MULTI-LAYER VIDEO STREAMING WITH HELPER NODES USING NETWORK CODING

Pouya Ostovari, Abdallah Khreishah, and Jie Wu
Computer and Information Sciences
Temple University

IEEE MASS 2013
Agenda

- Introduction
  - Network Coding Background
  - Priority-Based Network Coding
- Layered Video Streaming
  - Linear Programming
  - Distributed Solution
- Conclusions and Future Work
Network Coding in Wired Networks

- Single multicast session
  - Bottleneck problem (Ahlswede’00)

![Diagram of network coding](image)

No coding

Coding
Network Coding Classification

- **Local**
  - Hop-by-hop decoding
  - XOR operation

- **Global**
  - Decoding at the destination
  - Linear network coding (on a finite field)
Network Coding Classification

- **Intra-flow**
  - Within a flow
  - Robustness enhancement

- **Inter-flow**
  - Between different flows
  - Throughput/capacity enhancement

- **Joint inter- and intra-flow**
  - Within flow and between flows
Priority-Based Approaches

- New twist on the classic unequal error protection

Symbol-Level NC

Priority 1
Priority 2
Priority 3

Video Streaming NC
Video Streaming

- Delivering video stream using different resolutions to satisfy different client needs/constraints

- Multi-Layer Coding (Multi-resolution)
  - Base layer
  - Enhancement layers

- Multiple Description Coding (MDC)
  - Multiple independent video substreams
  - Receiving more substreams increases the video quality

<table>
<thead>
<tr>
<th>Substream</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Substream_1</td>
<td>Resolution_1</td>
</tr>
<tr>
<td>Substream_2</td>
<td>Resolution_2</td>
</tr>
<tr>
<td>Substream_N</td>
<td>Resolution_N</td>
</tr>
</tbody>
</table>
Inter-Layer Coding Strategies

- Random linear network coding (RLNC)

  \[ \alpha_1 L_1 + \beta_1 L_2 + \gamma_1 L_3 \]
  \[ \alpha_2 L_1 + \beta_2 L_2 + \gamma_2 L_3 \]
  \[ \alpha_3 L_1 + \beta_3 L_2 + \gamma_3 L_3 \]

- Triangular coding

  - Prefix coding

    \[ \alpha_1 L_1 \]
    \[ \alpha_2 L_1 + \beta_2 L_2 \]
    \[ \alpha_3 L_1 + \beta_3 L_2 + \gamma_3 L_3 \]

- Packets in lower layers are more important

  - Included in more coded packets
  - More chances to be decoded
Multi-Layer Video Streaming with Helpers

- **Links**
  - Cost: direct download from the server
  - Reliable links

- **Link capacity**
  - High capacity links: server to helpers
  - Low capacity links: helpers to users

- **Use of helpers**
  - System scalability for more users
  - Helpers: limited capacity and bandwidth
Resource Management

- Optimal resource management

Questions:

- **Content placement**: Which packets of each video should a helper node store?

- **Bandwidth allocation**: Which packets, and to which users, should each helper serve?

- NP-complete
Network coding changes the problem to a linear programming.

Storing $x$ percent of each segment.

No longer NP-complete.

Flow-based model using network coding.
Multi-Layer Video

- **Benefits of multi-layer**
  - Provides smooth playback for the users
  - Reduces the load on the server with a fixed number of users
  - More layers increases system scalability
Motivation

- Single video with 4 packets
- **No-layer approach**
  - 4 packets in the same layer
  - Load on the server: 4

\[ P_i = \sum_{j=1}^{4} a_{i,j} p_j \]
Motivation

- Single video with 4 packets
- Intra-layer approach
  - (Ostovari, Khreishah, and Wu, 2013)
  - 2 packets per layer
  - Load on the server: 2

Layer 1: $p_{1,1} | p_{1,2}$
\[ P_{1,i} = \sum_{j=1}^{2} \alpha_{i,j} p_{1,j} \]

Layer 2: $p_{2,1} | p_{2,2}$
\[ P_{2,i} = \sum_{j=1}^{2} \alpha_{i,j} p_{2,j} \]
Motivation

- Single video with 4 packets
- **Inter- and intra-layer coding**
  (Ostovari, Khreishah, and Wu, 2013)
  - Prefix coding (Triangular)
  - 2 packets per layer
  - Load on the server: 0

**Intra-layer coding**

**Prefix coding (Triangular)**

Layer 1: \( P_{1,1}, P_{1,2} \)

\[ P_{1,i} = \sum_{j=1}^{2} \alpha_{i,j} \cdot p_{1,j} \]

Layer 2: \( P_{2,1}, P_{2,2} \)

\[ P_{2,i} = \sum_{j=1}^{2} \alpha_{i,j} \cdot p_{2,j} \]
VoD with Intra-Layer NC

\[
\max \sum_{i,k: u_i \in U, m_k = q_i} \sum_{j,l: h_j \in N(u_i), l \leq c_i} x_{ji}^{kl}
\]

Objective function (maximize upload rate from helpers to users)

\[
\text{s.t.}
\]

\[
x_{ji}^{kl} \leq f_{ji}^{kl} \times r_k, \quad \forall j, i, l, k: u_i \in N(h_j), m_k = q_i, l \leq c_i
\]

The upload rate of a helper cannot exceed the rate of the stored videos

- \( x_{ji}^{kl} \): Upload rate of link from helper \( h_j \) to user \( u_i \) over layer \( l \) of video \( m_k \)
- \( f_{ji}^{kl} \): Fraction of the layer \( l \) of video \( k \) that is stored on helper \( h_j \)
- \( r_k \): Rate of each layer of video \( m_k \)
- \( N(u_i) \): Adjacent helpers to user \( u_i \)
- \( u_i \) s request: \( (c_i, q_i) = \) (quality level, video)
VoD with Intra-Layer NC

\[ \sum_{i,k:u_i \in N(h_j), m_k = q_i} \sum_{l \leq c_i} x_{ji}^{kl} \leq B_j, \quad \forall j : h_j \in H \]

\[ \sum_{k \in M} \sum_{l: l \leq L} f_j^{kl} \times v_k \leq S_j, \quad \forall j : h_j \in H \]

\[ \sum_{j : h_j \in N(u_i)} x_{ji}^{kl} \leq r_k, \quad \forall i, k, l : u_i \in U, m_k = q_i, l \leq c_i \]

- \( B_j \): The bandwidth limit of helper \( h_j \)
- \( S_j \): The capacity limit of helper \( h_j \)
- \( M \): Set of videos
- \( L \): Maximum number of layers

Bandwidth constraint from each helper to users

Storage constraint for each helper

Limits the total download of a user to the rate of the video
VoD with Inter- and Intra-Layer NC

- The difference is in the last constraint

\[
\sum_{l=1}^{l'} \sum_{j: h_j \in N(u_i)} x_{j_i}^{kl} \leq r_k \times l', \quad \forall i, k, l': m_k = q_i, 1 \leq l' \leq c_i
\]

Prefix limit on the download rate of each layer

<table>
<thead>
<tr>
<th>Intra-layer coding</th>
<th>Inter and Intra-layer coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layer 1</td>
<td>Layer 1</td>
</tr>
<tr>
<td></td>
<td>&lt; r</td>
</tr>
<tr>
<td>Layer 2</td>
<td>Layer 1+2</td>
</tr>
<tr>
<td></td>
<td>&lt; 2r</td>
</tr>
<tr>
<td>Layer 3</td>
<td>Layer 1+2 +3</td>
</tr>
<tr>
<td></td>
<td>&lt; 3r</td>
</tr>
</tbody>
</table>

- \(l\) : variable for the prefix relation
- The objective function and other constraints are the same
Live Streaming (TV)

- Videos are broadcast to the users
- Synchronous playback
  - Helpers do not need to allocate separate bandwidths to adjacent users that are watching the same video

\[
\begin{align*}
\text{Total bandwidth: } & x_1 + x_2 \\
\text{Total bandwidth: } & x
\end{align*}
\]
Distributed Algorithm

- **Dual optimization**
  - Solving Lagrange dual using the gradient method

- **Helper** $h_j$
  - Start from empty storage and dynamically adjust the amount of stored videos
  - Update and transmit Lagrange variables to adjacent users

- **User** $u_i$
  - Update and transmit Lagrange variables to adjacent helpers

- **Step control**
  - Slope of changes: fast convergence vs. oscillation
Simulations Setting

- MATLAB environment
- 100 random topologies
  - Random connections of helpers and users
  - Helpers: random bandwidth and capacity limit
  - Users: random requests
- Comparing with the optimal non-layer approach
- Measuring
  - Load on the server
  - Convergence to optimal solution in dynamic environments

<table>
<thead>
<tr>
<th>Video’s rate</th>
<th>Video’s size</th>
<th>Bandwidth capacity</th>
<th>Storage capacity</th>
<th>Num. of adjacent helpers to a user</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,2] kbps</td>
<td>[0.5,2] MB</td>
<td>[2,4] kbps</td>
<td>[0.5,2] MB</td>
<td>[1,3]</td>
</tr>
</tbody>
</table>
Simulation Results (Load)

- VoD
- Number of videos: 5
- Number of layers: 5
- DIST: a non-layer approach with intra-layer coding (Hao et al. 2011)

Number of helpers: 20

Number of users: 40
Simulation Results (Load)

- VoD
- Number of users: 50
- Number of helpers: 20

Number of layers: 5
Number of videos: 5
Simulation Results (Load)

- VoD
- Number of layers: 4
- Single video

Number of helpers: 10

Number of users: 10
Simulation Results (Convergence)

- VoD
- Layers: 4
- Videos: 5
- Users: 50
- Helpers: 20

Convergence to the optimal solution (LP)

The fraction of each video on helper h5
Simulation Results (Dynamic Users)

- VoD
- Layers: 4
- Videos: 5

Initial Users: 10
Helpers: 10

Adding 5 users
Adding 5 users
Adding 5 users
Removing 5 users
Removing 5 users

Convergence to the optimal solution (LP)
The fraction of each video on helper h8
Simulation Results (Dynamic Helpers)

- VoD
- Layers: 4
- Videos: 5
- Users: 20
- Initial helpers: 6

Adding 3 helpers
Adding 3 helpers
Removing 3 helpers

Convergence to the optimal solution (LP)

The fraction of each video on helper h3
Future Work and Challenges

- Other objectives
  - Fairness, layers with different weights, …
- Extension of layered VoD with unreliable links
  - Using symbol-level transmission work in layered VoD
- Cost-efficient helper provisioning
  - Based on user demands and resource availability
- Real implementation
Conclusions

Priority-Based Network Coding

- Data transmission
  - Transmitting the more important data with more redundancy
- Triangular coding in multi-layer video streaming
  - Increasing the number of received layers
- VoD and live streaming using helper nodes in multi-layer video streaming
  - Minimizing the load on the server