

FPGA Implementation of a Real Time Maximum Likelihood Space-Time Decoder on a MIMO Software Radio Test Platform

Peter J. Green and Desmond P. Taylor

Department of Electrical and Computer Engineering

University of Canterbury

Christchurch, New Zealand

Email: drgreenpeter@gmail.com

Email: taylor@elec.canterbury.ac.nz

Abstract—This paper describes the concept, architecture, development and demonstration of a real time, maximum likelihood Alamouti decoder for a wireless 4-transmit 4-receiver multiple input and multiple output (MIMO) Smart Antenna Software Radio Test System (SASRATS) platform. It is implemented on a Xilinx Virtex 2 Pro Field Programmable Gate Array (FPGA). Hardware, firmware, use of the Xilinx Core Generator Intellectual Property modules and experimental verification of the decoder are discussed.

Keywords—real-time implementation; Alamouti; FPGA; maximum likelihood decoder; MIMO; software radio test platform

I. INTRODUCTION

The proposed system implementation is developed on an existing MIMO Smart Antenna Software Radio Test System (SASRATS) platform [1], [2] designed to test and verify various space time architectures and algorithms. The 4 receivers complement a 4-transmitter space time (ST) encoding platform [3] designed and developed for real-time testing of ST coding schemes developed by Alamouti [4] and others mentioned in [5]. The primary objective is to increase system capacity and performance through the use of multiple antennas, employing spatial multiplexing and ST coding and decoding. Spatial multiplexing and diversity techniques are currently adopted in the IEEE 802.11n draft specification to fully exploit the benefit of MIMO channels.

The focus of this paper is on the digital baseband portion of the system, particularly the real-time implementation of the Alamouti decoder on a Xilinx Virtex 2 Pro FPGA. Other MIMO testbeds [6], [7] typically perform post processing operations such as channel estimation and Alamouti decoding in Matlab after capturing large batches of data. Real-time implementation of a 2×1 Alamouti decoder was briefly described by [8]. Our work describes in detail, the real-time implementation of a maximum likelihood 2×2 Alamouti decoding implementation extending to a 2×4 system on the SASRATS platform.

II. OVERVIEW OF THE SASRATS ARCHITECTURE

The basic architecture of the SASRATS receivers is shown in Figure 1. The analogue portion amplifies, translates and

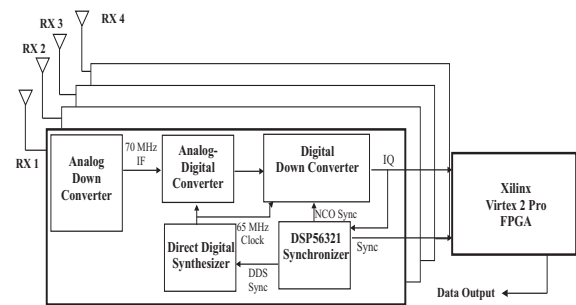


Figure 1. SASRAT 4 receiver system architecture with the Xilinx FPGA channel estimator

filters a received radio frequency signal at 915 MHz or 2.4 GHz to an intermediate frequency of 70 MHz where digitization and bandpass sampling occurs. The output of the analog to digital converter is then fed into a digital down converter which digitally downconverts, decimates and filters the input data to produce baseband in-phase (I) and quadrature phase (Q) signals for further processing. The SASRATS receivers work asynchronously with the transmitters and we have developed and implemented real-time algorithms for carrier and symbol timing synchronization [9] and also channel estimation operations [10] in DSP and FPGA.

We adopt a feedforward approach through the use of known training symbols (data-aided) or preambles at the transmitter to resolve magnitude and phase ambiguities in Rayleigh flat fading channels. We assume that the channels change only *slowly* during the period between training preambles. A portion of the FPGA performs real-time channel estimation as described in [10]. In a 4-transmitter and 4-receiver (4×4) MIMO system, each receiver must estimate 4 distinct channels with a total of 16 channel estimates for 4 receivers.

III. OVERVIEW OF ALAMOUTI SCHEME

The Alamouti scheme is the only orthogonal space-time block code using complex signals for two transmit antennas which provides full diversity of 2 and full rate of 1. For more

than two transmit antennas, the goal is to design transmission codes that achieve full diversity at the highest possible rate with low decoding complexity.

In our 2×2 MIMO implementation, we use two distinct training codes over 2 time multiplexed preamble slots at the transmitter. When one transmitter is sending training data in one time slot, the other is off. These 26-bit preambles are GSM training sequence codes (TSC) 0 and 1 [11]. The two transmitters then transmit 128 space-time encoded data symbols simultaneously before the cycle repeats. At the transmitter, the SASRATS transmitters are programmed to run a 2 transmit Alamouti encoding scheme, where two symbols, s_0 and s_1 , are transmitted simultaneously from two transmitters at time instant t . At time instant $t + T$, the symbols $-s_1^*$ and s_0^* are transmitted simultaneously from the transmitters where $*$ represents the complex conjugate. The transmission matrix is represented by

$$\mathbf{S} = \begin{bmatrix} s_0 & s_1 \\ -s_1^* & s_0^* \end{bmatrix} \quad (1)$$

The transmitted symbols travel through 2 independent channels h_0 and h_1 to a receiver where noises n_0 and n_1 are added to the received signals. h_0 and h_1 are complex multiplicative distortions assumed constant across two consecutive symbols. This is depicted in Figure 2.

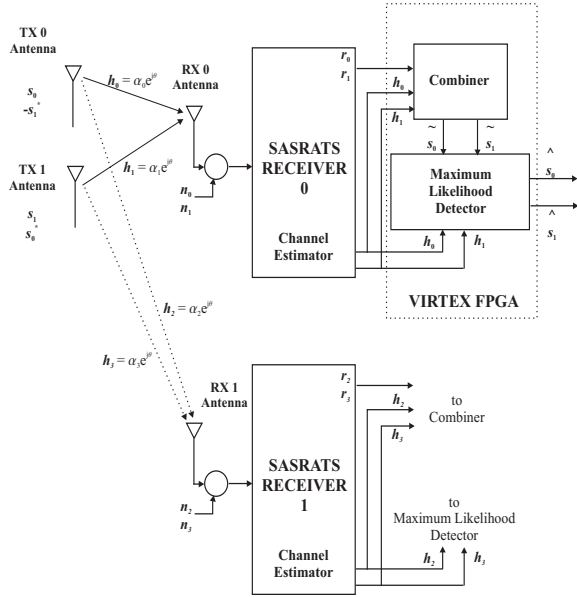


Figure 2. Block diagram of Alamouti decoding implementation on SASRATS platform

It is shown in [4] that at the input of the combiner, the receive signals are given by

$$\begin{aligned} r_0 &= r(t) = h_0 s_0 + h_1 s_1 + n_0 \\ r_1 &= r(t + T) = -h_0 s_1^* + h_1 s_0^* + n_1 \end{aligned} \quad (2)$$

In our implementation[10], a real-time FPGA based channel estimator produces the estimates \hat{h}_0 and \hat{h}_1 and this information is fed to the combiner to yield two combined output signals

$$\begin{aligned} \tilde{s}_0 &= \hat{h}_0^* r_0 + \hat{h}_1 r_1^* \\ \tilde{s}_1 &= \hat{h}_1^* r_0 - \hat{h}_0 r_1^* \end{aligned} \quad (3)$$

The signals \tilde{s}_0 and \tilde{s}_1 are sent to the maximum likelihood (ML) detector so that ML estimates \hat{s}_0 and \hat{s}_1 can be made of s_0 and s_1 . As we use PSK modulation of the symbols at the transmitter (equal energy constellations), the ML detector does not need channel estimates and the decision rule in the ML detector is simplified to choose s_i iff $d^2(\tilde{s}_0, s_i) \leq d^2(\tilde{s}_0, s_k), \forall i \neq k$ for \tilde{s}_0 and choose s_i iff $d^2(\tilde{s}_1, s_i) \leq d^2(\tilde{s}_1, s_k), \forall i \neq k$ for \tilde{s}_1 where $d^2(x, y)$ is the squared Euclidean distance between signals x and y .

The complexity of the combiner and ML detector depends on type of modulation. Binary phase shift keyed (BPSK) symbols are the simplest to detect. Detection of non equal energy modulation schemes require channel estimates in the ML detector and has higher complexity. The present work considers BPSK and QPSK implementations only.

Implementation of a MIMO 2 transmitter and 2 receiver Alamouti system, requires the estimation of 4 channels ($\hat{h}_0, \hat{h}_1, \hat{h}_2$ and \hat{h}_3), 2 at each receiver as shown in Figure 2. In this situation, the output of combiner yields 2 outputs

$$\begin{aligned} \tilde{s}_0 &= \hat{h}_0^* r_0 + \hat{h}_1 r_1^* + \hat{h}_2^* r_2 + \hat{h}_3 r_3^* \\ \tilde{s}_1 &= \hat{h}_1^* r_0 - \hat{h}_0 r_1^* + \hat{h}_3^* r_2 - \hat{h}_2 r_3^* \end{aligned} \quad (4)$$

where \hat{h}_2 and \hat{h}_3 are channel estimates from the second receiver. In the case of a 2×2 Alamouti implementation using PSK signals, the ML decoder remains unchanged except for the combiner. As seen from (4), the combiner output \tilde{s}_0 is actually the sum of \tilde{s}_0 from receiver 0 and \tilde{s}_0 from receiver 1. Likewise, \tilde{s}_1 is actually the sum of \tilde{s}_1 from receiver 0 and \tilde{s}_1 from receiver 1. Thus a $2 \times M$ Alamouti implementation can be easily implemented by summing together the appropriate combiner outputs from M receivers before feeding one ML detector.

In an extended version of Alamouti for 4 transmitters [12], full rate is achieved but the system is half rank (quasi-orthogonal) with some loss in diversity as transmitted symbols cannot be fully decoupled. Tarokh's STBC scheme [13] for 4 transmitters on the other hand, achieves complete orthogonality at half the full rate. Tarokhs scheme suffers no loss in diversity and receiver decoding is simpler as the transmitted symbols can be fully decoupled.

The decoding of the Alamouti encoded signals is a linear process and our SASRATS receiver system design implements the combiner and maximum likelihood detection on the Xilinx Virtex 2 Pro FPGA board using the *Xilinx Integrated System Enviroment (ISE) Foundation* design tool.

At the SASRATS receivers, the I and Q outputs are fed into a Xilinx University Program Virtex 2 Pro Development System board based on the Virtex 2 Pro XC2VP30 with 30,816 logic cells. This low cost development board from Digilent Inc. has four 20-bit wide ports which are ideal for our 4 receiver system.

The complete design is implemented using a top down hierarchical schematic entry approach on the *Xilinx Integrated System Enviroment (ISE) Foundation* design tool. VHDL code can also be integrated as a block with other schematic components if desired. We have also made extensive use of various *Xilinx Core Generator* intellectual property(IP) modules incorporated within the ISE Foundation toolset to shorten design cycle time.

IV. IMPLEMENTATION OF THE ALAMOUTI COMBINER AND ML DECODER

We begin by first describing the overall architecture of the Alamouti 2×1 decoding scheme for QPSK modulated received symbols as shown in Figure 3.

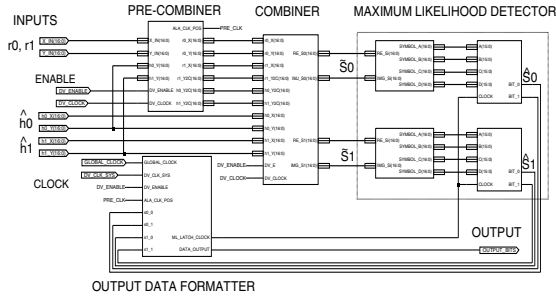


Figure 3. Block diagram of the Alamouti combiner and maximum likelihood detector implementation on the SASRATS receiver platform

The architecture consists of several blocks; the pre-combiner, combiner, ML detector and output data formatter. The inputs into the pre-combiner block consist of 16-bit I and Q data and channel estimates \hat{h}_0 and \hat{h}_1 which remain static for the duration of 128 data symbols. On receipt of the Data_Valid (DV) pulse from the channel estimator, the pre-combiner circuitry latches to capture r_0 and r_1 over two symbol periods and calculates the complex conjugates of \hat{h}_0 , \hat{h}_1 and r_1 needed in the combiner. This is achieved by performing a two's complement operation on the imaginary parts of \hat{h}_0 , \hat{h}_1 and r_1 using the Xilinx Two's Complement IP module.

The combiner block as shown in Figure 4 calculates \tilde{s}_0 and \tilde{s}_1 . The product terms $\hat{h}_0^* r_0$, $\hat{h}_1^* r_1^*$, $\hat{h}_1^* r_0$ and $\hat{h}_0^* r_1^*$ are first calculated in 4 separate Xilinx Complex Multiplier v2.0 IP blocks. The product terms $\hat{h}_0^* r_0$ and $\hat{h}_1^* r_1^*$ are then summed to compute \tilde{s}_0 . The signal \tilde{s}_1 is then formed by taking difference between $\hat{h}_1^* r_0$ and $\hat{h}_0^* r_1^*$ by two properly configured Xilinx Adder/Subtractor v7.0 IP cores respectively.

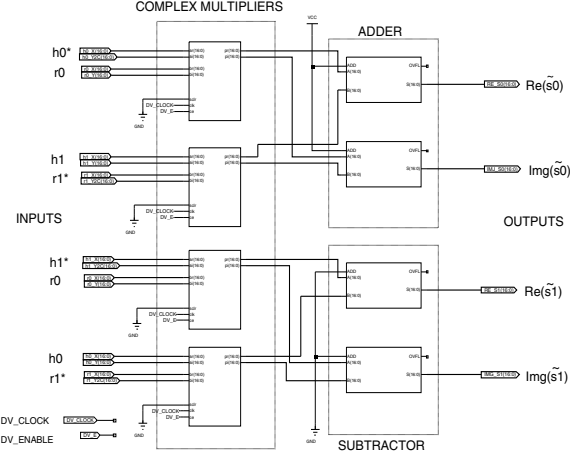


Figure 4. Block diagram of the combiner

The outputs \tilde{s}_0 and \tilde{s}_1 , are then fed into the maximum likelihood (ML) detector processing block. The ML block consist of 2 parallel and independent sets of Euclidean distance calculators and minimum distance comparators as shown in Figure 3 where the decision statistics, \tilde{s}_0 and \tilde{s}_1 are processed independently.

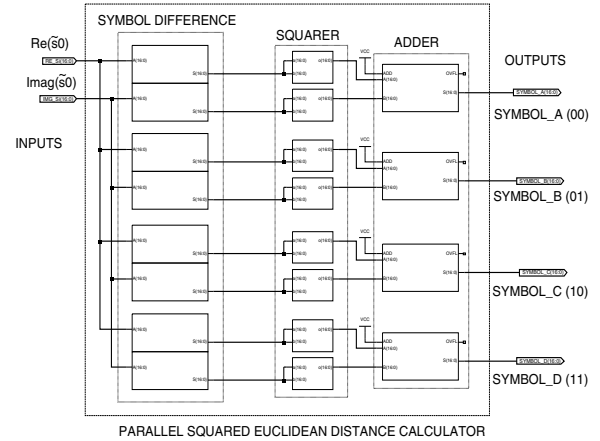


Figure 5. Block diagram of the squared euclidean distance block in the ML detector

The Euclidean distance calculator block shown in Figure 5 first calculates in parallel, the difference between the symbol decision statistic and 4 prestored QPSK symbols ($\pm 0.707 \pm j0.707$). The real and imaginary parts of each symbol are then squared and added together. The 4 squared Euclidean distance outputs (A,B,C and D) are fed into the minimum Euclidean distance comparator block shown in Figure 6.

The minimum Euclidean distance comparator is implemented using 6 two-input magnitude comparators. There are two outputs ($x > y, x < y$) from each comparator. The outputs from the various comparators are AND'ed together

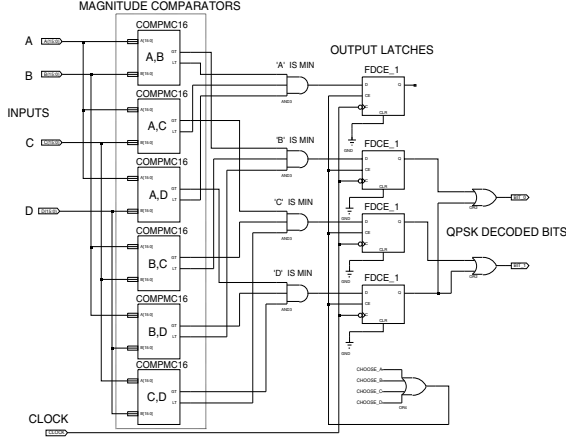


Figure 6. Block diagram of the minimum distance comparator block in the ML detector

and latched based on the following minimum magnitude selection criterion which chooses A iff $(A < B) \& (A < C) \& (A < D)$, B iff $(A > B) \& (B < C) \& (B < D)$, C iff $(A > C) \& (B > C) \& (C < D)$, D iff $(A > D) \& (B > D) \& (C > D)$. Only one of the four outputs goes high when the criterion is met. Then the latched outputs are fed into two OR gates to decode the estimated QPSK symbol into bits. Thus each magnitude comparator for \hat{s}_0 or \hat{s}_1 has one 2-bit output which represents either 00, 01, 10 or 11.

The output data formatter places the bit estimates of \hat{s}_0 and \hat{s}_1 in the correct time position resulting in a continuous serial bit output which can be stored and checked against the original serial bit stream sent at the transmitter for bit error rate measurements. The system outputs 4 bits for every pair of QPSK symbols received.

The Alamouti 2×2 decoding scheme on our testbed is implemented by duplicating the pre-combiner and combiner blocks for the second receiver where the combiner outputs of both receivers are summed together in a multi-receiver summer block as defined by (4), as shown in Figure 7, to form the new combined output \hat{s}_0 and \hat{s}_1 prior to ML detection. The same process is repeated for the Alamouti 2×3 and 2×4 schemes. In all cases, only one ML detector block is needed.

The same modular approach can be used to implement Tarokh's 4×1 orthogonal STBC [13] with some extensions to the receiver pre-combiner, combiner and ML detector design. Applying Tarokh's theory of complex generalised orthogonal designs [13] to a 4×4 scheme for example, requires the pre-combiner to store sets of 8 received symbols and 4 channel estimates per receiver prior to combining, ML detection and estimation of 4 symbols. Implementation is beyond the scope of this paper, but is straightforward.

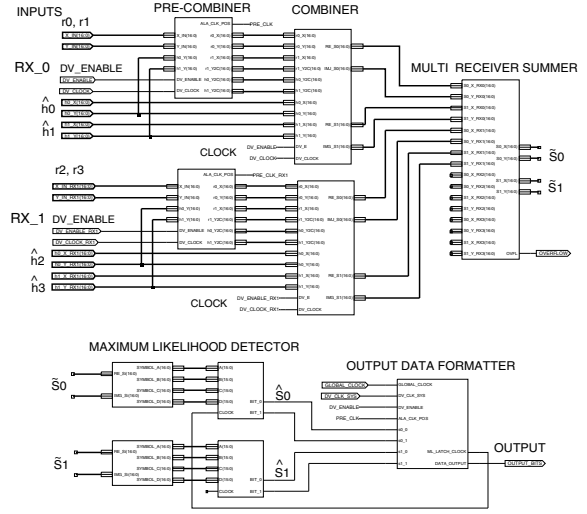


Figure 7. Block diagram of the 2 X 2 Alamouti implementation

V. EXPERIMENTAL VERIFICATION OF THE ML DETECTOR

The first experiment to test the operation of the ML detector is performed on a 2×1 setup of the SASRATS platform as shown in Figure 8. TX0 and TX1 each transmit time

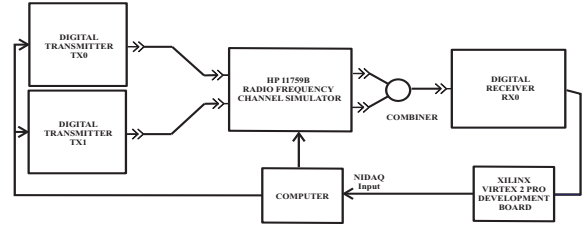


Figure 8. SASRATS setup with HP11759B for channel estimation verification

multiplexed GSM preambles TSC codes 0 and 1 respectively between data frames at 915 MHz. The two transmitters are programmed to transmit Alamouti space-time encoded data during the data frame. The modulation is BPSK and symbol rate is 1500 kbaud. In this experiment, a Hewlett Packard 11759B radio frequency channel simulator is programmed to generate two independent uncorrelated Rayleigh flat fading channels. 1048 kbit estimates from the output of the ML detector are captured by the NIDAQ card and compared with the actual transmitted bits in Matlab. Alamouti [4] assumes that they are constant across two consecutive symbols but in a practical implementation, this requirement is difficult to meet. In our experiment, it is assumed constant across the entire data frame of 128 symbols. We have verified in Matlab that at an average SNR value of 25 dB, there are virtually no errors proving the correct operation of the entire

system on FPGA. The experiment was repeated with QPSK modulation of data symbols with similar results.

To test operation with more than one receiver, the SASRATS platform is reconfigured into a 2×2 MIMO system with the HP11759B removed as the channel simulator cannot generate more than 2 Rayleigh fading channels. In the lab, 4 antennas spaced sufficiently apart are connected to the transmitter outputs and receiver inputs. It was found that under these conditions, the channels at 915MHz are highly correlated and experience almost no fading. We are able to process the information from the FPGA to confirm operation of the ML space-time decoder. At a high SNR value of 30 dB, there are no errors despite highly correlated channels using both BPSK and QPSK modulated symbols. We have also tested the SASRATS platform configured as a 2×4 Alamouti MIMO system with excellent performance.

VI. CONCLUSIONS

We have described the implementation of a real time maximum likelihood Alamouti decoder for use on our MIMO platform implemented on an FPGA using the Xilinx ISE tool and Core Generator IP modules. We have also experimentally verified the operation of the decoder in a closed Alamouti 2×1 diversity scheme using an RF channel simulator and also in an open 2×2 and 2×4 antenna based system under correlated channel conditions.

REFERENCES

- [1] P. Green and D. Taylor, "Smart antenna software radio test system," *Proceedings of the First IEEE International Workshop on Electronic Design, Test and Applications.*, vol. 1, pp. 68–72, Jan. 2002.
- [2] —, "Experimental verification of space-time algorithms using the smart antenna software radio test system (sasrats) platform," *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, vol. 4, pp. 2539–2544, 2004.
- [3] —, "Implementation of a high speed four transmitter space-time encoder using field programmable gate array and parallel digital signal processors," *Proceedings of the Third IEEE International Workshop on Electronic Design, Test and Applications.*, pp. 466–471, Jan. 2006.
- [4] S. Alamouti, "Space block coding: A simple transmitter diversity technique for wireless communications," *IEEE J. Select. Areas. Communication*, vol. 16, pp. 1451–1458, Oct. 1998.
- [5] D. Gesbert *et al.*, "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.
- [6] R. M. Rao *et al.*, "Multi-antenna testbeds for research and education in wireless communications," *IEEE Communications Magazine*, vol. 42, no. 12, pp. 72–81, 2004.
- [7] S. Caban *et al.*, "Vienna MIMO testbed," *EURASIP Journal on Applied Signal Processing*, vol. 54868, pp. 1–13, 2006.
- [8] P. F. P. Murphy and C. Dick, "An fpga implementation of alamoutis transmit diversity technique," *University of Texas WNCG Wireless Networking Symposium*, Oct. 2003.
- [9] P. Green and D. Taylor, "Implementation of four real-time software defined receivers and a space-time decoder using xilinx virtex 2 pro field programmable gate array," *Proceedings of the Third IEEE International Workshop on Electronic Design, Test and Applications.*, pp. 89–92, Jan. 2006.
- [10] —, "Implementation of a real-time multiple input multiple output channel estimator on the smart antenna software radio test system platform using the xilinx virtex 2 pro field programmable gate array," *Proceedings of the 2006 IEEE International Conference on Field Programmable Technology.*, pp. 257–260, Dec. 2006.
- [11] ETSI/GSM, "Multiplexing and multiple access on the radio path," *GSM Recommendations Document 05.02 Version 3.8*, Dec. 1995.
- [12] M. Rupp and C. Mecklenbrauker, "On extended alamouti schemes for space-time coding," *Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on*, vol. 1, pp. 115–119, Oct. 2002.
- [13] V. Tarokh, H. Jafarkhani, and A. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1456–1467, 1999.