Optimal Partitioning of Fine-Grained Scalable Video Streams

Mohamed Hefeeda
(joint work with Cheng-Hsin Hsu)

NOSSDAV 07
4 June 2007
Motivations: Internet Video Server

- Heterogeneous clients, even with same access technology
- \( \rightarrow \) Fine-Grained Scalable (FGS) Coding to cope with heterogeneity
**FGS Coding**

- MPEG-4 and H.264 standards support FGS

- Why FGS?
  - Utilization of server and client bandwidth
  - Efficient storage and customization of videos
FGS and Coding Inefficiency

- Coding (in)efficiency gap
  - FGS yields lower quality compared to nonscalable at same rate
- Base layer rate ($r_b$) controls this gap
  - Larger $r_b \Rightarrow$ smaller gaps
  - But, larger $r_b \Rightarrow$ disqualify clients with bandwidth $< r_b$
  - Trade-off that determines the quality for all clients

- Our Work (1)
  - Experimentally quantify and model quality gap between FGS and nonscalable streams
Our Work (2): Single FGS Sequence

- Find the best base layer rate for a single sequence to maximize quality for given client distribution

- Present optimal and efficient algorithm to solve it

- Useful when server pre-allocates bandwidth for individual sequences

- Also used as a step in the general problem
Our Work (3): Multiple FGS Sequences

- Find best base layer rates for multiple sequences concurrently streamed to diverse client sets to maximize quality for all clients, constrained by server bandwidth
Our Work (3): Multiple FGS Sequences

- We prove that it is NP-Complete

- Propose Branch-and-Bound algorithm that runs fast for many typical cases

- Propose Heuristic algorithm that produces near-optimal results and scales to large problems
Our Work in the Big Picture

- **Our Algorithms**
  - Characteristics of sequences
  - Video database
  - Camera

- **FGS Encoder or Transcoder**

- **Client bandwidth distribution**

- **Connections**:
  - Ethernet
  - Cable
  - DSL
  - Wireless
Quality Gap for FGS Streams

- **Instrument Reference Software of H.264**
  - Joint Scalable Video Model (JSVM ver 8.0)

- **Use several diverse video sequences**
  - Mobile, City, Harbour, Soccer, Crew (4CIF)

- **Encode with a given $r_b$**
  - decode at many bit rates
  - measure quality (PSNR) and compare to nonscalable

- **Repeat for several $r_b$ values**
Quality Gap for FGS Streams: Results

(a) Mobile

- Nonscalable
- FGS, $r_b=100$ kbps
- FGS, $r_b=750$ kbps

$\Delta(r_b)$

$\Delta(r_b)$
Quality Gap for FGS Streams: Results

(a) Mobile

(b) City
Quality Gap for FGS Streams: Results

- Gap is a decreasing function of $r_b$

- Smaller gaps for sequences with higher bit rates (4CIF)

- Gap is due to
  - Less accurate motion estimation, only base layer is used in estimation
  - Additional header overheads
Single-Sequence Formulation

- **Inputs**
  - Clients divided into $C$ classes, with bandwidth $b_1 < b_2 < \ldots < b_C$
  - Client distribution over classes: $f_c$
  - Quality at a given rate (e.g., R-D function): $q(r)$

- **Find $r_b$ such that**

$$\max_{r_b} \left( \sum_{c=1}^{C} q(b_c) f_c \right), \quad \text{where} \quad r_b \in [0, r_{\text{max}}]$$
Theorem 1

An optimal solution for the base layer rate that maximizes average perceived quality for all clients can be found at one of the rates $b_c$, where $1 \leq c \leq C$.

Using Theorem 1, we design a simple algorithm (FGSOPT) to solve the single-sequence problem in $O(C)$ steps.
Multiple-Sequence Formulation

- Generalize to $S$ sequences, each sequence has a client distribution
- Find base layer rates $R = \{r^s, 1 \leq s \leq S\}$ such that:

\[
\max_R \sum_{s=1}^{S} \left\{ \sum_{c=1}^{C^s} N^s q_c^s f_c^s \right\} \\
\text{s.t.} \quad 0 \leq R \leq R_{\text{max}}; \\
\sum_{s=1}^{S} \left\{ \sum_{c=1}^{C^s} N^s q_c^s f_c^s \right\} \leq B_{\text{max}}
\]
Theorem 2

Determining optimal base layer rates of multiple FGS sequences concurrently streamed by bandwidth-limited server is NP-Complete.

Proof

- By reducing the multiple-choice knapsack problem to the above problem
Multiple-Sequence: Branch & Bound Alg.

- **Idea of the B&B Algorithm (MFGSOPT)**
  - Incrementally construct a tree
  - Each level represents a sequence with its possible base layer rates (\(= C\) using Theorem 1)
  - Before expanding a branch use a BOUND function to compute an upper on the quality from that branch
  - The upper bound results in pruning many branches without sacrificing the optimal quality
Multiple-Sequence: B&B Algorithm

\[
\begin{align*}
\text{0} &: <-, -, -, -|0/0 > \\
\text{1} &: < b_1^1, -, -, -|3/2 > \\
\text{2} &: < b_2^1, -, -, -|2/1 > \\
\text{3} &: < b_3^1, -, -, -|4/3 > \\
\text{4} &: < b_3^1, b_1^2, -, -|8/9 > \\
\text{5} &: < b_3^1, b_2^2, -, -|10/7 > \\
\text{6} &: < b_3^2, b_1^2, b_1^1, -|13/8 > \\
\text{7} &: < b_3^1, b_2^2, b_2^3, -|15/10 > \\
\text{8} &: < b_3^1, b_2^2, b_3^3, -|12/9 > \\
\end{align*}
\]
Multiple-Sequence: Heuristic Algorithm

- **Idea of the Heuristic Algorithm (MFGS)**
  - Incrementally allocate more bandwidth to sequences that are expected to increase quality by higher margins for each bandwidth unit consumed
Evaluation

- **Setup**
  - H.264 reference software
  - Several video sequences
  - Different client distributions
  - Various typical streaming scenarios
Quality Improvement

- Up to several dB quality improvement, on average
B&B vs. Heuristic

- Heuristic algorithm produces near-optimal solutions
B&B vs. Heuristic

- Heuristic algorithm is much faster and scales with number of sequences
Conclusions

- Modelled the quality gap between FGS and nonscalable streams
  - Trade off between supported clients ranges and perceived quality

- Optimization problem to find best base layer rate:
  - Single sequence (optimal and efficient algorithm)
  - Multiple sequences (NP-Complete, B&B, Heuristic)

- Systematic algorithms to optimize quality
  - Compared to rule-of-thumb methods
Thank You!

Questions??

- Details are available in the extended version of the paper at:

  http://www.cs.sfu.ca/~mhefeeda