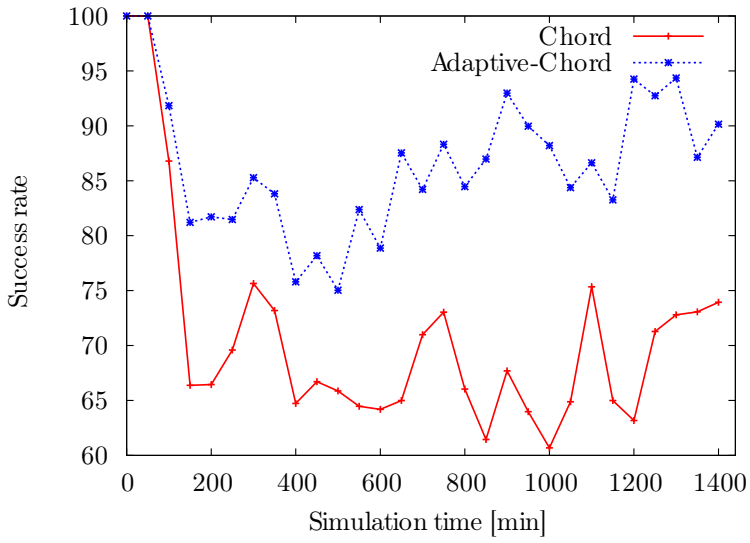
- Still significantly shorter download durations
- Average data rates higher for Adaptive-Chord

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

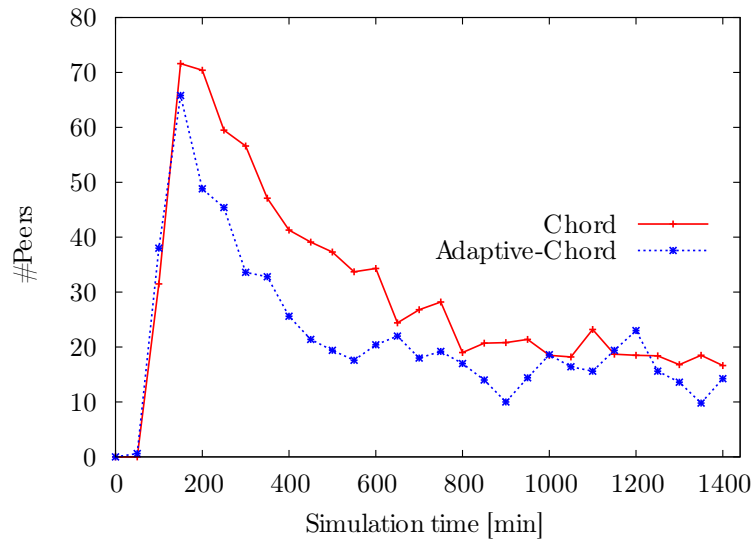
Evaluation

Szenario: High load - unstable popularities - churn

Success rate:



Overloaded peers:



- Higher success rate in Adaptive-Chord
- Overall load much lower than for nochurn scenarios because of offline peers

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING

Conclusion

- DHTs are not designed to handle heterogeneous files
- Largest amount of load is created by small group of files
- Small number of peers own most resources
- When creating mirrors in a DHT, a function is required to address peers by free capacity

- We propose a second P2P aside to accomplish such an addressing
- We propose a simple load balancing technique based on mirrors

- As a showcase we turned Chord into Adaptive-Chord
- Simulation showed that, even under churn, Adaptive-Chord lowers download time significantly



Philip Wette

Sonderforschungsbereich 901
Universität Paderborn
Fürstenallee 11
33102 Paderborn

http://sfb901.uni-paderborn.de/

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

SFB901
ON - THE - FLY COMPUTING

Evaluation

Handling Non-Uniform Identifiers in Chord

A method for handling non-uniform identifier distribution in Chord:

Thorsten Schütt, Florian Schintke, Alexander Reinefeld:
Chord#: Structured Overlay Network for Non-Uniform Load-Distribution, Technical Report, ZIB, Berlin, 2005.

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB901
ON - THE - FLY COMPUTING

Evaluation

Results

Szenario	v [kbps]	t [s]	# O	success
busy_unstable_nochurn				
Chord	118.6	806.7	158.6	71.8 %
Adaptive-Chord	275.4	85.9	18.2	97.1 %
idle_unstable_nochurn				
Chord	126.5	619.9	76.4	87.7 %
Adaptive-Chord	243.8	72.8	5.3	98.8 %
busy_unstable_churn				
Chord	249.2	514.4	29.4	38.1 %
Adaptive-Chord	379.8	186.1	21.8	43.6 %
idle_unstable_churn				
Chord	207.8	370.3	12.7	43.2 %
Adaptive-Chord	333.7	159.1	11	46.8 %

v data rate

t duration of a download

O Number of overloaded peers

success success rate

UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
SFB 901
ON - THE - FLY COMPUTING