

Title	A Systolic Array RLS Processor
Author(s)	Asai, T.; Matsumoto, T.
Citation	IEEE 51st Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo., 3: 2247-2251
Issue Date	2000-05
Type	Conference Paper
Text version	publisher
URL	http://hdl.handle.net/10119/4822
Rights	Copyright (c)2000 IEEE. Reprinted from IEEE 51st Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of JAIST's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org . By choosing to view this document, you agree to all provisions of the copyright laws protecting it.
Description	

A Systolic Array RLS Processor

Takahiro Asai, Member, IEEE, and Tadashi Matsumoto, Senior Member, IEEE

Wireless Laboratories, NTT Mobile Communications Network Inc.,

3-5 Hikari-no-oka, Yokosuka-shi, Kanagawa-ken 239-8536, Japan

+81-468-40-3556, FAX +81-468-40-3790

asai@mlab.yrp.nttdocomo.co.jp

Abstract This paper describes the outline of the systolic array recursive least-squares (RLS) processor that we developed primarily with the aim of broadband mobile communication applications. To perform the RLS algorithm effectively, this processor uses an orthogonal triangularization technique known in matrix algebra as QR decomposition for parallel pipelined processing. The processor board is comprised of 19 application-specific integrated circuit chips, each having approximately one Million gates. 32-bit fixed point signal processing takes place in the processor, with which one cycle of internal cell signal processing requires about 500 nsec, and boundary cell signal processing about 80 nsec. The processor board can estimate up to 10 parameters and takes approximate 35 μ s to estimate 10 parameters by using 41 known symbols. To evaluate this processor, we conduct minimum mean-squared error adaptive array in-lab experiments using a complex baseband fading/array response simulator. In terms of parameter estimation accuracy, the processor is found to produce virtually the same results as a conventional software engine using floating-point operations.

I. Introduction

Adaptive signal processing, a key part of adaptive equalizers, interference cancellers, and adaptive array antennas, will play an important role in future broadband wireless communications with signal transmission rates of several tens of Mbit/s. A signal processor that can estimate the parameters related to the communication channels on a real-time basis is indispensable in such applications. For adaptive parameter estimation, the recursive least-squares (RLS) algorithm achieves much faster convergence than the least-mean-square algorithm, however, its complexity increases in proportion to the square of the number of parameters to be estimated. To overcome this problem, several pipelining techniques for hardware implementation of the RLS algorithm, which are commonly referred to as systolic array techniques, have been proposed [1]-[5]. A systolic array processor

consists of cells of several kinds that are arranged regularly; adjacent cells are connected to each other. Systolic array processors have many desirable properties such as regularity and local interconnections that suit VLSI implementation. Furthermore, RLS signal processing on a systolic array processor is numerically stable under the condition of limited arithmetic precision [4].

This paper describes a systolic array RLS processor that we developed using application specific integrated circuit (ASIC) chips. Results of an adaptive array in-lab experiment conducted to evaluate the developed RLS processor by using a complex fading array response simulator are described. Its performance is compared with that of a computer program as well as with theoretical curves.

II. Configuration of the developed systolic array RLS processor

A. Summary of systolic array RLS algorithm

The square-root-free algorithm [1] is used in the systolic array RLS processor prototype. A block diagram of this algorithm is illustrated in Fig. 1, where the number of parameters to be estimated is three and β^2 ($0 < \beta \leq 1.0$) is the forgetting factor. For parallel pipelined processing, the systolic array RLS processor uses the orthogonal triangularization technique in matrix algebra that is sometimes referred to as QR decomposition. There are three types of processing cells that are used in this architecture. The circles and squares represent the boundary and internal cells, respectively. The final cell is a simple two-input multiplier. The dots along the diagonal of the array represent storage elements. After simple calculation at each cell, some of the resulting data are stored in the cells while the others are passed to adjacent cells. By repeating this procedure, data is passed from cell to cell across the array. The final cell produces an output equal to estimation error e .

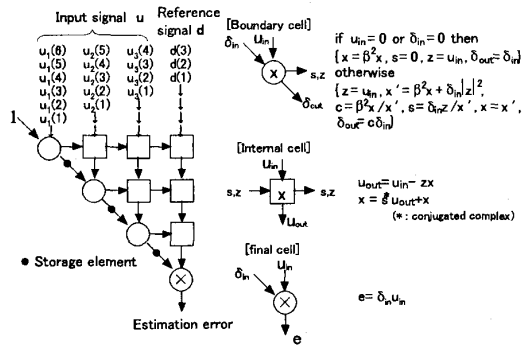


Fig. 1 System block diagram of systolic array RLS processor (in this example the number of parameters to be estimated is three)

To extract the weight vector, serial weight flushing [1] is used in the systolic array RLS processor. Fig. 2 shows an input data configuration for extracting the weight vector. Let $\mathbf{u}(n)$ denote the input vector and $d(n)$ the reference signal, both at time n . The corresponding estimation error to be obtained as the systolic array processor output is

$$e(n) = d(n) - \mathbf{w}^H(n)\mathbf{u}(n), \quad (1)$$

where $\mathbf{w}(n)$ is the weight vector at time n . Assuming

$$\mathbf{u}^H(n) = [0 \dots 0 \ 1 \ 0 \dots 0], d(n) = 0, \quad (2)$$

↑
i-th element

the estimation error can be expressed as

$$e(n) = -w_i^*(n). \quad (3)$$

Therefore, as shown in Fig. 2, the i -th weight can be obtained as the output of the systolic array processor in response to the input of Eq. (2). To extract the entire weight vector, we simply halt the updating of all stored values, and input a data matrix that consists of a unit diagonal matrix.

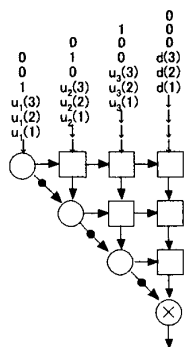


Fig. 2 Data configuration for serial weight flushing (the number of parameters to be estimated is three)

B. Prototyped board

RLS signal processing on a systolic array processor is known to be numerically stable under the condition of

limited arithmetic precision. In addition, for faster processing speed, the systolic array RLS processor uses fixed-point signal processing rather than floating-point processing. The bit allocations for the integer and fractional parts, required to achieve reasonable estimation accuracy, were determined through an exhaustive series of computer simulations: uniformly distributed random data having 40 dB dynamic range were input to a systolic array structure on a computer, which emulated fixed and floating point signal processing. The estimation accuracy with various bit allocation patterns for fixed-point processing was then compared to that with floating point processing. We found that the estimation error, defined as the difference in parameter estimates yielded by floating and 32-bit fixed-point signal processing, can be made smaller than 5%.

Fig. 3 shows a picture of the prototype systolic array RLS processor board, which is approximately 36cm*40cm. The forgetting factor, the number of parameters to be estimated and the number of unique word sequence can be set from a PC connected to the board. This systolic array RLS processor board can estimate up to 10 parameters. It is comprised of 19 ASIC chips, each having approximately one Million gates. One cycle of the internal cell signal processing takes about 500 nsec, while that of the boundary cell signal processing takes about 80 nsec. It takes approximate 35μs to estimate 10 parameters by using 41 known symbols.

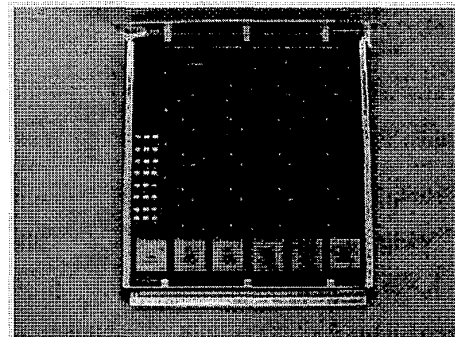


Fig. 3 Systolic array RLS processor board

III. Experiments

A minimum mean-squared error (MMSE) adaptive array antenna experiment was conducted using the prototype systolic array RLS processor board. We developed a complex baseband fading/array response simulator for performance evaluations of baseband sections of time-space equalizers [6]: the prototype board was connected to the simulator, and signal transmission experiments were conducted, all in the complex baseband domain.

A. Configuration of the complex baseband fading/array response simulator

The complex baseband fading/array response simulator simulates temporal and spatial radio wave propagation scenarios experienced by broadband mobile communications in real time. Fig. 4 shows a block diagram of the complex baseband fading/array response simulator. One desired and L interference users share the same channel. Signals transmitted from the L+1 mobile users are received by an N-element antenna array. Each path component is multiplied by its corresponding fading complex envelope, and then attenuated by multiplications by real constants. The fading path components are received by an N-element antenna array. The phase rotation on each of the N antenna elements depends on the array geometry and the path's direction of arrival (DOA). The array geometry and DOA information on each path is manually input to the system control PC. For each of the path components, the PC calculates the phase rotations on the N antenna elements, and the N path components received by the N elements are multiplied by the calculated N complex constants corresponding to their phase rotations element-by-element. The phase-rotated path components are then combined together, added to complex white Gaussian noise samples, and filtered corresponding to the receiver filters assumed. N statistically independent two-dimensional random numbers uniformly distributed over [0, 1] are generated, and converted into N complex white Gaussian noise samples by using a look-up table following the Box-Muller method. The N received composite signal samples received by the N antenna elements are then brought to the systolic array RLS processor board. Table 1 summarizes the hardware specifications of the simulator. 24-bit fixed point signal processing takes place: in-phase and quadrature components of signals are expressed in using a 24-bit data format. This ensures 16-bit accuracy at the output of the simulator, even in the presence of round off due to the fixed-point signal processing. The clock and frame timing are recovered perfectly at the receiver.

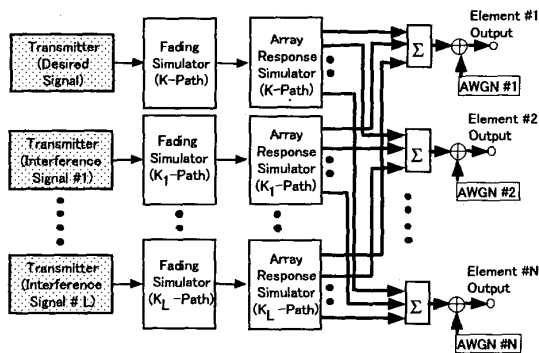


Fig. 4 Block diagram of the complex baseband fading/array response simulator

Table 1 Major specs of the complex baseband fading/array response simulator

Items	Specifications
Signal Representation	Complex Baseband Domain (I/Q Vector Channel)
Data Format	24-bit Fixed Point/Parallel Data
Sampling Speed	24 Msamples/sec
Number of Users	4 Max
Number of Paths	4 Max
Delay Time	5.2 msec Max (42nsec/step)
Doppler Frequency	2000 Hz Max
Number of Array Elements	8
Array Geometry	Linear and Circular

B. Real-time experiment system test bed

12 Msymbol/sec quaternary phase-shift keying (QPSK) signal bursts were passed through the simulator. The transmitted data stream was framed. Each frame included a 31 symbol unique word and 384 symbol information sequences. It was assumed that an N-element (N=1,2,4,8) linear array antenna with a minimum element spacing of half the wavelength was used. The systolic array RLS processor uses the fading/array response simulator output corresponding to unique word sequence, and calculates antenna weights (Fig. 5). The forgetting factor was set at 0.99.

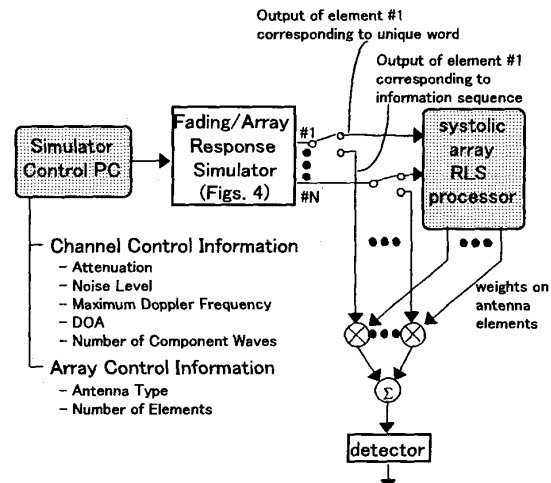


Fig. 5 Block diagram of adaptive array antenna system test bed

C. Results

Bit error rate (BER) performance in the additive white Gaussian noise channel (AWGN) is shown in Fig. 6. Fading is not present. Performance curves obtained through computer simulation and by theoretical analysis are also plotted in the figure. For computer simulations, conditions are the same as those used in

in-lab experiment except that floating point signal processing was performed for the systolic array RLS algorithm. It is found that the experimental performance curves agree well with those obtained by computer simulations. The 1-element BER curves for both the in-lab experiment and computer simulation agree well with the theoretical curves.

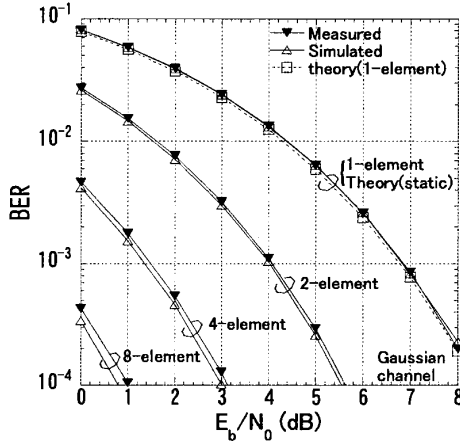


Fig. 6 BER performance on a Gaussian channel

BER performances in the presence of fading are shown in Fig. 7. 1-path Rayleigh fading was assumed. In-lab experimental results are almost the same as those of the computer simulations. The 1-element BER curves of both the in-lab experiment and computer simulation agree well with the theoretical curve. The fading variation is slow enough to eliminate the BER plateaus generally observed in the high E_b/N_0 range when the fading variation is too fast for the RLS algorithm.

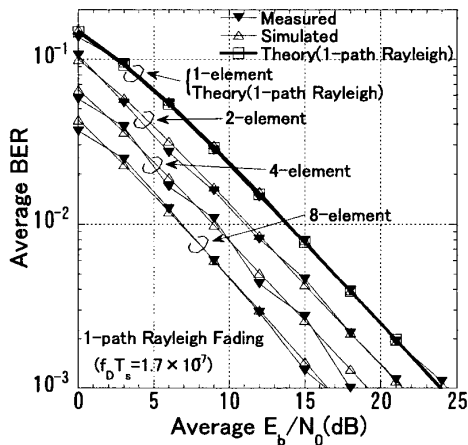


Fig. 7 BER performance on a 1-path Rayleigh fading channel (frequency-nonspecific)

BER performances with $L+1=4$ were evaluated in a Rayleigh fading channel. The desired signal's DOA was set at 0° , and the three interference components' DOAs were set at 10° , 30° and 40° . Each of the four

signals suffers from frequency-flat Rayleigh fading. Fig. 8 shows the results of the experiments in this environment. No difference in performance curves is observed between the experiment and simulation results. Since there are three interferers having the same signal strength as the desired signal, they can be suppressed if $N \geq 4$. This can be observed in Fig. 8. Spatial response obtained as results of the in-lab experiment is shown in Fig. 9 with the number of antenna elements as a parameter. $E_b/N_0=10\text{dB}$ was assumed. It is found that the interference signals are effectively suppressed with $N > 4$.

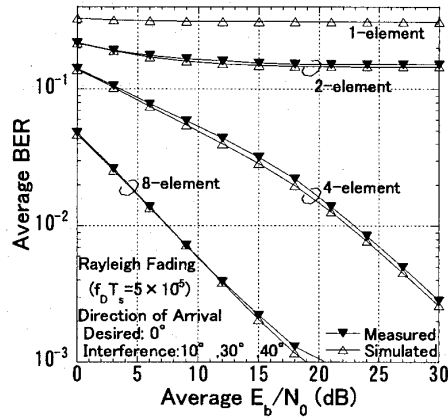


Fig. 8 BER performance on a Rayleigh fading channel (one desired signal, three interference signal)

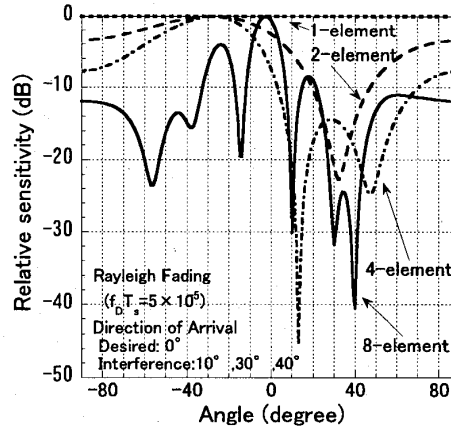


Fig. 9 In-lab experiment results on the adapted spatial response on a Rayleigh fading channel (one desired signal, three interference signal, $E_b/N_0=10\text{dB}$)

IV. Conclusions

This paper has outlined the systolic array RLS processor we developed primarily for broadband mobile communications applications. It is comprised of 19 ASIC chips, each having approximately one million gates. The processor uses 32-bit fixed-point signal

processing. The internal cell signal processing cycle is about 500 nsec, and boundary cell processing takes about 80 nsec. It takes approximate 35 μ s to estimate 10 parameters by using 41 known symbols. To evaluate the processor's performance, we conducted adaptive array experiments by using a complex baseband fading/array response simulator. The experimental results were then compared with those from a computer program using the same conditions except that the program used floating point processing. We found that the in-lab experimental results agreed well with those of the computer simulation results under various conditions. Furthermore, it was found that no difference in BER performance curves is observed between theoretical and in-lab experimental curves in both AWGN and 1-path Rayleigh fading channels. The adapted spatial response obtained by the in-lab experiments showed that the interference signals are well suppressed with an N-element antenna array (N>4) if there are one desired and three interference signals, each of which is sent over an independent one-path Rayleigh fading channel. A major conclusion of these experiments is that the systolic array RLS processor can well handle 12Msymbol/s QPSK signal bursts. It is verified that this systolic array RLS processor can be used for developments of space- and time-domain equalizers in broadband mobile multimedia communications.

Acknowledgements

The authors wish to thank Dr. N. Nakajima, senior vice president of NTT Mobile Communications Network, Inc., for his encouragement during this research.

References

- [1] S. Haykin, J. Litva, and T. J. Shepherd, *Radar Array Processing*, Springer-Verlag, 1993.
- [2] S. Haykin, *ADAPTIVE FILTER THEORY*, Prentice-Hall, 1996.
- [3] J. McCanny, J. McWhirter, and E. Swartzlander Jr., *Systolic Array Processors*, Prentice-Hall, 1989.
- [4] H. Leung and S. Haykin : "Stability of Recursive QRD-LS Algorithms Using Finite-Precision Systolic Array Implementation," *IEEE Trans. ASSP*, vol. 37, no. 5, pp.760-763, 1989
- [5] Christopher R. Ward, Philip J. Hargrave, and John G. McWhirter: "A Novel Algorithm and Architecture for Adaptive Digital Beamforming," *IEEE Trans. AP*, vol. 34, no. 3, pp. 338-346, 1986
- [6] S. Tsukamoto, T. Saso, T. Sakaki, H. Yoshino, and T. Matsumoto: "A Complex Baseband Fading/Array Response Simulator", submitted to *IEEE Trans. VT*