C-DMRC: Compressive Distortion-Minimizing Rate Control for Wireless Multimedia Sensor Networks

Scott Pudlewski, Tommaso Melodia, Arvind Prasanna
Wireless Networks and Embedded Systems Laboratory
Department of Electrical Engineering
State University of New York (SUNY) at Buffalo
e-mail: {smp25, tmelodia, ap92}@buffalo.edu

***Abstract*—This paper investigates the potential of the compressed sensing (CS) paradigm for video streaming in Wireless Multimedia Sensor Networks. The objective is to co-design a low-complexity video encoder based on compressed sensing and a rate-adaptive streaming protocol for wireless video transmission. The proposed rate control scheme is designed with the objectives to maximize the received video quality at the receiver and to prevent network congestion while maintaining fairness between multiple video transmissions. Video distortion is represented through analytical and empirical models and minimized based on a new cross-layer control algorithm that jointly regulates the video encoding rate and the channel coding rate at the physical layer based on the estimated channel quality. The end-to-end data rate is regulated to avoid congestion while maintaining fairness in the domain of *video quality* rather than data rate. The proposed scheme is shown to outperform TCP-Friendly Rate Control (TFRC).**

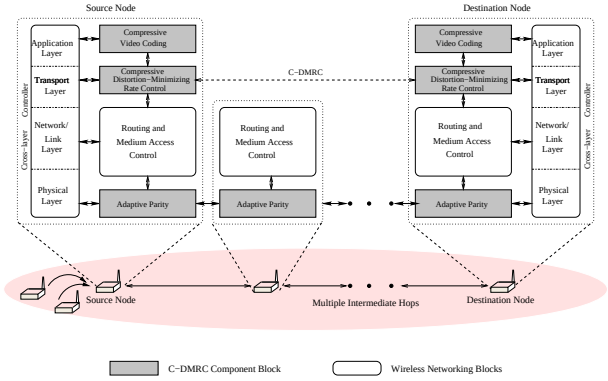

Fig. 1. Architecture of C-DMRC system.

I. Introduction

Wireless Multimedia Sensor Networks (WMSN) [1] are self-organizing wireless systems of embedded devices deployed to retrieve, distributively process in real-time, store, correlate, and fuse multimedia streams originated from heterogeneous sources. WMSNs will enable new applications including surveillance, storage and subsequent retrieval of potentially relevant activities, and person locator services.

In recent years, there has been intense research and considerable progress in solving numerous wireless sensor networking challenges. However, the key problem of enabling real-time quality-aware video streaming in large-scale multihop wireless networks of embedded devices is still open and largely unexplored [1]. There are two key shortcomings in systems based on sending predictively encoded video (e.g., MPEG-4 Part 2, H.264/AVC [2], [3], [4], H.264/SVC [5]) through a layered wireless communication protocol stack, i.e., *encoder complexity* and *low resiliency to channel errors.*

- **Encoder Complexity.** Predictive encoding requires complex processing algorithms, which lead to high energy consumption [1], [6]. Instead, new video encoding paradigms are needed to reverse the traditional balance of complex encoder and simple decoder, which is unsuited for embedded video sensors. Recently developed *distributed video coding* [7] algorithms (aka Wyner- Ziv coding [8]) exploit the source statistics at the decoder, thus shifting the complexity at this end. While promising for WMSNs [1], most practical Wyner-Ziv codecs require end-to-end feedback from the decoder [9], [10], which introduces additional overhead and delay. Furthermore, gains demonstrated by practical distributed video codecs are limited to 2-5 dBs PSNR [9], [10]. Distributed video encoders that do not require end-to-end feedback have been recently proposed [11], but at the expense of a further reduction in performance.
- **Limited Resiliency to Channel Errors**. Ideally, when one bit is in error, the effect on the reconstructed video should be unperceivable, with minimal overhead. In addition, the perceived video quality should gracefully and proportionally degrade with decreasing channel quality.

In this paper, we show how a new cross-layer optimized communication protocol stack based on the recently proposed compressed sensing (CS) paradigm [12], [13], [14], [15] can offer a viable solution to the aforementioned problems. Compressed sensing (aka "compressive sampling") is a paradigm that allows the faithful recovery of sparse signals from $M << N$ measurements where N is the number of samples required by the Nyquist sampling theorem. The CS paradigm can

offer an alternative to traditional video encoders by enabling imaging systems that sense and compress data simultaneously and much faster, *at very low computational complexity for the encoder*. Image coding and decoding based on CS has been recently explored [16], [17]. So-called single-pixel cameras that can operate efficiently across a much broader spectral range (including infrared) than conventional silicon-based cameras have also been proposed [18]. However, transmission of CS images and video streaming in wireless networks, and their statistical traffic characterization, are substantially unexplored.

We hereby introduce the Compressive Distortion-Minimizing Rate Control (C-DMRC), a new distributed cross-layer control algorithm that jointly regulates the CS sampling rate, the data rate injected in the network, and the rate of a simple parity-based channel encoder to maximize the received video quality over a multi-hop wireless network with lossy links. The cross-layer architecture of our proposed integrated congestion control and video transmission scheme is shown in Fig. 1. By jointly controlling the compressive video coding at the application layer, the rate at the transport layer, and the adaptive parity at the physical layer, we can leverage information at all three layers to develop an integrated optimal congestion-avoiding and distortion-minimizing system. Our work makes the following contributions:

- **Distortion-Based Rate Control.** C-DMRC leverages the *estimated received video quality* as the basis of the rate control decision. The transmitting node alters the quality of the transmitted video directly rather than controlling the data rate. By controlling congestion in this way, fairness in the quality of the received videos is maintained even over videos with very different compression ratios.
- **Rate Change Aggressiveness Based on Video Quality.** Based on this CS architecture, we develop a system where nodes adapt the *rate of change* of their transmitted video quality based on the impact that a change in the transmission rate will have on the received video quality. This means that the rate controller uses the information about the estimated received video quality directly in the rate control decision. If the sending node estimates that the received video quality is very high, it will be less likely to increase the rate dramatically, even if the RTT values indicate that the congestion in the network would allow the rate increase. Conversely, if a node is sending poor-quality video, it will gracefully decrease its data rate, even if the RTT indicates a congested network.
- **Video Transmission Using Compressive Sampling.** In our previous work [19], the rate control used the quality level of the MPEG encoded video to estimate the received video quality. In this work, the concept is taken further by using compressed sensed video, rather than traditional MPEG encoded video. Also, the rate control algorithm is refined to use the rate-distortion curve parameters directly, rather than using a parameterized version of the index of the MPEG encoding parameters.

The remainder of this paper is structured as follows. In Section II, we discuss related work. In Section III we introduce the C-DMRC system architecture. In Section IV, we introduce the proposed video encoder based on compressed sensing (CSV). In Section V, we introduce the rate control system. Section VI introduces an adaptive parity channel encoder. Finally, the performance results are presented in Section VII, while Section VIII we draw the main conclusions and discuss future work.

II. Related Work

The most common rate control scheme is the well-known transmission control protocol (TCP) [20][21][22]. It is well known that because of the additive increase/multiplicative-decrease algorithm used in TCP, the rate that it determines varies too quickly for high-quality video transfer [23]. In addition, TCP assumes that the main cause of packet loss is congestion [24]. However, in wireless networks channel errors must be taken into account if an accurate prediction of the network congestion is needed.

These considerations have led to a number of equation-based rate control schemes. Equation-based rate control analytically regulates the transmission rate of a node based on measured parameters such as the number of lost packets and the round trip time (RTT) of the data packets. Two examples of this are the TCP-Friendly Rate Control (TFRC) [24], which uses the throughput equation of TCP Reno [20], and the Analytical Rate Control (ARC) [25]. Both of these schemes attempt to determine a source rate that is fair to any TCP streams that are concurrently being transmitted in the network. However, in a WMSN, priority must be given to the delay-sensitive flows at the expense of other delay-tolerant data. Therefore, both TCP and ARC result in a transmission rate that is more conservative than the optimal rate. For this reason, in an effort to optimize resource utilization in resource-constrained WMSNs, our scheme does not take TCP fairness into account.

Recent work has investigated the effects of packet loss and compression on video quality. In [26], the authors analyze the video distortion over lossy channels of MPEG-encoded video with both inter-frame coding and intra-frame coding. A factor β is defined as the percentage of frames that are an intra-frame, or I frame, i.e., a frame that is independently coded. The authors then derive the value β that optimizes distortion at the receiver. The authors of [26] investigates optimal strategies to transmit video with minimal distortion. However, the authors assume that the I frames are received correctly, and that the only loss is caused by the inter-coded frames. In this paper, we assume that any packet can be lost, and attempt to use CS and adaptive parity in order to combat these losses and increase the received video quality.

QoS for video over the Internet has been studied in [27] and [28]. Both of these works deal with QoS of video over the Internet in a TCP or TCP-Friendly manner. In general, a WMSN will not be directly connected to the Internet, so following these assumptions will result in significant underestimation of the available video quality.

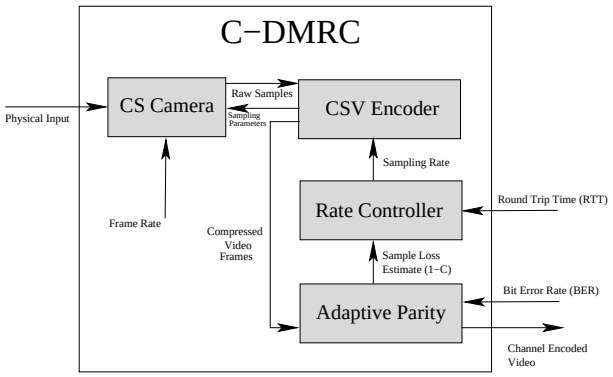

Fig. 2. Architecture of the C-DMRC video rate control system.

Finally, several recent papers take a preliminary look at video encoding using compressed sensing [29], [30], [31]. Our work is different in the following sense: (i) we only use information that can be obtained from a single-pixel camera [18] and do not use the original image in the encoding process at the transmitter. Hence, C-DMRC is compatible with direct detection of infrared or terahertz wavelength images, along with the ability to compress images during the detection process, avoiding the need to store the entire image before it is compressed; (ii) more importantly, we look at the problem from a networking perspective, and consider the effect of joint rate control at the transport layer, video encoding, and channel coding to design an integrated system that maximizes the quality of CS video transmitted over wireless links.

III. System Architecture

In this section, we describe the overall architecture of the compressive distortion-minimizing rate controller (C-DMRC). The system takes a sequence of images at a user-defined number of frames per second and wirelessly transmits an encoded video, where the encoding is done using compressed sensing. The end-to-end RTT is measured to perform congestion control for the video within the network, and the BER is measured/estimated to provide protection against channel losses. This system combines functionalities of the application layer, the transport layer and the physical layer to deliver video through a multi-hop wireless network to maximize the received video quality while accounting for network congestion and lossy channels. As shown in Fig. 2, there are four main components to the system.

A. CS Camera

This is the system where the compressed sensing image capture takes place. The details of compressed sensing are discussed in detail in Section IV-A1. The camera assumed for use in this system can be either a traditional CCD or CMOS imaging system, or a single pixel camera as discussed in [18]. In the latter case, the samples of the image are directly obtained by measuring the intensity of a random sample of small portions of the image, and summing the intensity through the use of a photodiode. The samples generated are then passed to the video encoder.

B. CSV Video Encoder

The CSV video encoder is discussed in Section IV-B. The encoder takes the raw samples from the camera and generates compressed video frames. The compression is based on the temporal correlation between frames. The number of samples, along with the sampling matrix (i.e. which pixels are combined to create each sample) are determined at this block. The number of samples, or *sampling rate*, is based on input from the C-DMRC block, while the sampling matrix is chosen so that the sender and receiver are both using the same sampling matrix for a given video stream.

C. Rate Controller

The C-DMRC block takes as input the end-to-end RTT of the previous packets and the estimated sample loss rate to determine the optimal sampling rate for the video encoder. This sampling rate is then fed back to the video encoder. The rate control law, which is designed to maximize the received video quality while preserving fairness among competing videos, is described in detail Section V. The CS sampling rate determined by the C-DMRC block is chosen to provide the optimal received video quality across the entire network, which is done by using the RTT to estimate the congestion in the network along with the input from the adaptive parity block to compensate for lossy channels.

D. Adaptive Parity

The Adaptive Parity block uses the measured or estimated sample error rate of the channel in order to determine a parity scheme for encoding the samples, which are input directly from the video encoder. The Adaptive Parity scheme is described in Section VI.

IV. CS Video Encoder (CSV)

In this section, we introduce the video encoder component of the compressive distortion-minimizing rate control system.

A. Video Model

1) Compressed Sensing Preliminaries: We consider an image signal represented through a vector $\mathbf{x} \in R^N$, where N is the vector length. We assume that there exists an invertible $N \times N$ transform matrix $\mathbf{\Psi}$ such that

$$\mathbf{x} = \mathbf{\Psi}\mathbf{s} \tag{1}$$

where $\mathbf{s}$ is a K-sparse vector, i.e., $||s||_0 = K$ with $K < N$, and where $|| \cdot ||_p$ represents p-norm. This means that the image has a sparse representation in some transformed domain, e.g., wavelet. The signal is measured by taking $M < N$ measurements from linear combinations of the element vectors through a linear measurement operator $\mathbf{\Phi}$. Hence,

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s} = \tilde{\mathbf{\Psi}}\mathbf{s}. \tag{2}$$

We would like to recover $\mathbf{x}$ from measurements in $\mathbf{y}$. However, since $M < N$ the system is underdetermined. Hence, given a

solution $\mathbf{s^0}$ to (2), any vector $\mathbf{s}^*$ such that $\mathbf{s}^* = \mathbf{s^0} + \mathbf{n}$, and $\mathbf{n} \in \mathcal{N}(\tilde{\mathbf{\Psi}})$ (where $\mathcal{N}(\tilde{\mathbf{\Psi}})$ represents the null space of $\tilde{\mathbf{\Psi}}$), is also a solution to (3). However, it was proven in [13] that if the measurement matrix $\mathbf{\Phi}$ is sufficiently incoherent with respect to the sparsifying matrix $\mathbf{\Psi}$, and K is smaller than a given threshold (i.e., the sparse representation $\mathbf{s}$ of the original signal $\mathbf{x}$ is "sparse enough"), then the original $\mathbf{s}$ can be recovered by finding the sparsest solution that satisfies (2), i.e., the sparsest solution that "matches" the measurements in $\mathbf{y}$. However, the problem above is in general NP-hard [32]. For matrices $\tilde{\mathbf{\Psi}}$ with sufficiently incoherent columns, whenever this problem has a sufficiently sparse solution, the solution is unique, and it is equal to the solution of the following problem:

$$P_1 : \ minimize \, ||s||_1$$

$$subject \, to : ||\mathbf{y} - \tilde{\mathbf{\Psi}}\mathbf{s}||_2^2 < \epsilon, \tag{3}$$

where ϵ is a small tolerance. Note that problem P_1 is a convex optimization problem [33]. The reconstruction complexity equals $O(M^2N^{3/2})$ if the problem is solved using interior point methods [34]. Although more efficient reconstruction techniques exist [35], the framework presented in this paper is independent of the specific reconstruction method used.

2) Frame Representation: We represent each frame of the video by 8-bit intensity values, i.e., a grayscale bitmap. To satisfy the sparsity requirement of CS theory, the wavelet transform [36] is used as a sparsifying base. A conventional imaging system or a single-pixel camera [18] can be the base of the imaging scheme. In the latter case, the video source only obtains random samples of the image (i.e., linear combinations of the pixel intensities). In our model, the image can be sampled using a scrambled block Hadamard ensemble [37]

$$\mathbf{y} = \mathbf{H_{32}} \cdot \mathbf{x}, \tag{4}$$

where $\mathbf{y}$ represents image samples (measurements), $\mathbf{H_{32}}$ is the 32×32 Hadamard matrix and $\mathbf{x}$ the matrix of the image pixels. The matrix $\mathbf{x}$ has been randomly reordered and shaped into a $32 \times \frac{N}{32}$ matrix where N is the number of pixels in the image. Then M samples are randomly chosen from $\mathbf{x}$ and transmitted to the receiver. The receiver then uses the M samples along with the randomization patterns for both randomizing the pixels into $\mathbf{x}$ and choosing the samples out of x to be transmitted (both of which can be decided before network setup) and recreates the image solving P_1 in (3) through a suitable algorithm, e.g., GPSR[1] [38], StOMP [39].

B. CS Video Encoder (CSV)

The CSV video encoder uses compressed sensing to encode video by exploiting the spatial and temporal redundancy within the individual frames and between adjacent frames, respectively.

[1]GPSR is used for image reconstruction in the simulation results presented in this paper.

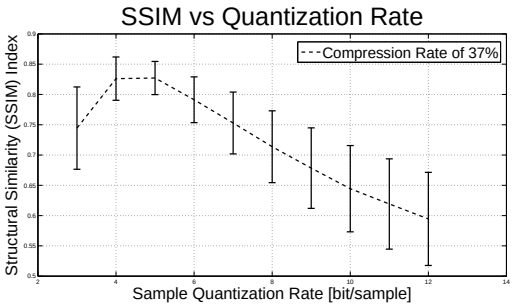

Fig. 3. Structural Similarity (SSIM) Index [40] for Images with a Constant Bit Rate of 37% of the Original Image Size for Varying Quantization Levels.

1) Intra-frame (I) Encoding: As stated above, each of the I frames are encoded individually, i.e., as a *single image* that is *independent of the surrounding frames*. Two variables mainly affect the compression of I frames; the sample quantization rate (Q), and the ratio of samples per pixel (γ), referred to as the sampling rate.

Sample Quantization Rate. The sample quantization rate (Q) refers to the number of bits per sample used to quantize the data for digital transmission. We conducted empirical studies to test the effect of quantization of the samples generated from linear combinations of these pixels as in (4) over a set of reference images with a constant overall compression rate, and reported in Fig. 3, which shows the SSIM[2] index [40] of a set of reference images for multiple quantization levels. The reference images used are 25 grayscale images from the USC Signal and Image Processing Institute image repository [41]. As Q decreases and less bits are being used to encode each sample, more samples can be obtained for the same compression rate. There is a clear maximum value at $Q = 5$.

Sampling Rate γ. The sampling rate γ is the number of transmitted samples per original image pixel. Again, an empirical study was performed on the images in [41] to determine the amount of distortion in the recreated images due to varying sampling rates, and is reported in Fig. 4.

The proposed CSV encoder is designed to: i) encode video at low complexity for the encoder, and ii) take advantage of the temporal correlation between frames. While the proposed method is general, it works particularly well for security videos, in which the camera is not moving, but only the objects within the field of view (FOV) of the camera are moving. Because of this, there will often be a large amount of redundancy from one frame of the video to the next. To exploit this redundancy within the framework of compressed sensing, we take the algebraic difference between the CS

[2]The SSIM index is preferred to the more widespread PSNR, which has been recently shown to be inconsistent with human eye perception [40]. SSIM is a more accurate measurement of error because the human visual system perceives structural errors in the image more than others. For example, changes in contrast or luminance, although mathematically significant, are very difficult to discern for the human eye. Structural differences such as blurring, however, are very noticeable. SSIM is able to weight these structural differences better to create a measurement closer to what is visually noticeable than traditional measures of image similarity such as mean squared error (MSE) or PSNR.

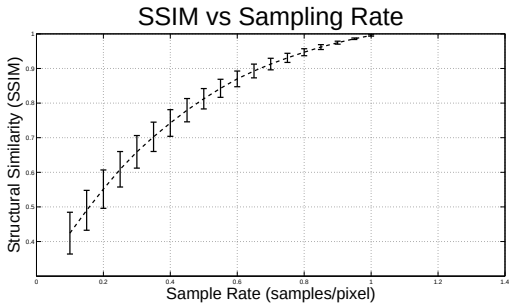

Fig. 4. Structural Similarity (SSIM) Index [40] for Images with varying levels of sampling rate γ.

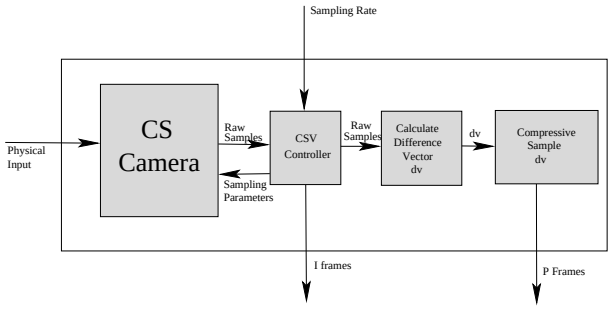

Fig. 5. Block Diagram for CS Video Encoder.

samples. Then, this difference is *again compressively sampled* and transmitted. If the image being encoded and the reference image are very similar (i.e. have a very high correlation coefficient), then this difference image will be sparser and have less variance than either of the original images, and can therefore be transmitted at the same quality using fewer samples and a lower Q than the original image.

2) Video Encoding: The video encoding process is determined by the type of encoding to be used for a given frame, as shown in Fig. 5. The pattern of the encoded frames is $IPPP\cdots PIPPP\cdots$, where the distance between two I frames is referred to as the group of pictures (GOP).

I frames are encoded using (4). The number of samples to include is determined as $\gamma \cdot N$, where N is the number of pixels in the unencoded frame and γ is the sampling rate. The rate control law to determine the current value for γ is discussed in Section V. The samples are then quantized with $Q = 5$ and transmitted over the channel.

P frames are also sampled using (4) with γ equal to the γ of the most recent I frame. The difference vector (dv) for frame t is then calculated with

$$dv = S_I^* - S_t^*, \tag{5}$$

where S_t^* is a vector containing all of the samples of the t^{th} frame, and S_I^* is the vector containing the samples of the most recent I frame. The dv is then compressed again using (4), quantized with $Q = 3$ and transmitted over the channel.

3) Video Decoding: The decoding process, shown in Fig. 6, uses (3) to determine the dv (in the case of a P frame) and

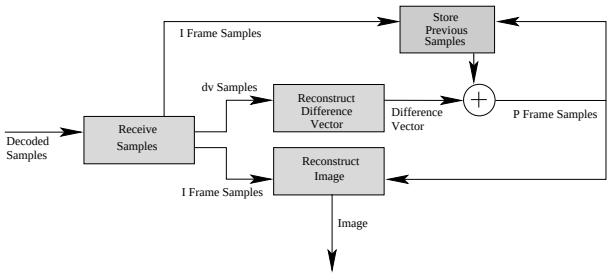

Fig. 6. Block Diagram for CS Video Decoder.

the original frame. For I frames, the frame can be directly reconstructed from the received samples. For P frames, the dv must first be reconstructed. Once this vector is reconstructed using (3), the samples for the t^{th} P frame are found by $S_P^* = dv + S_I^*$. The t^{th} frame is then reconstructed using (3) from S_P^*.

V. Rate Control Subsystem

In this section, we introduce the congestion avoidance rate control mechanism for use with the compressed sensed video encoder (CSV) described in Section IV-B. This rate control system both provides fairness in terms of video quality and maximizes the overall video quality of concurrent videos transported through the network.

To avoid network congestion, a sending node needs to take two main factors into account. First, the sender needs to regulate its rate in such a way as to allow any competing transmissions at least as much bandwidth as it needs to attain a comparable video quality as itself. Note that this is different from current Internet practice, in which the emphasis is on achieving fairness in terms of data rate (not video quality). Second, the sender needs to regulate its rate to make sure that packet losses due to buffer overflows are reduced, which can be done by reducing the overall data rate if it increases to a level which the network can not handle.

To measure congestion, the round trip time RTT is measured for the transmitted video packets, where RTT is defined as the amount of time it takes for a packet to go from the source to the destination and a small reply packet to go from the destination back to the source. In this paper, the change in RTT (ΔRTT) is measured as

$$\widetilde{\Delta RTT_t} = \frac{\sum_{i=0}^{N-1} A_i \cdot RTT_{t-i}}{N \cdot \sum_{i=0}^{N-1} A_i} - \frac{\sum_{i=1}^{N} A_i \cdot RTT_{t-i}}{N \cdot \sum_{i=1}^{N} A_i}, \tag{6}$$

which is the difference of the weighted average over the previous N received RTT measurements with and without the most recent measurement. The weights A_i are used to low-pass filter the round trip time measurements, i.e., to give more importance to the most recent RTT measurements and to make sure that the protocol reacts quickly to current network

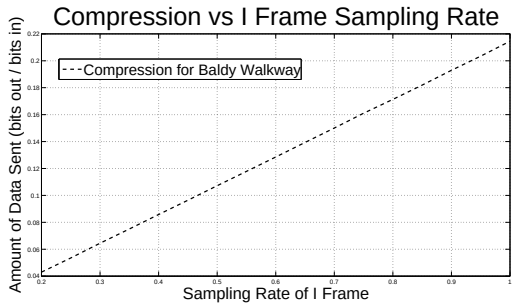

Fig. 7. Ratio of Encoder Output to Encoder Input vs I Frame Sampling Rate.

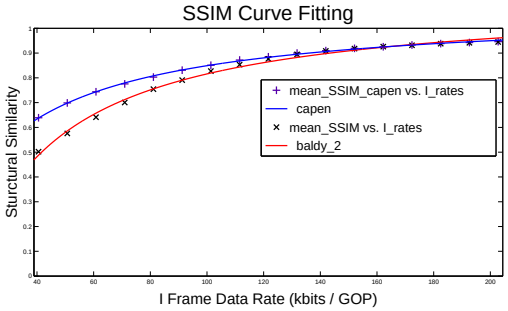

Fig. 8. Linear Least Square Curve Fitting for Two Videos using CS Video Encoder.

events, while averaging assures that nodes do not react too quickly to a single high or low measurement.

A. Indirect Rate Control

The video encoder described in Section IV generates two types of video frames; the I frame, which is an intra-encoded frame, and the P frame, which is an inter-encoded frame. The I frames are *independently encoded*, i.e., they are encoded using only the data contained within a single frame allowing these frames to be decoded independently of the previous frame. However, I frames do not take advantage of correlation between frames resulting in lower rate-distortion performance. P frames on the other hand are encoded based on previous frames by leveraging the temporal correlation between frames. Although this results in smaller frame sizes, it also allows errors to propagate from one frame to the next [26].

We present a novel approach in which the rate is not controlled directly, but instead the data rate is varied indirectly by varying the compression rate γ_I, defined in Section IV-B1. More specifically, γ_I is directly controlled by the rate controller, based on the RTT. Since the γ_I is linearly proportional to the compression of the I frames as seen in Fig. 7, this directly controls the compression rate of the entire video. This is important because the compression of the I frames can be directly controlled through one variable, while the compression of the P frames depends not only on γ_I, but also on the amount of redundancy between the P frame and the previous frame. Because of this linear relationship, we can confidently control the compression of the entire video by varying a single parameter.

We model the quality of the received video stream with a three-parameter model [26]

$$D_I = D_0 + \frac{\theta}{\gamma_I - R_0}, \tag{7}$$

where D_I represents the distortion of the video and R_I^m. The parameters D_0, θ and R_0 depend on the video characteristics and Q and can be estimated from empirical rate-distortion curves via a linear least-square curve fitting.

The rate control is based on the parameter δ, which is defined by

$$\delta = \frac{\theta}{(R_I^m - R_0)^2}, \tag{8}$$

which is the derivative of (7).

δ is used in (9) below to promote fairness in terms of distortion. If there are two nodes transmitting video and both notice the same negative value for $\Delta\widetilde{RTT_t}$, the sending node with the lower current video quality will take advantage of the decreased network congestion more than the node which is transmitting at a higher rate. The inverse is true for positive values of $\Delta\widetilde{RTT_t}$. This can be seen in Fig. 8. At lower compression levels, a change in the rate has a larger impact on the received image quality than an equal change will have at a higher rate. Similarly, $1-\delta$ results in a function with very low values at low rates, and higher values at higher rates. This $1-\delta$ is used to prevent a node from decreasing the rate significantly when the rate is already low, but encourage the node to decrease the rate when the data rate is already high

At the source node of each video transmission, the amount of data generated by the video source is implemented through the equation

$$\gamma I, t+1 = \begin{cases} \gamma I, t - \delta \cdot \beta \cdot \Delta\widetilde{RTT_t} & \text{if } \widetilde{RTT_t} > \alpha \\ \gamma I, t + (1-\delta) \cdot \kappa \cdot \Delta\widetilde{RTT_t} & \text{if } \widetilde{RTT_t} < \alpha \\ \gamma I, t & \text{else,} \end{cases} \tag{9}$$

where $\beta > 0$ and $\kappa > 0$ are both constants used to scale δ to the range of the sampling rate.

Channel errors are accounted for through the use of the adaptive parity scheme, described in Section VI. The adaptive parity scheme provides feedback to the C-DMRC rate controller indicating the expected sample delivery success rate C. Based on the value of C, the rate controller can determine how much to increase the data rate in order to compensate for the expected number of lost samples.

VI. Adaptive Parity-based Transmission

For a fixed number of bits per frame, the perceptual quality of video streams can be further improved by dropping errored samples that would contribute to image reconstruction with

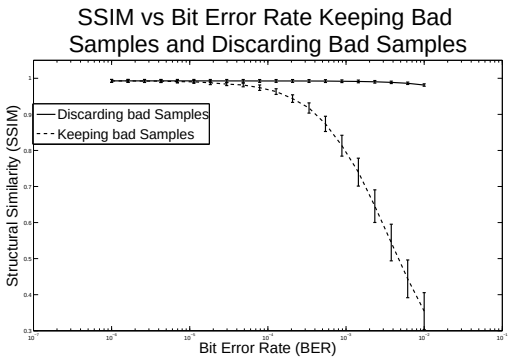

Fig. 9. SSIM for images with and without errored samples.

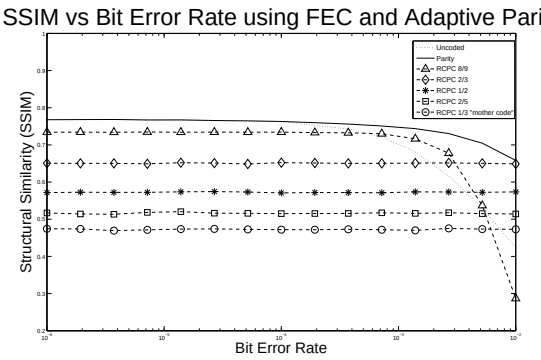

Fig. 10. Adaptive Parity vs RCPC Encoding for Variable Bit Error rates.

incorrect information. This is demonstrated in Fig. 9 which shows the image quality both with and without including samples containing errors. Though the plots in Fig. 9 assume that the receiver knows which samples have errors, it does demonstrate that there is a very large possible gain in received image quality is those samples containing errors can be found without adding too much overhead.

We studied this for images in [42]. It was shown that in CS, the transmitted samples constitute a random, incoherent combination of the original image pixels. This means that, unlike traditional wireless imaging systems, no individual sample is more important for image reconstruction than any other sample. Instead, the number of correctly received samples is the only main factor in determining the quality of the received image. Because of this, a sample containing an error can simply be discarded and the impact on the video quality, as shown in Fig. 9, is negligible as long as the amount or errors is small. This can be realized by using even parity on a predefined number of samples, which are all dropped at the receiver or at an intermediate node if the parity check fails. This is particularly beneficial in situations when the BER is still low, but too high to just ignore errors. To determine the amount of samples to be jointly encoded, the amount of correctly received samples is modeled as

$$C = \left(\frac{Q \cdot b}{Q \cdot b + 1}\right)(1 - BER)^{Q \cdot b}, \tag{10}$$

where C is the estimated amount of correctly received samples, b is the number of jointly encoded samples, and Q is the quantization rate per sample. To determine the optimal value of b for a given BER, (10) can be differentiated, set equal to zero and solved for b, resulting in $b = \frac{-1+\sqrt{1-\frac{4}{log(1-BER)}}}{2Q}$.

The optimal channel encoding rate can then be found from the measured/estimated value for the end-to-end BER and used to encode the samples based on (10). The received video quality using the parity scheme described was compared to different levels of channel protection using rate compatible punctured codes (RCPC). Specifically, we use the $\frac{1}{4}$ mother codes discussed in [43]. As these codes are punctured to reduce the redundancy, the effectiveness of the codes decreases as far as the ability to correct bit errors. Therefore, we are trading BER for transmission rate.

Figure 10 shows the adaptive parity scheme compared to RCPC codes. For all reasonable bit error rates, the adaptive parity scheme outperforms all levels of RCPC codes. The parity scheme is also much simpler to implement than more powerful forward error correction (FEC) schemes. This is because, even though the FEC schemes show stronger error correction capabilities, the additional overhead does not make up for the video quality increase compared to just dropping the samples which have errors.

VII. Performance Evaluation

We perform two sets of experiments[3] to verify the performance of the C-DMRC system. First, the rate controller is simulated using ns-2 version 2.33. In addition, to evaluate the effect of a real wireless channel, CS video streaming with the adaptive parity-based channel encoder is tested on a multi-hop testbed based on USRP2 software defined radios.

A. Evaluation of Rate Controller

The rate control algorithm of C-DMRC is compared directly to TFRC to assess: (i) the received video quality for a single video transmission; (ii) the overall fairness (as measured with Jain's Fairness Index); (iii) the overall received video quality over multiple video transmissions. The topology of the network is a Manhattan Grid consisting of 49 nodes (7x7). The senders and sink are chosen randomly for 10 random number seeds. All senders transmit video to a single sink node. Routing is based on AODV [44], and MAC on 802.11.

We use real video traces recorded at the University at Buffalo to simulate video traffic within the ns-2 simulator. Initially, trace files are obtained from the CSV video encoder for multiple values of γ_I. These trace files are input into ns-2, where the rate control decisions are made at simulation time. The network simulator determines the sampling rate γ_I, and derives the video size based on this value. After network simulation, a trace of the received samples is fed back into the CSV video decoder and the resulting received video frames

[3]All simulation code and videos used in these simulations is available at http://www.eng.buffalo.edu/wnesl/Scott.htm

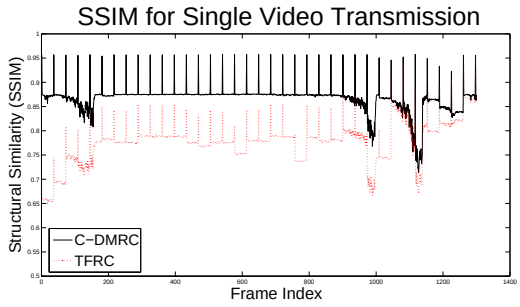

Fig. 11. SSIM of a single video transmission using C-DMRC and TFRC.

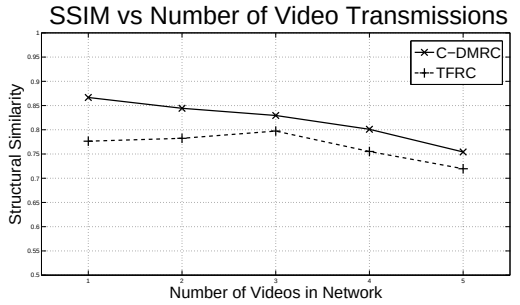

Fig. 12. SSIM of a multiple video transmissions using C-DMRC and TFRC.

are recreated. These video frames are then compared to the original uncompressed (and untransmitted) video.

The first experiment simulates a single-video transmission within the 49 node network. This is done to compare C-DMRC and TFRC in a best-case scenario (i.e., no inter-flow interference and sufficient available capacity). Figure 11 shows the instantaneous SSIM at each frame of a 1300 frame video. Clearly, C-DMRC results in a higher SSIM value for almost the entire video. The portions of the graph where both SSIM values drop represent portions of the video where the traffic originating from the video increased, resulting in an increase of RTT and a decrease in the sampling rate. Both C-DMRC and TFRC responded to this traffic increase quickly, but C-DMRC recovered much more quickly than TFRC.

The second simulations compare C-DMRC and TFRC with multiple videos simultaneously transmitted in the network. The number of video transmissions varied from 1 to 5, with each video starting 10 seconds (120 frames) after the previous. The varying starting rates assure that videos starting at different times are treated fairly.

Figure 12 shows the results from this simulation. For each of the simulations, C-DMRC results in a higher average SSIM than TFRC. The fairness is evaluated in Fig. 13, where Jain's Fairness Index [45] measures the fairness between multiple senders. Again, C-DMRC clearly performs better than TFRC.

B. Adaptive Parity Testbed Evaluation

The considered testbed setup uses USRP2 [46] and GNU radio [47]. A two-hop network is considered. The MAC protocol is IEEE 802.11 and the modulation scheme employed is differential quadrature phase shift keying (DQPSK), with a

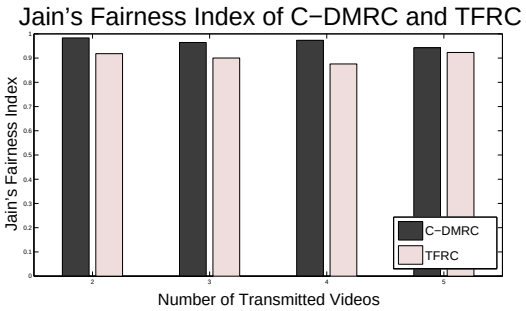

Fig. 13. Jain's Fairness Index of a multiple video transmissions using C-DMRC and TFRC.

physical layer data rate of $2\ \mathrm{Mbit/s}$. The radios are placed $6\ \mathrm{m}$ apart, and the frequency selected for transmission is $2.433\ \mathrm{GHz}$. A 100-byte pilot sequence precedes 2150 bytes of data in the IEEE 802.11 packet. The additional transport header consists of a sequence number (1 byte), source and destination addresses (2 bytes) and an I/P frame indicator (1 byte). A burst of packets corresponding to the samples of a video frame are transmitted through the two hop network. The BER is estimated by averaging the received pilot sequences of packets specific to a burst. On average, there are 6 packets/burst. Using this data, the number of samples per parity bit (b as introduced in 10) is calculated as a function of the quantization level of samples (Q) and the estimated BER. The parity bits are appended to each set of b samples for the subsequent packets.

Over the course of a 300 frame video, this system was able to correctly detect 92,719 out of 93,269 sample errors correctly, for a 99.41% error detection rate. The maximum sample error rate for any single frame was 0.0021, which results in a BER of 2.5×10^{-4} which, as shown in Fig. 9, results in a decrease in SSIM of less than 10%. This performance could be further improved by using better BER estimation, thus increasing the accuracy of the adaptive parity encoder.

VIII. Conclusions and Future Work

This paper introduced a new wireless video transmission system based on compressed sensing. The system consists of a video encoder, distributed rate controller, and an adaptive parity channel encoding scheme that take advantage of the properties of compressed sensed video to provide high-quality video to the receiver using a low-complexity video sensor node. Simulation results show that the C-DMRC system results in better received video quality in both a network with a higher load and a small load. Simulation results also show that fairness is not sacrificed, and is in fact increased, with the proposed system. Finally, the system was implemented on a USRP-2 software defined ratio, and it was shown that the adaptive parity scheme effectively combats errors in a real channel. We intend to implement the entire system using USRP-2 radios, including the video encoder. We will also measure the performance and complexity of this system compared

to state-of-the-art video encoders (H.264, JPEG-XR, MJPEG, MPEG), transport (TCP, TFRC) and channel coding (RCPC, Turbo codes).

REFERENCES

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A Survey on Wireless Multimedia Sensor Networks," *Computer Networks (Elsevier)*, vol. 51, no. 4, pp. 921–960, Mar. 2007.

[2] "Advanced Video Coding for Generic Audiovisual Services," ITU-T Recommendation H.264.

[3] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, 2003.

[4] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammar, and T.Wedi, "Video coding with H.264/AVC: Tools, performance, and complexity," *IEEE Circuits and System Magazine*, vol. 4, no. 1, 2004.

[5] T.Wiegand, G. J. Sullivan, J. Reichel, H. Schwarz, and M.Wien, "Joint Draft 11 of SVC Amendment," Doc. JVT-X201, July 2007.

[6] I. Akyildiz, T. Melodia, and K. Chowdhury, "Wireless multimedia sensor networks: Applications and testbeds," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, Oct. 2008.

[7] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," *Proc. of the IEEE*, vol. 93, no. 1, pp. 71–83, January 2005.

[8] A. Wyner and J. Ziv, "The Rate-distortion Function for Source Coding with Side Information at the Decoder," *IEEE Trans. on Information Theory*, vol. 22, pp. 1–10, January 1976.

[9] A. Aaron, S. Rane, R. Zhang, and B. Girod, "Wyner-Ziv Coding for Video: Applications to Compression and Error Resilience," in *Proc. of IEEE Data Compression Conf. (DCC)*, Snowbird, UT, March 2003, pp. 93–102.

[10] A. Aaron, E. Setton, and B. Girod, "Towards Practical Wyner-Ziv Coding of Video," in *Proc. of IEEE Intl. Conf. on Image Processing (ICIP)*, Barcelona, Spain, September 2003.

[11] T. Sheng, G. Hua, H. Guo, J. Zhou, and C. W. Chen, "Rate allocation for transform domain Wyner-Ziv video coding without feedback," in *ACM Intl. Conf. on Multimedia*, October 2008.

[12] D. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[13] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[14] E.J. Candes and J. Romberg and T. Tao, "Stable Signal Recovery from Incomplete and Inaccurate Measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[15] E. Candes and T. Tao, "Near-optimal Signal Recovery from Random Projections and Universal Encoding Strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[16] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk, "Compressive imaging for video representation and coding," in *Proc. Picture Coding Symposium (PCS)*, April 2006.

[17] J. Romberg, "Imaging via Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 14–20, 2008.

[18] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk, "Single-Pixel Imaging via Compressive Sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[19] S. Pudlewski and T. Melodia, "DMRC: Distortion-minimizing Rate Control for Wireless Multimedia Sensor Networks," in *Proc. of IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Hong Kong S.A.R., P.R. China, October 2009.

[20] M. Allman, V. Paxson, and W. Stevens, "TCP Congestion Control," IETF RFC 2581.

[21] K. Tan, J. Song, Q. Zhang, and M. Sridharan, "A Compound TCP Approach for High-Speed and Long Distance Networks," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April 2006, pp. 1–12.

[22] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," in *In Proc. of the Special Interest Group on Data Communications (ACM SIGCOMM)*, 1998.

[23] W.-T. Tan and A. Zakhor, "Real-time Internet video using error resilient scalable compression and TCP-friendly transport protocol," *IEEE Transactions on Multimedia*, vol. 1, no. 2, pp. 172–186, Jun 1999.

[24] M. Handley, S. Floyd, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," IETF RFC 3448.

[25] O. B. Akan and I. F. Akyildiz, "ARC: the analytical rate control scheme for real-time traffic in wireless networks," *Networking, IEEE/ACM Transactions on*, vol. 12, no. 4, pp. 634–644, Aug. 2004.

[26] K. Stuhlmuller, N. Farber, 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, Jun 2000.

[27] Q. Zhang, W. Zhu, and Y. Zhang, "End-to-End QoS for Video Delivery Over Wireless Internet," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 123–134, Jan 2005.

[28] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting Real-Time Video Over the Internet: Challenges and Approaches," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1855–1877, Dec 2000.

[29] V. Stankovic, L. Stankovic, and S. Cheng, "Compressive Video Sampling," in *In Proc. of the European Signal Processing Conf. (EUSIPCO)*, Lausanne, Switzerland, August 2008.

[30] J. Park and M. Wakin, "A Multiscale Framework for Compressive Sensing of Video," in *In Proc. of the Picture Coding Symposium (PCS)*, Chicago, Illinois, May 2009.

[31] R. Marcia and R. Willett, "Compressive coded aperture video reconstruction," in *In Proc. of the European Signal Processing Conf. (EUSIPCO)*, Lausanne, Switzerland, August 2008.

[32] A. Bruckstein, D. Donoho, and M. Elad, "From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images," *Preprint*, 2007.

[33] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, Mar. 2004.

[34] I. E. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, PA, USA: SIAM, 1994.

[35] M. Zhu and T. Chan, "An Efficient Primal-Dual Hybrid Gradient Algorithm for Total Variation Image Restoration," Technical report, UCLA CAM Report 08-34, 2008.

[36] A. Graps, "An Introduction to Wavelets," *IEEE Computational Science and Engineering*, vol. 2, pp. 50–61, 1995.

[37] L. Gan, T. Do, and T. D. Tran, "Fast Compressive Imaging Using Scrambled Block Hadamard Ensemble," *Preprint*, 2008.

[38] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, vol. 1, no. 4, pp. 586–598, 2007.

[39] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," *Preprint*, 2007.

[40] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, April 2004.

[41] USC Signal and Image Processing Institute, http://sipi.usc.edu/database/index.html.

[42] T. Melodia and S. Pudlewski, "A Case for Compressive Video Streaming in Wireless Multimedia Sensor Networks," *IEEE COMSOC MMTC E-Letter*, vol. 4, no. 9, October 2009.

[43] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 389–400, Apr 1988.

[44] C. E. Perkins, E. M. Belding-Royer, and S. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing," IETF RFC 3561.

[45] R. Jain, D.-M. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *DEC Research Report TR-301*, September 1984.

[46] Ettus Research LLC, http://www.ettus.com/.

[47] Eric Blossom, "GNU Software Radio," http://gnuradio.org/.