A Low Power Pipelined Maximum Likelihood Detector for 4x4 QPSK MIMO Wireless Communication Systems

J. H. Han[1], A.T. Erdogan[1,2], T. Arslan[1,2]
[1]The University of Edinburgh, School of Engineering and Electronics
Edinburgh, EH9 3JL, UK
[2]Institute of System Level Integration, The ALBA campus
Livingston, EH54 EG, UK
j.han@ed.ac.uk, Ahmet.Erdogan@ee.ed.ac.uk, Tughrul.Arslan@ee.ed.ac.uk

Abstract

The authors present a maximum likelihood (ML) detector for multiple-input multiple-output (MIMO) wireless communication systems. The ML detector has been specifically designed to reduce the implementation complexity without significant degradation in bit error rate (BER) performance. In order to identify the optimized fixed-point representation, the ML detector has been simulated with various representations for the received data. The computation process of the channel matrix and constellation symbols in ML detector is simplified by using normalized symbols. Simulation results are provided showing 42% saving in area usage and 68% saving in power consumption compared to a conventional architecture.

1. Introduction

High throughput is one of the key issues in wireless communication systems. Multiple-input multiple-output (MIMO) systems provide a breakthrough for achieving high data rates for wireless communication systems such as 3GPP, WiMax, and WLAN [1]. Since the introduction of a simple space-time diversity technique in [2], many researchers have studied MIMO systems in order to improve their performance in terms of bit error rate (BER) and capacity. Various algorithms for MIMO channel detection have been proposed in literature to reduce their complexity for practical applications. Maximum likelihood (ML) algorithm can provide the best BER performance for MIMO systems, while the number of searching steps increases exponentially with the number of receiver antenna (M) and the modulation method (Q), which is given by 2^{MQ}. Several methods are suggested to reduce the number of searching steps in the literature. Sphere decoder method [3] can reduce the searching steps without significant performance degradation as compared to ML method. VBLAST method with nulling and cancellation [4] has significantly reduced the searching steps with a coast of less BER performance than ML method.

Although the number of searching steps exponentially increases with the number of antennas and constellation, ML method can still provide lower complexity implementations compared to other methods mentioned above. For example, ML does not need complex matrix computations such as matrix inversion and decomposition [5], which are necessary in Sphere decoder and VBLAST methods.

In this paper, the authors present a fixed-point ML detector implementation for a 4x4 QPSK MIMO system. The ML detector has been implemented with various quantization levels for received symbols in order to investigate its BER performance and identify the optimum quantization level. In practical digital signal processing applications, fixed-point implementations are commonly used since they directly contribute to area and power savings. Moreover, a detection algorithm has been suggested in order to reduce the complexity of the ML detector by reducing the number of multiplications. The implemented ML detector has also been compared with the conventional and other methods in the literature in terms of BER performance, area usage and power consumption.

2. ML detection in MIMO systems

Fig. 1 shows a simple diversity diagram of a MIMO system where multiple antennas are used at transmitter

Proceedings of the 2006 Emerging VLSI Technologies and Architectures (ISVLSI'06)
0-7695-2533-4/06 $20.00 © 2006 IEEE

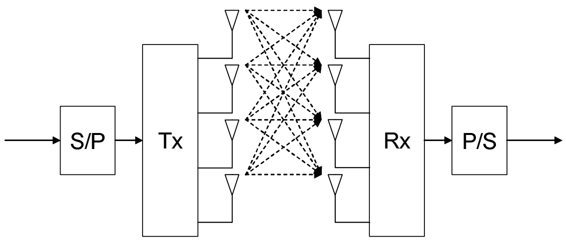


Figure 1. A simple diagram of a MIMO system

and receiver. The MIMO system can be represented as follows :

$$\mathbf{y} = \mathbf{Hx} + \mathbf{n} \qquad (1)$$

where $\mathbf{y} = [y_1, \ldots, y_M]^T$ is the vector of received symbols, **H** is the M (the number of Tx antenna) x N (the number of Rx antenna) channel matrix, $\mathbf{x} = [x_1, \ldots, x_N]^T$ is the vector of transmitted symbols, and **n** is additive white Gaussian noise (AWGN). The transmitted symbol vector, **x**, is represented by a modulated symbol, such as QPSK and QAM. In this paper, QPSK modulation, $s \in \{1+i, 1-i, -1-i, -1+i\}/\sqrt{2}$, which are generated by following Gray coding method, is used and **H** is assumed to be known at receiver.

The transmitted symbols, **x**, can be obtained by calculating a minimum Euclidian distance from the received symbols, **y**, the channel matrix, **H**, and the modulation symbols, $\mathbf{s} = [s_0, \ldots, s_N]^T$. Therefore, the ML detection algorithm can be represented with the equation below:

$$\arg\min \|\mathbf{y} - \mathbf{Hs}\|^2 \qquad (2)$$

A straightforward approach to solve Eq. (2) is an exhaustive search. However, the corresponding computational complexity grows exponentially with the number of antennas and the number of bits per symbol in the constellations. For example, for a 4x4 MIMO system with QPSK modulation, the required number of searching steps is 256, which is regarded as a limit for the use of ML algorithm for MIMO systems.

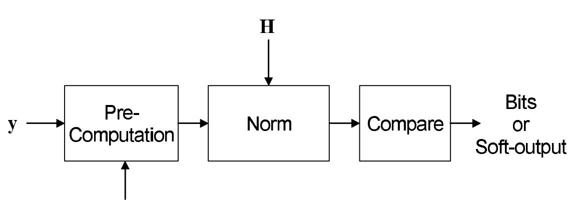


Figure 2. A block diagram of a ML detector

3. Fixed-point implementation

Fig. 2 illustrates the main blocks in a ML detector implementation. The received symbols, **y,** and the channel matrix, **H**, are represented in a fixed-point format (*t.p*) where *t* is the total number of bits, and *p* is the number of precision bits. The optimization of the number of total and precision bits is crucial in hardware implementations for saving area and power consumption.

Before describing the simulation results of fixed-point ML detector implementation, Eq (2) is slightly modified in order to reduce the matrix computations for **H** and **s**, as shown below:

$$\|\mathbf{y} - \mathbf{Hs}\|^2 = \|\mathbf{y} - \mathbf{H}(\mathbf{2^{-1/2}} \times \mathbf{s'})\|^2 = 2^{-1} \times \|\mathbf{y'}\text{-}\mathbf{Hs'}\|^2 \quad (3)$$

$$\arg\min \|\mathbf{y'}\text{-}\mathbf{Hs'}\|^2 \qquad (4)$$

where **s'** is $[s_0', \ldots, s_N']^T$, in which s' $\in \{1+i, 1-i, -1-i, -1+i\}$. Therefore, the matrix computation of **Hs'** can be implemented without using any multipliers. However, the computation of **y'** will require multiplying **y** with a parameter, w ($=\sqrt{2} \times 2^p$), which is determined by the number of precision bits, *p*, and the modulation method for **y**. Although, Eq. (4) requires multiplications for computing **y'**, it can still lead to less complexity when compared to Eq. (2), where the computation of **Hs** requires more multiplications compared to the multiplications required for obtaining **y'**. Moreover, in our implementation, there is no need for a memory block for storing the multiplication results, **Hs**, as in [6].

Fig. 3 illustrates BER performance with different fixed-point representations for E_b/N_0 of 5, 10, 15, and 20. The numbers inside the brackets represent the number of total and precision bits (*t.p*). For example, (8.5) denotes that the number of total and precision bits is 8 and 5, respectively. As can be seen, the BER performance gets better as the total number of bits increases. However, the performance is strongly affected by the number of precision bits. For example, *p*=3 provides the best BER when *t*=6. Therefore, we have simulated our ML detector with the best (*t.p*) combinations. Fig. 4 shows the BER performance versus E_b/N_0 for (6.3), (7.4), (8.4), (9.5), and (10.5). As can be seen from Fig. 4, the BER performance does not change significantly until Eb/N_0 is 10. In overall, a fixed-point representation of (9.5) can provide good

0-7695-2533-4/06 $20.00 © 2006 IEEE

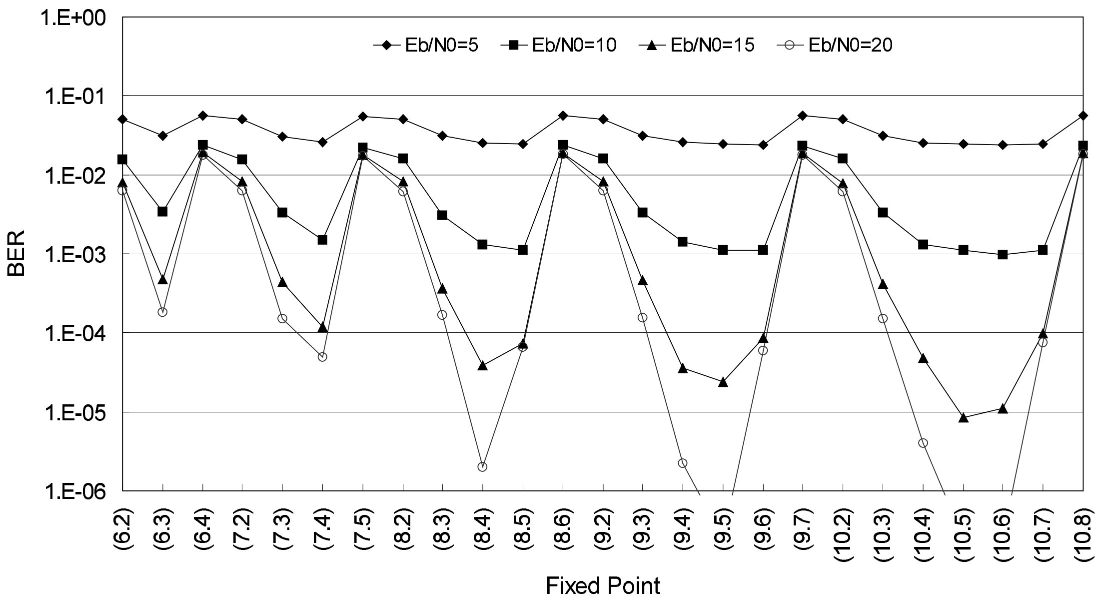


Figure 3. BER performance for different fixed-point values with Eb/N0=5, 10, 15, and 20

BER performance without any significant performance degradation.

4. Complexity reduction

The ML detector algorithm in Eq (2) represents a simple form for implementation. Although the complexity of the ML algorithm is less compared to other detector algorithms, implementing the ML detector still requires many multiplications and additions for computing the norm value. This results in increased area usage, power consumption, and critical path delay for practical implementations. In [7], a low complexity ML detector implementation is proposed where the required number of multiplications and additions are also provided. However, the detector proposed in [7] is not suitable for practical wireless communications systems since it can only be used for constellations s ∈ {1, i, -i, -1} with QPSK modulation. On the other hand, an approximation method for calculating the norm value has been introduced in [8], as shown below:

$$\|A+iB\|^2 \approx [\max(|A|, |B|)+0.5\min(|A|,|B|)]^2 \quad (5)$$

This approximation can reduce the complexity of ML detector implementation. However, it still requires a number of multipliers and adders. To reduce the complexity further, in this paper, Eq. (4) is transformed as follows:

$$\arg\min\left[\mathrm{Re}\left|\mathbf{y'}\text{-}\mathbf{Hs'}\right| + \mathrm{Im}\left|\mathbf{y'}\text{-}\mathbf{Hs'}\right|\right] \quad (6)$$

As can be seen, this equation does not require any multiplications, and hence it can reduce the implementation complexity of the ML detector much more compared to other methods suggested in literature. To verify its functionality, the BER performance of the ML detector based on Eq. (6) has been analyzed and compared with the BER performance achieved with the conventional method based on Eq. (2), as shown in Fig.

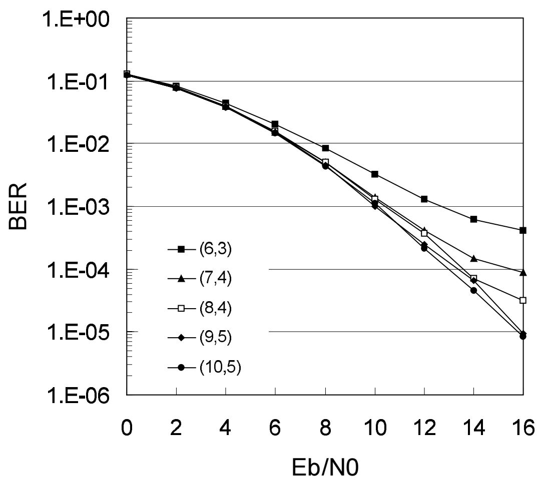


Figure 4. BER performance for different fixed-point implementations

Proceedings of the 2006 Emerging VLSI Technologies and Architectures (ISVLSI'06)
0-7695-2533-4/06 $20.00 © 2006 **IEEE**

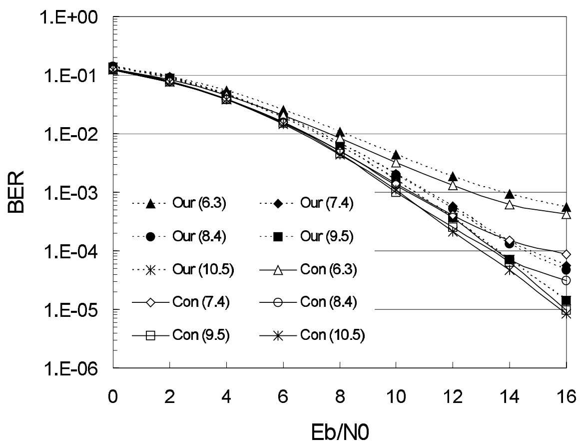


Figure 5. BER performance comparisons

5 where 'CON' and 'OUR' denote the results of ML detector implemented with Eq. (2) and Eq. (6), respectively. As can be seen, there is no significant degradation in BER performance between 'CON' and 'OUR'. Therefore, a less complex ML detector implementation can be achieved by using Eq. (6).

5. Implementation of maximum-likelihood detector

In previous sections, the ML detector has been investigated and simulated with various fixed-point representations to find the optimized quantization level for the received symbols, **y**, and the channel matrix, **H**. Fig. 6 illustrates a block diagram of the ML detector implemented in this paper for a 4x4 QPSK MIMO system. The architecture consists of five pipelined blocks which are named as pre-computation and norm1 (PCN1U), norm2 (N2U), norm3 (N3U), norm-summation (NSU), and decision (DEU). Here, the first three units are used to compute the real and imaginary

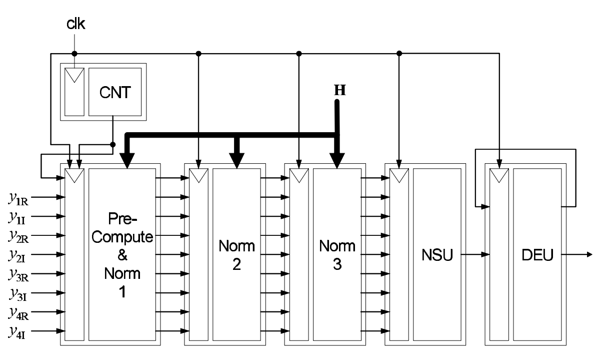


Figure 6. A block diagram of Maximum Likelihood detector implementation

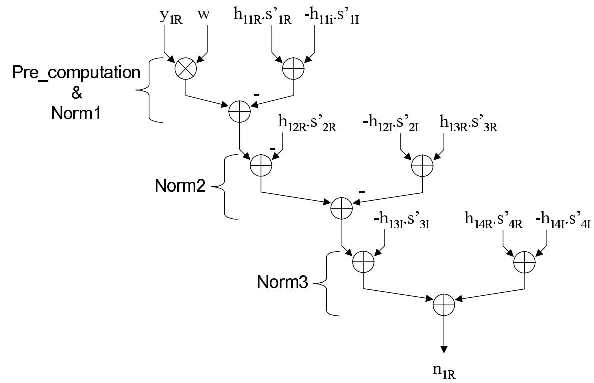


Figure 7. Pipelined data path for the computation of the norm value.

parts of **y'-Hs'** for each received symbol, as shown in Fig. 7. These are then summed by the NSU to generate the norm value. This process is repeated 256 times after which the minimum value is finally chosen by the decision unit (DEU). In Fig. 6, the counter (CNT) is used to provide all 256 symbol sequences.

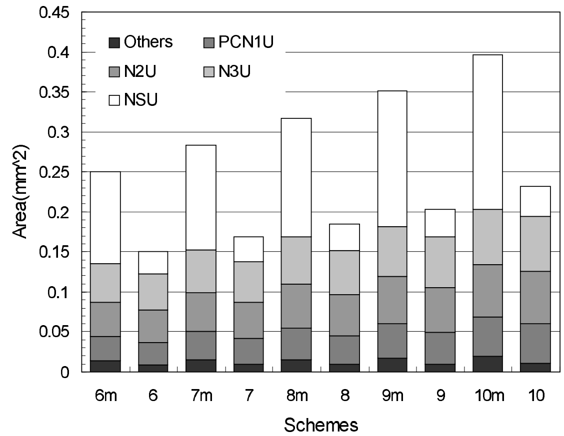
In our implementation, the NSU block has been implemented with and without using multipliers in order to investigate the impact of multiplier elimination in terms of area usage and power consumption.

6. Simulation results

The ML detector has been implemented as Verilog HDL and synthesized to a 0.18um standard cell library with Synopsys DesignCompiler™. RTL and gate level simulations have been performed using Cadence Verilog-XL™. The results of the power consumption were obtained with Synopsys PowerCompiler™ based on gate level simulations with a clock speed of 100MHz.

As illustrated in Fig. 8 and 9, our ML detector has been implemented for different fixed-point representations for evaluating its area usage and power consumption. In these figures, 6m and 6 denote to 6 bits word length for **y** and **H** with and without multiplier for computing the *norm* value in the ML detector, respectively.

Clearly, the ML detector implementations without multiplier can save more area and power than with multiplier. The main contribution for the savings is due to NSU block. Particularly, power consumption savings is more significant than the area usage savings. For example, with a word length of 9-bits for **y** and **H**, the ML detector without multiplier can save 42% area usage and 68% power consumption compared to the ML detector with multiplier.



Figur 8. Area results of the ML detector with and without multiplier for different fixed-point implementations

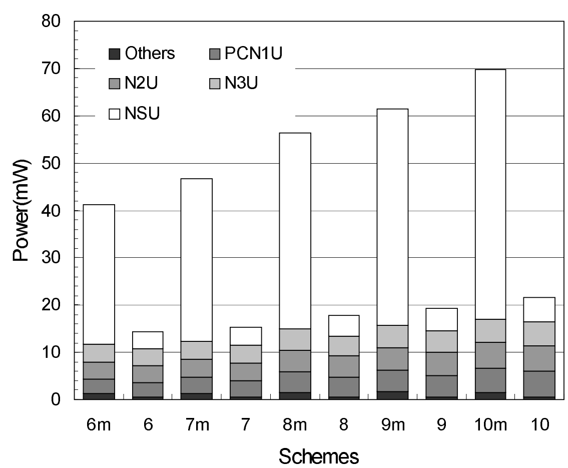


Figure 9. Power simulation results of the ML detector with and without multiplier for different fixed-point implementations

7. Conclusions

The authors have presented a fixed-point ML detector for 4x4 QPSK MIMO systems. The ML detector has been simulated with various quantization levels for input data in order to find an optimized fixed-point representation. The authors have also proposed an implementation method to reduce the complexity of the matrix and norm value computations for the ML detector. It is shown that eliminating the multiplier in the ML detector implementation results in significant savings in area usage and power consumption. For example, with 9-bits word length for y and H, area and power savings are 42% and 68% respectively compared with conventional implementations based around the use of multipliers.

8. References

[1] G. Foschini and M. Gans, “On limits of wireless communications in fading environment when using multiple antennas”, Wireless Personal Communications, vol. 6, pp. 311-335, 1998.

[2] S. M. Alamouti, “A simple transmit diversity technique for wireless communications”, IEEE Journal on Selected Areas in Communications, vol. 16, no. 8, pp. 1451-1458, Oct. 1998.

[3] B. M. Hochwald and S. ten Brink, “Achieving near-capacity on a multiple-antenna channel”, IEEE Trans. on Communications, vol. 51, no. 3, pp. 389-399, March 2003.

[4] P. W Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, “V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel”, International Symposium on Signals, Systems, and Electronics (ISSSE 98), pp.295-300, 29 Sept. – 2 Oct. 1998.

[5] L. M. Davis, “Scaled and decoupled Cholesky and QR decompositions with application to spherical MIMO detection”, IEEE Wireless Communications and Networking (WCNC 2003), vol. 1, pp. 326-331, 16-20 March 2003.

[6] D. Garrett, L. Davis, S. ten Brink, B. Hochwald, and G. Knagge, “Silicon complexity for maximum likelihood MIMO detection using spherical decoding”, IEEE Journal of Solid-State Circuits, vol. 39, no. 9, pp.1544-1552, Sept. 2004.

[7] A. Burg, N. Felber, and W. Fichtner, “A 50 Mbps 4x4 maximum likelihood decoder for multiple-input multiple-output systems with QPSK modulation”, 10th IEEE Int. Conf. on Electronics, Circuit and Systems (ICECS 2003), vol. 1, pp. 332-335, 14-17 Dec. 2003.

[8] D. Garrett, G. K. Woodward, L. Davis, and C. Nicol, “A 28.8 Mb/s 4x4 MIMO 3G CDMA receiver for frequency selective channels”, IEEE Journal of Solid-State Circuits, vol. 40, no. 1, pp. 320-330, Jan. 2005.