A LOW-COMPLEXITY HIGH-THROUGHPUT MIMO DECODER ON FPGA USING VB ALGORITHM

Himalaya.B[1] and Mr.P.Prakash[2]

[1] *PG Student, ME Computer and Communication, Sri Sairam Engineering College, Chennai-44.*
[2]*Associate Professor ,Department of IT, Sri Sairam Engineering College, Chennai-44.*

[1]himalayaece@gmail.com

[2] prakesh.it@sairam.edu.in

ABSTRACT

The traditional single-input single-output (SISO) channel cannot meet the continuously increasing demands for channel capacity and quality of service (QoS) in wireless communications because of the scarce spectrum bandwidth provided for modern wireless systems. By introducing multiple antennas into both transmitter and receiver, MIMO technique has emerged as a key technology for the next generation wireless systems. The recent years, multiple-input multiple-output (MIMO) technology has attracted great attentions in the area of wireless communications. The hardware implementation of MIMO decoders becomes a challenging task as the complexity of the MIMO system increases to improve the system efficiency. The hardware/software co-design architecture and implementations of Viterbo and Boutros (VB) algorithm is proposed for efficient MIMO decoder. Three levels of parallelisms are analyzed for an efficient implementation with the preprocessing and the decoding part on customized hardware. The decoders for a 4 by 4 MIMO system with 16-QAM modulation scheme are prototyped on a Xilinx FPGA device. The hardware implementations of the VB decoders show that support up to 37 Mbps data rate. The performances in terms of resource utilizations and BER are also compared between these two decoders.

Keywords—MIMO, VB, 16-QAM, BER, MIMO decoders, Xilinx FPGA.

1. INTRODUCTION

Wireless communication systems are dense compositions of signal processing and VLSI technologies. With the ever increasing demand of higher data rate and better quality of service, VLSI design and implementation method for wireless communications becomes more challenging, which urges researchers to provide new architectures and efficient implementations to meet high performance requirements.

The applications of MIMO technology have emerged at the forefront of the developing standards for next generation mobile communications and wireless networks. Combined with the orthogonal frequency-division multiplexing (OFDM) technique, MIMO is proposed to be incorporated into the fourth generation (4G) mobile communications system architectures to enhance voice and data transmissions.

2. RELATED WORK

. The SoC approach is introduced to simplify the design of the sphere decoder and to improve the decoding efficiency. The FPGA based hardware/software co-design techniques partition the complicated pre-processing tasks such as matrix factorizations and inversions to an embedded processor and the real-time decoding functions to customized hardware modules. The VB decoding algorithm is implemented on FPGA platform and is evaluated for decoding rates, BER performance, power consumptions, and area utilizations.

Three levels of parallelism are explored to accelerate the processing: the concurrent execution of the pre-processing on an embedded processor and the decoding on hardware cores; the parallel decoding of real/imaginary parts if complex constellation applies; and the concurrent execution of multiple steps during the closest lattice point search.

The MIMO decoding problem is essentially to search for the closest lattice point to the received point. The optimal detection strategy for a MIMO decoder is to perform a maximum likelihood (ML) search over all possible points inside a lattice structure and find the best one with smallest Euclidian distance to the received signal.

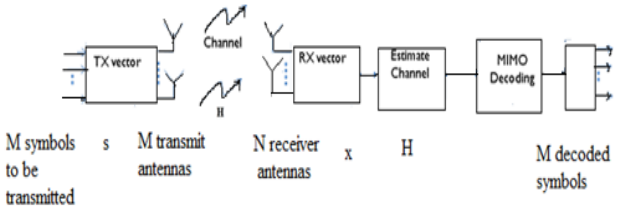

Fig 1 System Model

3. PROPOSED WORK

The Application specific integrated circuit implementation method is proposed to implement the MIMO decoder to get better efficiency. ASIC implementation is generally refined for a fixed number of antennas and a certain signal constellation similar to DSP processors. The loss of flexibility becomes a major limitation for ASIC implementations of sphere decoders.

Field programmable gate array (FPGA) devises are also widely used in signal processing field due to their flexible reconfiguration and support of parallelism. Combined with their huge processing capabilities, high data rates are

ensured for many computational intense algorithms implemented in FPGAs. The FPGA based MIMO Decoder design is proposed based on VB algorithm

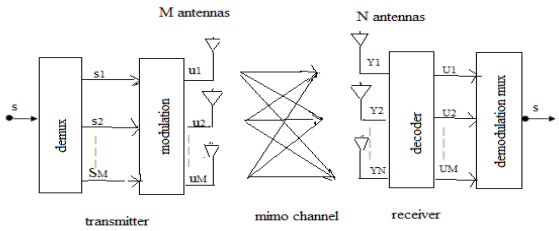

Figure 2 System Atrchitecture

VB algorithm steps

Step 1: Preprocessing and initialization:
Transform H into an upper triangular matrix by Cholesky [20] factorization algorithm. Calculate
$\bar{\rho}=\bar{y}H^{-1}$. Initialize the sphere radius $\sqrt{C}$ by an adaptive method [21]. Set dimension index $i=M$
and $d_{best}=\sqrt{C}$. Find the upper bound L_M and index u_M.
Step 2: Finite State Machine (FSM)
Upgrade $u_i= u_i +1$.
If $u_i < L_i$ and $i>1$ go to State A;
If $u_i < L_i$ and $i=1$ go to State B;
If $u_i > L_i$ go to State C.
Step 3: State A:
Expand the layer into (i-1) dimensional sublayer and find the S_i and T_i used to upgrade u_i and L_i.
Goes to State D.
Step 4: State B
Upgrade d_{new}.
If $d_{new} < d_{best}$, record the currently best distance and the best point. Set $i=M$. Go to State D.
If $d_{new} > d_{best}$, go to step2.
Step 5: State C
Stop if $i=M$, otherwise move the procedure one
step up $i=i+1$, and go to step 2;
Step 6: State D
Upgrade u_i and L_i, and go to step 2.

4. IMPLEMENTATION

5.1.1 16-QAM MODULATION

The 4x4 MIMO System with 16-QAM Modulation scheme is designed to spatial multiplex the user transmission data. The data bits are split into four 16-QAM streams which are transmitted independently.The 16-QAM Modulator module has 4 bits input and the output symbol is represented as in phased component and quadrature component. The constellation diagram for 16 QAM modulator acts as input to the MIMO encoder. The simulation result for 16-QAM modulator is discussed in detail in chapter 8.

The AWGN channel noise modelled with Gaussian random variables with zero mean and different variance to generate different dB level of noise. The generated channel noise is added to 16-QAM modulated signal to simulate the channel scenario on designed system model.

5.1.2 16-QAM DEMODULATION

The design of 16-QAM Demodulator for 4x4 MIMO System is not as easy as designing QAM modulator. The output symbols coming

from MIMO decoder are passed through wireless channel and the symbols get corrupted. The task of demodulator is to make a decision based on the minimum Euclidian distance.

5.1.3 VB PREPROCESSING

The basis reduction is performed to reduce the complexity of decoding procedure before the closest point search begins. This is called pre-processing, which involves Cholesky decomposition and matrix inversion. These complex matrix manipulations are difficult and too costly for hardware implementation, and they are not executed frequently in the sphere decoding algorithm. So only the iterative decoding procedures are considered to be implemented on FPGA hardware.The FPGA-based HW/SW co-design architecture for MIMO decoders with complex constellation structure is proposed, which partitions the complicated channel matrix pre-processing including matrix inversion and factorization into soft processor and the iterative decoding procedures into customized hardware modules. This architecture is able to significantly improve the decoding rate, and meanwhile keeps it easy to be implemented in hardware.

5.1.4 VB FSM

Finite State Machine (FSM) is a circuit with internal states. Unlike the regular sequential circuits, state transition of FSM is more complicated and the sequence exhibits no simple, regular pattern, as in counter or shift register. The next-state logic has to be constructed from scratch and is sometimes known as “random” logic. An FSM is specified by five entities: symbolic states, input signals, output signals, next-state function and output function. A state specifies a unique internal condition of system. As time progress, the FSM transits from one state to another state. The new state is determined by the next-state function, which is a function of the concurrent state and input signal.

To improve the decoding speed, a state level parallelism is developed based on the data flow dependency analysis between all the states. A finite state machine (FSM) is designed to control the transitions among these states. This state level parallelism is implemented in the FPGA hardware modules.

5.1.5 VB STATE PROCESSING

The VB-based decoder searches from the bottom layer to the top layer and scans each lattice index from the lower bound to the upper bound. When the algorithm reaches the top layer without violating the bound constraint, a valid lattice point is found. Then the new distance between the valid lattice point and the received point is calculated and compared with the currently best distance, which is initialized to be equal to radius.

If new distance is smaller than distance best, a closer lattice point is found and stored as the currently best lattice point. The searching

radius is upgraded to distance new. This process iterates until all the lattice points within the sphere are examined.

5. CONCLUSION AND FUTURE WORK

Hardware/software co-design architecture for typical MIMO lattice decoding algorithms has been designed and implemented in this project. The closest lattice point searching procedure is partitioned into the FPGA-based hardware modules. Three levels of parallel structures are designed in this co-design architecture to improve the decoding rate. The overheads involved in these parallel structures are also analyzed.The experimental result will show that the VB based decoders can reach up to 81.5 Mbps and 37.3 Mbps decoding rate respectively at 20 dB Eb/N0 for a 4x4 MIMO system with 16-QAM modulation, which are among the fastest MIMO decoders to the author's knowledge. The BER performance of the experimental prototype matches with the software simulation results.

The modified VB algorithm is proposed by analyzing the calculations involved in these four states of the searching procedure, it is clear that State D carries much more computational load than other states. This is because the hardware inefficient square root functions are used to upgrade Ui and Li. The requirements of the square root functions make the computation load unbalanced among the four states.

6. REFERENCES

[1]] G. J. Foschini and M. Gans, "On the limits of wireless communication in a fading environment when using multiple antennas," Wireless Per- sonal Commun., vol. 6, pp. 311–335, 1998.

[2] B. M. Hochwald and S. Ten Brink, "Achieving near-capacity on a multiple-antenna channel," IEEE Trans. Commun., vol. 51, no. 3, pp. 389–399, Mar. 2003.

[3] D. Gesbert, M. Shafi, S. Da-shan, P. J. Smith, and A. Naguib, "From theory to practice: An overview of mimo space-time coded wireless systems," IEEE J. Sel. Areas Commun., vol. 21, no. 3, pp. 281–302, Mar. 2003

[4] "Multiple-input multiple-output in UTRA (Release 7)," 3GPP TR25.876 (V7.0.0), 2006.

[5] E.Viterbo and J.Boutros,"A universal lattice code decoder for fading channels," IEEE Trans. Inf. Theory, vol. 45, no. 5, pp. 1639–1642, May 1999.

[6]C.P.SchnorrandM.Euchner,"Latticebasisreduction:Improvedprac- tical algorithms and solving subset sum problems," Math. Program., vol. 66, no. 2, pp. 181–191, 1994.

[7] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," IEEE Trans. Inf. Theory, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[8] A. Adjoudani, E. C. Beck, A. P. Burg, G. M. Djuknic, T. G. Gvoth, D. Haessig, S. Manji, M. A. Milbrodt, M. Rupp, D. Samardzija, A. B. Siegel, I. Sizer, T. , C. Tran, S. Walker, S. A. Wilkus, and P. W. Wolniansky, "Prototype experience for MIMO blast over third-generation wireless system," IEEE J. Sel. Areas Commun., vol. 21, no. 3, pp. 440–451, Mar. 2003. [15] M. Guillaud, A. Burg, M

[9] Z. Guo and P. Nilsson, "A VLSI architecture of the soft-output sphere decoder for MIMO systems," in Proc. IEEE Int. Midwest Symp. Cir- cuits Syst., 2005, pp. 1195–1198.

[10] O. Damen, A. Chkeif, and J. C. Belfiore, "Lattice code decoder for space-time codes," IEEE Commun. Lett., vol. 4, no. 5, pp. 161–163, May 2000.