A Game Theoretic Approach to Low Energy Wireless Video Streaming

Ali Iranli, Kihwan Choi, and Massoud Pedram

Dept. of EE-Systems, Univ. of Southern California, Los Angeles, CA 90089

Email: {iranli, kihwanch, pedram} @usc.edu

Abstract: *This paper presents a dynamic energy management policy for a wireless video streaming system, consisting of battery-powered client and server. The paper starts from the observation that the video quality in wireless streaming is a function of three factors: encoding aptitude of the server, decoding aptitude of the client, and the wireless channel. Based on this observation, the energy consumption of a wireless video streaming system is modeled and analyzed. Using the proposed model, the optimal energy assignment to each video frame is done such that the maximum system lifetime is achieved while satisfying a given minimum video quality requirement. Experimental results show that the proposed policy increases the system lifetime by 20%.*

1   Introduction

With the availability of mobile, communication and computing systems, we have seen an explosive growth in wireless multimedia applications, e.g., streaming audio and video. This trend, in turn, poses two challenges: (1) establishing and maintaining a stable channel for real-time operation and (2) power-aware operation so as to increase the lifetime of the battery-powered wireless system while meeting a minimum quality of service (QoS) requirement. Furthermore, it is desirable to provide a mechanism for graceful degradation in QoS so that a dynamic power manager (DPM) can incrementally trade off QoS for higher energy efficiency. Fine Granularity Scalability (FGS) coding technique [1], which was adopted as the standard in MPEG-4, provides an effective mechanism for graceful video quality degradation based on its hierarchical layer structure, which consists of a base layer and one or more (optional) enhancement layers. Although extensive studies have been conducted on the hierarchical layer structure of MPEG-4 and its error resiliency under fluctuations in the channel bandwidth [2][3][4], energy efficiency in a battery-powered server-client system has received little attention.

2   Lifetime Optimization Problem

The encoding/decoding aptitude of an image-processing core is a strong function of its operating frequency and voltage level. Thus, one can characterize the video quality VQ of frame j as: $VQ_i = f(e^s_j, e^c_j, \omega_j)$ where e^s_j and e^c_j denote the server and the client energy consumptions for frame j while ω_j denotes the wireless channel conditions for transmission of frame j.

We consider a wireless system operating over a fading channel. Time is assumed to be discrete. Each frame is processed in one timeslot of duration τ, where τ is the inverse of the frame rate. In each timeslot, the channel state changes among different states chosen from a finite set $\Omega = \{\omega_1, \omega_2, \cdots, \omega_n\}$, according to a probabilistic model. The server and the client are assumed to be battery-powered, each with a fixed number of available energy units. Each channel state ω_j determines the throughput that can be achieved per unit energy expended by the server/client. The video processing cores can operate with frequencies f_S and f_C in a range bounded by a lower bound f_{min} and an upper bound f_{max}.

The problem at hand is to maximize the lifetime of the system, given the remaining energy levels of the server (E^S_0) and the client

(E^C_0). In other words, the objective is to find an energy allocation pair (e^S_j and e^C_j) for each timeslot j of duration τ so as to maximize the overall *system lifetime*, Λ, (which is an integer multiple of τ) subject to: 1) $\forall j: \dfrac{e^s_j}{\tau} \le p^s_{max}$ *and* $\dfrac{e^c_j}{\tau} \le p^c_{max}$; 2) $avg(VQ_j) \ge VQ_{min}$;

$$3) \quad \sum_{j=1}^{\Lambda} e^s_j \le E^S_0 \ \ and \ \ \sum_{j=1}^{\Lambda} e^c_j \le E^C_0 \qquad\qquad (1)$$

where $avg(VQ_j)$ denotes the average video quality over the system lifetime. Constraint (I) signifies the fact that power consumption of the server and client is upper-bounded due to limitations on maximum current output of their battery sources. Constraint (II) guarantees the average video quality of the system whereas constraint (III) corresponds to the total energy bound. In the remainder of this paper, subscript j is used to indicate the time slot j in which the corresponding parameters are evaluated.

The aforementioned policy optimization problem, which involves a hierarchical variable determination process, is a form of multi-level optimization problem known as *Stackelberg game* [5].

3   A Game-theoretic Formulation

In our context, Strategy x for the client is the adoption of a specific vector of truncation lengths (*TL's* denoted by a_i's) for the sub-carriers and an operating frequency for the decoding image processing core, f^C, and therefore, set of strategies for client X is:

$$X = \Big\{(a_1, a_2, \cdots, a_n, f^C) \Big| \forall i : a_i \in TLS, f^C \in FS\Big\}, \quad \text{where} \quad n \text{ is the}$$

number of sub-carriers in the Orthogonal Frequency Division Multiplexing (OFDM) signal, *TLS* denotes the set of all (feasible) *TL*'s for the adaptive Viterbi decoder, and *FS* is the set of feasible frequencies for the image processing core. Strategy y for the transmitter is a choice of specific overall transmission power level, p_{amp}, a set of modulation levels for the different sub-carriers, b_i's, and operating frequency for the encoding image processing core, f^s. Therefore, set of possible strategies for the server Y would be:

$$Y = \Big\{(p_{amp}, b_1, b_2, \cdots, b_n, f^S) \Big| p_{amp} \in PLS, \forall i : b_i \in MLS, f^S \in FS\Big\}.$$

where *MLS* and *PLS* denotes the sets of (feasible) modulation levels for each sub-carrier and available power levels for signal transmission. These sets are known from chipset specification or the standard protocol supported by the chipset.

The overall objective of the client-server game is to ensure that we achieve an acceptable level of performance while maximizing the overall video service time. For solving this optimization problem, the server and the client take turns at the beginning and end of each timeslot. The server's goal is to minimize the overall energy consumption of the client-server system whereas the client's objective is to make sure that it will not exhaust its energy source any sooner than the server does. In this way, this two-player game results in extending the overall system lifetime, first by minimizing the energy consumption and then by ensuring that no player dies earlier than the other. Since the power consumption of the client can easily be derived the client actually solves the following equivalent optimization problem;
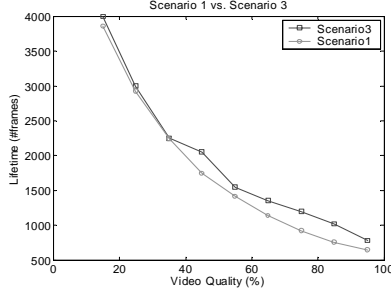
Figure 1. Lifetime comparison between Scenario 1 and 3

$$\arg\min_{\hat{X}} \left\{ \left| \frac{E_j^C}{\Lambda_{j-1}^S} - \frac{e_j^C}{\tau} \right| : \Gamma\hat{X} + \Phi\hat{Y} \le R\hat{E}Q^C, \hat{X} \in TLS^n \times FS \right\} \quad (2)$$

The ratio of E_j^C to Λ_{j-1}^S (i.e., the server's lifetime as reported in timeslot j-1) signifies the power dissipation target for the client in order for it to survive until the end of the server's expected lifetime. e_j^C denotes the power consumption (i.e., per-frame energy consumption) of the client, which is calculated from the *TL's* and the f^C value by equation: $e_j^C = <\rho, \hat{X}> + \varepsilon^C$ where ρ^T is a constant coefficient vector and $<a,b>$ is used to represent the inner product of vectors a and b. The client's objective in this optimization step is to find $\hat{X}^*$ such that its actual power consumption rate becomes as close as possible to its target power dissipation. $\Gamma\hat{X} + \Phi\hat{Y} \le R\hat{E}Q^C$ represents the matrix-vector form of inequality constraints (1). Here, Γ and Φ denote the coefficient matrices that account for the channel conditions and per-frame energy consumptions of the basic building blocks of the client. $R\hat{E}Q^C$ is a vector consisting of an upper bound on power consumption, p_{max}^C, and the required *BER* value for all sub-carriers.

The server (leader), on the other hand, attempts to minimize the overall energy consumption of the client-server system, given the channel conditions provided by the client, hence, $f_L(x,y) = e_j^C + e_j^S$. The optimization problem for the server can be mathematically formulated as:

$$\arg\min_{\hat{Y}} \left\{ \langle\rho,\hat{X}\rangle + \langle\vartheta,\hat{Y}\rangle : \Xi\hat{Y} \le R\hat{E}Q^S, \hat{Y} \in PLS \times MLS^n \times FS, \hat{X} \in \Psi_L(\hat{Y}) \right\} \quad (3)$$

Similar to the client case, writing the constraints in matrix-vector form would result in the coefficient matrix Ξ for linear estimation of the p_{max}^S, *VQ*, and BER in terms of the SNR and the modulation level. Here ϑ is a constant coefficient vector. $<\vartheta,\hat{Y}>$ signifies power consumption (i.e., per-frame energy consumption, e_j^S) of the server. $R\hat{E}Q^S$ is a vector representing maximum power consumption p_{max}^S and minimum requirements for *VQ* and BER.

4   Experimental Results

We implemented an MPEG-4 FGS streaming system on a high performance testbed. The hardware used was the Intel's Xscale processor, which supports 9 different frequencies from 200MHz to 733MHz. In order to simulate the system, we used the Simulink 5.0 environment from Matlab 6.5 Release 13. The characteristics of this channel were simulated and recorded for duration of 4800 frames, i.e., five minutes @15 frames/second. To show the effectiveness of our approach, three different scenarios were simulated and compared. In scenario number one, for each timeslot i, after the detection of channel conditions, we assigned enough energy to the server and client to support the specified average video quality. In scenario two, the average channel condition was used to determine the required energy in each timeslot. Finally, in scenario three our adaptive algorithm was used to calculate the required energy for each timeslot.

Figure 1 shows the lifetime versus required average video quality graphs for scenarios 1 and 3. Total initial energy for this experiment is set to 600 J. According to this graph, our approach increases the system lifetime by as much as 20% for higher values of the required video quality. On average, the proposed method increases the system lifetime by more than 15% over the whole range of video quality.

Figure 2 shows the comparison between scenarios 2 and 3. Since in scenario 2, the assigned energy to each frame is fixed and is selected according to the average channel behavior, the system consumes extra energy to produce higher video quality, and hence, has a shorter system lifetime. Figure 2(a) shows a comparison between the system lifetimes for scenarios 2 and 3. It is clear that the system lifetime is significantly increased for scenario 3 where we employed the proposed dynamic policy approach. Notice that the average video quality was maintained above the required value. However, for scenario 2, the average video quality is unnecessarily improved, which would be OK if there was no energy dissipation overhead (cf. Figure 2(b)).

5   Conclusion

An adaptive approach to the energy assignment of each frame is proposed. The proposed approach guarantees the minimum video quality for all frames. Simulations show the effectiveness of the proposed approach. Based on this approach the overall system lifetime is increased by 20%.

References

[1]   W. Li, "Overview of Fine Granularity Scalability in MPEG-4 Video Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.11, Mar. 2001.

[2]   M. van der Schaar, et al., "Scalable MPEG-4 Video Coding with Graceful Packet-loss Resilience over Bandwidth-varying Networks," *Proc. of the ICME*, vol.3, pp.1487-1490, 2000.

[3]   R. Cohen and H. Radha, "Streaming Fine-Grained Scalable Video over Packet-Based Networks," *Proc. of the GLOBECOM*, pp.288-292, 2000.

[4]   R. Yan, F. Wu, S. Li, and R. Tao, "Error resilience methods for FGS video enhancement bitstream," *The First IEEE Pacific-Rim Conference on Multimedia*, Dec. 13-15, 2000 Sydney, Australia.

[5]   S. Dempe, Foundations of Bilevel Programming, *Kluwer Academic Publishers*, Boston, 2002.
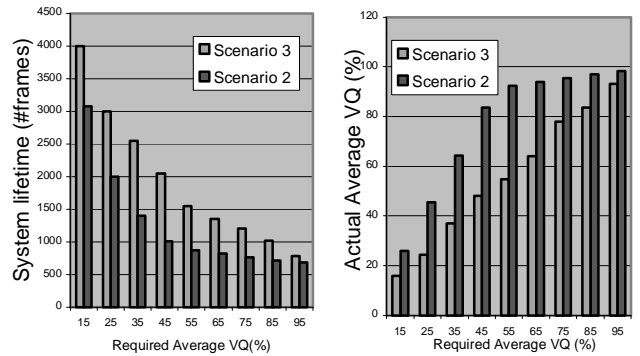
(a) System Lifetime   **(b) Video quality**

Figure 2. Experimental results