Reduced Complexity MIMO MMSE-DFE

Wen-Chih Kan and Gerald E. Sobelman
Department of Electrical and Computer Engineering
University of Minnesota
Minneapolis, MN 55455 USA
kanx0008@umn.edu, sobelman@umn.edu

Abstract— We present a reduced complexity MIMO MMSE-DFE coefficient computation algorithm which is based on an improvement to the fast block algorithm developed in Ref. [1]. We have achieved a nearly 50% reduction in complexity by eliminating the computation of unnecessary terms. The effects due to finite precision arithmetic on SNR are also considered and a minimum fixed-point word length of 32 bits results in no noticeable performance loss. Hardware implementation issues depend on several factors and these are also discussed.

I. INTRODUCTION

Current wireless systems would like to achieve a 1 Gb/s data rate. However, single-input, single-output (SISO) systems can typically only provide a spectral efficiency of 2~4 bps/Hz which requires a 250 MHz bandwidth in order to deliver a 1Gb/s transmission rate. Multiple-input, multiple-output (MIMO) techniques are one way to increase the spectral efficiency and a MIMO wireless system can provide a 1 Gb/s transmission using only a 20 MHz band, which is more practically available [2]. Refs. [3]-[4] have proposed designs for the implementation of a MIMO wireless system in the frequency-flat channel environment.

If the channel coherence bandwidth is less than the transmitted signal bandwidth, then the channel is called frequency selective or dispersive. Otherwise, it is called a frequency flat channel [6]. The dispersive channel distorts transmitted signals in the form of inter-symbol interference (ISI), which if not correctly compensated, can lead to a high bit error rate. Equalization has been the solution for the ISI problem. There exist several different types of equalization, namely, maximum likelihood sequence estimation (MLSE), linear equalization (LE), and decision feedback equalization (DFE). Among these, DFE has good performance with lower complexity than MLSE and hence it has received lots of research attention over the last twenty years. If the channel coherence time is much larger than the symbol interval of the transmitted signal, the channel is called slow fading. Otherwise, it is called fast fading [6]. In Ref. [5], two different transmission scenarios are presented which clearly explain the frequency selective channel with slow fading and with fast fading. Fig. 1 is a general block diagram of the receiver dealing with a frequency selective channel with fast fading (i.e., severe time selectivity as in Ref. [5]). If a channel has frequency selectivity and severe time selectivity, a computation is needed to constantly update the filter coefficients in order for the DFE to work effectively. A generic block diagram of a DFE structure is shown in Fig. 2.

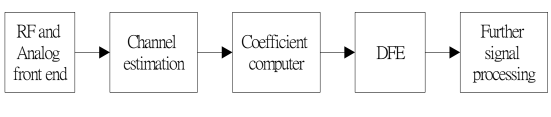

Figure 1. Receiver block diagram for frequency selective and time selective channels.

In Ref. [1], two fast MIMO MMSE-DFE coefficient computation algorithms (called SFBC-DFE and BFBC-DFE) are reported which reduce the computational complexity of the intense spectral factorizations of matrix rational spectra. In this paper, a new reduced complexity MIMO MMSE-DFE coefficient computation algorithm (or RCFBC-DFE for short) is presented. The architecture is based on the fast computation algorithm developed in Ref. [1]. Finite word length effects are important and hence an analysis of the degradation in SNR of the MIMO MMSE-DFE when the filter coefficients are computed in a fixed-point manner is also considered.

The remainder of this paper is organized as follows: In Section II, the system model of the MIMO MMSE-DFE and its SFBC-DFE and BFBC-DFE algorithms are discussed. The RCFBC-DFE is presented in Section III. Numerical examples, comparisons in complexity between RCFBC-DFE and block FBC-DFE and the finite precision effects of the fixed-point computer simulation are given in Section IV. Hardware implementation issues are discussed in Section V. Finally, the conclusions are given in Section VI.

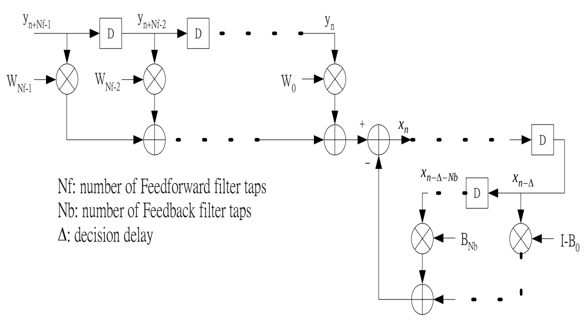

Figure 2. A generic DFE structure.

The notation used in this paper is summarized as follows:

1. $E[\cdot]$ is the expected value operator.
2. $(\cdot)^*$ represents the complex conjugate transposition.
3. $(\cdot)_{m\times n}$ denotes an m-by-n matrix.
4. $(\cdot)_{[i:j,:]}$ denotes the sub-matrix obtained by extracting rows i through j.
5. e_i represents the ith unit vector.

II. SYSTEM MODEL AND FAST ALGORITHMS

A. System Model

The system model used in this paper is based on a general linear, dispersive and noisy digital communication system depicted in Fig. 3 where the channel has n_i inputs and n_o outputs and $h^{(i,j)}$ denotes the channel impulse response between ith input and the jth output and $n^{(j)}$ represents AWGN of the jth output. If we further assume that the channel length is denoted by $v^{(i,j)}$ for channel $h^{(i,j)}$, the input and output relation can be written as the form,

$$y_k^{(i)} = \sum_{i=1}^{n_i}\sum_{m=0}^{v^{(i,j)}} h_m^{(i,j)} x_{k-m}^{(i)} + n_k^{(j)} \tag{1}$$

where k is the time instant. Furthermore, if an oversampling is assumed, say l samples per symbol period, the output, impulse response and noise would be $l \times 1$ column vectors.

By grouping all the output samples from the n_o channel into an $ln_o \times 1$ column vector, equation (1) can be written in a more compact form,

$$y_k = \sum_{m=0}^{v} H_m x_{k-m} + n_k \tag{2}$$

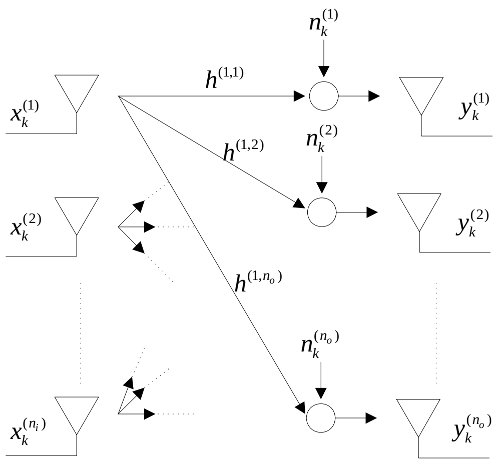

Figure 3. MIMO channel model

where H_m is an $ln_o \times n_i$ MIMO channel coefficient at time m, x_{k-m} is an $n_i \times 1$ column vector at time k-m, and v is the maximum length of all of the $n_o n_i$ channel impulse responses.

Over a block of N_f symbol periods, equation (2) can be expressed in a matrix form as follows,

$$\begin{bmatrix} y_{k+N_f-1} \\ y_{k+N_f-2} \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} H_0 & H_1 & \cdots & H_v & 0 & \cdots & 0 \\ 0 & H_0 & H_1 & \cdots & H_v & 0 & \cdots \\ \vdots & & \ddots & & \ddots & & \vdots \\ 0 & \cdots & 0 & H_0 & H_1 & \cdots & H_v \end{bmatrix} \bullet \begin{bmatrix} x_{k+N_f-1} \\ x_{k+N_f-2} \\ \vdots \\ x_{k-v} \end{bmatrix} + \begin{bmatrix} n_{k+N_f-1} \\ n_{k+N_f-2} \\ \vdots \\ n_k \end{bmatrix} \tag{3}$$

or more compactly,

$$y_{k+N_f-1:k} = Hx_{k+N_f-1:k-v} + n_{k+N_f-1:k}. \tag{4}$$

B. Fast Algorithms

It is shown in Ref. [1] that the optimum feedback and feedforward filters are equations (17) and (22) of Ref. [1]. In order to compute those filter coefficients, the fast Cholesky factorization is utilized on the matrix $R \equiv R_{xx}^{-1} + H^* R_{nn}^{-1} H = \overline{L}\overline{L}^*$,

where R_{xx} is the auto-correlation matrix of the input signals, R_{nn} is the auto-correlation matrix of the noise sources and $\overline{L}$ is the Cholesky triangle [8]. The auto-correlation functions are defined as follows.

$$R_{xx} \equiv E[x_{k+N_f-1:k-v} x_{k+N_f-1:k-v}^*]. \tag{5}$$

$$R_{nn} \equiv E[n_{k+N_f-1:k} n_{k+N_f-1:k}^*]. \tag{6}$$

Two fast factorizations, namely, *standard* and *block* factorization, were developed in Ref. [1] to perform the fast Cholesky factorization with a computational efficiency of $O[n_i^2(N_f+v)^2]$ compared to $O[n_i^3(N_f+v)^3]$ for the classical Guaussian elimination method.

III. REDUCED COMPLEXITY MIMO MMSE-DFE AND EFFECT OF FIXED-POINT IMPLEMENTATION

A. Reduced Complexity Algorithm

We now make the following observation to further improve the computational efficiency. The optimum feedback filter derived in Ref. [1] (i.e., its equation (17)) can be further simplified as follows:

$$B_{opt}^* = \left\lfloor B_0^* \quad B_0^*((\overline{L}_2\overline{L}_1^{-1})^*)_{[n_i\Delta+1:n_i(\Delta+1),:]} \right\rfloor \tag{7}$$

where B_0 is computed by equation (9), Δ is the optimum delay, and both $\bar{L}_1$ and $\bar{L}_2$ are partitions of sizes $n_i(\Delta+1) \times n_i(\Delta+1)$ and $n_i(N_f+v-\Delta-1) \times n_i(\Delta+1)$ in

$$\bar{L} \equiv \begin{bmatrix} \bar{L}_1 & 0 \\ \bar{L}_2 & \bar{L}_3 \end{bmatrix}. \tag{8}$$

In other words, we find that the sub-block $\bar{L}_3$ is not used so that it is not necessary to compute it.

B_0 is computed via the following equation,

$$B_0 e_i = \frac{R_3^{-1} e_i}{e_i^* R_3^{-1} e_i}, 1 \le i \le n, \tag{9}$$

where $R_3^{-1} = \overline{L_{13}}\,\overline{L_{13}}^*$ if we further partition $\bar{L}_1$ as,

$$\bar{L}_1 \equiv \begin{bmatrix} \bar{L}_{11} & 0 \\ \bar{L}_{12} & \bar{L}_{13} \end{bmatrix} \text{ with } \overline{L_{13}} \text{ of size } n_i \times n_i. \tag{10}$$

Since the feedforward filter (i.e., equation (22) of Ref. [1]) and the feedback filter are related through the following equation,

$$W_{opt}^* = \left[0_{n_i \times n_i \Delta} \quad B_{opt}^* \right] R^{-1} H^* R_{nn}^{-1}, \tag{11}$$

the feedforward filter can be easily simplified as the form,

$$W_{opt}^* = B_0^* ((\bar{L}_1^{-1})^*)_{[n_i\Delta+1:n_i(\Delta+1),:]} \bar{L}_1^{-1} (H^*)_{[1:n_i(\Delta+1),:]} R_{nn}^{-1} \tag{12}$$

where $\bar{L}_1^{-1}$ are partitions in

$$\bar{L}^{-1} \equiv \begin{bmatrix} \bar{L}_1^{-1} & 0 \\ \bar{L}_2^{(-1)} & \bar{L}_3^{-1} \end{bmatrix} \text{ (note that } \bar{L}_2^{(-1)} = -\bar{L}_3^{-1} \bar{L}_2 \bar{L}_1^{-1} \text{).} \tag{13}$$

From equations (7) and (12), we only need to compute the $\bar{L}_1^{-1}$ and $\bar{L}_2$ rather than the entire $\bar{L}$ matrix. Also, $\bar{L}_1^{-1}$ can be computed by inverting $\bar{L}_1$. Therefore, the actual computational complexity is only $O[n_i^2(\Delta+1)^2]$ compared to the previously reported $O[n_i^2(N_f+v)^2]$.

B. *Performance Analysis*

Our performance analysis is based on the Arithmetic SNR, which is defined by:

$$ASNR \equiv \frac{\frac{1}{n_i(N_f+v)} trace(R_{xx})}{\frac{1}{n_i} trace(R_{ee,\min})} \tag{14}$$

where R_{xx} has (N_f+v) blocks of another diagonal matrix D_x of size $n_i \times n_i$ on its diagonal.

This performance measure can be simplified to:

$$ASNR \equiv \frac{trace(D_x)}{trace(R_{ee,\min})}. \tag{15}$$

IV. Numerical Examples

The numerical examples are given below to show that we save nearly 50% of the computational load of the block FBC-DFE. Finite precision effects on the fixed point computation is also given here.

A. *Simulation Parameters*

Three different scenarios were considered in the simulations of finite precision effects.

- Scenario 1:
    - Channel: 2 transmit and 2 receive antennas
    - Channel length: 2
    - Feedforward taps: 3
    - Feedback taps: 2
    - Decision delay: 2
    - Input SNR: 10, 15, and 20dB
- Scenario 2:
    - Channel: 2 transmit and 2 receive antennas
    - Channel length: 4
    - Feedforward taps: 3
    - Feedback taps: 4
    - Decision delay: 2
    - Input SNR: 10, 15, and 20dB
- Scenario 3:
    - Channel: 3 transmit and 3 receive antennas
    - Channel length: 4
    - Feedforward taps: 3
    - Feedback taps: 4
    - Decision delay: 2
    - Input SNR: 10, 15, and 20dB

B. *Finite Precision Effects*

Based on the above three simulation scenarios, we conclude that 32-bit fixed-point arithmetic should be used throughout the computation for a 2 x 2 configuration in order to have no noticeable performance degradation compared to double-precision floating point. If the SNR at the receiver input is low, e.g. 10dB, the word length can be reduced to 24 bits. For 3 x 3 configurations, the minimum fixed-point word length is also 32 bits without any noticeable loss in performance. Figs. 4 ~ 6 are the finite precision effect for Scenario 1 through 3 respectively.

C. *Complexity Reductions*

The new RCFBC-DFE algorithm reduces the computational complexity by nearly 50% compared to the

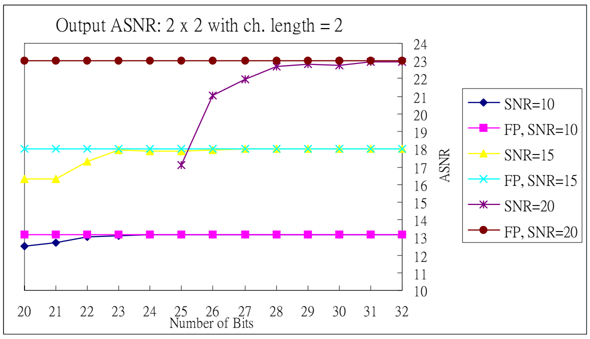

Figure 4. Finite Precision Effects for Scenario 1.

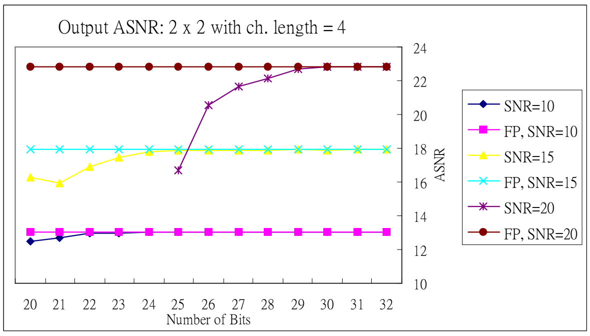

Figure 5. Finite Precision Effects for Scenario 2.

BFBC-DFE. For example, for Scenario 1, we only use 2420 multiplications compared to 4456 multiplications for the BFBC-DFE. Also for Scenario 2, about 51% of the multiplications are saved compared to the BFBC-DFE (3168 vs. 6438). The new algorithm also saves more than 50% of the number of additions required in the original algorithm. Figs. 7 and 8 show the number of multiplications and additions used by RCFBC and BFBC, respectively. In both figures, channels of length 2 and length 4 were considered.

V. HARDWARE IMPLEMENTATION ISSUES

The MIMO MMSE-DFE implementation depends on several factors, e.g., how fast the channel changes, system configurations, and the channel length. Given a 3G user with a carrier frequency of 2140 MHz traveling at 55 miles per hour, the user sees a channel coherence time of about 5.7 ms and we can assume the channel stays constant for at most 1 ms. This, in turn, requires the DFE to update its coefficients at a rate of 1000 times per second. If we further assume the channel and system configuration of Scenario 2, then the RCFBC-DFE will need 3168 32-bit multiplications and 2026 32-bit additions to update its coefficients every 1 ms. A TI TMS320VC55x DSP processor [8] can perform this task easily. However, if the system configuration is 6 x 6 and the channel length is 4, the RCFBC will require nearly 208,353 32-bit multiplications and 162,063 32-bit additions in 1 ms to update its coefficients. A TMS320VC55x can not handle that computation load. A dedicated 32-bit multiplier running at a speed of at least 209 MHz could be used for such an application [9].

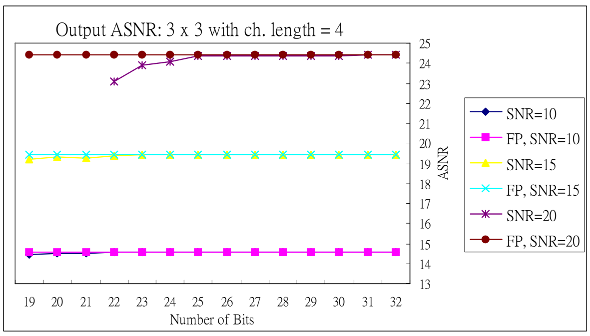

Figure 6. Finite Precision Effect for Scenario 3.

VI. CONCLUSIONS

As wireless communications systems reach toward 1 Gbps transmission, MIMO techniques and wideband transmission may be utilized to achieve this goal. A highly mobile user under this situation faces a fast changing, frequency selective channel which will introduce inter-symbol interference to the data transmission. Ref. [1] presented a MIMO MMSE-DFE coefficient computational algorithm which reduced the computation complexity to $O[n_i^2(N_f+v)^2]$ from $O[n_i^3(N_f+v)^3]$ for the classical Guaussian elimination method. We have further simplified that algorithm by eliminating the computation of unnecessary terms. Our algorithm lowers the computational complexity to $O[n_i^2(\Delta+1)^2]$. Numerical simulations show that a computational savings of nearly 50% can be achieved. For good performance comparable to that of double-precision floating-point arithmetic, the fixed-point word length should be 32 bits. Current high-speed, low-power multiplier options are available which can meet the required computational load. The actual hardware implementation

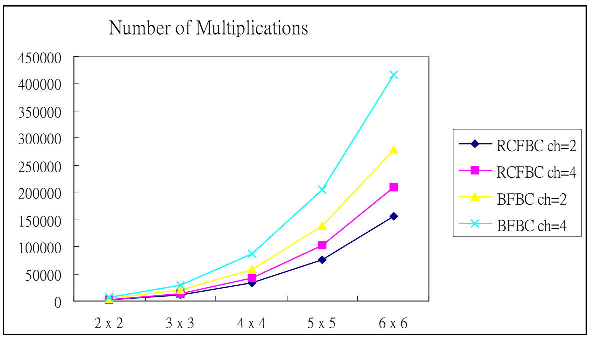

Figure 7. Number of multiplications used by RCFBC and BFBC when the channel length is 2 and 4.

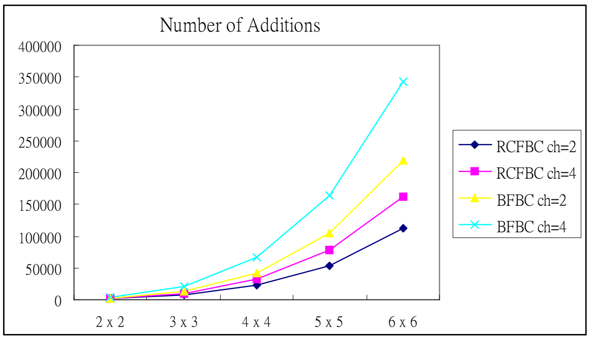

Figure 8. Number of additions used by RCFBC and BFBC when the channel length is 2 and 4.

depends on several factors, e.g. how fast the channel changes, the channel length and the number of transmit and receive antennas. In some cases, a dedicated hardware implementation with a high speed 32-bit multiplier and adder will be required. Current high-speed multipliers which can run at up to 1 GHz make this coefficient computation technique feasible.

REFERENCES

[1] N. Al-Dhahir and A. H. Sayed, "The Finite-Length Multi-Input Multi-Output MMSE-DFE", IEEE Trans. On Signal Processing, vol. 48, no. 10, pp. 2921-2936, Oct. 2000.

[2] A. J. Paulraj, D. A. Gore, and R. U. Nabar, "An Overview of MIMO Communications-A Key to Gigabit Wireless", Proc. of the IEEE, vol. 92, no. 2, pp. 198-218, Feb. 2004.

[3] Z. Guo and P. Nilsson, "A VLSI Architecture of the Schnorr-Euchner Decoder for MIMO Systems", in IEEE 6th CAS Symp. on Emerging Tech., Shanghai, China, vol. 1, pp. 65-68, May 2004.

[4] Z. Guo and P. Nilsson, "A VLSI Implementation of MIMO Detection for Future Wireless Communications", in the 14th IEEE 2003 International Symp. on Personal, Indoor and Mobile Radio Comm. Proc., vol. 3, pp. 2852-2856, Sep. 2003.

[5] S. N. Diggavi, N. Al-Dhahir, A. Stamoulis, and A. R. Calderbank, "Great Expectations: The Value of Spatial Diversity in Wireless Networks", Proc. of the IEEE, vol. 92, no. 2, pp. 219-270, Feb. 2004.

[6] J. W. Mark and W. Zhuang, Wireless Communications and Networking, Upper Saddle River, NJ: Pearson Education Inc., 2003.

[7] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd ed., Baltimore, MD: John HopkinsUniversity Press, 1996.

[8] Available: http://dspvillage.ti.com/

[9] H-C Chow; I-C Wey, "A 3.3 V 1 GHz high speed pipelined Booth multiplier", IEEE International Symposium on Circuits and Systems, ISCAS 2002, vol. 1, pp. 457-460, May 2002