# Performance Evaluation of Indoor Wireless Video System Using BLAST Testbed

Haitao Zheng and Dragan Samardzija
Wireless Communications Research Department
791 Holmdel-Keyport Rd. Holmdel, NJ 07733
Email: haitaoz, dragan@lucent.com

## Abstract

Multiple antenna system is shown to provide high capacity wireless communications. We have built a narrowband wireless BLAST testbed with multiple transmit and receive antennas. To validate the effectiveness of the testbed and BLAST technique, we transmit a H.263 video at a rate of 230 kbps. The video performance under different channel conditions and error handling options are discussed.

## 1. Introduction

The Bell Labs Layered Space-Time (BLAST) [1] architecture utilizes multi-element antenna arrays at both transmitter and receiver to provide high capacity wireless communications in a rich scattering environment. It has been shown that the theoretical capacity increases linearly as the number of antennas is increased. Two types of BLAST realizations have been developed, Vertical BLAST (VBLAST) and Diagonal BLAST (DBLAST). VBLAST is a simplified version where channel coding is applied to individual sub-layer, each corresponding to the data stream transmitted by a single antenna. DBLAST applies coding not only across the time, but also across the antennas (sub-layers), and implies higher complexity. We have built a narrowband wireless testbed based on VBLAST, which is used for verifications and performance evaluations of different algorithms related to the BLAST wireless communication architecture. To illustrate the high capacity gain provided by BLAST, we perform transmission of H.263 video coded at 230kbps over the VBLAST testbed and the performances under different channel designs are studied.

## 2. Narrowband VBLAST Wireless Test Bed

Let us now describe hardware components of the narrowband VBLAST wireless testbed. Radio frequency (RF) front end of the testbed consists of an antenna array, and the corresponding array of analog RF transmitters and receivers. In this particular experiment we used up to eight transmit and eight receive antennas. The carrier is at 1.95 GHz and the signal bandwidth is limited to 30KHz.

The baseband digital signal processing is executed using a DSP multiprocessor system: Pentek 4285 [2]. It consists of eight Texas Instrument's TMS320C40 DSPs, offering total processing power of 400 MIPS. The interfacing towards the baseband inputs and outputs of the array of analog RF transmitters and receivers is realized using a system of multi-channel A/D (Pentek 4275 [3]) and D/A (Pentek 4253 [4]) converters, respectively. The maximum sampling rate, per a baseband channel, is 100 KHz.

### 2.1 Modulation and Data Formats

In this particular experiment we use the QPSK modulation format, transmitting at 25 Ksym/sec, per subuser (i.e., per antenna). Further, the symbols are organized as follows (see Figure 1.):

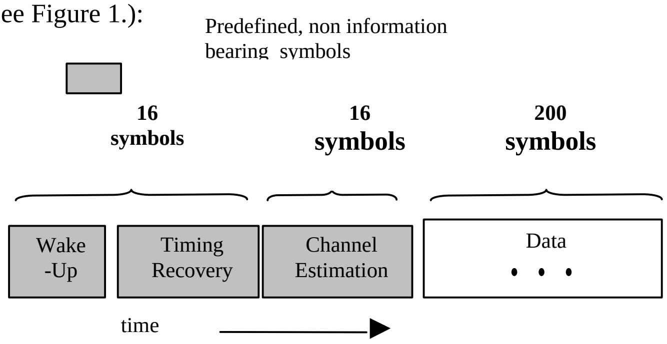

**Figure 1. Frame Structure**

Symbols 1 to 16 are used for synchronization, i.e., frame and symbol timing recovery. Note this part of the frame is identical for all the subusers. Symbols 17 to 32 compose a training sequence, which is used for estimation of the channel response. Between the subusers, the sequences are mutually orthogonal and with equal transmission power. Symbols 33 to 232 are information-bearing symbols. Considering the QPSK format, 400 bits are transmitted per frame, per subuser.

### 2.2 Baseband DSP Blocks

Let us now briefly describe the baseband digital signal processing blocks that are applied at the receiver.

A. **Detection of the frame start.**

In this block, we wait on a sufficiently strong signal that indicates the initialization of data transmission. At the transmitter, as said earlier, the synchronization sequence precedes the information-bearing signal. The sequence is supposed to indicate the start of the transmission for the listening receiver.

B. **Symbol timing recovery**.

After the strong signal is detected, the received signal, which is four times oversampled (i.e., four samples per symbol period), is crosscorrelated with predefined synchronization sequence, that exhibits good autocorrelation properties (in this case we apply a binary Barker sequence [4]). A crosscorrelation lag that results in highest crosscorrelation is used to establish symbol synchronization.

C. **Mitigation of hardware induced intersymbol interference (ISI)**.

At the transmitter side, the spectrum of the transmitting signal is shaped with an analog lowpass filter. Further, the processing at the RF front-end of the transmitter additionally distorts the spectrum. Consequently at the receiver, in order to mitigate the ISI caused by the spectrum shaping and its distortions, a fixed coefficient FIR filter is applied on received signal. The coefficients of the filter are precalculated using laboratory measurements of the received spectrum and its inverse.

D. **Estimation of the channel response.**

The estimation is based on using mutually orthogonal training sequences between the subusers. We choose the Hadamard sequences.

E. **The VBLAST Algorithm.** Based on the channel responses estimated in block D, we perform VBLAST algorithm. Note that the decision on transmitted data can be performed in this step, but instead, the soft and normalized outputs are passed to the channel decoder.

### 2.3 Channel Coding

As said earlier, in this experiment, each frame consists of 232 QPSK symbols, where 32 symbols are dedicated to synchronization and training. Therefore, 200 QPSK symbols per antenna is for data transmission. To achieve better coding efficiency, one single convolutional code is applied to all the subusers. We employ rate 1/2 and 1/3 convolutional codes of constraint length 8. In addition, a rate 2/3 code is obtained by puncturing the output of the rate 1/3 code. By multiplexing the coded bits into 8 sub-users, an interleaving of depth 8 is achieved naturally. At the receiver, the VBLAST algorithm is applied to extract the soft input, which is forwarded to the channel decoder. The data rate when there is no coding can be computed as 25*1000*2*(232-32)/232= 344.8276kbps. The frame structures for different coding rates are illustrated below.

**Table 1. Frame Structure**

Frame Size = 9.28ms, 3200 Data bits Per Frame

| Coding rate | Info Bits | Data Rate |
|---|---|---|
| 1 | 3200 | 345kbps |
| 2/3 | 2125 | 230kbps |
| 1/2 | 1600 | 172.4kbps |
| 1/3 | 1058 | 115kbps |

## 3. System Structure and Protocol Format

We are interested to validate VBLAST testbed performance by transmitting video data. The video quality is presented in terms of the Peak Signal-to-Noise Ratio (PSNR) of the video. The system structure is shown in Figure 2.

### a) Video Format

We choose an H.263+ coded video sequence, with a bit rate of 230kbps at 15fps. The following error-resilience features were implemented: 1) inserting one intra frame every five frames, 2) insert sync word in each GOB(slice). For more information about the H.263+ video coding technique, see [6].

### b) Packetization and Packet Level Error Handling

The video stream is packetized through the detection of GOB synchronization word. In another word, each GOB corresponds to one application packet, and the resulted packets are of different lengths. Each packet is accompanied by a 16-bit CRC check for content validation. Channel errors usually only partially corrupt a packet. If the protocol discards a packet containing only a small part of corrupted data, it also throws out error-free data within the packet. Indeed, the media decoder can detect and tolerate a certain amount of channel errors. To support this feature, it would be possible to still forward the corrupted packet to the video decoder and let the video decoder to detect the errors. Therefore, when the packet CRC fails, we consider the following two options for comparison:

I. Discard the packet
II. Forward the packet to video decoder

### c) Physical Frame Level Error Handling

Each physical layer frame is accompanied by a 16-bit CRC check. At the receiver, nearly all the errors can be detected. This indeed provides an accurate error indication. However, in the conventional system design, the physical layer does not communicate with the application layer. And it might simply discard the frame. For video/audio, this sometimes generates additional errors. Therefore, we have proposed to forward the frame error indication to the application layer[7]. One example would be replacing the corrupted physical layer frames as all 1s, which can be recognized by the media decoder as an invalid codeword and thus invokes error concealment to reduce or even eliminate the error effect. When video decoder is effective in terms of error detection, physical and network layer can simply forward the corrupted frames/packets to the video decoder for flexible error control. In this experiment, when CRC detects channel error, we compare three options in terms of error handling in physical layer:

A. Discard the frame
B. Forward the frame to video depacketization.
C. Replace the frame as all ones

It should be noted that when option I in packet level error handling, i.e. discard the packet is employed, the performance remains the same for option A to C. Therefore, we simply compare the following four options in terms of packet level and physical layer level error handling techniques: I, II+A, II+B and II+C.

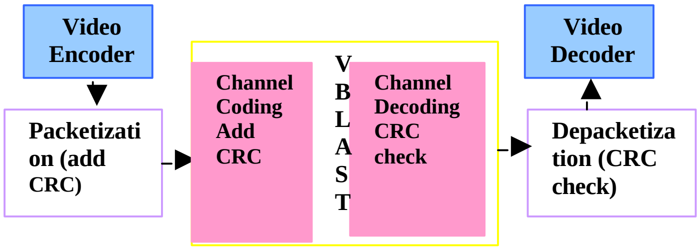

Figure 2. System Architecture

## 4. Experimental Results

We use VBLAST with eight transmit antennas and eight receive antennas, with the same frame structure shown in Figure 1 and Table 1. Figure 3 illustrates the received frame level error traces before channel coding. The raw frame error rate is about 30%, and in fact bursty. The performance after a rate 2/3 channel coding in Figure 4 shows that the frame error rate is reduced to 3% and the bursty error disappears.

Figure 5 illustrates the coded and uncoded frame error rate (FER), bit error rate (BER) at different SNR values, using rate 2/3 convolutional code. The video sequences are transmitted using the error traces generated from the real-time testbed results. The error traces reflects 16%-1.5% FER. The experiment is performed 20 times and the average PSNR performances are presented in Figure 6. The II+B option outperforms the others since the number of error bits within a frame is small. And the II+C option outperforms II+A and I for as much as 4dB PSNR gain. Therefore, it is concluded that the protocol architecture design can greatly impact the system performance. For multimedia application with error resilience, information should all be forwarded to the video decoder.

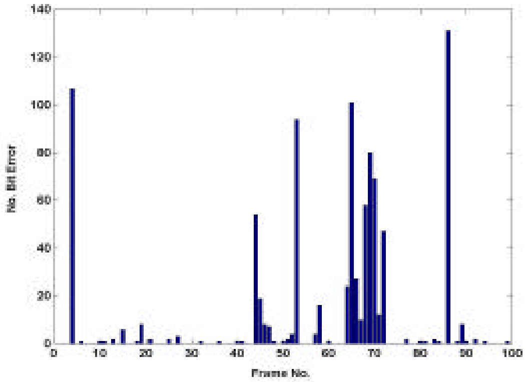

Figure 3. Frame/Bit Error Traces assuming no channel coding.

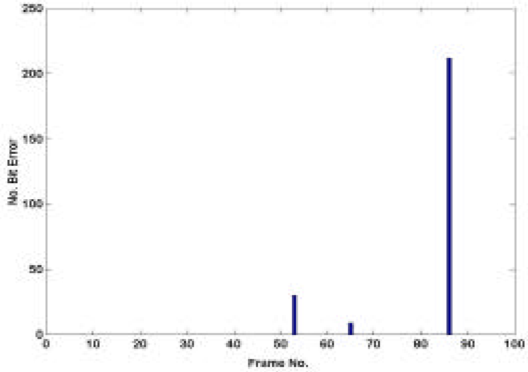

Figure 4. Frame/Bit Error Traces assuming rate 2/3 channel coding.

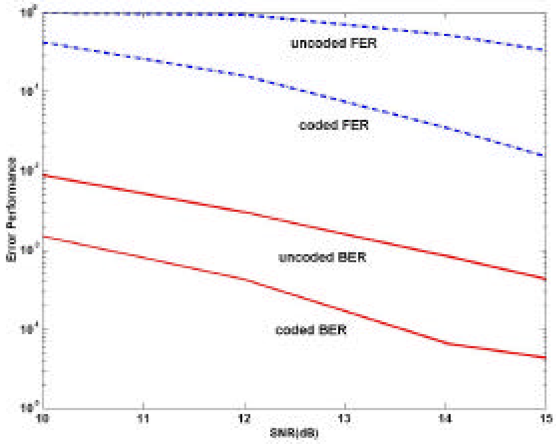

Figure 5. FER and BER Performances

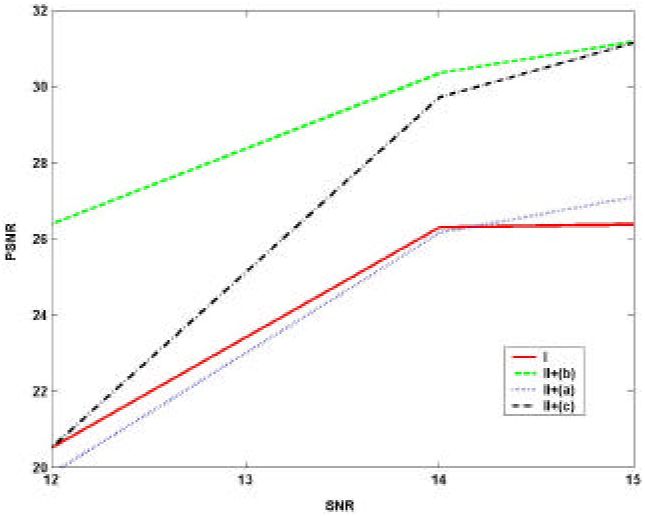

**Figure 6. Video Performances**

## 5. Conclusion

Wireless video transmission is performed using a narrowband VBLAST testbed employing multiple transmit and receive antennas. System performance is measured in terms of bit error rate and frame error rate. Simple convolutional coding can significantly reduce the number of corrupted frames. The system is validated by transmitting H.263 packet video. Error handling technique at both packet and physical frame level are discussed and compared. Since within a frame, the number of bit error is small, forwarding the original corrupted frames to the video decoder achieves the best performance.

## References

[1] G. J. Foschini, "Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multiple Antennas", Bell Labs Technical Journal, Vol. 1, No. 2, Autumn 1996, pp 41-59.

[2] Operating Manual, Model 4285, Octal TMS320C40 Digital Signal Processor for VMEbus Systems, Pentek, Inc. Part No. 800.42850, Rev. B.2, Sep. 1, 1998.

[3] Operating Manual, Model 4275, 32 Channel, 100 kHz, 16-bit Sigma-Delta A/D Converter MIX Module for VMEbus Systems, Pentek, Inc. Part No. 800-42750, Rev. C.4, July 3, 1998.

[4] Operating Manual, Model 4253, 32 Channel 100 kHz D/A Converter MIX Module for VMEbus Systems, Pentek, Inc. Part No. 800.42530, Rev. C, Aug. 26, 1998.

[5] Digital Communications by Satellite, James J. Spilker, Prentice Hall, 1977.

[6] ITU Recommendation H.263 Version 2, Video Coding for Low Bitrate Communication, Jan. 1998.

[7] H. Zheng and J. Boyce, "Packet Coding Schemes for MPEG Video over Internet and Wireless Networks", IEEE WCNC conference, Sept. 2000, Chicago, IL.