A HYBRID PULL-PUSH PROTOCOL IN HYBRID CDN-P2P MESH-BASED ARCHITECTURE FOR LIVE VIDEO STREAMING

Teerapat Sanguankotchakorn,Ph.D
Nonpawich Krueakampliw

School of Engineering and Technology
Asian Institute of Technology
Thailand

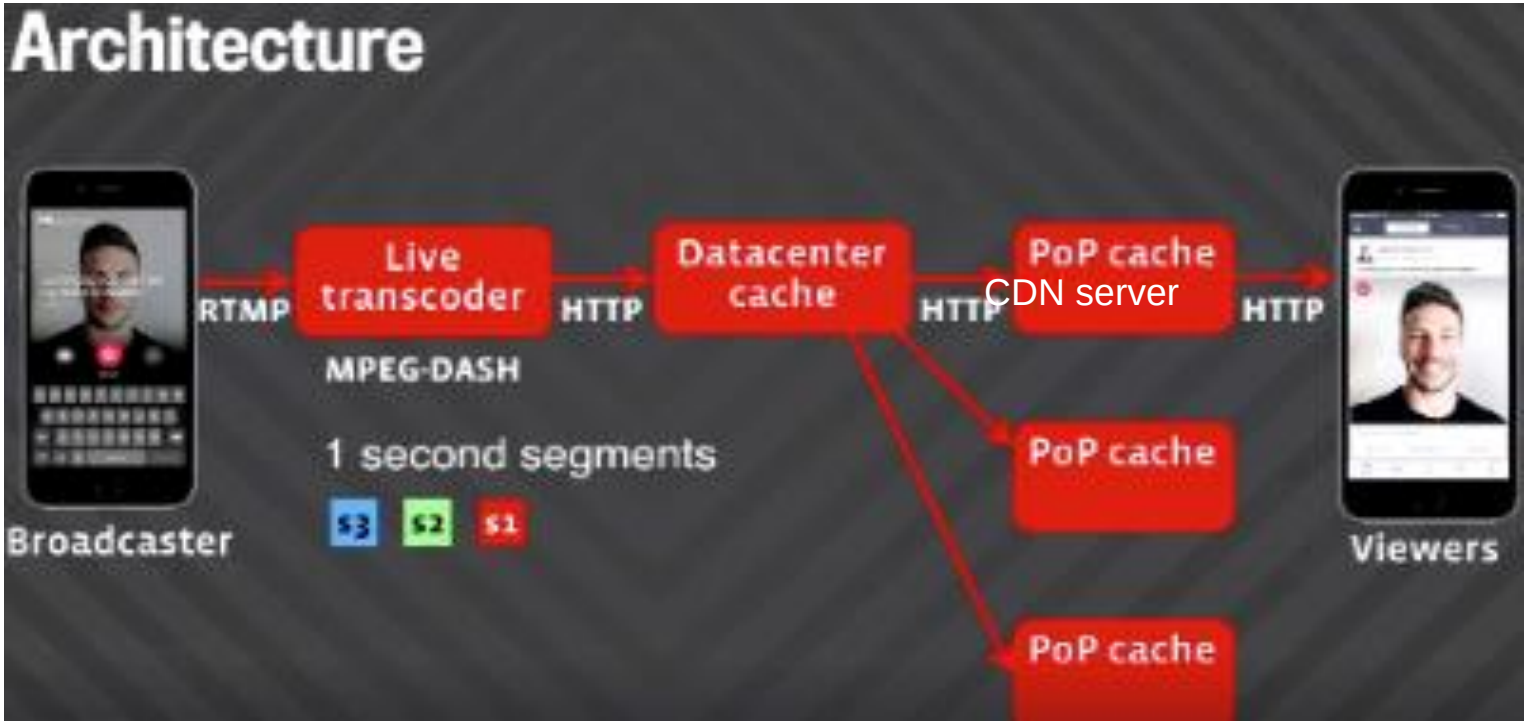
The 19th Asia-Pacific Network Operations and Management

OUTLINE

- Introduction
- Background & Literature Review
- Methodology
- Network Simulation & Results
- Conclusion

INTRODUCTION

Overview


Architecture
RTMP
Live transcoder
MPEG-DASH
HTTP
Datacenter cache
HTTP
PoP cache
CDN server
HTTP
PoP cache
PoP cache
1 second segments
s3
s2
s1
Broadcaster
Viewers

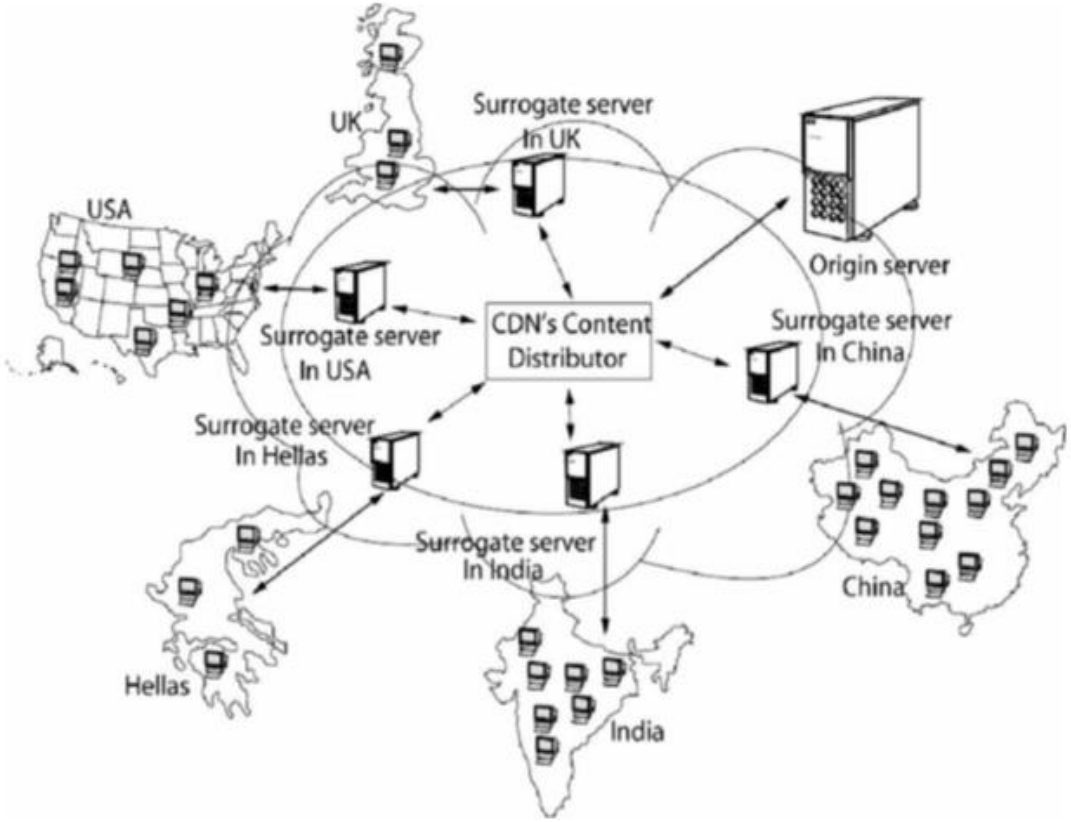
Problem

- P2P is widely used in live video streaming because of its potential in providing streaming at large scale.
- However, it causes delay from dynamic change of peers (tree based) and video retrieval (mesh-based).
- CDN is adopted to solve the problem of delay, because it brings the content closed to the users.
- To build an infrastructure with multiple CDN nodes is extremely expensive and difficult to maintain.
- Hybrid CDN–P2P (simple mesh) can help to reduce the cost of CDN.
- Protocol used in Hybrid CDN-P2P causes some end-to-end delay.
- Hybrid pull-push method in CDN-P2P Mesh based architecture will be proposed to reduce delay.

BACKGROUND

Background

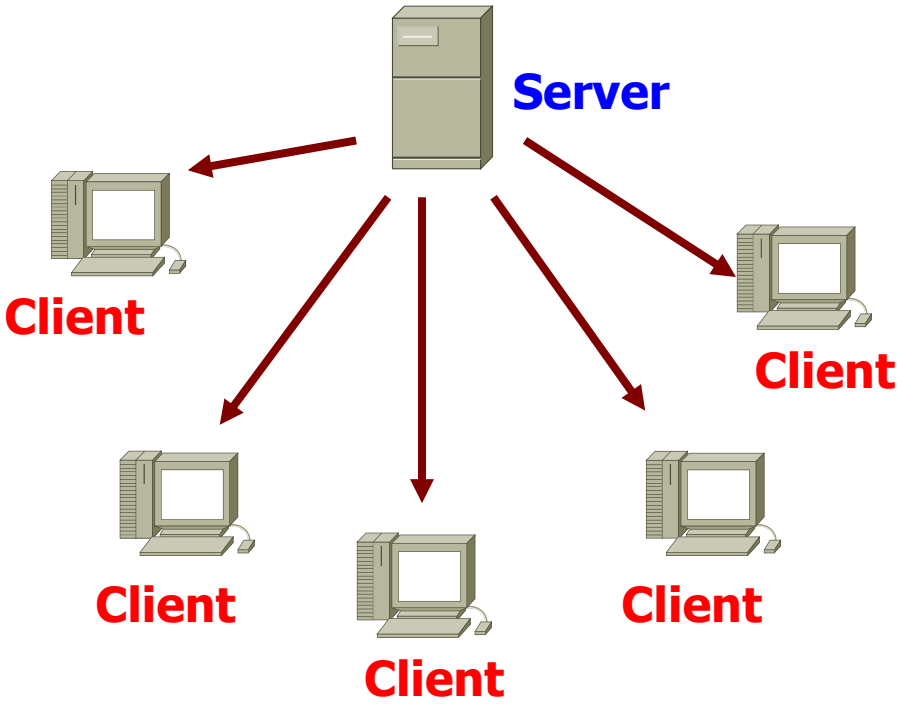
Content Delivery Network (CDN)



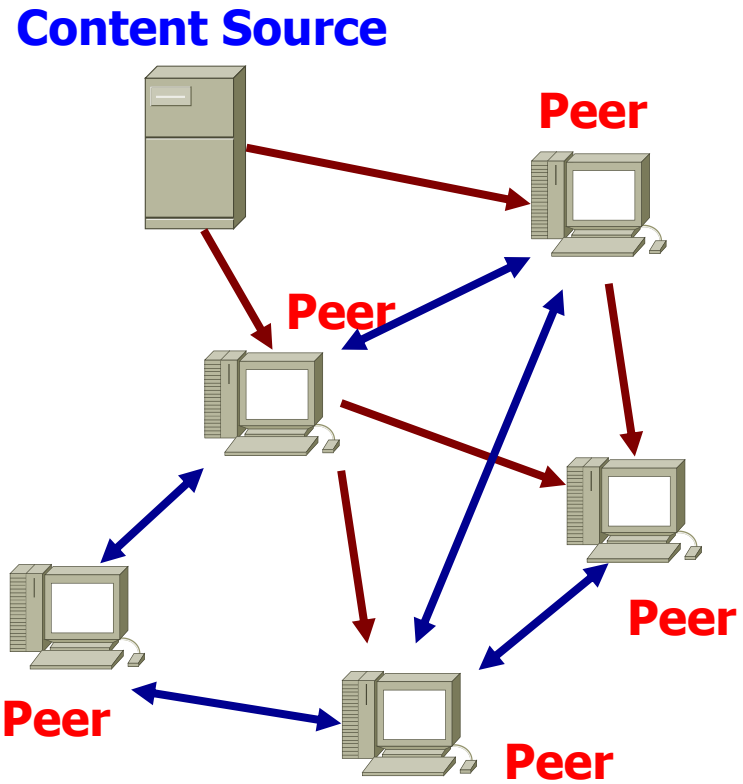
Typical Functionalities:

- Request redirection and content delivery services
- Content outsourcing and distribution services
- Management services

P2P vs Client-server Architecture

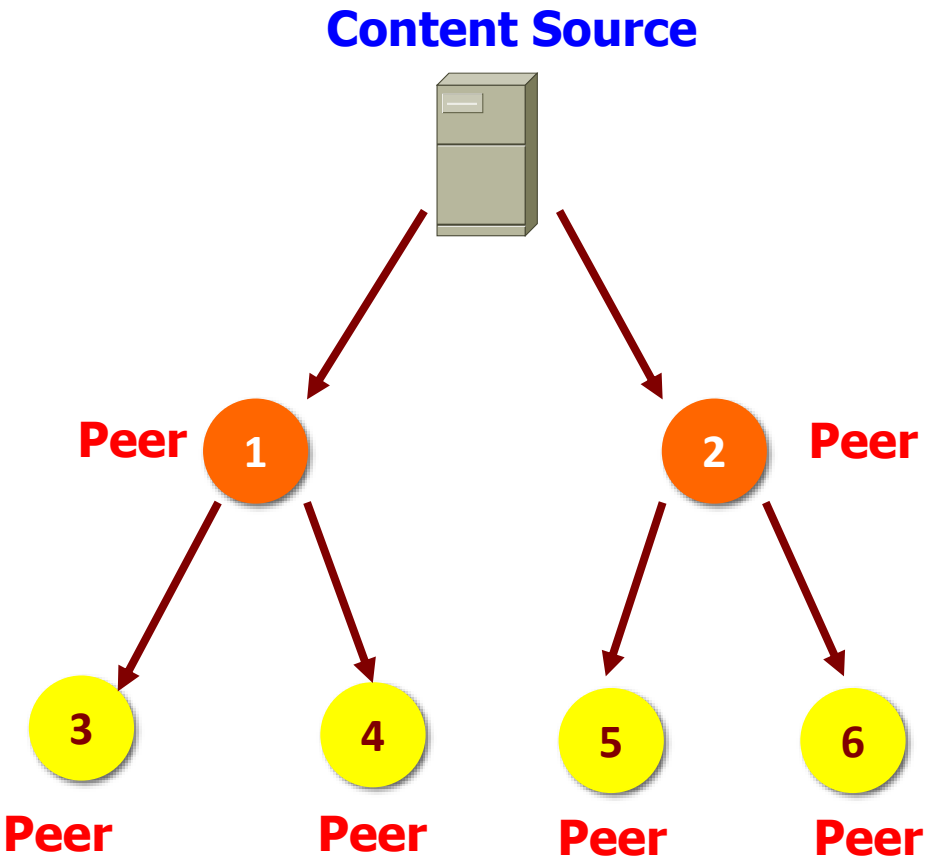


Client-Server Model



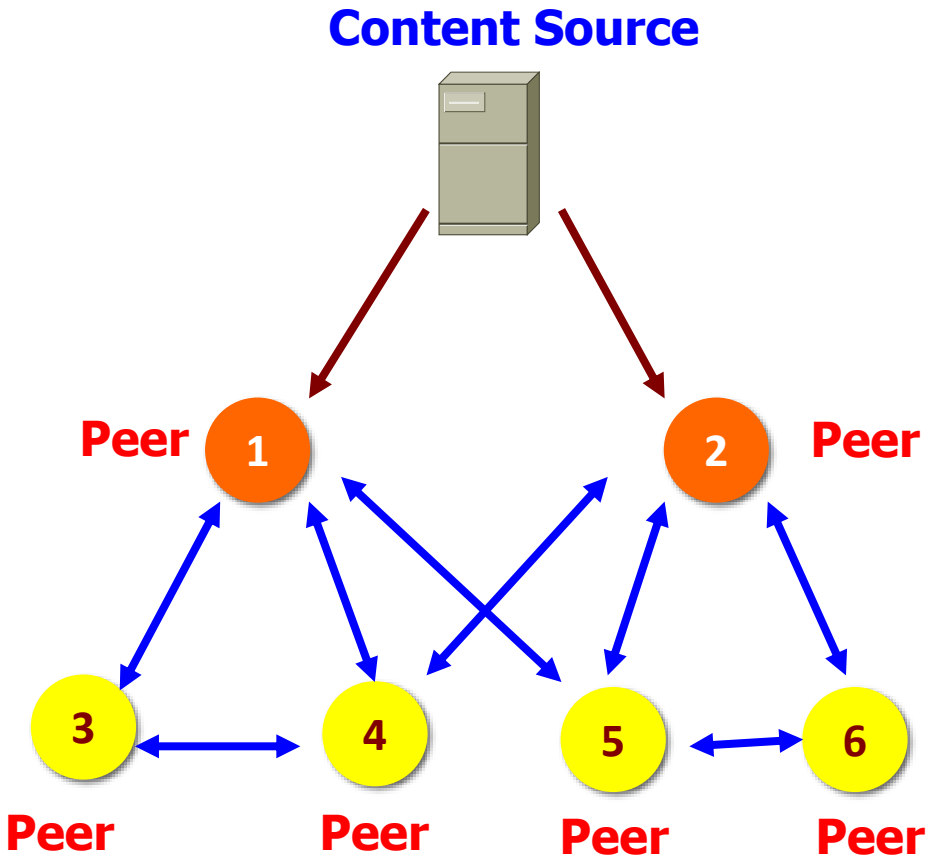
Peer-to-Peer Model

Structure of P2P: Tree - based Streaming



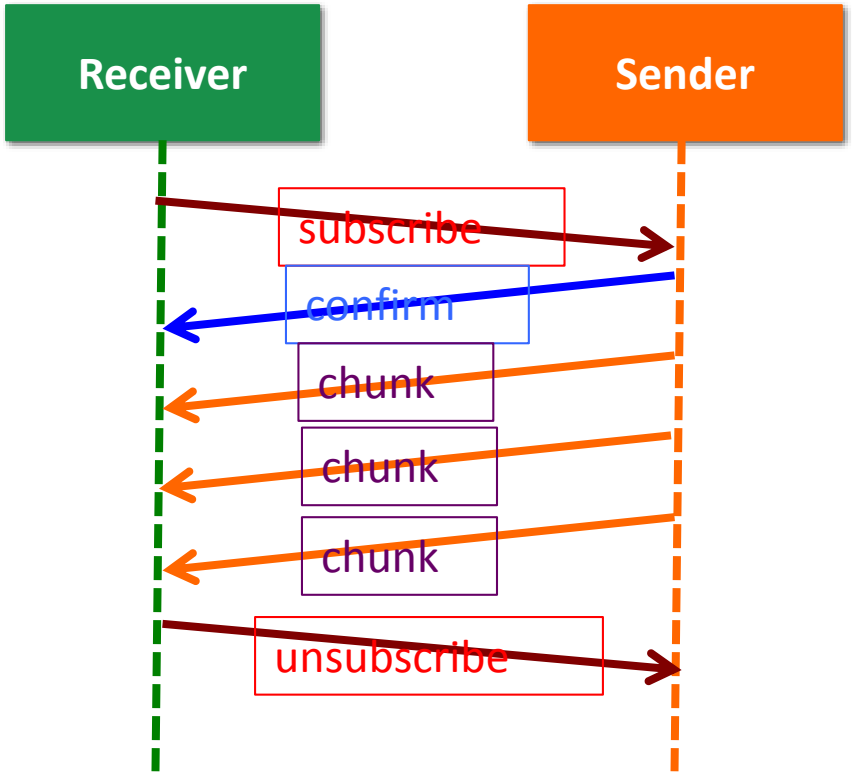
- Peers are organized in tree.
- Use server as a root.
- Each node has as many children as its capacity.
- Construction of tree determined by
  - E2E latency between peers
  - Available bandwidth
  - Underlying physical topology
  - Etc.
- Use Push method to transfer data.
- Has low delay.
- Single point failure problem.

Structure of P2P: Mesh - based Streaming



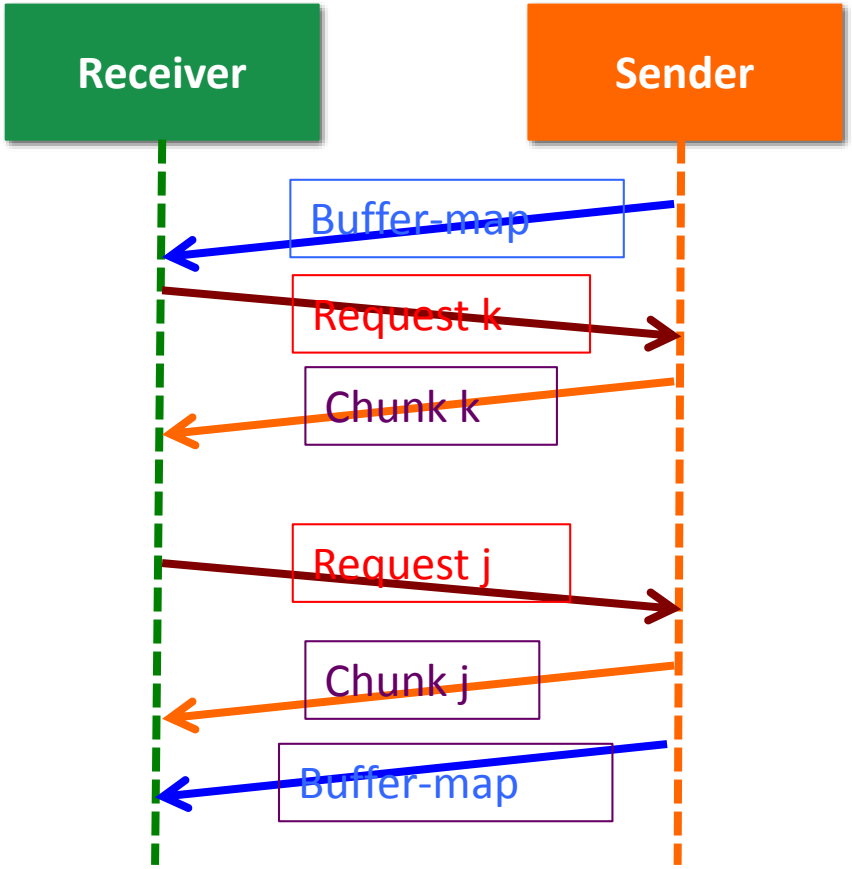
- Maintain large number of incoming connections, to overcome the potential neighbor departure.
- Peers receives randomly generated list of peers already connected to the overlay and tries to establish connections to them.
- Use Pull method to request necessary data from a number of neighbor nodes.
- Require large buffers to support pull data from neighbors.

Data Distribution in P2P: Push Method



- Nodes have to be arranged in tree structure.
- When child node finish subscription to parent node, selected chunk is pushed from a parent node to its child node.
- Delay occurs when a node leaves.

Data Distribution in P2P: Pull Method



- Used in mesh-based structure.
- Each peer has a buffer-map of the frames currently held by it and its neighbors.
- Peers request neighboring peers for the next frame needed.
- A Peer decides from whom to pull data based on its neighbors buffer-map.
- Decision is made based on the connection with highest bandwidth.
- Making requests of pull again and again and exchanging buffer maps can result in congestion.

Background

Comparison Tree-based and Mesh-based (Chin Yong Goh, Hui Shyong Yeo, Hyotaek Lim, 2012)

- Tree-based approach
  - More stable video delivery quality,
  - Low playback delay and end-to-end delay.
  - Under dynamic peer churn, the peer replacement and recovery mechanism consume certain amount of time.

- Mesh-based approach
  - More feasible toward dynamic peer churn environment.
  - Peers have more resources to choose.
  - The most popular approach for video streaming because of capability to create overlay, which can quickly accommodate the dynamic change of peers.

Previous work on hybrid CDN-P2P network

Author	Paper	Summary
D. Xu et al., 2006	Hybrid CDN-P2P architecture to limit the use of the CDN servers	- The architecture is more cost-effective than pure CDN architectures. - The P2P video streaming is faced with jitter - The CDN video streaming is costly - Hybrid CDN-P2P has advantages of P2P systems in cost as well as fast startup in CDN systems.
S.M.Y. Seyyedi et al.,2011	Hybrid CDN-P2P Architectures for Live Video Streaming: Comparative Study of Connected and Unconnected Meshes	- Compared the performance. - CDN-P2P connected mesh architecture has lower end-to-end delay and distortion than others. - CDN-P2P connected mesh has higher startup delay

LITERATURE REVIEW (cont.)

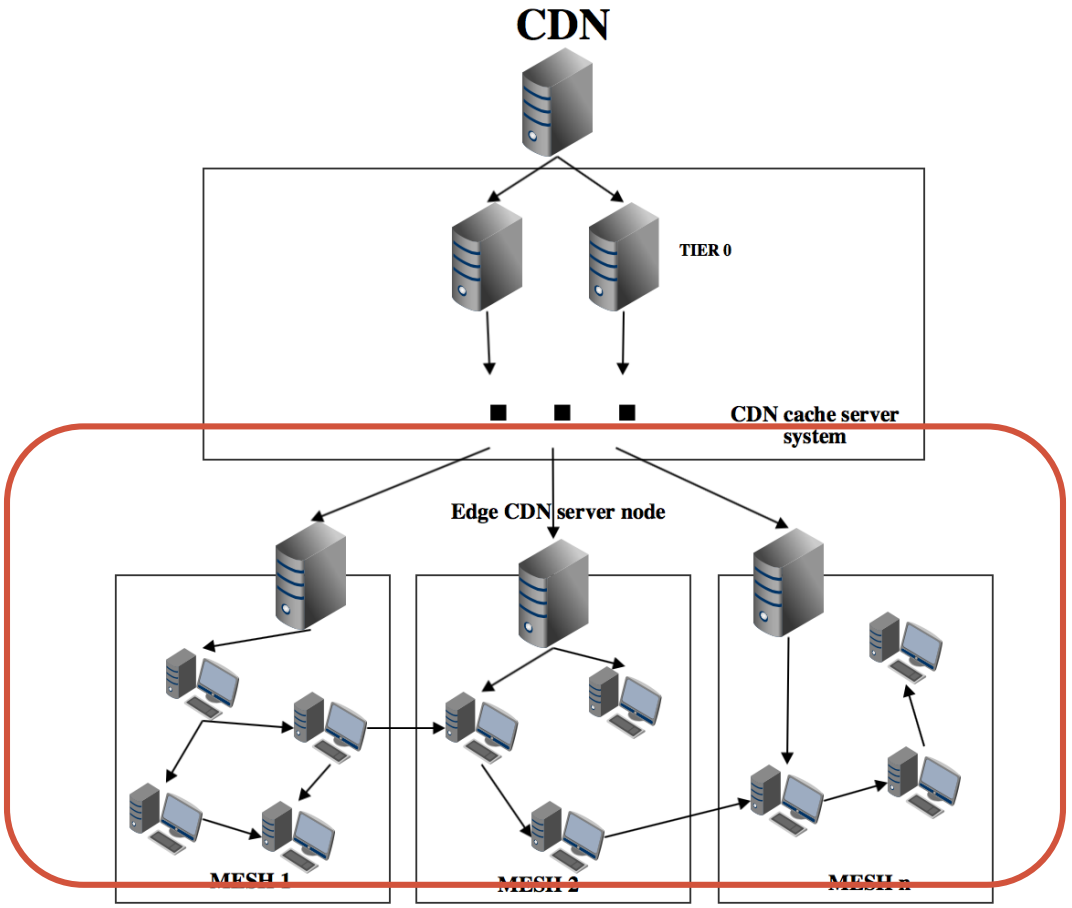
Author	Paper	Summary
Anahita Fellah Jahromi,2012	Temporal Scalable Live Video Streaming over Hybrid CDN-P2P Architecture	- Achieves higher quality by sending 2 layers: base layer, and enhancement layer. - A tracker divides upload bandwidth of peers into three ranges of numbers. - Simulation showed that distortion and hop count is reduced, the startup delay is the same as single layer but E2E delay is smaller.
Tran Thi Thu Ha et al.,2015	Design and Deployment of Low-Delay Hybrid CDN–P2P Architecture for Live Video Streaming Over the Web	- Compare the number of requests sent to CDN server in two cases: the proposed method, and traditional CDN server - Decrease the number of requests to CDN servers, reducing the cost of transmission and enhancing system's scalability

METHODOLOGY

Proposed method

Hybrid pull-push method in CDN-P2P Mesh based architecture

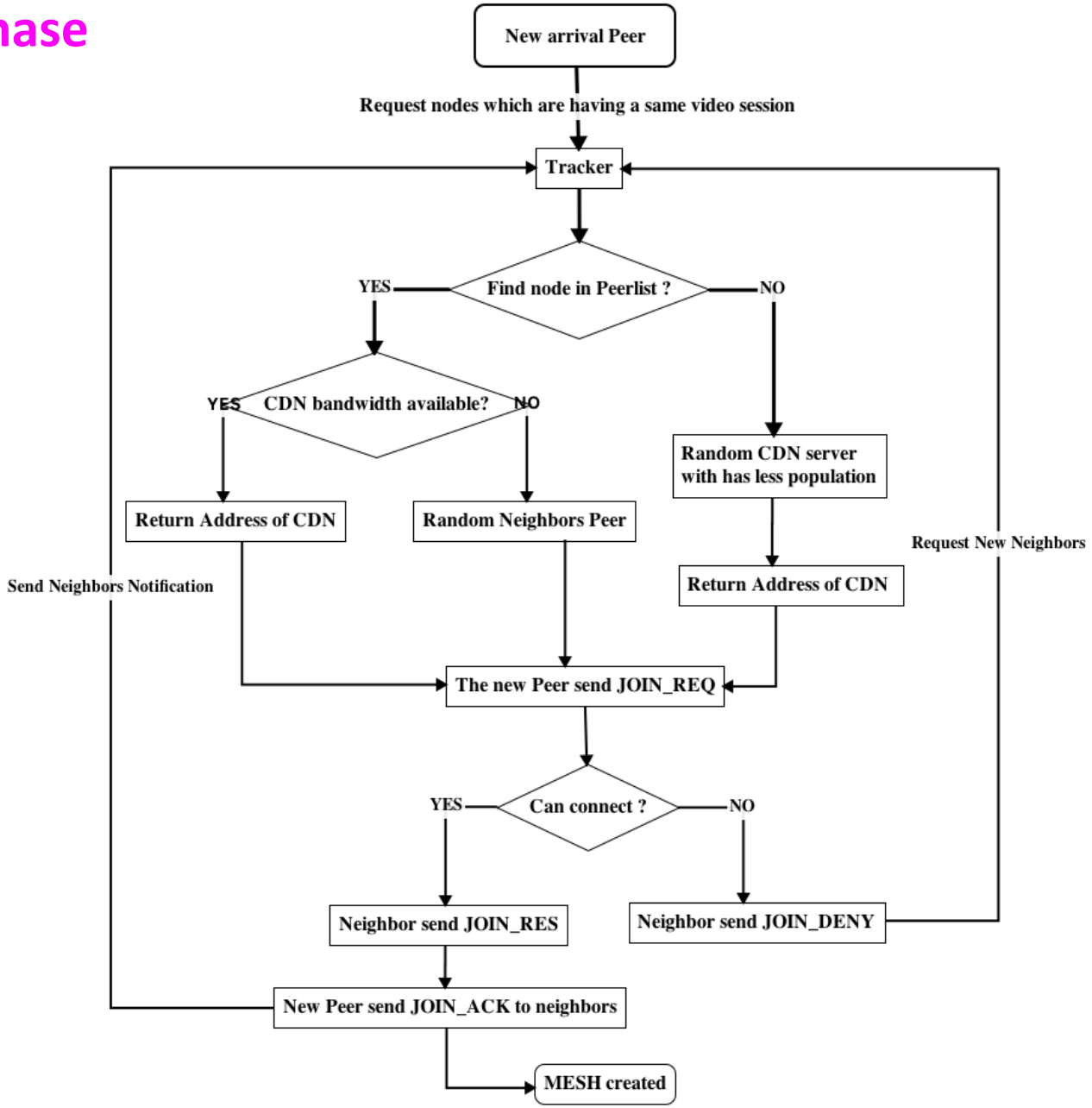
Network Architecture: Hybrid CDN-P2P architecture



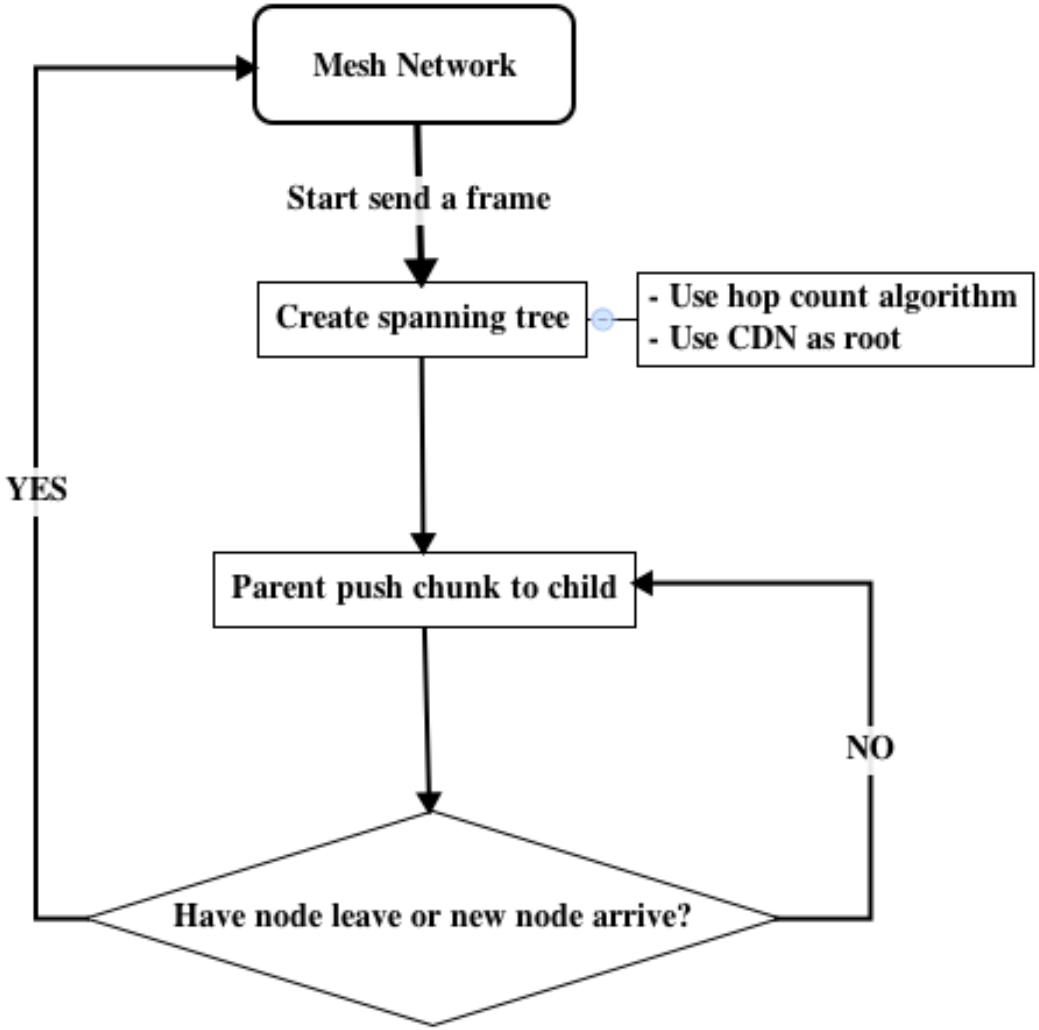
Comparison of the Proposed Method with the existing one

Process	General Method	Proposed Method
CDN-P2P Connection	Simple Mesh	Simple Mesh
Tracker	CDNP2Ptracker	CDNP2Ptracker
P2P Mesh Construction	Pull Method	Pull Method
Frame Distribution	Pull Method	Push Method
Peer Replacement	Pull Method	Pull Method

Pull Method phase

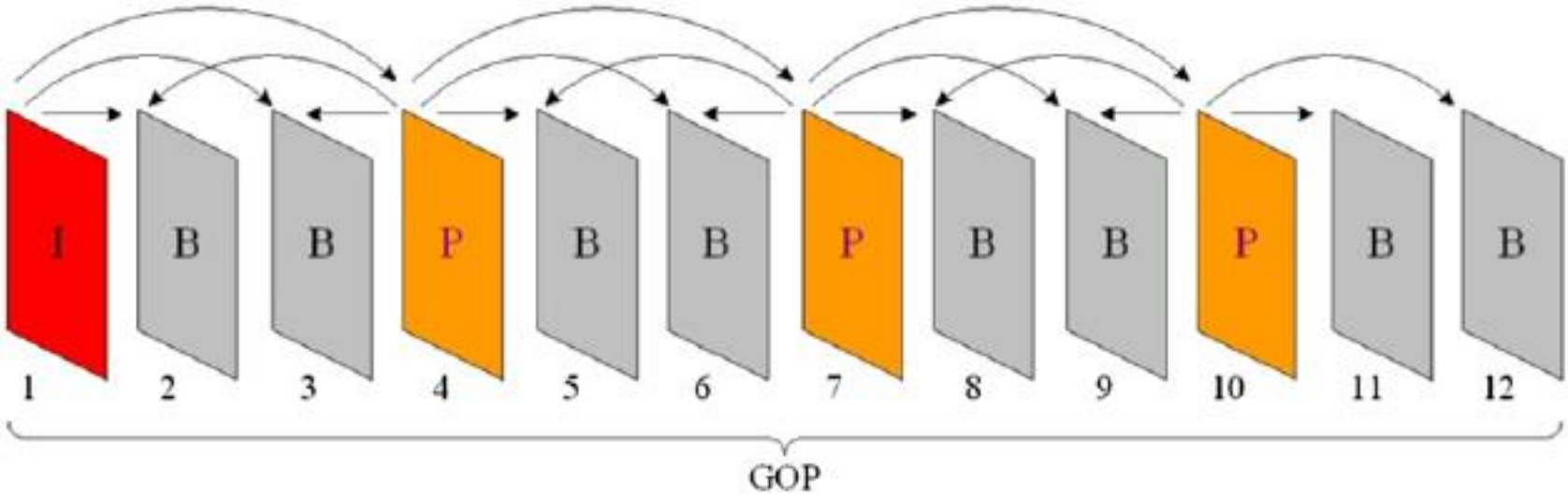


Push Method phase



MPEG Compression Standard

- **Group of Picture (GoP) Structure**



Simulation Result Metric

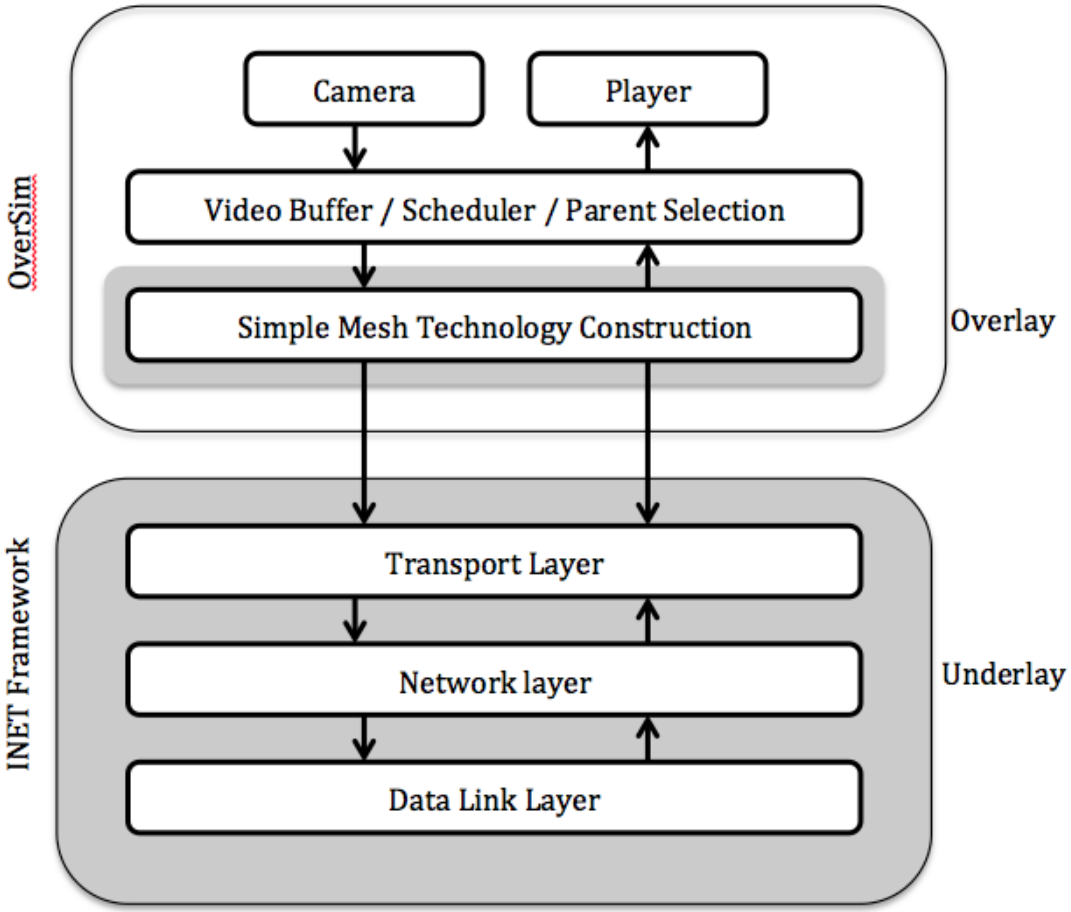
- **Start-up Delay**: The time between connecting to the network and starting video playback.
- **End-to-End Delay**: The time between creating a frame in the source node and playing it in the destination node.
- **Video Distortion**: Percentage of content loss
  - Total error frame size / Total frame size × 100
- **Overhead Control:** A considerable controlling messages, such as buffer-map messages, request and response messages
  - Overhead =

$$\frac{\text{(Total frame/s – Video frame/s)}}{\text{Total frame/s}}$$

NETWORK SIMULATION

Simulation Structure

OMNET++ Environment



NETWORK SIMULATION

OMNET++

- OMNeT++ is an object-oriented modular discrete event network simulation framework.
- Open source
- It has a generic architecture, so it can be used in various problem domains.
- Based on C++.

NETWORK SIMULATION

Overlay Part - OverSim

- OverSim is an open source, fast Peer-to-Peer (P2P) simulation platform.
- Based on the OMNet++ simulation environment.
- OverSim includes many well-known P2P protocols and applications that can be extended and modified.

NETWORK SIMULATION

Underlay Part – INET framework

- INET Framework is an open-source for the OMNeT++ simulation environment

- INET contains models for the Internet stack
  - (TCP, UDP, IPv4, IPv6, OSPF, BGP, etc.)
  - Wired and wireless link layer protocols.
  - MANET protocols, DiffServ, MPLS with LDP and RSVP-TE signaling, and many other protocols and components.

Hybrid CDN – P2P Architecture Creation

Underlay Network

Georgia Tech Internet Topology Model (GT-ITM) Library

To create

28 Backbone routers and 28 Access routers in each Backbone router

- In top-down mode
- All use drop-tail queue.
- Bandwidth (1 Mbps - 2 Mbps)
- Delays (15 ms - 20 ms.)

Hybrid CDN – P2P Architecture Creation

Overlay Network

- General Method Implementation

Module Tracker :	CDNP2PtrackerModule Library

Module CDN :	DenaCast CDN Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4CameraModules
Overlay Type :	SimpleMeshModules

Module Terminal :	DenaCast P2P Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4PlayerModules
Overlay Type:	SimpleMeshModules

Hybrid CDN – P2P Architecture Creation

Overlay Network

- **Proposed Method** Implementation

Module Tracker :	CDNP2PtrackerModule Library
Module CDN :	DenaCast CDN Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4CameraModules
Overlay Type :	SimpleMeshModules ← PullPushModules
Module Terminal :	DenaCast P2P Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4PlayerModules
Overlay Type:	SimpleMeshModules ← PullPushModules

Hybrid CDN – P2P Architecture Creation

Overlay Network

- **Proposed Method** Implementation

Module Tracker :	CDNP2PtrackerModule Library

Module CDN :	DenaCast CDN Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4CameraModules
Overlay Type :	PullPushModule

Module Terminal :	DenaCast P2P Library
Tier 1 :	DenaCastAppModules
Tier 2 :	MPEG4PlayerModules
Overlay Type:	PullPushModule

Hybrid CDN – P2P Architecture Creation

Overlay Type: PullPushModules Creation

- Mesh Construction — SimpleMeshModule
- Tree Construction — NtreeModule
- Data Distribution — NtreeModule

Hybrid CDN – P2P Architecture Creation

List of Libraries and Modules in use

- **Underlay Network: 1 Library**
  - Georgia Tech Internet Topology Model (GT-ITM) Library

- **Overlay Network: 8 Libraries**
  - CDNP2PtrackerModule Library
  - DenaCast CDN Library
  - DenaCast P2P Library
  - DenaCastAppModules
  - MPEG4CameraModules
  - MPEG4PlayerModules
  - SimpleMeshModules
  - NtreeModule

NETWORK SIMULATION

Simulation Environment

- **Operation System:** Window 7 32 bit
- **Simulation Framework:** Omnet++ v.4.6
- **Overlay Framework:** Oversim-20121206
- **Underlay Framework:** inetmanet-2.2

NETWORK SIMULATION

Parameter	Value
Maximum packet size	1000 Byte
Peer-side buffer	40 seconds
Buffer map exchange period	1 seconds
Video codec	MPEG4-Part 1
Video frame rate	25 fps
Number of frames in GoP	12 frames
Average video bitrate	512 Kbps
Number of neighbors	Random (3 , 5)
Node's Bandwidth	Random (764 Kbps , 1.5 Mbps)
Video chunk size	1 frame
Simulation duration	200 seconds
Number of CDN servers in CDN-P2P	7
Peer Node	50,100,150,200
Repeat	20 times

NETWORK SIMULATION

Simulation Architectures

1. P2P Simple Mesh-based

2. General Hybrid CDN-P2P Mesh-based

- CDN-P2P with Pull Method (Simple Mesh)

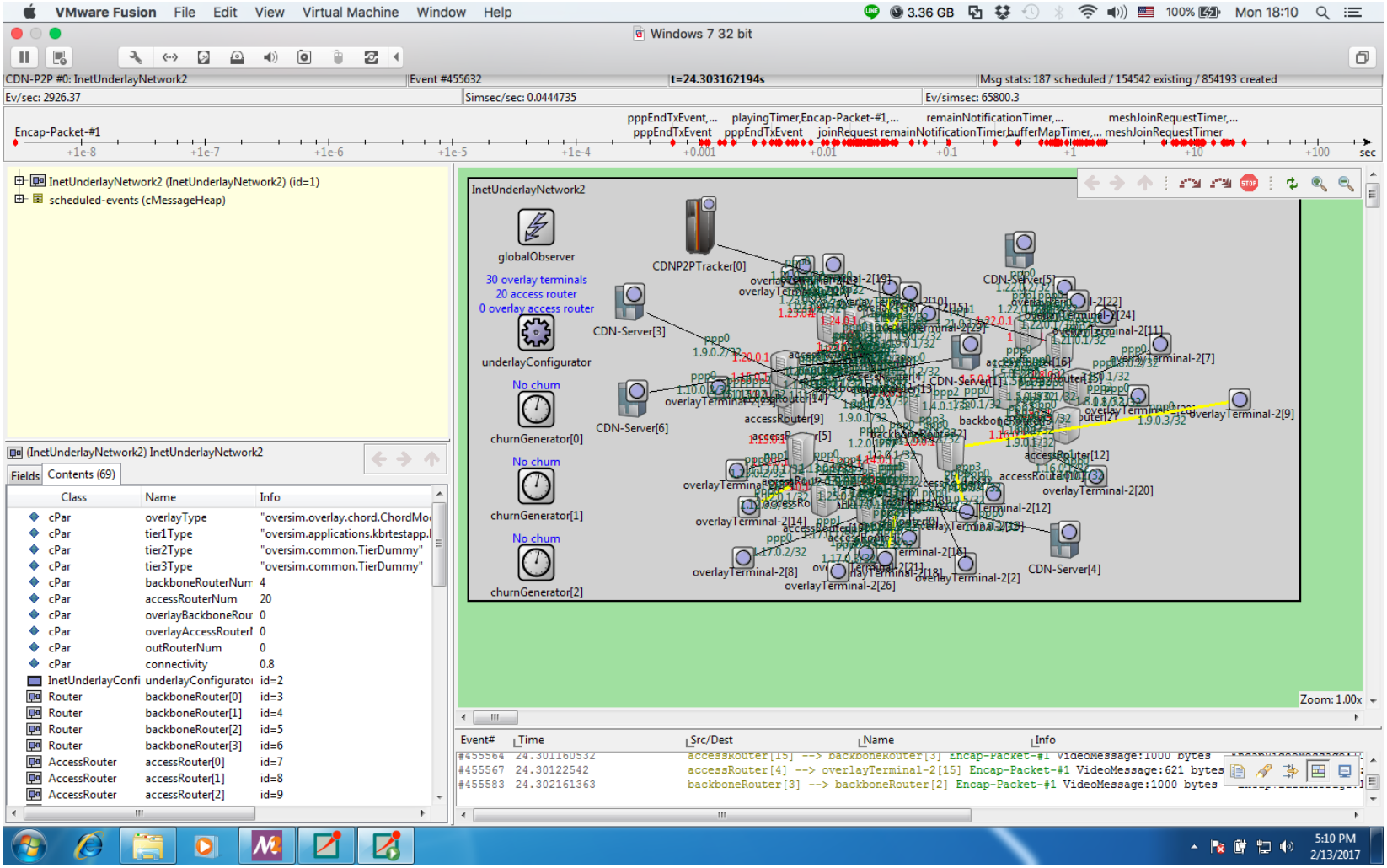
3. The Proposed Method

- CDN-P2P with Pull-Push Method

4. The Proposed Method

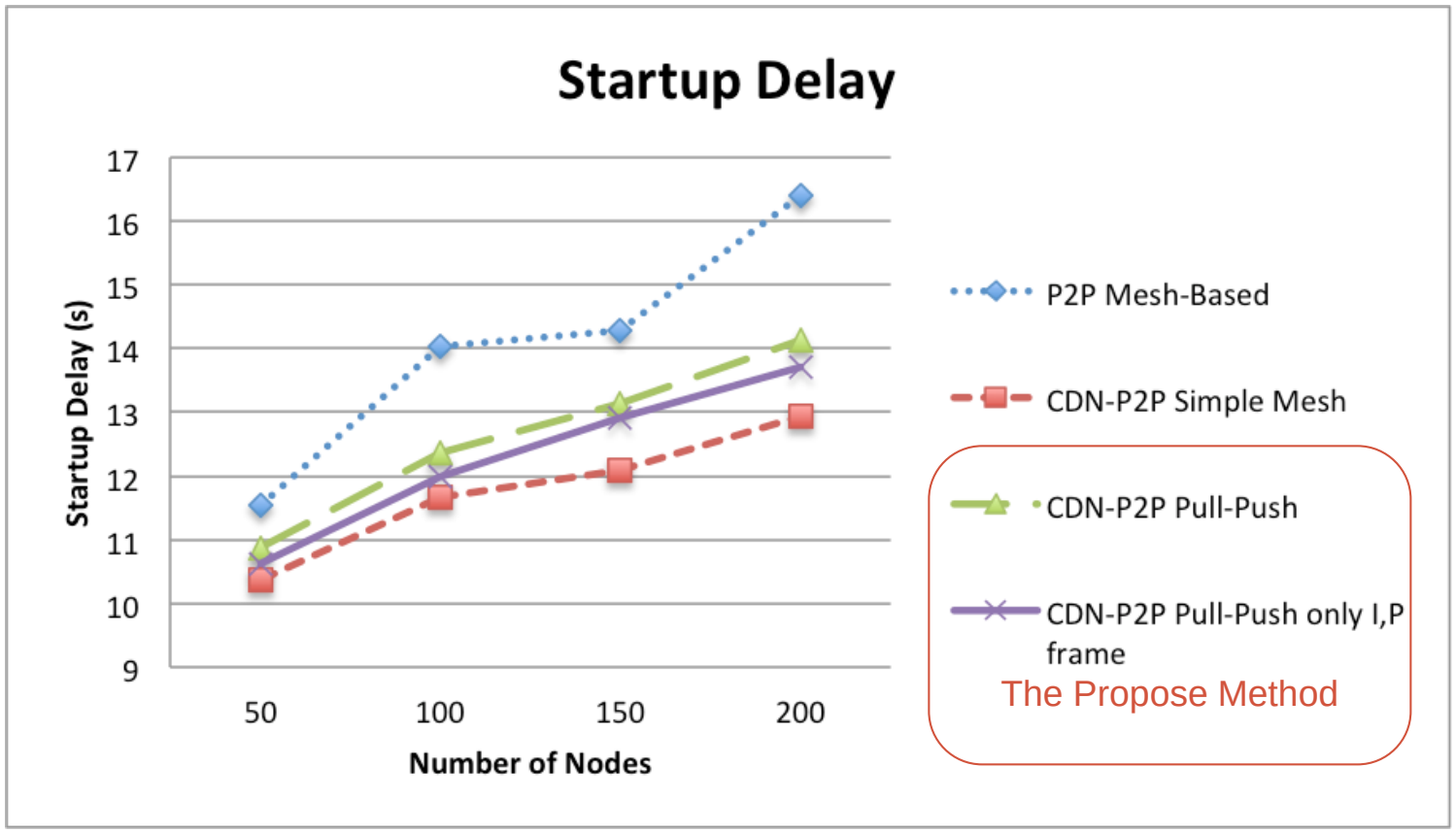
- CDN-P2P with Pull-Push Method with only I frame and P frame

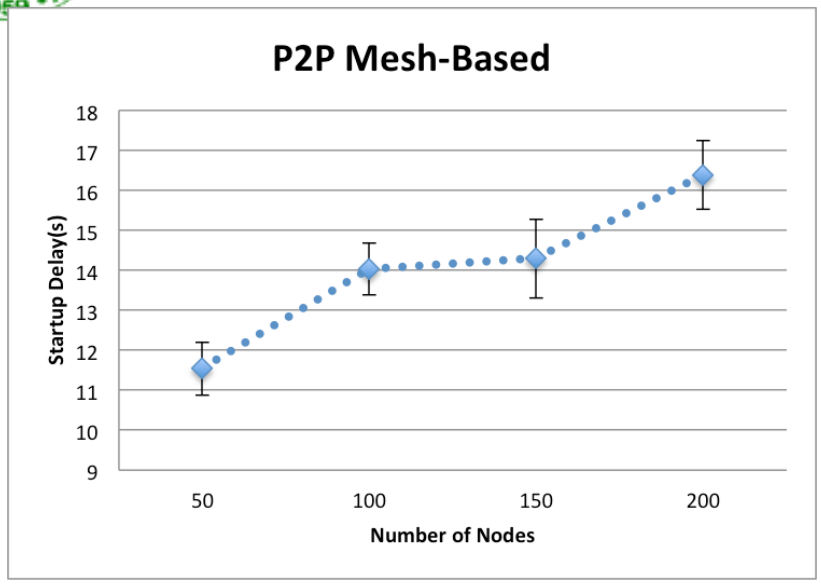
NETWORK SIMULATION

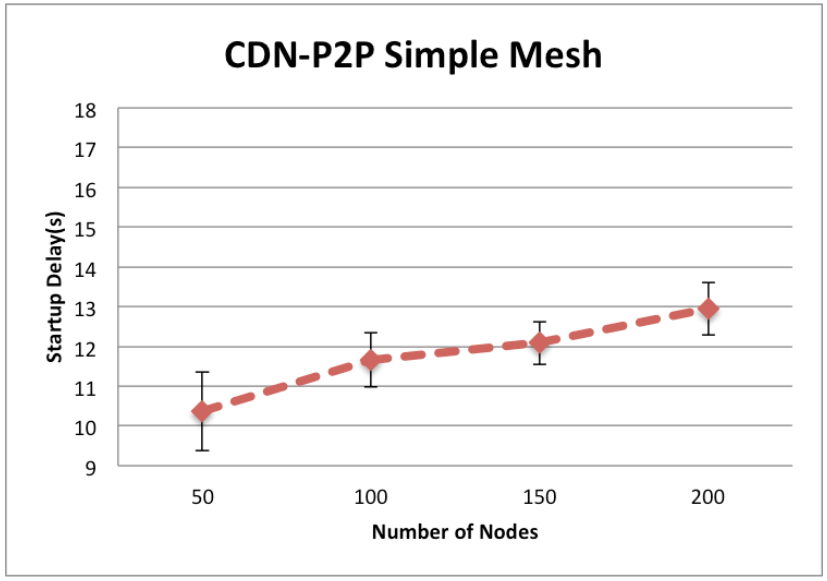


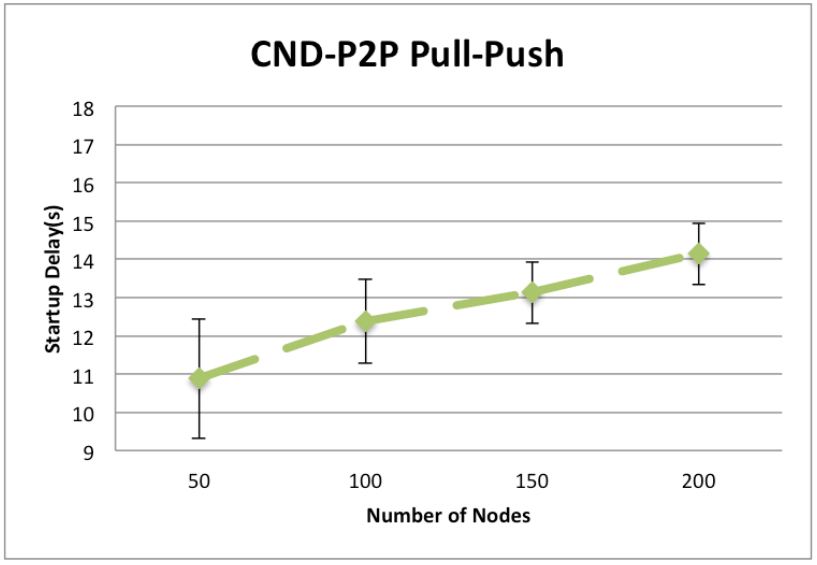
SIMULATION RESULT

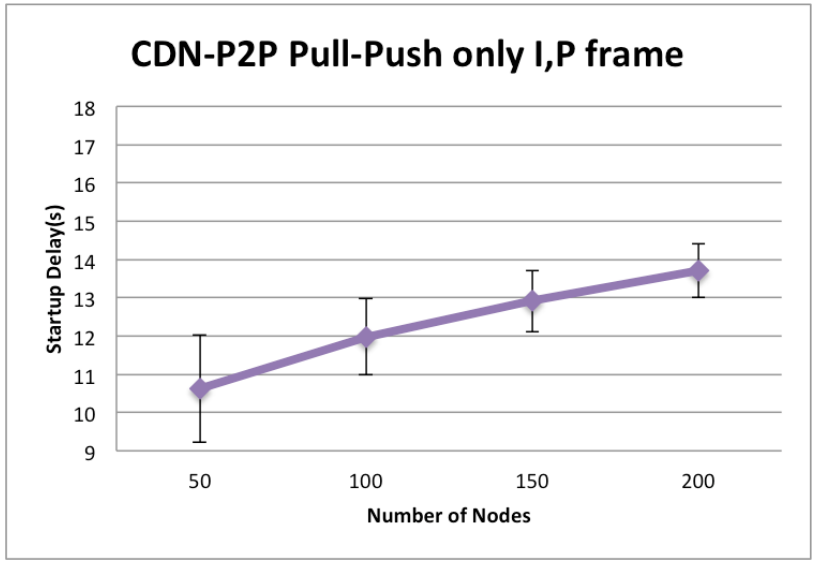
Startup Delay




P2P Mesh-Based
Startup Delay(s)
18
17
16
15
14
13
12
11
10
9
50
100
150
200
Number of Nodes

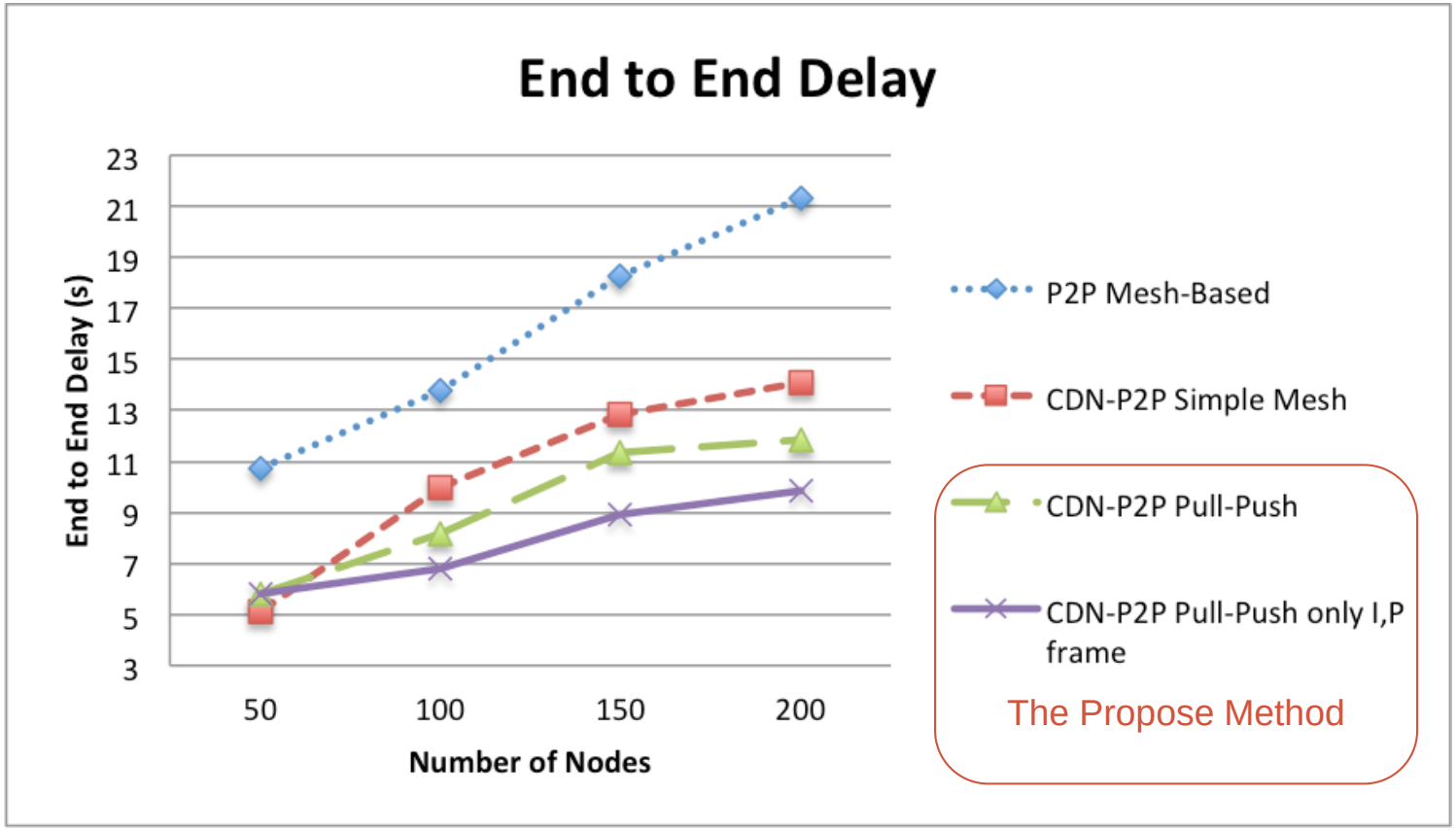

CDN-P2P Simple Mesh
Startup Delay(s)
18
17
16
15
14
13
12
11
10
9
50
100
150
200
Number of Nodes

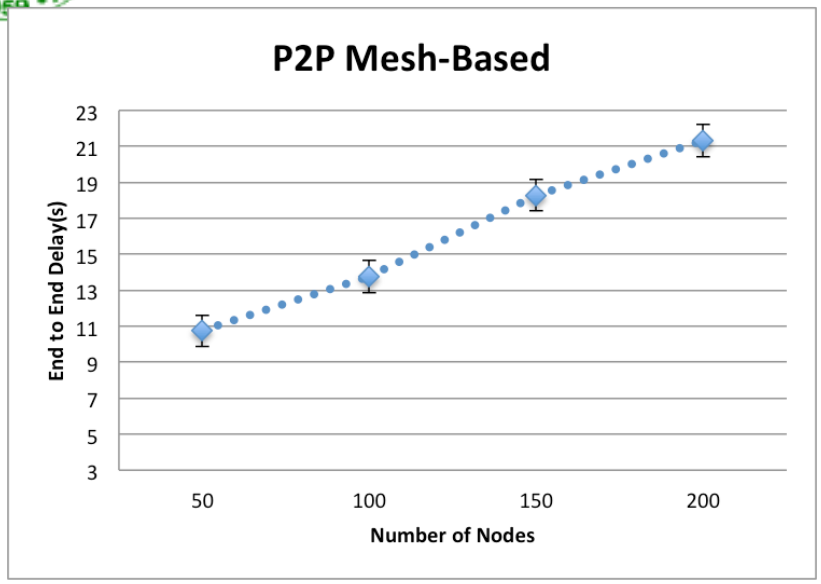

CND-P2P Pull-Push
Startup Delay(s)
18
17
16
15
14
13
12
11
10
9
50
100
150
200
Number of Nodes

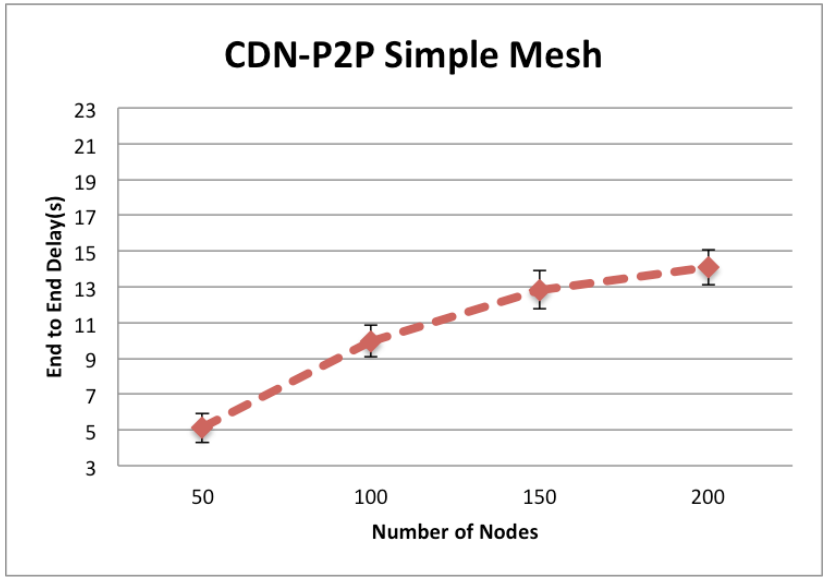

CDN-P2P Pull-Push only I,P frame
Startup Delay(s)
18
17
16
15
14
13
12
11
10
9
50
100
150
200
Number of Nodes

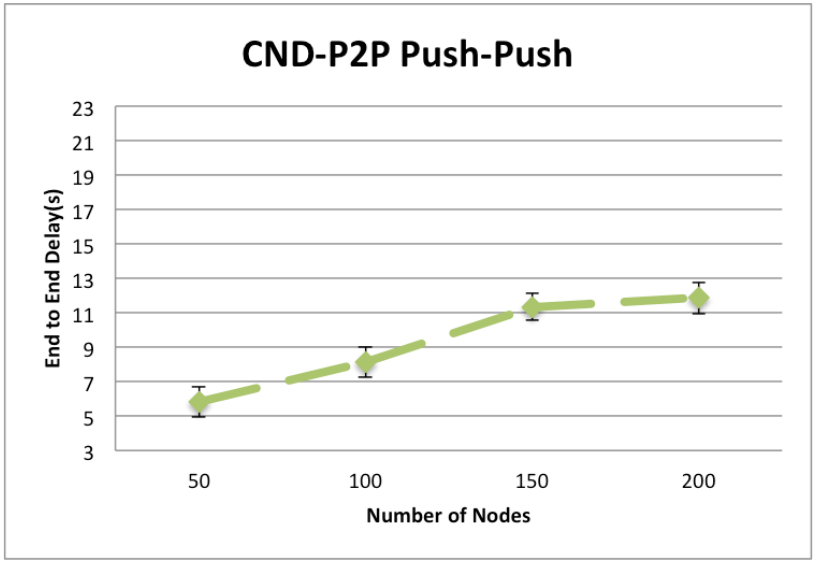
SIMULATION RESULT (cont.)

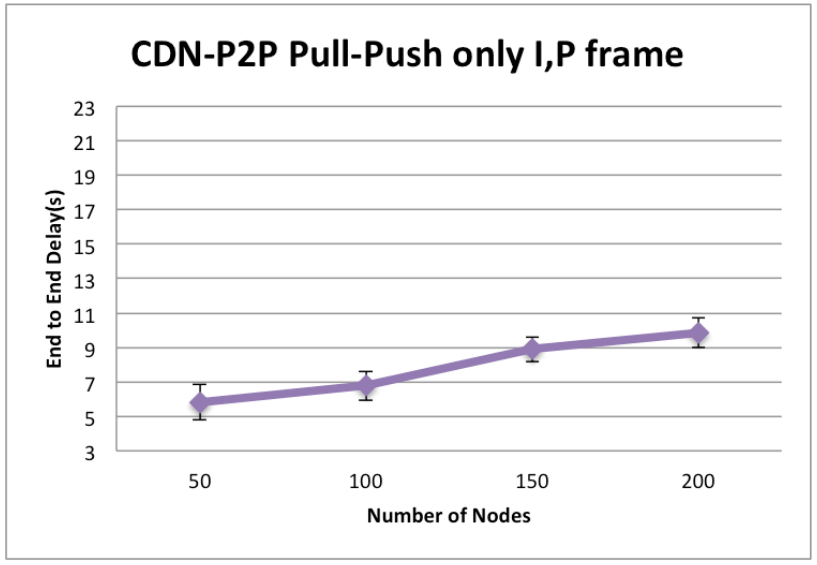
End-to-End Delay




P2P Mesh-Based
End to End Delay(s)
23
21
19
17
15
13
11
9
7
5
3
50
100
150
200
Number of Nodes

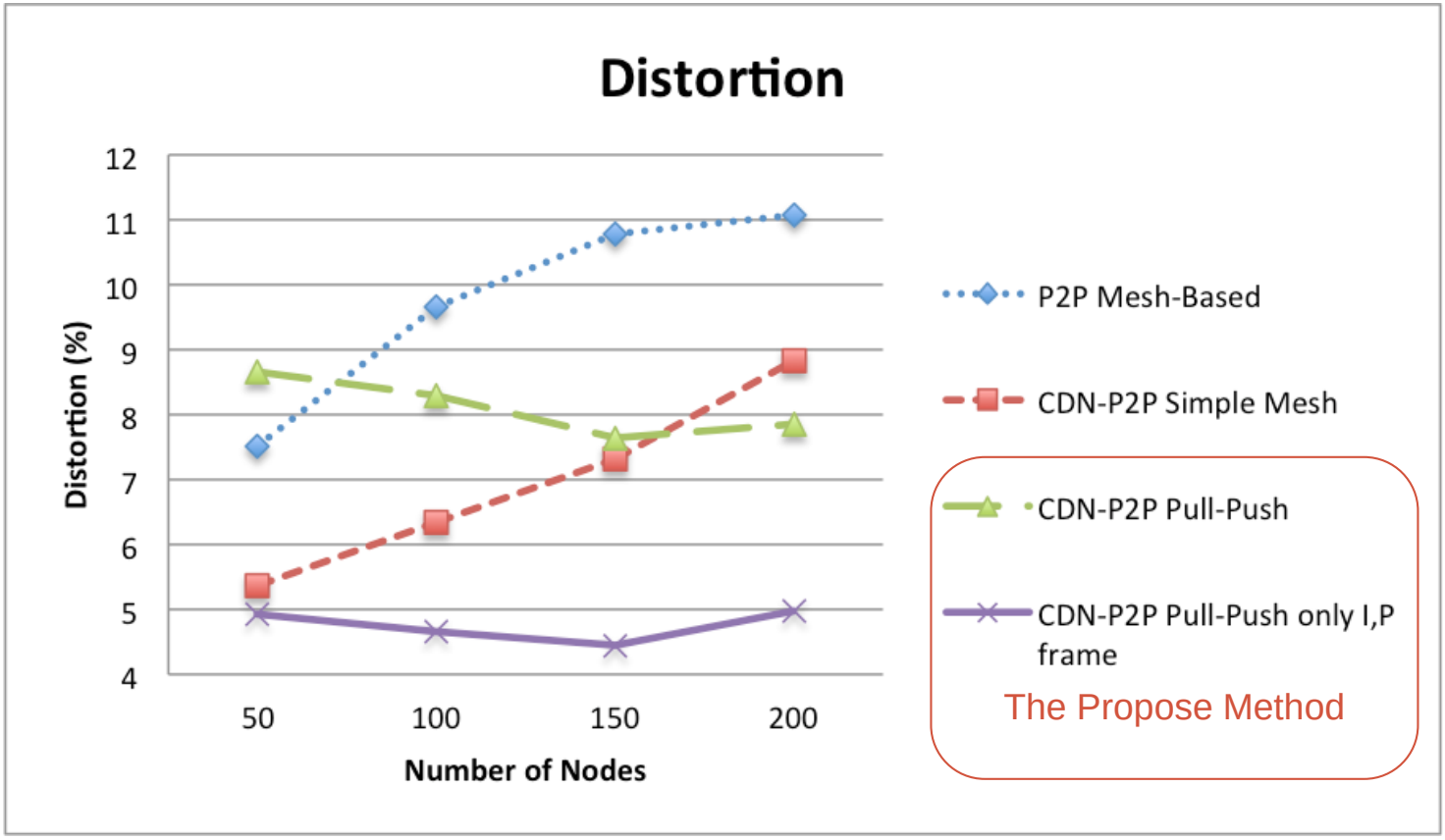

CDN-P2P Simple Mesh
End to End Delay(s)
23
21
19
17
15
13
11
9
7
5
3
50
100
150
200
Number of Nodes

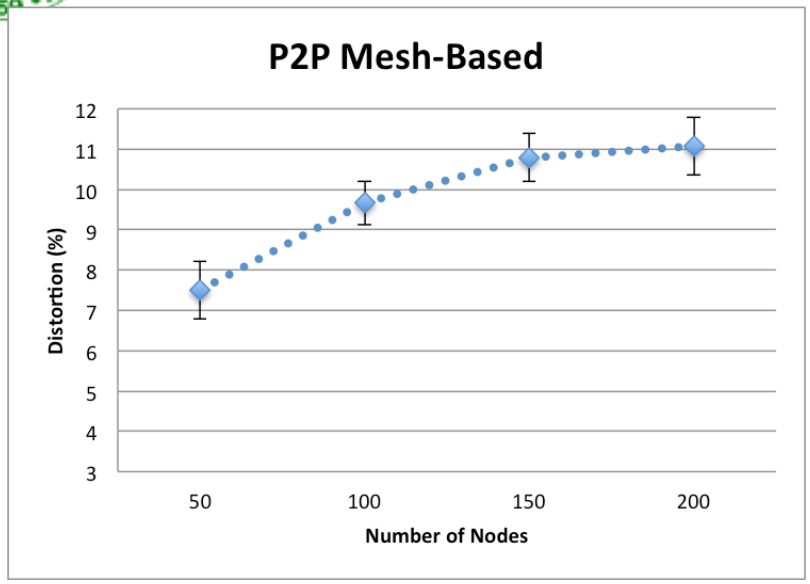

CND-P2P Push-Push
End to End Delay(s)
23
21
19
17
15
13
11
9
7
5
3
50
100
150
200
Number of Nodes

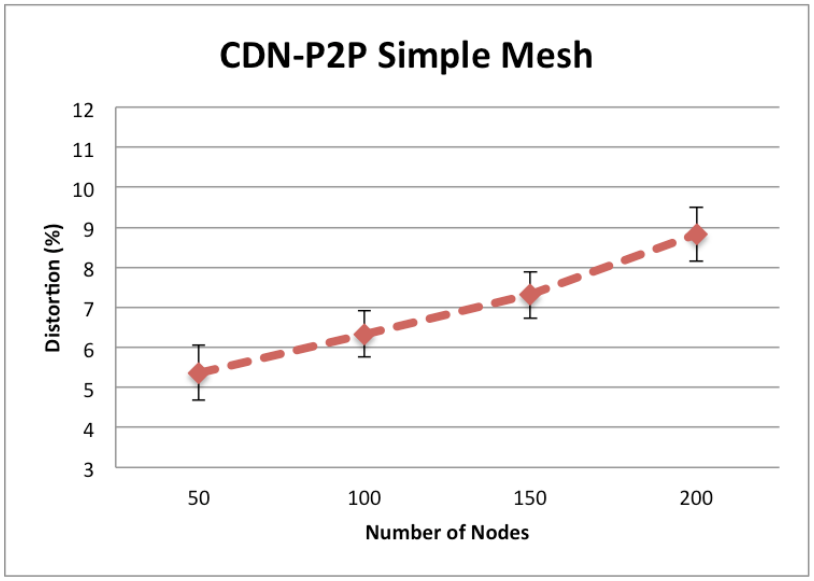

CDN-P2P Pull-Push only I,P frame
End to End Delay(s)
23
21
19
17
15
13
11
9
7
5
3
50
100
150
200
Number of Nodes

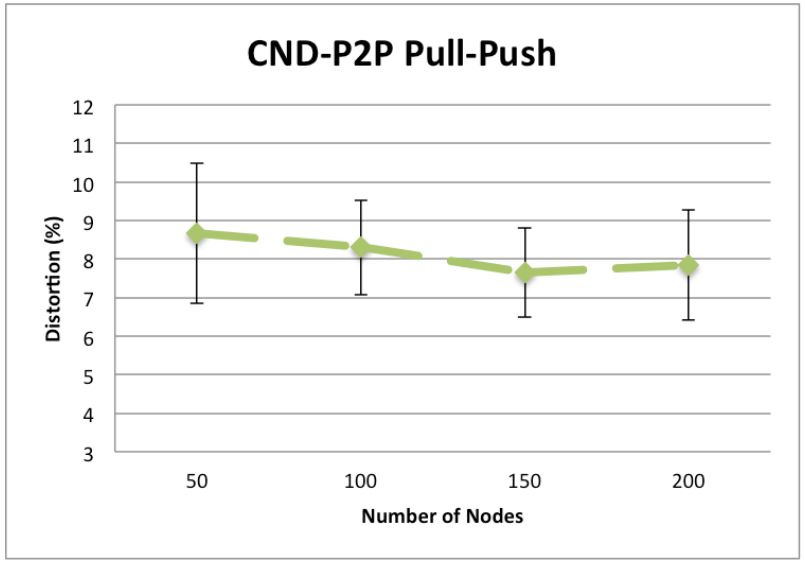
SIMULATION RESULT

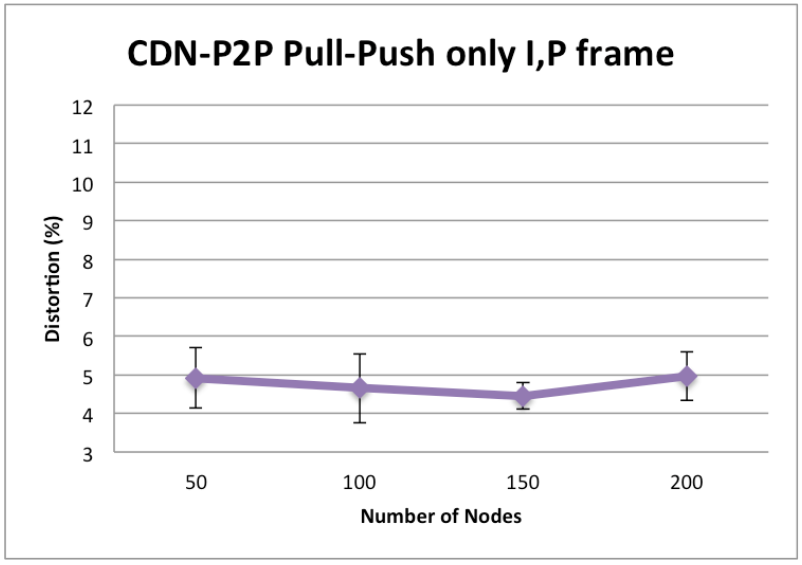
Video Distortion




P2P Mesh-Based
Distortion (%)
12
11
10
9
8
7
6
5
4
3
50
100
150
200
Number of Nodes

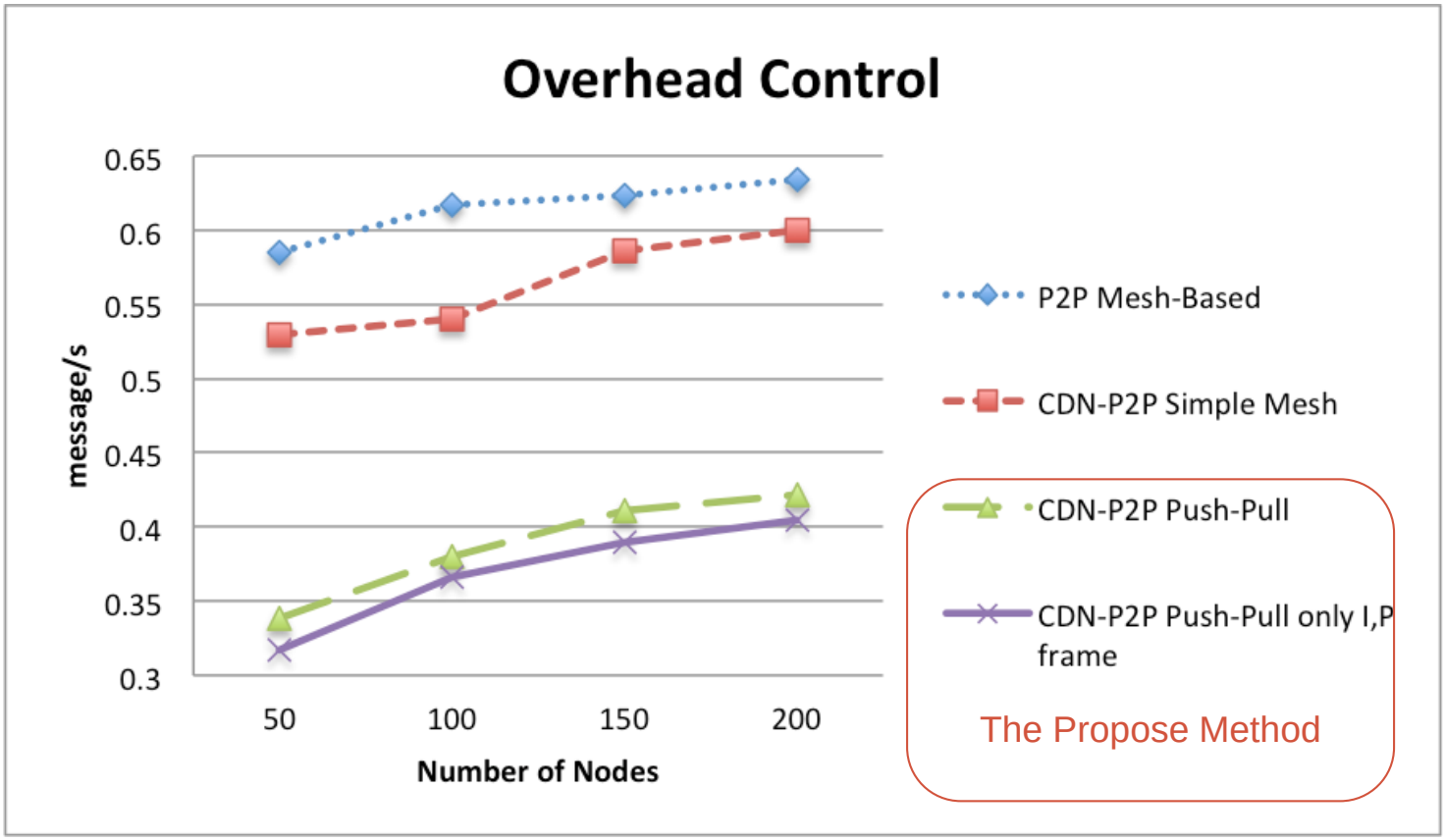

CDN-P2P Simple Mesh
Distortion (%)
12
11
10
9
8
7
6
5
4
3
50
100
150
200
Number of Nodes

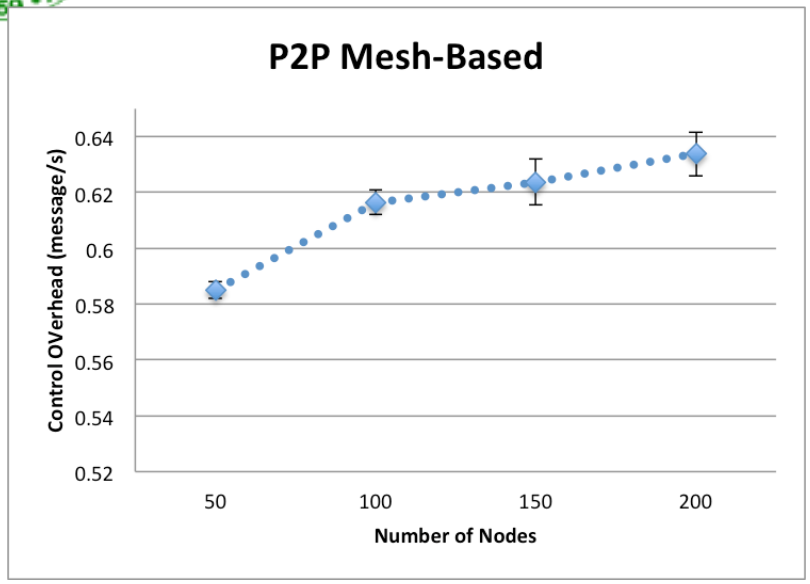

CND-P2P Pull-Push
Distortion (%)
12
11
10
9
8
7
6
5
4
3
50
100
150
200
Number of Nodes

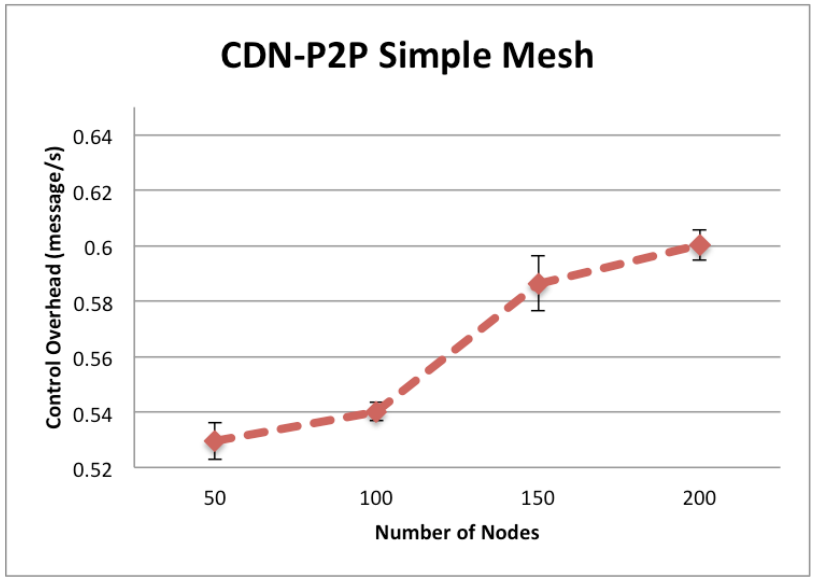

CDN-P2P Pull-Push only I,P frame
Distortion (%)
12
11
10
9
8
7
6
5
4
3
50
100
150
200
Number of Nodes

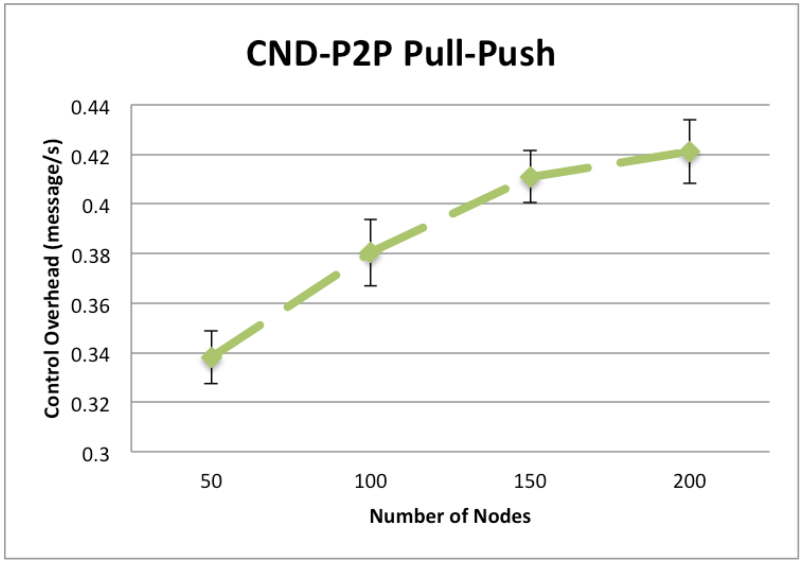
SIMULATION RESULT

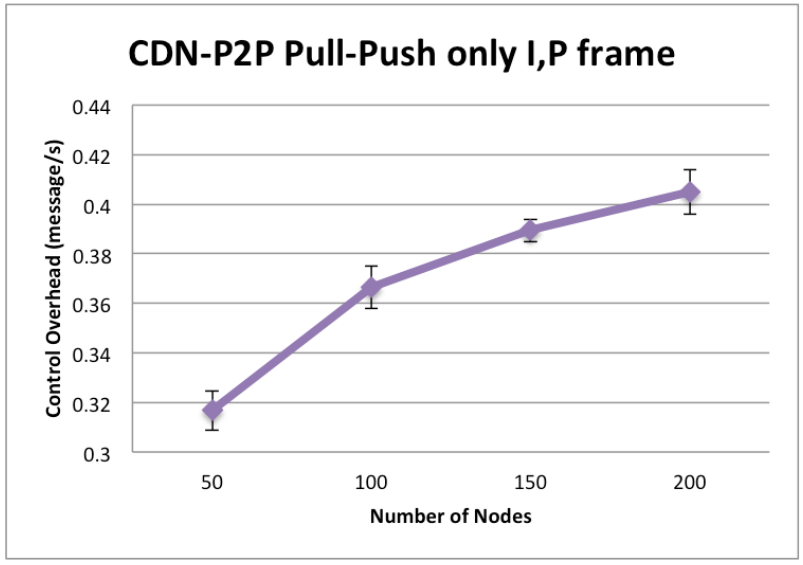
Overhead Control




P2P Mesh-Based
Control OVerhead (message/s)
0.64
0.62
0.6
0.58
0.56
0.54
0.52
50
100
150
200
Number of Nodes


CDN-P2P Simple Mesh
Control Overhead (message/s)
0.64
0.62
0.6
0.58
0.56
0.54
0.52
50
100
150
200
Number of Nodes


CND-P2P Pull-Push
Control Overhead (message/s)
0.44
0.42
0.4
0.38
0.36
0.34
0.32
0.3
50
100
150
200
Number of Nodes


CDN-P2P Pull-Push only I,P frame
Control Overhead (message/s)
0.44
0.42
0.4
0.38
0.36
0.34
0.32
0.3
50
100
150
200
Number of Nodes

CONCLUSION

CONCLUSION

- We simulate the performance of our proposed model using OMNET. We create the overlay and underlay network under this environment.
- The performance evaluation metrics are Start-up Delay, End-to-End Delay, Video Distortion and Control Overhead
- It is obvious that our proposed Hybrid pull-push method can reduce End-to-End Delay and Control Overhead drastically comparing with the CDN-P2P simple mesh.
- However our proposed model causes high video distortion rate when all I, B and P frames are put into account.
- In order to reduce the video distortion, only I and P frames are distributed .
- Therefore, the proposed method has significant improvement in terms of end-to-end delay, video distortion and overhead control.
- The proposed method has higher startup delay comparing with the CDN-P2P simple mesh due to the time needed to create tree structure.

THANK YOU

teerapat@ait.ac.th