IMPLEMENTATION OF THE LEAST SQUARES CHANNEL ESTIMATION ALGORITHM FOR MIMO-OFDM SYSTEMS

Samuli Tiiro, Jari Ylioinas, Markus Myllylä, and Markku Juntti

Centre for Wireless Communications
University of Oulu
P.O. Box 4500, FI-90014 University of Oulu, Finland
{samuli.tiiro,jari.ylioinas,markus.myllyla,markku.juntti}@ee.oulu.fi

ABSTRACT

The least squares (LS) channel estimation algorithm for a multiple-input multiple-output (MIMO) system with orthogonal frequency division multiplexing (OFDM) is considered in this paper. Two architectures for the algorithm are presented and the architectures are implemented using fixed point arithmetic. The minimum word lengths for the implementations are determined through computer simulations for 2×2 and 4×4 MIMO systems with quadrature phase-shift keying (QPSK) modulation. Field-programmable gate array (FPGA) synthesis simulation is done for the architectures using the obtained word lengths which provides complexity and latency results. The algorithm implementations are shown to consume reasonably small amount of hardware resources.

1. INTRODUCTION

Orthogonal frequency division multiplexing (OFDM) [1] is an attractive air interface for high-rate communication systems with large bandwidths. OFDM divides the frequency selective fading channel into several parallel flat fading subchannels and, thus, efficiently mitigates the multipath delay spread. The use of multiple transmit and receive antennas, i.e., multiple-input multiple-output (MIMO) channel provides the potential for tremendous capacity increase [2, 3]. Consequently, MIMO techniques [4, 5] have gained remarkable attention during the recent years. Combining MIMO processing [6, 7] with OFDM is the key enabling technology for several current and future broadband wireless access systems and standards. Important examples include wireless local area network (WLAN) or Wireless Fidelity (WiFi) systems [8], wireless metropolitan area network (WMAN), like Worldwide Interoperability for Microwave Access (WiMAX) [9], and the Third Generation (3G) cellular system Long Term Evolution (LTE) [10].

This research was financially supported in part by Tekes, the Finnish Funding Agency for Technology and Innovation, Nokia, Texas Instruments, Nokia Siemens Networks and Elektrobit. Catapult C Synthesis license was provided by Mentor Graphics.

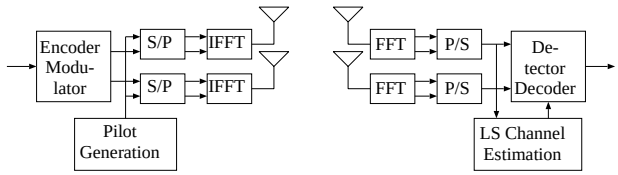

Fig. 1. 2×2 MIMO-OFDM system.

The channel state information (CSI) between each transmit and receive antenna pair is required at the receiver to coherently detect the information. Therefore, channel estimation is an important task that is required in wireless communication systems. Pilot-based channel estimation is a popular technique due to its simple implementation. In pilot-based channel estimation [11], known training symbols are inserted in the data stream, and the receiver can then use these symbols to perform channel estimation and synchronization. The least squares (LS) estimation is a common method for pilot-based channel estimation, as it offers good performance with reasonable complexity.

In this paper, a pilot-based channel estimator design for a MIMO-OFDM system is described and two channel estimator architectures are presented for the LS estimator. A word length study is done for both architectures in order to determine the minimum fixed point word lengths for the field-programmable gate array (FPGA) hardware implementation. The minimum word lengths are then used for obtaining hardware complexity results for the presented estimator architectures.

The rest of this paper is organized as follows. Section 2 describes the system model and the channel estimation algorithm is covered in Section 3. The proposed channel estimator architectures and the word length study can be found in Section 4. Synthesis results are presented in Section 5, and finally, paper is concluded in Section 6.

2. SYSTEM MODEL

In this paper, we consider a system model similar to the one presented in [12], and it is described as follows. Let us assume a MIMO-OFDM system with N_{T} transmit antennas and N_{R} receive antennas. As an example, a block diagram of a system with $N_{\mathrm{T}} = N_{\mathrm{R}} = 2$ is illustrated in Fig. 1. The discrete time channel impulse response (CIR) from the ith transmit antenna to the jth receive antenna at time n is denoted by $\boldsymbol{h}_{ij}[n] \in \mathbb{C}^L$ where L denotes the number of channel tap coefficients. The CIR for the entire $N_{\mathrm{T}} \times N_{\mathrm{R}}$ system at time n is defined in matrix form as

$$\boldsymbol{H}[n] = \begin{pmatrix} \boldsymbol{h}_{11}[n] & \dots & \boldsymbol{h}_{1N_{\mathrm{R}}}[n] \\ \vdots & \ddots & \vdots \\ \boldsymbol{h}_{N_{\mathrm{T}}1}[n] & \dots & \boldsymbol{h}_{N_{\mathrm{T}}N_{\mathrm{R}}}[n] \end{pmatrix} \in \mathbb{C}^{N_{\mathrm{T}}L \times N_{\mathrm{R}}}. \tag{1}$$

Correspondingly, we define the channel frequency response (CFR) matrix at time n

$$\tilde{\boldsymbol{H}}[n] = \begin{pmatrix} \boldsymbol{F}_K^L \boldsymbol{h}_{11}[n] & \dots & \boldsymbol{F}_K^L \boldsymbol{h}_{1N_{\mathrm{R}}}[n] \\ \vdots & \ddots & \vdots \\ \boldsymbol{F}_K^L \boldsymbol{h}_{N_{\mathrm{T}}1}[n] & \dots & \boldsymbol{F}_K^L \boldsymbol{h}_{N_{\mathrm{T}}N_{\mathrm{R}}}[n] \end{pmatrix} \in \mathbb{C}^{N_{\mathrm{T}}K \times N_{\mathrm{R}}}, \tag{2}$$

where $\boldsymbol{F}_K^L$ denotes the L first left columns of a K-point DFT matrix and K is the number of subcarriers used for transmitting data. The received signal block at the receiver at time n after the DFT can be presented as

$$(\boldsymbol{r}_1[n] \dots \boldsymbol{r}_{N_{\mathrm{R}}}[n]) = \boldsymbol{R}[n] = \boldsymbol{X}[n]\tilde{\boldsymbol{H}}[n] + \boldsymbol{\eta}[n], \tag{3}$$

where $\boldsymbol{X}[n] = \big(\mathrm{diag}(\boldsymbol{x}_1[n]) \dots \mathrm{diag}(\boldsymbol{x}_{N_{\mathrm{T}}}[n])\big) \in \mathbb{C}^{K \times N_{\mathrm{T}}K}$ and $\boldsymbol{x}_i[n] \in \Xi^K, i = 1, \dots, N_{\mathrm{T}}$ are the data vectors sent from the ith transmit antenna, Ξ denotes the modulation alphabet, $\boldsymbol{r}_j[n] \in \mathbb{C}^K$ is the data vector received at the jth receive antenna, and $\boldsymbol{\eta}[n] \in \mathbb{C}^{K \times N_{\mathrm{R}}}$ is the noise matrix at time n. White Gaussian noise is assumed with covariance matrix $\boldsymbol{\Sigma}_\eta = \mathrm{E}(\mathrm{vec}(\boldsymbol{\eta})\mathrm{vec}(\boldsymbol{\eta})^H) = \sigma_\eta^2 \boldsymbol{I}_{KN_{\mathrm{R}}}$. From (2) and (3), it can be seen that by combining the DFT matrices with the data matrix $\boldsymbol{X}[n]$, the received signal can also be expressed in terms of CIR as

$$\boldsymbol{R}[n] = \boldsymbol{\mathcal{C}}[n]\boldsymbol{H}[n] + \boldsymbol{\eta}[n], \tag{4}$$

where $\boldsymbol{\mathcal{C}}[n] = (\boldsymbol{\mathcal{C}}_1[n] \dots \boldsymbol{\mathcal{C}}_{N_{\mathrm{T}}}[n]) \in \mathbb{C}^{K \times N_{\mathrm{T}}L}$ and $\boldsymbol{\mathcal{C}}_i[n] = \mathrm{diag}(\boldsymbol{x}_i[n])\boldsymbol{F}_K^L$.

The channel model used in this work is a Ricean fading channel with temporal, spatial and spectral correlation proposed in [13].

3. PILOT-BASED LEAST SQUARES CHANNEL ESTIMATION

We consider an OFDM frame structure where each frame consists of N_{SYMB} consecutive OFDM symbols. The first and the third last are reserved for transmitting the pilot sequences. Out of the N_{C} available subcarriers, K are used for transmitting data and the pilot sequences are transmitted over all K tones. The pilot symbols are used for obtaining an estimate of the channel impulse response which is then used for the coherent detection of the transmitted symbols. From (4), the LS estimate of the CIR becomes

$$\hat{\boldsymbol{H}}_{\mathrm{LS}}[n] = (\boldsymbol{\mathcal{C}}^H\boldsymbol{\mathcal{C}})^{-1}\boldsymbol{\mathcal{C}}^H\boldsymbol{R}[n], \tag{5}$$

where n denotes the time indices of the pilot symbols. The time index of $\boldsymbol{\mathcal{C}}$ can be neglected as the pilot sequences do not depend on the block index n. It should be noted that if the noise $\boldsymbol{\eta}$ in (4) is white and Gaussian, then the LS estimator is also the maximum likelihood (ML) estimator and optimal in the sense that it attains the Cramer-Rao lower bound (CRLB).

The computation of the LS estimate requires an $N_{\mathrm{T}}L \times N_{\mathrm{T}}L$ matrix inversion which is computationally an intensive task. It was shown in [14] that with a proper pilot sequence design, the LS channel estimator for MIMO-OFDM can be simplified so that the matrix inversion is not needed while the estimator still achieves the minimum mean square error (MSE). In this work, local orthogonal pilot sequences were used and the sequence for ith transmit antenna is defined as

$$\boldsymbol{c}_i[k] = c_0[k]e^{j(2\pi\bar{L}/K)ik} \in \mathbb{C}^K, \tag{6}$$

where $\bar{L} \triangleq \lfloor \frac{K}{N_{\mathrm{T}}} \rfloor \geq L$, $i = 1, \dots, N_{\mathrm{T}} - 1$ and $k = 0, \dots, K - 1$. The training sequence at the first antenna is set to a random sequence of symbols for which $|\boldsymbol{c}_0[k]| = 1 \,\forall\, k$. The presented pilot sequence design reduces the matrix $(\boldsymbol{\mathcal{C}}^H\boldsymbol{\mathcal{C}})$ to a diagonal matrix $K\boldsymbol{I}_{N_{\mathrm{T}}L}$ and, thus, (5) can be rewritten as

$$\hat{\boldsymbol{H}}_{\mathrm{LS}}[n] = \frac{1}{K}\boldsymbol{\mathcal{C}}^H\boldsymbol{R}[n]. \tag{7}$$

4. ARCHITECTURE DESIGN AND WORD LENGTH STUDY

4.1. LS Estimator Architecture

As can be seen from (7), the channel estimator only needs to compute a complex matrix-matrix multiplication followed by scaling with a real value $\frac{1}{K}$. In OFDM systems, the number of subcarriers is usually chosen to be a power of two due to the FFT used in modulation and demodulation. In the case where all subcarriers are used for data transmission, i.e., $K = N_{\mathrm{C}}$ the multiplication can therefore be performed with a simple shift operation. An architecture for performing the computation in (7) is designed for 2×2 and 4×4 MIMO systems using quadrature phase-shift keying (QPSK). The top level block diagram for the proposed architecture for a 2×2 system is shown in Fig. 2. The channel estimator consists of $N_{\mathrm{T}}N_{\mathrm{R}}$ vector product units, and it

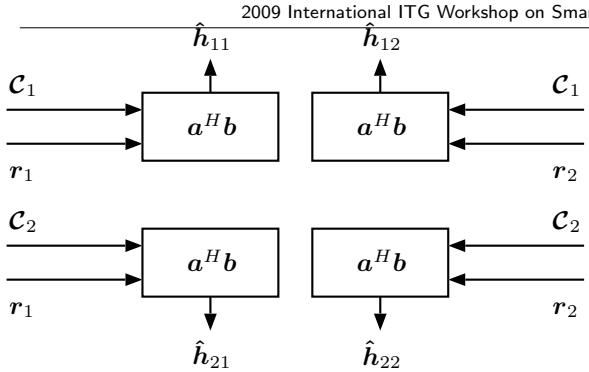

Fig. 2. Top level structure of the LS estimator for a 2×2 MIMO system.

is therefore easily scalable for different MIMO configurations. Each of the vector product units compute the CIR tap estimates for the corresponding MIMO link as

$$\hat{\boldsymbol{h}}_{ij}[n;l] = \frac{1}{K}\boldsymbol{\mathcal{C}}_{il}^H \boldsymbol{r}_j[n] \tag{8}$$

where $i = 1,\ldots,N_\mathrm{T}$, $j = 1,\ldots,N_\mathrm{R}$, $l = 0,\ldots,L-1$ and $\boldsymbol{\mathcal{C}}_{il}$ denotes the lth column of $\boldsymbol{\mathcal{C}}_i$. In one computing cycle, each of the units can compute one channel tap estimate. Therefore, the computation of a single CIR vector $\hat{\boldsymbol{h}}_{ij}$ takes L computing cycles in total. As there are no data dependencies between the different CIR vectors, it is possible to compute all of them simultaneously. In the proposed architecture, all of the $N_\mathrm{T}N_\mathrm{R}$ vector product units are ran in parallel, and, therefore, the entire CIR matrix $\hat{\boldsymbol{H}}$ can be computed in L computing cycles.

A block diagram of the vector product unit is shown in Fig. 3. The unit has four scalar inputs and two scalar outputs. The input vectors' elements are read from inputs $1-4$, where 1 and 2 are for the real and imaginary parts of $\boldsymbol{\mathcal{C}}_{il}$ and 3 and 4 are for the real and imaginary parts of $\boldsymbol{r}_j$, respectively. The vector product is computed in a straightforward manner by multiplying the elements of the two complex input vectors and then adding the product to the sum of previous elements' products. After all K elements of the input vectors are multiplied, the final sum is multiplied by the scaling factor $\frac{1}{K}$. The final result, i.e., the real and imaginary parts of $\hat{\boldsymbol{h}}_{ij}[n;l]$ can then be read from outputs 14 and 15, respectively. The complex multiplication of the vector elements is done with three real multipliers in order to simplify the implementation. For two complex numbers $z_1 = a + bi$ and $z_2 = c + di$, where $a,b,c,d \in \mathbb{R}$ and $i = \sqrt{-1}$, the product $z_1^* z_2$ can be expressed in terms of real and imaginary parts as $\Re\{z_1^* z_2\} = ac + bd$ and $\Im\{z_1^* z_2\} = ad - bc$ which requires four real multiplications. Same result can be achieved with only three real multiplications by rearranging the equations as

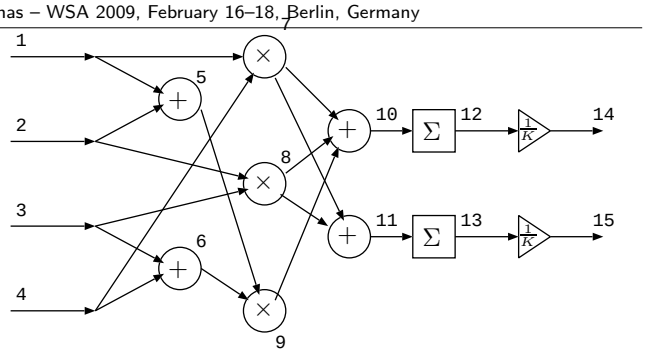

Fig. 3. Block diagram of the vector product unit.

$$\begin{aligned} \Re\{z_1^* z_2\} &= (a+b)(c+d) - ad - bc \\ \Im\{z_1^* z_2\} &= ad - bc. \end{aligned} \tag{9}$$

In (9), one multiplication is exchanged to three additions. This approach still results in a lower overall complexity as multiplication is computationally much more expensive operation than an addition. It should be noted that in Fig. 3 the signal sign flips required for the subtraction operations are left out.

Computing the LS channel estimate by using the precomputed matrix $\boldsymbol{\mathcal{C}}$ requires memory to store the $N_\mathrm{T}LK$ complex matrix coefficients. As matrix $\boldsymbol{\mathcal{C}}$ consists of diagonal matrices $\mathrm{diag}(\boldsymbol{c}_i), i = 1,\ldots,N_\mathrm{T}$ multiplied by the partial DFT matrix $\boldsymbol{F}_K^L$, it is possible to reduce the memory needed for the precomputed data. By adding one complex multiplication into the structure shown in Fig. 3, the LS estimate can be computed as

$$\hat{\boldsymbol{h}}_{ij}[l] = \frac{1}{K}(\boldsymbol{c}_i \odot (\boldsymbol{F}_K^L)_l)^H \boldsymbol{r}_j \tag{10}$$

where $(\boldsymbol{F}_K^L)_l$ is the lth column of the partial DFT matrix, $i = 1,\ldots,N_\mathrm{T}$, $j = 1,\ldots,N_\mathrm{R}$, $l = 0,\ldots,L-1$ and $\odot$ denotes elementwise vector multiplication. With this approach, only the pilot sequences $\boldsymbol{c}_i$ and the partial DFT matrix $\boldsymbol{F}_K^L$, i.e., $K(L+N_\mathrm{T})$ complex coefficients in total need to be stored in memory. This is, however, done on the expense of the computational complexity as an additional complex multiplication is required in each vector product unit. A block diagram of the vector product unit using this alternative method is shown in Fig. 4.

4.2. Word Length Study

In practical applications, fixed point presentation of the numbers is usually preferred over floating point because of its lower complexity. The downside of using fixed point numbers is the limited range of signals and loss of precision. A

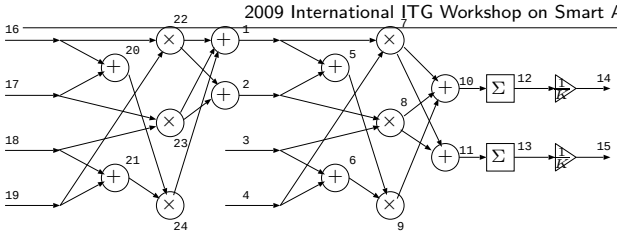

Fig. 4. Block diagram of the alternative vector product unit.

Table 1. System parameters for different channel profiles.

	EPA	ETU
CP length	4.69 μs	16.67 μs
Symbols per frame	7	6
OFDM symbol duration	71.35 μs	83.33 μs
Number of channel taps	9	7

word length study of the LS estimator was done in order to obtain the minimum word lengths for which the fixed point implementation does not exhibit significant performance degradation.

The word lengths were determined with computer simulations using 3G LTE system parameters with two different channel profiles. The channel profiles considered in this work were Extendend Typical Urban (ETU) and Extended Pedestrian-A (EPA) [15]. The simulation parameters for both channel profiles are presented in Table 1. The systems were simulated with mobile velocity of 70 km/h. 2×2 and 4×4 MIMO systems with QPSK were considered. Out of the $N_{\mathrm{C}} = 512$ available subcarriers $K = 300$ were used for transmitting data. System bandwidth was 5 MHz with the carrier frequency of 2.4 GHz. The data was coded using a $\frac{1}{2}$-rate turbo code with 8 decoder iterations and the coding was performed over a single OFDM frame. As the receiver front-end, a K-best list sphere detector (LSD) [16] was used with list sizes of 8 for 2×2 QPSK system and 32 for 4×4 QPSK system.

In a system where there are no null subcarriers present, it is sufficient to consider only the L first taps of the CIR as the rest of the taps only contain noise energy. By excluding these noise taps, the estimation accuracy is improved. This also results in simpler channel estimation as less taps are required to be estimated. Due to the null subcarriers used in the system considered here the data is demodulated with a 512-point FFT while the LS estimation is performed using a K-point DFT matrix. This causes some of the energy to leak over all K taps. By only estimating the L first taps of the CIR some performance loss will result due to the leakage energy. It was shown in [17], that if some additional taps beyond the L first are also included in the estimation, the MSE of the LS estimator can be improved. This will increase the size of the data matrix used in the LS estimation and, thus, results in higher estimator complexity. However, most of the leakage energy is contained only in a few taps outside the L first ones and therefore the increase in complexity is minor. An optimal number of additional taps exist for each SNR point that results in the smallest MSE [17] but in order to keep the estimator design simple, a fixed value for each antenna configuration was determined with computer simulations.

Table 2. Additional channel taps considered in the LS estimation.

MIMO	Channel profile	Channel length L	Taps used L'
2×2	EPA	2	$0, \ldots, 4, 297, \ldots, 299$
4×4	EPA	2	$0, \ldots, 4, 297, \ldots, 299$
2×2	ETU	23	$9, \ldots, 26, 296, \ldots, 299$
4×4	ETU	23	$0, \ldots, 25, 297, \ldots, 299$

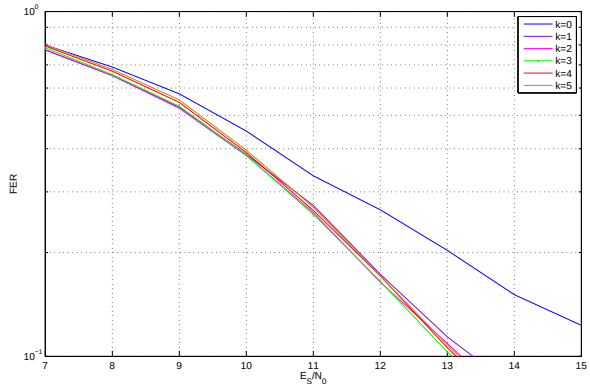

Fig. 5. The FER vs. SNR of the LS estimator for 2×2 QPSK in EPA channel with additional channel taps included in the estimation.

Estimators with different numbers of additional taps were simulated. The taps considered in each case are $\boldsymbol{h}[n] = (h[n;0] \ldots h[n;L+k-1]h[n;K-k] \ldots h[n;K-1])$, i.e., k additional taps on each side and, thus, $2k$ additional taps in total. The simulation length in each case was 1000 frames and the frame error rate (FER) was measured. As an example, the results for EPA channel with 2×2 antenna configuration can be found in Fig. 5, where it can be seen that $k = 3$ results in the best FER performance. It should be noted that the performance difference between $k = 1, \ldots, 5$ is minor and therefore even smaller values of k could be used to obtain a better trade-off between the performance and the complexity. The number of taps that resulted in the best FER performance in each case were used in this work and the taps considered can be found in Table 2.

To obtain the fixed point word lengths, a C language implementation of the channel estimation algorithm was written. The fixed point variables were implemented using Mentor Graphics algorithmic C data types [18]. The integer

widths, the fraction widths and the signedness were defined for all signals in the implementation. The range for a signed fixed point signal is from $(-0.5) \times 2^I$ to $(0.5 - 2^{-W}) \times 2^I$, where W is the total word length and I is the number of bits reserved to represent the integer part.

To determine the integer widths for the input and output signals of the LS estimator, a floating point simulation was performed. Using the signal samples collected during the simulation, a histogram was drawn for each signal from which the signal ranges could be determined. The internal word lengths as well as the fraction widths of the input and output signals were determined from the fixed point simulations. Signal word lengths were studied one by one by comparing the fixed point performance to the corresponding floating point implementation. The word length of each signal was reduced until the minimum that did not exhibit noticeable performance loss was found. The length of each simulation was 10000 frames and the frame error rate (FER) was used as the performance measure. The minimum word lengths were determined so that the fixed point performance was close to the floating point performance up to the FER of 10^{-1}. The obtained minimum word lengths can be found in Table 3. The word lengths therein are presented as (a, b), where a represents the number of bits reserved for the integer part and the sign and b denotes the number of bits reserved for the fraction part.

Table 3. Word lengths for the LS estimator.

	EPA		ETU	
Signal in Figure 3	2×2	4×4	2×2	4×4
1,2	(2,5)	(2,5)	(2,6)	(2,7)
3,4	(2,5)	(4,4)	(3,4)	(4,5)
5	(2,5)	(2,5)	(2,7)	(2,8)
6	(4,3)	(4,4)	(4,5)	(4,6)
7,8	(3,7)	(4,9)	(3,10)	(4,10)
9	(4,7)	(5,8)	(4,10)	(5,11)
10	(3,12)	(4,9)	(4,12)	(5,10)
11	(3,12)	(4,9)	(4,12)	(4,10)
12,12	(10,7)	(10,9)	(9,9)	(8,10)
14,15	(2,8)	(2,8)	(0,10)	(0,10)
Signal in Figure 4				
16,17	(2,5)	(2,5)	(2,5)	(2,7)
18,19	(2,5)	(2,6)	(2,5)	(2,7)
20	(2,5)	(2,5)	(3,5)	(3,6)
21	(2,5)	(2,6)	(3,5)	(3,6)
22	(1,7)	(1,7)	(1,5)	(1,8)
23	(2,5)	(2,6)	(2,5)	(2,7)
24	(2,5)	(2,6)	(3,4)	(2,7)

The performance of the fixed point implementations using EPA channel profile can be found in Fig. 6, where fixed point 1 denotes the architecture using the vector product units shown in Fig. 3 and fixed point 2 denotes the second presented architecture using the vector product units described in Fig. 4. Correspondingly, the results for the estimator using ETU channel profile can be seen in Fig. 7. All of the presented FER results are average values over the 10000 simulated frames and they were simulated using the minimum word lengths found in Table 3. It can be seen that the fixed point implementations provide near floating point performance above the FER of 10^{-1} which was chosen as the threshold value as it corresponds to the operating point of the system.

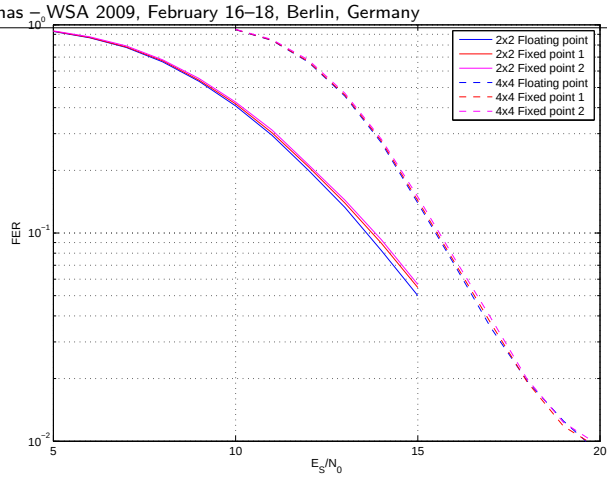

Fig. 6. FER vs. SNR for the LS estimator with EPA channel profile.

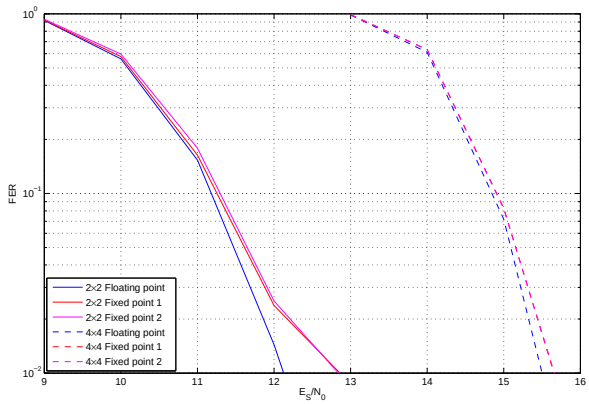

Fig. 7. FER vs. SNR for the LS estimator with ETU channel profile.

5. SYNTHESIS RESULTS

The channel estimation algorithm was implemented using the presented architectures and synthesis simulation was done.

Table 4. Memory requirements for the LS estimator implementations.

Estimator	$\boldsymbol{\mathcal{C}}$	$\boldsymbol{c}$	$\boldsymbol{F}_K^{L'}$	$\boldsymbol{R}$	Total [kbit]
LS 1 (2×2 EPA)	2×33600	-	-	2×4200	75.6
LS 2 (2×2 EPA)	-	2×4200	33600	2×4200	50.4
LS 1 (4×4 EPA)	4×33600	-	-	4×4800	153.6
LS 2 (4×4 EPA)	-	4×4200	38400	4×4800	74.4
LS 1 (2×2 ETU)	2×148800	-	-	2×4800	307.2
LS 2 (2×2 ETU)	-	2×4200	130200	2×4800	148.2
LS 1 (4×4 ETU)	4×156600	-	-	4×5400	648.0
LS 2 (4×4 ETU)	-	4×5400	156600	4×5400	199.8

The system parameters used with the implementations were those presented in Section 4.2. Implementation of the estimator architectures was studied using Mentor Graphics Catapult C tool. The algorithm was implemented in C code and fixed point representation with the obtained minimum word lengths were used for defining the signals. From the C code implementation, the tool created a VHDL code for which a synthesis simulation was performed using Mentor Graphics Precision RTL Synthesis software. The synthesis simulation for the architectures was done by using Xilinx Virtex-4 FPGA with 75 MHz operation frequency as the target platform. The device utilization for each architecture are shown in Table 5. The target latency for the implementations was chosen to be less than the OFDM burst duration $T_{\mathrm{B}} = T_{\mathrm{CP}} + T_{\mathrm{OFDM}}$ so that the channel estimate for one symbol could be computed during the reception of the next symbol. The OFDM burst durations are $T_{\mathrm{B_{EPA}}} = 71.354\mu$s and $T_{\mathrm{B_{ETU}}} = 83.333\mu$s for the EPA and ETU channel models, respectively. The vector product units for the EPA channel are pipelined in order to result in a faster implementation and to meet the latency requirements. Due to the longer length of the ETU channel, more channel taps need to be computed than in the case of the EPA channel. Therefore, the vector product units for the ETU implementation also employ partial unrolling of the computation loops in addition to pipelining. In partial unrolling, some iterations of the computation loop are executed in parallel and it will therefore significantly speed up the implementation. However, this is done on the expense of increased complexity. For the ETU channel, an unrolling factor of 2 was found to be sufficient to meet the latency requirements. Due to the unrolling, a larger memory bandwidth is required so that the vector product units can read two coefficients from the same memory during one clock cycle.

In the synthesis simulation, the memory required for the precomputed matrices was assumed to be available outside the LS estimator and therefore was not included in the device utilization figures. Using the minimum word lengths and the data matrix dimensions, the amount of required memory can be computed. The memory requirements for each design are shown in Table 4. It can be seen that the estimator 1 has lower complexity with every configuration as there is less computation required. The reduced memory requirements of estimator 2, however, are more significant, especially with 4×4 MIMO. As the memory requirements for each design do not exceed the available memory on the platforms [19] used in the synthesis simulations, the choice of architecture can be done based on whether a faster execution or less computation is desired.

Table 5. Synthesis results for the LS estimator implementations.

Estimator	Device	CLB Slices	DSP48s	Latency
LS 1 (2×2 EPA)	XC4VFX12	277	14	32.76 μs
LS 2 (2×2 EPA)	XC4VFX12	317	20	32.87 μs
LS 1 (4×4 EPA)	XC4VLX40	1919	32	34.15 μs
LS 2 (4×4 EPA)	XC4VLX40	2424	32	34.25 μs
LS 1 (2×2 ETU)	XC4VLX25	519	26	64.91 μs
LS 2 (2×2 ETU)	XC4VLX40	749	32	65.32 μs
LS 1 (4×4 ETU)	XC4VLX100	5647	32	65.75 μs
LS 2 (4×4 ETU)	XC4VLX100	7372	32	66.13 μs

The simulated architectures were designed for a system with $K = 300$ data subcarriers. The architectures are quite easily scalable to support different kind of configurations. However, in systems that use more subcarriers for data transmission, maintaining the same latency might become an issue. More intensive loop unrolling can be done for the vector product units to maintain or even to reduce the latency but this will dramatically increase the hardware complexity. The combination of pipelining and unrolling of the computation loops will also increase the number of memory reads per clock cycle and, therefore, larger unrolling factors will also require larger memory bandwidths. Systems with more data subcarriers also require larger DFT matrix and longer pilot sequences to be used and, thus, more memory needs to be available for the precomputed data.

6. CONCLUSIONS

The LS channel estimation algorithm for MIMO-OFDM was considered. A channel estimator architecture based on precomputed data was designed. A modified architecture that

requires less precomputed data to perform the estimation was also presented. The architectures were implemented using fixed point arithmetic for 2×2 and 4×4 MIMO-OFDM systems with 300 data subcarriers and QPSK modulation. Minimum word lengths for each design were studied and they were shown to provide near floating point performance up to 10^{-1} FER. A synthesis simulation was done for the architectures from which latency and complexity results were obtained. The designs were shown to be implementable on the available FPGA platforms as well as to meet the latency requirements based on the system parameters.

7. REFERENCES

[1] R. van Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House, London, 2000.

[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications, Kluwer Academic Publishers*, vol. 6, pp. 311–335, 1998.

[3] E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, vol. 10, no. 6, pp. 585–595, 1999.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, no. 2, pp. 744–765, Mar. 1998.

[5] S. 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.

[6] D. Gesbert, M. Shafi, D. Shiu, P. J. Smith, and A. Naguib, "From theory to practice: An overview of MIMO space-time coded wireless systems," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 3, pp. 281–302, 2003.

[7] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO communications — A key to gigabit wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 198–218, Feb. 2004.

[8] ANSI/IEEE Standard 802.11,1999 Edition (R2003), "Information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," 2003.

[9] IEEE Standard 802.16, "IEEE standard for local and metropolitan area networks part 16: Air interface for fixed broadband wireless access systems," 2004.

[10] 3rd Generation Partnership Project (3GPP), "Physical layer aspect for evolved universal terrestrial radio access (UTRA)," Tech. Rep., 3rd Generation Partnership Project (3GPP), 2006.

[11] J. K. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 4, pp. 686–693, Nov. 1991.

[12] H. Miao and M. Juntti, "Space-time channel estimation and performance analysis for wireless MIMO-OFDM systems with spatial correlation," *IEEE Trans. on Vehicular Technology*, vol. 54, no. 6, pp. 2003 – 2016, Nov. 2005.

[13] E. Kunnari and J. Iinatti, "Stochastic modelling of rice fading channels with temporal, spatial and spectral correlation," *IET Communications*, vol. 1, no. 2, pp. 215–224, Apr. 2007.

[14] Y. Li, "Simplified channel estimation for OFDM systems with multiple transmit antennas," *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, pp. 67–75, Jan. 2002.

[15] 3rd Generation Partnership Project (3GPP); Technical Specification Group Radio Access Network, "Evolved universal terrestrial radio access (E-UTRA); user equipment (UE) radio transmission and reception (release 8)," Tech. Rep., 3rd Generation Partnership Project (3GPP), 2007.

[16] K. Wong, C. Tsui, R. K. Cheng, and W. Mow, "A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, Scottsdale, AZ, May 26-29 2002, vol. 3, pp. 273–276.

[17] J.-J. van de Beek, O. Edfors, M. Sandell, S.K. Wilson, and P.O. Borjesson, "On channel estimation in OFDM systems," in *Proceedings of the IEEE Vehicular Technology Conference*, Chicago, IL, USA, July 25–28 1995, vol. 2, pp. 815–819.

[18] "Mentor Graphics (accessed on 17 September 2008) Algorithmic C Data Types. URL: http://www.mentor.com/products/esl/high_level_synthesis/ac_datatypes.cfm," .

[19] "Xilinx (accessed on 28 September 2008) Virtex-4 Product Table. URL: http://www.xilinx.com/publications/matrix/matrix_v4.pdf," .