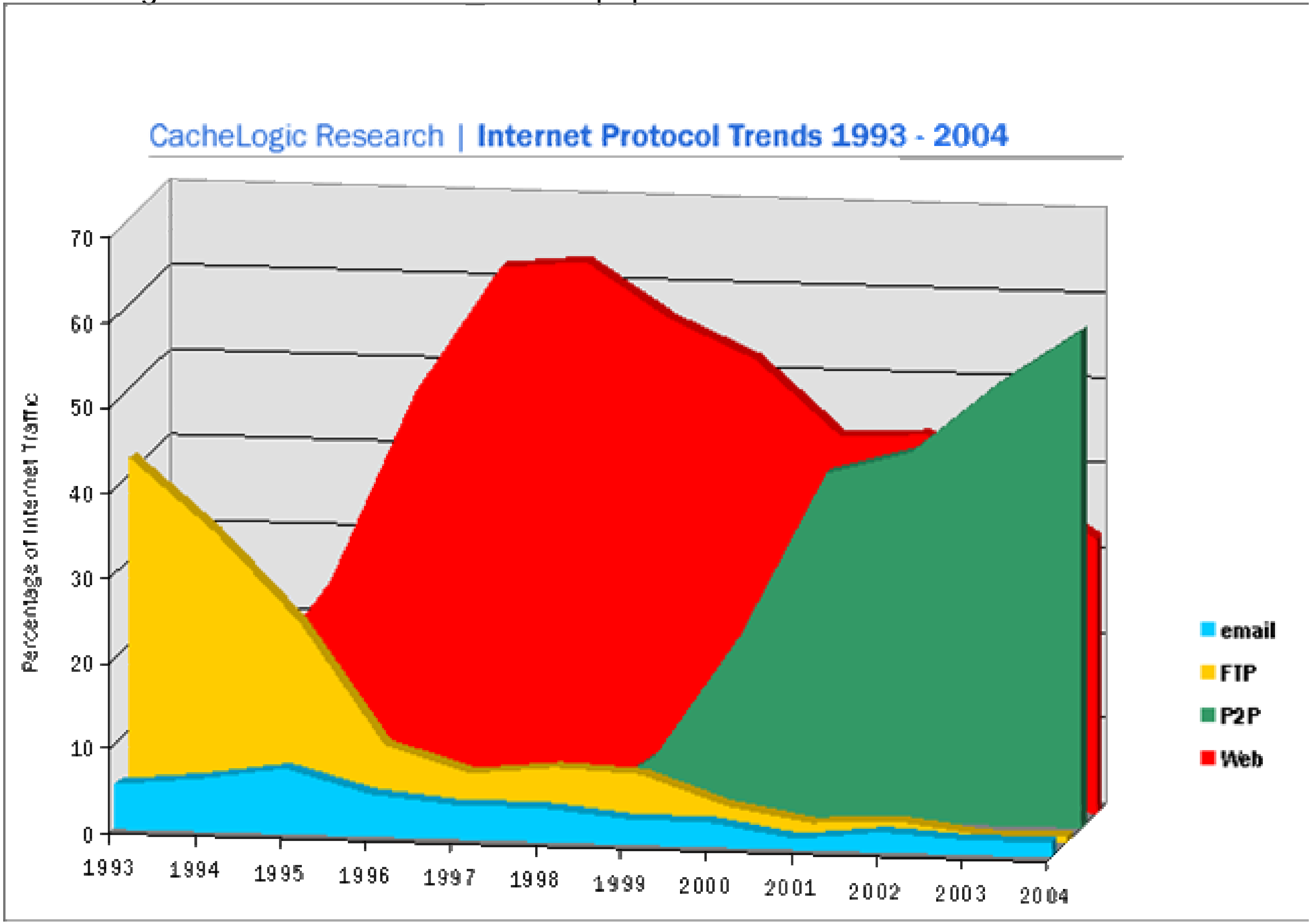
Smart Content Delivery and Storage

Deterministic Source Coding for Efficient Content Distribution

Tamás Lukovszki
lukovszki@inf.elte.hu

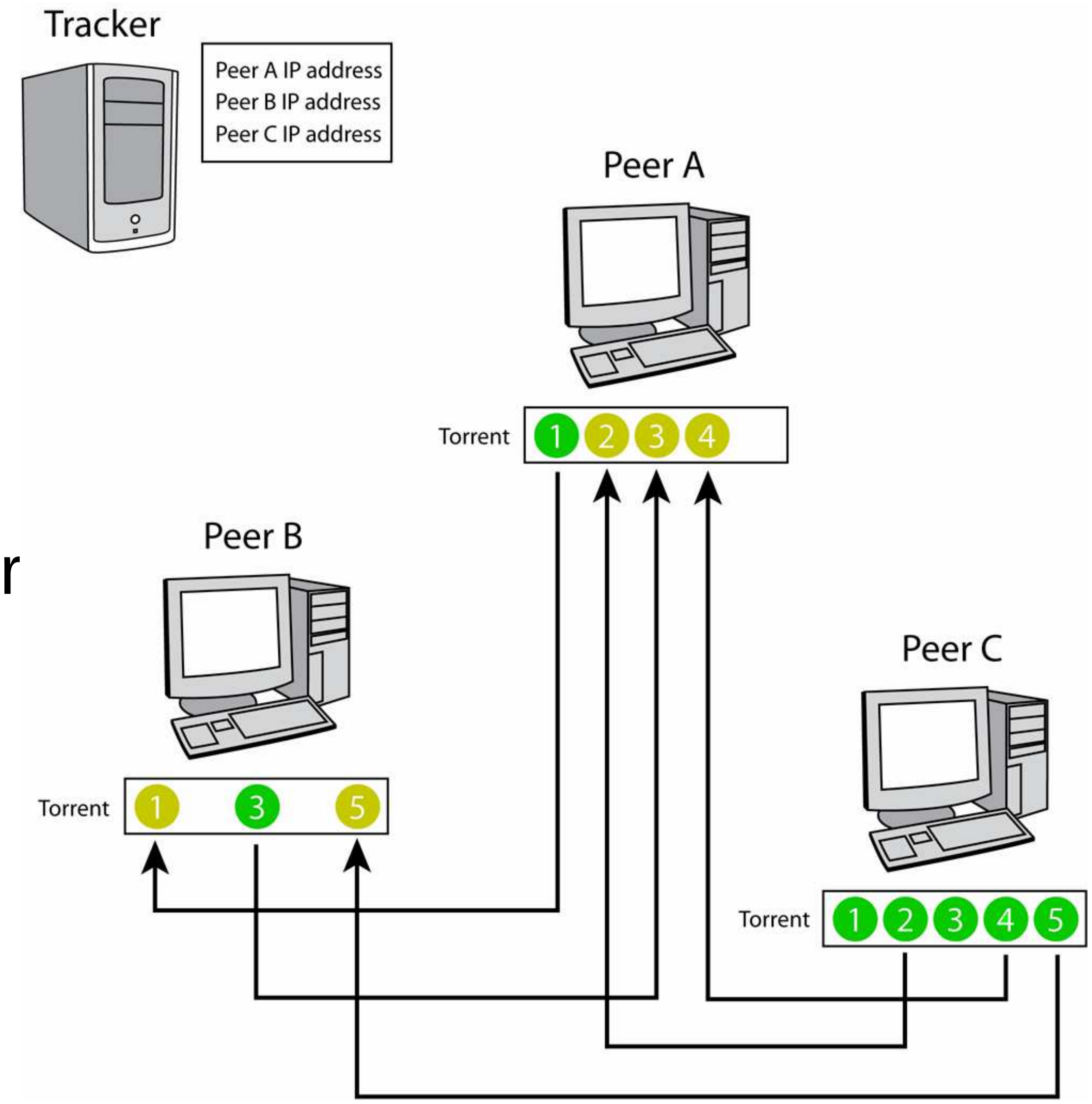
Traffic in the Internet

- http://www.cachelogic.com/research/2005_slide07.php#



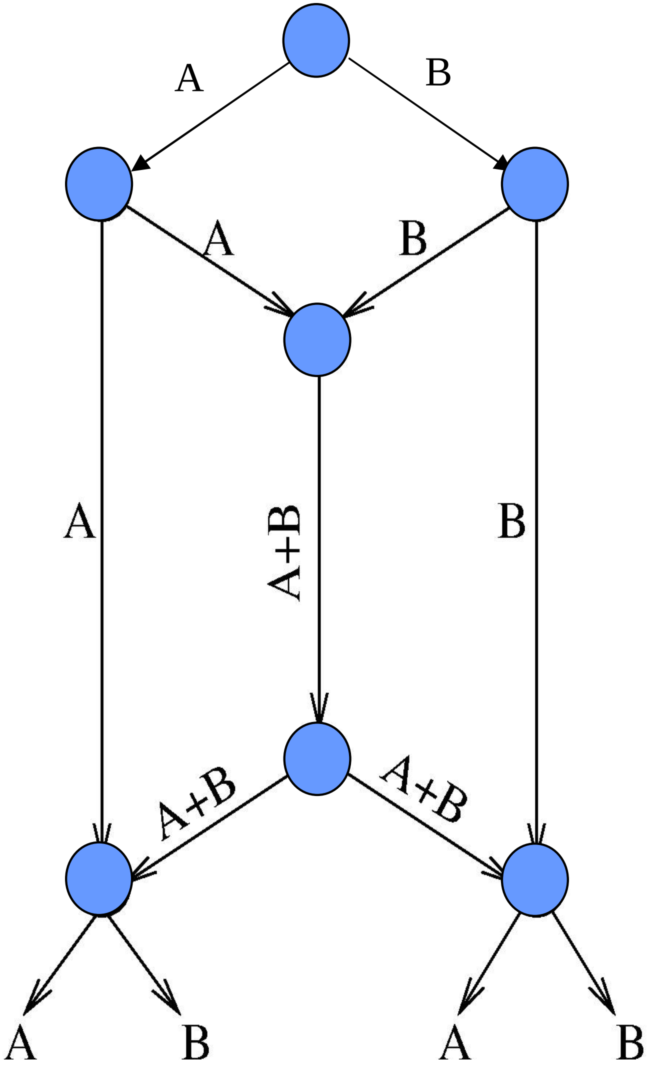
BitTorrent

- File-sharing potocol
- File is divided into pieces
- During downloading the pieces the peers also upload them to each other
- Tit-for-tat
- A problem: rear blocks



Network Coding

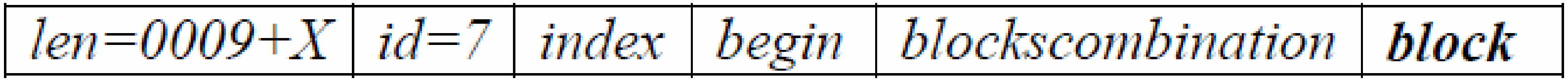
- Nodes are able to combine packets
- For all graphs, there is
  a network code, s.t.
  from a source node
  to a set of destination nodes
  the rate of the multicast can achieve
  the „cut bound”.
  [Ahlswede et al. 2000]
- Computable in polynomial time
  [Jaggi et al. 2005]
- Random Linear Coding is popular



Transmitting blocks with network coding in BitTorrent* [2011]

- Rear pieces problem in wireless/mobile environment

- BitTorrent divides the blocks into subblocks
- Encode the subblocks of a block
  - 0-1 coefficients



len=0009+X	*id=7*	*index*	*begin*	*blockscombination*	***block***

- Backward compatibility with the original BitTorrent protocol

Extending BitTorrent with Network Coding

„Rescueing tit-for-tat with network coding” [Locher et al. 2007]

- Transmit random combination of blocks
- Decode the file after collecting enough linearly independent blocks

- Increases the diversity of blocks
- Avoids the problem of rear pieces
- Robust
- 0-1 coefficients: coding vectors ~ bit vectors

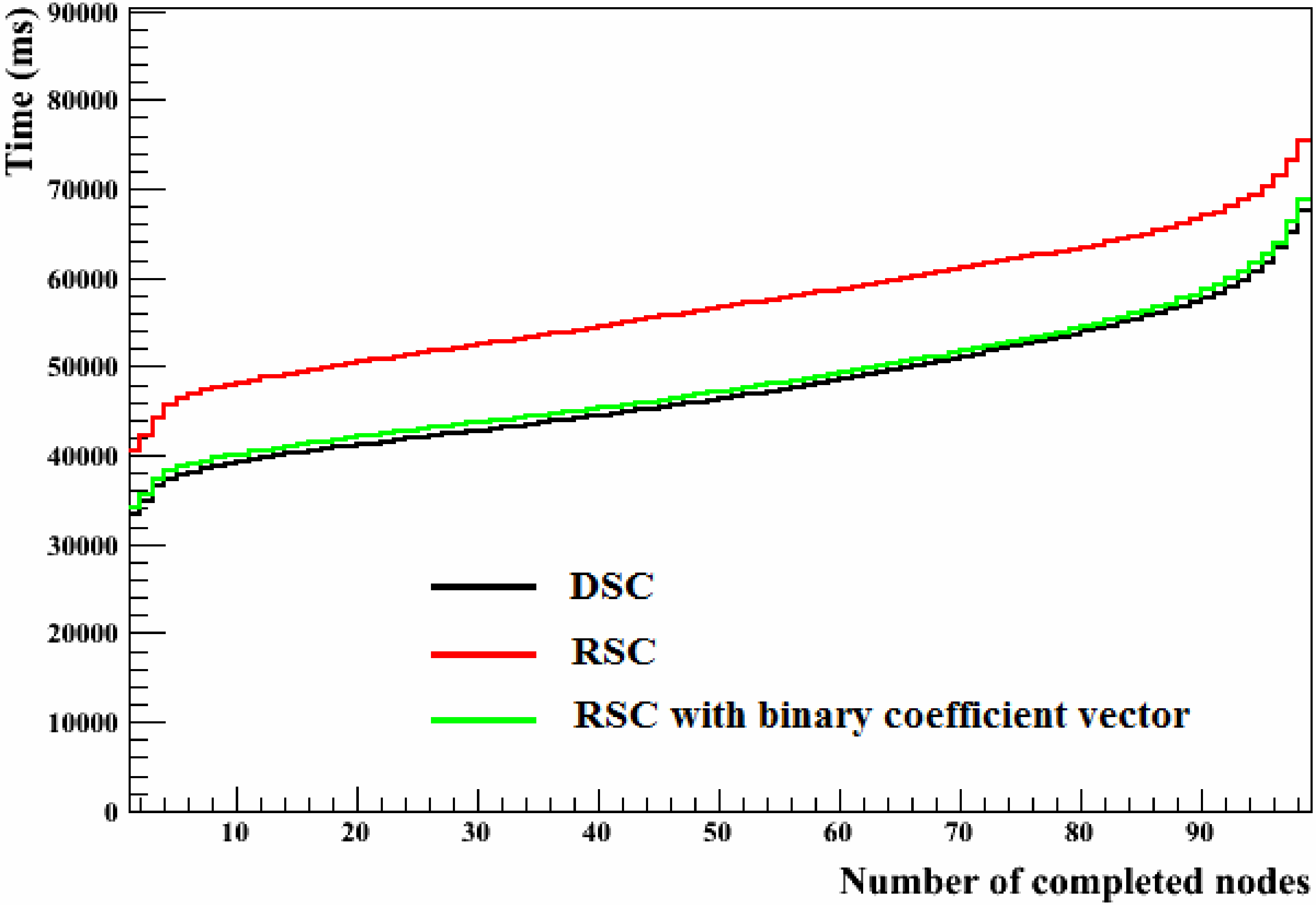
Determistic Source Coding* [2012]

- Random coding
  - does not guarantees decodability after downloading d blocks
  - relatively high communication overhead (coding vectors)

- Deterministic coding
  - decodability is guaranteed
  - easy to check the independence of the vectors
  - significantly lower communication overhead

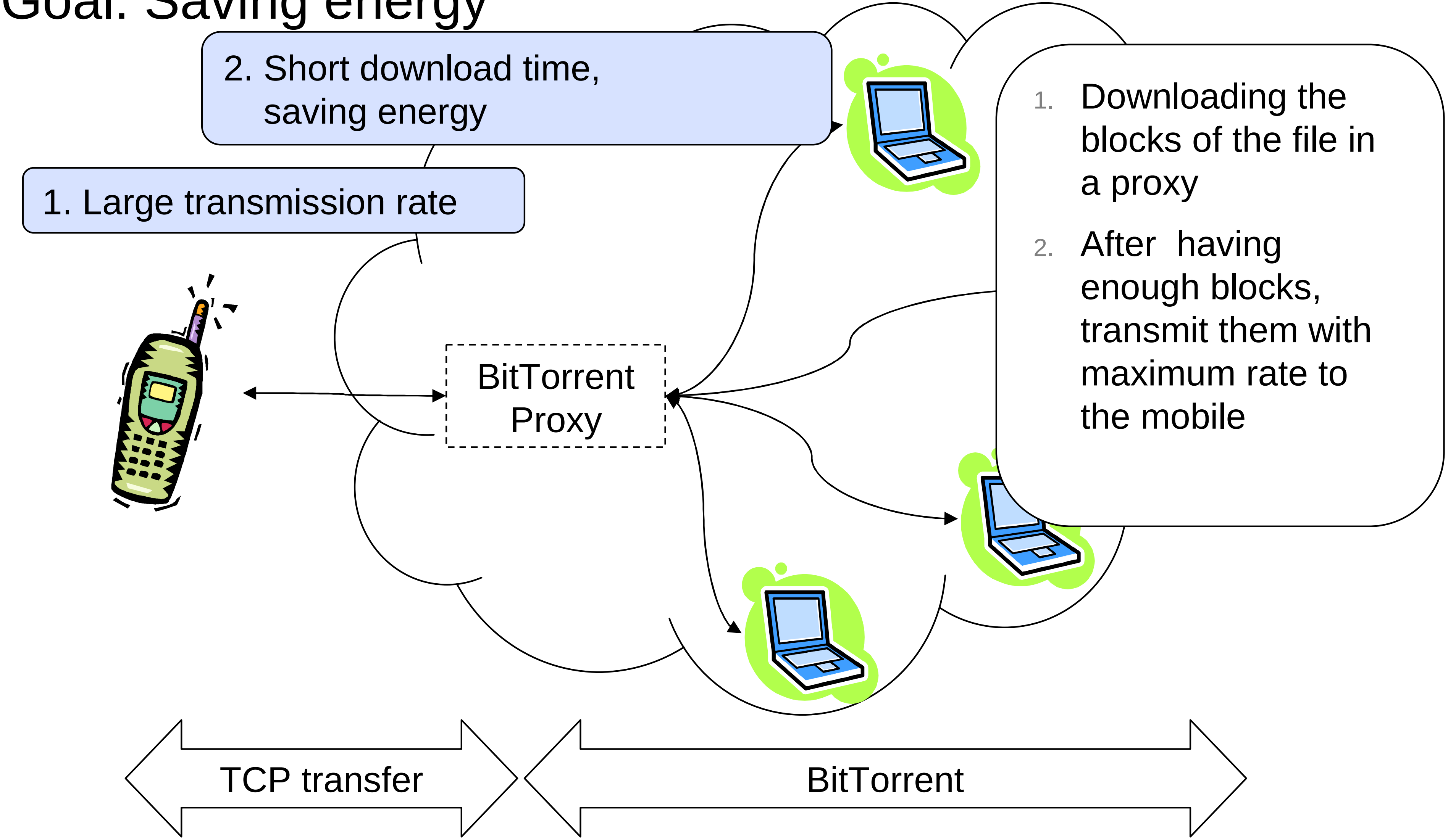
$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_d \end{pmatrix} = \begin{pmatrix} 1 & a_{i_1} & a_{i_1}^2 & \dots & a_{i_1}^{d-1} \\ 1 & a_{i_2} & a_{i_2}^2 & \dots & a_{i_3}^{d-1} \\ 1 & a_{i_3} & a_{i_3}^2 & \dots & a_{i_3}^{d-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & a_{i_d} & a_{i_d}^2 & \dots & a_{i_d}^{d-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_d \end{pmatrix}$$

Simulation Results



BitTorrent using proxies* [2012]

- BitTorrent on mobiles
- Goal: Saving energy



2. Short download time, saving energy

1. Large transmission rate

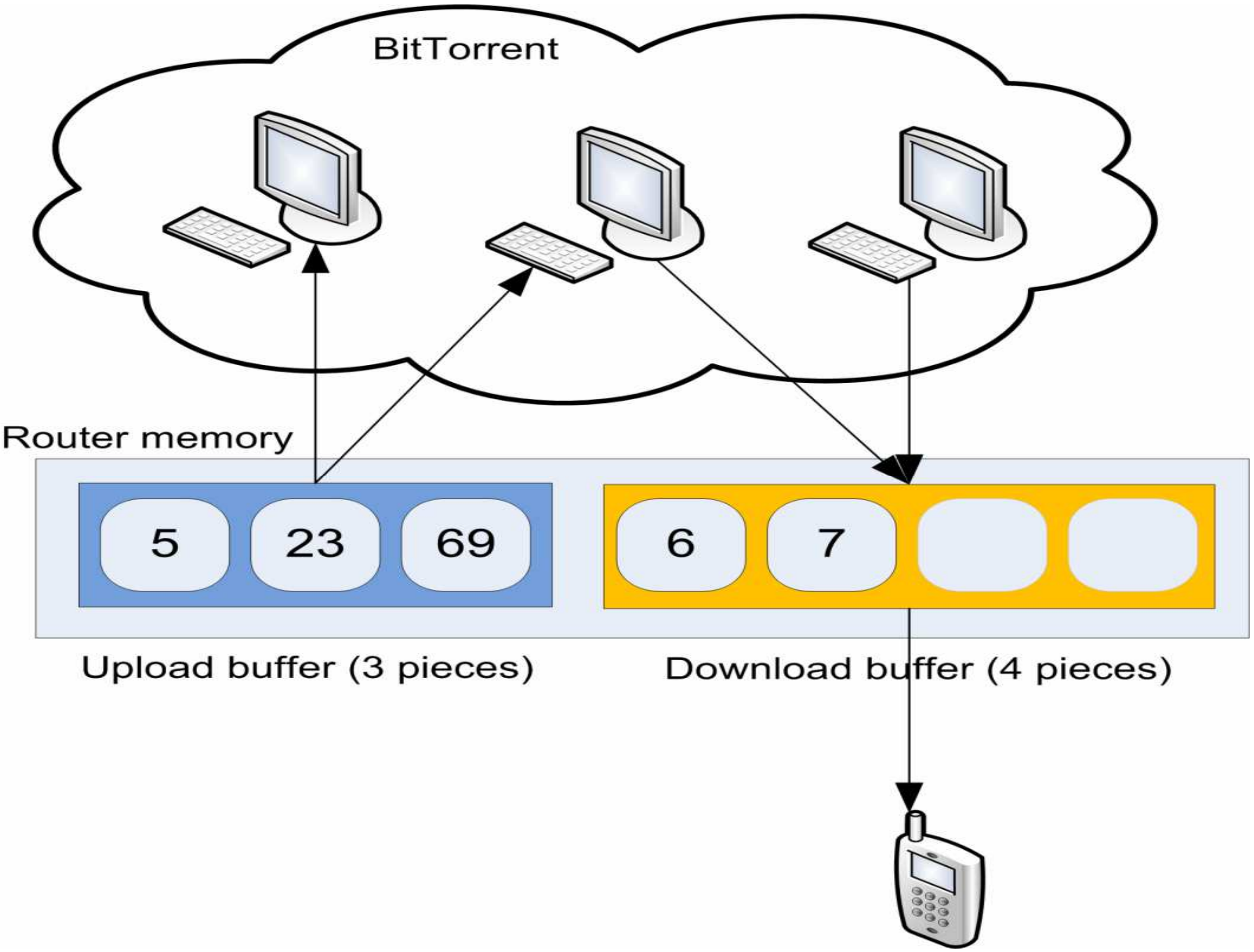
1. Downloading the blocks of the file in a proxy
2. After having enough blocks, transmit them with maximum rate to the mobile

Limits of a Router

- Limited storage
  - E.g.: 8 MB flash and 32 MB RAM
  - Most BitTorrent contents are larger

- Delete the blocks after transmitting them to the mobile
  - Problem:
    - BitTorrent protocol advertises the downloaded blocks
    - Withdrawal is not possible
  - If we do not advertise out blocks, other peers are not able to download from us
    - Bad for us: due to the tit-for-tat rule, slower dowload

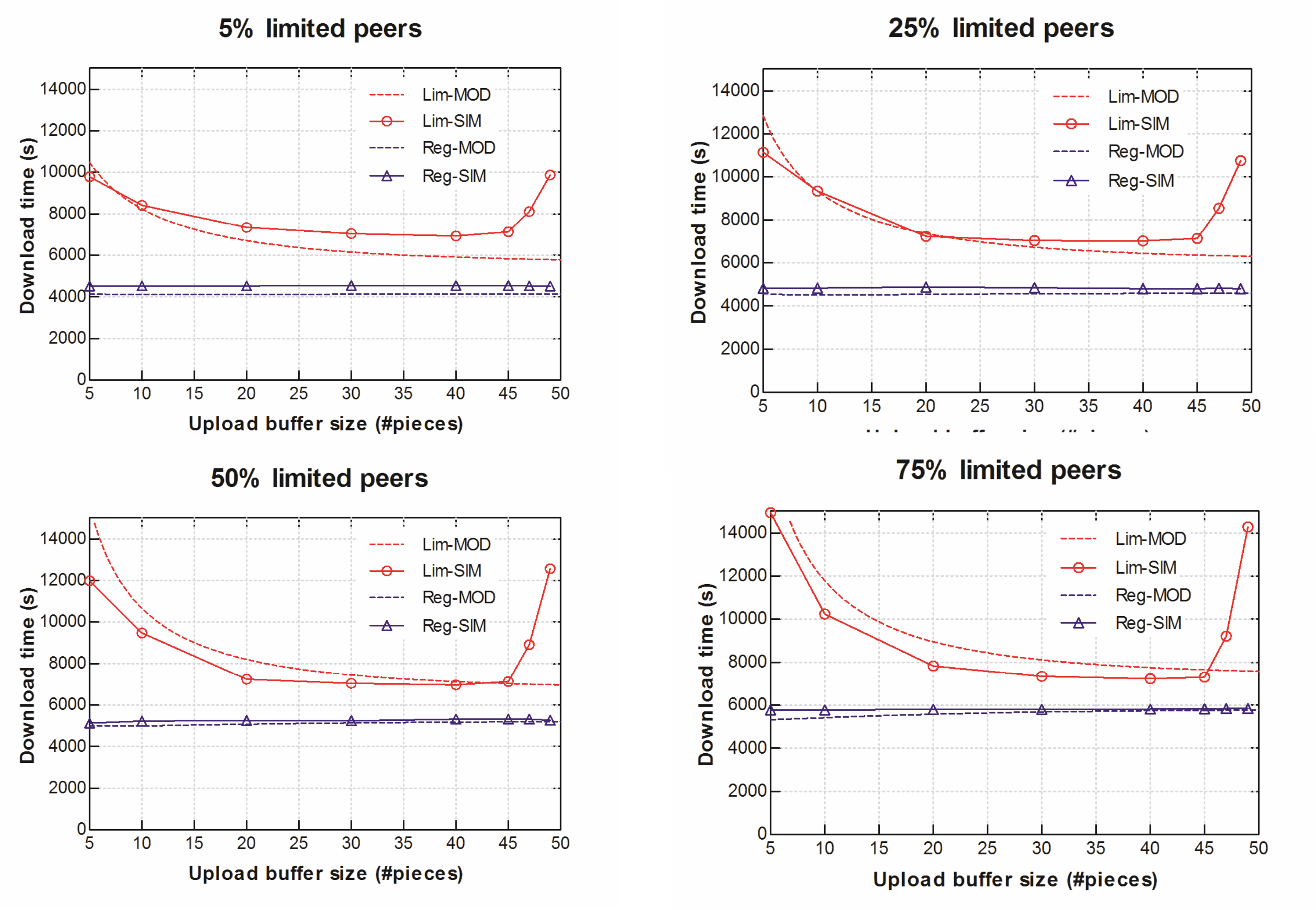
Upload/Download Puffer



- How to divide the buffer?

Download time

- Mathematical model for the analysis of the download and upload rate of the peers
- Simulation results (SIM) vs. analytical results (MOD):



Open Source Library for Deterministic Source Coding

- Developing an open source library for deterministic network coding methods
  - Deterministic source coding has been developed and analyzed in our carrier project
- Main focus: supporting peer-to-peer streaming
  - particularly in mobile environments