

Yuanbin Guo, Dennis McCain Nokia Research Center Irving, TX, 75039 Email: {Yuanbin.Guo, Dennis.McCain}@nokia.com Joseph R. Cavallaro Dept. of Electrical and Computer Engineering Rice University Houston, TX, 77005 Email: cavallar@rice.edu

Abstract—In this paper, we present a novel FFT-accelerated iterative Linear MMSE chip equalizer in the MIMO CDMA downlink receiver. The reversed form time-domain matrix multiplication in the Conjugate Gradient iteration is accelerated by an equivalent frequency-domain circular convolution with FFT-based "overlap-save" architecture. The iteration rapidly refines a crude initial approximation to the actual final equalizer taps. This avoids the Direct-Matrix-Inverse with $O((NL)^3)$ complexity, and reduces the standard CG complexity from $O((NL)^2)$ to $O(NL \log_2(NL))$. Simulation demonstrates strong numerical stability and promising performance/complexity tradeoff, especially for very long channels.

I. INTRODUCTION

UMTS and CDMA2000 extensions optimized for data services lead to standards of High-Speed-Downlink-Packet-Access (HSDPA) and its equivalent 1X EV-DO [1] to support multimedia wireless services. On the other hand, MIMO (Multiple-Input-Multiple-Output) technology has emerged as a significant breakthrough to significantly increase the spectral efficiency. Recently, MIMO extensions for 3G and beyond wireless systems have received more and more attention from the research community [1]. However, the original inventions known as D-BLAST and a more realistic V-BLAST [2] were proposed for narrow band and flat-fading channels. In a multipath channel, the conventional Rake receiver could not provide satisfactory performance because of the short spreading gain. A LMMSE (Linear-Minimum-Mean-Square-Error)-based chip equalizer is proposed to restore the code orthogonality and suppress both the Multiple-Access-Interference and Inter-Symbol-Interference [3]-[5] in a CDMA downlink. However, the LMMSE equalizers involve the inverse of a large correlation matrix with prohibitive $O((NL)^3)$ complexity for real-time hardware implementation [3], where N is the number of Rx antennas and L is the channel length.

Although the adaptive LMS algorithm avoids the Direct-Matrix-Inverse (DMI), it suffers from stability problems because of its dependency on good step size. A Conjugate Gradient algorithm proposed in [4] could be considered as a fast algorithm with $O((NL)^2)$ complexity. However, the eigenvalue spread of the covariance matrix and the required filter length increase with the MIMO signal dimension. The large eigenvalue spread slows down the convergence rate, especially for the MIMO system. In order to capture the effects of mobile speed in the fading channel, the equalizer filters must be updated several times in one WCDMA slot. It is shown that this complexity still limits the real-time implementation for compact handset hardware [5].

In this paper, we show that the CG iteration is essentially reduced to performing matrix-vector multiplication for the correlation matrix once per iterative step. Because the correlation matrix assumes a block-Toeplitz structure, we apply a divide-and-conquer methodology to accelerate the CG iteration. We first transform the direct form block-Toeplitz structure of the correlation matrix in the CG iteration to a reversed form block-Toeplitz structure. The new timedomain matrix multiplication is accelerated by equivalent FFT-based "overlap-save" computing architectures in the frequency domain. Superfast [6] acceleration to $O(NL \log_2(NL))$ is then achieved. The proposed iterative acceleration is compared with another FFT-based solution using circulant approximation [5], where circular corners need to be added to approximate the block Toeplitz structure with a circulant structure. In high Signal-to-Noise-Ratio (SNR) range, this increases the condition number and degrades the system stability and performance. Unlike the FFT-based solution using the circulant approximation, no approximation of the matrix structure is necessary in the proposed algorithm. The acceleration does not degrade the performance and demonstrates strong numerical stability for well and mild conditioned systems and facilitates fixed-point parallel implementation.

II. SYSTEM MODEL

The system model of the MIMO Multi-Code CDMA downlink using M Tx antennas and N Rx antennas is described as follows. First, the high data rate symbols are demultiplexed into KM lower rate substreams, where K is the number of spreading codes used for data transmission. The substreams are broken into M groups, where each substream in the group is spread with a spreading code of spreading gain G. The groups of substreams are then combined and scrambled with scrambling codes and transmitted through the m^{th} Tx antenna. The chip level signal at the m^{th} transmit antenna is given by $d_m(i) = \sum_{k=1}^{K} s_m^k(g)c_m^k(i) + s_m^P(g)c_m^P(i)$ where g, iand k are the symbol, chip and code indices respectively. $s_m^k(g)$ is the g^{th} symbol of the k^{th} code at the m^{th} substream. In the following, we focus on the g^{th} symbol index and omit the index for simplicity. $c_m^k(i) = c_k(i)c_m^{(m)}(i)$ is the composite spreading code sequence for the k^{th} code at the m^{th} substream where $c_k(i)$ is the user specific spreading code and $c_m^{(m)}(i)$ is the antenna specific scrambling long code. $s_m^P(g)$ denotes the pilot symbols at the m^{th} antenna. $c_m^P(i) =$ $c^P(i)c_m^{(m)}(i)$ is the composite spreading code for pilot symbols at the m^{th} antenna. The received chip level signal at the n^{th} Rx antenna is given by

$$r_n(i) = \sum_{m=1}^M \sum_{l=0}^L h_{m,n}(l) d_m(i - \tau_l) + z(i)$$
(1)

where $h_{m,n}(l)$ constructs the channel matrix between the m^{th} Tx and the n^{th} Rx antennas.

By collecting the $L_F = 2F + 1$ consecutive chips with center at the i^{th} chip from each Rx antenna in $\mathbf{r}_n(i) = [r_n(i+F), \cdots, r_n(i), \cdots, r_n(i-F)]^T$ and packing the signal vectors from all antennas, we form a vector as $\mathbf{r}_A = [\mathbf{r}_1(i)^T, \cdots, \mathbf{r}_n(i)^T, \cdots, \mathbf{r}_N(i)^T]^T$. Here F is the observation

window length at the receiver to support the different channel models. The received signal is then given by,

$$\mathbf{r}_A(i) = \sum_{m=1}^M \mathbf{H}_m \mathbf{d}_m(i) + \mathbf{v}(i)$$
(2)

where $\mathbf{v}(i)$ is the additive Gaussian noise and $\mathbf{d}_m(i) = [d_m(i + F), \cdots, d_m(i), \cdots, d_m(i - F - L)]^T$ is the transmitted chip vector for the m^{th} Tx antenna. $\mathbf{H}_m = [\mathbf{H}_{m,1}^T, \mathbf{H}_{m,2}^T, \cdots, \mathbf{H}_{m,N}^T]^T$ is the channel matrix with block Toeplitz structure.

III. SUPERFAST ITERATIVE CHIP EQUALIZER

A. Conjugate Gradient Chip Equalizer

A Linear MMSE based chip equalizer estimates the transmitted chip samples filtered by a linear FIR filter as $\hat{\mathbf{d}}_m(i) = \hat{\mathbf{w}}_m^H \mathbf{r}_A(i)$. The LMMSE chip equalizer is given by $\hat{\mathbf{w}}_m^{opt} = \arg\min_{\mathbf{w}_m} E[||\mathbf{d}_m - \hat{\mathbf{w}}_m^H(i)||^2] = \sigma_d^2(i)\mathbf{R}_{rr}(i)^{-1}\hat{\mathbf{h}}_m$ where $\mathbf{R}_{rr}(i)$ is the covariance matrix and the channel coefficients are estimated as $\hat{\mathbf{h}}_m(i) = E[\mathbf{r}_A(i)\mathbf{d}_m^H(i)]$ using the pilot symbols. The iterative Conjugate-Gradient algorithm proposed for the MIMO receiver [4] involves two steps in J iterations for the m^{th} transmit antenna as:

1) Initialization

$$\mathbf{v}_{m,0} = \mathbf{0} \tag{3}$$

$$\gamma_0 = \mathbf{h}_m; \quad \mathbf{\Delta}_0 = \mathbf{h}_m \tag{4}$$

$$\delta_0 = \gamma_0^H \gamma_0; \quad \delta_1 = \delta_0 \tag{5}$$

2) Iteration: for j = 1 : J

$$\boldsymbol{\Gamma}_{j} = \boldsymbol{\mathrm{R}}_{rr} \boldsymbol{\Delta}_{j-1} \quad \delta_{j+1} = \boldsymbol{\Gamma}_{j}^{H} \boldsymbol{\Gamma}_{j} \tag{6}$$

$$\alpha = \delta_j / \boldsymbol{\Delta}_{j-1}^H \boldsymbol{\Gamma}_j; \quad \beta = \delta_{j+1} / \delta_j \tag{7}$$

$$\mathbf{v}_{m,j} = \mathbf{w}_{m,j-1} + \alpha \mathbf{\Delta}_{j-1} \tag{8}$$

$$\mathbf{\hat{A}} = \mathbf{\hat{\Gamma}} + \beta \mathbf{\hat{A}} + \mathbf{\hat{A}}$$
(10)

$$\Delta_j = \mathbf{1}_j + \beta \Delta_{j-1} \tag{10}$$

The matrix-vector multiplication $\Gamma_j = \mathbf{R}_{rr} \Delta_{j-1}$ in the iteration has the dominant complexity. Both the covariance matrix and the channel estimation vector could be partitioned to sub matrices and vectors according to the number of Rx antennas. The N(L+1) * N(L+1) covariance matrix is constructed as $\mathbf{R}_{rr} = (\mathbf{R}_{n_1,n_2})$ where each of the submatrices \mathbf{R}_{n_1,n_2} is the cross-covariance matrix between the Rx antennas n_1 and n_2 . The Δ_{j-1} vector is a N(L+1) vector and can also be partitioned as $\Delta_{j-1} = [\Delta_{j-1,1}, \cdots, \Delta_{j-1,N}]^T$. Thus, the matrix-vector multiplication in $\Gamma_j = \mathbf{R}_{rr} \Delta_{j-1}$ is participated into sub-blocks as $\Gamma_j = \left(\sum_{n=1}^N \mathbf{R}_{1,n} \Delta_{j-1,n}, \cdots, \sum_{n=1}^N \mathbf{R}_{N,n} \Delta_{j-1,n}\right)^T$. A standard VLSI architecture for the Direct Matrix-vector Multiplication (DMM) $\Gamma_{j,n_1,n_2} = \mathbf{R}_{n_1,n_2} \Delta_{j-1,n_2}$ is shown in Fig. 1.

B. Frequency-domain Computing Architecture

It is easy to show that the covariance matrix has banded block Toeplitz structure with only N(L+1) independent elements $E_{n_1,n_2}(l)$ for $l \in [0, L]$ [5]. By defining the upper triangular corner matrix as

$$\mathbf{C}_{n_1,n_2}^L = \begin{pmatrix} E_{n_1,n_2}(L) & \cdots & E_{n_1,n_2}(1) \\ & \ddots & \vdots \\ 0 & & E_{n_1,n_2}(L) \end{pmatrix}, \quad (11)$$

we can expand the \mathbf{R}_{n_1,n_2} matrix to an Expanded-Form matrix with concatenation as

$$\mathbf{Z}_{n_1,n_2} = \left(\begin{array}{c} \mathbf{C}_{n_1,n_2}^L \\ \mathbf{0}_{1 \times L} \end{array} \right) \quad \mathbf{R}_{n_1,n_2} \quad \left(\begin{array}{c} \mathbf{0}_{1 \times L} \\ (\mathbf{C}_{n_1,n_2}^L)^H \end{array} \right) \right).$$
(12)



Fig. 1. A standard VLSI architecture for the DMM computation

It can be shown that this is a Toeplitz matrix of size $(L+1) \times (3L+1)$. Thus the matrix computation by the correlation matrix is equal to

$$\Gamma_{j,n_1,n_2} = \mathbf{Z}_{n_1,n_2} \cdot \begin{pmatrix} \mathbf{0}_{L \times 1} \\ \mathbf{\Delta}_{j-1,n_2} \\ \mathbf{0}_{L \times 1} \end{pmatrix}.$$
(13)

1) Frequency domain FFT Acceleration: From the sliding window feature of the \mathbf{Z}_{n_1,n_2} matrix, the matrix-vector multiplication is actually a linear convolution, where the filter taps vector is $\mathbf{Y}_{n_1,n_2} = [E_{n_1,n_2}^*(L), \cdots, E_{n_1,n_2}^*(1), E(0), \cdots, E_{n_1,n_2}(L)]$ and the input data vector is $\mathbf{X}_{n_2} = [\mathbf{0}_{1 \times L}, \ \mathbf{\Delta}_{j-1,n_2}^T, \ \mathbf{0}_{1 \times L}]$. This linear convolution can be implemented using the time-domain linear FIR filter where the general equation is given by $\Gamma_{j,n_1,n_2}(l) =$ $\sum_{k=0}^{2L} y_{n_1,n_2}(k) \Delta_{j-1,n_2}(L-l+k) \text{ for } \{\Delta_{j-1,n_2}(l) = 0, l < 0 \text{ and } l > L\}.$ On the other hand, from the FIR filtering interpretation and the features of the FFT algorithm, we can implement the linear convolution from the circular convolution by using the FFT in the frequency-domain. To convert the circular convolution to the linear convolution, we need to take care of the edges to avoid frequency alias. It is known that there are two equivalent architectures for the FFT-based filtering of a long sequence, i.e., the "overlap-add" and "overlap-save" architectures [7]. Because of the simplicity in the control logic design, the "overlap-save" based architecture is applied in the derivation. The details are omitted here due to the limited space. It can be shown that the complexity is dominated by two FFTs and one IFFT of length (3L + 1), which is given by $3(3L+1)\log_2(3L+1)/2 + (3L+1)$ complex multiplications. In this way, we have reduced the complexity from $O((L+1)^2)$.

2) Reversed-Form FFT Acceleration Architecture: To further reduce the complexity, the matrix multiplication in $\Gamma_{j,n_1,n_2} = \mathbf{R}_{n_1,n_2} \Delta_{j-1,n_2}$ is transformed to the following reversed-format matrix-vector multiplication, where the elements of Δ_{j-1,n_2} form the Topelitz matrix and the independent elements of the correlation matrix form the multiplying vector $\mathbf{Y}_{n_1,n_2} = (E_{n_1,n_2}^*(L), \cdots, E_{n_1,n_2}(0), \cdots, E_{n_1,n_2}(L))^T$. Thus, Γ_{j,n_1,n_2} equals to

$$\begin{array}{cccc} \Delta_{j-1,n_2}(L) & \cdots & \Delta_{j-1,n_2}(0) & 0 \\ 0 & & & \\ \vdots & \ddots & & \ddots \\ 0 & \Delta_{j-1,n_2}(L) & \cdots & \Delta_{j-1,n_2}(0) \end{array} \! \! \right) \! \ast \! \mathbf{E}_{n_1,n_2}$$



Fig. 2. "Overlap-save" FFT-based architecture for the Matrix-vector Multiplication in CG iteration.

which can be viewed as data sequence \mathbf{Y}_{n_1,n_2} filtered by the tap coefficients Δ_{j-1,n_2} . With the "overlap-save" FFT-based FIR filtering architecture, the filter Δ_{j-1,n_2} is first padded with zeros to make a vector of length (2L + 1). The vector $[E_{n_1,n_2}^*(L), \cdots, E_{n_1,n_2}^*(1)]$ is considered as the L "overlap-save" values from the first block. They are concatenated with the (L +1) new values $[E_{n_1,n_2}(0), E_{n_1,n_2}(1), \cdots, E_{n_1,n_2}(L)]$ to make a vector of length (2L + 1). This vector is input to the FFT module of length (2L + 1). The FFT results are multiplied with the FFT result of the filter taps. An IFFT module computes the interim results and the first L samples are dropped to generate the final result as in $\Gamma_{j,n_1,n_2}(l) = \lambda_{n_1,n_2}(l+L)$ $l \in [0, L]$.

This procedure is shown in Fig. 2. The number of complex multiplications is given by $(6L + 3) \log_2(3L + 1)/2 + (3L + 1)$. After extracting the commonality, the optimized iterative algorithm with the reversed-form FFT acceleration is summarized as following. For the m^{th} transmit antenna, the iteration is accelerated with FFT in the frequency domain:

- 1) Initialization: Same initialization for $\mathbf{w}_{m,0}$, γ_0 , Δ_0 and δ_0 , δ_1 as a general CG algorithm. Moreover, initialize $\Psi_{n_1,n_2} = FFT(\mathbf{Y}_{n_1,n_2})$ for $n_1 \in [1, N]$ and $n_2 \in [1, N]$.
- Iteration: for j = 1 : J: To compute the MIMO matrix-vector multiplication Γ_j = **R**_{rr}Δ_{j-1}in frequency domain, we first initialize the partitioned vector Γ_j = [Γ^T_{j,1}, · · · , Γ^T_{j,N}]^T = **0**. Then, for n₂ = 1 : N, we compute the element-wise FFT **F**_{j,n2} = FFT([Δ^T_{j-1,n2} **0**_{1×L}]). In the inner loop for n₁ = 1 : N, the frequency domain dot-product and the element-wise IFFT are computed as following:

$$\boldsymbol{\Phi}_{n_1,n_2} \quad = \quad \mathbf{F}_{j,n_2} \circ \boldsymbol{\Psi}_{n_1,n_2}; \tag{14}$$

$$\Theta_{n_1,n_2} = IFFT(\Phi_{n_1,n_2}); \tag{15}$$

$$\Gamma_{j,n_1,n_2}(l) = \theta_{n_1,n_2}(l+L) \quad l \in [0,L];$$
(16)

$$\boldsymbol{\Gamma}_{j,n_1} = \boldsymbol{\Gamma}_{j,n_1} + \boldsymbol{\Gamma}_{j,n_1,n_2}; \quad (17)$$

where "o" denotes "dot-product". With the partitioned Γ_j vector, the scalars are computed as $\delta_{j+1} = \sum_{n=1}^{N} \Gamma_{j,n}^H \Gamma_{j,n}$; $\omega_j = \sum_{n=1}^{N} \Delta_{j-1,n}^H \Gamma_{j,n}$; $\alpha = \delta_j / \omega_j$ and $\beta = \delta_{j+1} / \delta_j$. The final equalizer taps are adjusted in the same manner as the conventional CG iteration.

Because the FFT of the covariance vectors is not dependent on the iteration and transmit antenna, we can first compute the FFT results



Fig. 3. The BER performance of a fully loaded system (K = 14) under 13 taps modified Pedestrian-B channel for a 2×2 MIMO system.

of the covariance vectors in the initialization part common to all M transmit antennas and J iterations. The result is saved in distributed $\Psi_{N\times N}$ sub-vectors for the N receive antennas. The channel vector is also partitioned into N sub-vectors. This reduces the number of FFTs for this part from $(N^2 * J * M)$ to only N^2 . In the iteration part, the Γ_j vector is first partitioned and initialized to zero vectors. By re-arranging the loop structure of n_2 and n_1 , the FFT of the zero-padded Δ_{j-1,n_2} vector is only necessary for the outer loop. Thus, we only need to compute the frequency domain dot-product and the IFFT for individual Γ_{j,n_1,n_2} in the inner loop. An accumulator will generate the Γ_{j,n_1} sub-vectors. After the n_1, n_2 loop, two scalars δ_{j+1} and ω_j are computed from the inner product.

IV. PERFORMANCE AND COMPLEXITY

A. Performance

We provide the simulation results in a MIMO HSDPA simulation chain for long delay spread channels. We compare the performance of four different schemes: the LMS algorithm, the FFT circulant approximation in [5], the DMI using Cholesky decompositions and the proposed FFT accelerated CG algorithm. The I-METRA modified Pedestrian-B channel models with 13 and 15 paths and mobile speed of 3 km/h [8] are simulated. The channel state information is estimated from the CPICH (Common Pilot Channel) at the receiver. 10% of the total transmit power is dedicated to the pilot symbols. In the figures, L_h is shown as the channel delay spread. Fig. 3 and Fig. 4 show the performance for $L_h = 13$ and $L_h = 15$ respectively. To show the potential of the iterative algorithm, 8 iterations are computed. Both the CG and FFT-based algorithm show small divergence from the DMI at the small to medium SNR range. Both the CG and the circulant approximation FFT solution are much better than the LMS solution. For the very high SNR range, we see the performance degradation with the circulant approximation in this very bad channel situation. In the range of SNR > 16 dB, the FFTbased circulant approximation even fails because the high condition number corrupts the stability of the system equation. But the FFT accelerated iterative algorithm follows the DMI solution very closely.

B. Numerical Stability

The system stability is determined by the condition number of the system matrix. The condition number of a matrix is defined as the



Fig. 4. The BER performance of fully loaded system (K=14) under 15 taps modified Pedestrian-B channel for a 2×2 MIMO system.

TABLE I Condition Number vs. SNR: $L_h = 13 \& L_h = 15$

SNR(dB)	8	12	16	20	24	L_h
$\kappa(\mathbf{R}_{rr})$	7.9	14.3	28.7	44.9	79.7	
$\kappa(\mathbf{R}_{cir})$	11.0	53.5	97.6	417.8	1280	13
$\kappa(\mathbf{R}_{rr})$	16.6	32.1	55.3	64.3	92.2	
$\kappa(\mathbf{R}_{cir})$	33.7	75.2	190	1501	3302	15

ratio of the maximal eigenvalue over the minimum eigenvalue as $\kappa(\mathbf{R}_{rr}) = \frac{\lambda_{max}(\mathbf{R}_{rr})}{\lambda_{max}(\mathbf{R}_{rr})}$. If the condition number is large, the matrix tends to be ill conditioned with the convention that $\kappa = \infty$ for a singular matrix. An ill conditioned matrix has very bad numerical sensitivity in matrix inverse. In the following, we analyze the 2-norm condition number of the original covariance matrix and the circulant approximation matrix used in [5] for different SNR ranges. In the top part of Table I, the condition number $\kappa(\mathbf{R}_{rr})$ is shown for an $L_h =$ 13 channel. The \mathbf{R}_{rr} and \mathbf{R}_{cir} are the original covariance matrix and the circulant approximation matrix, respectively. The original matrix is reasonably well or mild conditioned. The condition number increases for higher SNR. After the corner is added to make \mathbf{R}_{cir} , the condition number becomes worse with increasing SNR. This is compatible with the analysis and reduces the numerical stability in the tap solver. In the bottom part of Table I, we analyze the condition number for a more difficult channel case when $L_h = 15$. First we notice that the condition number is higher than the $L_h = 13$ case. After adding the corner to form \mathbf{R}_{cir} , the condition number increases dramatically, especially for SNR = 24 dB, $\kappa(\mathbf{R}_{cir})$ becomes 3290, which may lead to the singularity of the circulant matrix for fixedpoint numerical solution with limited word length.

C. Complexity

Fig. 5 shows the complexity with different CG iteration architectures for increasing channel length. The DMI is not shown because its $O((NL)^3)$ complexity is obviously much higher. DMM/FIR-EF refers to the Direct-Matrix Multiplication using Expanded-Form FIR as in (13). DMM/FIR-RF denotes the Reversed-Form FIR architecture. It is shown that the conventional CG algorithm using either DMM architecture has $O\{MJ(N(L+1))^2\}$ complexity and the EF's complexity is higher than the RF because RF's filter length is shorter. FFT-EF indicates the FFT-based architecture with the Expanded-Form in (12) and FFT-RF denotes the FFT-based architecture for



Fig. 5. The complexity of equivalent Matrix-Vector Multiplication architectures.

the reversed-form matrix. It is shown that both have complexity at order of several FFT operations. Because the number of FFTs in the iterative MIMO chip equalizer is $(N^2 + MNJ)$ and the number of IFFTs is $M * (N^2 + J)$, the dominant complexity is given by $O\{(M * N^2 + MNJ)/2 * (2L + 1) * \log_2(2L + 1)\}$ for simplicity. Moreover, the proposed FFT-RF has the lowest complexity because the FFT length is reduced. It has been shown in [5] that the complexity of the FFT-based circulant approximation is $O\{[(N^2/2 + 2MN)(\log_2 L_F) + (N^3 + MN^2)]L_F/2\}$, where the $L_F \leq 3L + 1$ is determined by simulation. Thus, the FFT-RF acceleration has comparable complexity as the circulant approximation but better numerical stability and BER performance for very long channels.

V. CONCLUSION & ACKNOWLEDGEMENTS

In this paper, we present a novel reversed form FFT-accelerated iterative algorithm to reduce the complexity of a MIMO chip equalizer from $O((NL)^2)$ to $O(NL \cdot \log_2(NL))$. The proposed iterative architecture demonstrates strong numerical stability, especially for very long channels. Dr. J. R. Cavallaro is supported in part by NSF under grants ANI-9979465 and EIA-0224458 and EIA-0321266.

REFERENCES

- A. Wiesel, L. Garca, J. Vidal, A. Pags and Javier R. Fonollosa, *Turbo linear dispersion space time coding for MIMO HSDPA systems*, 12th IST Summit on Mobile and Wireless Communications, Aveiro, Portugal, June 15-18, 2003.
- [2] G. D. Golden, J. G. Foschini, R. A. Valenzuela and P. W. Wolniansky, Detection algorithm and initial laboratory results using V-BLAST spacetime communication architecture, Electron. Lett., Vol. 35, pp.14-15, Jan. 1999.
- [3] Y. Guo, J. Zhang, D. McCain and J. R. Cavallaro, Scalable FPGA architectures for LMMSE-based SIMO chip equalizer in HSDPA downlink, 37th IEEE Asilomar Conference, Monterey, CA, 2003.
- [4] P. Radosavljevic, J. R. Cavallaro and A. D. Baynast, ASIP architecture implementation of channel equalization algorithms for MIMO systems in WCDMA downlink, IEEE Vehicular Technology Conference (VTC), Los Angeles, CA, Sept. 2004.
- [5] Y. Guo, J. Zhang, D. McCain and J. R. Cavallaro, *Efficient MIMO equalization for downlink multi-code CDMA: complexity optimization and comparative study*, to appear in IEEE GlobeCom 2004.
- [6] V. Y. Pan, Structured matrices and polynomials: unified superfast algorithms, springer, 2001.
- [7] J. G. Proakis, D. G. Manolakis, Digital Signal Processing: Principles, Algorithms and Applications (3rd Edition), Prentice Hall, 1996.
- [8] J. P. Kermoal, L. Schumacher, K. Pedersen and P. Mogensen, A Stochastic MIMO radio channel model with experimental validation, IEEE JSAC, Vol. 20, No. 6, pp.1211-1226, August 2002.