

Title	A Systolic Array RLS Processor
Author(s)	ASAI, Takahiro; MATSUMOTO, Tadashi
Citation	IEICE Transactions on Communications, E84-B(5): 1356-1361
Issue Date	2001-05-01
Type	Journal Article
Text version	publisher
URL	http://hdl.handle.net/10119/4676
Rights	Copyright (C)2001 IEICE. T. Asai and T. Matsumoto, IEICE Transactions on Communications, E84-B(5), 2001, 1356-1361. http://www.ieice.org/jpn/trans_online/
Description	

PAPER

A Systolic Array RLS Processor

Takahiro ASAI[†] and Tadashi MATSUMOTO[†], *Regular Members*

SUMMARY This paper presents the outline of the systolic array recursive least-squares (RLS) processor prototyped primarily with the aim of broadband mobile communication applications. To execute 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 comprises 19 application-specific integrated circuit chips, each with approximately one million gates. Thirty-two bit fixed-point signal processing takes place in the processor, with which one cycle of internal cell signal processing requires approximately 500 nsec, and boundary cell signal processing requires approximately 80 nsec. The processor board can estimate up to 10 parameters. It takes approximately 35 μ s to estimate 10 parameters using 41 known symbols. To evaluate signal processing performance of the prototyped systolic array processor board, processing time required to estimate a certain number of parameters using the prototyped board was compared with using a digital signal processing (DSP) board. The DSP board performed a standard form of the RLS algorithm. Additionally, we conducted 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.

key words: *RLS algorithm, channel estimation, QR decomposition, parallel pipelined processing, ASIC*

1. 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 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 have been proposed [1]–[5] for hardware implementation of the RLS algorithm, which are commonly referred to as systolic array techniques. A systolic array processor comprises cells of several kinds that are arranged regularly; adjacent cells are con-

nected to each other. Systolic array processors have many desirable properties such as uniformity 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]. Implementation issues of systolic array RLS processor using a DSP chip are described in [1]. Hardware implementation of a systolic array processor using commercially available processors is reported in [6]. However, their specifications are described only in part in [1] and [6]. This paper describes outline of the systolic array processor prototyping using application specific integrated circuit (ASIC) chips. The prototyped processor board performs fixed-point operations for faster processing whereas Refs. [1] and [6] use floating-point operations. Processing time required to estimate a certain number of parameters using the prototyped board is compared with using a DSP board that performs a standard form of the RLS algorithm. Additionally, results of an adaptive array in-lab experiment conducted to evaluate the prototyped systolic array RLS processor by using a complex baseband fading array response simulator are described. The processor performance from the experiments is compared with that of a computer simulation as well as with theoretical curves.

2. Configuration of the Prototyped Systolic Array RLS Processor

2.1 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

Manuscript received May 2, 2000.

Manuscript revised October 23, 2000.

[†]The authors are with Wireless Laboratories, NTT DoCoMo, Inc., Yokosuka-shi, 239-8536 Japan.

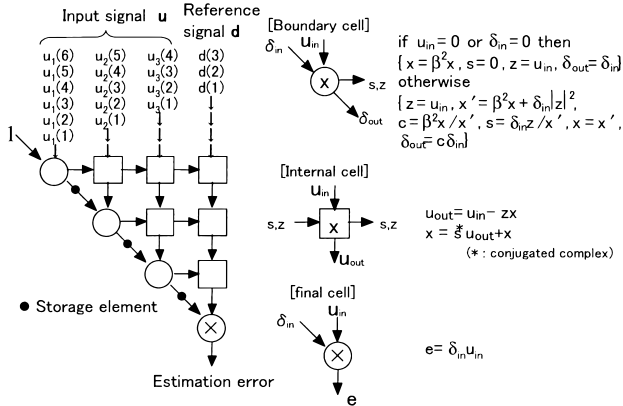


Fig. 1 System block diagram of systolic array RLS processor. (in this example the number of parameters to be estimated is three)

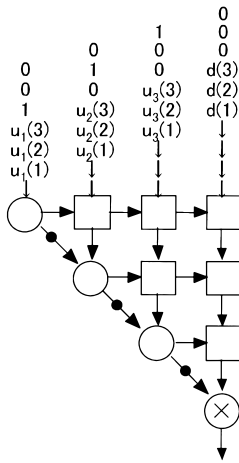


Fig. 2 Data configuration for serial weight flushing. (the number of parameters to be estimated is three)

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 . To extract the weight vector, serial weight flushing [1] is used in the systolic array RLS processor. Figure 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

$$\mathbf{u}^H(n) = [0 \dots 010 \dots 0], \quad d(n) = 0, \quad (2)$$

$\mathbf{u}^H(n)$ consists of a string of zeros, except for the i -th element which is set equal to 1, 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

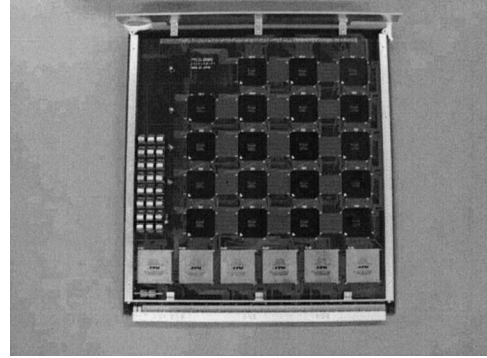


Fig. 3 Systolic array RLS processor board.

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 stop updating all the stored values, and input a data matrix that consists of a unit diagonal matrix.

2.2 Prototyped Board

RLS signal processing on a systolic array processor is known to be numerically stable under the condition of limited arithmetic precision [1]. In addition, for faster processing, 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. Estimation accuracy should depend on the number of the parameters to be estimated. This means that required total data width for reasonable estimation accuracy should depend on the number of the parameters. A major limiting factor for hardware implementation of signal processor boards is, in general, that the numbers of pins and wires required for interconnection between ASIC chips are limited to a certain level. From this viewpoint, the maximum number of parameters to be estimated was first determined in the prototyping, taking into account the hardware constraints.

In computer simulations, uniformly distributed random data with a 40-dB dynamic range was input to the systolic array structure. The estimation accuracy with various bit allocation patterns for fixed-point signal processing was compared to that with floating-point signal processing. We then decided to use 32-bit fixed-point signal processing. More details about the process for determining the data width can be seen in [7]. Figure 3 shows a picture of the prototyped systolic array RLS processor board, which is approximately 36×40 cm. The forgetting factor, the number of parameters to be estimated, and the number of unique word sequences are set from a PC connected to the board. This systolic array RLS processor board can estimate

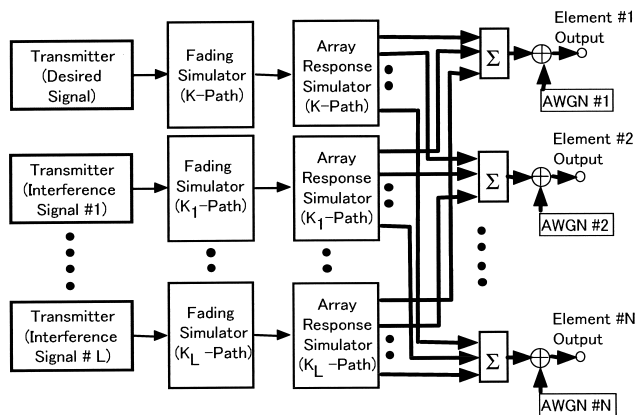


Fig. 4 Block diagram of the complex baseband fading/array response simulator.

up to 10 parameters. It comprises 19 ASIC chips, each having approximately one million gates. One cycle of the internal cell signal processing takes approximately 500 nsec, while that of the boundary cell signal processing takes approximately 80 nsec. It takes approximately 35 μ s to estimate 10 parameters using 41 known symbols.

Besides the systolic array prototyping, we also developed an RLS signal processing software on an Analog Devices ADSP21062 floating-point DSP chip for performance comparison. The DSP-based implementation uses a standard form of the RLS algorithm. A conclusion of the comparison is that the systolic array-based parameter estimator is roughly one hundred times as fast as the DSP-based estimator.

3. Experiments

A minimum mean-squared error (MMSE) adaptive array antenna experiment was conducted using the prototyped systolic array RLS processor board. The prototyped board was connected to a complex baseband fading/array response simulator, which we developed to evaluate performances of baseband sections of S/T-equalizers beforehand [8]. Signal transmission experiments were then conducted, all in the complex baseband domain.

3.1 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. Figure 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

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

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. 42 nsec/step
Doppler Frequency	2000 Hz Max.
Number of Array Elements	8
Array Geometry	Linear and Circular

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 into 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 assumed receiver filters. 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. Twenty-four bit fixed-point signal processing takes place: in-phase and quadrature components of signals are expressed in 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.

3.2 Real-Time Experiment System Test Bed

Twelve 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 the unique word sequence, and calculates antenna

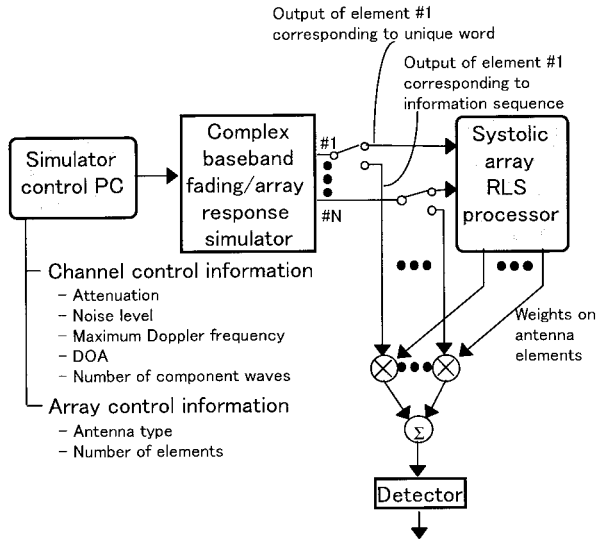


Fig. 5 Block diagram of adaptive array antenna system test bed.

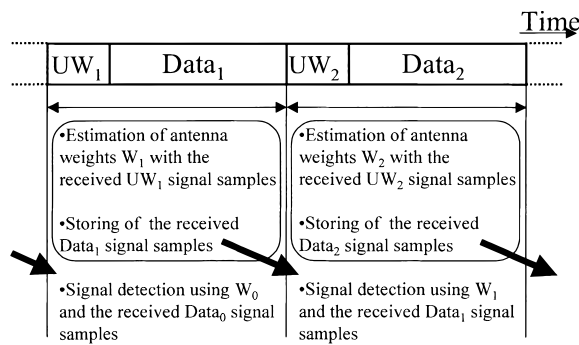


Fig. 6 Signal detection time chart.

weights (Fig. 5). The forgetting factor was set to 0.99. As shown in Fig. 6, the systolic array RLS processor estimates antenna weights using received signal samples corresponding to the unique word (UW) sequence while storing received signal samples corresponding to the data sequence. During the consecutive frame, the stored received signal samples are combined using the calculated antenna weights, and signal detection is performed. As a result, a detection delay of one frame duration is incurred.

3.3 Results

Bit error rate (BER) performance in the additive white Gaussian noise channel (AWGN) is shown in Fig. 7. Fading is not present. Performance curves obtained by computer simulation and by theoretical analysis are also plotted in the figure. For the computer simulations, conditions are the same as those used in the in-lab experiments except that floating-point signal processing was performed for the systolic array RLS algorithm. It was found that the experimental performance curves

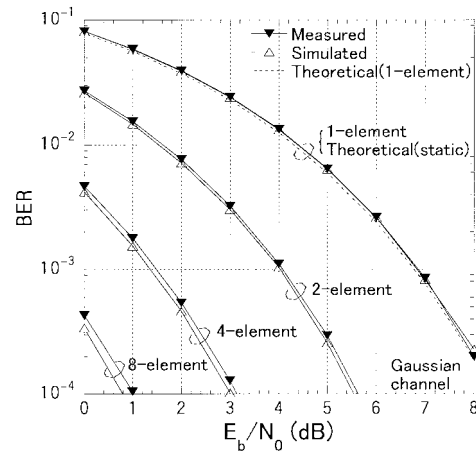


Fig. 7 BER performance on a Gaussian channel.

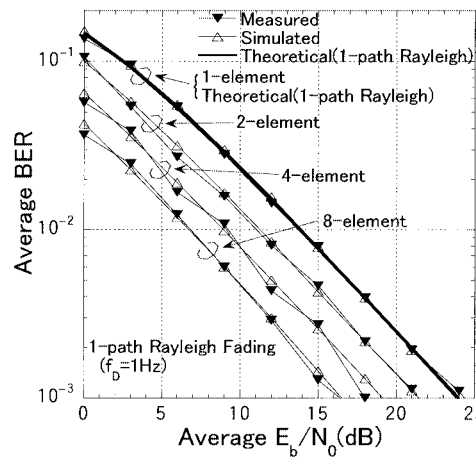


Fig. 8 BER performance on a 1-path Rayleigh fading channel. (frequency-nonselctive)

agree well with those obtained by computer simulation. The 1-element BER curves for both the in-lab experiment and computer simulation agree well with the theoretical curves. BER performance levels in the presence of fading are shown in Fig. 8. One-path Rayleigh fading was assumed. The 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 sufficiently slow 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. The BER performance with $L + 1 = 4$ was evaluated in a Rayleigh fading channel. The DOA of the desired signal was set to 0° , and the DOAs of the three interference components were set to 10° , 30° , and 40° . Each of the four signals suffers from frequency-flat Rayleigh fading. Figure 9 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

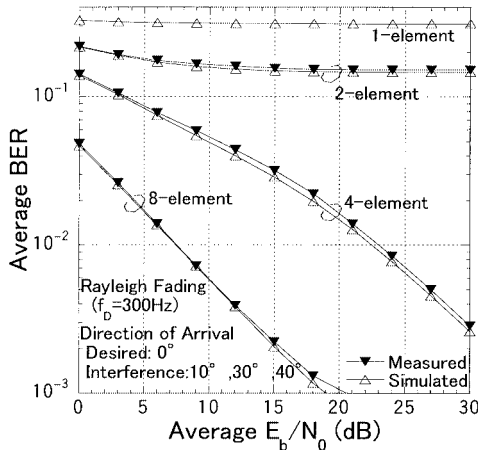


Fig. 9 BER performance on a Rayleigh fading channel. (one desired signal, three interference signal)

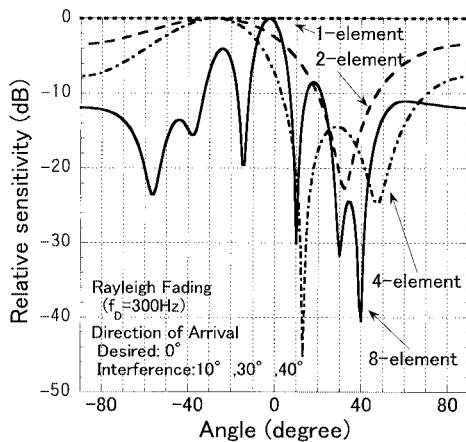


Fig. 10 In-lab experimental results on the adapted spatial response on a Rayleigh fading channel. (one desired signal, three interference signal, $E_b/N_0 = 10$ dB)

three interferers with the same signal strength as the desired signal, they can be suppressed if $N \geq 4$. This can be observed in Fig. 9. Spatial responses obtained as the results of the in-lab experiment are shown in Fig. 10 with the number of antenna elements as a parameter and where $E_b/N_0 = 10$ dB was assumed. It was found that the interference signals are effectively suppressed with $N \geq 4$.

4. Conclusions

This paper outlined the systolic array RLS processor we prototyped primarily for broadband mobile communications applications. The processor comprises 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 approximately 500 nsec, and boundary cell processing takes approximately 80 nsec. It takes approximately 35 μ s to estimate 10 parameters using 41 known symbols. To

evaluate signal processing performance, we have also developed DSP-based RLS processor. As a result, the systolic array-based parameter estimator was roughly one hundred times as fast as the DSP-based estimator. To evaluate the processor's performance, we conducted adaptive array experiments using a complex baseband fading/array response simulator. The experimental results were then compared with those from a computer simulation under 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, we found that no difference in BER performance curves was 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 ($N \geq 4$) if there are one desired and three interference signals, each of which is sent over an independent one-path Rayleigh fading channel. The major conclusion of these experiments is that the systolic array RLS processor can well handle 12 Msymbol/s QPSK signal bursts. It is verified that this systolic array RLS processor can be used in the development of space- and time-domain equalizers in broadband mobile multimedia communications.

Acknowledgement

The authors wish to thank Dr. Nobuo Nakajima, former senior vice president of NTT DoCoMo, Inc., for his encouragement during this research.

References

- [1] S. Haykin, J. Litva, and T. Shepherd, *Radar Array Processing*, Springer-Verlag, Berlin, 1993.
- [2] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, New Jersey, 1996.
- [3] J. McCanny, J. McWhirter, and E. Swartzlander, Jr., *Systolic Array Processors*, Prentice-Hall, New York, 1989.
- [4] H. Leung and S. Haykin, "Stability of recursive QRD-LS algorithms using finite-precision systolic array implementation," *IEEE Trans. Acoust., Speech & Signal Process.*, vol.37, no.5, pp.760-763, 1989.
- [5] C. Ward, P. Hargrave, and J. McWhirter, "A novel algorithm and architecture for adaptive digital beamforming," *IEEE Trans. Antennas & Propag.*, vol.AP-34, no.3, pp.338-346, 1986.
- [6] R. Lackey, H. Baurle, and J. Barile, "Application-specific super computer," *Real Time Signal Processing XI, Proc. of SPIE*, vol.977, pp.187-195, 1988.
- [7] T. Asai, H. Yoshino, and T. Matsumoto, "A systolic array processor for parallel processing of RLS algorithm," *Proc. 1999 Communications Society Conference of IEICE*, B-5-53, p.288, 1999.
- [8] S. Tsukamoto, T. Saso, T. Sakaki, H. Yoshino, and T. Matsumoto, "A complex baseband fading/array response simulator," submitted to *IEEE Trans. Vehicular Technology*.



Takahiro Asai received his B.S. and M.S. degrees from Kyoto University, Kyoto, Japan, in 1995 and 1997, respectively. In 1997, he joined NTT DoCoMo, Kanagawa, Japan. Since then, he has been involved in the research of time-space signal processing for very high-speed mobile signal transmission. He is a member of the institute of electrical and electronics engineers.



Tadashi Matsumoto received his B.S., M.S., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 1978, 1980, and 1991, respectively. He joined Nippon Telegraph and Telephone Corporation (NTT) in April 1980. From April 1980 to May 1987, he was involved in the research of signal transmission technologies such as modulation/demodulation schemes, as well as radio link design for mobile communications systems. He participated in the R&D project of NTT's high capacity mobile communications system where he was responsible for the development of the base-station transmitter/receiver equipment for the system. From May 1987 to February 1991, he studied error control strategies such as forward error correction, trellis-coded modulation, and automatic repeat request in digital mobile radio channels. He developed an efficient new automatic repeat request scheme suitable to the error occurrence in TDMA mobile signal transmission environments. He was involved in the development of a Japanese TDMA digital cellular mobile communications system. He led the development of the facsimile and data communications service units for the system. In July 1992, he transferred to NTT DoCoMo, Inc., Kanagawa, Japan. From February 1991 to April 1994, he intensively studied multiuser detection schemes for multipath mobile communications environments. He also concentrated on the research of a maximum a *posteriori* probability (MAP) algorithm and its reduced complexity version for decoding concatenated codes. From 1992 to 1994, he served as a part-time lecturer at Keio University. In April 1994, he moved to NTT America, where he served as Senior Technical Advisor of the joint project with NTT and NEXTEL Communications. In March 1996, he returned to NTT DoCoMo. Since then, he has been conducting research on time-space signal processing for very high-speed mobile signal transmission. Presently, he is an Executive Research Engineer at NTT DoCoMo. He is a senior member of the institute of electrical and electronics engineers.