Design of MIMO Testbed with an FPGA Board for Fast Signal Processing

Konstanty S. Bialkowski, Peerapong Uthansakul, Marek E. Bialkowski and Adam Postula
University of Queensland, School of Information Technology, St. Lucia, 4072 Brisbane, Qld, Australia
Tel: +61-7-33658308, Email: {ksb; ppg; meb; adam}@itee.uq.edu.au

Abstract

This paper describes the design of a Multiple Input Multiple Output testbed for assessing various MIMO transmission schemes in rich scattering indoor environments. In the undertaken design, a Field Programmable Gate Array (FPGA) board is used for fast processing of Intermediate Frequency signals. At the present stage, the testbed performance is assessed when the channel emulator between transmitter and receiver modules is introduced. Here, the results are presented for the case when a 2x2 Alamouti scheme for space time coding/decoding at transmitter and receiver is used. Various programming details of the FPGA board along with the obtained simulation results are reported.

1. Introduction

Multiple Input Multiple Output (MIMO) wireless communications is an emerging cost-effective technology that offers significant improvements to data throughput in a non-line-of-sight environment [1], [2], [3]. In contrast to the traditional single transmit single receive antenna wireless system (also known as SISO), the MIMO system utilizes multiple element antennas (MEAs) both on transmit and receive sides of the communication link. The capacity gain of MIMO in a multi-path propagation environment is achieved at the expense of multiple element antenna transceivers which add complexity to the design of an overall system. Investigating properties of actual channels and determining optimal transmission schemes for MIMO has been the subject of research in many parts of the world. One component of this activity concerns the development of suitable signal propagation models [4-8] to predict MIMO performance under varying physical conditions. An ultimate objective is to test in practice these models along with new transmission schemes. This task can be accomplished using a MIMO testbed.

MIMO testbeds, such as described in [9-15] aim to measure variables such as the elements of the complex channel matrix, in addition to traditional communication system parameters like bit error rate (BER) or signal-to-noise ratio (SNR). The major challenge in the design and development of MIMO testbeds is handling of an increased data transmitted on multiple channels formed by a multiple element antenna system and a scattering environment. A special type of signal processor is required to tackle this task in an efficient manner.

This paper is concerned with the design, development and testing of a full 2x2 MIMO testbed which employs a Field Programmable Gate Array for fast parallel processing of transmitted data. The operation and performance of this MIMO testbed is investigated by assuming Alamouti scheme for space time coding/decoding at transmitter and receiver. This coding/decoding scheme is entirely implemented in FPGA hardware. The design concerns such issues as modulating/demodulating formats, the use of training sequences, and symbol synchronization. In order to test various stages of this system the purpose-developed channel emulator is used.

The paper is organized as follows. Section II shows the current status of the design and development of the full MIMO testbed. Section III presents results obtained with this testbed. Finally, section IV concludes this paper.

2. Full MIMO Testbed

2.1. Physical Setup

As already mentioned, while testing various transmission schemes a MIMO testbed requires an advanced signal processing device to handle an increased amount of transmitted and received data. As the transmission takes place over virtual parallel channels, preferably this processing needs to be done in parallel manner. One of the most powerful devices for parallel signal processing is the Field Programmable Gate Array (FPGA).

The FPGA selected to design a MIMO testbed with parallel processing capabilities is Altera's Stratix II FPGA with the Altera Digital Signal Processing Kit. This FPGA board features two high speed Analogue to Digital converters (ADC) and digital to analogue convectors (DAC). The two ADCs are capable of up to 125MSamples/sec and 12 bits of precision, whereas the two DACs are capable of 165MSamples/sec and 14bits of precision. In addition to high speed data processing, the board offers a 100Mbit Ethernet port for retrieval of data via a high speed interface.

2.2. System Description

A typical MIMO testbed includes a transmitter and a receiver each including a Multiple Element Antenna (MEA) connected to an RF front end followed by an up or down conversion to Intermediate Frequency (IF) module. In addition filters may be present depending on the type antennas (wideband or narrowband) that are used. This is to minimize noise before processing of an IF signal.

Our MIMO testbed is a 2x2 MIMO system and features all of the necessary modules besides RF and up/downconverters, which will be included in the near future. At this stage, a channel emulator block inside the FPGA is implemented to compensate for the lack of a real transmission channel.

The system uses a 2x2 Almouti Scheme for space time coding/decoding at transmitter and receiver. This involves coding one stream of data across two channels. The data rate of each output channel is equal to that of the equivalent SISO system.

The FPGA modules assume the presence of digital signals in the IF band at the transmitter side. These digital signals are converted to analogue ones via the DACs for RF transmission. On the receive side, an IF band signal is sampled via the ADC, and demodulated digitally by an FPGA module. These modules are illustrated in Fig. 1.

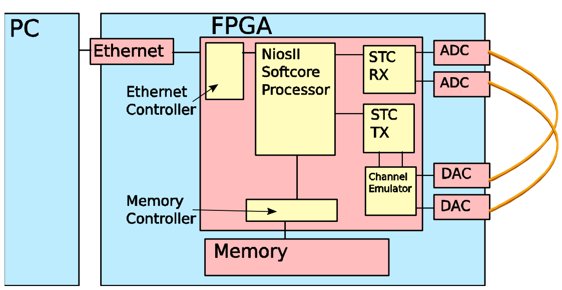


Fig. 1. Schematic of MIMO testbed including FPGA board

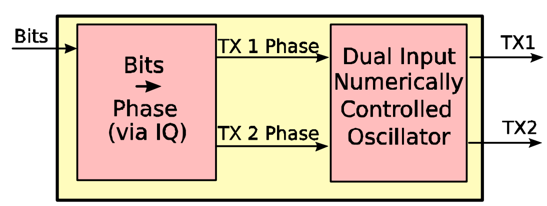
In order to post process and to gather results, a PC is used to interface with the FPGA module. Different types of data can be processed, ranging from the IF band signal, to the input and output bit streams.

In the current stage the completed module is the FPGA board with the particular modulation and coding scheme. The connection between the transmitter and the receiver modules is obtained by coaxial wires. This is to test modulation and demodulation functions when the data is unaffected by the random nature of wireless communication channel. Next, the coaxial connection is augmented with the channel emulator. While implementing this function, we use the signal scattering model described in [16]. This theoretical scattering model is valid for indoor environments and has already been tested against the data obtained for real measured channels. Good agreement between the two has been noted. Confirmation of validity of this channel emulator gives us confidence that if our MIMO testbed proves functioning properly for the emulated channel it should also work well in a real wireless environment.

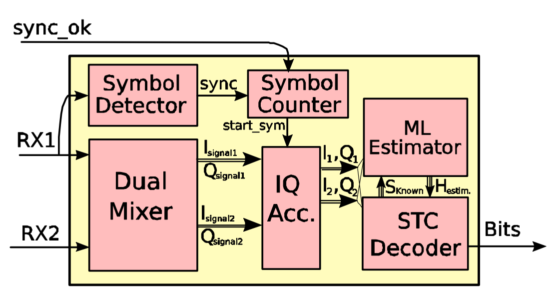
Inside the FPGA board the following modules are constructed: 1 transmit module (Space Time Modulator), 1 receive module (Space Time Demodulator including channel matrix **H** estimator), 1 channel emulator and control circuitry. These are illustrated in Fig. 2.

2.2.1. Transmit Module. The transmit module consists of two small modules, an IQ mapper and a numerically controlled oscillator (NCO). The IQ mapper encodes the input bitstream into a space-time code and then creates in-phase (I) and quadrature (Q) symbols, which are represented as discrete phases sent to the numerically controlled oscillator. The NCO consists of a look up table of sine values, with 14bits of accuracy, which can be runtime configured to generate frequencies between 97kHz and 50MHz (for the 100MHz on-board crystal being used). It is currently applied to generate a 6.25MHz waveform, with phase offset as defined by the IQ mapper.

The TX module has 4 modes of operation, both channels transmit, a single channel transmits (1 or 2) or no transmission occurs. In normal operation two modes are used, channel 1 only mode and both channels mode. This is done in order to allow the training sequence to be received. The inputs to the complete TX module include the bitstream and a mode selector.



(a)



(b)

Fig.2. The configuration of (a) transmit module (b) receive module.

2.2.2. Receive Module. The receiver module consists of two more complex modules, a mixer module and an IQ de-mapper. The mixer module mixes the received signal with a local cosine and sine signal, produced by the NCO. Only one NCO is required for both of these trigonometric functions, due to the phase difference between them being constant (90 degrees). In addition, the NCO is shared for mixing of both received signals.

One of the most critical sections of this module is synchronization with the carrier frequency and symbol changes. Synchronization changes can only occur when the receiver is expecting a training sequence. Carrier, symbol and decoding synchronization are all linked. A training sequence is selected, in which symbol transitions are easily detected. By measuring the offsets between consecutive symbol transitions in the training sequence, the offset of the carrier and position for space time decoding can be acquired.

From the mixing process, I and Q waveforms are obtained for both received signals. In order to get the value of I and Q channel, an integration process is used. In digital logic this is very simple, as it involves accumulation over a symbol period. Typically I and Q channel values are obtained via a low pass filter and sampling. Our approach does not require any multipliers.

During the training sequence the H matrix is calculated using the Maximum Likelihood (ML) estimation method. The offsets between consecutive symbol changes are used to decipher what known symbols are being transmitted. The inputs to this block are the accumulated I and Q channels and the known symbols. Through a process of 8 complex multiplications (32 real multiplications) each element of the H matrix can be calculated. The output of this block is stored in the H matrix which is used as one of the inputs for ML decoding.

Decoding of the Alamouti based STC signal are performed by using the I and Q channels and the estimated H channel. Two consecutive I and Q values are stored for each channel, and through a process of 8 complex multiplications (32 real multiplications); the original 2 symbols are determined. These multiplications are done using 4 multiplier blocks and hence are spread over 8 clock cycles in order to save and reuse resources. The two symbols are then unmapped from IQ back to bits and reassembled into the bitstream.

During normal transmission, the H matrix outputted by the ML estimator is compared to the actual H matrix used in the channel emulator.

2.2.3. Control Circuitry. In order to control the various logic functions, and to allow interactive processing of results, a softcore processor (Nios II) is instantiated inside the FPGA. This processor is configured to run at the same clock rate as the IF band processing module (100MHz), and the uCLinux operating system is used. UCLinux is selected due to its advanced networking functions, and flexibility. The processor acts as a gateway between the hardware modules and a PC, via Ethernet and a web based (HTTP) interface. This processor controls the input stream which is sent via each transmit module, and the channel matrix H used in the channel emulator. The transmitted data includes a training sequence (as described previously) followed by the message data. On the receive side, the processor allows the retrieval of stored data, which includes the estimated channel matrix H, and the received bit stream. Using the input and output bit stream a Bit Error Rate (BER) calculation can be performed.

2.2.4. Channel Emulator. In order to emulate the wireless channel, the channel matrix **H** representation is used as given in (1).

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1.1}$$

$$\begin{bmatrix} y_1(t) & y_1(t+1) \\ y_2(t) & y_2(t+1) \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1(t) & x_1(t+1) \\ x_2(t) & x_2(t+1) \end{bmatrix} + \begin{bmatrix} n_1(t) & n_1(t+1) \\ n_2(t) & n_2(t+1) \end{bmatrix} \tag{1.2}$$

Where **y** is the received signal vector, **x** is the transmitted signal vector, and **n** is the noise vector. Both **y** and **x** have two symbols period for the same channel matrix **H** due to the application of Alamouti scheme. Both real and imaginary parts of the signal need to be known. In order to synthesize the imaginary component, the λ/4 delayed signal is used. Due to the choice of λ /4 delay, the complex signal must be multiplied by **H***, where (·)* is denoted as the conjugate operation. Due to the real component of the signal being required after the channel emulator, the equations used for this block are shown in (2).

$$\begin{aligned} y_i(t) &= x_1(t) real(h_{i1}) + x_1(t-d) imag(h_{i1}) \\ &+ x_2(t) real(h_{i2}) + x_2(t-d) imag(h_{i2}) \end{aligned} \tag{2}$$

Note that d is a 90 degree phase delay of one symbol period. In our case, the symbol period is given by 32 samples, which includes two cycles of sine. Hence d is equal to 4 sample periods.

The channel matrix **H** is obtained using a signal scattering model described in [16]. In this model, the scattering environment is represented by a rectangular region of dimensions 200λ×200λ with transmitter and receiver equipped in MEA located on opposite sides of the rectangle. We assume that 600 scatterers are uniformly distributed within the rectangular region. Other distributions of scattering objects can also be covered by this model. For a single transmission path from transmit antenna j to receive antenna i, waves from the transmit antenna are intercepted and then reflected by scatterers. Summing the contributions from all scatterers, the channel matrix elements h_{ij} can be determined. The scattering coefficients are random variables, as described in [16].

Each channel matrix realization is stored into the buffer of the FPGA, in order to perform emulation of the wireless channel. The 1000 different channel matrices are saved in terms of real and imaginary parts. At this stage, the Alamouti scheme for a 2x2 MIMO system with the maximum likelihood (ML) technique for channel estimation is implemented.

With respect to the noise vector **n**, a uniformly distributed random number generator is used to select a random number from a table of pre-generated Gaussian distributed random numbers.

3. Results

In its current form, the MIMO testbed involves transmitting IF signals over a wire. A wireless channel between the transmitter and the receiver, based an indoor scattering signal model [16] is introduced in software to test 2x2 Alamouti transmission scheme. The current data rate is assumed to be 3.125Mbit per second. The signal magnitudes shown in Fig 3-5 use the normalized representation of the FPGA's fixed point numbers. An automatic gain control module applies amplification to the signal, when the signal is considered too small. Over the wire, a signal loss of around 3.5dB is observed.

The training sequence signal is shown in Fig. 3. It can be seen that the signal is designed to allow simple detection of symbol boundaries. When the STC modem is in training sequence mode, a signal is only transmitted from TX1, and received on both RX1 and RX2. RX1 is used to synchronize on these symbols. The choice of only transmitting training sequence on one transmitter, means that at the receive side, the training sequence is phase shifted and decreased in magnitude. Therefore the problem of synchronization in this mode can be solved like in a SISO system.

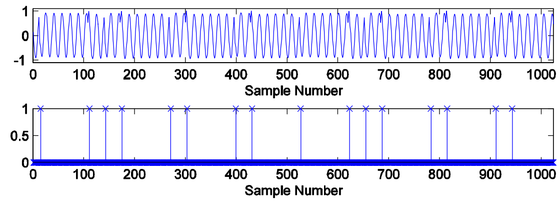
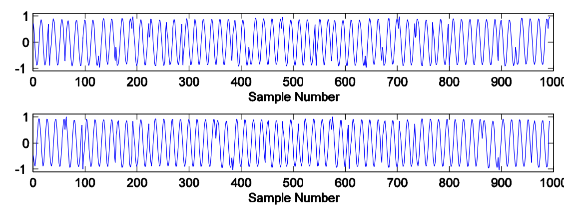


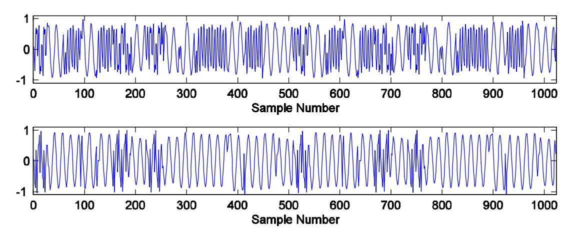
Fig. 3. Training sequence and symbol boundary detection.

Due to synchronization on the TX1 to RX1 signal (represented by h_{11} in the channel matrix), all terms of the channel matrix **H** are relative to h_{11}, when performing calculations at the receiver site.

After proper synchronization during the training sequence, the performance of space-time coding in a real implementation can be evaluated under different channel matrix realizations. The received signals under varying channel matrix data are shown in Fig. 4. The first of these is for an ideal channel (Channel matrix **H** is an identity matrix), and second concerns the data generated by the scattering model of [16].



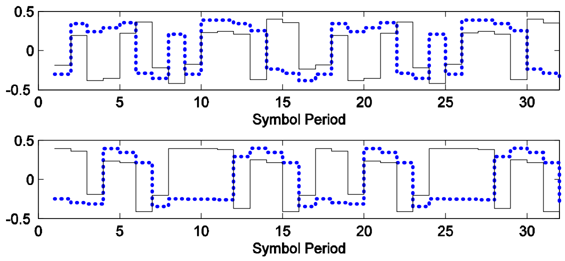
(a)



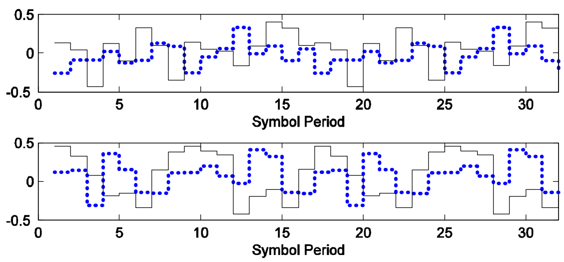
(b)

Fig. 4. The received signals for (a) an ideal identity matrix channel (b) an indoor scattering model [16].

The different stages of decoding the signal in Fig 4.b are presented in Fig 5. The first step involves decomposing the received signal into I and Q pairs. This is done by integrating over a symbol period, and storing the result.



(a)



(b)

Fig. 5. Signal decomposition into IQ (shown as dotted and solid respectively) for both received signals with an ideal channel with matrix **H** given as an identity matrix (b) an indoor scattering model [16].

Results shown in Fig. 4 and 5 indicate that the MIMO system is successfully implemented in FPGA hardware. The input and output bit streams are the same and all signals of I and Q channels are correctly identified.

For the Altera Stratix II FPGA, the complexity of the design can be measured in Advanced Lookup Tables (ALUTs). The FPGA has some other configurable logic components such as 9-bit DSP blocks, memory bits, phase lock loops (PLLs), and delay lock loops (DLL). The current resource usage in the FPGA for each QPSK modem is shown in Table 1.

Table 1: Resource usage for STC QPSK Modulator, Demodulator and Channel Emulator with and without NIOS II soft-core processor

Resource	IF Processing	IF + NIOS Processor	Total Available
ALUT	2453 (5%)	6000 (12%)	48,352
9-bit DSP Blocks	24 (9%)	32 (11%)	288
PLLs	1 (8%)	1 (8%)	12
Memory Bits	94,200 (3%)	596000 (23%)	2,544,192

It can be seen from Table 1 that the resource usage is quite small relative to the overall FPGA capabilities. The configuration of the NIOS II processor in this design is one classified as full featured, and when running at 100MHz the processor is capable of up to 113 DMIPS (a common software benchmark). In the full featured classification this processor includes hardware multiplication, 64Kbytes of on chip RAM, and instruction and data caches to improve performance. In future designs, only the hardware circuitry would need to be modified, meaning that there is plenty of room for future additions in terms of hardware processing.

In addition to BER calculations the system is also capable of comparing the estimated H matrix to the actual H matrix. This capability is demonstrated in the results presented in Fig. 6 where the noise effect on BER performance under the condition of perfect synchronization is studied. 10,000 bits are randomly sent to the channel emulator and the bit error rate (BER) at the receiving module is recorded. While implementing noise of a certain SNR in the channel emulator, each symbol is two periods of a sine wave. Perfect signal synchronization is established through the training sequence and manual intervention. Two cases of channel estimation are used, the first is perfect, and the second is using the ML estimation block. The results for perfect channel estimation shown in Fig. 6 match the known result of 2x2 STC MIMO systems at the receiver [17].

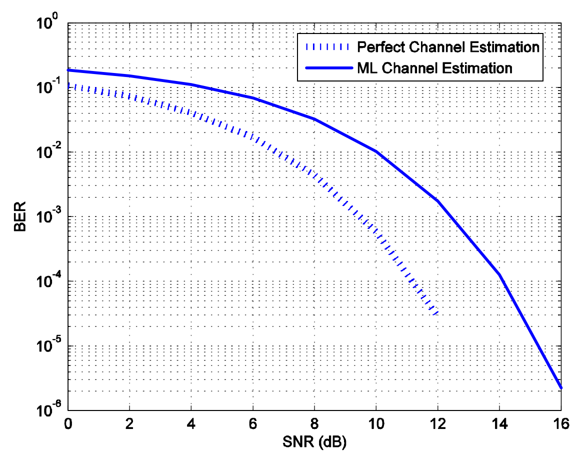


Fig. 6. BER characteristics for 2x2 MIMO Alamouti scheme under varying signal to noise ratios obtained with the FPGA based system.

As the noise can cause improper synchronization, leading to a further increase in BER, a MATLAB program was developed to investigate the impact of decoding misaligned symbols at the receiver. In the algorithm, symbols remained as constellation points, and were not synthesized as in the FPGA, making it easier to establish the relationship between symbol timing and BER. The effect on BER based on the percentage symbol timing error is shown in Fig. 7. It can be seen that small amounts of error (<10%) do not significantly impact BER. However larger errors greatly increase BER.

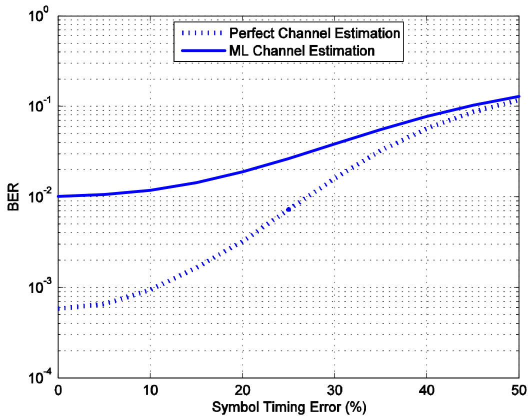


Fig. 7. BER performance due to symbol timing error.

4. Conclusion

In this paper the design and development of a full MIMO testbed, which involves FPGA for fast parallel signal processing has been presented. At the present stage, the 2x2 MIMO system for space time coding Alamouti scheme has been implemented. Its performance has been studied using an indoor channel emulator. The obtained results indicate that the developed testbed works properly with respect to processing of the IF signals. Therefore it shows a great promise to be of great assistance to test various MIMO transmission schemes in real environments when an RF circuity is incorporated.

It has to be noted that the RF circuitry including amplifiers, mixers and oscillators for use at 2.4GHz have already been purchased. Therefore it is expected that the assembling and testing process will take place in the near future.

5. Acknowledgments

The authors acknowledge the financial support of the Australian Research Council via Grant **DP0450118.** Also acknowledged is the Australian Research Council Communications Research Network (ACoRN).

6. References

[1] G. J. Foschini and M. J. Gans, "On limits of wireless communication in a fading environment when using multiple antennas", *Wireless Personal Communications*, vol. 6, no. 3, pp. 311-335, Mar. 1998.

[2] A. Paulraj, D. Gore, R. Nabar, and H. Bolcskei, "An overview of MIMO communications -A key to gigabit wireless," Proc. IEEE, vol. 92, no. 2, pp. 198-218, Feb. 2004.

[3] D. Gesbert, M. Shafi, D. Shiu, P. J. Smith and A. Najuib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 281-302, Apr. 2003.

[4] M. Jensen and J. W. Wallace, "A review of antennas and propagation for MIMO wireless systems," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 11, pp. 2810-2824, Nov. 2004.

[5] P.N. Fletcher, M. Dean and A.R. Nix, "Mutual coupling in multi-element array antennas and its influence on MIMO channel capacity," *Electronics Letters*, vol. 39, No. 4, pp. 342-344, Feb 2003.

[6] T. Svantesson and A. Ranheim, "Mutual coupling effects on the capacity of multiple antenna systems," in *Proc. IEEE ICASSP'01*, vol. 4, May 2001, pp. 2485–2488.

[7] R. Janaswamy, "Effect of element mutual coupling on the capacity of fixed length linear arrays," *IEEE Antennas and Wireless Propagation Letters*, vol. 1, pp. 157-160, 2002.

[8] J. W. Wallace and M. A. Jensen, "Mutual coupling in MIMO wireless systems: A rigorous network theory analysis," *IEEE Transactions on Wireless Communications*, vol. 3, no. 4, pp. 1317-1325, July 2004.

[9] J.W. Wallace and M.A. Jensen, "Statistical characteristics of measured MIMO wireless channel data and comparison to conventional models," *Proc. IEEE VTC*, vol. 2, 2001, pp. 1078-1082.

[10] K. Yu, M. Bengtsson, B. Ottersten, D. McNamara, P. Karlsson, and M. Beach, "Modeling of wide-band MIMO radio channels based on NLOS indoor measurements," *IEEE Trans. Veh. Technol.*, vol. 53, pp.655-665, May 2004.

[11] R. Stridh, K. Yu, B. Ottersten, and P. Karlsson, "MIMO channel capacity and modeling issues on measured indoor radio channel at 5.8 GHz," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 895-903, May 2005.

[12] V. Jungnickel, V. Pohl and C. V. Helmolt, "Experiments on the element spacing in multi-antenna systems," *Proc. IEEE VTC,* vol. 2, 2003, pp. 1124-1126.

[13] S.W. Ellingson,"A flexible 4x16 MIMO testbed with 250MHz-6GHz tuning range", in Proc. 2005 Intl. IEEE AP-S Symp., Washington, 3-8 July, 2005.

[14] K.S. Bialkowski, S. Zagriatski, A. Postula, and M.E. Bialkowski, "Indoor antenna diversity testbed", in Proc. IEEE Intl AP-S Symp., 3-8 July 2005, Washington, D.C.

[15] K.S. Bialkowski, S. Zagriatski, A. Postula, and M.E. Bialkowski, "MIMO Testbed with an insight into signal strength distribution around transmitter/receiver sites", in Proc. EuMW 2005, Paris, October 2005.

[16] P. Uthansakul, M. E. Bialkowski, S. Durrani, K. Bialkowski and A. Postula, "Effect of line of sight propagation on capacity of an indoor MIMO system," *in Proc. IEEE International Antennas and Propagation Symposium*, Washington DC, USA, July 3-8, 2005.

[17] B. Vucetic and J. Yuan, *Space-Time Coding.* Chichester, U.K.: Wiley, 2003.