Cross-Layer Co-Exploration of Exploiting Error Resilience for Video Over Wireless Applications

Amin Khajeh, Minyoung Kim, Nikil Dutt, Ahmed M. Eltawil, Fadi J. Kurdahi

University of California, Irvine, CA 92697, USA

{akhajehd, minyounk, dutt, aeltawil, kurdahi}@uci.edu

ABSTRACT

In this paper, we propose a cross-layer algorithm/architecture co-exploration for wireless multimedia systems to coordinate interactions among sub-layer optimizers for improvements in energy/QoS/reliability. By exploiting the inherent redundancy in wireless multimedia systems, we generate an expanded design space over traditional, layer-specific approaches. Specifically, we control the error resilient encoder at the application layer to provide awareness of architectural exploration at the physical layer (WCDMA) allowing new design points with lower power consumption via aggressive voltage scaling. While trying to reduce energy consumption, the fault tolerant technique compensates the effect of the hardware and network errors due to aggressive voltage scaling and lossy transmission, respectively. Our experiments on H.263 video over WCDMA communication system demonstrate that co-exploration enlarges the feasible design space which results in significant power savings of 34% in the H.263 encoder more than 20% in the WCDMA modem.

1. INTRODUCTION

Mobile multimedia applications on battery-powered devices running typically operate under harsh wireless conditions, requiring good strategies for energy/QoS provisioning. This requires resource management policies at different layers including: a specific video encoding/decoding algorithm at the application layer; data transfer protocol and network monitoring at the middleware layer; Fault Tolerant technique (FT) at the physical layer. Whereas traditional approaches considered policies locally at each abstracted layer, our research expands the design space for energy/QoS by investigating opportunities for cross-layer co-exploration. As shown in Figure 1, without loss of generality, we consider a case study of a hypothetical wireless video communication platform. At the application layer, we use PBPAIR [1] energy-efficient error-resilient encoding. At the middleware layer, selective protection [2] and UDP-Lite bit-error resilient packetization [3] is utilized. Lastly, we explore aggressive dynamic voltage scaling (AVS) on WCDMA memory [4, 5] at the physical layer.

Policies made at one layer can affect behavior at other layers. For example, if AVS is used at the physical layer (WCDMA), errors gradually start to appear in the hardware. Specifically, embedded memory is typically affected before logic. These errors will then propagate through the system and manifest themselves at the application layer, as illustrated in Figure 2 that shows the impact of the WCDMA physical errors (due to voltage scaling) on the quality of the video (peak signal to noise ratio, PSNR) for a sample video sequence. Each line demonstrates the effect of PBPAIR with selective protection (application/middleware layer). Class-1 and Class-2 protection ensures bit-error free transmission except TCOEF and Inter-TCOEF of H.263 payload [6], respectively. From the sample video sequence, we observe that data for Class-1 and Class-2 protection takes up to 25% and 80% of the H.263 encoded stream, respectively. To guarantee bit-error free transmission, UDP-Lite packetization protects part of datagram where WCDMA cannot perform AVS on that part of datagram.

For this example, Figure 2 shows that WCDMA bit error rate (BER) less than $10^{-5}$ is acceptable regardless of other layers' policy. On the other hand, WCDMA BER greater than $10^{-4}$ results in undesirable quality of service with differential impact due to selective protection. It should be pointed out that the above results are generated from the straightforward composition of existing individual techniques. There still remain opportunities for a larger design space such as increasing PSNR by manipulating algorithmic parameters of upper layer to accommodate physical layer’s characteristic. For instance, we can manipulate PBPAIR parameter to consider the BER, which in turn leads to acceptable PSNR when BER is $10^{-4}$.

There have been several studies on cross-layer optimizations for multimedia [7, 8, 9, 10]. In [7, 8], the authors have...
identified interactions between the different layers followed by coordinated optimization. They specifically explore middleware and OS scheduling policies to control the overall energy consumption without architectural exploration, whereas we specifically study cross-layer co-exploration. The authors of [9, 10] study the issues of cross-layer optimization as a new paradigm for network architecture to make better use of network resources. Those efforts are, however, mainly focused on the architectural decisions in networking, not tuning the system parameters for QoS-energy optimization.

The main contribution of our work is algorithm/architecture co-exploration using cross-layer optimizations to develop a reconfigurable video coding and processing system. The key idea underlying the co-exploration is that we manipulate the parameters of the reconfigurable video coding to provide awareness of architectural characteristics at the physical layer. This paper contains the following specific contributions: Comprehensive analysis of cross-layer QoS-energy tradeoffs and coordinated interaction enable us to tune video coding parameters on highly resource limited devices. In our chosen H.263 video communication application and WCDMA simulation environment, our approach enlarges the feasible design space by exploiting error resiliency due to the redundant nature of video coding.

2. SUB-LAYER POWER-AWARE TECHNIQUES

We now illustrate sub-layer power-aware techniques for the three layers shown in Figure 1.

2.1. Physical Layer

We consider WCDMA at the physical layer. The authors in [4, 5] have shown that in some systems such as wireless and multimedia, the computational engine does not need to be 100% correct, 100% of the time. In fact, an insightful view of wireless systems shows that these systems, by design, can accommodate a large degree of channel induced errors while meeting the stated system requirements such as target Bit Error Rate (BER). Therefore, by merging the system design with the circuit design, a whole new design space can be explored where controlled hardware errors can be treated in a similar manner to channel errors, thus contributing to the noise floor while still meeting stated system metrics. In this approach, the key idea is to intentionally vary the operating conditions to a point where errors start to occur and to exploit the now extended design space to optimize other aspects of the design, namely power consumption across different layers.

It is a well known fact that supply voltage reduction is the most effective means of reducing power consumption. However, in traditional voltage scaling techniques embedded memories are the bottleneck and typically the power controller unit reduces both voltage and frequency simultaneously in order to maintain the correct functionality of the embedded memory (in general embedded memories fail faster than logic under aggressive voltage scaling).

In the proposed Aggressive Voltage Scaling (AVS) technique, we separate the error-resilient memories (ERM) from the rest of the system. ERMs are memories that store data that is inherently redundant such as raw data buffering memories in the modem. In general, ERMs consume a large portion of the design area and they store raw soft bit values that have multiple levels of redundancy. At the algorithmic level, coding redundancy exists to protect the data against the communication noise. Also most of the time, the receiver experiences a relatively higher SNR than the minimum required for the modulation. Therefore, to achieve more power savings we use this redundancy and slack to allow some limited and controllable errors to occur in hardware.

In the AVS technique, the error-resilient memories are supplied from a variable source while the rest of the system is supplied in a traditional manner (i.e., nominal supply). Dur-
ing operation, the voltage to the ERMs is aggressively lowered and errors are allowed to occur in the system, that will be corrected at the application layer. In this paper, we explore such an approach to a cross-layer system that merges both the modem and the source coding in one system. A variable supply (could be on chip or off-chip) supplies the ERMs while a standard supply provides the rest of the modem with nominal \( V_{dd} \). The authors are aware that this is one example, and that the techniques of error resilience as well as the power saving are design dependent. However, the main point of the paper is that it is possible to utilize the knowledge of the requirements of the application using the modem to reduce power.

Figure 3 shows the block diagram of the receiver which utilizes the AVS technique. The received signal goes into the WCDMA receiver which is equipped with an AVS controller. Based on the operating condition (i.e., bit rate and the received signal’s SNR and etc.), the controller decides on the proper voltage for the ERMs. It is very important to quantify the effect of aggressive voltage scaling on the ERMs’ functionality. To do so, we set up a HSpice simulation on 6-Transistor Static random access memory (SRAM) in 65nm technology using Predictive Technology Model (PTM) [11] and statistically calculated the probability of failure (Read Access Failure, Write Failure and Destructive Read Failure) for voltages lower than nominal \( (V_{dd}\text{nominal} = 0.9\text{v}) \) [4]. The graph under the Error-Resilient-Memories in Figure 3 shows how the ERMs’ behave under AVS. As the result of process variation, the errors are gradually increasing in log-domain with the voltage reduction.

2.2. Network Layer

The network layer attempts to monitor the current network status and control the data transmission based on the network status. The impact of transmission errors on video quality depends on the spatial and temporal location of the error. For instance, errors in packet headers or motion vectors can cause drastic video quality degradation. A single bit error can damage a major part of a frame due to resynchronization of variable length codes. For the errors propagated among consecutive P-frames, intra coding (that we will discuss in the following subsection) can stop error propagation in temporal domain.

In this context, protecting the entire bitstream is expensive because of bandwidth limitations and delay constraints. Selective protection on the most critical information [2] combined with bit-error resilient packetization scheme for UDP-Lite [3] can be effectively used for this purpose. In this particular work, we assume selective protection and UDP-Lite implementation are available and only focus on the co-exploration of the other two layers (application and physical layer).

2.3. Application Layer

We consider PBPAIR (Probability Based Power Aware Intra Refresh) [1] as an application layer technique. The PBPAIR scheme inserts intra-coding (i.e., coding without reference to any other frame) to enhance the robustness of the encoded bitstream at the cost of compression efficiency. Intra-coding improves error resilience, but it also contributes to reducing encoding energy consumption since it does not require motion estimation (which is the most power consuming operation in a predictive video compression algorithm). In particular, PBPAIR utilizes partial intra-coding. Partial intra-coding attempts to alleviate the error propagation by using intra-coded macro block (MB) — the burden of refreshing is distributed, thereby producing a much smoother output rate — as can be seen in Figure 4.

Fig. 4. Partial Intra-coding

The remaining issue is how we pick the most appropriate MBs to maximize error resiliency while reducing energy consumption. For this purpose, the PBPAIR scheme takes two parameters i) the user’s QoS expectation (Intra Threshold: IntraTh), and ii) the network packet loss rate \( (\text{PLR}: \alpha) \), and raw video sequences as inputs to generate a bitstream robustly encoded against network transmission errors. Figure 5 illustrates simplified version of PBPAIR algorithm. PBPAIR re-evaluates the probability of correctness of each MB (as illustrated as Encoding Mode Selection in Figure 5) to decide encoding mode (I: Intra-coded MB, P: Predictively-coded MB) and to find motion vector \( (ME: \text{Motion Estimation}) \) in Figure 5). The encoding mode selection is done by comparison between probability of correctness of a MB \( (\sigma) \) and a threshold value \( (\text{IntraTh}) \). A MB with lower \( \sigma \) than \( \text{IntraTh} \) should be encoded as intra MB (refresh) since that particular MB has already experienced a sufficient amount of inter-coding up to that point and \( \text{IntraTh} \) values can be considered as requested error resiliency level. For a MB that is determined to be encoded as a inter-coded macro block, motion estimation based on a heuristic that considers both network condition and image content is performed.
It should be pointed out that the parameters can be easily manipulated to cope with other layers’ operating condition. For example, PBPAIR increases intra-coding by lowering the IntraTh parameter when there is high network packet loss (monitored at network layer). Indeed, PBPAIR controls the coding efficiency, error resiliency, and power consumption by tuning these parameters to achieve cross-layer co-exploration. We explain these details in our approach (Section 3).

3. CO-EXPLORATION

In this section, we discuss how to coordinate the individual techniques (physical layer and application layer) in a cross-layer manner based on the operating condition. Figure 6 shows the overall system model as an exemplar for our cross-layer co-exploration approach. As mentioned in the previous section, we assume a perfect packetizer e.g., UDP-Lite [3] for selective protection. To do so we need to:

- quantify the effect of memory errors at different bit rate on WCDMA bit error rate (BER), and
- explore the methods of taking the WCDMA BER into account and be compensating for at the application layer.

In the WCDMA system, the data buffering and de-interleaving memories (Error Resilient Memories) consume approximately 50% of the overall memory required for the entire modem. Furthermore, two instances of the ERMs alone account for 45% of the total on-chip memory power. These two instances are used to buffer the received data right after RAKE combining and to store the de-interleaved symbols prior to rate matching and processing by the decoder. In our simulation, we used the memory model error that we described in Section 2.1 for these two memories and simulated the WCDMA end-to-end physical layer for different bit rates. Figure 7 shows the effect of the memory errors on the WCDMA BER for different transmission bit rates. As expected, given the same SNR, for higher bit rates, the memory errors have higher impact on the BER since there is less redundancy in the system.

Once we quantified the effect of ERM errors on WCDMA bit error rate, We need to pass this information to the application layer in order to adapt to it. First, we attempt simple composition. As shown in Figure 6, the sender encodes raw video (application layer) then packetizes (network layer) and transmits encoded bitstream via WCDMA (physical layer) to the network. At the application layer, PBPAIR generates a robust bitstream against monitored network packet loss rate, $\alpha$. At the physical layer, WCDMA induces bit errors in the memory at the receiver side to reduce power consumption via voltage scaling. We define this impact as $\beta$. (i.e., $\beta = 1 - (1 - \text{ber})^n$, where ber and $n$ represent bit error rate and number of bits, respectively.) Under the condition of $\alpha$ network packet loss rate and $\beta$ WCDMA induced error, the probability of error in the path (encoding $\rightarrow$ transmission) can be calculated as following:

$$\alpha' = 1 - (1 - \alpha) \times (1 - \beta)$$

As shown in the Application Layer of the Sender in Figure 6, in this simple composition, we feed this adjusted error $\alpha'$ to the original PBPAIR instead of $\alpha$ to generate bitstream against network condition ($\alpha$) as well as physical layer error impact ($\beta$). Interestingly, this leads to an unstable system since the impact of bit error in WCDMA memory, $\beta$, will continuously increase due to the larger encoded bitstream (i.e., $\beta = 1 - (1 - \text{ber})^n$ with larger $n$) by inserting intra-coding to refresh the errors. This effect highlights the need for cross layer awareness to avoid such an undesired outcome.

Alternatively, a successful approach is to attempt to maintain $\beta$ at the similar level by manipulating the quantization value of encoding. Recall that PBPAIR encodes a macro block with intra-coding if it has lower $\sigma$ (probability of correctness) than IntraTh (a given threshold). The $\sigma$ value exponentially decreases by the probability of correct transmission with $\alpha$ error (i.e., $1 - \alpha$). In other words, the PBPAIR encoding mode selection is approximately based on the following inequality:

$$\sigma \approx (1 - \alpha)^{N_\alpha} < \text{IntraTh}$$

A simple calculation leads to

$$N_\alpha = \left\lceil \frac{\ln(\text{IntraTh})}{\ln(1 - \alpha)} \right\rceil$$

where $N_\alpha$ indicates the refresh period.

Since the length of bitstream is inversely proportional to the refresh period, we can calculate the ratio of encoded size corresponding to the error.

$$\frac{\text{EncodedSize}_{\alpha'}}{\text{EncodedSize}_\alpha} \approx \frac{N_\alpha}{N_{\alpha'}} \approx \frac{\ln(1 - \alpha')}{\ln(1 - \alpha)} = 1 + \frac{\ln(1 - \beta)}{\ln(1 - \alpha)}$$

On the other hand, we observe that a quantization value ($Q\text{value}_\alpha$) and encoded size has power relation according to the regression on profiled video stream like below:

$$\text{EncodedSize}_\alpha = c_1 \times Q\text{value}_\alpha^{c_2}$$

where $c_1$ and $c_2$ are constants depending on the video input characteristic.

Therefore, given $\alpha$, $\beta$, and $Q\text{value}_\alpha$, we can figure out the new quantization value $Q\text{value}_{\alpha'}$ to maintain $\beta$ at an appropriate level.

$$\frac{N_\alpha}{N_{\alpha'}} \approx \frac{\text{EncodedSize}_{\alpha'}}{\text{EncodedSize}_\alpha} = \left(\frac{Q\text{value}_{\alpha'}}{Q\text{value}_\alpha}\right)^{c_2}$$

$$\therefore, Q\text{value}_{\alpha'} = \left[Q\text{value}_\alpha \times \left(\frac{N_\alpha}{N_{\alpha'}}\right)^{\frac{1}{c_2}}\right]$$

$^1$The exact equation in [1] also takes into account the similarity factor between consecutive frames. However, we do not need that level of accuracy in this particular work.

$^2$Finding the constants $c_1$ and $c_2$ is out of scope of this paper. We estimate $c_1$ and $c_2$ for the test video sequences, and the result is consistent with our previous work [7].
When $Q_{value_{\alpha'}}$ is out of bound (i.e., larger than 31 in case of PBPAIR implementation), we consider that as an infeasible design point.

In summary, we considered cross-(Application and Physical)-layer co-exploration by exposing physical errors in the WCDMA up to the PBPAIR application layer. Specifically, we provide PBPAIR with $\alpha'$ and $Q_{value_{\alpha'}}$ to consider the network packet loss ($\alpha$) and the impact of WCDMA error ($\beta$) induced by AVS in a cooperative manner. In this section, we illustrated that individual optimization techniques (such as PBPAIR and AVS) or simple composition without considering other layers’ characteristic may result in drastic quality degradation. In the following section, we will discuss the benefit of power saving using this co-exploration.

### 4. EXPERIMENTS

We use a PBPAIR implementation from our previous work [1], and feed the new $\alpha'$ instead of the PLR ($\alpha$) of the original PBPAIR. We assume that a simple copy scheme is used for error concealment at the decoding side. Note that we use a uniform distribution of frame discard to generate the packet loss pattern. For simplicity, but without loss of generality, we use the frame loss rate to denote the network packet loss rate. For the encoding/decoding power consumption, we utilize the Simics [12] full system simulation platform, capable of simulating target systems that include real network connection and run operating systems and workloads. Specifically, we use the Simics model of a PowerPC-based Ebony card [13] that boots Linux 2.4 to estimate energy consumption.

Figure 8 shows the energy savings on the simulation platform introduced in [5]. For the power savings calculations we used 65nm technology. The x-axis represents the errors in the data buffering memories of the WCDMA receiver. From Figure 3 one can relate the memory supply voltage to the memory errors. Based on the value of the memory supply voltage, dynamic and leakage power of the memory can be calculated. Class-1 and Class-2 protection corresponds to 25% and 80% of the memory supplied with high voltage (nominal voltage) to guarantee the bit-error free operation, respectively. As shown in Figure 8, more protection leads to less power savings.

Next, we present experimental results that illustrate the benefit of cross-layer co-exploration. Figure 9 shows the video quality and energy consumptions of our co-exploration. It should be pointed out that as mentioned in Section 1, the feasible design space for video streaming is the case when $\beta$ is less than $10^{-4}$ (Figure 2), which corresponds to the bottom 5 design points in Figure 7 for the 64Kbps and 5dB SNR. However, by applying the proposed cross layer approach we can expand the design space to accommodate physical layer errors up to $10^{-2}$ with minimal video degradation as shown in Figure 9. This expanded design space utilizes the error resilient nature of enhanced PBPAIR encoding with $\alpha'$ parameter. Note that in this situation, we achieve more that 20% energy savings from WCDMA (Figure 8) without any quality...
loss. To further expand the design space to accommodate higher physical layer error rates ($10^{-1}$ WCDMA memory error) PBPAIR with selective protection can be exploited where protection of Class-2 is assumed. As shown in Figure 9(a), the PSNR is improved at a cost of higher WCDMA memory energy (22% less energy reduction comparing with No protection as shown in Figure 8). Decoding energy consumption will also increase by 3% as compared with No protection case.

Based on these results at extremely high physical error rates ($10^{-1}$ WCDMA memory error) the power management protocol can select the following design points: (i) Class-2 protection, (ii) No protection which results in 4% and 9% total energy reduction at the cost of 1dB and 4dB PSNR degradation, respectively. Note that any power savings achieved at the physical layer due to the proposed cross layer approach is in addition to the 17% to 34% encoding energy reduction achieved at the application layer (as compared with other error-resilient techniques) as documented in [1].

The novel aspect of our approach is that instead of optimizing each layer by itself, we try to reduce the power consumption of the overall system by passing on necessary information between different layers and finding the optimum operating parameters for the total system while meeting the required metrics.

5. CONCLUSION AND FUTURE WORK

In this paper, we discussed the impact of algorithm/architecture co-exploration in the context of transmitting H.263 video over a wireless WCDMA modem. It is shown that cross-layer co-exploration by targeting data buffering memories in the WCDMA system and adaptively changing the data rate based on the network information at the application layer, results in a considerable reduction in power consumption of 34% in the H.263 encoder more than 20% in the WCDMA modem. The proposed approach expands the feasible design space allowing designers to explore larger tradeoffs during system design.

3Encoding energy consumption does not depend on selective protection.

6. REFERENCES


