# Error-Exploiting Video Encoder to Extend Energy/QoS Tradeoffs for Mobile Embedded Systems

Kyoungwoo Lee, Minyoung Kim, Nikil Dutt, and Nalini Venkatasubramanian

**Abstract** Energy/QoS provisioning is a challenging task for video applications in power-constrained mobile embedded systems. Many error-resilient video encodings allow us to exploit errors and generate a range of acceptable tradeoff spaces by controlling the amount of errors in the system. This expanded tradeoff space allows system designers to comparatively evaluate different operating points with varying QoS and energy consumption by aggressively exploiting error-resilience attributes, and can potentially result in significant energy savings. Specifically, we propose an error-aware video encoding technique that intentionally injects errors (drops frames) while ensuring QoS in accordance with error-resilience. The novelty of our approach is in *active exploitation of errors* to vary the operating conditions for further optimization of system aspects. Our experiments show that our error-exploiting video encoding can reduce the energy consumption for an encoding device by 37% in video conferencing over a wireless network, without video quality degradation, compared to a standard video encoding technique for a test video stream. Furthermore, we present the adaptivity of our approach by incorporating the feedback from the decoding side to achieve the QoS requirement under dynamic network status.

## 1 Introduction

Due to the rapid deployment of wireless communications, video applications on mobile embedded systems such as video telephony and video streaming have grown dramatically. A major challenge in mobile video applications is how to efficiently allocate the limited energy resource in order to deliver the best video quality. A significant amount of power in mobile embedded systems is consumed by video pro-

Kyoungwoo Lee · Minyoung Kim · Nikil Dutt · Nalini Venkatasubramanian
Department of Computer Science, School of Information and Computer Sciences, University of California, Irvine, CA 92697, USA
e-mail: `{kyoungwl,minyounk,dutt,nalini}@ics.uci.edu`

cessing and transmission. Also, error resilient video encodings demand extra energy consumption in general to combat the transmission errors in wireless video communications. Thus, it is challenging and essential for system designers to explore the possible tradeoff space and to increase the energy saving while ensuring the quality satisfaction even under dynamic network status. In this paper, we introduce the notion of *active error exploitation* to effectively extend the tradeoff space between energy consumption and video quality, and present an adaptive error-exploiting video encoding to maximize the energy saving with minimal quality degradation.

Tradeoffs between energy consumption and QoS (Quality of Service) for mobile video communication have been investigated earlier [3, 5, 12, 14, 17]. It is interesting to observe that the delivered video data is *inherently error-tolerant*: spatial and temporal correlations between consecutive video frames are used to increase the compression efficiency, and results in errors at the reconstructed video data. These naturally induced errors from the encoding algorithms degrade the video quality, but they may not be perceived by the human eye. This inherent error-tolerance of video data can be exploited to increase the energy reduction for mobile embedded systems. For instance, relaxing the acceptable quality of the video reduces the overhead for the exhaustive searching algorithm by exploring a partial area instead of the entire region. Further, we exploit errors actively for the purpose of energy reduction. In our study, one way of active error exploitation is to intentionally drop frames before the encoding process. By dropping frames, we eliminate the entire video encoding process for these frames and thereby reduce energy consumption while sacrificing some loss in the video quality. Note that the effects of dropping frames on video quality are partially canceled with the nature of error-tolerance in video data.

To cope with transmission errors such as packet losses, error-resilient video encoding techniques [2, 9, 15, 16, 18] have been investigated to reduce the effects of transmission errors on QoS. Most existing error resilient techniques judiciously adapt their resilience levels considering the network status. Interestingly, our approach, combining these error-resilient techniques with intentional dropping frames, presents several pros and cons. First, we can improve the video quality to the level that error-resilient techniques achieve by considering these frame drops as packet losses occurring in the network. Second, we can increase the error margins that video encoders potentially exploit for maximal energy reduction, i.e., we can drop more frames. On the other hand, the error-resilience increases the compressed video data in general, and so raises the energy consumption for data transmission. Thus, this active error-exploitation approach with error resilient techniques significantly enlarges the tradeoff space among energy consumption for compression, energy consumption for transmission, and QoS in mobile video applications. Furthermore, our error exploiting video encoding scheme extends the applicability of error resilient schemes, even when the network is error-free.

In this paper, we propose a new knob, *error injection rate* (*EIR*) that controls the amount of data to be dropped. This EIR knob can be used to explore the tradeoff space between the energy consumption and video quality, unlike in previous approaches. Specifically, we present an error-exploiting video encoding with EIR based on an existing error-resilient video encoding, PBPAIR (Probability-Based

Power-Aware Intra-Refresh) [9]. Our new approach, called Error-Exploiting PBPAIR or *EE-PBPAIR*, is composed of two units: *error-injection unit* and *error-canceling unit*. The error-injection unit drops frames intentionally according to EIR. And the error-canceling unit applies PBPAIR to encode video data resilient against intentional frame drops. Active error exploitation can reduce the overheads for transmission and even the decoding, and results in the energy savings of all components in an encoding-decoding path in distributed mobile embedded systems. However, very aggressive error injection in EE-PBPAIR can degrade the video quality significantly, and there is a need to monitor the delivered video quality in distributed systems and to adjust the EIR to ensure satisfactory quality. Thus, we also present *adaptive EE-PBPAIR*, which adapts the EIR based on the quality feedback from the decoding side while minimizing the energy consumption.

The contributions and results of our work are:

- We propose the notion of active error exploitation, that extends the energy/QoS tradeoff space for video encoding on power-constrained embedded systems.
- We present an error-exploiting video encoding, *EE-PBPAIR*, by dropping frames intentionally in accordance with an error resilient scheme, PBPAIR [9].
- We present a feedback-based quality adjustment technique by adapting the EIR to meet the quality constraint – *adaptive EE-PBPAIR*.
- We demonstrate the efficacy of our approach: as compared to a standard video encoding based on H.263 [7], EE-PBPAIR reduces the energy consumption of an encoding device by up to 37% over a video stream without quality degradation, and by up to 49% at the cost of 10% quality degradation.

## 2 Background

Energy and QoS aware adaptations have been studied for video applications on mobile embedded systems in a cross-layer manner [12, 17]. In particular, Mohapatra et al. [12] explored the effects of video encoding parameters such as quantization scale, IP-ratio, and motion estimation algorithms on energy consumption and QoS, and proposed an integrated power management technique with middleware adaptations aware of system configurations. On the other hand, Eisenberg et al. [3] exploited the knowledge of the concealment method at the decoder to reduce the transmission power. However, although energy/QoS aware schemes have studied the tradeoff between energy consumption and QoS, they did not take into account error resilience against unreliable transmission and active error exploitation.

One of the most effective methods to achieve error-resilient video is to introduce the intra-coded frame (I-frame) periodically: since I-frames are decoded independently, they protect the propagation of the transmission errors and even encoding errors in previous frames. However, the transmission of I-frames causes delay and jitter (due to relatively large size) compared to predictively-coded frames (P-frames), and the loss of I-frames is more sensitive on QoS than that of P-frames [2, 9]. To mitigate both the propagation of the transmission errors and the overheads of large I-frames, recently intra-MB (Macroblock) refresh approaches have been proposed

[2, 9, 16]. Intra refresh techniques distribute intra-MBs among frames, which not only removes the overheads of I-frames but also improves the error-resilience. While most intra-MB refresh techniques have been focused on alleviating the effects of the transmission errors on the video quality, Kim et al. [9] proposed an energy-efficient and error-resilient video encoding named PBPAIR, and presented tradeoffs among error resilience, encoding efficiency, and energy consumption for video encoding. However, existing error resilient techniques have focused on how to manage the errors from network in a passive manner (*passive error exploitation*). On the contrary, our *active error exploitation* maximizes the feature of applications by intentional error injection – an approach has not been applied to video encoding earlier.

Our approach actively exploits the error tolerance of video data by injecting errors intentionally at the application level, and an error resilient video encoder along with *adaptive error injection* minimizes the effect of injected errors on the QoS. Therefore, our error-exploiting video encoding approach aggressively exploits error resilience to achieve the maximal energy gain while ensuring the QoS, and further opens opportunities to expand the tradeoff spaces between QoS and energy consumption in mobile embedded systems.

## 3 Our Approach

### *3.1 Fundamentals of Active Error Exploitation*

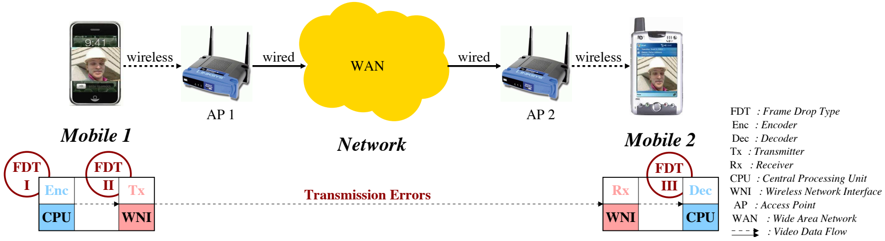


**Fig. 1** System Model and Frame Drop Types I/II/III for Active Error Exploitation

Fig. 1 shows our system model for mobile video conferencing. This mobile video conferencing system consists of two mobile devices and the network environment. Each mobile device is modeled as a mobile embedded system composed of a CPU and a WNI, where video data is encoded (or decoded) and transmitted (or received). The network consists of WAN and two wireless access points, each of which provides the wireless communication channel for each mobile device. For simplicity, we consider one path from an encoder to a decoder for mobile video conferencing.

We *exploit errors actively*. In our study, active exploitation of errors means intentional frame dropping in mobile embedded systems. For the purpose of energy reduction, video frames can be dropped by any component in Fig. 1. For instance, the Decoder can drop the delivered video data to reduce the decoding energy (*Frame Drop Type III* as in Fig. 1). Another possibility is that the Transmitter drops video data to save the communication energy, and error resilient techniques take care of the dropped data in advance (*Frame Drop Type II*). Further, the Encoder can drop frames intentionally before the encoding process, and encode the rest of frames robust against the dropped frames, which are considered as lost packets in network (*Frame Drop Type I*). Note that dropping frames at the Encoder is the most effective in terms of energy reduction since it affects the energy consumption across all the following components in an encoding-decoding path as drawn in Fig. 1, and the energy consumption for encoding (Enc EC) is relatively high compared to those for the other components in our system model. Therefore, in this particular work, we only consider *Frame Drop Type I* (i.e., intentional frame drop at the Encoder) for active error-exploitation approach. *Type II* and *III* remain as our future work.

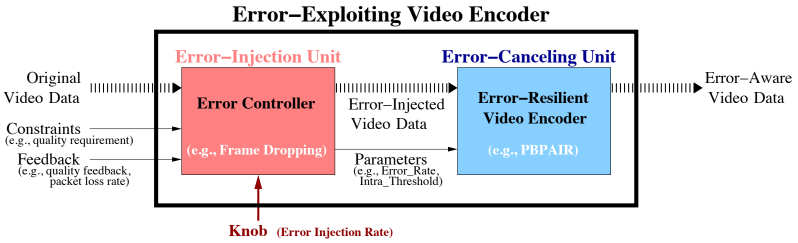

Error–Exploiting Video Encoder
Error–Injection Unit
Error–Canceling Unit
Original Video Data
Error Controller
(e.g., Frame Dropping)
Error–Injected Video Data
Error–Resilient Video Encoder
(e.g., PBPAIR)
Error–Aware Video Data
Constraints (e.g., quality requirement)
Feedback (e.g., quality feedback, packet loss rate)
Parameters (e.g., Error_Rate, Intra_Threshold)
Knob (Error Injection Rate)

**Fig. 2** Error-Exploiting Video Encoder: Error-Injection Unit and Error-Canceling Unit

Our error-exploiting video encoder is composed of two units, *error-injection unit* and *error-canceling unit*, as shown in Fig. 2. The *error-injection unit* controls the amount of errors for the purpose of energy reduction, and the *error-canceling unit* reduces the effects of the injected errors on the video quality using an error-resilient video encoder. The *Error Controller* acts as an *error-injection unit*, taking into account the constraint (e.g., required video quality) and the feedbacks from the decoding side (e.g., reconstructed video quality) and from the network (e.g., packet loss rate); furthermore, it intentionally injects the amount of errors according to a new knob – error injection rate (EIR), and generates the error-injected video data as illustrated in Fig. 2. Finally, the *Error-Resilient Video Encoder* acts as an *error-canceling unit*, and generates the error-aware video data by encoding the error-injected video data with parameters in preparation for downstream network packet losses as well as intentionally injected errors.

## 3.2 EE-PBPAIR: An Error-Exploiting Video Encoder

We now present *EE-PBPAIR* (Error-Exploiting PBPAIR), an approach that injects errors intentionally by "Dropping Frames" as an *error-injection unit*, and encodes video resiliently with "PBPAIR" as an *error-canceling unit* as shown in Fig. 2.

Dropping frames is one way of injecting errors intentionally. In this study, we consider a simple frame dropping approach, PFD (Periodic Frame Dropping). PFD periodically drops frames according to EIR. For instance, PFD with 10% of EIR drops every $10^{th}$ frame. PFD evenly distributes the effects of frame dropping on QoS over a video stream. Note that the quality will be deliberately maintained by the fea-

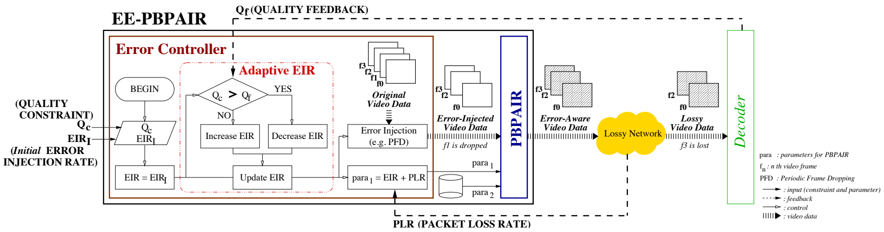


**Fig. 3** Flow of Error Controller and Adaptive EIR in EE-PBPAIR for Mobile Video Applications

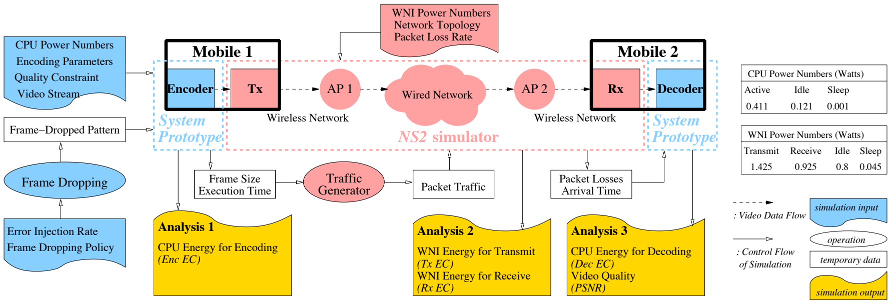
ture of error-resilient PBPAIR. We use PBPAIR as an error-resilient video encoder since the authors in [9] have demonstrated its energy efficiency while maintaining video quality and robustness against network packet losses. PBPAIR takes two parameters as shown in Fig. 3. The first parameter ($para_1 = Error\_Rate$) indicates the current network status such as packet loss rate (PLR), and the second parameter ($para_2 = Intra\_Threshold$) represents the level of error resilience requested. To consider both injected errors and packet losses, EE-PBPAIR calculates the sum of EIR and PLR for $para_1$ while PBPAIR originally takes PLR as $para_1$. For instance, the first parameter ($para_1$) in EE-PBPAIR is set to 15% when EIR is 10% while PLR in network is 5%. Note that active error exploitation is orthogonal to PBPAIR and can be applied to any error-resilient and energy-efficient video encoding technique which adapts algorithmic parameters according to the network status.

Our error-exploiting video encoder saves the energy consumption in several ways: i) intentional frame dropping saves energy consumption since EE-PBPAIR skips frame encodings according to EIR. ii) the energy consumption for video encoding is reduced since EE-PBPAIR adaptively introduces the more intra-MBs instead of inter-MBs for error resilience due to the intentional frame drops. iii) intentional frame dropping can reduce the encoded video file size, which propagates the energy saving downstream to the Transmitter, the Receiver, and even the Decoder.

A high EIR increases the energy reduction for encoding but decreases the QoS, if it is beyond a manageable point for error-resilient encoding. To keep the QoS degradation minimal, our approach is able to constrain the EIR based on the feedback from the decoding side. Fig. 3 describes this adaptive EIR feature in *Error Controller*. *Error Controller* takes the quality constraint $Q_c$ and sets the initial error injection rate $EIR_I$. Then it receives the feedback such as $Q_f$ and $PLR$ as shown in the feedback loop of Fig. 3. If $Q_f$ is less than $Q_c$, the current EIR is bad in terms of QoS, and so the EIR is decreased. Otherwise, it is increased (the flow of "Adaptive EIR" in Fig. 3). Based on EIR, the error injection module periodically drops frames. Thus, *Error Controller* forwards the error-injected video data to the *PBPAIR* as shown in Fig. 3. Also $para_1$ is delivered to the *PBPAIR*, which encodes the error-injected video data robust against the amount of errors indicated as $para_1$, with $para_2$ selected by PBPAIR methodology. Consequently, the encoded video data is now error-aware, i.e., it is cognizant of injected errors and packet losses as illustrated in Fig. 3. This adaptive video encoder adjusts EIR to meet the quality constraint with minimal energy consumption. So we believe that our adaptive approach can be effectively used to adjust our video encoder under a dynamic network environment for maximal energy reduction while ensuring the QoS. Note that the frequencies of feedbacks such as $Q_f$ and *PLR* are beyond this work, and we assume that feedback channels are reliable.

# 4 Experiments

## *4.1 Experimental Setup*



| CPU Power Numbers (Watts) | | |
|---|---|---|
| Active | Idle | Sleep |
| 0.411 | 0.121 | 0.001 |

| WNI Power Numbers (Watts) | | | |
|---|---|---|---|
| Transmit | Receive | Idle | Sleep |
| 1.425 | 0.925 | 0.8 | 0.045 |

**Fig. 4** Experimental Framework for Mobile Video Conferencing System - *System Prototype* + *NS2*

For interactive multimedia applications such as mobile video conferencing in distributed embedded systems, an end-to-end experimental system framework is a necessity since all components in a distributed system work interactively and affect other components in terms of energy consumption and performance. Thus, we evaluated EE-PBPAIR on top of an end-to-end framework as shown in Fig. 4 consisting of a *System Prototype* [10] and *NS2* simulator [13]. The *System Prototype* emulates a mobile embedded system and is detailed in our technical report [10].

The left side of Fig. 4 shows the preprocessing step, where a pattern of dropped frames is generated according to an EIR. CPU power numbers, video encoder parameters, network status (PLR), and quality constraint are inputs to *System Prototype*, where a video encoder compresses a video stream. *System Prototype* analyzes the first set of results – Analysis 1 – such as the energy consumption for encoding (*Enc EC*), and calculates the encoded size and the encoding completion time of each video frame, which are used for generating the network traffic for the following network simulation. Analysis 1 succinctly shows the CPU energy for encoding at the sender. Next, *NS2* simulates the generated network traffic with a set of configurations including the network topology and WNI power values, and estimates the energy consumption (*Tx* and *Rx EC*) for WNIs – Analysis 2. Thus Analysis 2 captures the end-to-end networking effects, including those of the transmitter and the receiver. Finally at the receiver, the *System Prototype* decodes the transmitted video data based on generated packet losses and frame arrival times from *NS2*, and evaluates the energy consumption for decoding (*Dec EC*) and the video quality measured in PSNR (Peak Signal to Noise Ratio) in Analysis 3. Thus Analysis 3 captures the CPU energy for decoding at the receiver (Power consumption numbers for CPU [6] and WNI [8] are configured as shown in the tables on the right side of Fig. 4). By combining Analysis 1, Analysis 2 and Analysis 3, we are able to measure the entire end-to-end energy savings for our proposed scheme.

Using *NS2*, we simulate the network consisting of two IEEE 802.11 WLANs (Wireless Local Area Network) and a wired network connecting them as shown in Fig. 4. Each WLAN is composed of one access point (AP 1 or AP 2), and one mobile device (Mobile 1 or Mobile 2). We exclude the effects of traffic from other mobile stations since they affect the energy consumption of WNI in our mobile devices. Instead, we limit the data rate of WNI, which constrains the encoded bit rate, and show clearly the effects of the varying data size generated by the Encoder. For wireless connection, the data rate is set at 1 Mbps, considered to be an actual data rate [4, 11], and the link layer delay at 25 $\mu$s. *NS2* generates packet losses for a given PLR. Each encoded video frame is composed of multiple packets if its size is larger than MTU (Maximum Transfer Unit), 1.5 KB in our simulation. A frame is considered lost if any packet of the frame is lost through the network simulation.

Recall that our EE-PBPAIR approach combines PFD with PBPAIR. PBPAIR takes two parameters, $para_1$ and $para_2$. We set $para_1$ (*Error_Rate*) as the sum of EIR and PLR. For comparison, $para_2$ (*Intra_Threshold*) is chosen for requested quality with the same compression efficiency as GOP-K (Group-Of-Picture with K) [9]. In this study, GOP-K based on H.263 [7] is defined as a standard video encoder, where K indicates the number of P-frames between I-frames. In GOP-K, we change K for resilience against the transmission errors in network as in [1, 9]. As a test video sequence, *FOREMAN* in QCIF format (176×144 pixels) is used. To constrain the bandwidth, we consider that the bitrate is 64 kbps (kilobits per second) and frame rate is 5 fps (frames per second).

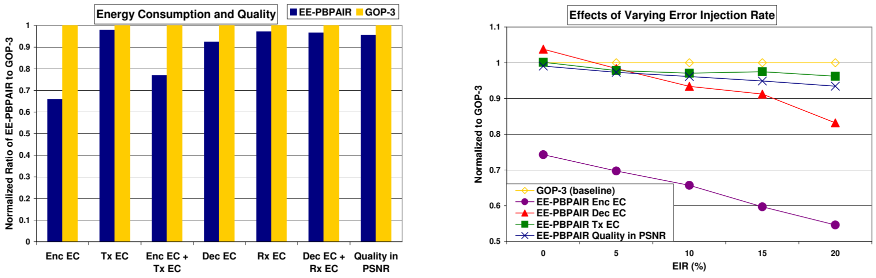
## 4.2 Experimental Results

We present three sets of results. First we show the energy reduction due to active error exploitation (Section 4.2.1). Second, we demonstrate the expanded design space allowing better exploration of tradeoff alternatives (Section 4.2.2). Finally, Section 4.2.3 demonstrates the efficacy of adaptive EE-PBPAIR that maintains quality under dynamic network conditions by incorporating the quality feedback.

### 4.2.1 Energy Reduction from Active Error-Exploitation

To show the effectiveness of our proposed technique, we evaluates EE-PBPAIR with 10% EIR in comparison to GOP-3 considering 10% of PLR in network.

Fig. 5(a) shows the effectiveness of an error-exploiting approach on the energy reduction. The plots present the normalized energy consumption and the video quality of EE-PBPAIR to those of GOP-3, and clearly show that EE-PBPAIR is very effective compared to GOP-3 in terms of each category of energy consumption with slight quality degradation. Specifically, EE-PBPAIR consumes 34% less energy than GOP-3 for encoding (Enc EC) since it drops 10% of video frames and compresses more macro-blocks with less expensive intra encodings than predictive encodings. In terms of energy consumption for transmitting video data (Tx EC), EE-PBPAIR



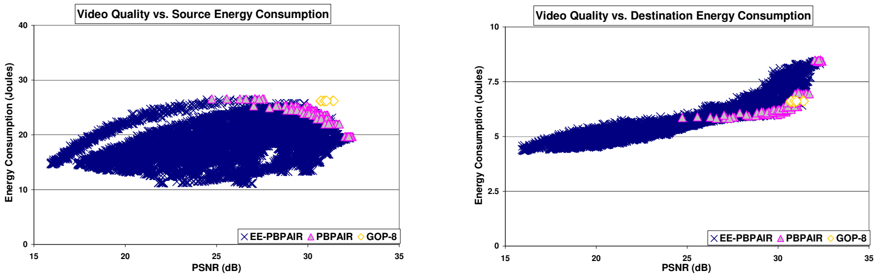
(a) EIR = 10%

(b) EIR = 0% to 20%

**Fig. 5** Effects of Error Injection Rate on Energy Consumption and Video Quality in EE-PBPAIR compared to GOP-3 (PLR = 10%, FOREMAN 300 frames)

sends a similar amount of data within less time than GOP-3, which results in the slight energy reduction. Thus, the energy consumption for the source (Enc EC and Tx EC) is reduced by 23% with EE-PBPAIR, at the cost of 4% quality degradation. Note that 1% quality degradation indicates about 0.31 dB reduction from the PSNR value for GOP-3. At the destination, EE-PBPAIR reduces the energy consumption by 8% for the decoding (Dec EC), which mainly results from dropping 10% frames at the source. Note also that more intra-encoded MB results in more energy consumption for the decoding but 10% frame dropping compensates for this effect. EE-PBPAIR saves the energy consumption for the receiver (Rx EC) by 3% mainly due to the smaller duration for receiving. The energy consumption at the destination (Dec EC + Rx EC) is reduced by 5%. These results are very effective in energy reduction with respect to all energy categories at the cost of slight quality degradation, which is an acceptable tradeoff for power-hungry mobile embedded systems.

We now illustrate how EIR is effective as a knob to tradeoff the quality for energy reduction. To observe the effects of varying EIR on quality and energy consumption, we compare EE-PBPAIR with GOP-3 by varying EIR from 0% to 20%. Fig. 5(b) shows the normalized video quality and each energy consumption of EE-PBPAIR to those of GOP-3. Since we adapt $para_2$ of PBPAIR to minimize the transmission overhead, the energy consumption for the data transmission (Tx EC) of EE-PBPAIR with varying EIR is close to that of GOP-3. With an increase of EIR, quality is still managed within an insignificant degradation of quality, and this quality management is mainly because of the error-resilient feature of EE-PBPAIR. With 20% EIR, the loss of quality is 7% in PSNR. Fig. 5(b) clearly shows that increasing the EIR significantly saves energy consumption for encoding (Enc EC). Since the portion of intra-MBs for each frame is increasing for error resilience, the energy consumption for the decoding is higher than GOP-3 with low EIR between 0% and 5%. However, with an increase of EIR, the number of frames to be decoded is decreasing and thus the energy consumption decreases. With 20% EIR, we obtain 45% energy reduction for encoding, and 17% reduction for decoding at the cost of 7% quality degradation.



(a) Video Quality in PSNR vs. Source Energy Consumption (Enc EC + Tx EC)

(b) Video Quality in PSNR vs. Destination Energy Consumption (Dec EC + Rx EC)

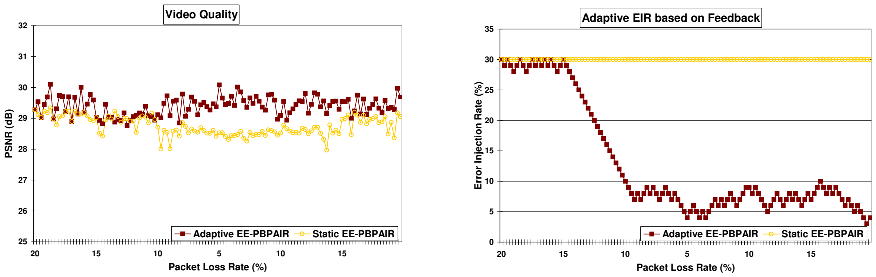
**Fig. 6** Extended Tradeoff Space between Video Quality and Energy Consumption by EE-PBPAIR in comparison to GOP-8 and PBPAIR (EIR = 0% to 50%, PLR = 5%, FOREMAN 300 frames)

### 4.2.2 Extended Energy/QoS tradeoff

Fig. 6(a) plots the energy consumption at the source vs. quality of EE-PBPAIR compared to PBPAIR and GOP-8, and clearly shows that design space of EE-PBPAIR is much larger and more effective than those of PBPAIR and GOP-8. As compared to PBPAIR, the tradeoff space of EE-PBPAIR subsumes all spaces for PBPAIR since indeed EE-PBPAIR with 0% of EIR is PBPAIR. As compared to GOP-8, EE-PBPAIR generates a better design space in terms of the energy reduction (by up to 37%) without losing video quality, and presents even better video quality with less energy consumption. Further, relaxing the quality requirement (such as 10% QoS degradation) compared to GOP-8 increases the energy reduction at the source by up to 49%. Thus, EE-PBPAIR very effectively expands the design space between the source energy consumption and video quality by exploiting the intentional errors. Fig. 6(b) depicts the tradeoff space between the energy consumption at the destination and the video quality, and clearly shows that EE-PBPAIR greatly extends the spaces explored by PBPAIR and GOP-8. However, the energy saving at the destination using EE-PBPAIR is less effective than that at the source since the resilience approach encodes more intra-MBs, which decreases the energy saving resulting from the intentional error injection. Even then, EE-PBPAIR can save the energy consumption by 3% without losing QoS compared to GOP-8.

### 4.2.3 Adaptive EE-PBPAIR: Ensuring Quality under Dynamic Network

To show the effectiveness of our adaptive EE-PBPAIR by updating EIR, we model a dynamic network and compare adaptive EE-PBPAIR to static EE-PBPAIR (i.e., EE-PBPAIR with a fixed EIR). For this experiment, PLR begins with 20% and decreases by 5% every 20 runs and after 5% PLR it increases by 5% until it reaches 15%. Each run captures 300 frames of video encoding. The horizontal axis in Fig. 7 represents this scenario with varying PLR. The quality constraint is set to 29.6 dB in PSNR, which is about 10% quality degradation from GOP-3 without any losses. Static EE-PBPAIR encodes the video data with a fixed EIR = 30% (since 30% EIR degrades the video quality significantly in some dynamic network situations as shown in Fig. 7(a)) while adaptive EE-PBPAIR starting with 30% EIR and updates it



(a) Adaptive EE-PBPAIR delivers better video quality than static EE-PBPAIR

(b) Adaptive error injection rate according to quality feedback

**Fig. 7** Adaptive EE-PBPAIR Robust to Varying PLR under Dynamic Network Status

according to the quality feedback. Fig. 7(a) draws the PSNR values for adaptive EE-PBPAIR in comparison to static EE-PBPAIR, and shows that the delivered quality of adaptive EE-PBPAIR is consistently better than that of static EE-PBPAIR. EE-PBPAIR adapts the EIR according to the feedback with respect to the video quality as shown in Fig. 7(b). In conclusion, this EIR adaptivity with EE-PBPAIR adjusts the quality of service based on the feedback for mobile video applications in distributed embedded systems while minimizing the energy consumption.

## 5 Summary and Future Work

Mobile video applications pose significant challenges for battery-constrained embedded systems due to high processing power for compression algorithms and transmission of a large volume of video data. Fortunately, video applications tolerate errors inherently, and we exploit this error tolerance of video data for the purpose of the energy savings. Active error exploitation – – intentional frame dropping together with error-resilient video encoding – – can achieve significant energy gains while ensuring the video quality. We present a new approach that injects errors intentionally to balance the dual goals of energy efficiency and satisfactory QoS.

In this paper, we demonstrated our approach in two phases for video conferencing applications running on resource-limited mobile systems. First we presented EE-PBPAIR that combines an error-resilient video encoder (PBPAIR) with intentional frame dropping to significantly reduce the energy consumption for the entire encoding-decoding path of the video conferencing application. Our experiments also demonstrated that the active error exploitation of EE-PBPAIR allows system designers to consider larger tradeoff spaces than previous approaches: GOP-K and PBPAIR. Further, we proposed an adaptive EE-PBPAIR by controlling a new knob, error-injection rate (EIR), in order to satisfy the delivered quality based on the feedback under the dynamic network status.

Our future work includes intelligent frame dropping techniques for further energy reduction with minimal quality degradation. We also plan to extend active error exploitation to the system level combined with error-aware architectures and network protocols to maximize the energy reduction for distributed embedded systems.

## References

1. Liang Cheng and Magda El Zarki. An adaptive error resilient video encoder. In *SPIE Visual Communication and Image Processing*, July 2003.
2. Liang Cheng and Magda El Zarki. PGOP: An error resilient techniques for low bit rate and low latency video communications. In *Picture Coding Symposium (PCS)*, Dec 2004.
3. Y. Eisenberg, C. Luna, T. Pappas, R. Berry, and A. Katsaggelos. Joint source coding and transmission power management for energy efficient wireless video communications. *IEEE Trans. Circuits Syst. Video Technology*, 12:411–424, 2002.
4. L. Guo, X. Ding, H. Wang, Q. Li, S. Chen, and X. Zhang. Exploiting idle communication power to improve wireless network performance and energy efficiency. In *IEEE International Conference on Computer and Communications (INFOCOM)*, pages 1–12, April 2006.
5. Al Harris, Cigdem Sengul, Robin Kravets, and Prashant Ratanchandani. Energy-efficient multimedia communications in lossy multi-hop wireless networks. *IFIP Mobile and Wireless Communication Networks*, 162:461–472, 2005.
6. Intel Corporation, http://www.intel.com/. *Intel PXA255(R) Processor: Developer's Manual*.
7. ITU-T. *H.263 Draft: Video Coding for Low Bitrate Communication*, May 1996.
8. Yu Jiao and Ali R. Hurson. Adaptive power management for mobile agent-based information retrieval. In *IEEE Advanced Information Networking and Applications (AINA)*, pages 675–680, March 2005.
9. M. Kim, H. Oh, N. Dutt, A. Nicolau, and N. Venkatasubramanian. PBPAIR: An energy-efficient error-resilient encoding using probability based power aware intra refresh. *ACM SIGMOBILE Mobile Computing and Communications Review*, 10:58–69, July 2006.
10. Kyoungwoo Lee, Minyoung Kim, Nikil Dutt, and Nalini Venkatasubramanian, Tech Rep. (http://www.ics.uci.edu/∼kyoungwl/eepbpair/). *Adaptive EE-PBPAIR: A Novel Error-Exploiting Video Encoder Incorporating End-to-End QoS Feedback*, Dec 2007.
11. Jens Meggers, Gregor Bautz, and Anthony Sang-Bum Park. Providing video conferencing for the mobile user. In *IEEE Conference on Local Computer Networks*, page 526, March 1996.
12. S. Mohapatra, R. Cornea, H. Oh, K. Lee, M. Kim, N. Dutt, R. Gupta, A. Nicolau, S. Shukla, and N. Venkatasubramanian. A cross-layer approach for power-performance optimization in distributed mobile systems. In *Next Generation Software Program in conjunction with IPDPS*, page 218.1, April 2005.
13. NS2. Network Simulation version 2, http://www.isi.edu/nsnam/ns/.
14. Clark N. Taylor, Sujit Dey, and Debashis Panigrahi. Energy/latency/image quality tradeoffs in enabling mobile multimedia communication. In *Proc. of Software Radio: Technologies and Services*, pages 55–66. Springer Verlag, Jan 2001.
15. Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos. Review of error resilient coding techniques for real-time video communications. *IEEE Signal Processing Magazine*, 17:61–82, July 2000.
16. S. Worrall, A. Sadka, P. Sweeney, and A. Kondoz. Motion adaptive error resilient encoding for mpeg-4. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 3, May 2001.
17. Wanghong Yuan, Klara Nahrstedt, Sarita V. Adve, Douglas L. Jones, and Robin H. Kravets. Design and evaluation of a cross-layer adaptation framework for mobile multimedia systems. In *Proceedings of SPIE/ACM Multimedia Computing and Networking Conference (MMCN)*, January 2003.
18. Rui Zhang, Shankar L. Regunathan, and Kenneth Rose. Video coding with optimal inter/intra-model switching for packet loss resilience. *IEEE Journal on Selected Areas in Communications*, 18(6):966–976, June 2000.