

Overlay Multimedia Networking

Presented by
Dai-boong Lee



Multimedia Networking Applications

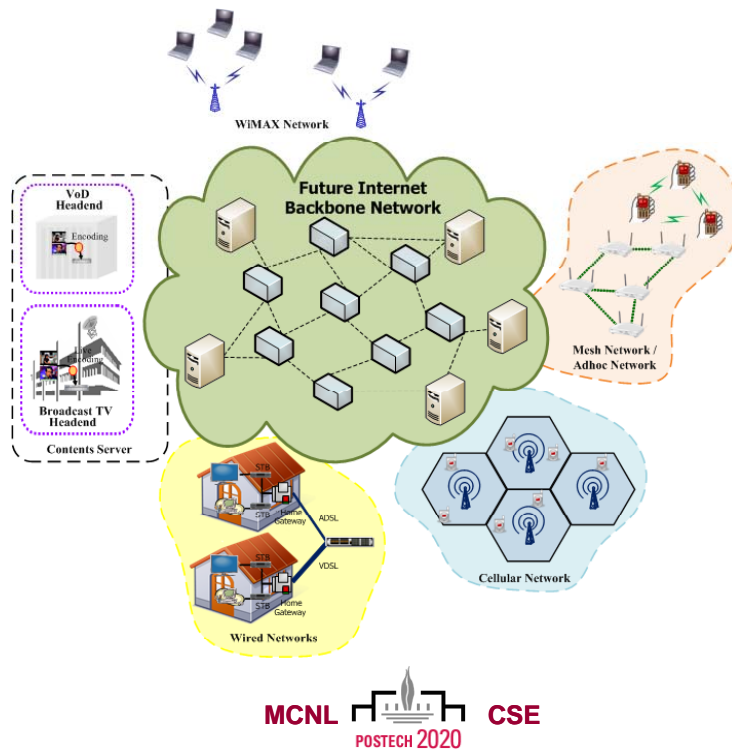
- Internet
 - Simple protocol
 - Complex operations is performed by end systems
 - Performs simple operations for packet forwarding in the network
 - Best effort service
 - No guarantees on delay & packet loss

- Various applications
 - IPTV service
 - VoD service
 - Multimedia P2P service
 - Video conferencing
 - Etc..

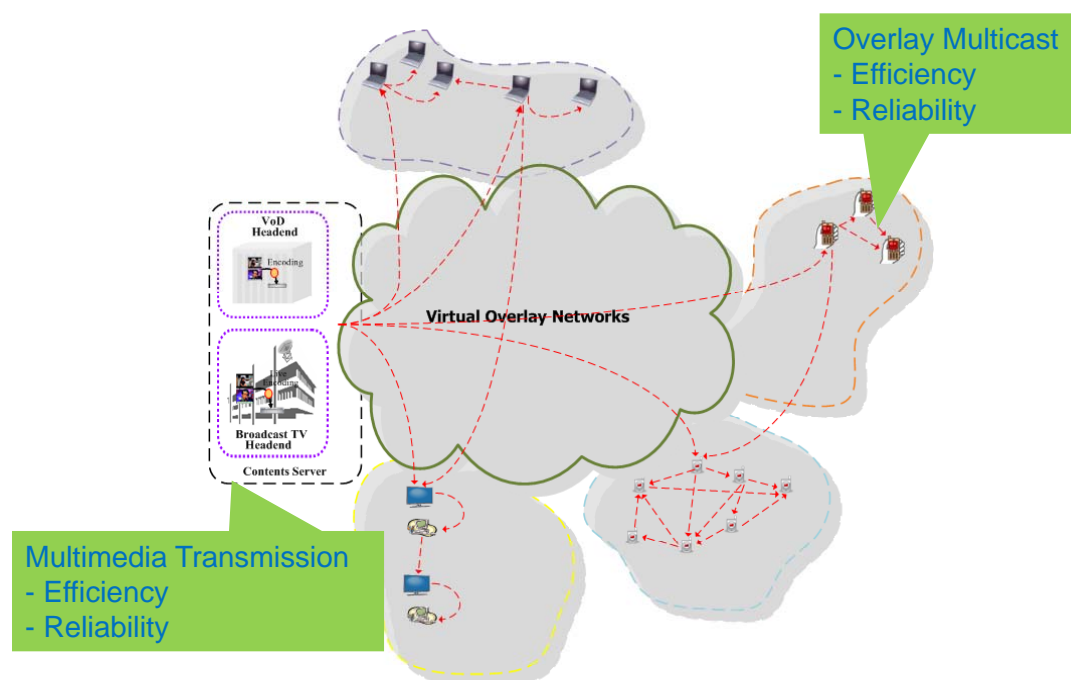


Challenge and Research Areas

❑ Multimedia Transmission over Wired/Wireless Networks



❑ Overlay Networks



Overlay Multicast Tree



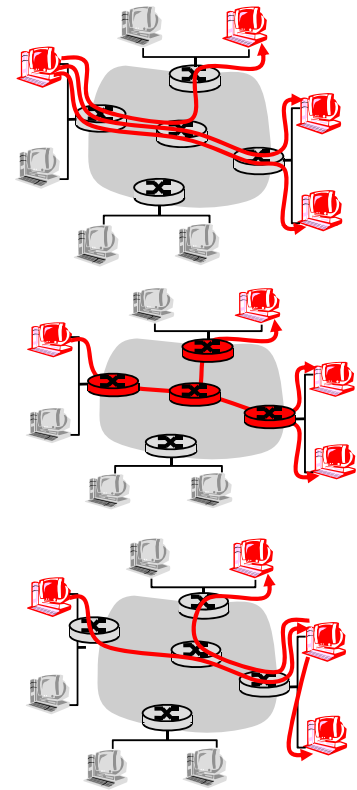
Introduction

- ❑ Many research efforts have been devoted to IP multicast so far
 - The limitations of IP multicast
 - Class-D address ignored over unicast network
 - High computational complexity of router
 - Actually very difficult to change all unicast routers
- ❑ Overlay multicast has been considered as an alternative of IP multicast in the recent years
 - Data packets are replicated at end-hosts
 - The proposed approaches can be classified into three classes:
 - **Mesh-first** : Create a more densely connected structure first
ex) Narada, NICE, etc
 - **Tree-first** : The data delivery tree is created first
ex) Yoid, HMTP, Overcast, etc.
 - **Implicit** : The mesh and the tree are simultaneously defined by the protocol
ex) CAN-multicast, Scribe, etc

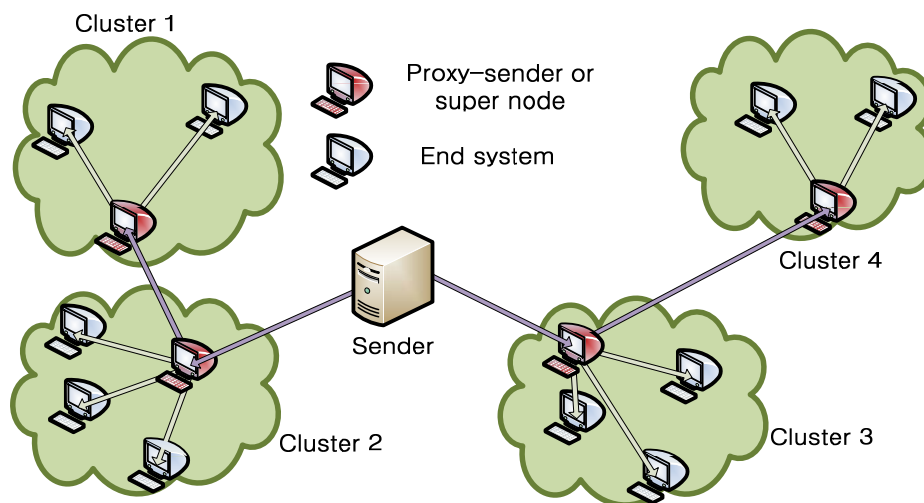


□ Problems of overlay multicast

- More redundant traffics than IP multicast
 - Much lower than unicast
- Time delay can be increased
 - Processing and relay delay on application layer
 - It may be a critical problem for real-time media.
- Fragile tree
 - Robust tree to node failure
- etc



Overlay Network

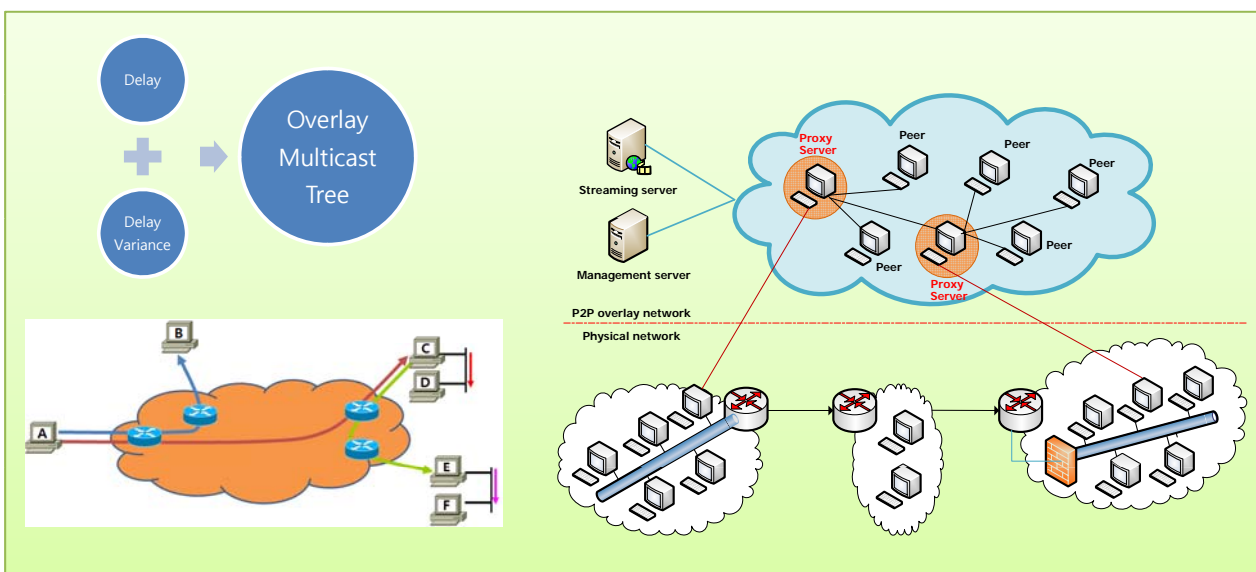


Algorithm 1

: Overlay Multicast Tree for the Synchronized Real-time Media Multicast Service

Overview

OMT for the Synchronized Real-time Media Multicast Service



Problem Description

- To reduce the required computational complexity, we assume

$$DELAY_{p_i}^s \gg DELAY_j^{p_i}$$

$DELAY_{p_i}^s$: the delay between the sender and the proxy-sender p_i

$DELAY_j^{p_i}$: delay between the j_{th} end-system and the proxy-sender p_i in the i_{th} cluster

- Delay and Delay variance of the end-system

$$DELAY_j^s \approx DELAY_{p_i}^s$$

$$VAR_j^s \approx VAR_{p_i}^s = \left| DELAY_{p_i}^s - \frac{1}{N} \sum_{j=1}^{n_p} m_j \cdot DELAY_{p_j}^s \right|^2$$

N : the number of end-systems participating in the multicast ($N = \sum_{i=1}^{n_p} m_i$)

n_p : the number of clusters

m_i : the number of end-systems in the i_{th} cluster



-
- Determine an overlay multicast tree to minimize

$$\frac{1}{N} \sum_{i=1}^{n_p} m_i \cdot \{ (1-\alpha) \cdot DELAY_{p_i}^s + \alpha \cdot VAR_{p_i}^s \}$$

subject to $s_s \leq ST_s$ and $s_{p_i} \leq ST_{p_i}$

s_s : the numbers of streams handled by the sender

s_{p_i} : the numbers of streams handled by proxy-sender

ST_s : the maximum numbers of streams supported by the sender

ST_{p_i} : the maximum numbers of streams supported by the proxy-sender



Tree Constriction

- ❑ Minimize not only average delay but also delay variance
 - NP-complete Problem
 - Orthogonal genetic algorithm is employed to obtain the near optimal solution with a low computational complexity
 - Genetic algorithm + Experimental design method

- ❑ Genetic algorithm is an exploratory procedure that is often able to locate near optimal solutions to complex problems
 - A high computational complexity

- ❑ Experimental design method
 - Examine all combinations of factors and levels
 - Check only some of representative combinations
 - Reduce the searching range during the cross-over operation

Initialization:

$P = \{\text{Randomly create an initial population of } K \text{ binary combinations}\}$ //Create initial K -population.

$Generation_number = 0;$

Population Evolution:

while ($Generation_number < Maximum_Generation_Number$) do

$P' = \emptyset$

while ($K/2$ times) do

 Randomly select three parent-combinations from P .

 Perform orthogonal cross-over, mutation, and check-and-repair operations on them.

$P' = P' \cup \{\text{After evaluation is done, generate 2 offspring-combinations for the next generation}\}$

End of while loop

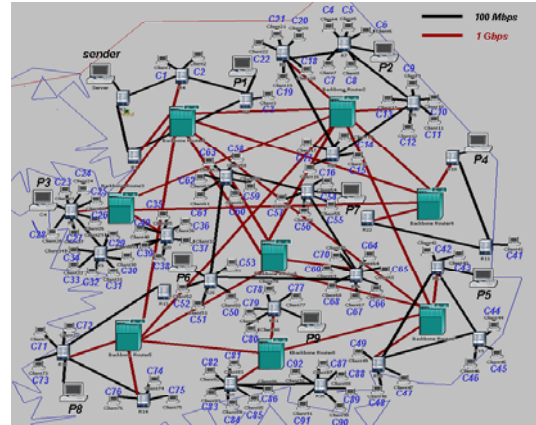
$Generation_number = Generation_number + 1$

$P = P'$

End of while loop

Simulation Results

- ❑ Video stream
 - Frame rate : 30 fps (frames per second)
 - 1 GOP consists of 12 frames (IBBPBBPBBPBB)
 - Bandwidth : 4.12Mbps
- ❑ Network characteristic
 - The link bandwidths of core network : 1Gbps
 - The link bandwidths of access network : 100Mbps
- ❑ The number of clusters is set to 10, which is based on IP addresses and the number of supportable streams of end-systems



- ❑ Performance comparison with existing algorithms

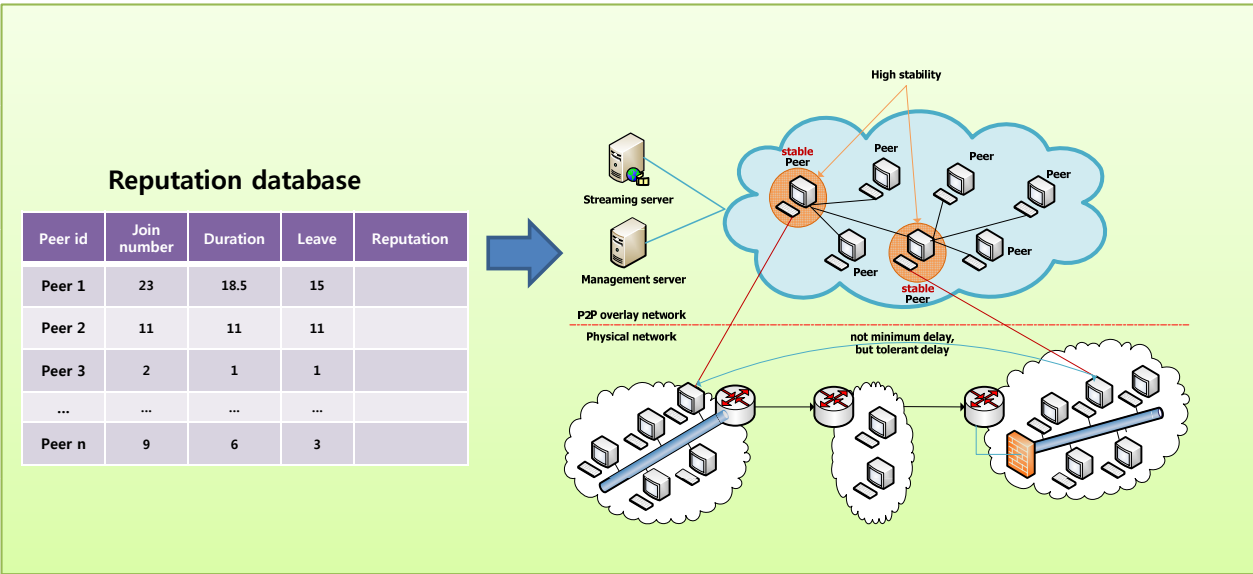
<i>Algorithm</i>		<i>Measure</i>	<i>Avg. delay (msec)</i>	<i>VAR_{tree} (sec)</i>
MDA			613	13.659
DCMALTP			631	15.397
CPT			664	18.377
BTP			965	41.032
Proposed Alg.	$\alpha = 0.0$		561	10.786
	$\alpha = 0.9$		677	8.736
	$\alpha = 1.0$		1001	6.904

Algorithm 2

: Reputation-Based Stable Overlay Multicast for Live Streaming

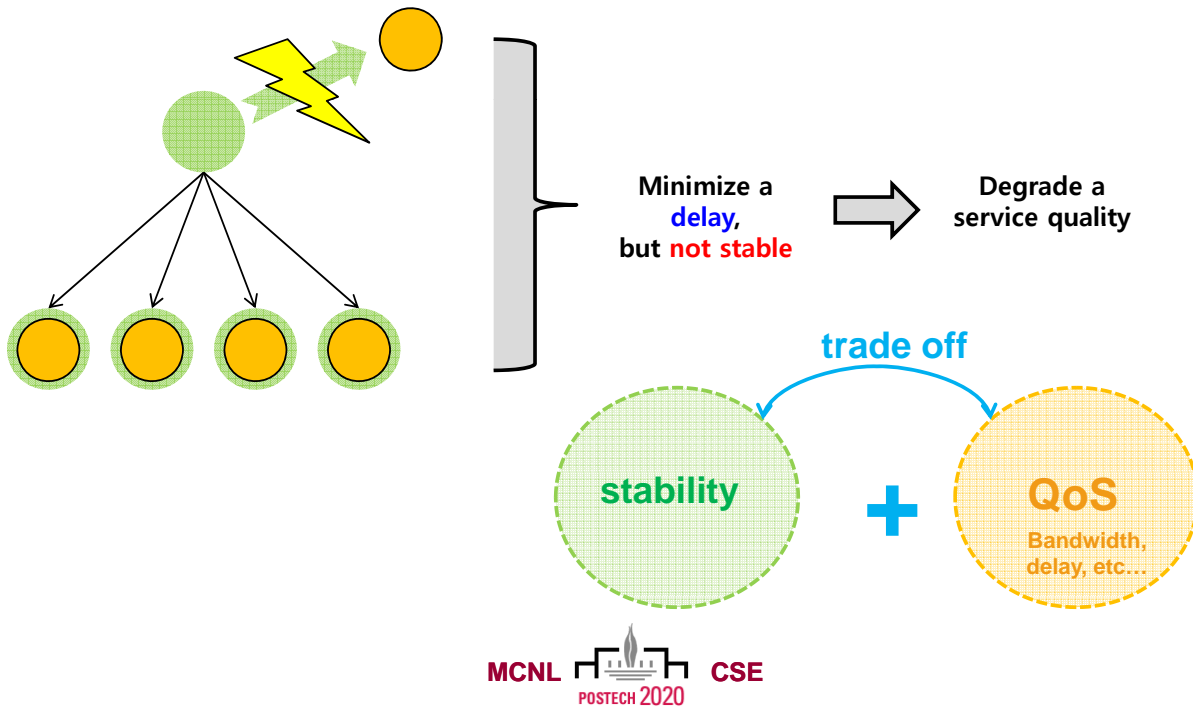
Overview

Reputation based stable overlay multicast for live streaming



Motivation

- An overlay network may be **dynamic** with frequent joins and leaves
- Organize a **stable** overlay network for live streaming



Motivation

- Stable?

Data Set	Avg. Ratio of Stable Nodes per Snapshot	Ratio of Stable Nodes per Trace	Ratio of Representative Trees to All Per-Block-Trees	Avg. Representativity
CCTV3 Trace1	77.7%	8.4%	0.17%	82.3%
CCTV3 Trace2	73.8%	5.5%	0.13%	80.9%
DragonBall Trace1	85.0%	8.5%	0.22%	79.1%
DragonBall Trace2	84.6%	15.4%	0.28%	80.5%

TABLE I
STATISTICS OF STABLE NODES AND REPRESENTATIVE PER-BLOCK-TREES.

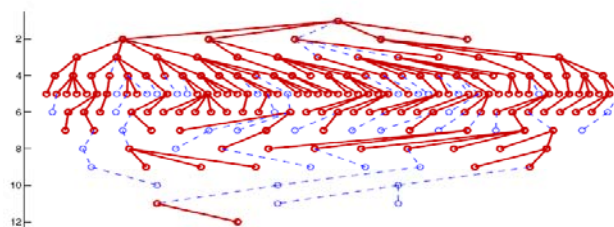
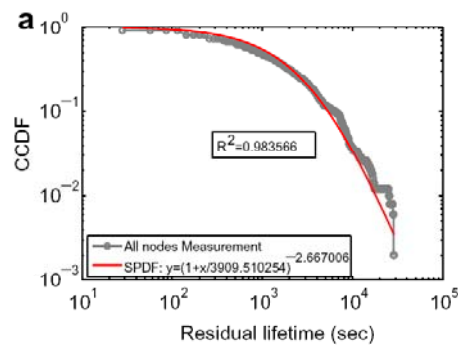


Fig. 4. A snapshot of a representative per-block-tree from CCTV3 Trace-1. The thick circles and lines correspond to stable nodes and their connections.

Stable Peers: Existence, Importance, and Application in Peer-to-Peer Live Video Streaming
 Feng Wang, Jiangchuan Liu, Yongqiang Xiong
 Simon Fraser University
 IEEE INFOCOM'08, Phoenix, AZ, USA, April 15-17, 2008

Motivation

- CCDF of residual lifetime for PPLive node



Improving stability for p2p multicast overlays by active measurements

Y Tian, D Wu, G Sun, KW Ng

The Chinese University of Hong Kong

Journal of Systems Architecture, 2008 - Elsevier

mTreebone: A Hybrid Tree/Mesh Overlay for Application-Layer Live Video Multicast

Feng Wang, Yongqiang Xiong, Jiangchuan Liu

Simon Fraser University

IEEE ICDCS, 2007



Related Work

- Traditional method
 - ESM, ZigZag, NICE, Narada, Chunkyspread, Coolstreaming, Chainsaw, Bullet and so on.
- Method with considering peer's stability
 - mTreebone (tree)
 - Stable peers... (hybrid : tree + mesh)
 - Improving stability... (tree)
- Problems
 - They need much time to evaluate peer's stability.
 - They does not consider QoS.
 - They dose not provide stability measure with considering overlay topology.

Reputation

- Estimation of the reputation
 - 1. By using a peer's age in the session (current reputation)
 - Peer's lifetime model : Pareto distribution
 - Can predict peer's residual lifetime through peer's age
 - 2. By using a peer's past reputation
 - Peer's history score

Current reputation

- Estimation of peer's residual lifetime from its hazard rate
 - the length of time it is expected to be online in future could be approximated through hazard rate

$$\frac{1}{h(x)} = \frac{\beta + x}{\alpha}$$

- Estimation of peer's current reputation from its estimated residual lifetime

- $L_i(t)$: current reputation of peer i at time t
- S : session length
- t : current time
- α, β : parameter of the shifted Pareto distribution
- a_i : current age of peer i

$$L_i(t) = \begin{cases} \frac{1}{S-t} \cdot \frac{\beta + a_i}{\alpha} & \text{if } \frac{\beta + a_i}{\alpha} \leq S - t \\ 1 & \text{otherwise} \end{cases}$$

Estimated residual lifetime
Residual session length

Past reputation

- Measure of past reputation
 - Duration (d hours)
 - Peer's duration in the session
 - Leave behavior (l times)
 - Whether to leave gracefully or ungracefully when peer tunes out
 - Join number (n times)
 - The number of session that peer joined

Past reputation

- Estimation of past reputation
 - Notation
 - H_i : past reputation of peer i
 - d_i : duration of peer i
 - l_i : leave behavior of peer i
 - n_i : join number of peer i
 - N_{trust} : inflection point of trust factor

$$H_i = \frac{1}{1 + \left(\frac{n_i}{N_{trust}}\right)^{-K}} \cdot \left[\alpha \cdot \frac{d_i}{n_i} + \beta \cdot \frac{l_i}{n_i} \right]$$

Average duration
Graceful leave ratio
Trust factor

$$K \geq 1, \quad \alpha + \beta = 1, \quad \alpha, \beta \geq 0$$

Reputation

□ Estimation of the reputation

- Notation

- $R_i(t)$: reputation of peer i at time t
- H_i : past reputation of peer i
- $L_i(t)$: current reputation of peer i at time t

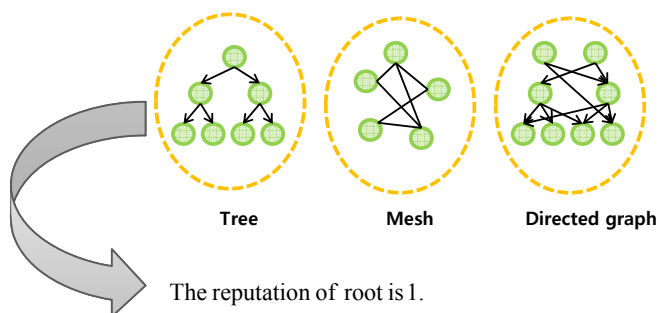
$$R_i(t) = \alpha \cdot H_i + \beta \cdot L_i(t)$$

$$\alpha + \beta = 1, \quad \alpha, \beta \geq 0$$

Stability of overlay network

□ Stability of overlay multicast network

- The stability can be measured via the peer's reputation
- Network topology should be considered for measuring the stability
 - Peer's stability depends on neighbor's stability



The reputation of root is 1.

n : the number of peer in the overlay network

$A(i)$: ancestor set of peer i

$R_i(t)$: reputation of peer i at time t

$$\text{Stability of peer } i : \prod_{j \in A(i)} R_j(t)$$

Goal

- Goal
 - Construct multicast tree with considering both stability and delay
- Problem formulation

Determine T

$$\text{Maximize } \left(\alpha \cdot \frac{\sum_{v_i \in V_T} S_i(t)}{|V_T|} + (1 - \alpha) \cdot \frac{1}{\max_{v_i \in V_T} (D_i)} \right)$$

Subject to

$$G_j \leq C_j \quad \text{for } \forall j \in [1, n]$$

$G = (V, E)$: a graph (full mesh)

$V = (v_1, v_2, \dots, v_n)$: a set of nodes

$E = (e_1, e_2, \dots, e_m)$: a set of edges ($m = n \cdot (n-1)/2$)

$T = (V_T, E_T)$: a multicast tree ($V_T = V, E_T \subset E$)

$S_i(t)$: stability of peer i at time t

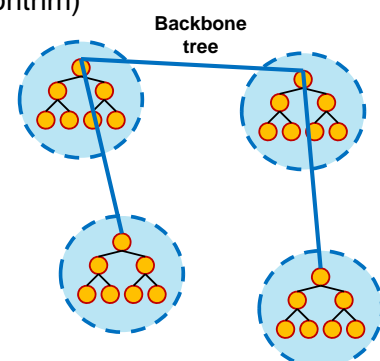
D_i : delay of peer i from the source

G_i : out-degree of peer i

C_i : available out-degree of peer i

Algorithm

- 1. construct the backbone tree
 - A. select the super-node candidate based on reputation in each cluster
 - B. randomly select the super-node in super-node candidate
 - C. construct the backbone tree with considering both delay and stability
 - Heuristic algorithm (bees algorithm, genetic algorithm)
- 2. construct the sub-tree
 - A. construct the sub-tree with considering both delay and stability in cluster
 - Heuristic algorithm (bees algorithm, genetic algorithm)
 - Root is super-node
 - B. repeat the 1.B – 2.A



Experiments

□ Environment

- Cluster : 10
- Node in cluster : 100
- Reputation
 - Cluster : gaussian distribution (mean : 0.5)
- Delay
 - Intra-cluster : gaussian distribution (mean : 100ms)
 - Inter-cluster : gaussian distribution (mean : 400ms)
- Degree
 - 3~6 (gaussian distribution)

Experiments

□ Bees algorithm

Evaluation function	Avg. stability	Min. stability	Avg. delay	Max. delay
$0.999 * \text{avg_stability} + 0.001 / \text{max_delay}$	0.722148	0.453428	0.957239	1.621741
$0.99 * \text{avg_stability} + 0.01 / \text{max_delay}$	0.708421	0.336344	0.896106	1.544022
$0.9 * \text{avg_stability} + 0.1 / \text{max_delay}$	0.653131	0.299647	0.821717	1.235447

□ Genetic algorithm

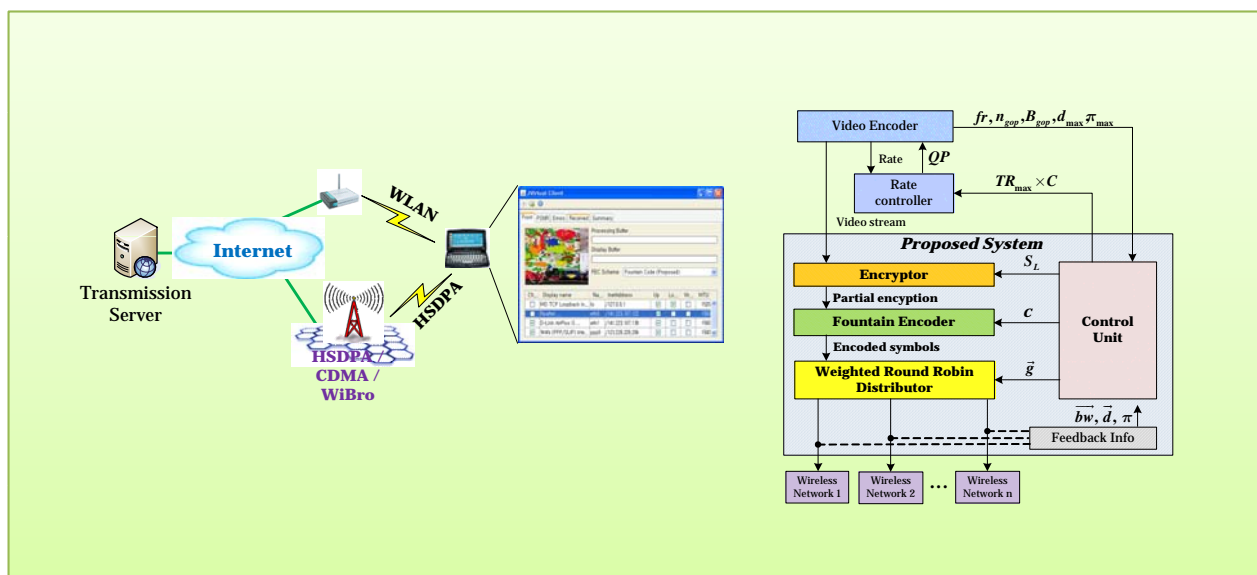
Evaluation function	Avg. stability	Min. stability	Avg. delay	Max. delay
$0.999 * \text{avg_stability} + 0.001 / \text{max_delay}$	0.710988	0.503420	1.103267	1.803492
$0.99 * \text{avg_stability} + 0.01 / \text{max_delay}$	0.698009	0.403829	1.023966	1.771332
$0.9 * \text{avg_stability} + 0.1 / \text{max_delay}$	0.630240	0.302342	0.892340	1.563211

Multimedia Transmission

: Fountain Code-Based Virtual Path for Video Transmission over Multiple Wireless Networks

Overview

Fountain Code-Based Virtual Path for Video Transmission over Multiple Wireless Networks



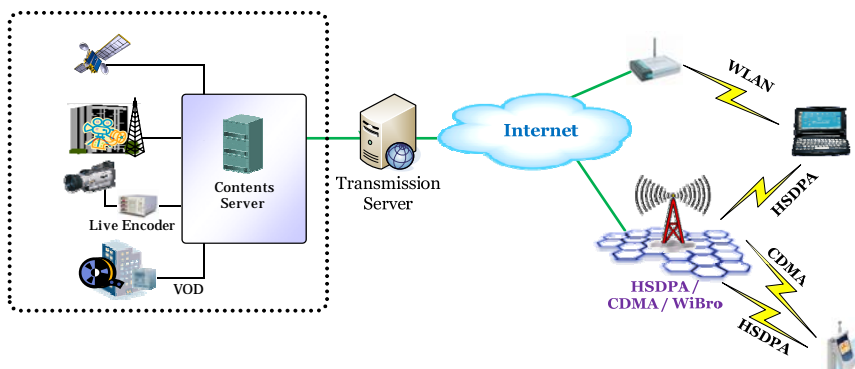
Path Diversity

- ❑ Mobile devices with multiple network interfaces are emerged
- ❑ This kind of devices may be general in near future
- ❑ Software defined radio may accelerate this trend



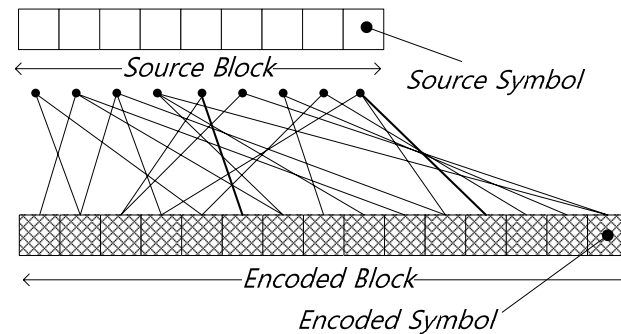
Advantages of Using Path Diversity

- ❑ Parallel transmission
 - Increase bandwidth
- ❑ Path independency
 - Compensate burst errors
- ❑ Eliminate vertical handoff delay



Fountain Code

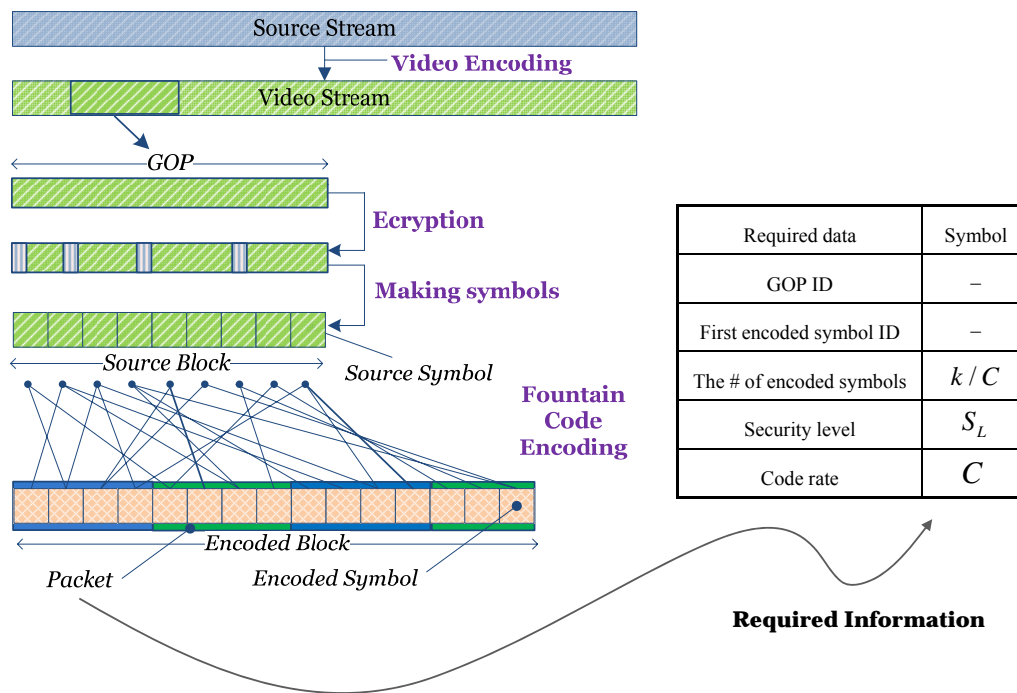
- ❑ LT code, Raptor code, Online code
- ❑ Rateless code
 - Infinite encoding symbols (infinite redundancy)
- ❑ Erasure code (Binary-erasure channel)
- ❑ Using synchronized pseudo random number generator



Advantages of Fountain Code

- ❑ Unreliable paths can be used for better video quality
 - Consuming little more bandwidth to eliminate packet loss effect
 - Greedy approach
- ❑ Very high packet loss rate can be compensated
 - Mobile device can use all possible network resources
 - Adaptive coding for time varying characteristics of wireless channels
- ❑ If decoding probability is guaranteed, then
 - All packets has the same priority
 - Simple weighted round-robin scheduling is enough to distribute packets among possible paths

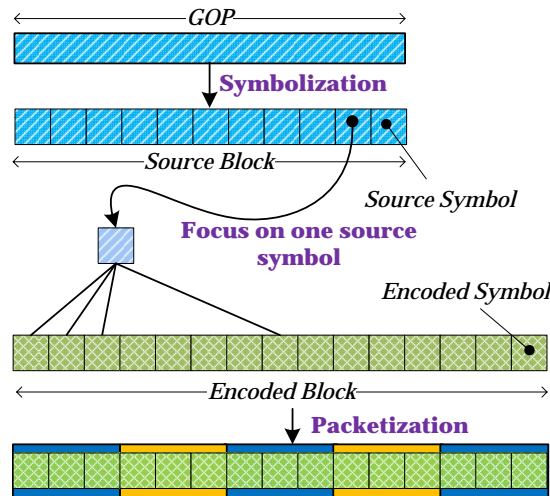
Deployment of Fountain Code



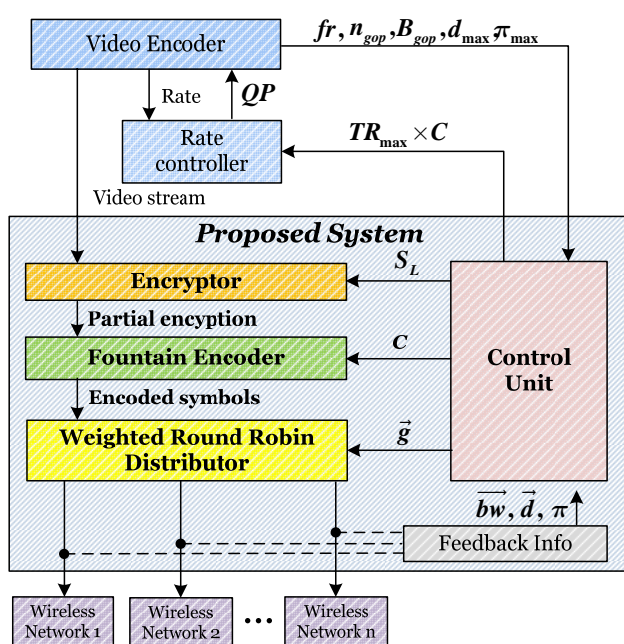
Fountain Code Improvement

- ❑ Require more than one thousand symbols
 - One packet has more than one symbols
 - The packet loss affects multiple continuous encoded symbols
 - It is possible that lost symbols have the links to the same source symbol
- ❑ Packetization-aware deployment is required
 - Encoded symbols in the same packet don't have the same links to any source symbols
 - The packet loss doesn't severely (dominantly) hinder the decoding of any source symbols

Fountain Code Improvement

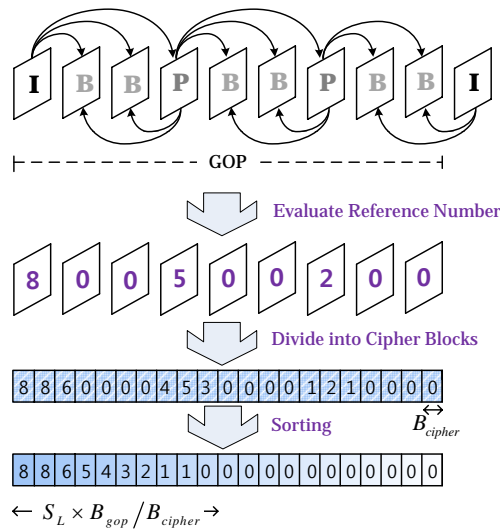


Deployment of Virtual Path Layer

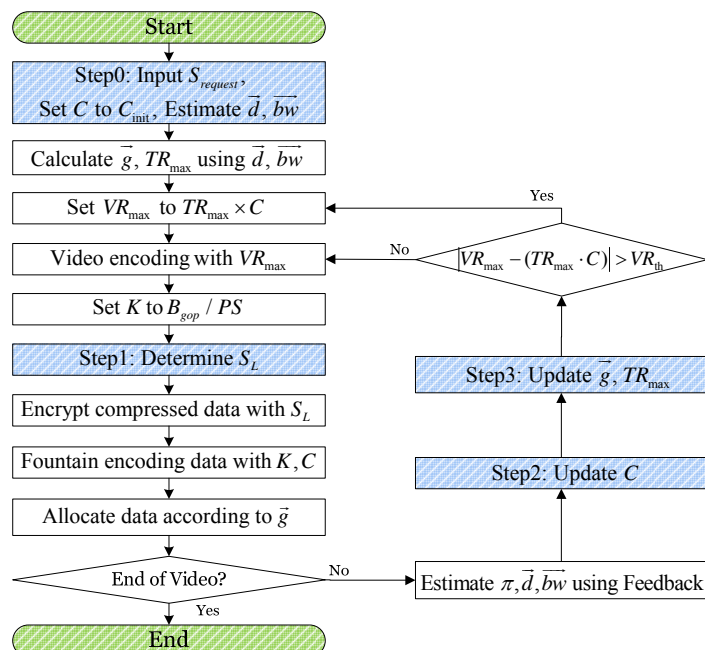


Symbols	Definitions
fr	frame rate (frames per second)
n_{gop}	number of frames in a GOP
B_{gop}	amount of bits for a GOP
d_{max}	maximum tolerable delay
π_{max}	maximum tolerable block loss rate
QP	quantization parameter
TR_{max}	maximum transmission rate
S_L	security level
C	code rate
\vec{g}	rate distribution vector
\vec{bw}	available bandwidth vector
\vec{d}	delay vector
π	end-to-end packet loss rate

Partial Encryption of Video

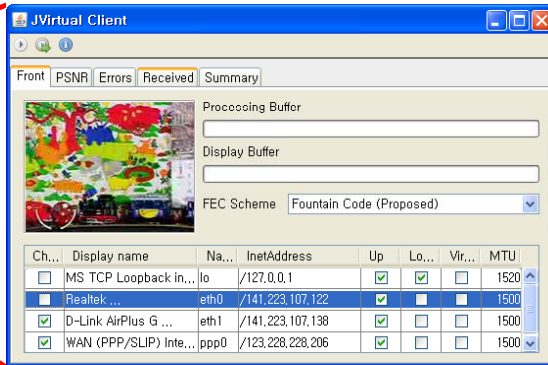
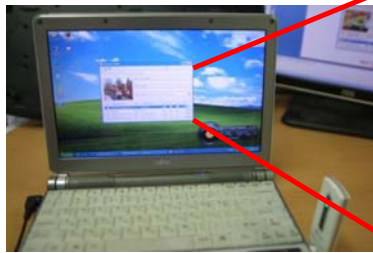
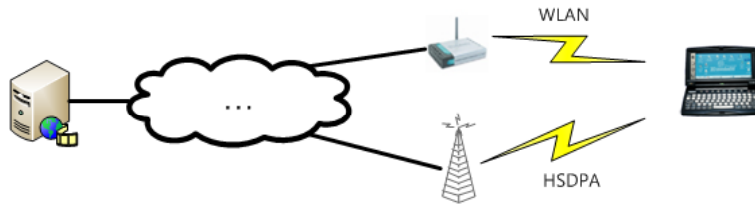


Proposed Algorithm

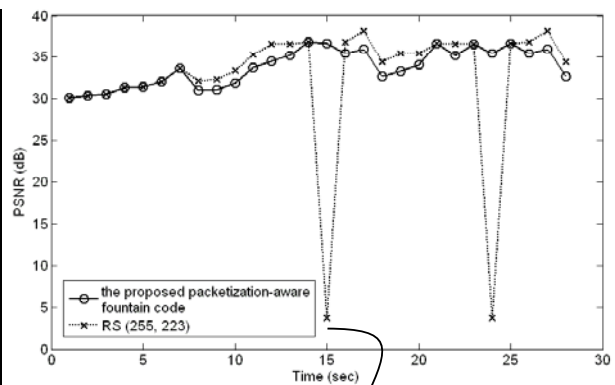


Implementation

- Real-time video transmission over HSDPA and WLAN



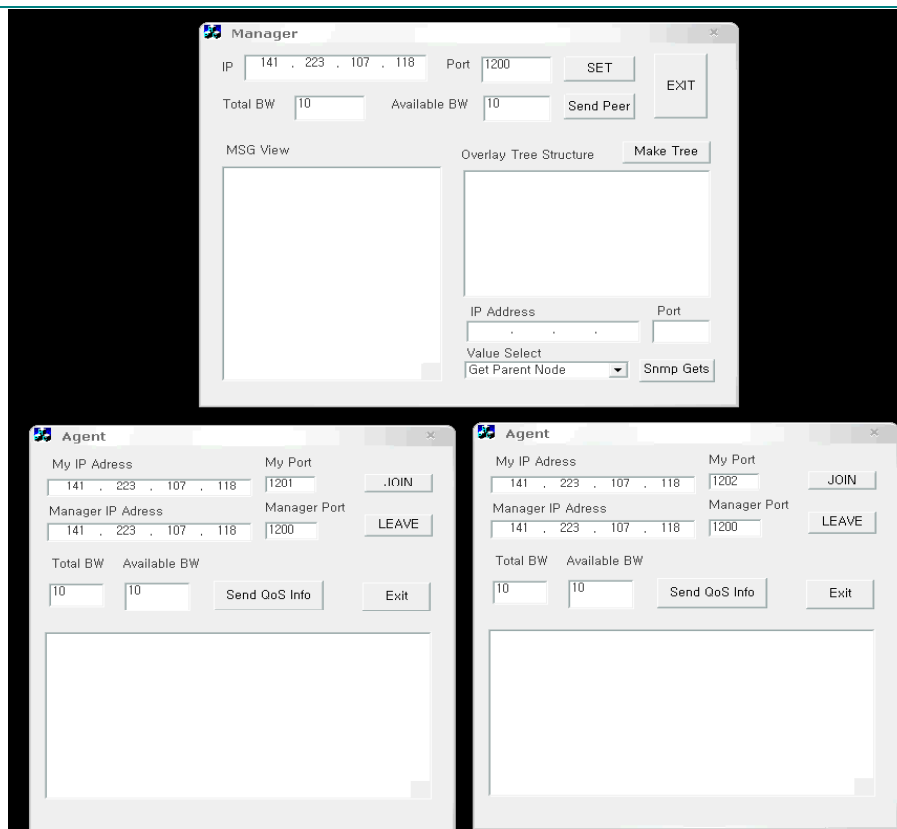
Video sequence	Used networks	Avg. PSNR (dB)	Avg. video rate (bps)
Foreman	HSDPA	31.59	126,320
	WLAN	32.57	157,781
	HSDPA and WLAN	34.47	302,245
Harbour	HSDPA	28.16	127,599
	WLAN	29.15	159,828
	HSDPA and WLAN	31.86	311,661
Mobile	HSDPA	24.79	133,178
	WLAN	25.01	168,181
	HSDPA and WLAN	26.78	311,198
Soccer	HSDPA	31.23	119,106
	WLAN	32.13	147,709
	HSDPA and WLAN	34.34	292,783



Demonstration

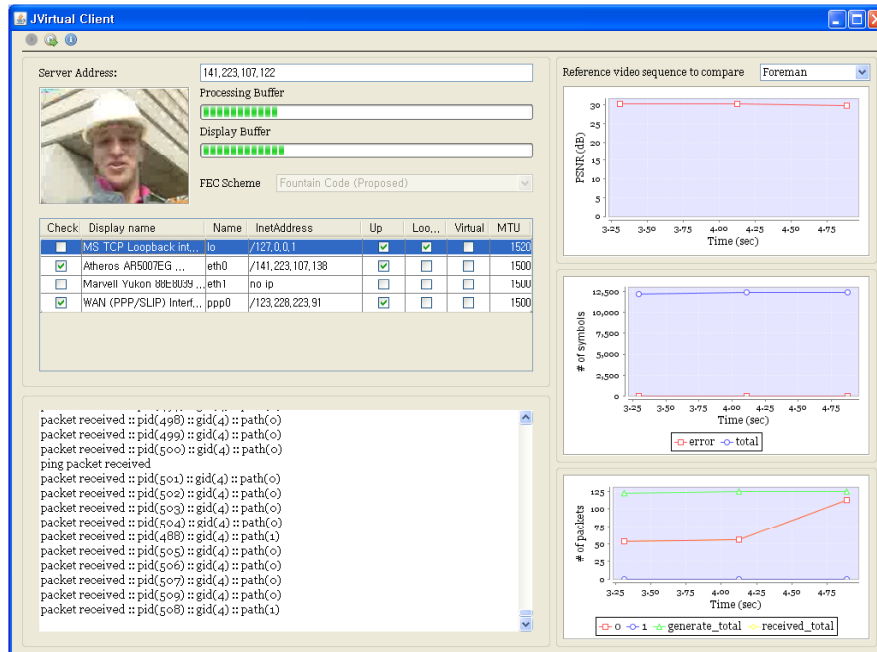


Overlay Multicast System

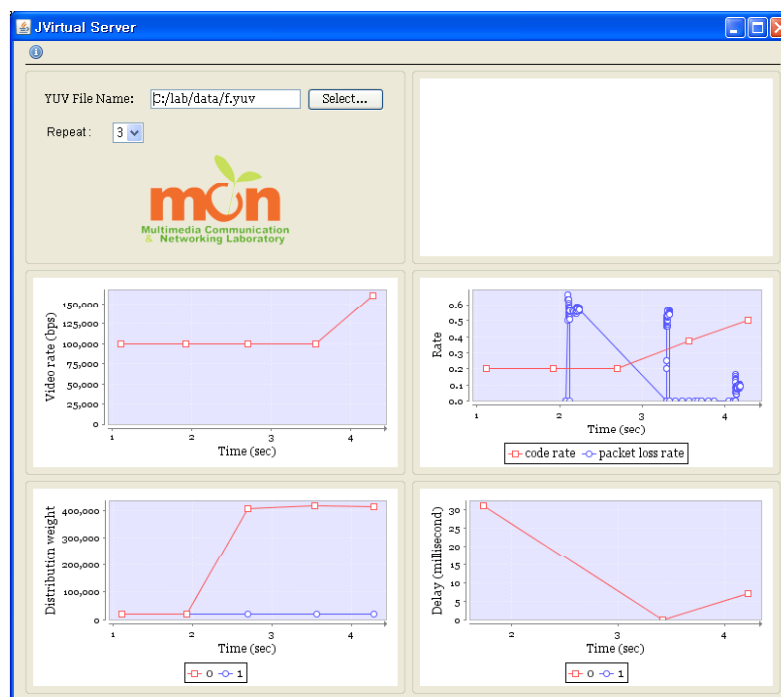


Multi Path Video Transmission System

Client



Server



Future Works

Dynamic Service Configuration

