CS 5262: Multimedia Networking and Systems

Video Quality Metrics

Instructor: Cheng-Hsin Hsu

Outline

- **Overview on Video Quality Metrics**

- **A Sample Application**

  - **Joint Packet Scheduling and Stream Adaptation in Multihomed Video Streaming**

- **Tools for Calculating Video Quality and Conducting Simulations/Experiments**

Overview of Video Quality Metrics

Visual Impairments Caused by Packet Loss

Standard H.264 Video
10% Packet Loss

RADVISION Scalable Video
10% Packet Loss

0   1   2   5   10   15   20

Packet Loss Rate %

RADVISION®
Delivering the Visual Experience®

Quality of Experience and Quality of Service

- **QoE: Subjective measurements of users' experience**

  - **What a user (customer) wants? ← about human beings**

- **QoS: Objective measurements of the delivered service**

  - **How good is the received content? ← about content**

- **We refer to them as subjective and objective quality metrics**

Quality Metrics

- **Subjective Metrics**
  - Hire people to score individual videos
  - Expensive, cannot be realtime ← E.g., cable TV systems cannot use subjective metrics to recover from network congestion
  - Not reproducible ← rerunning the experiments leads to different results

- **Objective Metrics**
  - Algorithms to analyze content, or infer video quality based on network conditions ← Cheap
  - Can be deployed in live networks for realtime monitoring
  - Reproducible

Subjective Metrics

- **Voice – Mean Opinion Score (MOS)**

  - Users grade voice quality from 1 to 5

  - Above 4 is good quality

  - Various variations with difference score ranges

- **Video – ITU-R BT.500**

  - Several modes are defined

  - E.g., Double Stimulus Impairment Scale (DSIS): first show the full-quality video, then show the impaired one. Viewers are informed the order. Viewers are asked to score the impaired video.
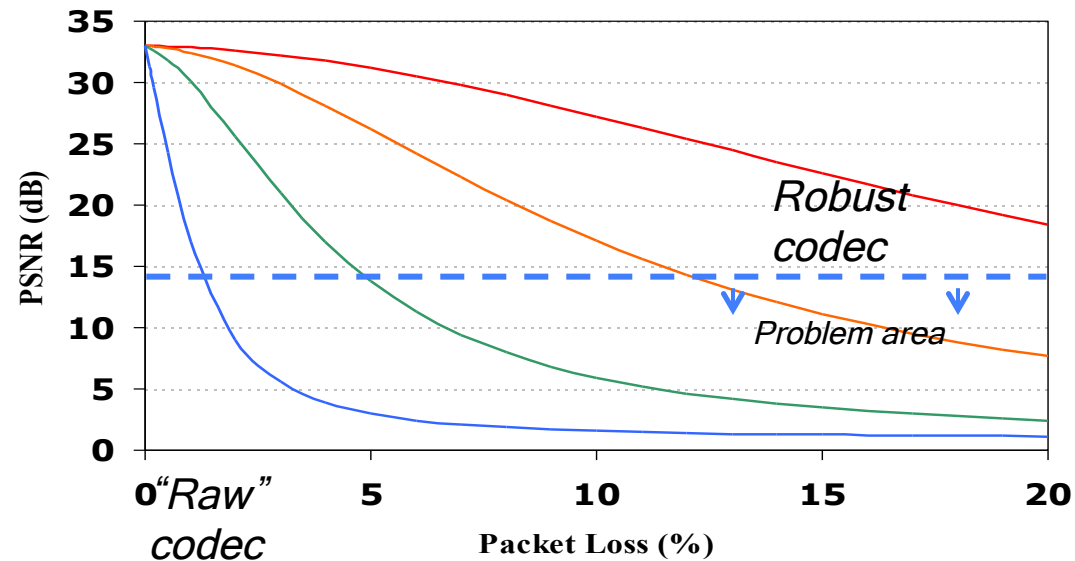
Objective Metrics (1/2)

- **Packet Based Metrics**
  - Use network measurements and (optionally) codec properties to infer the degraded video quality
  - Low complexity and work without original videos

- **Example V-Factor**
  - $V = f(QER, PLR, R)$
  - QER: codec quality
  - PLR: packet loss ratio
  - R: video complexity
  - Adopted by Sprint

Source: http://www.dsp-ip.com

Objective Metrics (2/2)

- **Content Based Metrics**
  - Compute the quality level using the video itself
  - Used in research labs for, e.g., comparing video codec performance

- **Classified into three groups**
  - **Full reference**: assuming both original and impaired videos are available ← less practical, but widely used in research labs
  - Reduced reference: original videos are analyzed and a summary is compared against the impaired video
  - No reference: metrics that do not need original videos ← ideal metrics

Full Reference Metrics

- **Most quality metrics consider Y-component (luminance) only**

- **MSE (mean-square-error) and PSNR (peak signal-to-noise ratio) are pixel based metrics**

$$MSE = \frac{1}{M} \times \sum_{i=1}^{M} (x_i - y_i)^2$$

$$PSNR(dB) = 10log_{10} \sum_{i=1}^{M} \frac{255^2}{(x_i - y_i)^2}$$
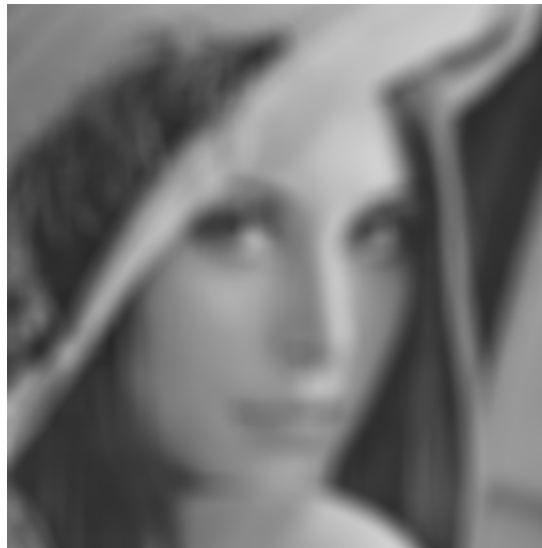
Problems with MSE/PSNR

MSE=0, original picture

- **MSE/PSNR does not map to user-perceived quality all the time**

- **Still researchers are using them ← Why?**
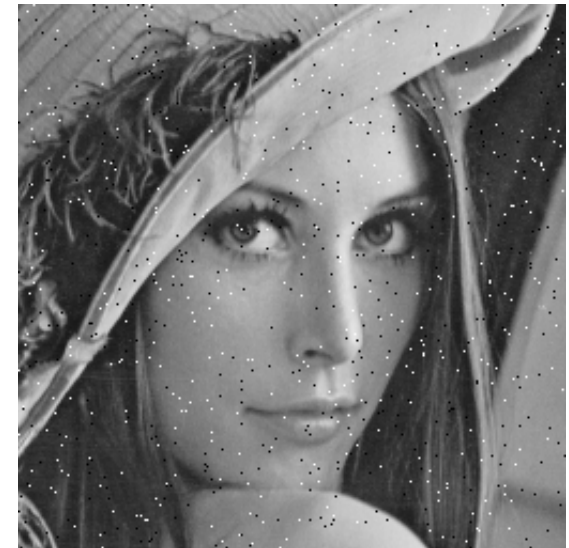

MSE=225, MSSIM=0.949
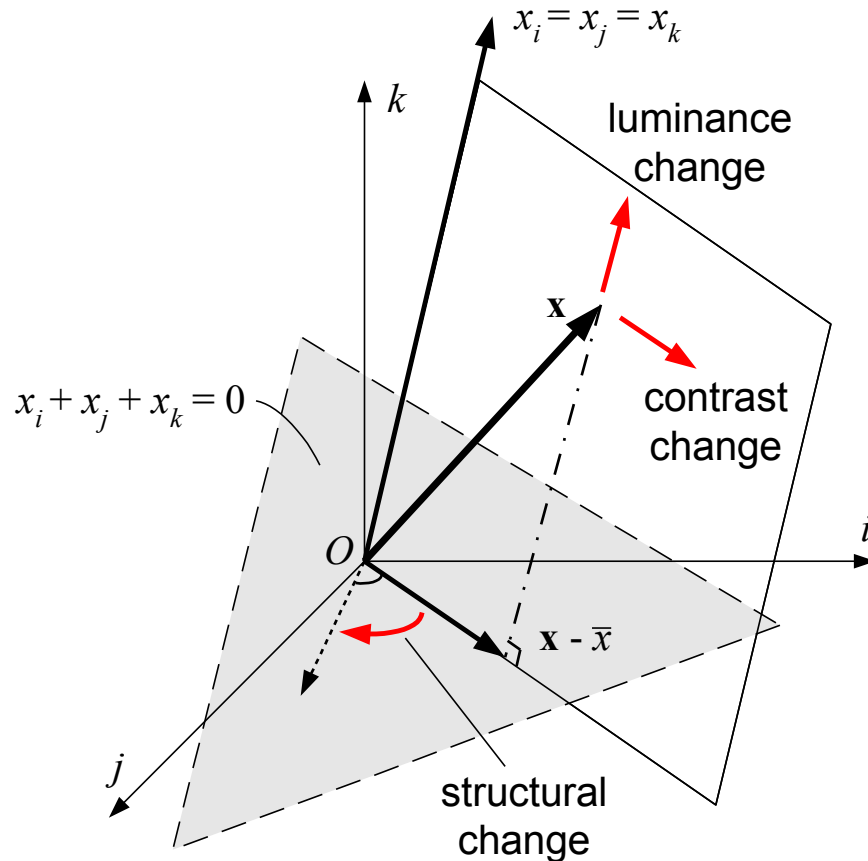

MSE=225, MSSIM=0.688

MSE=225, MSSIM=0.723

Structural Similarity Index (SSIM)

- **New metric proposed in 2004, which measures the similarity between the original and impaired images (extension for videos have also been proposed)**

- **Designed to address the limitations of MSE/PSNR**

- **Between [0, 1], where 1 indicates (iff) two images are identical**

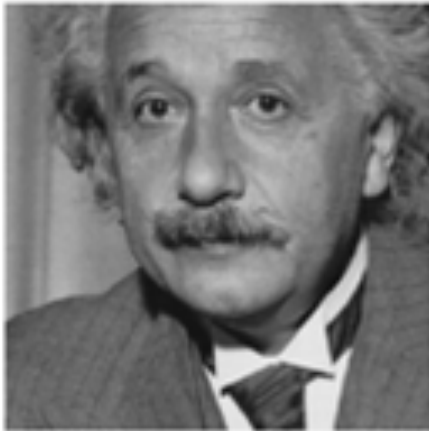# Structural Similarity Index (SSIM)

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = l(\boldsymbol{x}, \boldsymbol{y}) \cdot c(\boldsymbol{x}, \boldsymbol{y}) \cdot s(\boldsymbol{x}, \boldsymbol{y})$$

Source: Wang *et al.*, *IEEE Trans. Image Processing,* 2004

Examples of SSIM

(a) MSE=0, SSIM=1
CW-SSIM=1
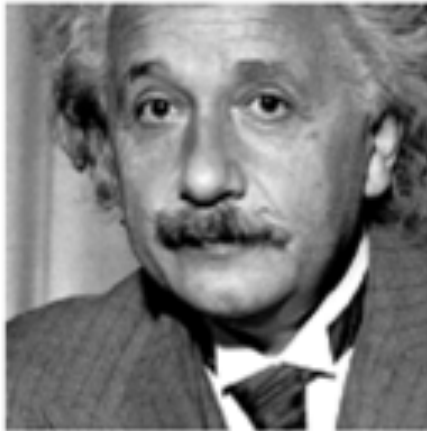
(b) MSE=306, SSIM=0.928
CW-SSIM=0.938
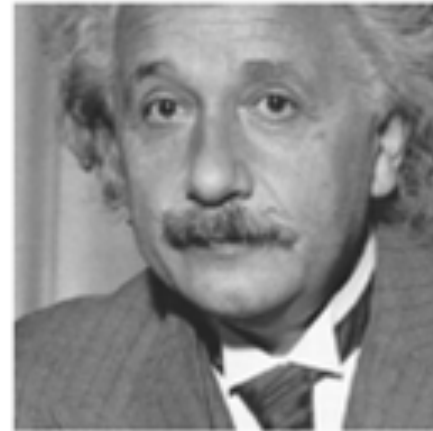
(c) MSE=309, SSIM=0.987
CW-SSIM=1.000
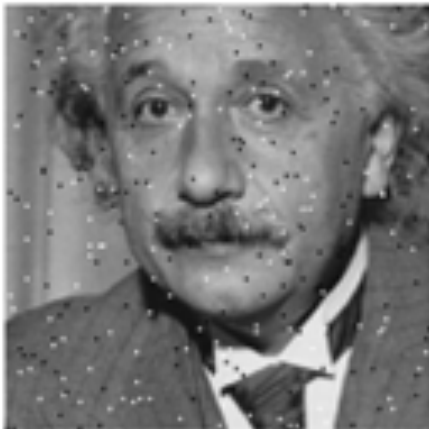
(e) MSE=313, SSIM=0.730
CW-SSIM=0.811

(f) MSE=309, SSIM=0.580
CW-SSIM=0.633

(g) MSE=308, SSIM=0.641
CW-SSIM=0.603

Source: https://ece.uwaterloo.ca/~z70wang/research/ssim/

Performance Comparison

PSNR vs. MOS

MSSIM vs. MOS

Source: https://ece.uwaterloo.ca/~z70wang/research/ssim/
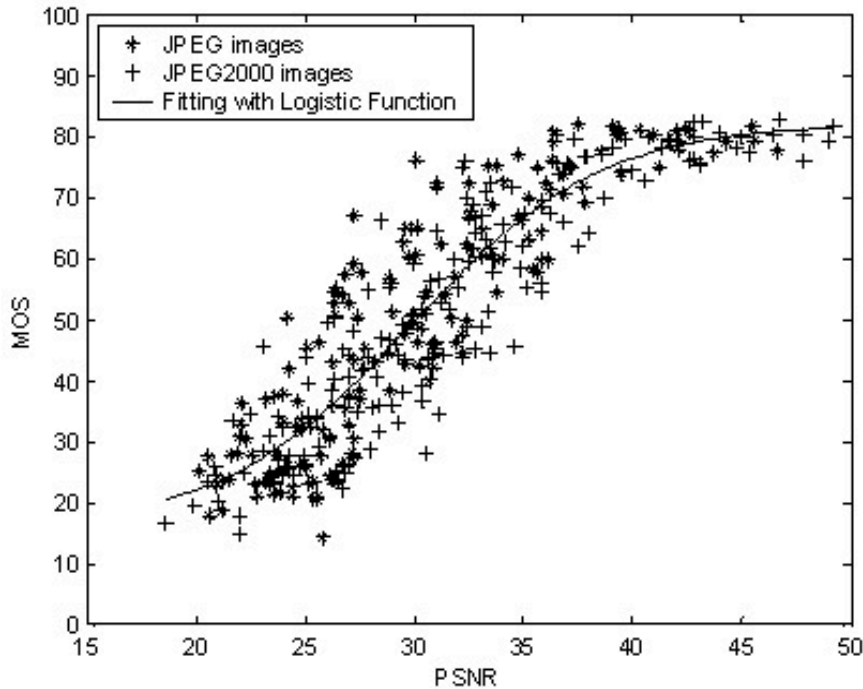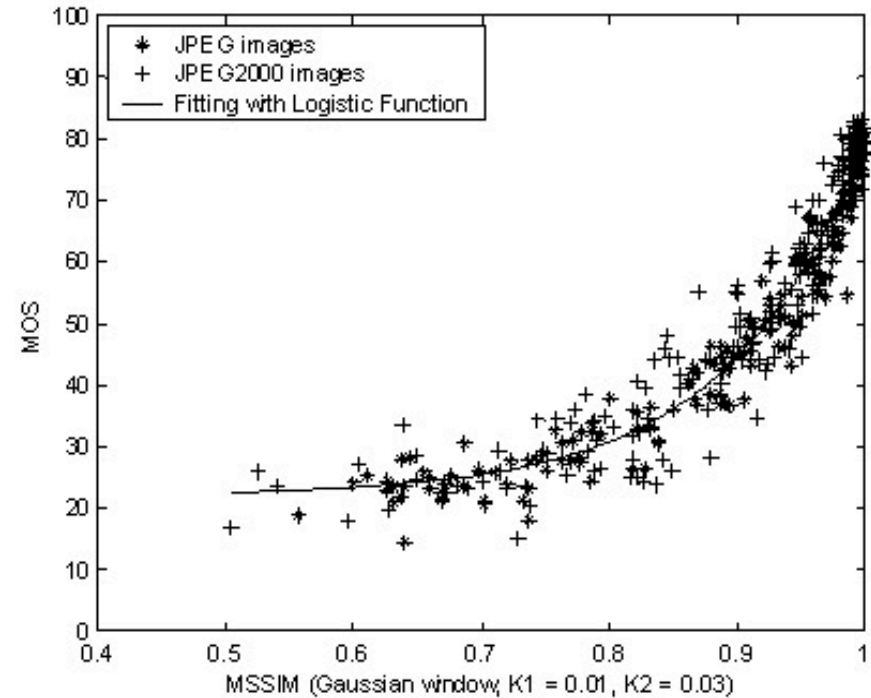
Multihomed Video Streaming –

A Sample Project using Objective Video Quality as Optimization Criterion

Offloading Traffic from Cellular Networks

- Video streaming has high bandwidth requirements
- However, T-mobile and AT&T recently reported more than 50 times of data traffic increase [Open Mobile Summit '09]
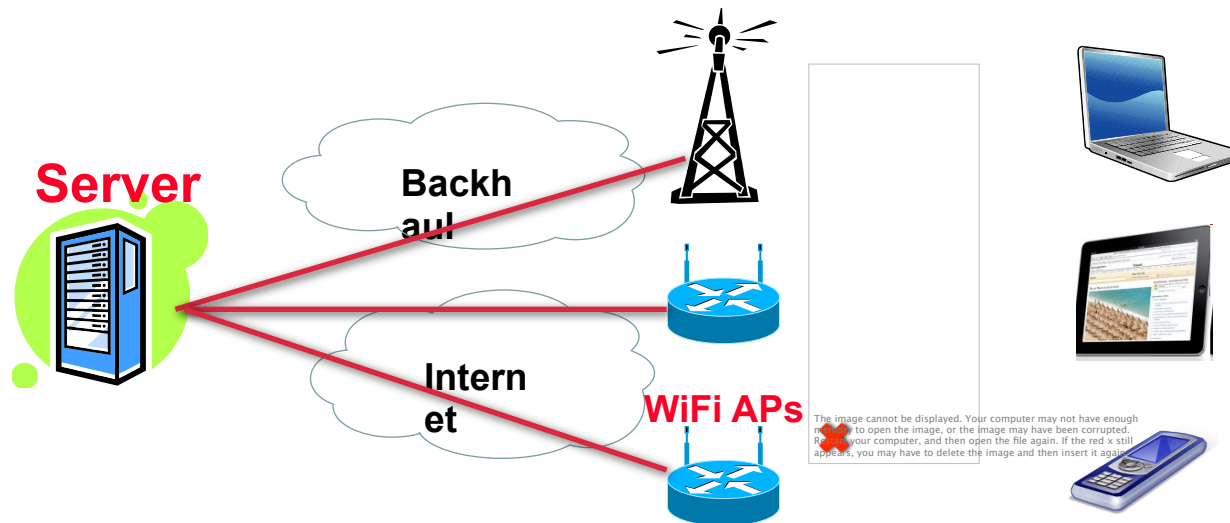
- **This is called** **multihoming,** **which is attractive to**
  - **ISPs, such as T-Mobile, for lower transit cost**
  - **Subscribers for better quality-of-service**

Dynamic Network Coditions

□ **Problem: access networks are heterogeneous and dynamic**

□ **Employ scalable video: frames are coded into multiple layers**

  – **incremental quality improvement**

  – **complicated interdependency due to prediction**
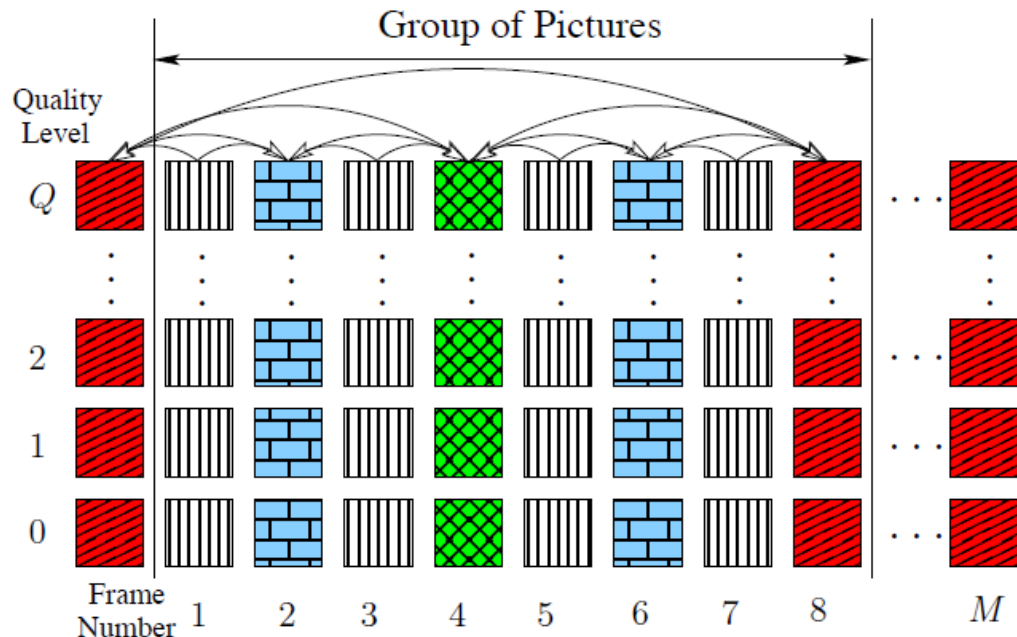
Challenges and Problem Statement

- **Determine streaming rate on each access network is hard** [Hsu ISM'10]

  - streaming at a rate close to end-to-end network capacity leads to congestion, and late packets
  - streaming at a low rate wastes available resources
  - need a network model to proactively prevent congestion

- **Packets of scalable streams have complex inter-dependency**

  - need a video model to predict expected quality

- **The problem: determine (i) what video packets to send, (ii) over which network interface, and (iii) at what rate, so that the overall streaming quality is maximized**

Notations

□ **Scalability**

- **Client: $u=1,\ldots,U$**

- **Temporal: Different frames with inter-frame prediction $m=1,\ldots,M_u$**

- **Spatial: Quality layers $q=0,\ldots,Q_u$**

- **Multihoming: networks $n=1,\ldots,N$**

- **Network Abstraction Layer Unit (NALU) : $g_{u,m,q}$**

□ **Scheduling**

If $g_{u,m,q}$ is sent over network n

- **Deterministic:** $x_{u,m,q,n} \in \{0, \boxed{1}\}$

- **Randomized:** $x_{u,m,q,n} \in [0, 1]$

Video Quality Model

□ **Truncation distortion: capturing loss of a NALU $g_{u,m,q}$**

- **A packet is decodable if all packets in lower quality ($q' < q$) layers are received**

$$e_{u,m} = \boxed{\hat{\delta}_{u,m}} + \sum_{q=0}^{Q_u} \left(1 - \prod_{q' \leq q} x_{u,m,q'}\right) \boxed{\delta_{u,m,q}}$$

Distortion if all packets are received

Additional distortion If $g_{u,m,q}$ is not decoded
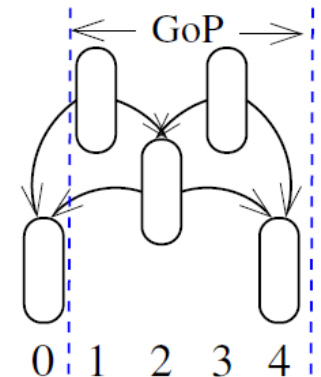
□ **Drifting distortion: capturing error propagation**

- **Inter-frame predictions based on imperfectly reconstructed parent packets, $P_{u,m}$**

- **Convex increasing function**

$$y_{u,m} = \alpha_{u,m} + \sum_{k \in P_{u,m}} \boxed{\beta_{u,m,k}} e_{u,k}$$

Nonnegative

- **Parameters: Estimated from actual data**

GoP

0 1 2 3 4

Network Model

□ **Packet loss probability (p_n) depends on**

  - **Rate: (r_n)**

  - **Available bandwidth (c_n)**

  - **Packet decoding deadline (t_0)**

□ **Model**

  - **M/M/1 model** $p_n = e^{-\frac{t_0(c_n - r_n)}{\alpha_n}}$

  - **Increasing in c_n, decreasing in r_n**

  - **α_n : linear regression parameter**

  - **accurate in streaming video applications [Zhu et. al '05]**

□ **Assumption : statistical independence of different networks**

  - **Good approximation using a two-timescale approach [Jiang et al. '10]**

  - **Network converges to steady-state in between scheduling events**

Problem Formulation

☐ **Cost minimization problem ← a cost function of distortion (MSE)**

- **Accounts for service differentiation and fairness among users and frames**

$$\min_x \quad C(\boldsymbol{d})$$

Cost function (increasing, convex)

$$\text{s.t.} \quad r_n = \sum_{u=1}^{U} \frac{F_u}{M_u} \sum_{m=1}^{M_u} \sum_{q=0}^{Q_u} s_{u,m,q} x_{u,m,q,n},$$

Rate

$$p_n = e^{-t_0(c_n - r_n)/\alpha_n},$$

Loss probability

$$x_{u,m,q} = \sum_{n=1}^{N} (1 - p_n) x_{u,m,q,n},$$

$$e_{u,m} = \hat{\delta}_{u,m} + \sum_{q=0}^{Q_u} (1 - \prod_{q' \leq q} x_{u,m,q'}) \delta_{u,m,q},$$

Not convex

$$y_{u,m} = \alpha_{u,m} + \sum_{k \in \boldsymbol{P}_{u,m}} \beta_{u,m,k} e_{u,k},$$

$$d_{u,m} = e_{u,m} + y_{u,m},$$

$$\sum_{n=1}^{N} x_{u,m,q,n} \leq 1,$$

$$x_{u,m,q,n} \in \{0, 1\}.$$

Randomized scheduling

SRDO

---

0.   INPUT: $P_{\max}$ is the maximum packet loss rate
1.   **let** $\boldsymbol{x} = \{x_{u,m,q,n} = 0 \mid \forall u, m, q, n\}$
2.   **sort** $g_{u,m,q}$ on $\frac{\delta_{u,m,q}}{s_{u,m,q}}$
3.   **for** $\hat{n} = \mathrm{argmin}_{n=1}^{N} p_n$
4.   **let** $g_{\hat{u},\hat{m},\hat{q}}$ be the next unsent NALU
5.   **if** sending $g_{\hat{u},\hat{m},\hat{q}}$ on $\hat{n}$ causes $p_{\hat{n}} > P_{\max}$ **return** $\boldsymbol{x}$
6.   **else update** $\boldsymbol{x}$ with $x_{\hat{u},\hat{m},\hat{q},\hat{n}} = 1$
7.   **if** no more unsent NALU **return** $\boldsymbol{x}$

PRDO

1. let $\boldsymbol{x} = \{x_{u,m,q,n} = 0 \mid \forall u, m, q, n\}$
2. **forever**
3.   **if** $\boldsymbol{g}_d$ is empty **return** $\boldsymbol{x}$
4.   **let** $\boldsymbol{g}_d$ be all immediately decodable NALGs
5.   **for** $g_{u,m,q} \in \boldsymbol{g}_d$
6.   **for** $n = 1$ to N
7.   **compute** $b_{u,m,q,n}$ based on $\boldsymbol{x}$
8.   **let** $\frac{b_{\hat{u},\hat{m},\hat{q},\hat{n}}}{s_{\hat{u},\hat{m},\hat{q}}} \geq \frac{b_{u,m,q,n}}{s_{u,m,q}}$ $\forall u, m, q, n$
9.   **if** $b_{\hat{u},\hat{m},\hat{q},\hat{n}} \leq 0$ **return** $\boldsymbol{x}$
10.   **update** $\boldsymbol{x}$ with $x_{\hat{u},\hat{m},\hat{q},\hat{n}} = 1$, **update** $\boldsymbol{g}_d$.

Term-by-Term Convex Approximation

Goal: Obtain a convex superset of the constraint set

1. Term-by-term convex approximation (TTC)

$$x_{u,m,q} \leq \sum_{n=1}^{N} \min(1 - p_n, x_{u,m,q,n}),$$

$$e_{u,m} \geq \hat{\delta}_{u,m} + \sum_{q=0}^{Q} (1 - \min_{q' \leq q} x_{u,m,q'}) \delta_{u,m,q},$$

- **Polynomial number of constraints in U,M,Q,N**
- **Weak approximation of the probability of successful packet delivery $x_{u,m,q}$**

Multilinear Convex Approximation

Goal: Obtain a convex superset of the constraint set

2. Multilinear convex approximation (MC)

- Convex envelope of multilinear functions [Sherali '97]
  - Minimum of affine functions
- Tightest convex approximation
- Exponential number of constraints in Q,N
- Constraint on $x_{u,m,q}$ depends exclusively on N, NOT on problem parameters

Hybrid Convex Approximation

Goal: Obtain a convex superset of the constraint set

3. Hybrid Convex Approximation (HC)

- **Term-by-term approximation for truncation distortion $e_{u,m}$**
- **Multilinear approximation for probability of successful packet delivery $x_{u,m,q}$**

$$x_{u,m,q} \leq \sum_{n=1}^{N} \min(1 - p_n, x_{u,m,q,n}),$$

$$e_m \geq \hat{\delta}_m + \sum_{q=0}^{Q} \delta_{m,q} - \min \left\{ L_m^2(\xi, \bar{\mathbf{x}}) := \sum_{q=0}^{Q} \left(\sum_{q'=0}^{q-1} \prod_{i \leq q'} \xi(i) \delta_{m,q'} + \sum_{q'=q}^{Q} \prod_{i \leq q', i \neq q} \xi(i) x_{m,q} \delta_{m,q'} \right) - Q \sum_{q=0}^{Q} \prod_{i \leq q} \xi(i) \delta_{m,q} \right.$$

$$\text{s.t.} \quad \xi \in \{0,1\}^{Q+1}, L_m^2(\xi, \bar{\mathbf{x}}) \leq \sum_{q=0}^{Q} \bar{x}_{m,q} \delta_{m,q} \quad \forall \bar{\mathbf{x}} \in \{0,1\}^{Q+1} \Big\},$$

- **Polynomial complexity in U, M, Q, exponential in N**
- **Good trade-off of approximation accuracy vs. complexity for low N**

Solving the Convex Approximations

□ **Properties of our convex approximations**

- **Non-empty compact set of solutions**

- **Strong duality**

- **Non-empty set of dual optimal solutions**

□ **These properties are important for the performance of numerical methods [Boyd et al. 04']**

□ **We use CVX to solve our convex programs**

– **a convex program solvers based on Matlab**

– **developed at Stanford**

Simulation Setup

- **Scheduling period : M = 32**

- **Number of quality enhancement layers : Q=7**

- **Number of access networks : N=3**

- **Decoding deadline : t_0 = 1 sec**

- **SVC video streams: Crew, Harbour, City, and Soccer**

- **Trace-driven simulations (NS-2)**
  - **Data from subnets at Stanford University and DT Labs Berlin**
  - **Used Abing to measure end-to-end available bandwidth and round-trip time**
  - **Run 300 simulations for each setup**
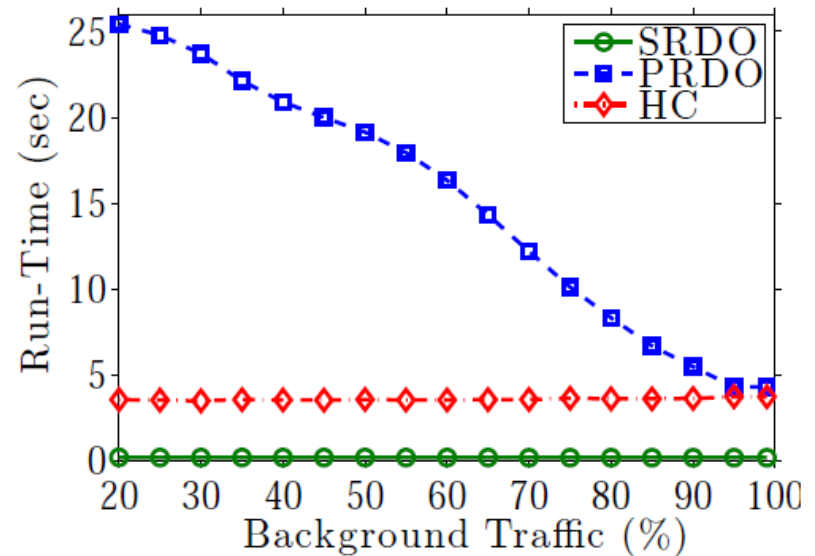
Comparison against Current Solutions
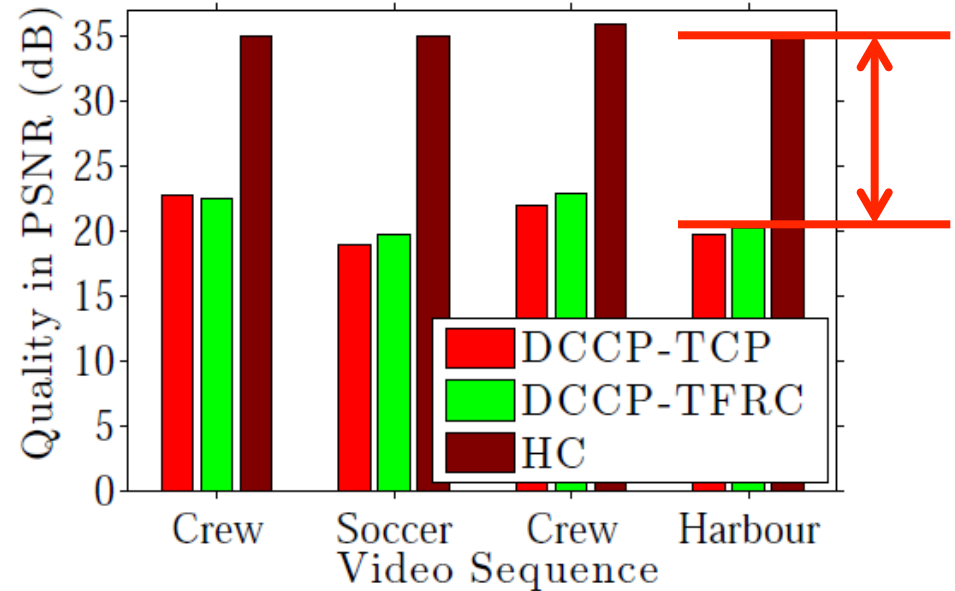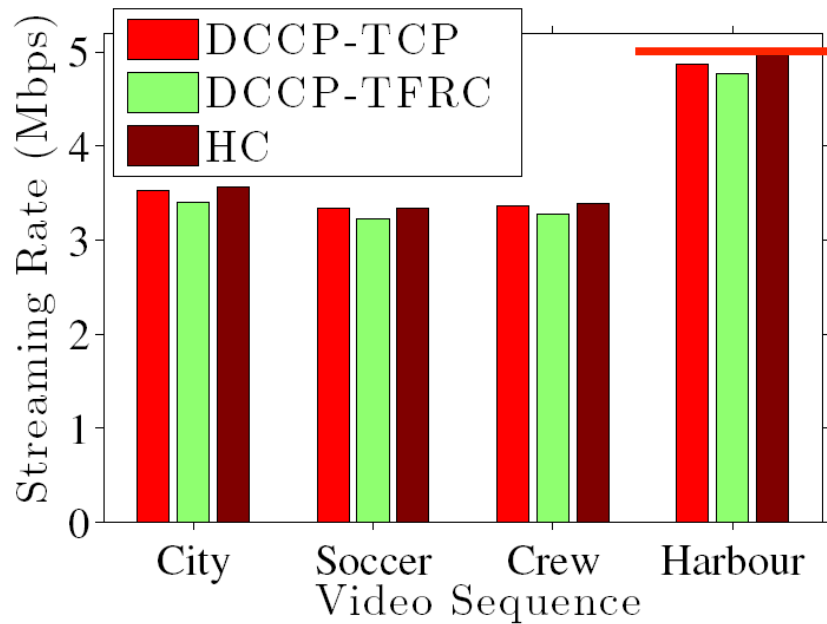
– **Proposed algorithms are TCP-Friendly**

– **Proposed algorithms constantly outperform current ones by more than 10 dB**
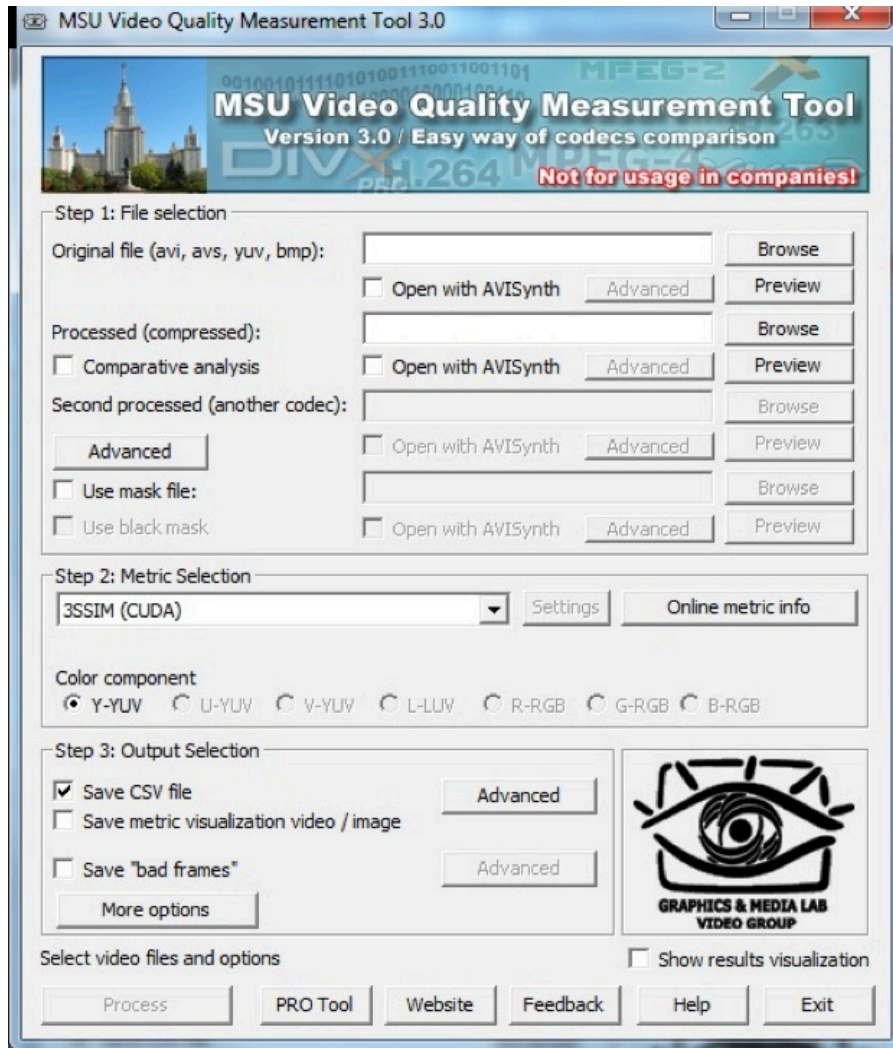
Complexity versus Performance
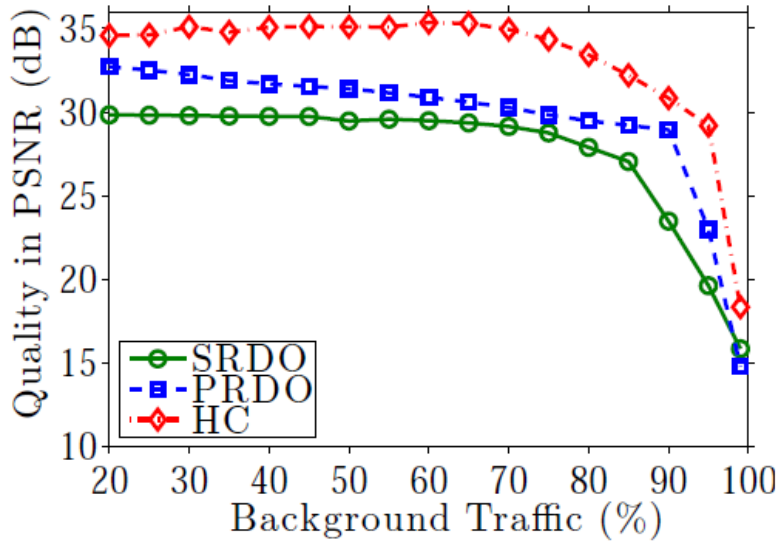
Convex solution outperforms heuristics in performance

Convex solution has a reasonable time complexity

Useful Tools

Tools to Compute Video Quality (1/2)

- **MSU Video Quality Measurement Tool**

  - **MSU Graphics and Media Lab, Moscow State University**

  - **Supports 20 quality metrics: including variations of PSNR, SSIM, and VQM (another popular metric we didn't discuss)**

  - **Supports 20 video file formats**

  - **Comes with academic/ commercial versions**

  - **http://graphics.cs.msu.ru/**

Tools to Compute Video Quality (2/2)

- **VQM (Video Quality Metric)**
  - A U.S. national standard (ANSI T1.801.03-2003), and an international ITU Recommendations (ITU-T J.144, and ITU-R BT.1683, in 2004)
  - Public tool available
  - **http://www.its.bldrdoc.gov/vqm/**

- **SSIM**
  - Matlab implementation at Prof. Wang's utility page
  - **https://ece.uwaterloo.ca/~z70wang/research/ssim/**

- **MSE/PSNR**
  - PSNRStatic comes with JSVM software
  - Write your own

Other Resources for Evaluation

- ## Video Traces
  - Arizona State: http://trace.eas.asu.edu/, long video sequences coded in SVC, AVC, MPEG-4, MPEG-2, and MDC coders
  - TU Berlin http://www.tkn.tu-berlin.de/research/trace/ltvt.html, long video sequences coded in MPEG-4 and H.263

- ## Video Sequences
  - Xiph Open-source Video Production http://media.xiph.org/, pointing to many other links for Raw video sequences

- ## Codecs
  - AVC Reference Coder http://iphome.hhi.de/suehring/tml/
  - SVC Reference Coder http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm
  - X264 Coder http://www.videolan.org/developers/x264.htm
  - Nokia's 3D Coder/Decoder http://research.nokia.com/research/mobile3D

Other Resources for Evaluation

- **Streaming Tools**
  - Darwin Open-source Version of QuickTime Server **http://dss.macosforge.org/**
  - VLS VideoLAN's Streaming Server **http://www.videolan.org/vlc/streaming.html**
  - VLC VideoLAN's Player **http://www.videolan.org/vlc/**
  - Live555 Streaming Library **http://www.live555.com/liveMedia/**

- **Misc**
  - Matlab Central's File Exchange **http://www.mathworks.com/matlabcentral/**

Summary

- **Quality metrics are means to quantify the performance of multimedia systems, and can be classified into**
  - Subject (tester) versus objective (program)
  - Full reference, no reference, and reduced reference

- **Quality metrics play central roles on optimizing multimedia system**
  - Simple metrics are preferred for good properties, e.g., convex/concave of MSE/PSNR, for efficient algorithms