Power Efficient Multimedia Communication over Wireless Channels

Xiaoan Lu, *Student Member, IEEE,* Elza Erkip, *Member, IEEE,*
Yao Wang, *Senior Member, IEEE,* and David Goodman, *Fellow, IEEE*
Electrical and Computer Engineering, Polytechnic University, Brooklyn, New York
Contact E-mail: xlu@vision.poly.edu

This material is based upon work partially supported by the National Science Foundation under Grant No. 0219822, the New York State Center for Advanced Technology in Wireless Internet, and Philips Research USA.

Abstract

In this work we introduce an approach for minimizing the total power consumption of a mobile transmitter due to source compression, channel coding and transmission subject to a fixed end-to-end source distortion. We illustrate our approach both on an abstract class of sources and channels and on a realistic H.263 video transmission system through a wireless channel. Performance under different channel environments and implementation schemes are investigated. Our numerical analysis shows that optimized settings can reduce the total power consumption by a significant factor and prolong battery life considerably compared to fixed parameter settings.

Keywords

Video transmission, wireless channel, power consumption, error resilience, power allocation

I. Introduction

Efficient use of the limited battery power is one of the major challenges in portable information devices. The management of power becomes even more critical with devices integrating complex video signal processing techniques with communications. Some of the key technologies that affect the battery life in this respect are source signal compression, channel error control coding and radio transmission.

Classically, source and channel coding literature and in particular joint source and channel coding techniques mainly focus on designing codes that minimize the overall distortion of the source as it travels through the channel [1], [2]. However, for mobile units that have limited battery capacities, power consumption of base-band processing should also be taken into account. On the other hand, the work on improving the battery life has focused on separate components such as algorithms and hardware design for specific video and channel coders and low power transmitter design [3], [4]. Joint optimization of source compression, channel coding and transmission to balance the quality of service and power requirements of the mobile has only recently attracted interest. In [5], the authors look at some specific operation modes of the H.263 video coder that result in different compression rates and almost the same perceptual quality. Multi-stage coded modulation is utilized to accommodate rates in the different modes. The authors find that, by judiciously selecting operating modes in response to mobile environment, lower energy consumption can be achieved. The work by Appadwedula et. al. [6], considers minimization of the total energy of a wireless image transmission system. By choosing the coded source bit rate for the image coder, redundancy for the Reed-Solomon (RS) coder, transmission power for the power amplifier and the number of fingers in the RAKE receiver, the total energy due to channel codec, transmission and the RAKE receiver is optimized subject to end-to-end performance of the system. The proposed system is simulated for an indoor office environment subject to path loss and multipath. Significant energy saving is reported. In [7], [8], by changing the accuracy of motion estimation different power and distortion levels for H.263 encoder are provided [9]. The coded bits are packetized and unequally protected using RS codes and are transmitted over a CDMA system operating over a flat fading channel. The authors consider the processing power for source coding and channel coding as well as transmission power for a given video quality and propose a power-minimized bit-allocation scheme.

This paper is an extension of our previous work [10], [11] and considers how to adjust the operating parameters of the source coder, channel coder and transmitter to minimize the total power dissipation while keeping the end-to-end distortion of the source constant. In this paper we concentrate on total transmitter power, envisioning a situation such as uplink cellular communications where the base station receiver does not have power limitations. Similarly, for the downlink scenario the goal would be to minimize the receiver power consumption. For peer-to-peer communication situations such as ad-hoc networks or cellular networks when the base station does not do transcoding, one would optimize over both the transmitter and receiver power levels.

In order to examine the theoretical limit of power savings obtainable by adaptation of operating parameters, and to investigate how the channel condition influences the adaptation strategy, we first consider the transmission of a Gauss-Markov source [12] over an additive white Gaussian noise (AWGN) channel. We assume the source is compressed using transform coding, with which one can easily adjust the tradeoff between coding efficiency and power consumption by varying the transform vector dimension. For this study, we ignore channel coding, and examine how to optimally vary the transform dimension of the source coder and the radiated energy per bit at the transmitter according to the channel quality, or the distance between mobile and base station. We consider path loss, so the channel quality varies inversely with distance between the mobile and the base station. When the distance is large, one must spend more energy on transmitting each compressed bit with low error probabilities. Our study shows that in this case, one should reduce the source rate by using a high transform dimension. On the other hand, when the distance is small, it is relatively inexpensive to send each bit, so the source coder can operate at a high bit rate by using a lower transform dimension and hence lower computation complexity. We observe that up to ten fold savings in power consumption can be obtained by adapting the operating parameters based on channel conditions, compared to using fixed settings. These results are presented in Section II.

The proceeding study, although insightful, does not accurately capture characteristics of real video sources. We also consider a practical wireless video communication system that uses standard H.263 [13] video codec and the RS channel codec. The channel is characterized by AWGN model. For source encoding, an H.263 [13] compliant video coder is used. Two parameters, the quantization step size and the INTRA update rate, are used to control the source coder bit rate and the power consumption. The INTRA rate also affects the error resilience of the coded bit stream. In practice, other rate control and error control schemes can further be used. We develop simple models for the power consumption of the H.263 source encoder and RS channel coder, both implemented in software, and validate our models through measurements. Combining our models with power radiated from the transmitter, and models of distortion caused by compression and transmission [2], we formulate the power optimization problem subject to a given distortion constraint. By solving this problem numerically we show that optimized settings can result in substantial power savings. We also discuss the effects of various parameters in the system. In particular, we investigate the effect of channel error rate on the optimized parameters by varying the distance between the mobile and the destination. We also study the effect of the video content and compare optimal levels of compression and channel coding of two video sequences, one with slow motion and another with fast motion. The implementation platforms (e.g., software versus hardware) of the source and channel coder affect the power spent on signal processing, and in return change the power optimal settings. We consider two communication systems: in one system both compression and channel coding are implemented by hardware; and in the other both are in software. The ratio of power consumption of signal processing and transmission and hence optimal parameters vary significantly in these two applications.

One main difference between our study and those in [5], [7] and [8], which also consider the H.263 video codec, lies in the source coding parameter used to control the tradeoff between the power consumption and coding efficiency. We choose to vary the INTRA rate (the frequency that a macroblock is coded in the INTRA mode), which controls how often to conduct motion estimation. In [5], the selection is made among five fixed encoding modes and corresponding coded modulation configurations. The distortion caused by compression is the same for each coder. In our scheme, the compression distortion can be varied as long as the overall end-to-end distortion is kept constant. Hence our technique provides a larger search space for power optimization. On the other hand, the study in [7], [8] uses a fixed INTRA rate, but changes the motion estimation method among several modes. We believe that varying the INTRA rate is more effective, because it can lead to a larger adaptation space in the source coded bit rate and power consumption by H.263 encoder.

Another difference between our study and those in [5], [7] and [8] lies in the way the end-to-end distortion is determined. We use an analytical model reported in [2] which has been shown to be quite accurate. The study in [5], [7] and [8], however, relies on the distortion determined from actually coded and transmitted video sequences (albeit the transmission process is simulated based on some models). The distortion model enables us to determine the optimal parameter settings for a set of channel conditions without running real video sequences through a video codec. Thus our approach is computationally more affordable.

Before going to the individual sections, it is worthwhile to discuss how the proposed adaptation may be implemented in practice. We envision the existence of a "power manager", whose operations may be distributed between the portable terminals and a base station. One likely realization is to pre-compute the adaptation rules for a discrete set of quality-of-service (QoS) requirements, channel conditions, video sequence properties (e.g. in terms of motion activity), and implementation architectures (e.g. hardware versus software), and store these rules as a look-up table in the base station. In this paper, we only consider the distortion as a QoS measure. End-to-end delay could be an additional measure. In addition to channel conditions, remaining battery energy on the portable device could be another input parameter, which will constrain the search space in power allocation. Finally, our power optimized compression, channel coding and transmission algorithms can be part of a larger system in which the "power manager" is also able to change level of encryption, battery discharge profiles and voltage levels based on parameters of the environment. Some of our progress towards this integration can be found in our DREAM-IT (Dynamically Reconfigurable Energy Aware Multimedia Information Terminals) project web page [14].

II. Total power optimization for abstract sources and channels

In this section we consider a simple case in which we ignore the channel coding and we send source compressed bits through an AWGN channel. We have a simple cyclic redundancy check (CRC) to detect errors.

The total power consumed by the mobile at the application layer and at the physical layer consists of the power dissipated by the source compressor P_s and the power used to transmit the compressed bits through the channel P_t, resulting in total power $P_{tot} = P_s + P_t$ (watts). Our goal is to solve the problem

$$\text{Minimize } P_{tot} \text{ subject to } D_{tot} \leq D_0, \tag{1}$$

where D_{tot} refers to the total distortion due to source compression and channel errors. Total allowed distortion D_0 will be determined by the particular source and application at hand and will be different for telephone calls and video conferencing.

To gain insight into the solution, we first consider a special case where we fix source distortion D_s and channel error rate individually. We assume we have access to a number of source compression techniques which provide the same source distortion with varying complexity and rate. We plot the operational rate-distortion curves of these compression schemes in Fig. 1(a). Note that operational rate-distortion curve describes the rate and distortion tradeoff for a particular compression algorithm and parameter setting. Generally algorithms that compress more, resulting in lower compressed rates for identical "quality" have higher complexity and require more processing power. Hence for fixed source distortion D_s in Fig. 1(a), the processing power P_s is a decreasing function of the compressed bit rate R_s shown in Fig. 1(b). On the other hand, as the number of bits representing a source sample increases and to keep the bit error rate of the channel constant, we need higher transmission power P_t as in Fig. 1(c). Combining in Fig. 1(d), we see that there exists an optimal operating point R_s^* that minimizes P_{tot} for fixed end-to-end source distortion.

As an example of this optimization we consider a class of transform coders of varying vector dimension N, which is the block size of the transform, and squared error distortion function. In such a coder, using a longer vector length N can increase the coding efficiency, at the expense of more computational power. Also, the compressed bit stream becomes more error prone as N increases, thus requiring a higher transmit power per bit to maintain the same distortion due to transmission errors. Therefore, an intermediate value of N is likely to be best, depending on the actual channel condition. We consider a first order Gauss-Markov source with variance σ_s^2 and autocorrelation function $c(k) = \sigma_s^2|\rho|^k$, where ρ is the correlation coefficient [12]. We assume the source is sampled at a rate of f_s samples/second. The operational distortion-rate function of a transform coder using the optimal transform of dimension N is

$$D(R_s) = \epsilon\sigma_s^2(1-\rho^2)^{\frac{N-1}{N}}2^{-2R_s}, \tag{2}$$

where ϵ depends on the quantizer used for the transform coefficients [15]. Note that R_s is the number of bits/sample and $D(R_s)$ is the distortion per source sample in terms of mean square error (MSE).

We assume NR_s compressed bits from one vector are grouped as one packet and CRC is applied at the packet level. The compressed bits are transmitted through an AWGN channel of noise spectral density N_0 using differential phase shift keying (DPSK). The probability of bit error p_e is given by [16]

$$p_e = \frac{1}{2}e^{-\frac{E_{b,rec}}{N_0}} = \frac{1}{2}e^{-\frac{P_{rec}}{R_s f_s N_0}}, \tag{3}$$

where $E_{b,rec}$ is the received energy per bit and P_{rec} is the received power in watts. Considering only path loss, the radiated power at the transmitter is $P_{rad} = P_{rec}d^\alpha$, where d

is the distance between the mobile and the receiver and the exponent α depends on the propagation medium.

Since our source is compressed using transform coding of dimension N, we will assume that N source samples (which we call a "symbol") are received correctly only when all the NR_s bits describing the vector are correct at the receiver. This results in the probability of symbol error p_s

$$p_s = 1 - (1 - p_e)^{NR_s}. \tag{4}$$

From (4), we see that when the bit error rate p_e is the same, the symbol error rate p_s increases with N, which means that the compressed bit stream is more sensitive to transmission errors.

We assume that when a vector is incorrectly received, CRC check indicates this and all samples in this vector are replaced by the mean value of the source, leading to MSE equal to the source variance σ_s^2. On the other hand, if the vector is received correctly, the MSE is $D(R_s)$. Thus, the total end-to-end distortion per sample can be expressed as

$$D_{tot} = (1 - p_s)D(R_s) + p_s\sigma_s^2. \tag{5}$$

Let us now turn to the calculation of total power P_{tot}. Transforming a source vector of dimension N requires N^2 operations. Each operation includes one complex multiplication and one complex addition. Typically the number of operations necessary for the quantizer and the entropy coder following the transform is negligible with respect to N^2. Assuming the power dissipated is proportional to the number of operations, with proportionality constant c_s, the power requirement of the source coder is $P_s = c_s f_s N^2/N = c_s f_s N$ watts. We assume the total power consumption at the mobile due to transmission is $P_t = c_t P_{rad}$, where c_t is a scaling factor that is device and implementation specific and accounts for the power used for modulation and amplification [17].

Using (2), (3), (4) and (5), the received power P_{rec} required to keep the total distortion at level D_0 is given by $P_{rec} = -R_s f_s N_0 \ln(2p_e)$, with $p_e = 1 - (1 - p_s)^{1/(NR_s)}$ and

$$p_s = \frac{(D_0 - \epsilon\sigma_s^2(1-\rho)^{\frac{N-1}{N}} 2^{-2R_s})}{(\sigma_s^2 - \epsilon\sigma_s^2(1-\rho)^{\frac{N-1}{N}} 2^{-2R_s})}. \tag{6}$$

Combining, the total power due to signal processing and transmission is

$$P_{tot} = c_s f_s N + (-c_t d^\alpha R_s f_s N_0 \ln(2p_e)), \tag{7}$$

and the optimization in (1), that is minimizing (7) with a constraint on (5), can be carried out with respect to transform coder dimension N and compression rate R_s. In our formulation, the value of the constants c_s and c_t are device and implementation specific and can be determined experimentally.

We illustrate the results for source variance $\sigma_s^2 = 1$, noise power spectral density $N_0 = 1 \times 10^{-15}$ watts/Hz, correlation coefficient $\rho = 0.9$, quantizer parameter $\epsilon = 1$, propagation constant $\alpha = 3.6$ and total distortion $D_0 = 0.1$ in Fig. 2. The vector dimension N is chosen from $\{1, 2, \ldots, 32\}$, and distance d is varied from 30 to 900 m. Note that $D_0 = 0.1$ corresponds to a SNR value of 10 dB for the source. We have plotted both the total power and transmit power as a function of the transform coding dimension N. For each N, the rate R_s is chosen so that $D_{tot} = D_0$. The vector size N that minimizes total power is denoted by a star. All the powers are in watts and normalized with respect to the $c_s f_s$, which is determined by the specific implementation. Experiments [18] suggest that P_s and P_t are comparable, so we have chosen $c_s/(c_t d^\alpha N_0)$ to be in the range of 0.1 to 10 around $d = 200$ m.

In Fig. 2(a), we consider $c_s/(c_t d^\alpha N_0) = 6.25$, modelling a scenario in which the mobile is close to the base station, or equivalently channel attenuation is low. We observe that as the transform coding dimension N increase, the total power and signal processing power, which is the difference between total power and transmit power, increases. The optimal N in this case is equal to 1 and results in source coding rate $R_s = 2$ bits/sample. Hence for good channel conditions, one does not need to spend a lot of power compressing the source; low attenuation enables more source bits to be transmitted through the channel.

Figure 2(b) shows total power and transmit power for $c_s/(c_t d^\alpha N_0) = 0.16$, simulating the case when the mobile is far away from the base station. The optimum source coding dimension N is 5 and the corresponding rate R_s is 0.8 bits/sample. Since the channel attenuation is high, it is relatively expensive to send each bit reliably, therefore, we need to compress the source sample more. The level of compression to minimize total power consumption is therefore *location dependent*.

Figure 3 and Table I summarize minimum total power consumption as a function of the distance of the mobile from the base station. This optimization requires that the compression and transmission strategies be adapted to the distance. On the same plot we also show two scenarios in which not only the overall distortion is kept constant at $D_0 = 0.1$ but the coding parameters and consequently the source coding distortion are also fixed. Then in order to keep the total distortion constant the transmit power P_t must increase with distance. As expected, fixed high source bit rate ($R_s = 2$ bits/sample) performs well for small distances, but requires considerably more total power, a factor of two, than the optimized case for large distances. Conversely, a low source bit rate algorithm ($R_s = 0.7$ bits/sample) is better suited for large distances, and the total power dissipation is almost ten times larger than the optimized scenario for small distances.

III. Power optimization for wireless video transmission

In this section we extend our results to a practical scenario and consider a video signal compressed by an H.263 encoder, channel coded by an RS channel code and transmitted over a wireless channel. We assume a slowly fading channel whose coherence time is large, so we only consider the effects of additive noise and path loss. We assume DPSK is the modulation scheme. A block diagram of the system is illustrated in Fig. 4.

In an H.263 encoder [13], each macroblock (containing 16×16 pixels) in a frame can be coded in either the INTRA mode or the INTER mode. In the INTRA mode, a macroblock is directly transformed using Discrete Cosine Transform (DCT) and the DCT coefficients are quantized and entropy coded. In the INTER mode, a macroblock is predicted from a previously coded frame using motion-compensated prediction, and the prediction error is coded using the INTRA mode method. Except the first frame in which all macroblocks are coded in the INTRA mode, a macroblock in other frames is normally coded in the INTER mode, unless the prediction is unsuccessful, or if it is forced to be coded in the INTRA mode for error-resilience purpose. An important encoder parameter is the INTRA rate β, which controls how often a macroblock is forced to be coded in the INTRA mode. The rate control algorithm in the encoder can regulate the bit rate by adjusting the quantization parameter and the coded frame rate (or essentially the frame skipping rate) according to the status of the encoder buffer. From the user perspective, a user can specify the INTRA rate β and the target bit rate R_s, and the encoding frame rate f_s. We denote the distortion introduced at the source encoder when the bit rate is R_s and the INTRA rate is β by $D_s(\beta, R_s)$. Note that $D_s(\beta, R_s)$ corresponding to different values of β gives us different operational rate-distortion curves as in Fig. 1(a), with higher curve corresponding to larger β. This is because a larger β means more macroblocks are coded using the INTRA mode, which yields higher distortion at the same bit rate than the INTER mode.

The total distortion at the decoder D_t is generally higher than D_s due to transmission errors. We define $D_v = D_t - D_s$ as the additional distortion incurred by transmission errors. This additional distortion depends on the channel error characteristics, the channel coder, the transmission power, as well as the error resilience of the source coder. We assume that the bits contained in each RS coded block corresponds to one packet. We further assume a CRC code is inserted to each packet, so that any packet containing un-correctable channel errors can be detected. We use p_L to represent the residual packet loss rate after RS channel decoding, which depends on the code rate r and the transmission energy per bit E_b, for fixed channel parameters, such as noise level and the distance between mobile and destination. The error resilience of the H.263 encoder depends on the INTRA rate β and other error-resilience tools incurred. We will assume that all other error-resilience options are fixed so that the error resilience of the coded bitstream is controlled solely by β. With the above assumptions, D_v can be abstracted as a function of β and p_L only. We will use the models developed in [2] for both $D_s(\beta, R_s)$ and $D_v(\beta, p_L)$, which are briefly described in Section III-A.

To solve the power optimization problem of (1), we also need to model the power consumptions of the video coder, channel coder and the transmitter. One of the contributions of this paper is the development and validation of power consumption models of H.263 and RS coders. These models are described in Section III-B. Using the distortion and power consumption models, the total power consumption at the transmitter can be minimized by adjusting the source and channel coder parameters β, R_s, r, and the transmit energy E_b, for a fixed set of channel parameters. The power savings that result from this optimization and the effect of various system and environment parameters are in Section III-C.

A. Distortion models

A.1 Rate-distortion performance of the source encoder

Stuhlmüller et al. [2] derived a rate distortion model for an H.263 compliant coder based on simulation data. The INTRA update scheme forces a macroblock to be coded in the INTRA mode after every $T-1$ frames [13]. The distortion model derived in [2] is:

$$D_s(\beta, R_s) = \frac{\theta(\beta)}{R_s - R_0(\beta)} + D_0(\beta), \tag{8}$$

where $\beta = \frac{1}{T}$ is the INTRA rate, R_s is the encoding bit rate in kbits/second and D_s is the distortion in terms of MSE per source sample. The functions $\theta(\beta), R_0(\beta), D_0(\beta)$ are given by:

$$\begin{aligned} \theta(\beta) &= \theta_P + \Delta\theta_{IP}\beta, \\ R_0(\beta) &= R_{0P} + \Delta R_{0IP}\beta, \\ D_0(\beta) &= D_{0P} + \Delta D_{0IP}\beta. \end{aligned} \tag{9}$$

The parameters θ_P, $\Delta\theta_{IP}$, R_{0P}, ΔR_{0IP}, D_{0P} and ΔD_{0IP} depend on the spatial detail and the amount of motion in the sequence [2]. For two video sequences considered in this paper, the values of these parameters are provided by Stuhlmüller and are summarized in Table II. As expected, the distortion $D_s(\beta, R_s)$ decreases as the bit rate R_s increases for a fixed INTRA rate β. On the other hand, $D_s(\beta, R_s)$ increases as β increases, for a fixed R_s.

A.2 Channel model

We use an AWGN model and consider the effects of path loss. This is justified for wireless environments for which the terminal is stationary or moving slowly, so the received signal strength can be tracked. Our analysis can easily be extended to faster moving mobiles with realistic speeds by considering the average of p_L over possible attenuations dictated by the fade level or the channel state. With binary DPSK as the modulation scheme, the probability of bit error is $p_e = \frac{1}{2}e^{-\frac{E_{b,rec}}{N_0}}$ as in (3), where $E_{b,rec}$ is the received energy per bit and N_0 is the noise power spectral density. Similar to Section II, the received energy is related to the radiated energy as $E_b = d^\alpha E_{b,rec}$, assuming a fixed fading attenuation of one. Here d denotes the distance between the mobile and the receiver. Then the bit error rate in terms of the radiated energy per bit E_b is:

$$p_e = \frac{1}{2} e^{-\frac{E_b}{d^\alpha N_0}}, \tag{10}$$

A.3 Error rate of the channel code

The RS (n, k) channel code converts every k information symbols into an n-symbol block by appending $n - k$ parity symbols. Any error pattern with less than $t_c = \lfloor \frac{n-k}{2} \rfloor$ symbol errors can be corrected [16].

The effect of transmission errors on the decoded video depends on the residual block error rate p_L, which is the probability that a block cannot be corrected after the channel decoder. It can be calculated as:

$$p_L(r, E_b) = \sum_{l=t_c+1}^{n} p_d(n, l), \tag{11}$$

where the channel code rate $r(= \frac{k}{n})$, and the block error density $p_d(n, l)$ denotes the probability of l symbol errors within a block of n symbols. For a symbol containing m bits,

$$p_d(n, l) = \binom{n}{l} p_s^l (1 - p_s)^{n-l} \tag{12}$$

with $p_s = 1 - (1 - p_e)^m$ being the symbol error probability.

A.4 Distortion at the video decoder

While motion compensation yields significant gains in coding efficiency, any residual transmission error will cause inter-frame error propagation. Hence increasing INTRA rate β also increases the error resilience of the video coder. Stuhlmüller et al. [2] propose a model for distortion introduced by transmission errors:

$$D_v(\beta, p_L) = \sigma_{u0}^2 p_L \sum_{t=0}^{T-1} \frac{1 - \beta t}{1 + \gamma t}, \tag{13}$$

where leakage γ describes the efficiency of loop filtering to remove the introduced error, σ_{u0}^2 describes the sensitivity of the video decoder to an increase in error rate and p_L is the residual block error rate of (11). The parameters γ and σ_{u0}^2 depend on the characteristics of the video sequence and are given for two test sequences in Table II.

B. Power consumption models and measurements

B.1 Power consumption model of the source encoder

In this section, we present a power consumption model for the H.263 encoder and further validate the model using measured power consumption data on a laptop computer running an H.263 encoder software.

For an INTRA macroblock, the computation consists of DCT transform, quantization and entropy coding. The energy consumed for computing DCT coefficients E_{DCT} is determined by the dimension of DCT transform blocks. For block-based H.263 coder, all transform blocks have the same size, hence E_{DCT} is a constant. The DCT coefficients are then divided by quantization step sizes, scanned using a zigzag scan and converted into symbols. Each symbol consists of a run-length of zeros and a non-zero value. These symbols are converted into binary streams using a combination of Huffman coding and fixed length coding. We denote the energy consumed by quantization and the following entropy coding together as E_Q. Obviously, the energy required for quantization is independent of the bit rate, but that for entropy coding may be dependent on the bit rate, because at a low bit rate, more coefficients are quantized into zeros and fewer symbols need to be Huffman coded. But the exact form of the dependency of the energy consumption for entropy coding is not clear. For simplicity, we assume that E_Q is a constant, independent of R_s. We will validate this assumption by measurements. Thus we model the energy consumption of an INTRA macroblock by

$$E_I = E_{DCT} + E_Q. \tag{14}$$

For a macroblock coded in the INTER mode, extra computation E_{ME} is required for motion estimation. Motion estimation entails the search of a best matching macroblock in the reference frame over a certain search range. The computation required and hence energy consumption depend on the search range, the search algorithm and its implementation. We consider the case where the default search range specified by the H.263 standard is used, so that E_{ME} is a constant for a given search algorithm and its implementation. The energy consumed by coding an INTER macroblock is modelled by

$$E_P = E_{DCT} + E_Q + E_{ME}. \tag{15}$$

Because a macroblock is forced to be coded in the INTRA mode after each $T-1$ INTER frames, the average power consumption is:

$$P_s(\beta) = c_s f_s N_{MB} \frac{E_I + (T-1)E_P}{T}, \tag{16}$$

where c_s is a scaling factor which depends on the actual implementation of the coder, f_s is the encoding frame rate and N_{MB} is the number of macroblocks in one frame.

The source encoder power consumption model in (16) can also be written as

$$P_s(\beta) = c_s(a_s + b_s(1-\beta)), \tag{17}$$

where $a_s = f_s N_{MB}(E_{DCT} + E_Q)$, indicating the power consumption required for computing DCT, quantization and entropy coding, $b_s = f_s N_{MB} E_{ME}$, indicating the power required for performing motion estimation.

Equation (17) clearly shows that the power consumption of the source coder decreases linearly with β. On the other hand, when β is high, the source rate R_s increases, hence transmitting the stream consumes more transmission power. This is the tradeoff we will investigate. Our goal is to determine the β which minimizes the overall power consumption. Before we solve the power optimization problem, we validate the proceeding power consumption model by measurement data.

B.2 Verification of the source encoder power consumption model

In order to validate the above model, we measured the power consumption of a public-domain software H.263 video coder [19] running on a SONY VAIO laptop computer with a 450 MHz Pentium III microprocessor. The circuit diagram is illustrated in Fig. 5(a). A Tektronix current probe is used to measure the current (in amps) in the circuit while the voltage is held constant at V_c volts. To eliminate the effect of the power consumption by programs running at the background by the operating system, we first measure the current consumption I_{idle} when no other task is running. Then the current $I_0(t)$ with the H.263 program running is measured. The difference $I(t) = I_0(t) - I_{idle}$ is taken as the actual current level for H.263 coder. Four-second video sequences of "foreman.qcif" and "mother-daughter.qcif" are coded at different INTRA rates β and bit rates R_s. For each set of parameters, and each video sequence, $I_0(t)$ is recorded, and the energy

$$E = \int (I_0(t) - I_{idle})\, V_c \, \mathrm{dt} \quad \text{(joules)}$$

and power

$$P = \frac{E}{4} \quad \text{(watts)}$$

are calculated. [1]

Figures 5(b) and (c) show the results of these measurements for the sequence "foreman.qcif". We observe from these figures that the power is primarily determined by the INTRA rate β. The variation of the power consumption with the bit rate R_s is relatively small and does not have a consistent trend. Assuming $c_s = 1$, we derive constants a_s and b_s from the measurement data using the least squares method. Results are shown in Table III. The power consumption curves corresponding to the model using the fitted parameters are also shown in Figs. 5(b) and (c). Our model, which ignores the influence of R_s on P_s, fits quite well with the average power consumption at different R_s values, when β varies over a large range.

We conducted the same experiment for another video sequence "mother-daughter.qcif", which has significantly less motion than "foreman.qcif". The resulting parameters are also shown in Table III. We see that the values of a_s are roughly equal for both sequences, but b_s is larger for "foreman.qcif". Recall that a_s depends on $E_{DCT} + E_Q$, that is, the energy consumption for computing DCT, quantization, and entropy coding, which is independent of the motion content of a sequence. On the other hand, b_s depends on E_{ME}, the energy consumption for performing motion estimation, which depends on the search algorithm and its implementation. In the H.263 codec software used in our measurement [19], a spiral full-search algorithm is used, which requires more computation for a video containing larger motion. That is why b_s is significantly larger for "foreman.qcif" than for "mother-daughter.qcif". Had a brute-force full search algorithm been used, b_s would have been the same for difference sequences. In general, however, the parameter b_s may be a content-dependent constant, but the parameter a_s should be content-independent.

[1]Although the actual software implementation in our laptop computer took longer than four seconds we divide the total energy by four seconds to get the power consumption, envisioning the situation that a faster computer would be able to accomplish the encoding operation in real-time using the same total energy.

Although the proposed model in (17) is validated only for a software H.263 codec running on a particular computer, we believe the model is general for both software and hardware implementations since it depends on the number of computations, and is applicable to both software and hardware implementations. In essence the model has only two parameters a_s and b_s, which varies with the actual algorithm and its implementation. When the same algorithm (requiring the same amount of computations) is run on different computers, we expect that parameters a_s and b_s will increase or decrease with the same factor, depending on the microprocessor speed. To account for this effect, we introduce the scaling factor c_s, which allows us to use one parameter to reflect the change in the implementation platform for the source encoder.

In [20], the authors reported that a uniform power cost can be associated for all the instructions without loss in accuracy. Hence it is reasonable to assume that the same H.263 algorithm implemented using an application specific integrated circuit (ASIC) or a DSP processor will still follow the model in (17), but the scaling factor c_s will be much smaller. More generally, if different algorithms are used, the relative magnitudes of a_s versus b_s could be different.

B.3 Power consumption of the channel encoder

With an RS(n,k) code, for each group of k symbols, $n-k$ parity symbols are generated, involving $k(n-k)$ operations. Assuming the energy consumption is linearly proportional to the number of operations, we model the energy consumption required for each information bit by

$$E_c = c_c \frac{n-k}{m} = c_c \frac{2t_c}{m} \text{ (joules/bits)}, \tag{18}$$

where c_c is a scaling factor that depends on the actual implementation,$t_c = \lfloor \frac{n-k}{2} \rfloor$. Thus, the power consumption of an RS(n,k) encoder acting on a compressed stream with bit rate R_s bits/second is

$$P_c(r, R_s) = E_c R_s = c_c \frac{n(1-r)R_s}{m} \text{ (watts)}. \tag{19}$$

Using the same set-up for the power consumption measurement for the software H.263 encoder (see Fig. 5(a)), we validate the above model for a public-domain software RS coder [21] with a block size of $n = 255$ symbols, where each symbol consists of $m = 8$

bits, for different values of k. A total of 10,000 iterations are used for each k to make the energy consumption sufficiently large so that it can be measured by our experimental set-up. The measurement data and the model proposed are both shown in Fig. 5(d). We see that the model that assumes a linear relationship between E_c and t_c in (18) fits the measurements very well. We observe that the power consumption of the RS channel coder is about 1% of the source encoder. Hence in our numerical analysis of Section III-C we keep the ratio between the channel encoder and source encoder power consumptions for both software and hardware implementations to be around 1%.

In [22], the same model as in (18) was used to characterize the energy consumption by a hardware RS encoder. Therefore, it is reasonable to assume that by varying c_c, we can use (19) to characterize the power consumption of an RS channel coder on different implementation platforms.

B.4 Power consumption of the transmitter

Let E_b represent the radiated energy per bit at the transmitter antenna. The total transmission power is then given by

$$P_t(r, R_s, E_b) = c_t R_c E_b = c_t \frac{R_s}{r} E_b \text{ (watts)}, \tag{20}$$

where $R_c = \frac{R_s}{r}$ is the total bit rate, and c_t is a scaling factor that translates the radiated energy into the actual power consumption for transmission [17] (including modulation and signal amplifiers) in a wireless device.

C. Power optimization and allocation

We are now ready to formulate the power optimization problem for wireless video transmission. Our goal is to guarantee a total distortion below a specified level D_0 while minimizing the total power consumption due to signal processing and communication at the transmitter. Hence we would like to solve:

Minimize total power consumption P_{tot} subject to total distortion $D_{tot} \leq D_0$.

We note that the above optimization can be seen as an application of the Dynamic Algorithm Transform (DAT) framework of [23], which was proposed as a systematic framework for designing a reconfigurable system. DAT sets up the optimization problem based on: (1) input state space, (2) configuration space, (3) energy models and (4) DSP algorithm performance models.

The state space $\mathcal{S}$ collects all input parameters, which are fixed for a given channel environment, a given video sequence, and the type of source and channel encoders. The state vector $\mathbf{s}$ for our video transmission system includes (1) parameters for the total distortion $\mathbf{s}_{dis}$, (2) parameters for the power consumption models of the source encoder $\mathbf{s}_{s,enc}$, channel encoder $\mathbf{s}_{c,enc}$, (3) parameters of the channel model $\mathbf{s}_{c,para}$, and (3) the scaling factors $\mathbf{s}_{scale}$:

$$\mathbf{s} = [\mathbf{s}_{dis} \quad \mathbf{s}_{s,enc} \quad \mathbf{s}_{c,enc} \quad \mathbf{s}_{c,para} \quad \mathbf{s}_{scale}]. \tag{21}$$

where

$$\begin{aligned}
\mathbf{s}_{dis} &= (\sigma^2_{u0}, \gamma, \theta_P, R_{op}, D_{OP}, \Delta\theta_{IP}, \Delta R_{OIP}, \Delta D_{OIP}), \\
\mathbf{s}_{s,enc} &= (a_s, b_s), \\
\mathbf{s}_{c,enc} &= (n, m), \\
\mathbf{s}_{c,para} &= (\alpha, N_0, \text{distance}), \\
\mathbf{s}_{scale} &= (c_s, c_c, c_t).
\end{aligned}$$

The parameters to be optimized by the adaptation system are described in the configuration space $\mathcal{C}$. For our work, the configuration space $\mathbf{c}$ is a collection of the INTRA rate β, source bit rate R_s, channel code rate r and transmit energy level E_b, i.e.,

$$\mathbf{c} = [r \quad \beta \quad R_s \quad E_b]. \tag{22}$$

The total power consumed by the transmitting mobile device consists of power dissipated by the source encoder, the channel encoder and transmitter. Therefore

$$\begin{aligned}
P_{tot}(\mathbf{c}) &= P_s(\beta) + P_c(r, R_s) + P_t(r, R_s, E_b) \\
&= c_s(a_s - b_s(1-\beta)) + \frac{c_c n R_s (1-r)}{m} + \frac{c_t R_s E_b}{r} \quad \text{(watts)}.
\end{aligned} \tag{23}$$

In our work, we will keep the end-to-end distortion constant. Combining (8) and (13), we find that the total distortion D_{tot} of the video stream due to source compression and residual transmission errors can be expressed as

$$D_{tot}(\mathbf{c}) = D_s(\beta, R_s) + D_v(\beta, p_L)$$

$$= \frac{\theta(\beta)}{R_s - R_0(\beta)} + D_0(\beta) + \sigma_{u0}^2 p_L(r, E_b) \sum_{t=0}^{T-1} \frac{1-\beta t}{1+\gamma t}. \quad (24)$$

Using DAT terminology, our optimization problem is now to find out the optimal operating point in the configuration space

$$\mathbf{c}_{opt} = \text{argmin}_{\mathbf{c}}\{P_{tot}(\mathbf{c}) : D_{tot}(\mathbf{c}) \leq D_0\}. \quad (25)$$

We will consider a finite set of possible values for each variable in the configuration vector, and use full-search to find the optimal operation point. To satisfy the distortion constraint, only three out of the four configuration variables are free. We choose to let $T = \frac{1}{\beta}, r, E_b$ be the free variables. For each combination of T, r, E_b in the configuration space, using the distortion model in (24), we determine R_s so that $D_{tot} = D_0$. Then we evaluate P_{tot} in (23) corresponding to this set of $\{T, r, E_b, R_s\}$. The optimal operating point $\mathbf{c}_{opt}$ is the set that leads to the minimal P_{tot}.

Although the input state space contains many variables, only a few parameters need to be updated frequently. First of all, scaling factors $\mathbf{s}_{scale}$ and the channel encoder parameters $\mathbf{s}_{c,enc}$ can be fixed for a given mobile device with fixed implementations of source and channel encoders. The parameters associated with the source distortion model ($\mathbf{s}_{dis}$) and the source encoder power consumption ($\mathbf{s}_{s,enc}$) depend on the characteristics of the transmitted video. Instead of deriving the actual parameters for each video being transmitted, it is possible to consider only a limited number of video categories, each corresponding to a different application with a different motion magnitude. For example, a video-phone session typically involves low-motion scenes, whereas a live recording of a sports event will contain high motion. Thus we can pre-determine the parameters for each video category and store them in a look-up table. At the beginning of each video transmission session, the appropriate parameters can be uploaded and fixed for the entire session. The only parameters that need to be updated frequently are those reflecting the channel conditions, i.e. $\mathbf{s}_{c,para}$.

In a practical system, we envision implementing the proposed adaptation framework in the following way. After each short time interval (which depends on how fast the channel condition changes), a "power manager" updates the channel parameters $\mathbf{s}_{c,para}$ in the state-space, and then determines the optimal operating point $\mathbf{c}_{opt}$ in terms of T, r, E_b, R_s, using either a full-search method or possibly some fast search algorithms. Alternatively, we can pre-compute the optimal $\mathbf{c}_{opt}$ for a set of possible channel state variables, and store them in a look-up table. Because of the small dimension of the configuration space, even the full-search procedure takes negligible computations, compared to that required for video encoding. Therefore, we have not focused on development of fast algorithms for computing the optimal solution.

C.1 Choice of parameters

We summarize our input space $\mathbf{s}$ and configuration space $\mathbf{c}$ in Table IV. We set the total allowed distortion to be $D_0 = 60$ for an 8 bit per color per sample video source. This corresponds to peak SNR (PSNR) = 30.38 dB, which indicates good but not excellent video quality. We would like to investigate the effect of different noise levels on the level of compression and channel coding. Similar to Section II, we can interpret varying noise levels in terms of the distance of the mobile from the base station.

The scaling factors c_s, c_c and c_t can affect the optimization results significantly. It is known that in today's wireless devices, the power consumed by base-band signal processing and by transmission are roughly the same [18] and the power consumption by the channel coder is significantly smaller than that by source coding. This was also verified by our power measurements of Section III-B. Therefore, we choose c_s, c_t so that P_s and P_t are on the same order of magnitude, and we choose c_c so that P_c is much smaller than P_s and P_t. In this work, we choose $c_s = 0.054c_t$, and $c_c = 5.4 \times 10^{-7}c_t$, where c_t will be determined by the particular device. While the above assumption holds for a hardware implementation, a software implemented source and channel code consume significantly more power. Therefore we also consider c_s and c_c ten times as large in our numerical analysis.

The video to be transmitted is another important factor which could affect the optimization. Fast-moving pictures require more computation for motion estimation than slow-moving pictures, thus requiring more source coding power for the same source coding distortion. This is mainly reflected in the source coding power consumption model parameter b_s. Differences in motion content and scene-change frequency also influence the source distortion and transmission distortion, leading to different parameter values in the models ((8) and (13)). We consider two video sequences: "foreman.qcif" and "mother-daughter.qcif", which have different motion patterns. The sequence "foreman.qcif" contains fast-moving objects and scene changes, as well as changes due to camera pan and zoom; whereas "mother-daughter.qcif" contains only localized motions of heads and shoulders.

C.2 Results

Table V summarizes the three scenarios that we examine, which vary in video source, and assumed implementation platform for the source and channel coders. Note the power is normalized with respect to c_t. In Scenario A, we illustrate our numerical analysis for "foreman.qcif" implemented in hardware. We show in Fig. 6 the numerical results for two different fixed effective noise levels for Scenario A. In Figure 6(a), we consider a small distance between the mobile and the base station, modelling a scenario in which the noise level is low. Fig. 6(b) shows the total power and transmit power for a larger distance, hence larger noise power. Each point in the solid curve is obtained by fixing E_b, and finding the parameters r, β, R_s that minimize the total power consumption while reaching the desired D_{tot}. As expected, for both channel conditions, there is an intermediate value of E_b that leads to minimal total power consumption. Comparing Fig. 6(a) and Fig. 6(b), we see that when the distance increases, the optimal strategy uses higher E_b to guarantee the same total distortion, resulting in a higher total power consumption.

Table VI lists the optimal configuration space $\mathbf{c}_{opt}$ that yield lowest P_{tot} for a given distance. Fig. 7(a) shows the minimum total power consumption as a function of the distance between the mobile and the base station for Scenario A. As expected, E_b always increases with the distance. As for other parameters, the trend is the same as the abstract scenario of Section II for relatively large distance. Since transmission power is dominant, we would like to send fewer bits and have more error correction. This forces the video coder to decrease β and R_s, and the channel coder to add more redundancy by decreasing r. However, this trend is reversed for small distances. Now source coding is more costly in terms of power. Since β is the dominant factor in the compression power, the optimization keeps it at the largest value of 50%. Fig. 7(a) also illustrates the power consumption of two scenarios in which (r, β, R_s) are fixed and only E_b is allowed to vary with distance to keep the total distortion constant at D_0. As expected, fixed high R_s performs well when the mobile is close to the base station. Within the distance range considered in this paper, the power consumed by the fixed high R_s scheme is 6.6 times greater than the optimized scenario at large distances. The reverse is true for small R_s and the total power consumption of a fixed small R_s scheme is 1.2 times greater the optimal one at small distances.

Scenario B illustrates the minimum power consumption for “mother-daughter.qcif”. This video sequence differs from “foreman.qcif” in motion content. Results are shown in Fig. 7(b) and the trends are similar to 7(a). Comparing Fig. 7(b) with Fig. 7(a) and using Table VI, we argue that the motion content of the video sequence does not have too much influence on the optimal operation points, except the required bit rate. This potentially simplifies the operation of the “power manager”. For slow-moving pictures, one can operate at lower source bit rate R_s because there are more zeros after quantization for slow-moving video sequence. Overall a much smaller power consumption is required in comparison to the fast-moving sequence “foreman.qcif”.

Finally, Fig. 7(c) shows the optimization results for Scenario C, where we have set c_s and c_c to larger values to model a situation where source and channel coding are performed in software. As explained before we assume the source and channel coders in software require ten times more power than hardware implementation. Video sequence is the same as Scenario B. Comparing with Fig. 7(b) we observe that since compression is more costly, we now need more INTRA coded macroblocks to reduce compression power, resulting in higher β. This results in higher error resilience in the source coder and enables the channel code rate r to increase. The total power consumption is higher than the hardware implementation and is more pronounced for small distances where signal processing power dominates. The power savings are relatively small in this scenario since signal processing power is dominant and less space is left for optimization.

IV. Conclusion and future work

This research provides a framework for optimizing the total power consumption of a mobile transmitter subject to a given end-to-end distortion. The total power consumption incorporates the source compressor, channel coder and transmitter power. We investigate two scenarios: An abstract class of sources and channels (Gauss-Markov source transmitted over an AWGN channel) and a practical H.263 wireless video transmission. For both system models, the optimization problems are formulated based on models of distortions and power consumptions. One of our contributions is the derivation and validation of a power consumption model for H.263 video coder and RS coder. From our numerical studies we observe that for both the abstract and practical source/channel models optimum distortion-power operating points are dependent on the distance of the mobile from the base station. For large distances since transmission of each bit is costly, one needs to compress the source as much as possible. These highly compressed bits are more error prone, so stronger channel codes are needed for proper error control.

In order to gain more insight for the practical H.263 video transmission system, we examine three scenarios and illustrate the effects of distance, video motion content and implementation platform on the choice of the operating parameters of the source and channel encoders and on the optimal power consumption of the transmitter. As the distance gets large, one needs to use higher transmit energy to combat the channel errors. Consequently the total power consumption increases with the distance. Our study also shows that the content of the video sequence does not have too much influence on the optimal parameters for the source encoder, channel encoder and transmitter, except the required bit rate to reach a desired distortion. A video with more significant motion requires a higher bit rate and a higher total total power consumption, especially at larger distances.

Implementation platform of signal processing algorithms (hardware versus software) also affect the power optimization. Software encoders consume significantly more power than hardware encoders. Hence for software implementation, the INTRA rate β is higher to reduce power consumption at the source coder, which in turn leads to higher code rate r. The total power consumption is much higher than that of hardware implementation.

Although this paper mainly focused on transmit power, other scenarios include optimizing the receiver power consumption (for cellular downlink) or optimizing transmit and receive energies jointly (for peer-to-peer communications, or for no base station transcoding). At the transmitter the source coder dissipates most of the signal processing power, whereas at the receiver channel decoder is the bottleneck. Another possible extension is to multiuser scenarios, where interference from nearby users affects the channel error rate of the mobile and one needs to jointly optimize the total powers of all the users.

Acknowledgment

The authors would like to thank Ramesh Karri, Piyush Mishra and Wanyee Lo for their help in measuring the power consumption of the source encoder. We also would like to thank Dr. Klaus Stuhlmüller for providing details of the distortion models used in this paper and model parameters for our test sequences.

References

[1] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal Inter/Intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966–976, June 2000.

[2] K. Stuhlmüller, N. Färber, M. Link and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.

[3] T. H. Meng, A. C. Hung, E. K. Tsern, and B. M. Gordon, "Low-power signal processing system design for wireless applications," *IEEE Personal Communications*, pp. 20–31, June 1998.

[4] A. A. Abidi, "Low-power radio-frequency IC's for portable communications," in *Proceedings of the IEEE*, April 1995, pp. 544–569.

[5] T. Lan and A. H. Tewfik, "Power optimized mode selection for H.263 video coding and wireless communications," in *Proceedings of International Conference on Image Processing*, 1998, vol. 2, pp. 113 –117.

[6] S. Appadwedula, M. Goel, N. R. Shanbhag, D. L. Jones, and K. Ramchandran, "Total system energy minimization for wireless image transmission," *Journal of VLSI Signal Processing Systems*, vol. 27, no. 1/2, pp. 99–117, February 2001.

[7] Q. Zhang, Z. Ji, W. Zhu, and Y. Q. Zhang, "Power-minimized bit allocation for video communication over wireless channels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 398–410, June 2002.

[8] Z. Ji, Q. Zhang, W. Zhu, and Y. Q. Zhang, "Joint power control and source-channel coding for video communication over wireless networks," in *Proceedings of IEEE Vehicular Technology Conference*, Oct 2001, pp. 1658–1662.

[9] K. Cheung and L. M. Po, "A hierarchical block motion estimation algorithm using partial distortion measure," in *Proceedings of International Conference on Image Processing*, April 1997, vol. 3, pp. 606–609.

[10] E. Erkip, Y. Wang, D. Goodman, Y. Wu, and X. Lu, "Energy efficient coding and transmission," in *Proceedings of IEEE Vehicular Technology Conference*, May 2001, vol. 2, pp. 1444–1448.

[11] X. Lu, Y. Wang, and E. Erkip, "Power efficient H.263 video transmission over wireless channels," in *Proceedings of International Conference on Image Processing*, Sept 2002, vol. 1, pp. 533–536.

[12] Y. Wang, J. Ostermann, and Y. Q. Zhang, *Video Processing and Communications*, Prentice Hall, 2002.

[13] "ITU-T Recommendation H.263: Video Coding for Low Bit Rate Communication," 1998.

[14] "Dream-it: Dynamically reconfigurable energy aware multimedia information terminals," http://eeweb.poly.edu/dream-it.

[15] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.

[16] J. G. Proakis, *Digital Communications*, McGraw Hill, New York, NY, 2001.

[17] G. Hau and T. B. Nishimura and N. Iwata, " A highly efficient linearized wide-band CDMA handset power amplifier based on predistortion under various bias conditions," *IEEE Transactions On Microwave Theory and Techniques*, vol. 49, pp. 1194–1201, June 2001.

[18] S. Ohr, "Batteries take on wireless challenge," *EE Times*, October 6, 2000.

[19] University of British Columbia, "H.263+ codec," available at http://dspftp.ece.ubc.ca.

[20] M. Singh and V. K. Prasanna, *System-Level Energy Tradeoffs for Collaborative Computation in Wireless Networks*, Kluwer Academic Publishers, 2002.

[21] P. Karn, ," http://www.ka9q.net/code/fec/.

[22] M. Goel, S. Appadwedula, N. R. Shambhag, K. Ramchandran, and D. L. Jones, "A low-power multimedia communication system for indoor wireless applications," in *1999 IEEE Workshop on Signal Processing Systems*, 1999, pp. 473 –482.

[23] M. Goel and N. R. Shanbhag, "Dynamic Algorithm Transformations (DAT): A systematic approach to low-power reconfigurable signal processing," *IEEE Transactions On VLSI*, vol. 7, pp. 463–476, Dec 1999.

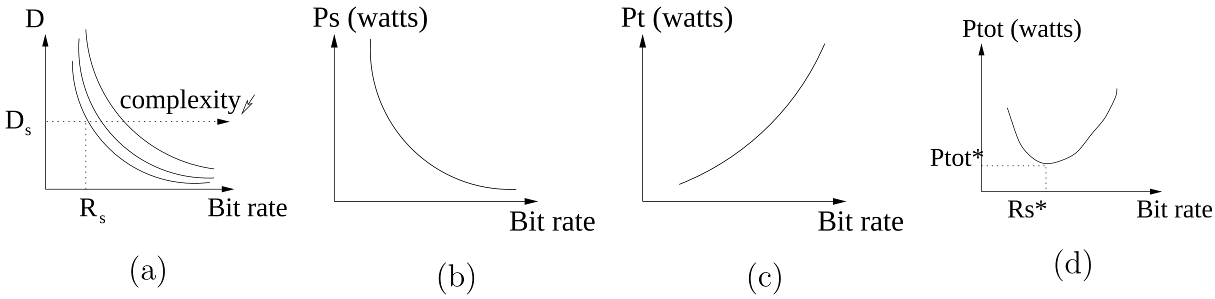


Fig. 1. Minimizing total power for fixed end-to-end distortion. (a) The operational rate-distortion curve; (b) signal processing power versus bit rate; (c) transmission power versus bit rate; (d) total power consumption versus bit rate.

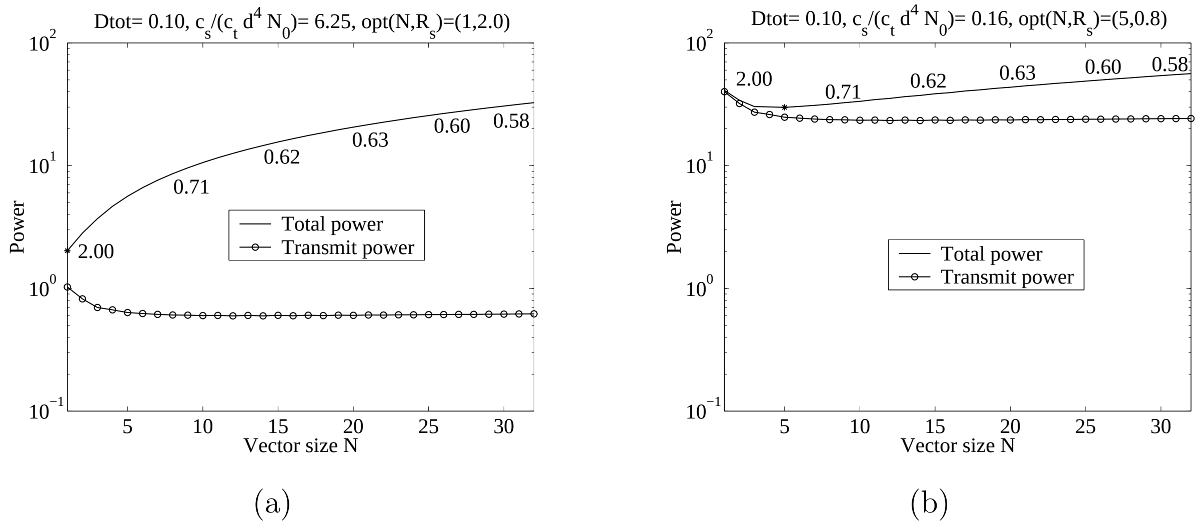


Fig. 2. Total power minimization for transform coding and AWGN channel for two channel conditions. The powers are normalized to $c_s f_s$. Numbers along curves (a) when the mobile is close to the receiver and (b) when the mobile is far away from the receiver represent optimal compression rates R_s for a given N.

TABLE I

Optimal operating points (N, R_s) as a function of distance for transform coding and AWGN channel.

Distance (m)	Vector size N	Bit rate R_s (bits/sample)
30	1	2
90	2	1.5
180	5	0.8
330	12	0.67

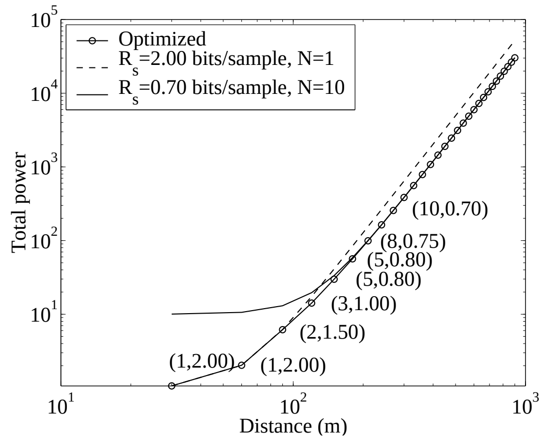


Fig. 3. Total power minimization for transform coding and AWGN channel. The powers are normalized to $c_s f_s$. Numbers along curve represent optimizing (N, R_s) pairs for a given distance.

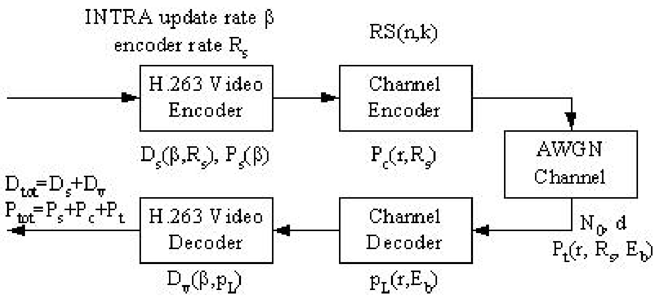


Fig. 4. Wireless video transmission system.

TABLE II

Parameter values of distortion models $\mathbf{s}_{dis}$

	θ_P	$\Delta\theta_{IP}$	R_{0P}	ΔR_{0IP}	D_{0P}	ΔD_{0IP}	γ	σ_{u0}^2
"foreman.qcif"	4258.9	11311.0	4.4	40.2	-2.3	-27.4	2.6	4322.4
"mother-daughter.qcif"	912.4	3371.9	-5.0	77.0	-0.6	-5.1	0.5	183.0

We would like to thank Dr. Klaus Stuhlmüller for the data.

TABLE III

Parameters for the source encoder power consumption $\mathbf{s}_{s,enc}$

	a_s (watts)	b_s (watts)
"foreman.qcif"	22.20	17.60
"mother-daughter.qcif"	23.51	12.34

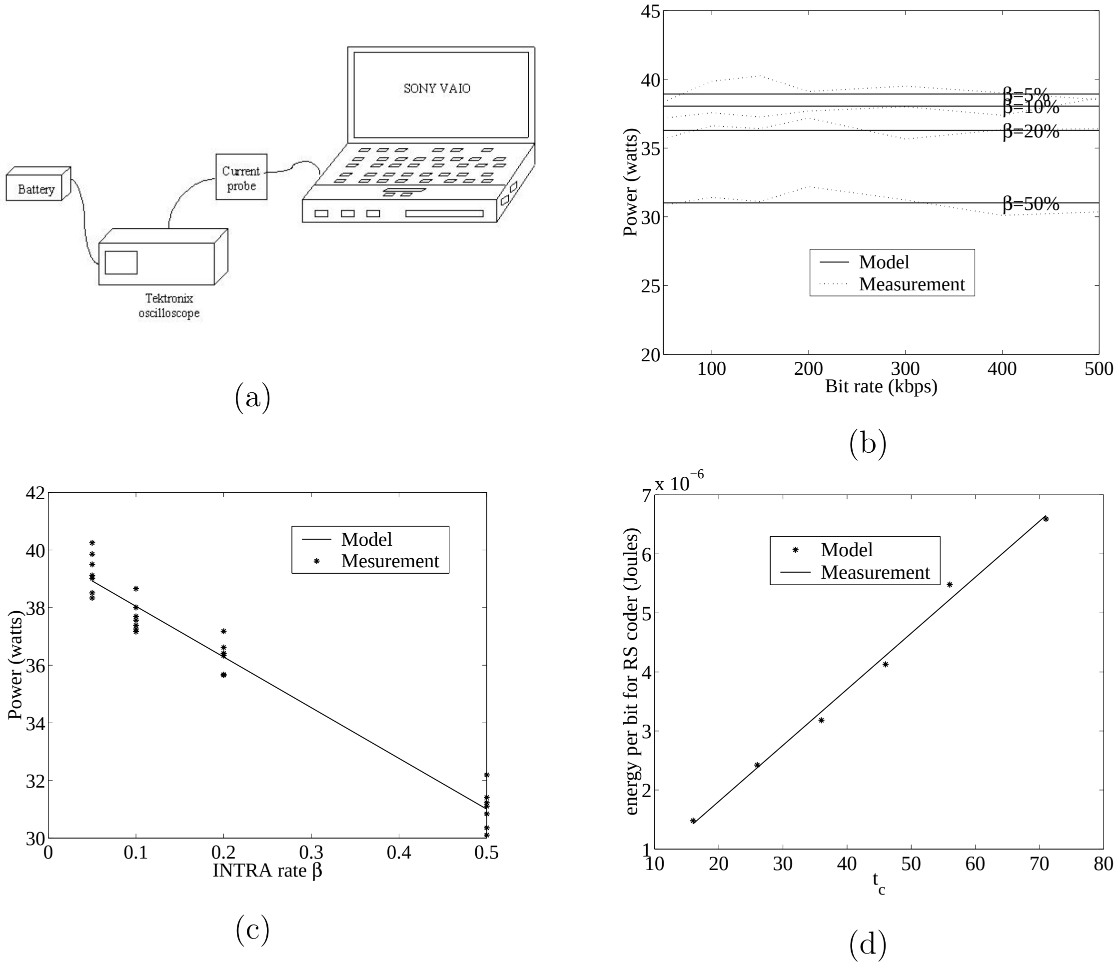


Fig. 5. Verification of source power consumption model. We have the software running on a SONY VAIO laptop with a 450 MHz Pentium III processor. Current is measured for different settings of H.263 and RS coder at a constant voltage. (a) Power measurement set-up; (b) Power consumption versus bit rate for different INTRA rates of an H.263 coder; (c) Power consumption versus INTRA rate for an H.263 coder; (d) Power consumption for an RS coder.

TABLE IV

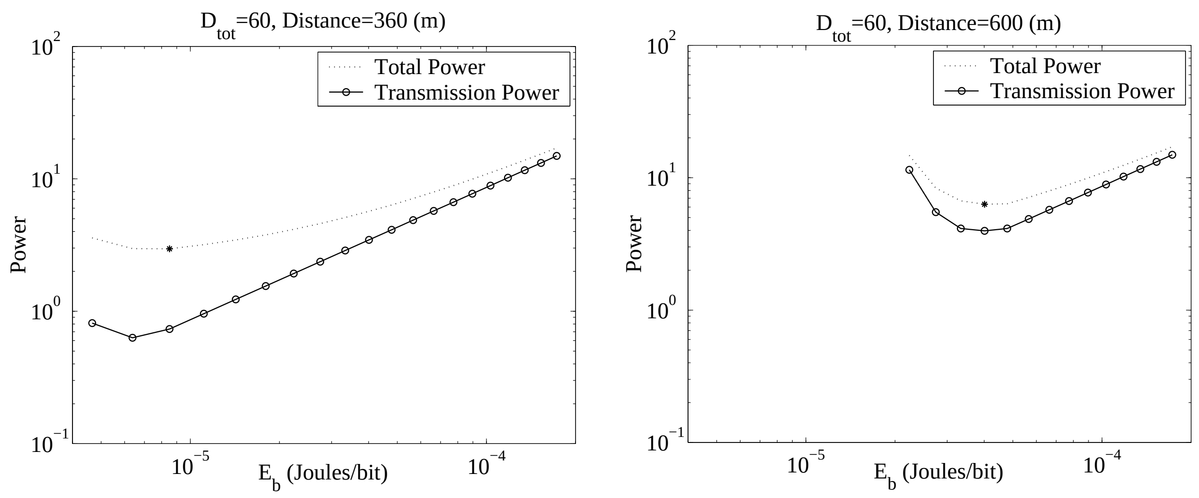
PARAMETERS FOR WIRELESS VIDEO TRANSMISSION SYSTEM

input space $\mathbf{s}$		
$\mathbf{s}_{dis}$	parameters of distortion models for two sequences, see Table II	
$\mathbf{s}_{s,enc}$	parameters of source power consumption models for two sequences, see Table III	
$\mathbf{s}_{c,enc}$	m : number of bits in a symbol	8
	n : block size (symbols)	222
$\mathbf{s}_{c,para}$	α : propagation constant	3.6
	N_0 : noise power spectral density	1×10^{-15} watts/Hz
	distance	30-900 m
$\mathbf{s}_{scale}$	different parameters for hardware and software implementations	
configuration space $\mathbf{c}$		
$T = \frac{1}{\beta}$	INTRA interval	{33, 25, 16, 12, 9, 5, 3, 2}
r	code rate	{0.18, 0.27, 0.36, 0.45, 0.55, 0.64, 0.73, 0.82, 0.91}
E_b	transmit energy per bit	8.3×10^{-10}-1.7×10^{-4} joules/bit, 30 steps
R_s	source bit rate	adjusted to keep the end-to-end distortion

TABLE V

TESTING SCENARIOS

Scenario	Sequence	Implementation	motion content	associated figure
A	“foreman.qcif”	hardware	fast motion	Fig. 7(a)
B	“mother-daughter.qcif”	hardware	slow motion	Fig. 7(b)
C	“mother-daughter.qcif”	software	slow motion	Fig. 7(c)



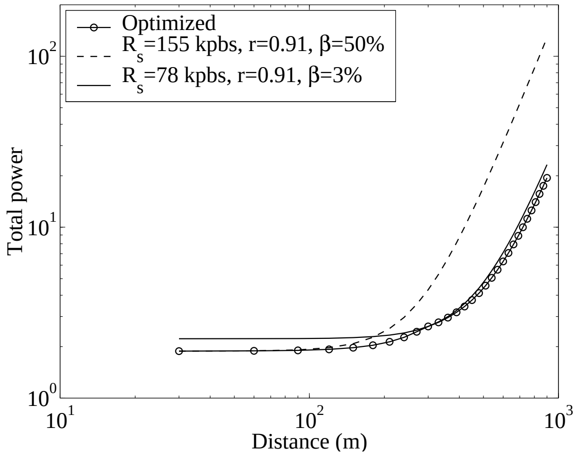
(a) (b)

Fig. 6. Total power and transmit power versus E_b for test sequence "foreman.qcif" for hardware implementation, showing an optimal operation point exists (a) when the mobile is close to the base station; (b) when the mobile is further away from the base station. The powers are normalized to c_t

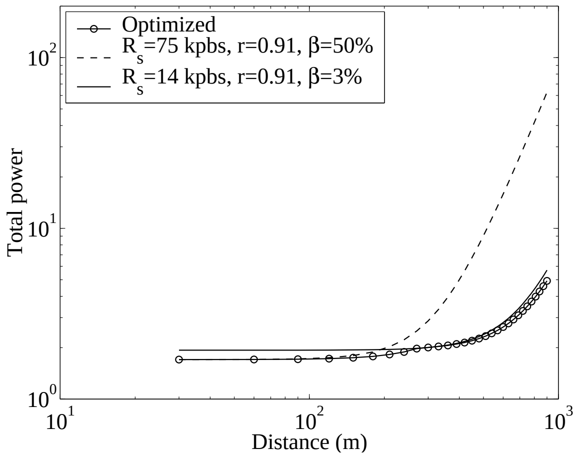
TABLE VI

Optimal operation points $\mathbf{c}_{opt}$ as a function of distance for H.263 video transmission.

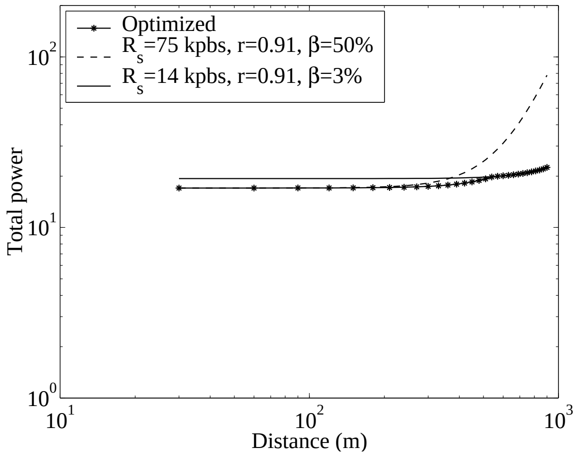
Scenario	Distance (m)	code rate r	INTRA rate β (%)	Bit rate R_s (kbps)	E_b (joules/bit)
A	30	0.91	50	154.86	1.01×10^{-8}
	300	0.91	3	78.46	4.66×10^{-6}
	540	0.82	3	81.06	2.75×10^{-5}
	870	0.82	3	81.06	1.73×10^{-4}
B	30	0.91	50	74.68	1.01×10^{-8}
	300	0.91	3	14.04	4.66×10^{-6}
	540	0.82	3	14.10	2.75×10^{-5}
	870	0.82	3	14.10	1.73×10^{-4}
C	30	0.91	50	74.68	1.01×10^{-8}
	300	0.91	50	74.68	4.66×10^{-6}
	540	0.91	50	74.68	3.33×10^{-5}
	870	0.91	3	14.19	1.73×10^{-4}



(a)



(b)



(c)

Fig. 7. Total power minimization for wireless video transmission for (a) Scenario A; (b) Scenario B; (c) Scenario C of Table V. The powers are normalized to c_t.