

Smart content distribution

Lukovszki Tamás



SZÉCHENYI TERV



MAGYARORSZÁG MEGÚJUL

A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Smart content distribution



- Simplicity of creating, distributing content causing an enormous load on the networks.
- Challenge: development of more cost-effective communication.
- Multimedia real-time content distribution is a very complex task.
 - Caching, virtualization, distributed storage, streaming, content centric networking



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Smart content distribution



- Peer-to-peer (P2P) content distribution:
 - considered as one of the most efficient and most popular method for distributing content
 - P2P: connection between equal partners
- It allows to share multimedia content for millions of users at the same time.
 - Peers download and upload content from each other
 - Radically reduces the load of the server



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Smart content distribution



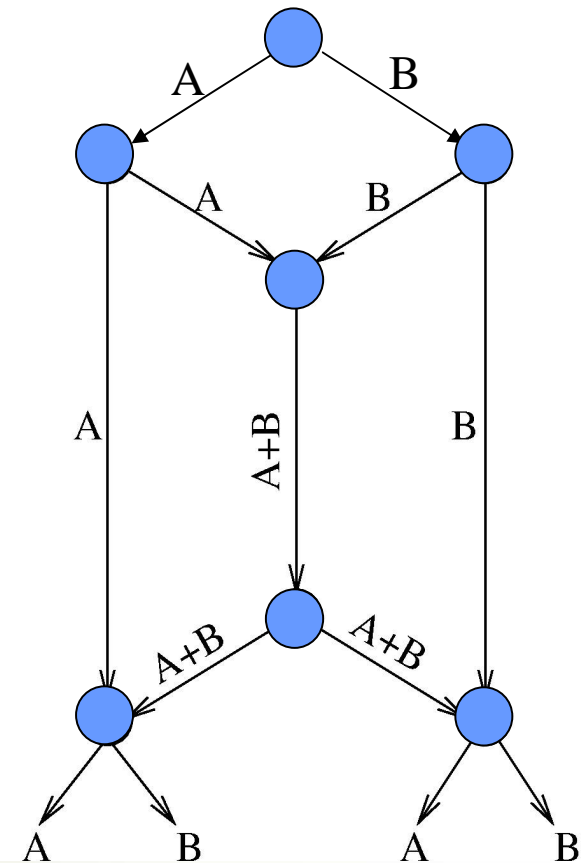
- Our Focus:
 - Open source Java network coding library
 - EIT project (Smart Ubiquitous Content, 2013)
 - Overlay management, neighbor selection
 - Network coding based P2P streaming
 - Network coding based distributed storage
 - Early identification of P2P traffic



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

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”. [Ahlsweede et al. 2000]
- Computable in polynomial time [Jaggi et al. 2005]
- Random Linear Coding is popular



Open Source Network Coding Library



- Developing an open source Java Network Coding library
 - Support for deterministic network coding method
 - Deterministic source coding has been developed and analyzed in our previous projekt in 2012
- Key: supporting peer-to-peer streaming
 - particularly in mobile environments
 - limited memory and computing resources (typical for mobile devices)
- EIT projekt: Smart Ubiquitous Content 2013
 - Partners: U. Trento, Telecom ParisTech, France Telecom Orange, KTH, T-Labs at TU-Berlin, BME, ELTE

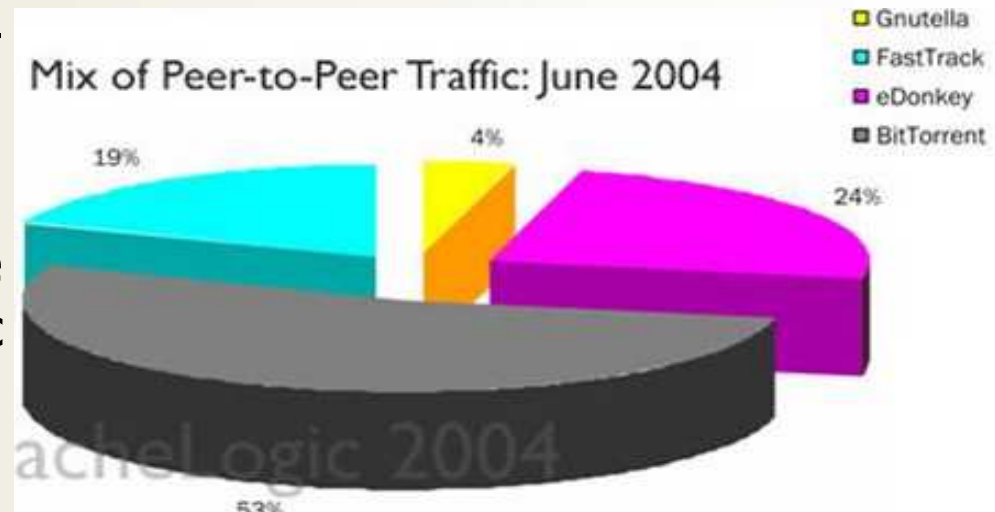


A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

BitTorrent Networks

. BitTorrent:

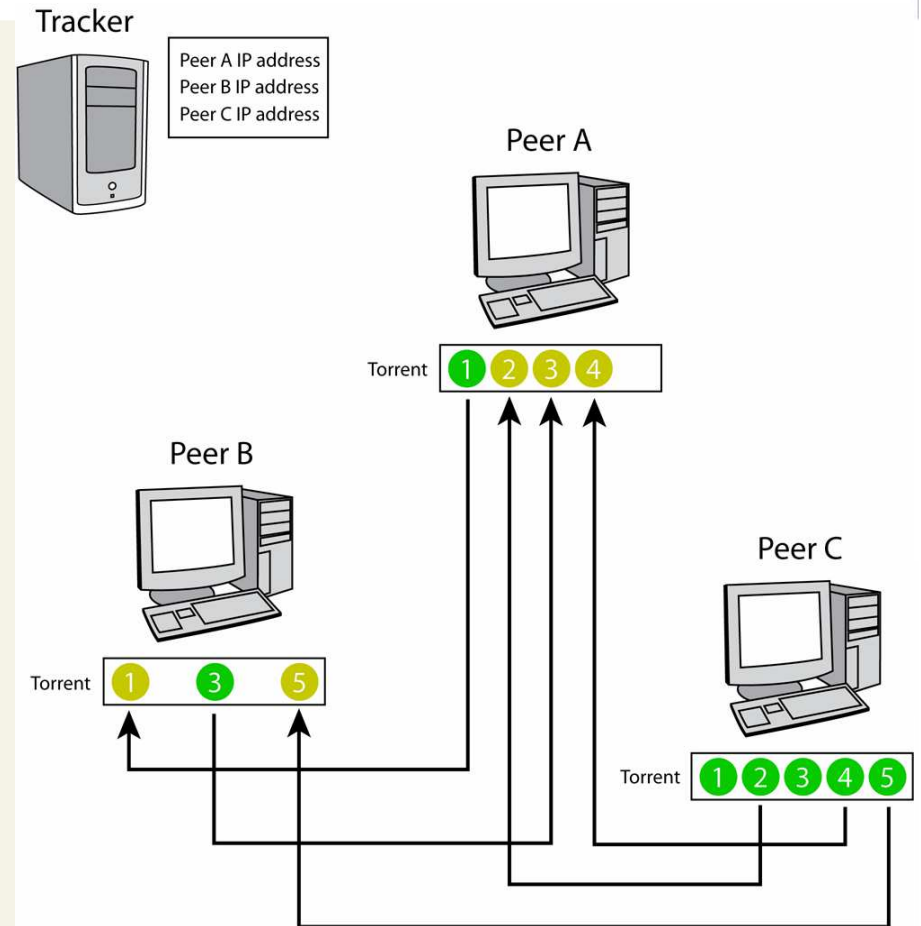
- . Efficient and very popular file sharing system
- . Unstructured P2P network
- . 2001 – Bram Cohen – BitTorrent Inc.
- . And nowadays:
More than 30% of the overall Internet traffic



BitTorrent Networks



- Initialization
 - Peer connects to the tracker
 - obtains a list of neighboring peers downloading the same file
- File download
 - File is divided into pieces
 - Peers download pieces from their neighborhood
 - In the same time they upload to other peers



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Overlay Management for BitTorrent-Like Networks



- The effectiveness of downloading largely depends on the neighbors of the peer.
- We have developed a new neighbor selection strategy.
- Distributing b blocks in a network of n peers in

$O(b + \log n)$ time steps, w.h.p.*

- Improves on previous best upper bound.
- Optimal up to a constant factor.
- Clients: no modification; Tracker: only slight modification

*An event E occurs with high probability (w.h.p.), if given $n > 1$, $\Pr[E] > 1 - 1/n^c$, where $c \geq 1$ is a constant



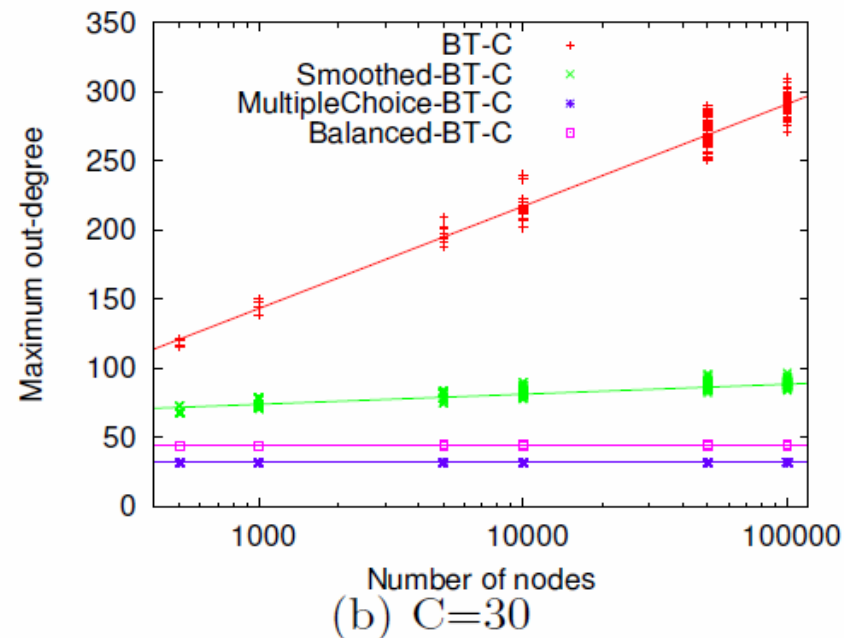
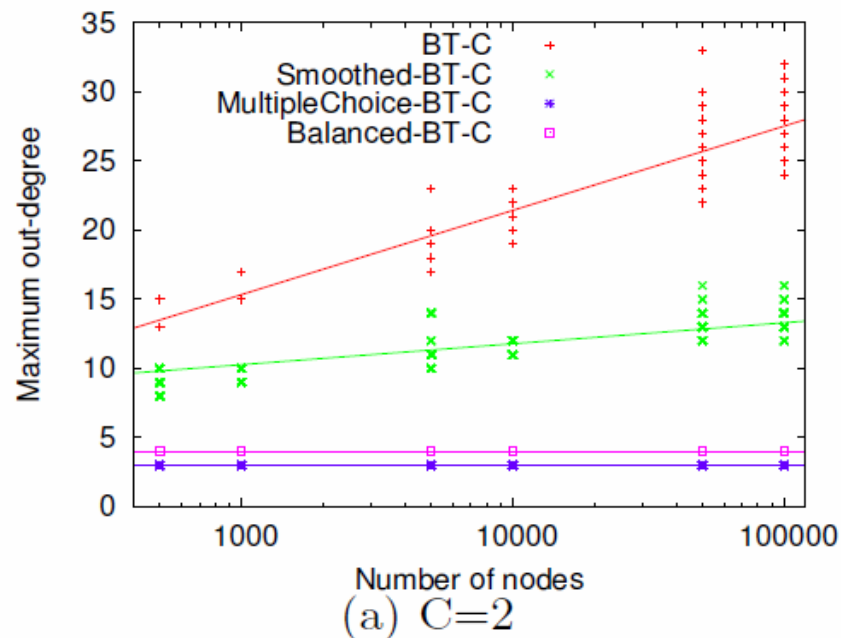
A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Simulations: Out-Degree



Parameters: n varies from 500 to 100000, $C=2,30$

Simulations have been repeated 100 times

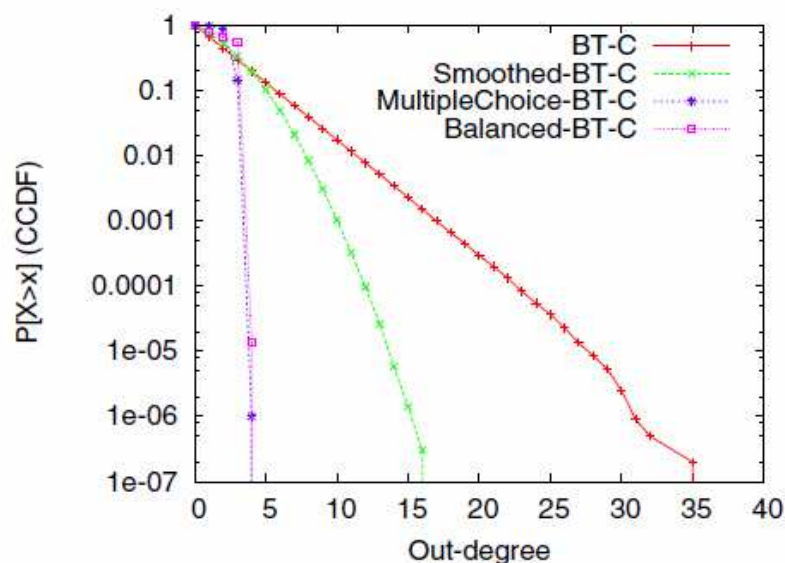


Simulations: Out-Degree

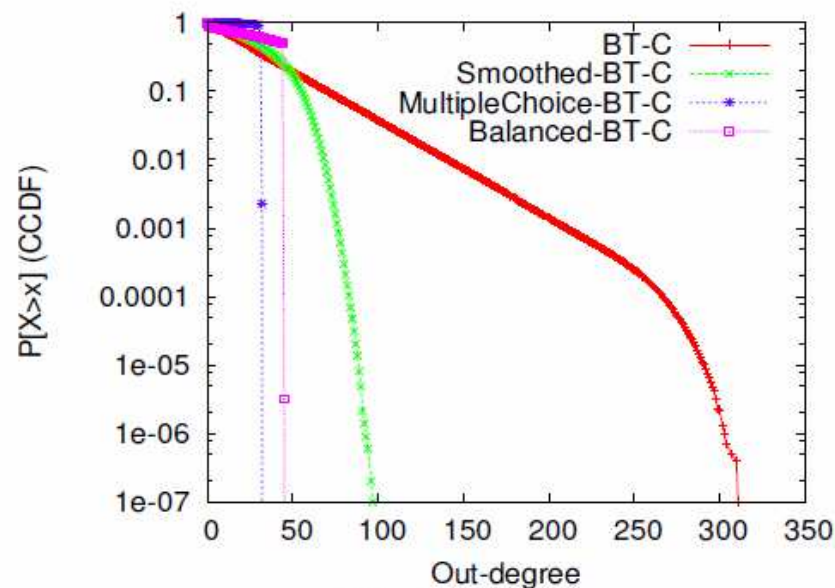


Parameters: n varies from 500 to 100000, $C=2,30$

Simulations have been repeated 100 times



(a) $C=2$



(b) $C=30$



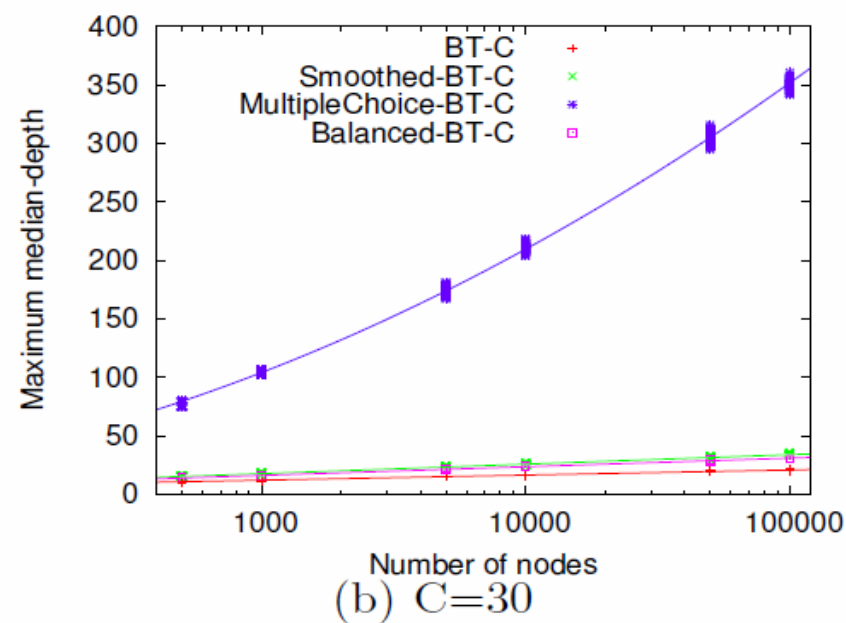
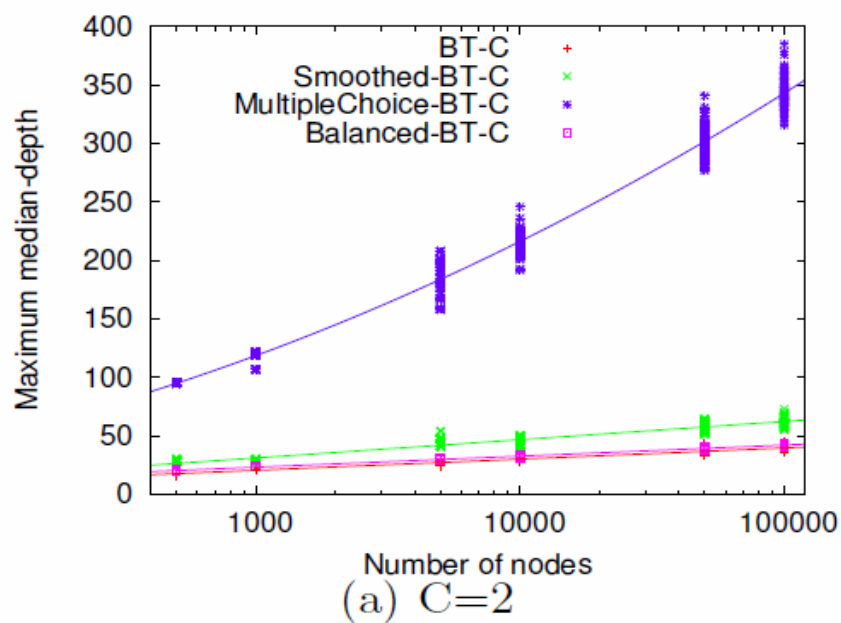
A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Simulations: Median Depth



Parameters: n varies from 500 to 100000, $C=2,30$

Simulations have been repeated 100 times



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

PeerSim Simulations: Download Time

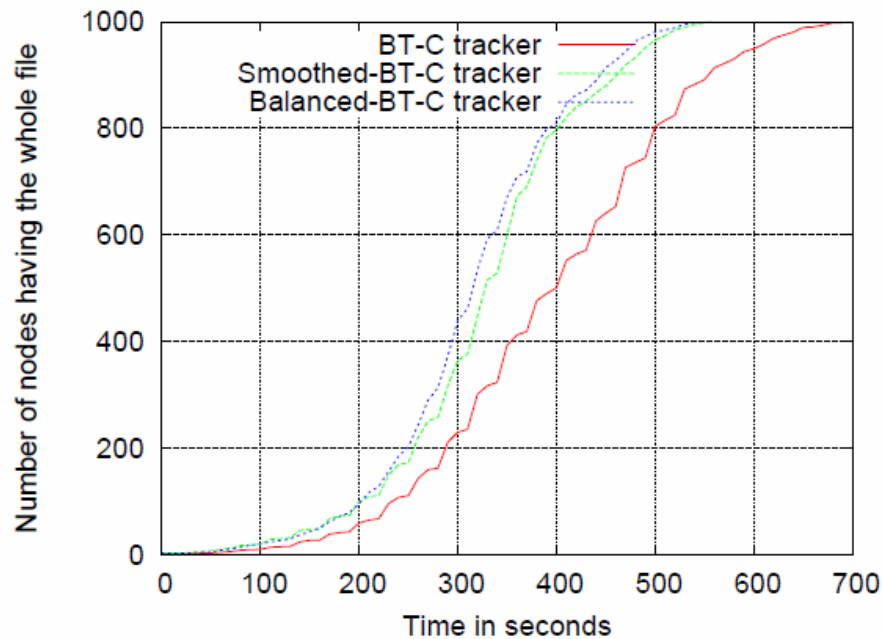


Constant delay and link bandwidth

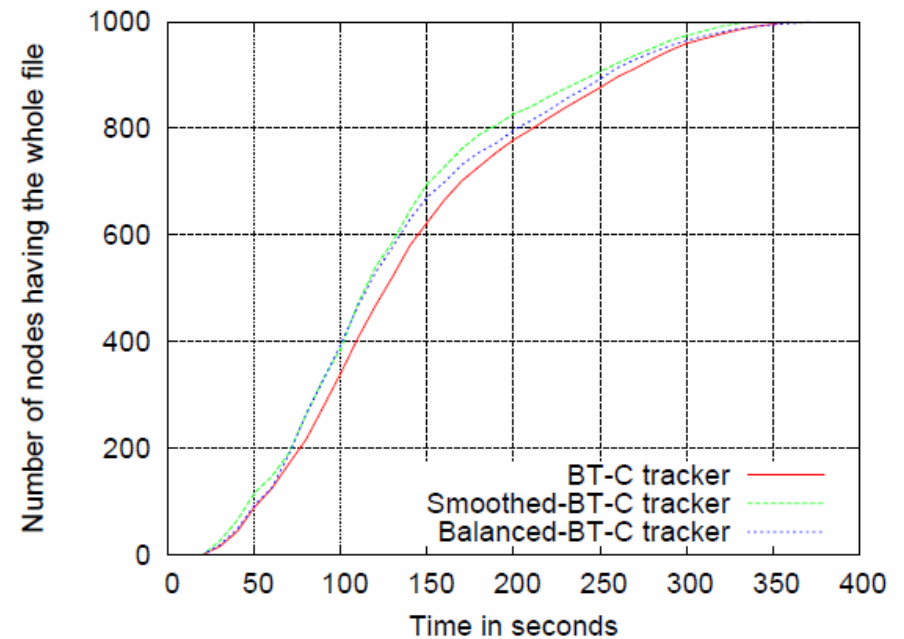
1000 peers joining one by one in a short time period (flash crowd)

File size 10MB

Simulations repeated 20 times



(a) $C=4$



(b) $C=30$

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
- Alternative solution: deterministic network coding [Agocs, Balaton, Lukovszki 2012]
 - Decodability is guaranteed
 - Checking independence of vectors: disjoint \Leftrightarrow independent
 - Lower communication overhead



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Network Coding Based BitTorrent Simulator



- Round based simulator
- Overlay management: several neighbor selection strategies

The screenshot shows a window titled "Torrent simulator" with two tabs: "Configuration" and "Running". The "Running" tab is active. On the left, there are several buttons: "Peers Lose All Data", "Balance Edges", "Initialize" (with a right arrow), and "Run Simulation". Above the "Initialize" button, it says "Graph diameter: 3". The main area displays a network graph with 10 nodes (0-9) and many edges. Node 0 is red, while others are green. Below the graph, a text box shows the simulation results:

```
Finished simulation at round: 8  
0: Pieces got: 10 Need more: false  
1: Pieces got: 10 Need more: false  
2: Pieces got: 10 Need more: false  
3: Pieces got: 10 Need more: false  
4: Pieces got: 10 Need more: false  
5: Pieces got: 10 Need more: false  
6: Pieces got: 10 Need more: false  
7: Pieces got: 10 Need more: false  
8: Pieces got: 10 Need more: false
```

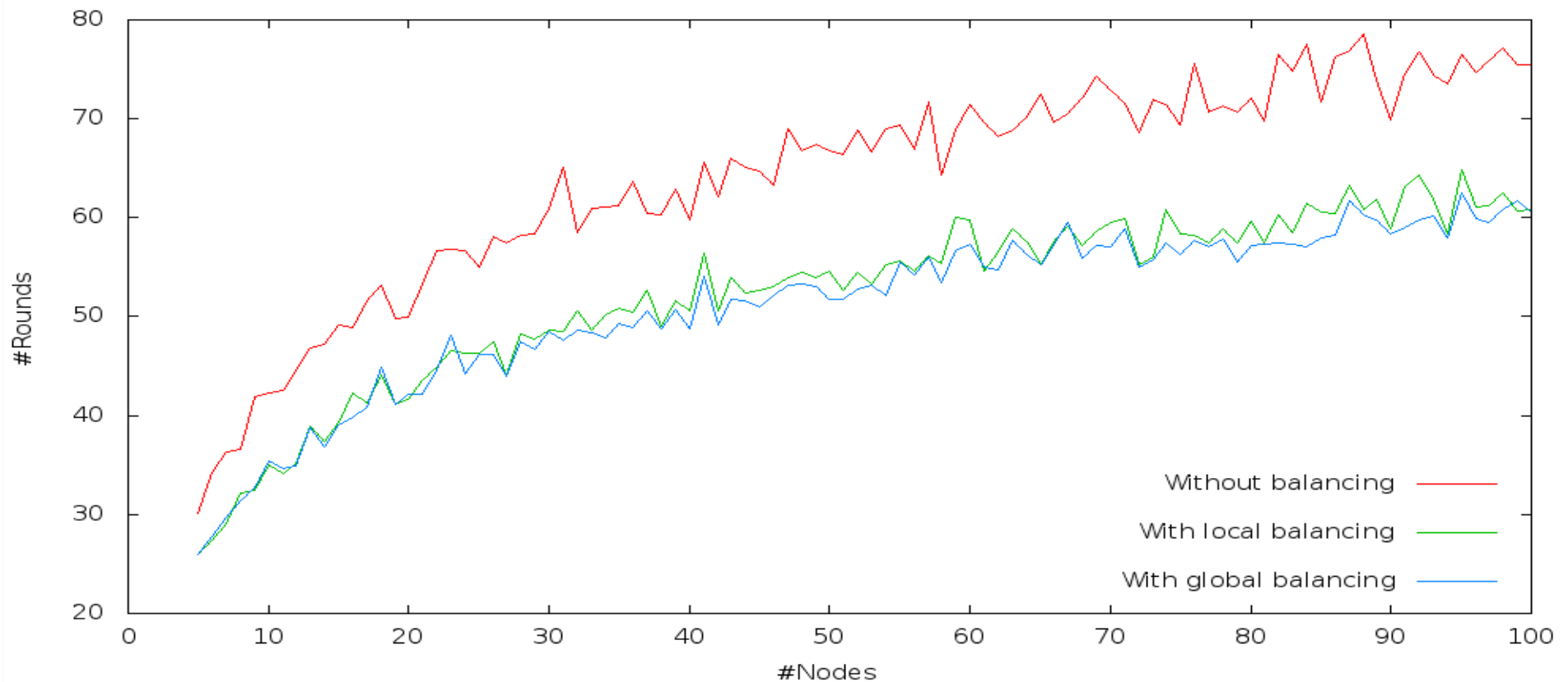


A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Network Coding Based BitTorrent Simulator



Number of rounds until all peers have downloaded the file. The file has been cut into 20 blocks. In each round each peer sends one block. Each result is the average of 30 simulations.

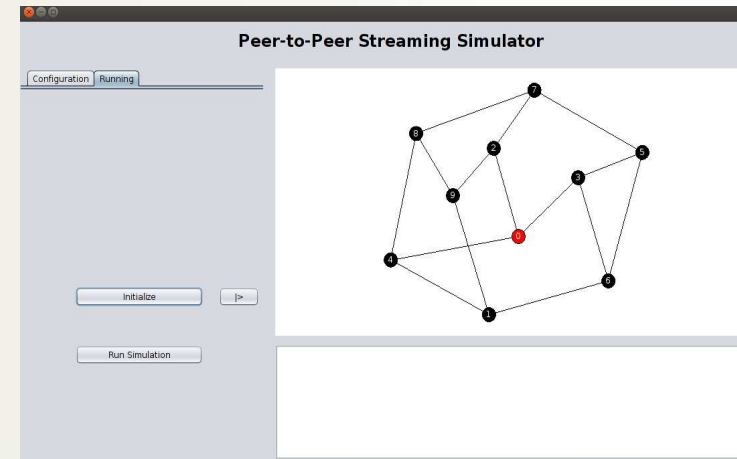
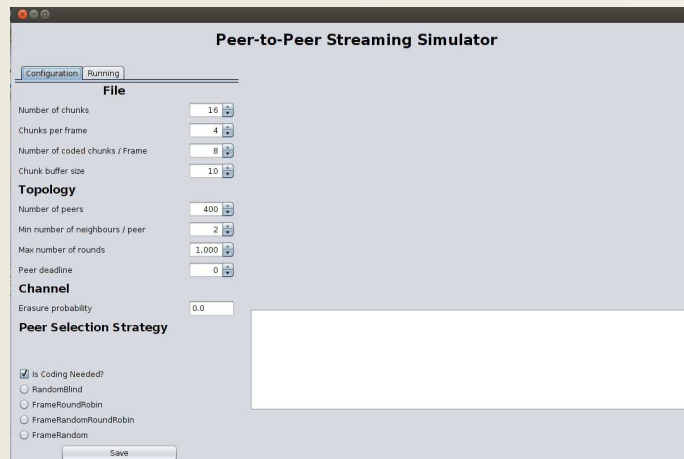


A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Network Coding Based Streaming Simulator

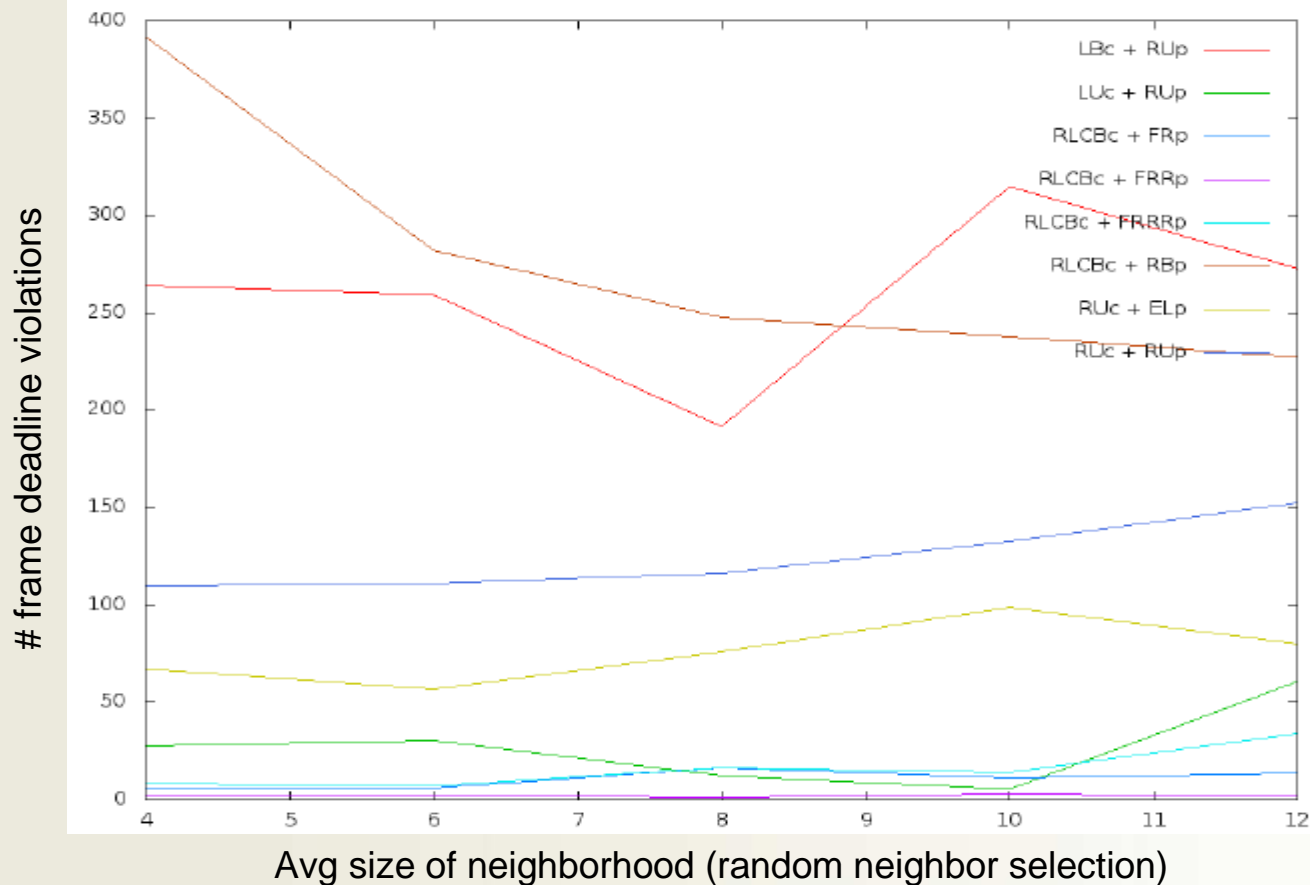


- Round based simulator
- Overlay management:
 - random neighbor selection
 - balanced neighbor selection
 - random d-regular graphs
- Chunk selection strategies
- Peer selection strategies
- Buffer management
- Erasure channels



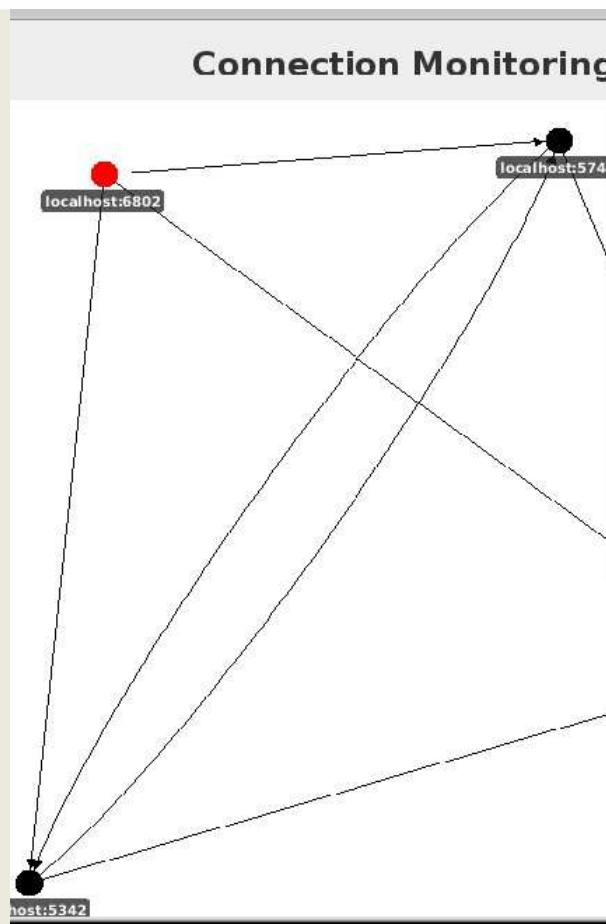
A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Network Coding Based Streaming Simulator

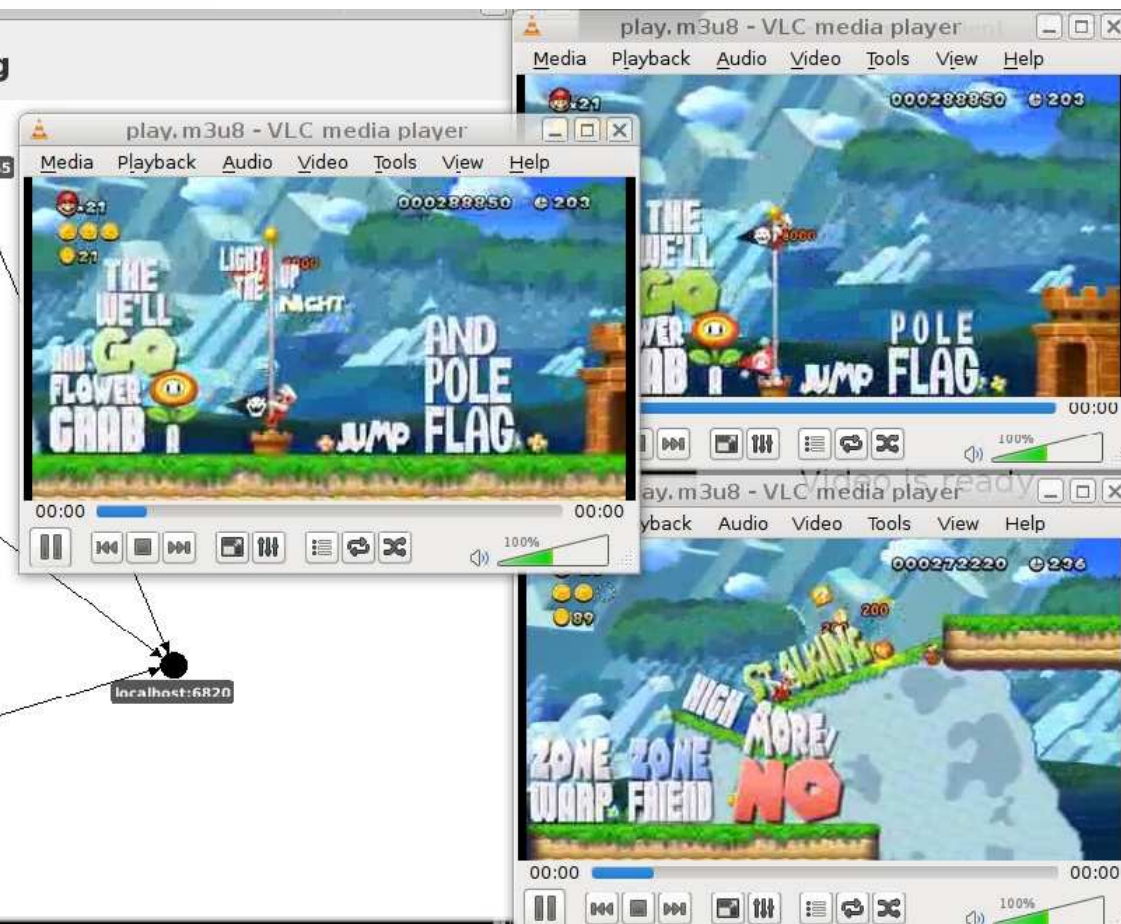


Network Coding Based Streaming

Connection Monitoring



The diagram illustrates a network topology for connection monitoring. It features four nodes: localhost:6802 (top left, red dot), localhost:5745 (top right, black dot), localhost:5342 (bottom left, black dot), and localhost:6820 (bottom right, black dot). Lines connect localhost:6802 to localhost:5745 and localhost:5342. Lines also connect localhost:5745 to localhost:5342 and localhost:6820. Finally, lines connect localhost:5342 to localhost:6820.



Three overlapping windows of VLC media player are shown, each displaying a different frame from a Super Mario Bros. game. The windows are titled 'play.m3u8 - VLC media player'. The top window shows a scene with the text 'THE WELL', 'LIGHT UP THE MOUNTAIN', 'AND POLE FLAG', 'FLOWER GRAB', and 'JUMP FLAG'. The middle window shows a scene with the text 'THE WELL', 'POLE FLAG', 'JUMP FLAG', and 'JUMP FLAG'. The bottom window shows a scene with the text 'HIGH MORE!', 'ZONE ZONE', 'WARP FRIEND', and 'NO'. Each window includes a progress bar, playback controls, and a volume slider.

Network Coding Based Distributed Storage Simulator



- Distributed storage systems use redundancy to provide reliable access to data spread over individually unreliable nodes.
- In case of a node failure a new node must substitute the failing one.
- We are examining, how network coding can be used to reduce the repair traffic in distributed storage systems
- Developing a network coding based distributed storage simulator
- Considering also erasure channels



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Network Coding Based Distributed Storage Simulator



Applet Viewer: AppletGui.class
Applet

Start

Storage node count: 16
Storage Node availability: 90
Data Collector connection count: 7
Storage Node connection count: 5
Data piece count: 8
Stored piece count in Storage Node: 2
Storage Node vector send count: 1
Failing node count: 1
Show storage nodes connections:
Show source node connections:

First round Next failure Next repair New Data Collector came delete node 10 Repair node 10

start 00 10 20 30
01 11 21 31
02 12 22 32
03 13 23 33 dc



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Participants



Researcher:

Dr. Tamás Lukovszki (team leader)

Sándor Laki

Students:

Dániel Csubák

Marcell Kressz

Péter Varga

Péter Vörös



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Publications



1. Sándor Laki, Tamás Lukovszki: On a Balanced Neighbor Selection Strategy for Tracker-Based Peer-to-Peer Networks. In: *13th IEEE International Conference on Peer-to-Peer Computing (P2P'13)*, 2013.
2. Sándor Laki, Tamás Lukovszki: Balanced Neighbor Selection for BitTorrent-Like Networks. In: *21st European Symposium on Algorithms (ESA 2013)*, Springer, LNCS, Vol. 8125, pp 659-670, 2013.
3. László Blázovics, Tamás Lukovszki: Fast Localized Sensor Self-Deployment for Focused Coverage. In: *9th International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics (ALGOSENSORS 2013)*, Revised selected papers: Springer, LNCS Vol. 8243, pp 83-94, 2014.
4. László Blázovics, Tamás Lukovszki, B. Forster: Surrounding robots -- A discrete localized solution for the intruder problem. In: *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 18(3), 315-319, 2014.
5. Béla Hullár, Sándor Laki, András György: Efficient methods for early protocol identification. Accepted: *IEEE Journal on Selected Areas in Communications (JSAC)*, Special Issue on Deep Packet Inspection: Algorithms, Hardware, and Applications, 2014.



A projekt a Kutatási és Technológiai Innovációs Alap támogatásával valósul meg.

Thank you!