Aalto University
School of Science

Development and measurement of a scalable video coding based client

Jun Liu

Department of Computer Science and Engineering
Aalto University, School of Science
jun.liu@aalto.fi

August 21, 2014

Introduction

- Background
- Motivation And Contribution
- Software Architecture
- Adaption Algorithm Improvement
- Measurement Results
- Conclusion And Future Works

Background

- Dynamic Adaptive Streaming over HTTP (DASH, 2010): multicoded video contents (e.g. AVC), adaption to a dynamic network conditions, re-using existing HTTP cache infrastructures
- Scalable video coding (SVC, 2007): a new video compression standard, high scability (quality, spatial, temporal), layer based architecture
- SVC-DASH:
  - Advantages: save server disc spaces, good cache performance, high bandwidth utility, various mobile devices
  - Drawbacks: complex coding, high energy consumption

Motivation and Contribution

- Motivation:
  - Lack of software developed for SVC-DASH clients
  - Most of the proposed SVC-DASH clients are based on video decoding system
  - Difficulty in development and maintainence and not user-friendly
- Contribution:
  - A lightweight SVC-DASH client based on PYTHON
  - Client based on Mplayer, better for user experience measurements
  - Modification of adaption algorithm
  - User experience measurements, e.g. starting delay, quality selection performance and happiness metric

Software Architecture

- Software Architecture:

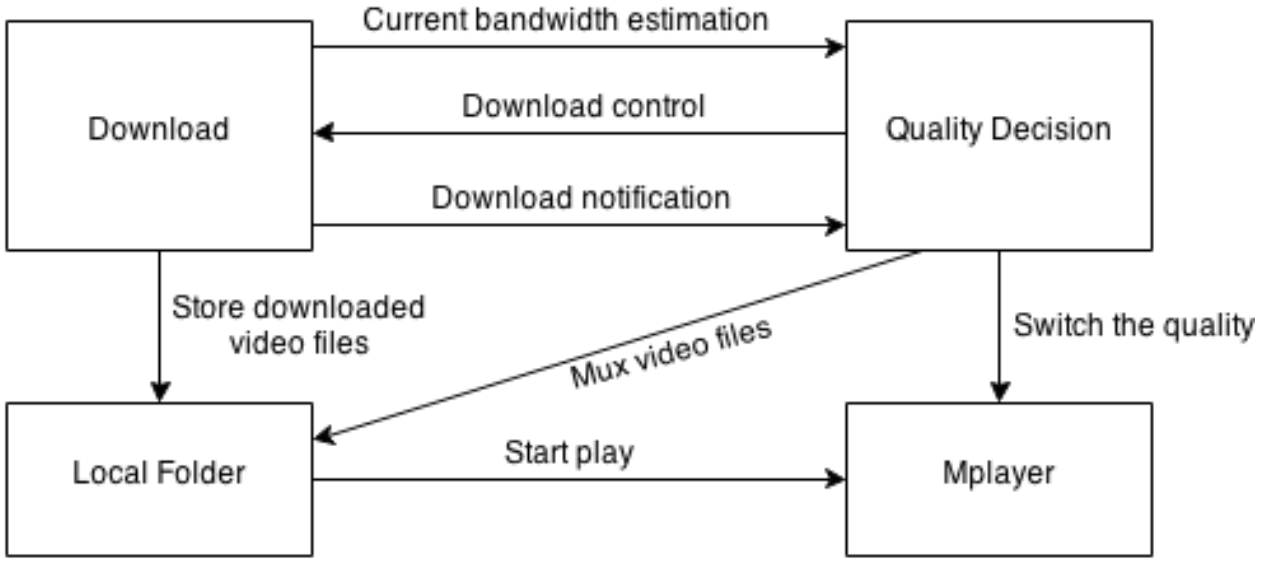

Figure : Software architecture

Adaption Algorithm Improvement

- Hard switch algorithm:
  Supposed estimated bandwidth is b_j for jth segment, t_i is the threshold speed of ith layer:
  If $t_i < b_j < t_{i+1}$:
  Select ith layer for $j+1$th segment
- Improved soft switch algorithm:
  If $0.9 * t_i < b_{j-1} < 1.1 * t_{i+1}$ and $0.9 * t_i < b_j < 1.1 * t_{i+1}$:
  Do not change the layer level for jth segment
  Else if $b_j > t_i$:
  Select ith layer for $j+1$th segment
- Happiness metric: $(\prod_{i=0}^{N-1} q_i)^{1/N} - \frac{1}{N-1} \sum_{i=0}^{N-2} (q_i - q_{i+1})^2$
  N is the total segment numbers, q_i is the selected layer level for ith segment

Measurement Results

Measurement parameters:

- Video file: 704x576, total 500 frames, 3 layers
- Demux the video as 2/5/10 segments, files saved on server.
- Layer bandwidth thresholds are 24.782KB/s, 208.604KB/s, 596.307KB/s
- Use ’trickle’ for bandwidth limitation, and ’Mplayer’ for video play

Measurement Results

Starting delay:

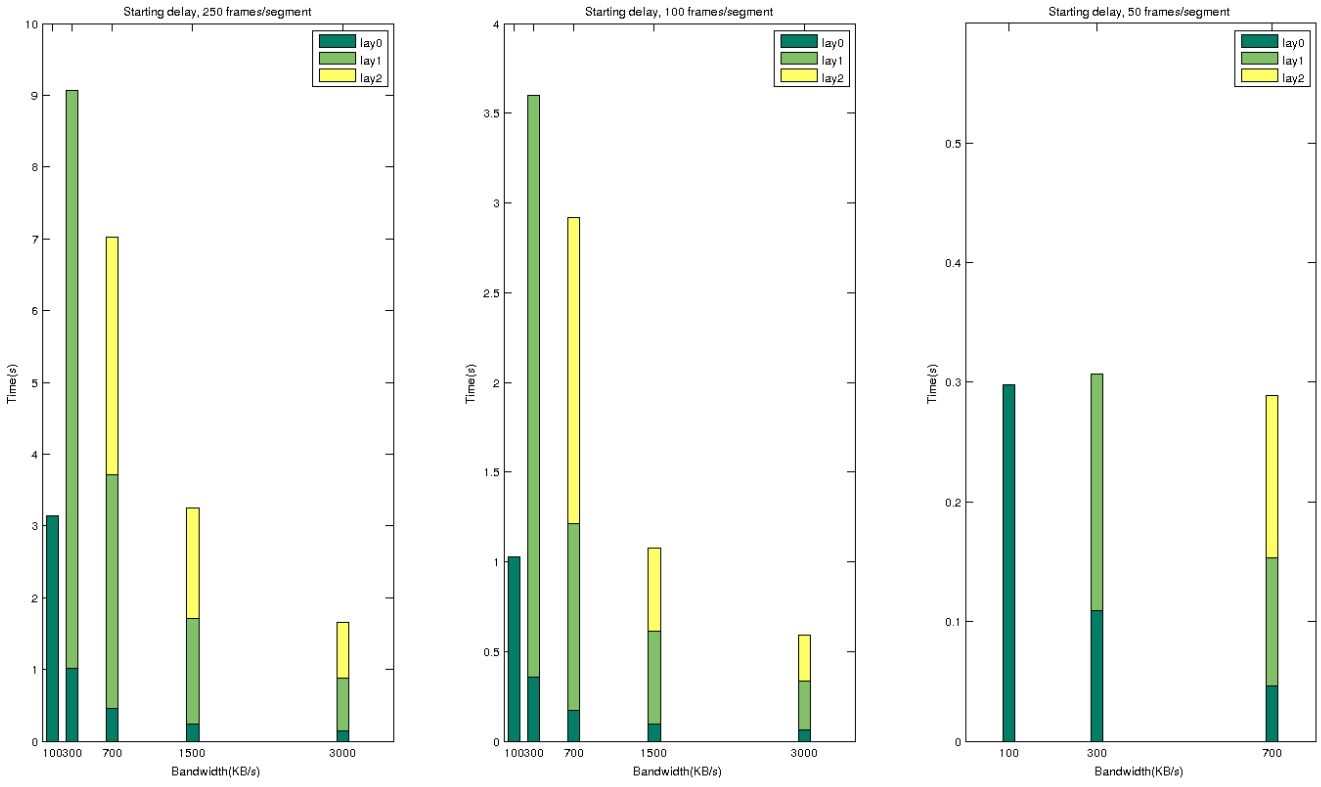

Figure : Starting delay for various size of segments

Measurement Results

Layer Selection:

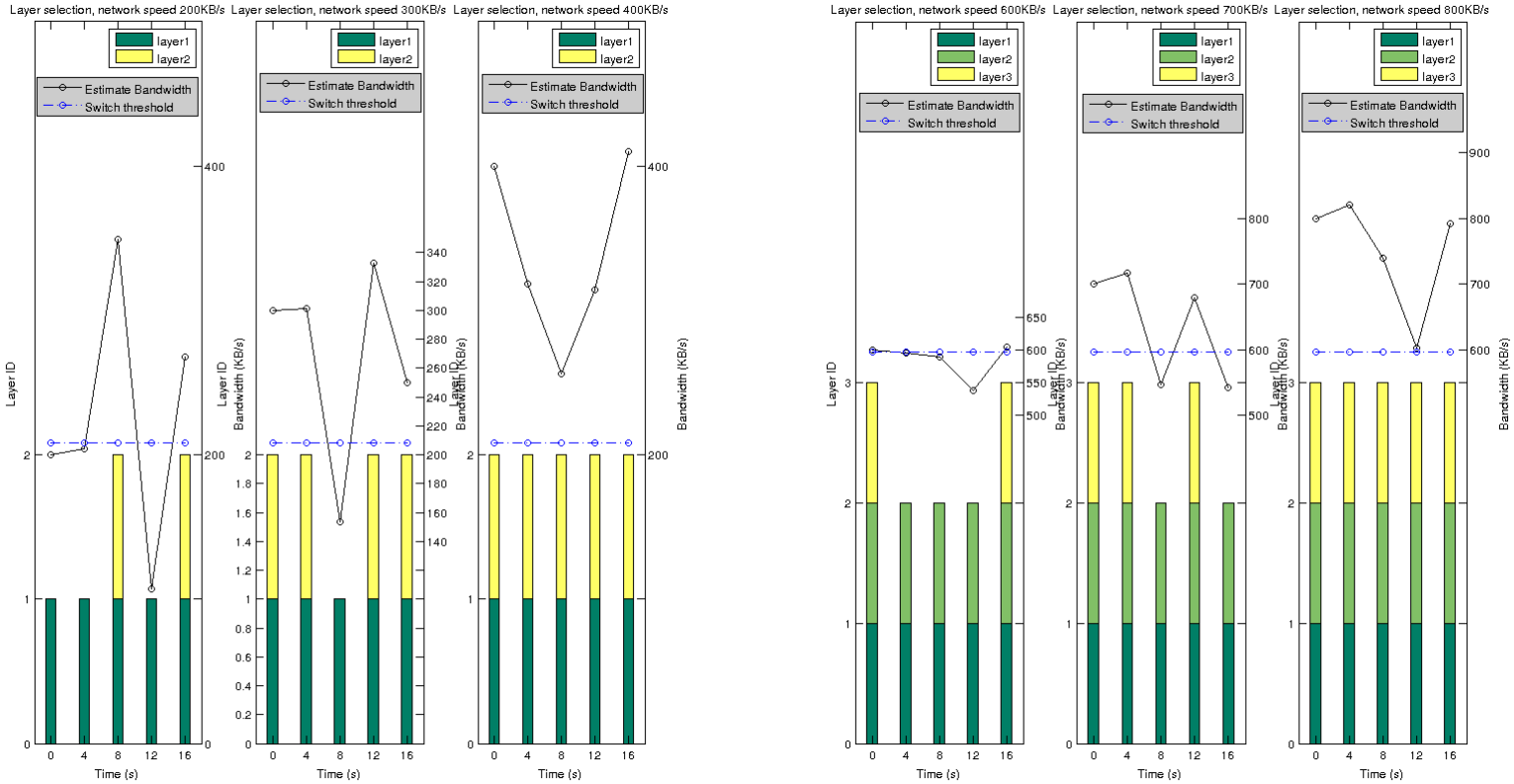

Figure : Layer selection under different bandwidth

Measurement Results

Algorithm comparison: happiness1 = 1.85, happiness2 = 2.3

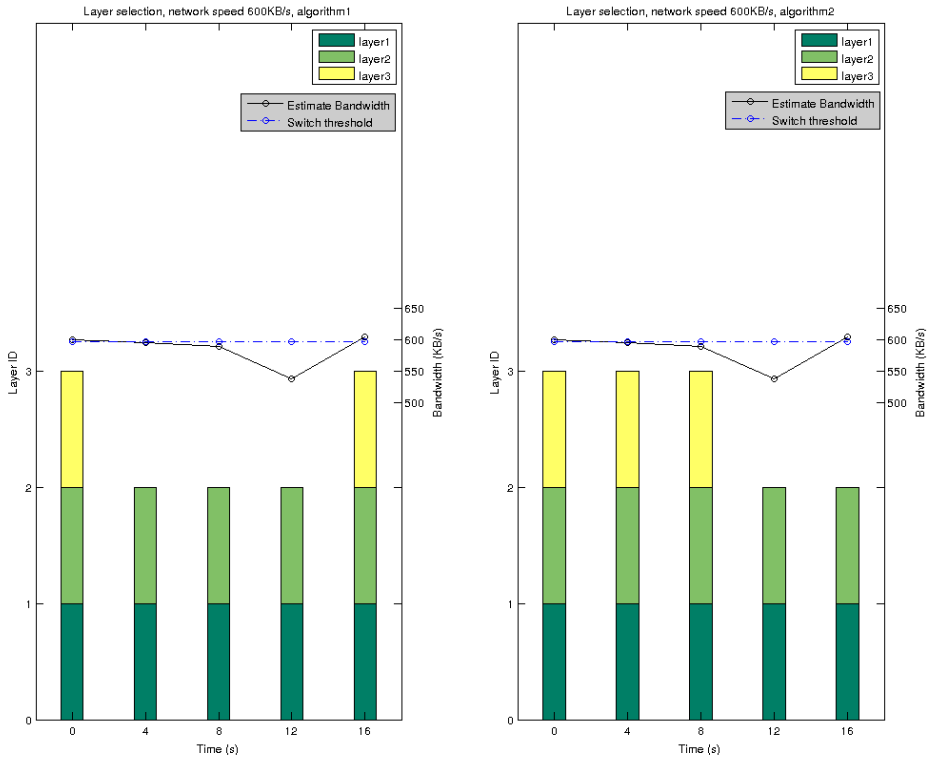

Figure : Comparison of algorithm

Conclusion and Future Works

Conclusion:

- A lightweight client for SVC-DASH
- Design an algorithm for adaption
- User oriented measurements

Future works:

- Add buffers in the client for the adaption for the network flexibility
- Adaption algorithm improvement based on the user happiness metrics
- More user oriented measurements, e.g. buffering ration, average bitrate, rendering quality and rate of buffering events.