Network Coding for Large Scale Content Distribution

IEEE Infocom 2005

Presented by 唐崇實

Outline

- Introduction
- Contributions of This Paper
- End-system Cooperative Content Distribution
- Content Distribution Model
- Performance Evaluation
- Summary and Discussion

Introduction

- Typical content distribution solution: based on placing dedicated equipment inside or at the edge of the Internet
- New paradigm: based on a fully distributed architecture where commodity PCs are used to form a cooperative network and share their resources including storage, CPU, bandwidth

Introduction

- A new end-system cooperative solution using network coding is proposed to overcome most of the problems in existing end-system cooperative schemes such as BitTorrent
  - Inefficiencies are more pronounced
    - in large and heterogeneous populations
    - during flash crowds
    - in environments with high churn
    - when incentive mechanisms are in place

Contributions of this paper

- A practical system based on network coding for file distribution to a large number of cooperative users
  - Knowledge of underlying network topology is not required
  - Nodes make decisions of how to propagate packets based on local information only
- Network coding performs better than transmitting uncoded blocks, or using erasure codes
  - Performs better by almost a factor of 2 compared to source coding, and by a factor of 3 compared to not coding
  - Improve download rates by almost 20% compared to source coding and by more than 30% compared to no coding
- Network coding system is very robust to extreme situations with sudden server and node departures

Content Distribution Using End-system Cooperation Techniques

- Tree-Based Cooperative Systems
  - Creating and maintaining shortest-path multicast trees
  - Bandwidth limited: transfer rate to a client is limited by bottleneck link
- Mesh Cooperative Architectures
  - The most popular one is BitTorrent
  - If nodes make local decisions, same block may travel over multiple competing paths, hence, network resources are under-utilized and the download rates decrease
- Erasure Codes (source coding)
  - Digital Fountain: enables end-hosts to efficiently reconstruct the original content of size n from a subset of any n symbols from a large universe of encoded symbols
- Network Coding

Content Distribution Model

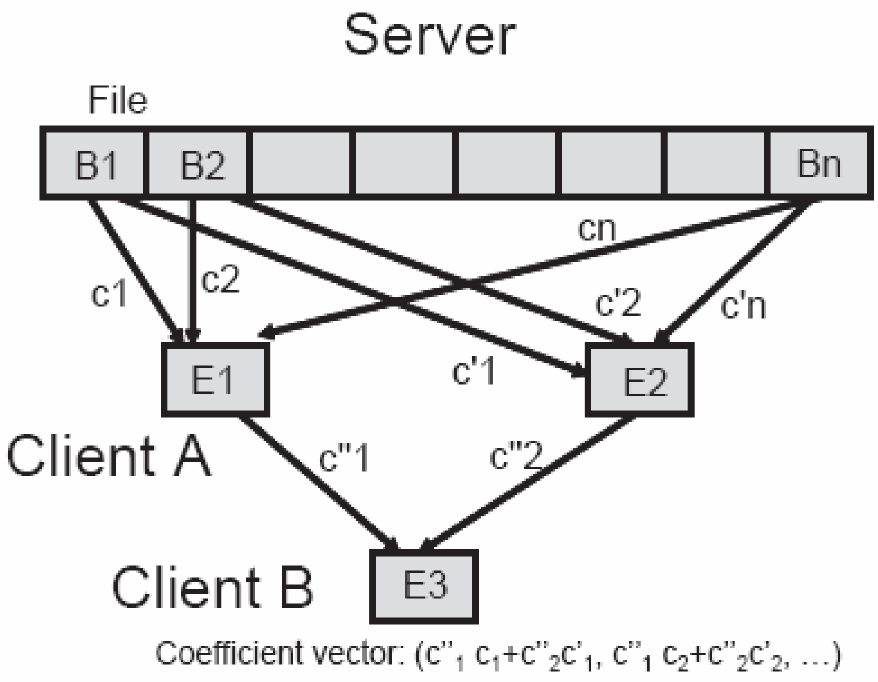
- Use this model to either distribute
  - Blocks of original file (no coding), or
  - Blocks of encoded information where encoding happens only at the source (source coding), or both at the source and at the network (network coding)
- Collaborative content distribution network
  - A population of users are interested in retrieving a file which originally exists in a single server (or an end host)
  - The capacity of the server is limited, thus users contribute their bandwidth resources to help other users
  - The server divides the file into k blocks, and the clients collaborate with each other to assemble all the k blocks to reconstruct the original file

Content Propagation of Uncoded & Source-encoded Information

- Assume each user only knows about the blocks it has downloaded and the blocks that exist in its neighbors
- Commonly used heuristics based on local information
  - *Random block:* the block to be transferred is decided at random among the blocks that exist in the source
  - *Local Rarest:* the block to be transferred is picked among the rarest block in the neighborhood
  - *Global Rarest:* a baseline scheme which is not practical in large network, the block to be transferred is the system-wise rarest among all blocks that exist in the neighborhood

Content Propagation with Network Coding

- Initially all users are empty and that user A contacts Server to get a block.
- Server combines all blocks of the file to create an encoded block $E1 = \sum_{i=1}^{n} c_i B_i$ where $c_1, c_2 \ldots c_n$ are randomly selected coefficients
- A node can recover the original file after receiving *k* blocks for which the associated coefficient vectors are *linearly independent* to each other.



Incentive Mechanism

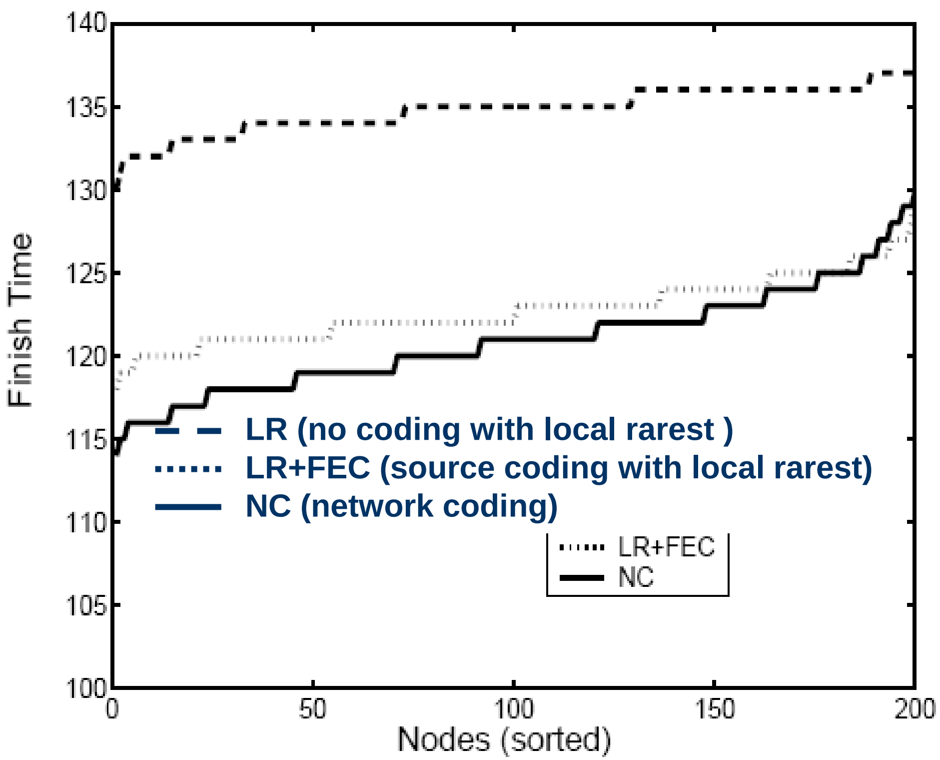
- Two mechanisms to discourage free-riding
  - Give priority to exchanges over free uploading to other nodes: when contention for the upload capacity, the user will preferentially upload blocks of information to neighbors from which it is also downloading blocks
  - Tit-for-tat approach used in BitTorrent: a user does not upload content to another user unless it has also received enough content from that user
- Given that nodes make decisions based on local information
  - A node may end-up downloading blocks that are already popular across the system and can not be traded easily
  - This effect gets amplified when the network frequently reconfigures

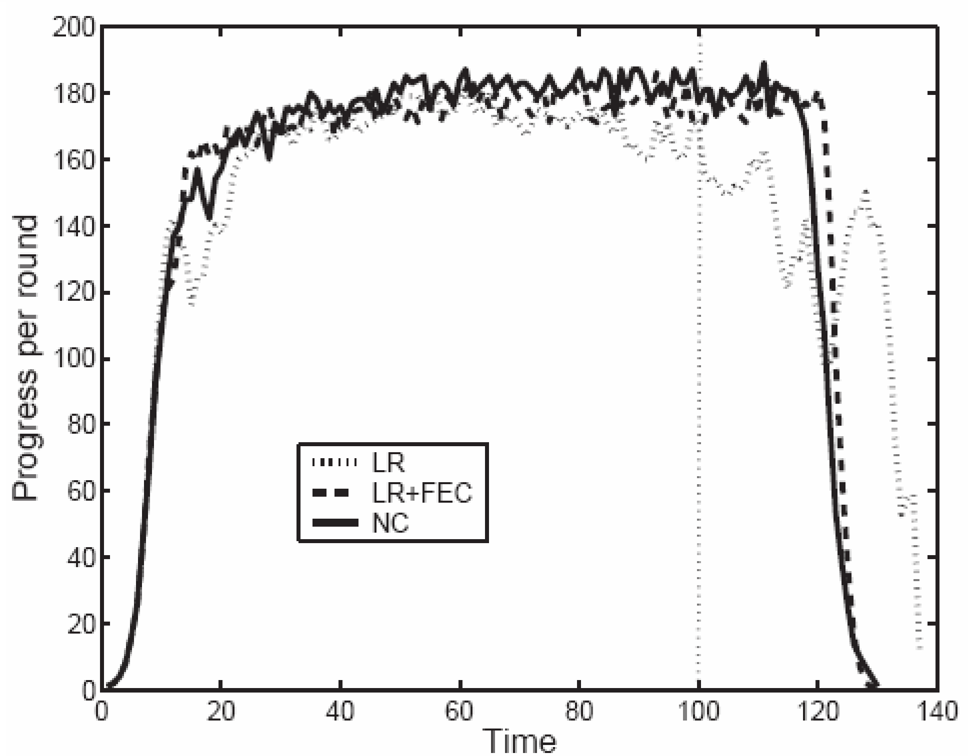
Performance Evaluation

- A simulator to compare the performance of content propagation using network coding, not coding at all, and coding only at server
  - Input: a set of nodes with constraints in upload and download capacities, file size and capacity of single server
  - Support dynamic user populations with node joining and leaving the system, and topology reconfiguration
- Experimental results on
  - Homogeneous topologies
  - Topologies with clusters
  - Heterogeneous capacities
  - Dynamic arrivals and departures
  - Incentive mechanisms: Tit-for-tat

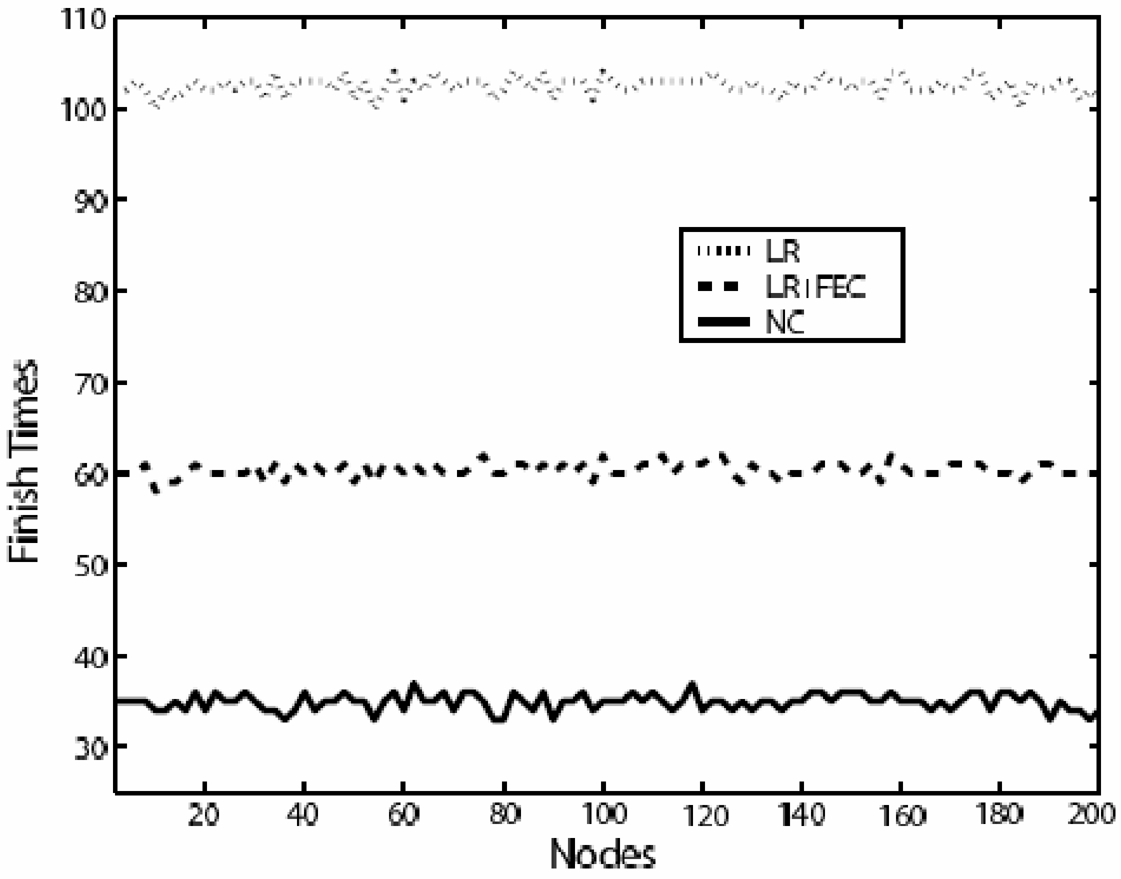
Homogeneous Topologies

- A well connected network of 200 nodes
- All nodes have the same access capacity of 1 block per round



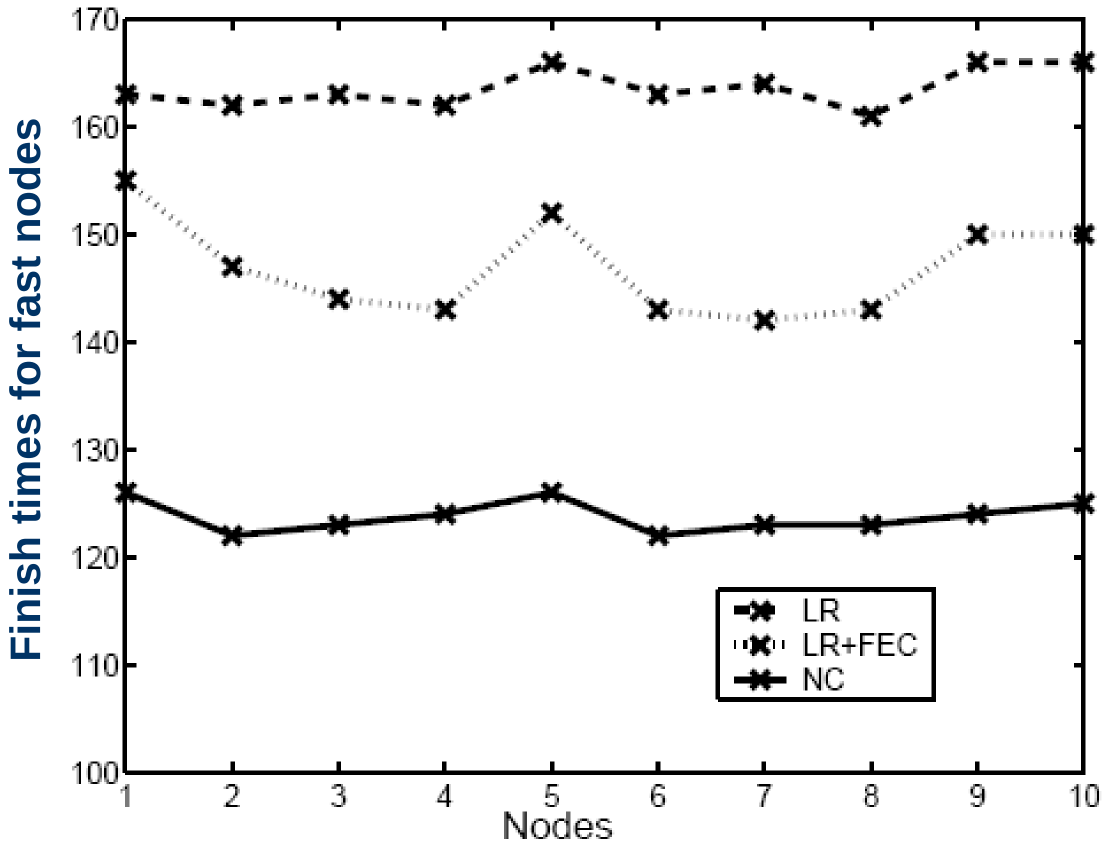


Topologies with Clusters



- Two clusters of 100 nodes each
- 8 blocks / round between nodes within a cluster
- 4 blocks / round between clusters
- Capacity of server: 4 blocks / round
- Server departs at round 30

Heterogeneous Capacities



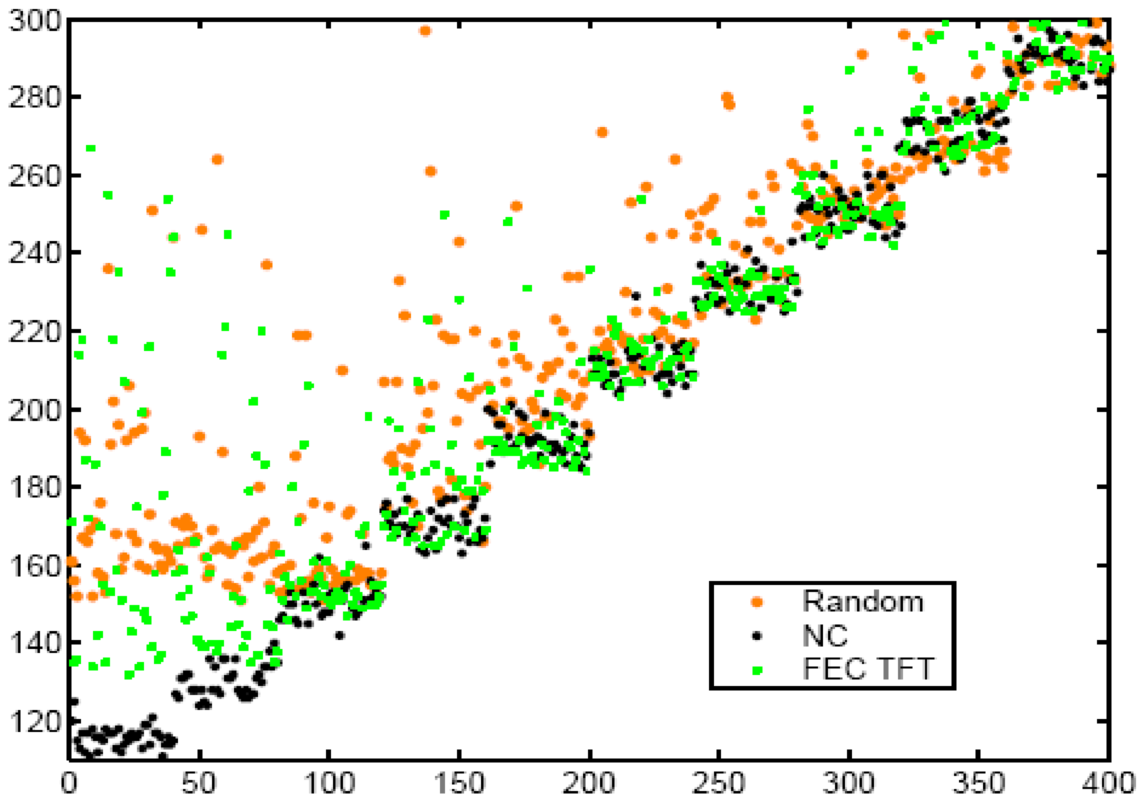
- 10 fast nodes
- 190 slow nodes
- File size: 400 blocks
- Capacity of server and fast nodes: 4 blocks/round

Heterogeneous Capacities

- As the capacity difference between fast nodes and slow nodes increase, fast nodes experience even worse performance when network coding is not used
- One fast peer, 50,100 and 200 slow peers for 3 cases

Method	x2	x4	x8
Random	107 →	166 →	281
Local Rarest	106 →	135 →	208
Source Coding Random	84	113	134
Source Coding LR	78	92	106
Global Rarest	75	92	98
Network Coding	69	72	73

Dynamic Arrivals

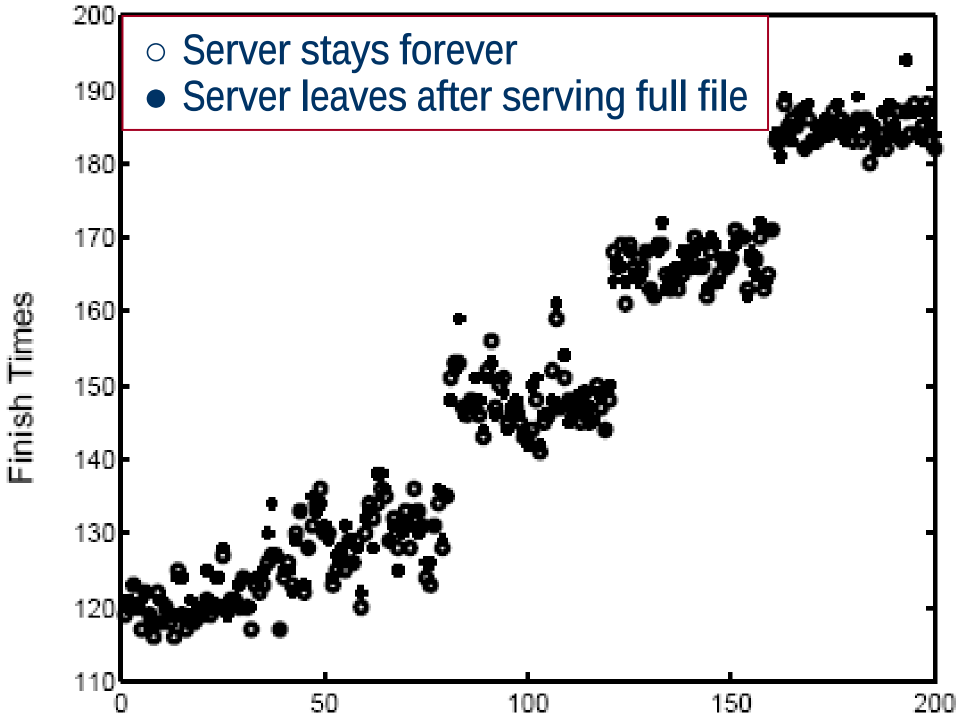


Network coding

- **40% improvement to source coding**
- **200% improvement to no coding**

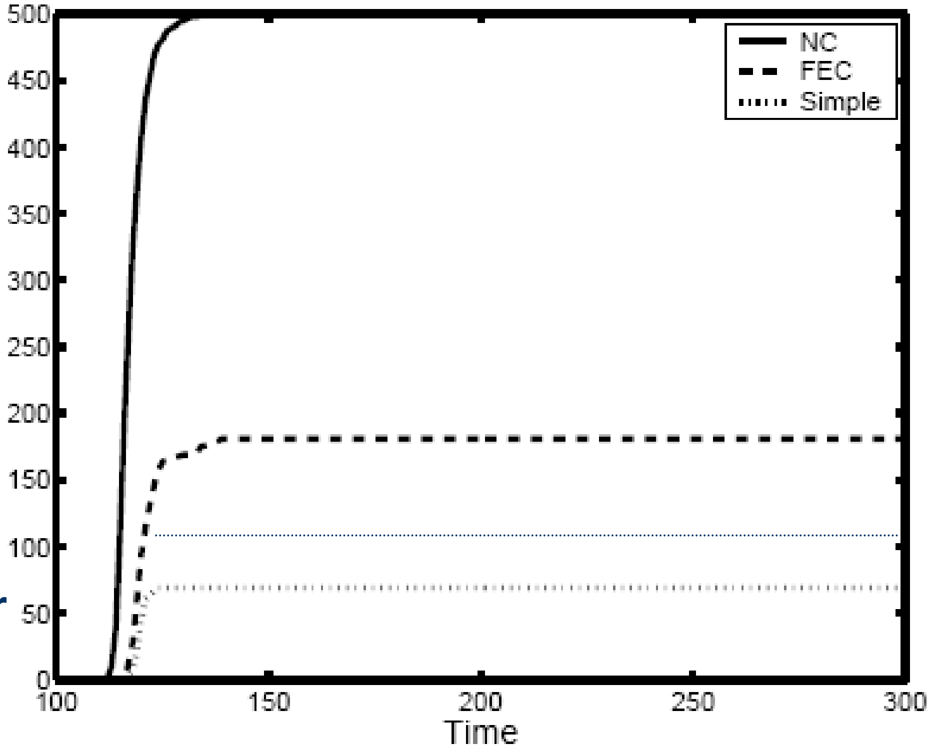
Nodes arrive in batch of 40 nodes every 20 rounds and stay in the system 10% extra rounds

Robustness to Node Departures

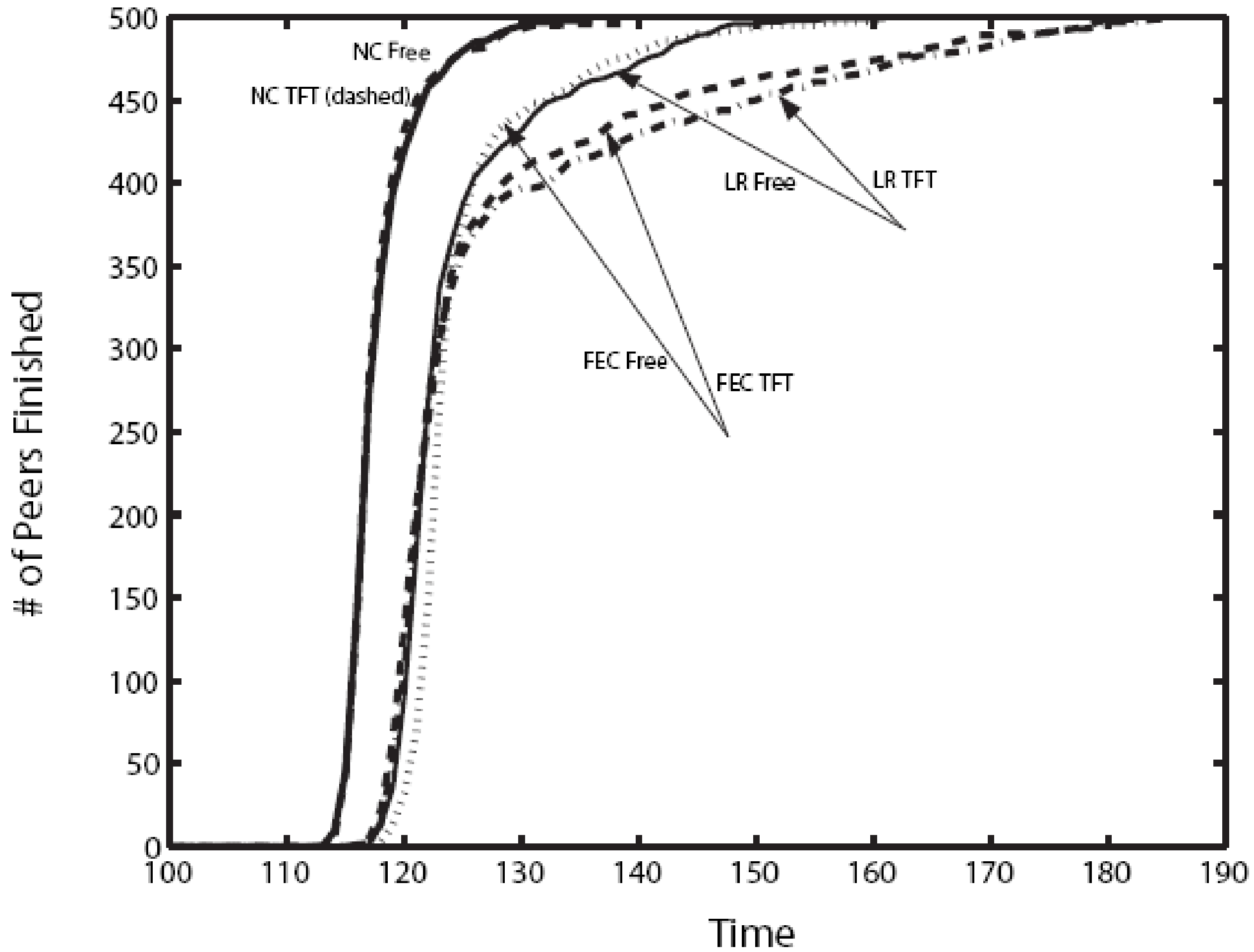


- Nodes finish roughly at the same time independent of server behavior

- 500 nodes in the network
- The server stays for 5% extra rounds
- Nodes leave immediately after downloading the full file



Incentive Mechanisms: Tit-for-tat



Summary and Discussion

- Main advantage of network coding
  - Choosing the correct block of file to download from other nodes is difficult without global information
  - With network coding, each generated block is a combination of all blocks available to the transmitter, if any of them is useful downstream, the generated block is useful
- Network coding performs better
  - when nodes have heterogeneous access capacities
  - when node arrivals and departures are not synchronized
  - when there are natural bottlenecks in overlay topology
  - When incentive mechanisms are in place to discourage free riders

Summary and Discussion

- Design and implementation issues
  - Speed of encoding and decoding
    - O(*k*) operations in encoding, *k* is number of blocks
    - Invert a *k*x*k* matrix in $O(k^3)$ and reconstruct the original file in $O(k^2)$ operations (reconstruction cost dominates the running time because it involves reads from HD)
  - Protection against malicious nodes
    - A malicious node can introduce arbitrary blocks in the system and make the reconstruction impossible
    - Nodes may not perform coding