COMPLEX-VALUED QR DECOMPOSITION IMPLEMENTATION FOR MIMO RECEIVERS

Perttu Salmela, Adrian Burian, Harri Sorokin, and Jarmo Takala*

Department of Computer Systems
Tampere University of Technology
Tampere, Finland
{perttu.salmela, harri.sorokin, jarmo.takala}@tut.fi

*Nokia Research Center
Tampere, Finland
adrian.burian@nokia.com

ABSTRACT

Multiple input multiple output (MIMO) transmission is an emerging technique targeted at 3G long term evolution (LTE) systems. One vital baseband function in MIMO receivers is QR decomposition of the channel matrix. In this paper, a processor based complex-valued QR decomposition is presented. The processor is enhanced with complex arithmetic and inverse square root function units. The proposed processor fits well with the real-time requirements of the MIMO receiver. The computing power is tailored for typical MIMO systems. Due to the generality of the applied computing resources it can also be used for other tasks. Also, the presented principles can be applied on any customizable processor architectures to accelerate QR decomposition.

Index Terms— QR decomposition, MIMO, TTA, processor

1. INTRODUCTION

Extremely high data rates are expected in 3G long term evolution (LTE) [1]. Multiple input multiple output (MIMO) transmission is one technique enabling high data rates in 3G LTE. Advanced transmission techniques require sophisticated algorithms in the receiver and efficient implementation of such algorithms is crucial. The QR decomposition is one of the many baseband functions of the MIMO receiver. It transforms a complex-valued channel matrix $\boldsymbol{H}$ to a decomposition of an orthogonal $\boldsymbol{Q}$ and an upper triangular $\boldsymbol{R}$ matrices. The matrices are required by a list sphere detection (LSD) algorithm, which detects the received, complex-valued symbols.

There are several ways to obtain QR decomposition. It can be computed, e.g, using Householder transformations, Givens rotations, or Gram-Schmidt process. However, quite often some variations of the basic algorithms are used. The inherent regularity of matrix operations can be utilized with systolic structures [2]. Elementary operations can be alleviated with a CORDIC algorithm [3] which lends itself to low-complexity hardware realization. Such an approach was followed, e.g., in [4, 5, 6]. Another way to alleviate hardware complexity is to carry out computations in logarithmic domain. This practice is used in [7].

The MIMO receiver requires relatively small matrix size and low processing speed for QR decomposition. Therefore, extensively parallel solutions like systolic array processors in [2, 6] or processor arrays with reduced dimensions [4] are not justified to be used for such systems. Furthermore, dedicated hardware [8] or structures without bit accurate multipliers [7] have limited applicability for other tasks. They have to idle for a long periods, whereas more flexible processors could be used for other tasks between successive QR decompositions, which results in an efficient utilization of available resources.

In this paper, an application-specific instruction-set processor (ASIP) is customized for complex-valued QR decomposition. The matrix is decomposed according to the modified Gram-Schmidt process [9]. The computation of $1/\sqrt{x}$ function is alleviated with very low complexity approximation, which is based on the binary representation of fixed-point numbers. With 160 MHz clock frequency the required QR decompositions of 3G LTE 4×4 MIMO receiver can be computed within the coherence time of the channel and the processor takes only 16.3 kgates.

Due to the programmability and bit-accurate multipliers the processor is applicable also for other real or complex-valued baseband tasks. It is shown that QR decomposition and the fast Fourier transform (FFT), required by orthogonal frequency division multiplexing (OFDM) demodulation, could share the same computing resources, which would result in area efficient and economical MIMO OFDM receivers. The proposed principles can be applied on any customizable processor.

2. QR DECOMPOSITION FOR MIMO TRANSMISSION

In the MIMO systems, symbol detection algorithms like LSD require triangularization of the channel matrix. Such a triangularization of complex-valued matrix can be carried out with QR decomposition.

2.1. MIMO Transmission

In principle, there are multiple transmit and receive antennas in the MIMO systems. The MIMO system transmitting a symbol $\boldsymbol{s}$ can be described with

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{n} \tag{1}$$

where $\boldsymbol{y}$ is the received symbol, $\boldsymbol{n}$ is the noise vector and $\boldsymbol{H}$ is the complex-valued channel matrix. The number of rows and columns of $\boldsymbol{H}$ equals to the number of receive and transmit antennas, respectively. Maximum-likelihood detection estimates the transmitted symbol $\boldsymbol{s}'$ by solving

$$\boldsymbol{s}' = \arg\min_{\boldsymbol{s}} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{s}\|^2 , \tag{2}$$

which gives the optimum result. However, solving (2) is intractable with large constellations and multiple antennas and, therefore, simpler algorithms must be applied.

Instead of solving (2), the QR decomposition of $\boldsymbol{H}$ is used to alleviate computations. The maximum likelihood detection can be substituted with

$$\boldsymbol{s}' = \arg\min_{\boldsymbol{s}} \|\boldsymbol{y}' - \boldsymbol{R}\boldsymbol{s}\|^2 \quad \text{where } \boldsymbol{y}' = \boldsymbol{Q}^H\boldsymbol{y} \tag{3}$$

```
for k = 1 : n
    R_{k,k} = ||H_{1:n,k}||
    Q_{1:n,k} = H_{1:n,k} / R_{k,k}
    for j = k + 1 : n
        R_{k,j} = Q^H_{1:n,k} H_{1:n,j}
        H_{1:n,j} = H_{1:n,j} − Q_{1:n,k} R_{k,j}
    end
end
```

Fig. 1. Modified Gram-Schmidt algorithm transforms $\boldsymbol{H}_{n\times n}$ to the orthogonal $\boldsymbol{Q}_{n\times n}$ and upper triangular $\boldsymbol{R}_{n\times n}$. Conjugated transpose is denoted with $(\cdot)^H$.

Approximating $\boldsymbol{s}'$ is far easier with (3), since the $\boldsymbol{R}$ is in upper triangular form. Now, the Euclidean distance in (3) can be computed by gradually increasing the dimensions of the symbol vector. Partial solutions too far away from the received symbol, can be discarded, which efficiently limits the search space.

2.2. Modified Gram-Schmidt QR Decomposition

In this paper, the modified Gram-Schmidt algorithm [9] is used for the QR decomposition. The modified Gram-Schmidt algorithm has better numerical properties than the classical Gram-Schmidt algorithm. In principle, the algorithm orthogonalizes a set of vectors. The modified Gram-Schmidt algorithm is shown in Fig. 1 for square matrix $\boldsymbol{H}_{n\times n}$. A straightforward implementation of the algorithm would require square root function for distance computations and division operation, but they are demanding to implement on hardware. The algorithm requires $2n^3$ arithmetic operations, which is also the lower limit of the number of clock cycles with sequential execution. Since the matrix size is determined by the number of antennas, it is relatively small. In this paper, a 4×4 matrix is assumed, which conforms with [10] where at maximum four transmit or receive antennas are suggested.

3. TRANSPORT TRIGGERED ARCHITECTURE PROCESSOR

The proposed QR decomposition is implemented on a transport triggered architecture (TTA) processor. However, the presented principles can also be applied on other customizable processors. The TTA is an application-specific instruction-set processor template [11] where parallel computing resources can be tailored according to the application. Basically, the TTA processor consists of computing resources which are connected via an interconnection network. The TTA processors are modular, as they are tailored by including only the necessary function units (FU). Since the processor can be tailored, the interconnection network does not have to contain all the connections, which reduces the bus load and power consumption and allows a higher clock frequency.

The organization of the proposed TTA processor is shown in Fig. 2. There are two dedicated units for complex-valued arithmetic in the processor. The applied practice of using complex numbers as the native data type accelerates computations significantly. If a real-valued emulation of complex numbers were used, elementary arithmetic operations would take several clock cycles and also loading and storing numbers would require more memory accesses. In practice, the complex numbers are presented as 32-bit words where real and imaginary parts use upper and lower 16 bits, respectively.

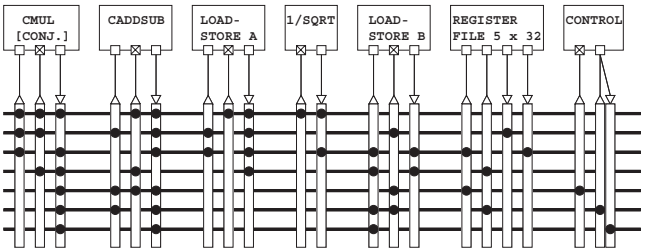

Fig. 2. Proposed TTA processor for QR decomposition. The processor has FUs for optionally conjugated complex-valued multiplication, complex-valued addition and subtraction, approximation of $1/\sqrt{x}$, and for loading or storing data. Filled circles denote connections between FUs and buses.

The complex multiplier FU of the processor supports two operations; it can compute the normal complex multiplication of two fixed-point operands and complex multiplication with conjugated multiplicand. The modified Gram-Schmidt algorithm in Fig. 1 requires conjugated multiplication for computing the vector norm and for conjugated dot product. The second complex-valued arithmetic FU supports complex addition and subtraction operations.

The modified Gram-Schmidt algorithm requires division with the (real-valued) norm $\|\cdot\|$. However, the multiplication is simpler operation in hardware than division. For this reason, the division is substituted with multiplication with an inverse value, i.e., inverse of Euclidean norm. Thus, it is inverse of the square root. With this substitution two demanding operations, division and square root, are replaced with computation of the inverse square root function, $1/\sqrt{x}$, and multiplication. Naturally, also $1/\sqrt{x}$ is very demanding function but a low complexity circuitry for approximation of $1/\sqrt{x}$ is used. The accelerated computation of $1/\sqrt{x}$ can be used also for $\sqrt{x}$ function with one multiplication, since $x\frac{1}{\sqrt{x}} = xx^{-\frac{1}{2}} = x^{\frac{1}{2}} = \sqrt{x}$.

4. LOW-COMPLEXITY APPROXIMATION OF INVERSE SQUARE ROOT FUNCTION

The proposed approximation method relies heavily on the binary representation of fixed-point numbers. In this paper, 11 fractional bits, four integer bits, and the sign bit are used for presenting real or imaginary parts of complex numbers. Instead of approximating highly non-linear $1/\sqrt{x}$, the method approximates more softly non-linear $1/\sqrt{1+u}$ after appropriate substitutions. This practice is justified by noting that x can be presented in fixed-point format as

$$x = \underbrace{0.000\ldots0}_{\alpha}1u \tag{4}$$

where α denotes the number of leading zeros. Since

$$x \times 2^{\alpha} = 1.u \quad \text{and} \quad x = 2^{-\alpha} \times 1.u\,, \tag{5}$$

the desired form

$$1.u = 2^0 + u = 1 + u \tag{6}$$

can be obtained. The same principle is used also if $x \geq 1$. In this case the direction of bitwise shifting is reversed. In other words, the α may have negative sign. With the derived substitution

$$\frac{1}{\sqrt{x}} = \frac{1}{\sqrt{2^{-\alpha}(1+u)}} = 2^{\frac{\alpha}{2}}\frac{1}{\sqrt{1+u}}\,, \tag{7}$$

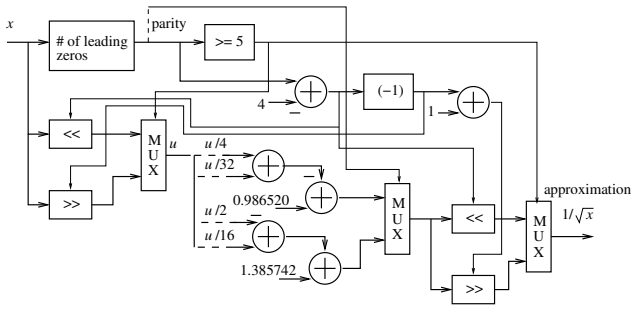

Fig. 3. Inverse square root approximation unit. Changing the sign of the number is denoted with (-1). Left and right shifts are denoted with $<<$ and $>>$, respectively, and comparison is denoted with $>=$.

which results to two cases depending on the remainder of $\alpha/2$. For even values $\alpha = 2k$

$$\frac{1}{\sqrt{x}} = 2^k \frac{1}{\sqrt{1+u}} \tag{8}$$

and for odd values $\alpha = 2k + 1$

$$\frac{1}{\sqrt{x}} = 2^k \sqrt{2} \frac{1}{\sqrt{1+u}} \,. \tag{9}$$

Hardware implementation of multiplication with 2^k is simple shifting. Since the non-linearity has been softened when compared to $1/\sqrt{x}$, first order polynomials can be used to approximate terms $1/\sqrt{1+u}$ and $\sqrt{2}/\sqrt{1+u}$. Furthermore, for the simplicity of hardware, the coefficients should be chosen in such a way that multipliers are not needed. Instead of multipliers, adders and constant shifters can be used if the softly non-linear curves are modeled with

$$\frac{1}{\sqrt{1+u}} \simeq 0.965820 - \frac{1}{4}u - \frac{1}{32}u \tag{10}$$

and

$$\sqrt{2}\frac{1}{\sqrt{1+u}} \simeq 1.385742 - \frac{1}{2}u + \frac{1}{16}u \,. \tag{11}$$

The constant terms in decimal format have been searched exhaustively and they can be presented with the applied 16-bit fixed-point numbers.

The structure of the inverse square root approximation unit is shown in Fig. 3. It consists of four shifters, constant shifters, adders, subtracters, multiplexers, and a unit for changing the sign of the number. The word lengths of computations are very short, since $u < 1$ and some intermediate values are even scaled with right shifting. The main benefit of the proposed approximation method is its low-complexity. Usually, conventional $1/\sqrt{x}$ computation methods rely on dedicated look-up tables and multipliers [12]. Such resources increase the hardware costs significantly. If high accuracy is require, it can be obtained with Newton's iterations, which can be computed with the multiplier FU of the processor.

5. RESULTS

The proposed QR decomposition processor is synthesized with Synopsys Design Compiler on 0.13 μm standard cell technology for obtaining complexity and performance estimates.

Table 1. Area of the proposed processor and execution time of 2048 QR decompositions.

Clock frequency	Area	Execution time T_{QR}
269 MHz	23.2 kgates	1.058 ms
212 MHz	17.7 kgates	1.343 ms
160 MHz	16.3 kgates	1.779 ms

5.1. Real-Time Requirements

The *coherence time* indicates how long the channel impulse response is essentially invariant. It can be expressed as

$$t_{coh} = c/(v f_c) \tag{12}$$

where c is the speed of light, v speed of receiver, and f_c is the carrier frequency. With $f_c = 2.4$ GHz, $v = 250$ km/h, and $c = 3 \times 10^8$ m/s the coherence time $t_{coh} = 1.8$ ms. The OFDM modulation will be used in 3G LTE MIMO systems. The spectrum of the signal consists of several subcarriers in OFDM signal and there are at maximum 2048 subcarriers in 3G LTE signal [1]. The channel matrix $\boldsymbol{R}$ must be computed for all the subcarriers within $t_{coh} = 1.8$ ms.

5.2. Complexity and Performance

The proposed QR decomposition of complex-valued 4×4 matrix takes 139 clock cycles. So, the decompositions for all the 2048 subcarriers takes

$$T_{QR} = (2048 \times 139)/f \tag{13}$$

where f is the clock frequency. The execution times for several clock frequencies are tabulated in the Table 1. The execution time should be compared to the available time frame of 1.8 ms. Naturally, some computing time should be reserved for higher level control flow and polling for a new input matrix $\boldsymbol{H}$ in real application.

The area of the proposed processor with different clock frequencies is given in the Table 1. The area is expressed in terms of gate equivalents. With lower clock frequency the synthesis process can instantiate slower but simpler components, which results in savings in the total gate count.

5.3. Resource Sharing with FFT

With high clock frequency, the idle time could be used for FFT, in principle. The FFT is used to demodulate OFDM signal and it also requires complex-valued arithmetic. Sharing the same resources among the two functions would be more area efficient than having dedicated processors. The proposed QR decomposition processor is not able to compute FFT. Therefore, a hybrid of QR decomposition processor and TTA FFT processor [13] should be used.

The combined processor should posses the features and resources required by both the QR decomposition and FFT functions. The resource combination can be created with the aid of multiset unions, i.e., the maximum multiplicity of resources of both processors determines how many resources of respective type are instantiated [14]. The TTA allows to include or exclude computing resources and to modify the interconnection network freely. Thus, such a combined processor could be generated with modest effort.

The FFT must be computed for each received OFDM symbol for each antenna, so it is assumed that there will be four processors. There are at maximum seven OFDM symbols in 0.5 ms subframe

[1]. So, the symbol time $T_S = (0.5 \text{ ms})/7 = 71.4\ \mu$s. The TTA processor presented in [13] is capable of performing 2048-point FFT in 12332 clock cycles. The total number of required clock cycles in 1.8 ms time frame is

$$C = \frac{1.8 \text{ ms}}{T_S} \times 12332 \text{ cycles} + (2048 \times 139 \text{ cycles})/4 \qquad (14)$$

where the computation load of QR decompositions is distributed among four processors, since there are four antennas and four FFT based OFDM demodulators. The required clock frequency is

$$f_{required} = C/1.8 \text{ ms} = 212 \text{ MHz}. \qquad (15)$$

Again, some computing time should be saved for higher level control flow of the main program.

5.4. Discussion

QR decomposition is typically implemented with a systolic array and computations are alleviated with CORDIC algorithm. In [8] a complex-valued matrix inversion based on QR decomposition is presented. The method uses squared Givens rotations. Instead of traditionally triangular array of processing elements (PE), the PEs are mapped to a linear array structure. Inverting a complex-valued 4×4 matrix takes 175 cycles. As a drawback, such an array processor is not flexibly programmable like ASIPs.

In [4] a floating-point real-valued programmable ASIP for QR and singular-value decomposition is presented. The ASIP contains CORDIC module and ASIPs can be structured as an array for high throughput. The ASIPs are programmable, but the structure resembles array processors as the PEs are substituted with the presented ASIPs.

Computations in $\log_2$ domain are applied in [7]. The parallel architecture for 4×4 matrix inversion with the aid of QR decomposition takes 72 kgates and achieves a latency of 0.24 μs. As a drawback, such an architecture is too fast for the requirements of 3G LTE MIMO receiver. The architecture has to idle most of the time and the computing resources cannot be used for other tasks, since the architecture is not programmable. In addition, computing in $\log_2$ domain may not be suitable for other functions.

A structure with fully programmable Nios processor and CORDIC accelerator in FPGA is presented in [5]. The CORDIC elements are used for QR decomposition and the following back substitution for solving a set of equations is computed on Nios processor. As a drawback, the accelerating CORDIC elements are not tightly connected to the data path of the processor. Instead, the CORDIC accelerator and Nios processor communicate via memory.

In this paper, the ASIP implementation of the QR decomposition overcomes the aforementioned drawbacks. High computing capacity is not targeted due to the small matrix size. Instead, relatively low complexity, flexibility, and programmability are the main objectives. The flexibility is a consequence of the programmability and bit-accurate multiplication, addition, and subtraction FUs, which can be used also for other computations. The simple approximation method of the $1/\sqrt{x}$ contributes to the low-complexity. The programmability is not limited to minor changes of the algorithm, since all the data transports on internal buses are defined individually in the instruction word.

6. CONCLUSIONS

A processor based complex-valued QR decomposition was developed in this paper. The modified Gram-Schmidt algorithm was applied. The algorithm was justified with the versatility of the processor with complex-valued multipliers. The results showed that the throughput of the proposed implementation fits well with typical requirements of the MIMO receiver. The computing resources could be shared by mapping the QR decomposition and the FFT to the same combined processor and these essential functions of the MIMO OFDM receiver could be processed with practical clock frequency.

7. REFERENCES

[1] R. Bachl, P. Gunreben, S. Das, and S. Tatesh, "The long term evolution towards a new 3GPP* air interface standard," *Bell Labs Technical Journal*, vol. 11, no. 4, pp. 25–51, Mar. 2007.

[2] S. Y. Kung, *VLSI Array Processors*. Upper Saddle River, NJ, USA: Prentice-Hall, 1987.

[3] R. Andraka, "A survey of cordic algorithms for fpga based computers," in *Sixth ACM/SIGDA Int. Symp. on Field-Programmable Gate Arrays*, Monterey, CA, USA, Feb. 1998, pp. 191–200.

[4] Z. Liu, K. Dickson, and J. V. McCanny, "Application-specific instruction set processor for SoC implementation of modern signal processing algorithms," *IEEE Tran. on Circuits and Systems*, vol. 52, no. 4, pp. 755–765, Apr. 2006.

[5] "Implementation of CORDIC-based QRD-RLS algorithm on Altera Stratix FPGA with embedded Nios soft processor technology," Altera Corporation, San Jose, CA, USA, white paper, Mar. 2004.

[6] A. Maltsev, V. Pestretsov, R. Maslennikov, and A. Khoryaev, "Triangular systolic array with reduced latency for QR-decomposition of complex matrices," in *IEEE Inter. Symp. on Circuits and Systems*, Kos, Greece, May 2006, pp. 385–388.

[7] C. K. Singh, S. H. Prasad, and P. T. Balsara, "VLSI architecture for matrix inversion using modified Gram-Schmidt based QR decomposition," in *Int. Conf. on VLSI Design*, Bangalore, India, Jan. 2007, pp. 836–841.

[8] F. Edman and V. Öwall, "A scalable pipelined complex valued matrix inversion architecture," in *IEEE Int. Symp. on Circuits and Systems*, vol. 5, Kobe, Japan, May 2005, pp. 4489–4492.

[9] G. H. Golub, *Matrix Computations*. Baltimore, MD, USA: John Hopkins University Press, 1989.

[10] "3GPP TR 25.876 multiple input multiple output in UTRA," 3rd Generation Partnership Project, Tech. Rep., Oct. 2005.

[11] H. Corporaal, "Design of transport triggered architectures," in *4th Great Lakes Symp. Design Autom. of High Perf. VLSI Syst.*, Notre Dame, IN, USA, Mar. 1994, pp. 130–135.

[12] W. F. Wong and E. Goto, "Fast hardware-based algorithms for elementary function computations using rectangular multipliers," *IEEE Trans. On Computers*, vol. 43, no. 3, pp. 278–294, Mar. 1994.

[13] T. Pitkänen, R. Mäkinen, J. Heikkinen, T. Partanen, and J. Takala, "Low-power, high-performance TTA processor for 1024-point fast fourier transform," in *Embed. Comp. Systs.: Architectures, Modelling, and Simulation*, vol. LNCS 4017. Berlin, Germany: Springer-Verlag, 2006, pp. 227–236.

[14] P. Salmela, C.-C. Shen, S. S. Bhattacharyya, and J. Takala, "Synthesis of DSP architectures using libraries of coarse-grain configurations," in *IEEE Workshop on Sign. Proc. Systs.*, Shanghai, China, Oct. 2007, pp. 475–480.