Easwari Engineering College
Department of Computer Science and Engineering

A Network Coding Equivalent Content Distribution Scheme for P2P-VoD Streaming

Project Members:
L. Karthik Chandran
M. Karthik
Karthika Sindhu

Guide:
Mrs. N. Senthamarai, M.E., (Ph.D)
A.P. Sr. Gr.

Objectives

- To solve the problems with content distribution in P2P networks
  - To reduce delays caused by random access (assuming that the network quality is ideal).
  - To facilitate video access by new peers even if initial parent node is unable to respond.

Motivation

- Video streaming services are on the rise
  - P2P technology, especially, is being used more often to share larger files such as videos.
  - The algorithm proposed in this paper can reduce delays and improve seek performance in videos.

Literature Survey (1)

Distributed Storage to Support User Interactivity in Peer-to-Peer Video Streaming

Ken Yiu, W.-P., Xing Jin ; Gary Chan, S.-H. [IEEE 2006]

Techniques/algorithms/Approaches used :

- Distributed Video Storage using DHT and VMesh

Performance Achieved :

- Pointers to peers which are storing video segments far away from the current playback position can help a user seek a far away position more efficiently.
- The seeking latency is very low.

Drawbacks :

- If a node loses some segments because of background traffic, it needs to request for retransmission from its parent.
- In the case of a large group of users, the number of clusters is very large and the search cost is high.

Literature Survey (2)

A Dynamic Skip List-based Overlay for On-Demand Media Streaming with VCR Interactions

Dan Wang, Jiangchuan Liu [IEEE 2008]

Techniques/algorithms/Approaches used :

- MULTI-SUPPLIER DATA SCHEDULING :

Network Coding based Data Scheduling.

Performance Achieved :

- Dynamic Skip List (DSL), for on-demand overlay media streaming.

Drawbacks :

- Layers are not well-balanced, which leads to inaccurate jumps and hence more segment losses.
- Under high speeds, the quality of the system is much worse.

Literature Survey (3)

P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment

Do, T.T., Hua, K.A. ; Tantaoui, M.A. [IEEE 2004]

Techniques/algorithms/Approaches used:

- Join Algorithm
- Parent selection using:
  - Round Robin Selection
  - Smallest Delay Selection
  - Smallest Distance Selection

Performance Achieved:

- Allows clients to join the system faster and also requires less time for an affected client to recover from a failure

Drawbacks :

- High Seek Latency
- High Segment miss rate

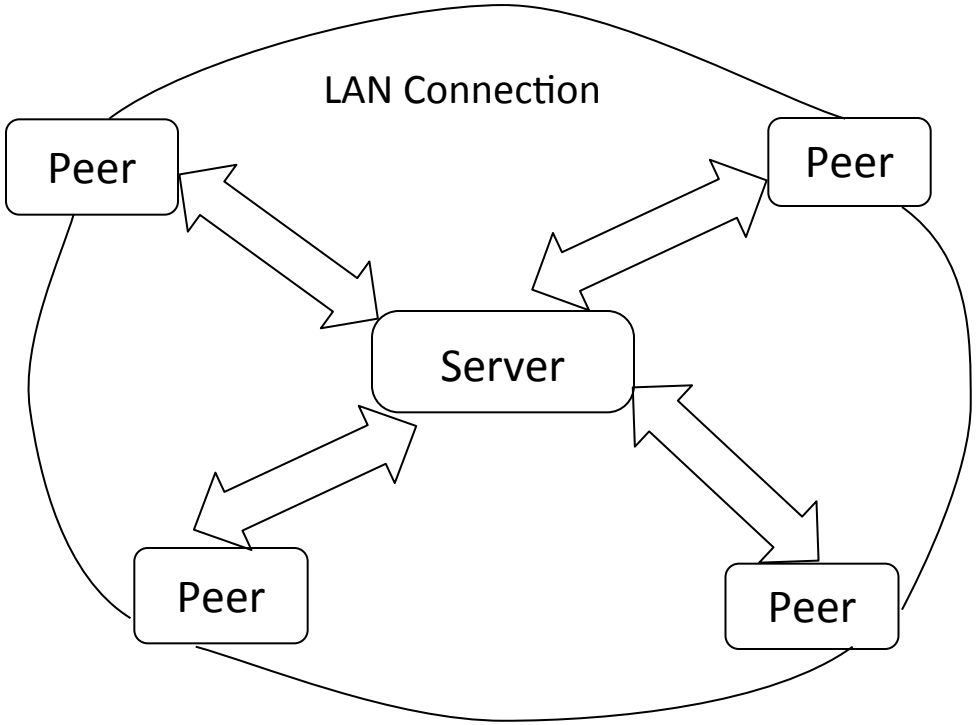
Issues in the Existing Systems

- Search cost is High
- Jump performance is poor
- Startup time is long
- Needs to maintain index for locating parent peers

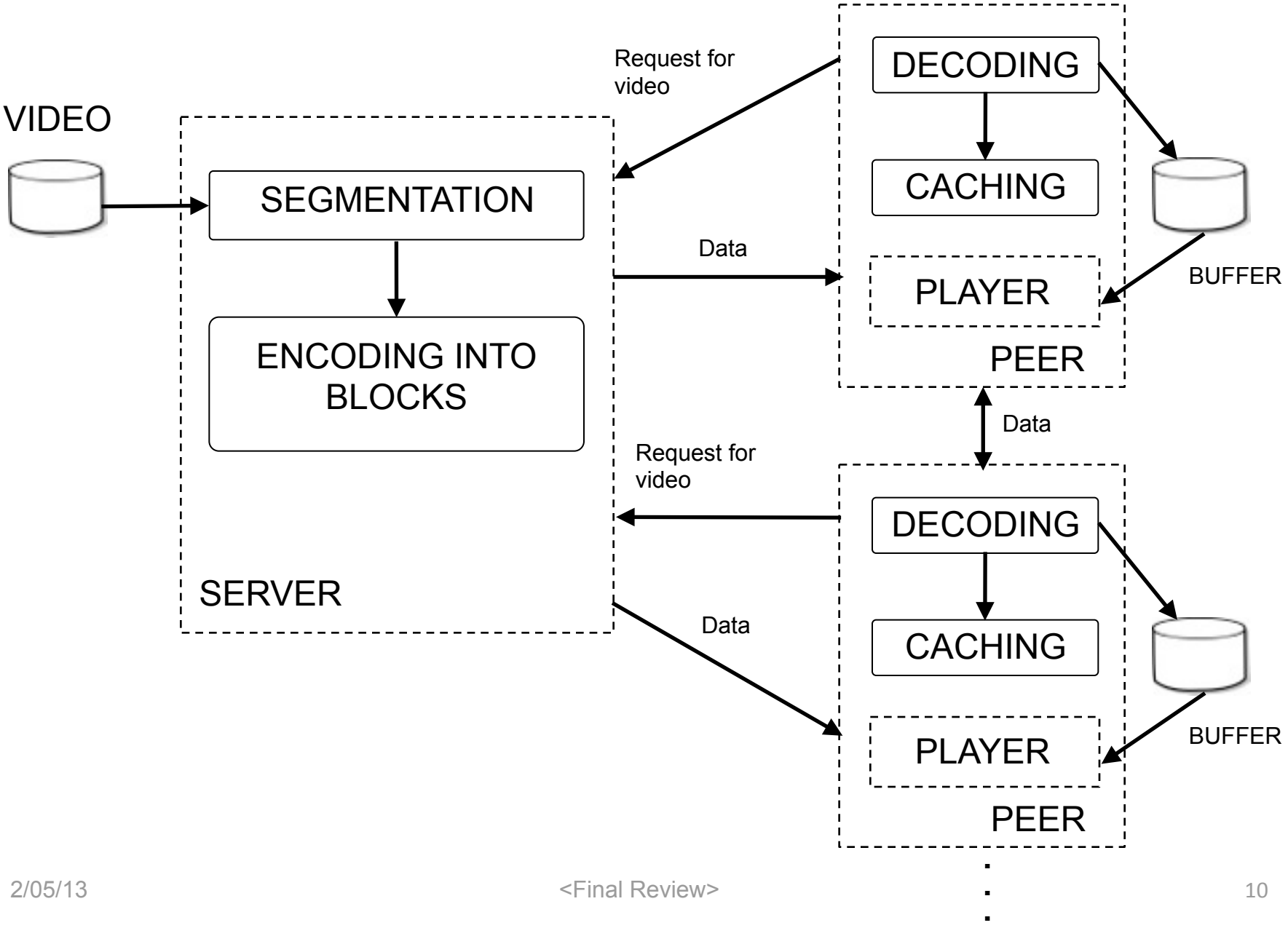
Proposed System

- Child peer need not search for new parent peers to view next segment.
- With sufficient encoded blocks, the original segment can be decoded.
- Seek and jump performance is vastly improved.

System Architecture



Functional Architecture



List of Modules

1. Segmentation & Encoding
2. Retrieval & Playback

Module 1
Segmentation & Encoding

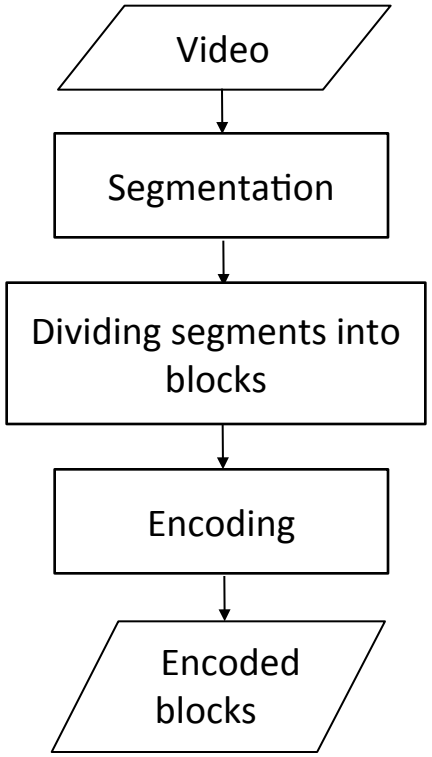
- Input : Video
- Divide video into X segments
- Further, divide each segment into blocks b using Interleaving scheme
- Encode the blocks using a random co-efficient vector using NCECD scheme
- Output : Encoded blocks

FLOWCHART

Segmentation & Encoding

Basic interleaving scheme

NCECD scheme



Segments
(X1,X2…Xn)

Blocks
($b_{X,1}$; $b_{X,2}$… $b_{X,m}$)

Encoded blocks
Using random coding coefficient vectors
$E_{bX,i} = \sum_{j=1}^{m} c\,(i,j)\,.\,b\,(X,j)$

$f_i = [c_{i,1},\ c_{i,2}\ldots.]$

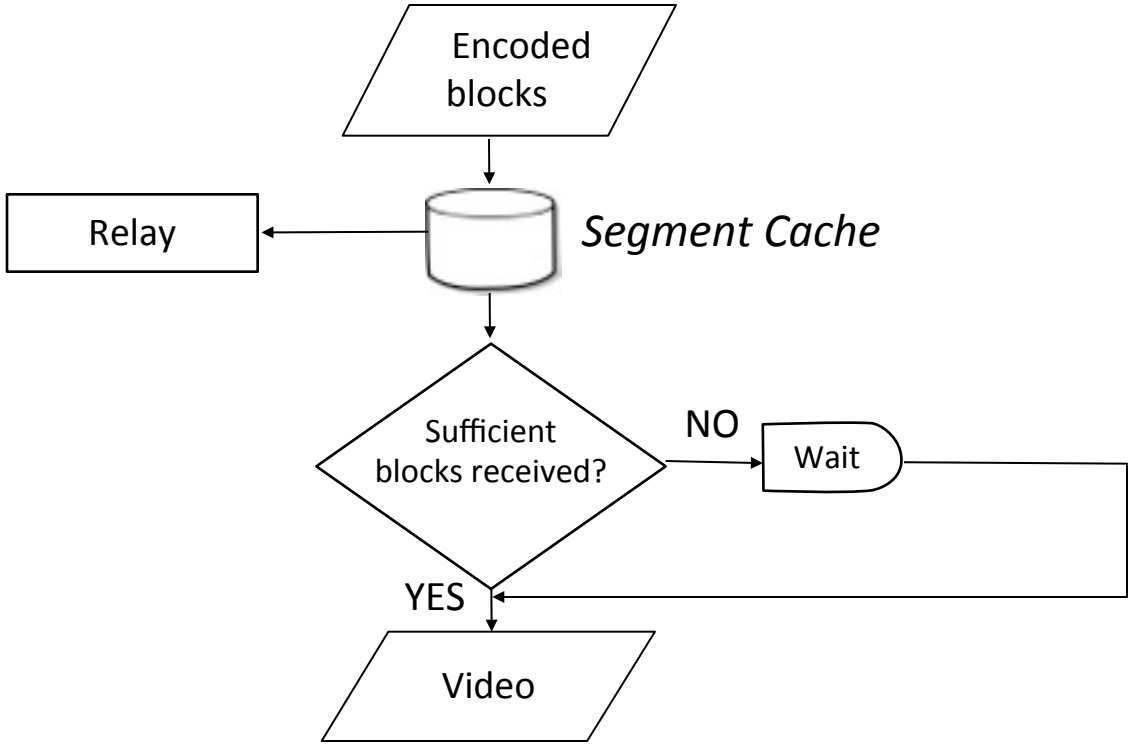
$F = M \times M$
$E = [E_{bX,1}\ ;\ E_{bX,2}\ \ldots..]$
Recovered using $X = F^{-1}\,E^{T}$

Module 2
Retrieval and Playback

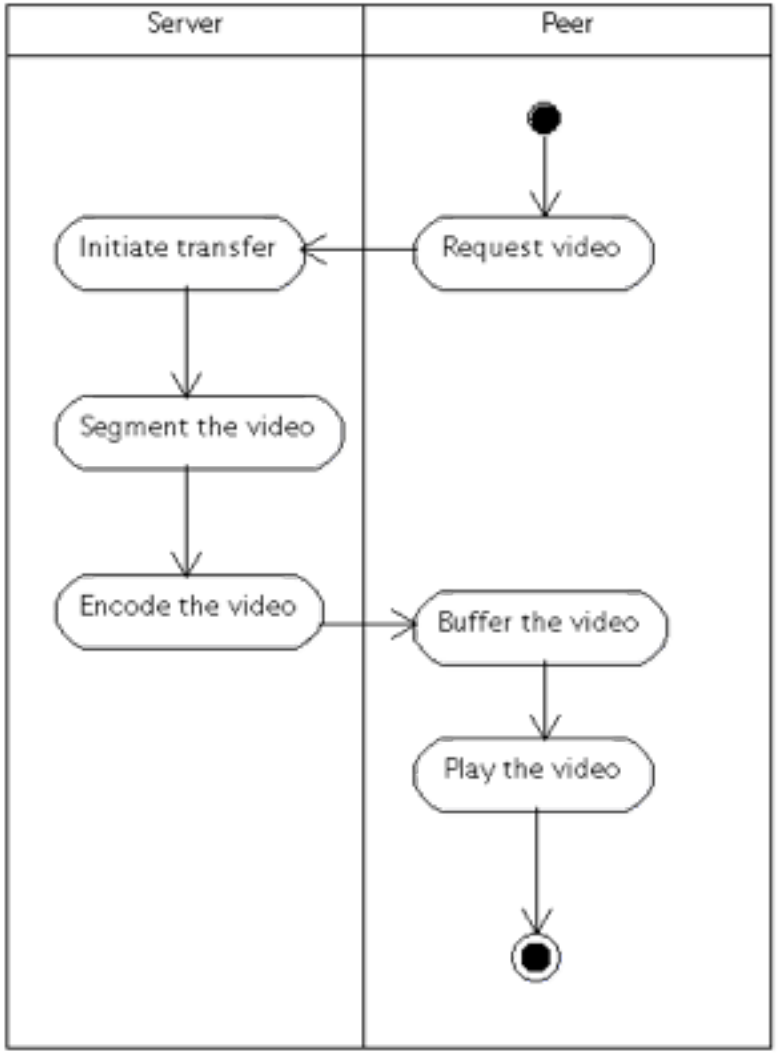
- Input : Encoded blocks
- Retrieve the desired segments
- Store other blocks in local cache and relay when required
- When sufficient blocks are available, start playback
- Decode the video as its buffered
- Output : Video playback

FLOWCHART

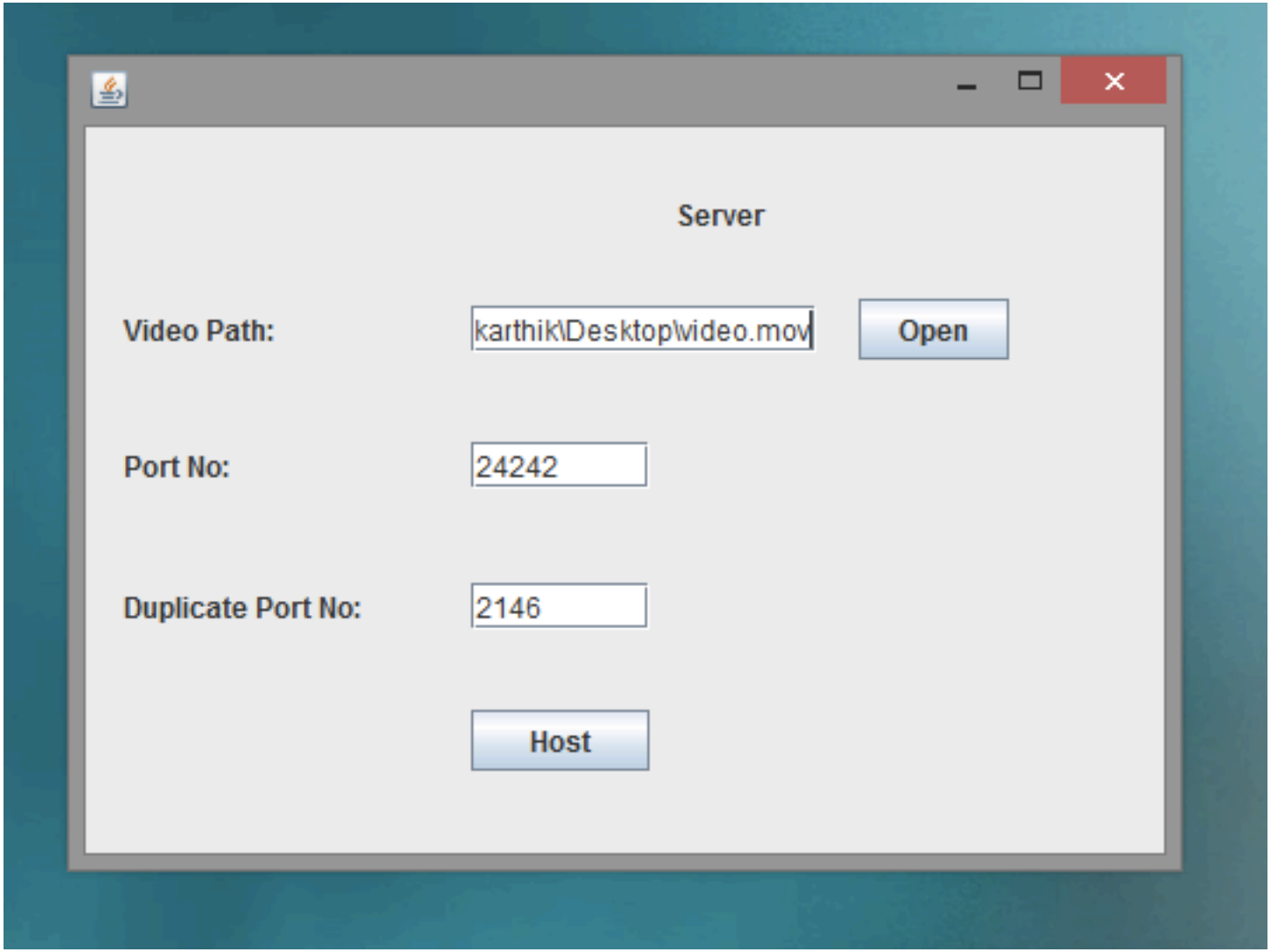
Retrieval and Playback



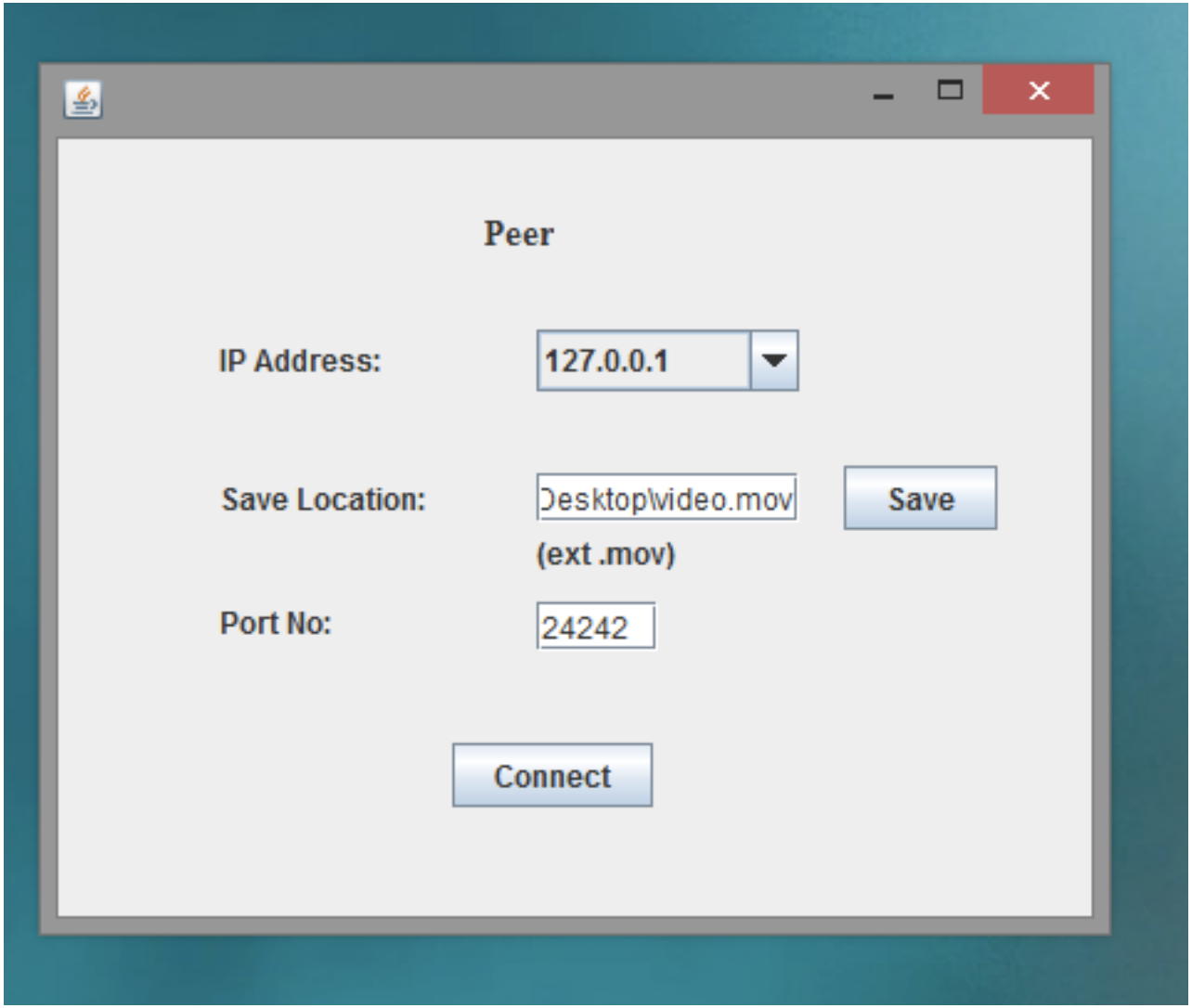
Activity Diagram



Screenshots

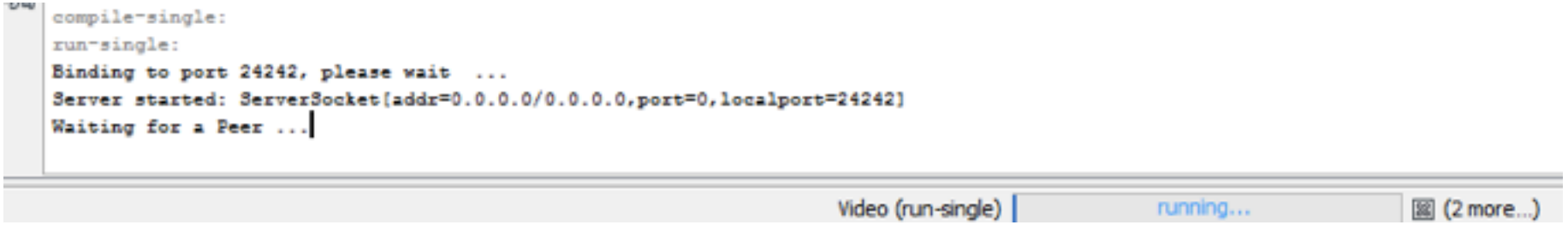


Screenshots



Screenshots

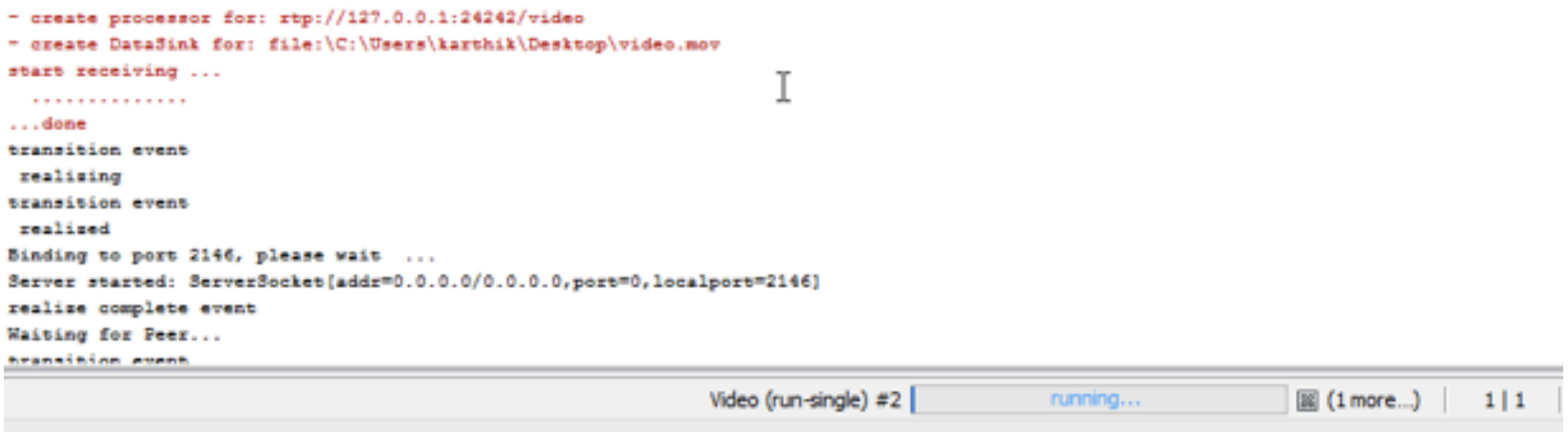
Before hosting video



After hosting video

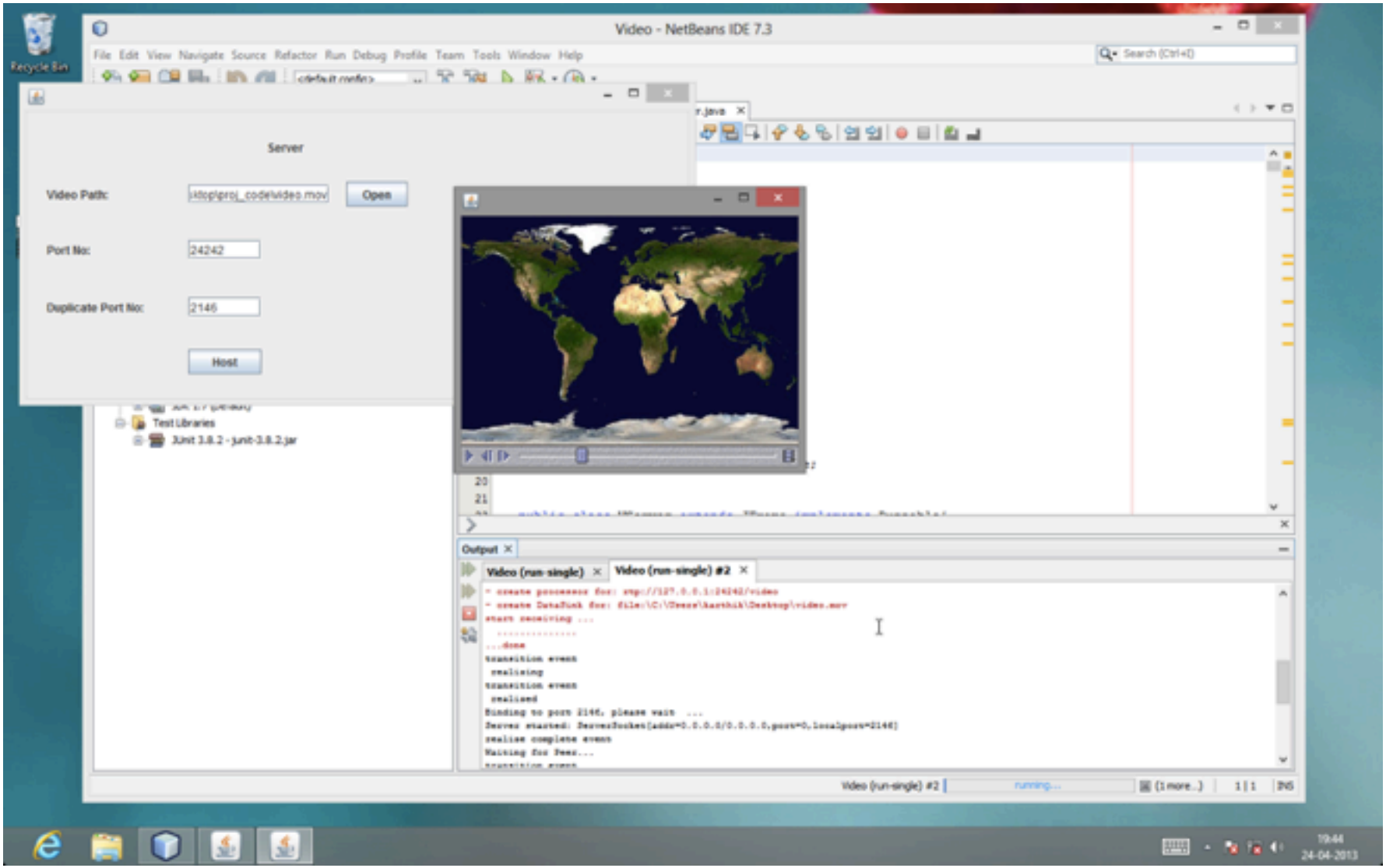
Screenshots

Peer receiving video



Screenshots

Final video playback



System Requirements

- OS Platform : Windows 7/8
- Development tools required : NetBeans 7
  - Language Platform : Java (JDK 1.7)
  - Additional software : JRE, JMF package
- Additional H/W Req.: LAN connection
- System : Intel Core processor, 1 GB RAM

References

1. Yung-Cheng Kao, Chung-Nan Lee, Peng-Jung Wu, and Hui-Hsiang Kao, (2012), “A Network Coding Equivalent Content Distribution Scheme for Efficient Peer-to-Peer Interactive VoD Streaming”, IEEE Transactions On Parallel And Distributed Systems, IEEE, Vol. 23, Issue: 6, pp: 985-994
2. Dan Wang, Jiangchuan Liu, (2008), “A Dynamic Skip List-based Overlay for On-Demand Media Streaming with VCR Interactions”, IEEE Transactions on Parallel and Distributed Systems, IEEE, Vol. 19, Issue: 4, pp: 503 - 514
3. Do, T.T., Hua, K.A., Tantaoui, M.A., (2004), “P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment”, In the proceeding of “IEEE International Conference on Communications, 2004”, pp: 1467 - 1472 Vol.3
4. Ken Yiu, W.-P., Xing Jin, (2006), “Distributed Storage to Support User Interactivity in Peer-to-Peer Video Streaming”, In the proceeding of “IEEE International Conference on Communications, 2006 (ICC '06)”, pp: 55- 60

Thank You

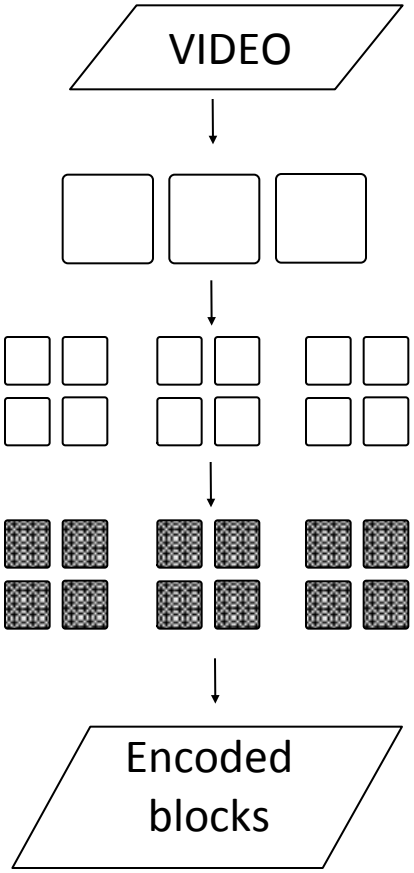
FLOWCHART

Segmentation & Encoding

Input

Basic interleaving scheme

NCECD scheme



Segments
(X1,X2…Xn)

Blocks
($b_{X,1}$; $b_{X,2}$… $b_{X,m}$)

Encoded blocks
Using random coding coefficient vectors

$$E_{bX,i} = \sum_{j=1}^{m} c\,(i,j)\,.\,b\,(X,j)$$

$$f_i = [c_{i,1},\ c_{i,2}\ldots.]$$

$$F = M \times M$$

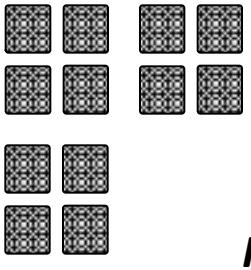
$$E = [E_{bX,1}\ ;\ E_{bX,2}\ \ldots..]$$

$$\text{Recovered using } X = F^{-1}\,E^{T}$$

Output

FLOWCHART

Retrieval and Playback



Input

Peer

Peer

Peer

Peer

P2P N/W

Relaying and obtaining segments

Encoded Blocks

Segment Cache

Is sufficient data Received?

Video Playback

Decoded as buffered

Output