Optimal File Distribution in Peer-to-Peer Networks

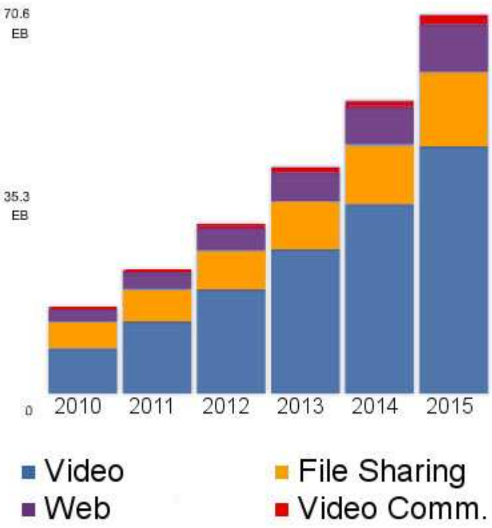
Kai-Simon Goetzmann

Technische Universität Berlin
goetzmann@math.tu-berlin.de

joint work with Tobias Harks, Max Klimm, and Konstantin Miller

ISAAC 2011, Yokohama

Motivation



*Cisco VNI June 2011

Question:

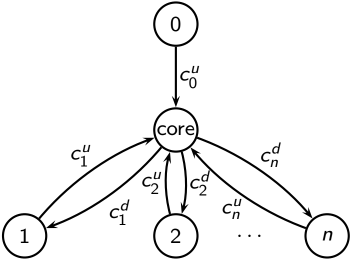
How can peer-to-peer downloads be scheduled optimally?

Outline

- Part I: Problem Statement
- Part II: Hardness
- Part III: Approximation

Part I

Problem Statement

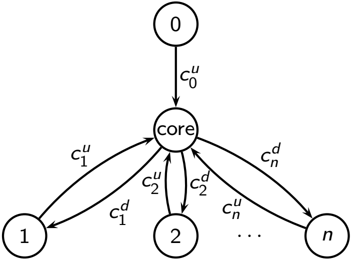


Instance $I = (N, c_N^d, c_N^u)$

- set $N = \{1, \ldots, n\}$ of peers
- peer 0 (server) owns file of unit size at time 0
- download capacity c_i^d of peer i
- upload capacity c_i^u of peer i

- $s_{i,j} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ (integrable) sending rate function
- $x_i(t) = \int_0^t \sum_{k \in N} s_{k,i}(\tau) d\tau$ fraction of file owned by i at t

Problem Statement



Instance $I = (N, c_N^d, c_N^u)$

- set $N = \{1, \ldots, n\}$ of peers
- peer 0 (server) owns file of unit size at time 0
- download capacity c_i^d of peer i
- upload capacity c_i^u of peer i

- $s_{i,j} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ (integrable) sending rate function
- $x_i(t) = \int_0^t \sum_{k \in N} s_{k,i}(\tau) d\tau$ fraction of file owned by i at t

Symmetric peers: $c_i^d = c_i^u =: c_i$ for all $i \in N$.

Feasible solution

Feasible solution $S = (s_{i,j})_{i,j \in N}$

- $s_{i,j} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is an integrable function
- unique sender: $\forall i \in N \; \exists p(i) \in N : s_{k,i} \equiv 0 \quad \forall k \neq p(i)$
- download completed:
  $s_{i,j}(t) = 0 \quad \forall i, j \in N \setminus \{0\}, t \in \mathbb{R}_{\geq 0} : x_i(t) d\tau < 1$
- upload capacity: $\sum_{j \in N} s_{i,j}(t) \leq c_i^u \quad \forall i \in N, t \in \mathbb{R}_{\geq 0}$
- download capacity: $s_{p(j),j}(t) \leq c_j^d \quad \forall j \in N, t \in \mathbb{R}_{\geq 0}$

Feasible solution

Feasible solution $S = (s_{i,j})_{i,j \in N}$

- $s_{i,j} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is an integrable function
- unique sender: $\forall i \in N \; \exists p(i) \in N : s_{k,i} \equiv 0 \quad \forall k \neq p(i)$
- download completed:
  $s_{i,j}(t) = 0 \quad \forall i, j \in N \setminus \{0\}, t \in \mathbb{R}_{\geq 0} : x_i(t) d\tau < 1$
- upload capacity: $\sum_{j \in N} s_{i,j}(t) \leq c_i^u \quad \forall i \in N, t \in \mathbb{R}_{\geq 0}$
- download capacity: $s_{p(j),j}(t) \leq c_j^d \quad \forall j \in N, t \in \mathbb{R}_{\geq 0}$

Definition

completion time: $C_i = \inf\{t \in \mathbb{R}_{\geq 0} : x_i(t) = 1\}$
makespan: $M = \max_{i \in N} C_i$

Part II

Hardness

Capacity Expansion Lemma

For a peer i and a time interval $[t_1, t_2]$,

- total upload of i in $[t_1, t_2]$: $u_i(t_1, t_2) = \int_{t_1}^{t_2} \sum_{j \in N} s_{i,j}(\tau) d\tau$
- idleness of i in $[t_1, t_2]$:
  $z_i(t_1, t_2) = c_i(t_2 - \max\{C_i, t_1\}) - u_i(t_1, t_2)$

For $t \geq 0$,

- $X(t) = \sum_{i \in N} x_i(t)$
- $Z(t) = \sum_{i \in N} z_i(0, t)$

Lemma (Capacity Expansion Lemma)

Let I be an instance with $c = \max_{i \in N} c_i$, Then, for all solutions

1. $u_i(t_1, t_2) + z_i(t_1, t_2) \leq \max\{0, (t_2 - t_1)c_i - 1 + x_i(t_1)\}$
2. $X(k/c) + Z(k/c) \leq 2^k$ *for all* $k \in \mathbb{N}$

▸ Proof

Theorem

The peer-to-peer file distribution problem for heterogeneous symmetric peers is strongly NP-hard.

Proof.

By reduction from 3-PARTITION. ▸ Details □

Part III

Approximation

Algorithmic Idea

- need combinatorial lower bound
  - $u_i(t_1, t_2) + z_i(t_1, t_2) \leq \max\{0, (t_2 - t_1)c_i - 1 + x_i(t_1)\}$
- Algorithmic Idea: let algorithm satisfy this inequality for all peers with $x_i(t_1) > 0$ with equality
  - Every peer downloads at full download rate
    $\Rightarrow$ complete peers in order $1, \ldots, n$ where $c_1 \geq \cdots \geq c_n$
  - no idleness as long not every download started
- Problem: This might not be possible
- Solution: Time-varying resource augmentation
  - allow to violate upload and download constraints by a factor of $\sqrt{2}$

Scale-Fit Algorithm

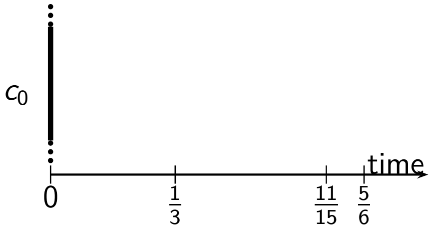
- Sort peers such that $c_1 \geq \cdots \geq c_n$
  (Peer 0 has unused upload capacity c at time 0)
- Consider next point in time t where some peer i has unused upload capacity c
  - Choose downloaders: Choose next k peers $i_1, \ldots, i_k$ such that $c/\sqrt{2} \leq \sum_{j=i_1,\ldots,i_k} c_j \leq \sqrt{2}c$
  - Resource augmentation:
    - If $c < \sum c_j$, augment upload capacity s.t. $c = \sum c_j$
    - If $\sum c_j < c$, augment download capacities s.t. $c = \sum c_j$
  - Peers $j = i_1, \ldots, i_k$ download from i with full capacity
  - Release capacity: For $j = i_1, \ldots, i_k$, at time $t + \frac{1}{\alpha c_j}$
    - Peer j has unused capacity c_j
    - Peer i has unused capacity $c_j \cdot c / \sum_{j'} c_{j'}$
- When all downloads are finished, stop.

Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.

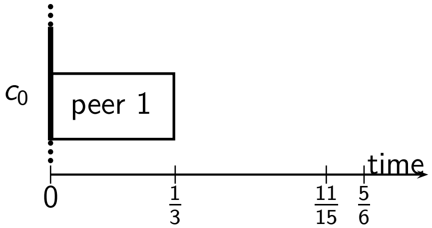
Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



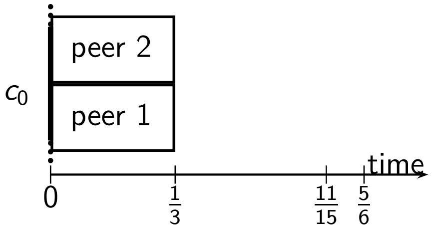
Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



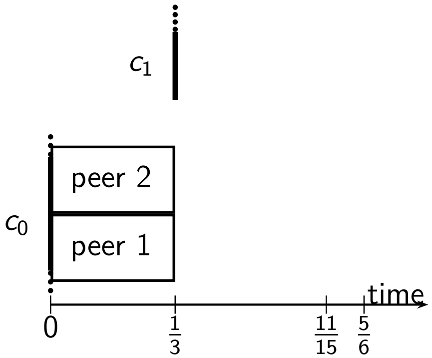
Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



Example

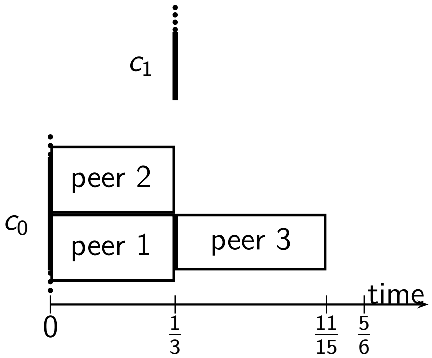
$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



Scale-Fit Example

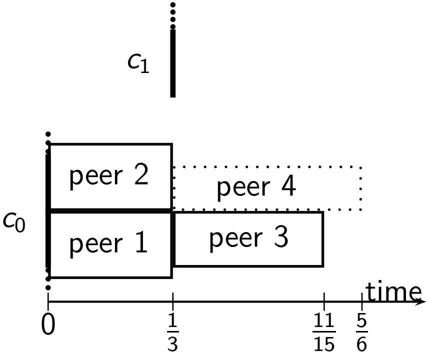
Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



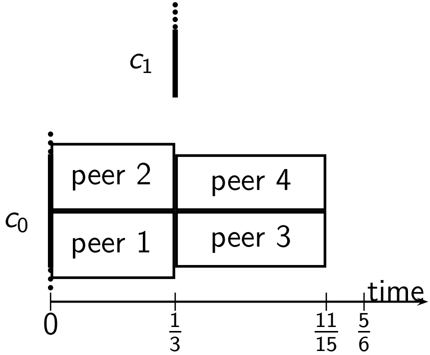
Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



Example

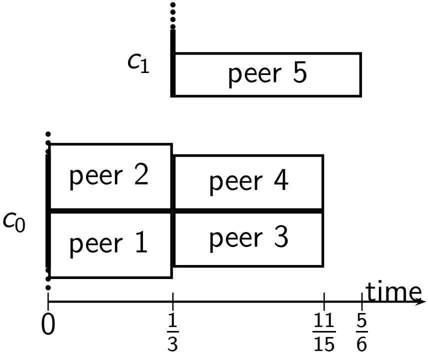
$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.



Scale-Fit Example

Example

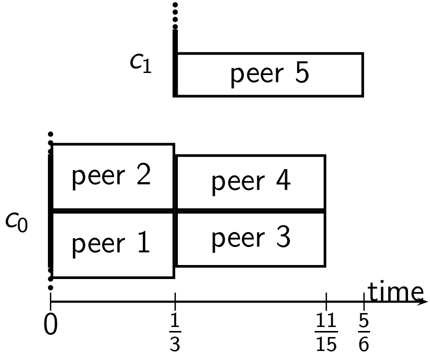
$c_0 = 5,\ c_1 = 3,\ c_2 = 3,\ c_3 = 5/2,\ c_4 = 2,$ and $c_5 = 2.$

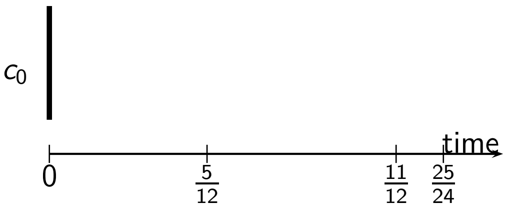


Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.

Scaling factor: $5/4$

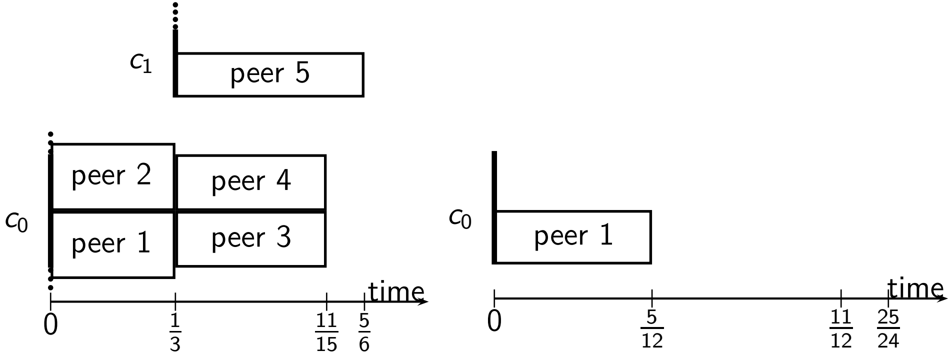




Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.

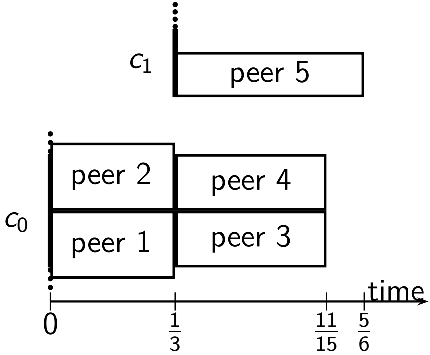
Scaling factor: $5/4$

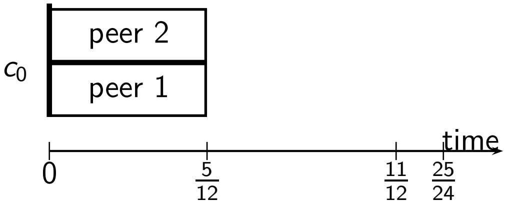


Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.

Scaling factor: $5/4$

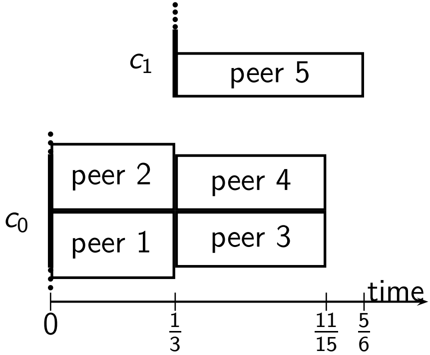


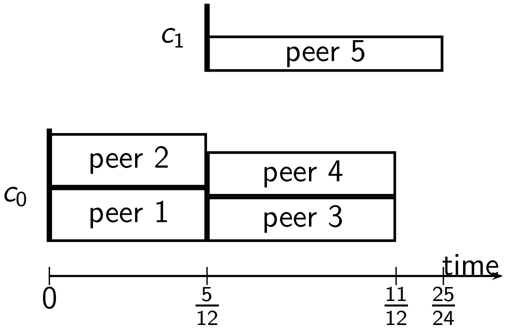


Example

$c_0 = 5$, $c_1 = 3$, $c_2 = 3$, $c_3 = 5/2$, $c_4 = 2$, and $c_5 = 2$.

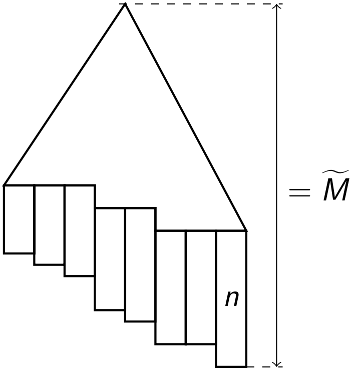
Scaling factor: $5/4$





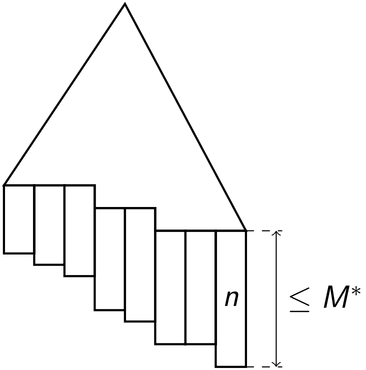
Theorem

If $c_0 \geq c_i\ \forall i \in N$, then SCALE-FIT *is a $2\sqrt{2}$-approximation.*



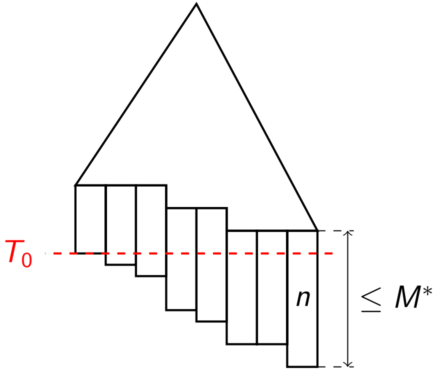
Theorem

If $c_0 \geq c_i \; \forall i \in N$*, then* SCALE-FIT *is a* $2\sqrt{2}$*-approximation.*



Theorem

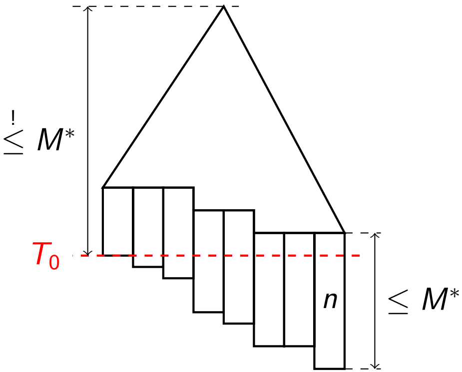
If $c_0 \geq c_i \; \forall i \in N$, then Scale-Fit *is a $2\sqrt{2}$-approximation.*



Analysis of Scale-Fit I

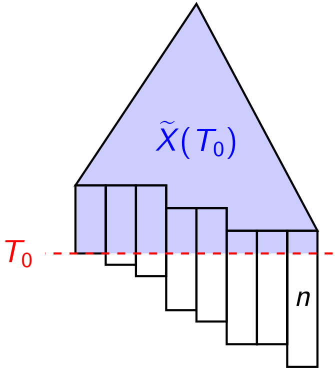
Theorem

If $c_0 \geq c_i \; \forall i \in N$, then Scale-Fit *is a $2\sqrt{2}$-approximation.*



Theorem

If $c_0 \geq c_i \; \forall i \in N$, then SCALE-FIT *is a $2\sqrt{2}$-approximation.*

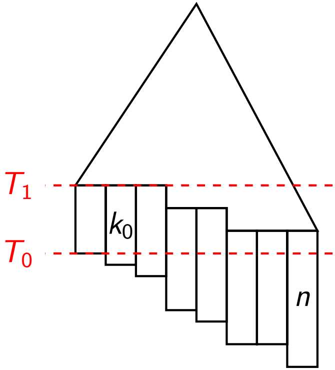


- Assume $X^*(T_0) = n > \widetilde{X}(T_0)$

Analysis of Scale-Fit I

Theorem

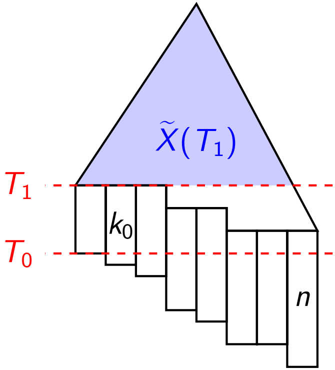
If $c_0 \geq c_i \; \forall i \in N$, then Scale-Fit *is a* $2\sqrt{2}$*-approximation.*



- Assume $X^*(T_0) = n > \widetilde{X}(T_0)$
- $T_1 = T_0 - 1/c_{k_0}$

Theorem

If $c_0 \geq c_i \; \forall i \in N$, then Scale-Fit *is a $2\sqrt{2}$-approximation.*



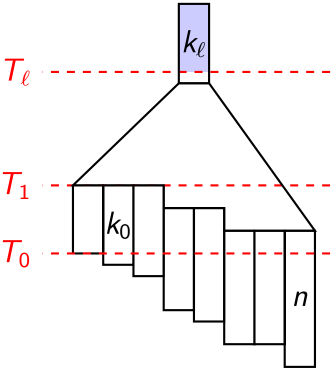
- Assume $X^*(T_0) = n > \widetilde{X}(T_0)$
- $T_1 = T_0 - 1/c_{k_0}$
- $\widetilde{X}(T_1) < X^*(T_1)$

$\vdots$

Analysis of Scale-Fit I

Theorem

If $c_0 \geq c_i \ \forall i \in N$, then Scale-Fit *is a $2\sqrt{2}$-approximation.*



- Assume $X^*(T_0) = n > \widetilde{X}(T_0)$
- $T_1 = T_0 - 1/c_{k_0}$
- $\widetilde{X}(T_1) < X^*(T_1)$

  $\vdots$

- $\widetilde{X}(T_\ell) < X^*(T_\ell)$, $k_\ell = 1$
  Contradiction!

Theorem

For the peer-to-peer file distribution problem with heterogeneous symmetric peers, the following holds

1. *If* $c_0 \geq c_1$, SCALE-FIT *is a* $2\sqrt{2}$*-approximation*
2. *If* $c_0 < c_1$, *uploading the file to peer* 1 *and applying* SCALE-FIT *is a* $(1 + 2\sqrt{2})$*-approximation.*

Conclusion

- Optimal peer-to-peer file distribution with heterogenous symmetric peers is strongly NP-hard
- There is a $(1 + 2\sqrt{2})$-approximation

Not in this talk:

- Homogenous symmetric peers are easy (even if the server has a different capacity)
- Fixed number of peers is easy

Open problems:

- better approximation factor, PTAS, asymmetric peers, restricted connectivity, file divided in chunks...

Conclusion

- Optimal peer-to-peer file distribution with heterogenous symmetric peers is strongly NP-hard
- There is a $(1 + 2\sqrt{2})$-approximation

Not in this talk:

- Homogenous symmetric peers are easy (even if the server has a different capacity)
- Fixed number of peers is easy

Open problems:

- better approximation factor, PTAS, asymmetric peers, restricted connectivity, file divided in chunks...

Thank you for your attention!


THE 21ST INTERNATIONAL SYMPOSIUM ON MATHEMATICAL PROGRAMMING
ISMP
AUGUST 19-24
ismp2012.org
BERLIN 2012

Capacity Expansion Lemma - Proof

$u_i(t_1, t_2) + z_i(t_1, t_2) \leq \max\{0, (t_2 - t_1)c_i - 1 + x_i(t_1)\}$

Proof:

- peer i needs at least $(1 - x_i(t_1))/c_i$ time units to finish download
- $u_i(t_1, t_2) + z_i(t_1, t_2) \leq c_i(t_2 - t_1 - (1 - x_i(t_1))/c_i)$
  $\leq \max\{0, (t_2 - t_1)c_i - 1 + x_i(t_1)\}$

$X(k/c) + Z(k/c) \leq 2^k$ for all $k \in \mathbb{N}$

Proof:

- By induction over k, using the first inequality

Back

Hardness Proof I

- 3-Partition instance $P = \{k_1, \ldots, k_n\}$, $n = 3m$, $B = \sum_{i=1}^{n} k_i/3$, w.l.o.g. $m = 2^\ell$
- Construct instance I of peer-to-peer distribution problem
  - m subset peers $s_1, \ldots, s_m$ with $c_{s_i} = B$
  - s_1 owns the file at time 0
  - for each k_i
    - a master element peer p_0^i with $c_{p_0^i} = k_i$
    - $2\ell k_i - 2$ element peers $p_1^i, \ldots, p_{2\ell k_i - 2}^i$ with $c_{p_k^i} = k_i/(\ell k_i - 1)$
- P is a yes-instance $\Leftrightarrow M^* \leq \ell/B + \ell$

Hardness Proof II

"⇒":

Example

$P = \{3, 2, 2, 2, 2, 1\}$
$m = 2^1, \ell = 1$
$B = 6$

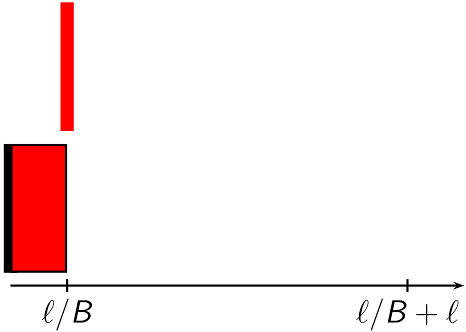
Hardness Proof II

"⇒":

1. complete all subset peers in $[0, \ell/B]$

Example

$P = \{3, 2, 2, 2, 2, 1\}$
$m = 2^1, \ell = 1$
$B = 6$

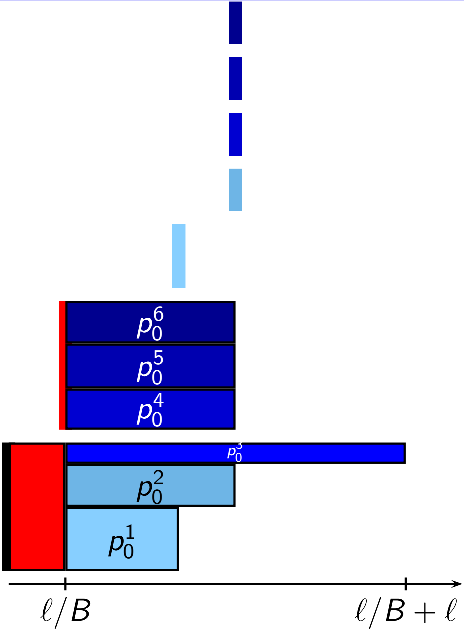


"⇒":

1. complete all subset peers in $[0, \ell/B]$
2. use partition serve all master element peers with full rate

Example

$P = \{3, 2, 2, 2, 2, 1\}$
$m = 2^1, \ell = 1$
$B = 6$



Hardness Proof II

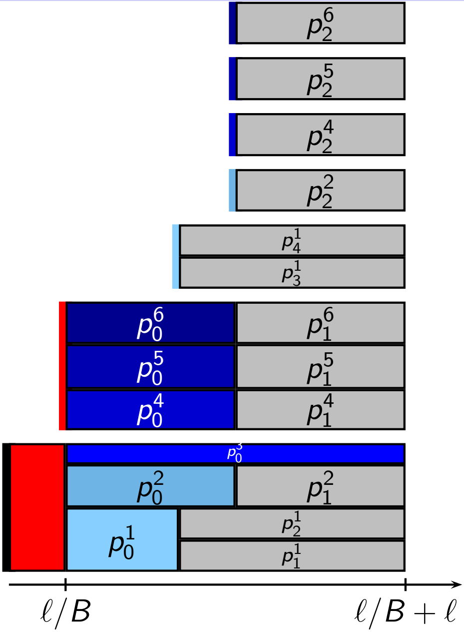
"⇒":

1. complete all subset peers in $[0, \ell/B]$
2. use partition serve all master element peers with full rate
3. serve all element peers

Example

$P = \{3, 2, 2, 2, 2, 1\}$
$m = 2^1, \ell = 1$
$B = 6$

Back to Theorem



Hardness Proof III

"$\Leftarrow$": Useful observations:

- ▷ no element peer sends the file to another peer
- ▷ suffices to show that there is no solution the subset peers and master element peers with $Z(\ell/B+\ell) \geq \sum_{i\in P}(2\ell k_i - 2)$

Case 1: $x_{s_i}(\ell/B) = 1$ for all subset peers s_i

- ▷ Cap. Exp. Lemma: $x_i(\ell/B) = 0$ for all master element peers
- ▷ P is a no-instance $\Rightarrow$ not all master element peers can be served at full rate
- ▷ Cap. Exp. Lemma: $Z(\ell/B+\ell) < \sum_{i\in P}(2\ell k_i - 2)$

Hardness Proof IV

Case 2: $x_i(\ell/B) < 1$ for all master element peers and one subset peer

- capacity not sufficient to serve all peers
- Cap. Exp. Lemma: $Z(\ell/B + \ell) < \sum_{i \in P}(2\ell k_i - 2)$

Case 3: $x_{p_0^i}(\ell/B) = 1$ for a master element peer

- Inequality 2. of Cap. Exp. Lemma is strict $\Rightarrow$ $Z(\ell/B + \ell) < \sum_{i \in P}(2\ell k_i - 2)$

Back to Theorem