RELAXED LOOK-AHEAD TECHNIQUE FOR PIPELINED IMPLEMENTATION OF ADAPTIVE MULTIPLE-ANTENNA CDMA MOBILE RECEIVERS

Ramin Baghaie, Stefan Werner and Timo Laakso
Helsinki University of Technology, Laboratory of Telecommunications Technology
P.O. BOX 3000, 02015 HUT, Finland
Tel: +358-9-4511; fax: +358-9-451 2474
e-mail: {ramin.baghaie, stefan.werner and timo.laakso}@hut.fi

ABSTRACT

In this paper, Relaxed Look-Ahead technique for pipelined implementation of an adaptive Direct-Sequence Code Division Multiple Access receiver is proposed when multiple antennas are utilized for mobile communications. Adaptive multiple-antenna receivers can provide insensitivity to the interfering powers. They also provide room for more users or require smaller number of antennas than the matched filter solution. A number of approximation techniques are utilized to pipeline the adaptive algorithm used for the proposed multiple-antenna receiver. The resulting pipelined receiver achieves a higher throughput or requires lower power as compared to the receiver using the serial algorithm. With the aid of simulations, for different levels of pipelining and different number of antennas, the signal-to-interference ratio and the bit error rate versus the relative interfering power are illustrated.

1 INTRODUCTION

Transformations are modifications to the computational structure in a manner that the input-output behavior is preserved. By using transformations it is possible to explore a number of alternative architectures, and depending on the application, to choose those which result in higher throughput, reduced area or lower power consumption [1-4]. In the area of digital communications, there is a growing need for circuits with higher speed having a low power consumption. Two popular approaches for achieving higher processing speed are pipelining and parallel processing. However, the pipeline approach could be more desirable in mobile communications due to its lower hardware cost.

Pipelined DSP algorithms allow us to tradeoff speed, power and area during the course of VLSI implementation. Pipelining is simply accomplished by placing latches at any feed-forward cutsets of the data flow graph representation of the algorithm. However, pipelining of algorithms having a feedback loop is not a trivial task [1].

Inserting latches to pipeline the recursive loop of such algorithms is useful for execution of multiple interleaved independent data, but not for improving their iteration bound [5]. That is why different algorithm transformation techniques such as the Look-Ahead (LA) and the Relaxed Look-Ahead (RLA) have been proposed for the pipelining of recursive DSP algorithms [1-3,6]. By utilizing the above mentioned techniques one can improve the period of the iteration bound.

These transformations introduce additional concurrency in a serial algorithm at the expense of hardware overhead. The look-ahead technique has been successfully applied to a number of such algorithms [1]. The LA technique, however, results in a large hardware overhead as it transforms a serial algorithm into an equivalent pipelined algorithm. This equivalency is in terms of the input-output behavior [1-3].

The RLA technique, however, involves approximating the algorithms obtained via the look-ahead technique. Through these approximations, the technique maintains functionality of the algorithm rather than the input-output behavior.

A number of approximations such as sum, product and delay relaxation are possible and each result in a different algorithm. Depending on the approximation, there may or may not be a performance degradation. In the context of adaptive filtering, the approximation can be very harsh and still result in minimal performance loss [1]. Unlike the LA technique, the application of the RLA technique modifies the original algorithm and therefore a convergence analysis is necessary. This could be considered as one drawback when using the relaxed look-ahead technique, since this analysis can be cumbersome. However, despite of this, the resulting pipelined algorithm requires lower hardware overhead as compared to the conventional LA technique and still achieves a higher throughput compared to the serial algorithm [1].

This increased of throughput as a result of pipelining can be exchanged for either reducing power or reducing chip area. Reducing power can be done in combination with power supply scaling [4,7]. Area reduction, however, can be achieved in combination with folding transformation [1]. Both power and area reductions are of great importance when implementing mobile communication systems.

This paper is organized as follows. Section 2 presents the structure of the multiple-antenna receiver. In Section 3, the pipelined implementation of the adaptive receiver is discussed. In Section 4, simulation results are reported. Concluding remarks are provided in Section 5.

2 RECEIVER'S STRUCTURE

In [8], a stochastic gradient algorithm was proposed which only requires knowledge of the desired user's spreading code. In [9], the idea in [8] was generalized by including multiple antennas and also employing adaptive algorithms.

The structure of the receiver equipped with N antennas is shown in Figure 1 [9]. Each of the N antenna branches contains a linear filter whose coefficients are to be optimized.

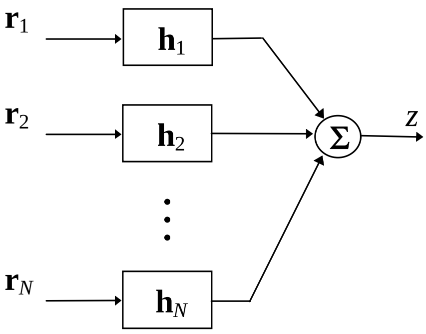

Figure 1. Structure of the linear receiver

The filtered signals from each antenna are then added together to form a decision variable. In Figure 1, $\mathbf{r}_i$ denotes the received signal after chip-matched filtering at antenna i, $\mathbf{h}_i$ contains the complex filter coefficients for the ith antenna, and z is the decision variable formed by adding the filtered outputs from each antenna.

The filter coefficients and the received sequences from the antennas are collected in vectors as:

$$\mathbf{h} = \left[\mathbf{h}_1^{\mathrm{T}} \ldots \mathbf{h}_N^{\mathrm{T}}\right]^{\mathrm{T}} \tag{1}$$

$$\mathbf{r} = \left[\mathbf{r}_1^{\mathrm{T}}\ \mathbf{r}_2^{\mathrm{T}} \ldots \mathbf{r}_N^{\mathrm{T}}\right]^{\mathrm{T}} \tag{2}$$

Using the above notation, the output from the receiver can be written as:

$$z = \mathbf{h}^{\mathrm{H}}\mathbf{r} \tag{3}$$

The objective is to find the filter $\mathbf{h}$ such that the output is minimized under the constraints that the desired user's code sequence in every antenna can pass undistorted. The minimization problem can now be formulated as [9]:

$$\begin{aligned} &\hat{\mathbf{h}} = \arg\min_{\mathbf{h}} E\left\{|z|^2\right\} \\ &\text{subject to:} \quad \mathbf{C}^{\mathrm{H}}\mathbf{h} = \mathbf{u} \end{aligned} \tag{4}$$

where matrix $\mathbf{C}$ and vector $\mathbf{u}$ are:

$$\mathbf{C} = \begin{bmatrix} a_1\mathbf{s}_{1,1} & 0 & \cdots & 0 \\ 0 & a_2\mathbf{s}_{1,2} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & a_N\mathbf{s}_{1,N} \end{bmatrix} \tag{5}$$

$$\mathbf{u} = \left[|a_1|^2\ |a_2|^2 \ldots |a_N|^2\right]^{\mathrm{T}} \tag{6}$$

where $\mathbf{s}_{1,i}$ is the code sequence with length G and a_i is the complex phase factor of the desired user at the antenna element. The solution to this problem is found by the method of Lagrange multipliers, see, e.g., [10]:

$$\mathbf{h}_{\mathrm{opt}} = \mathbf{R}^{-1}\mathbf{C}\left[\mathbf{C}^{\mathrm{H}}\mathbf{R}^{-1}\mathbf{C}\right]^{-1}\mathbf{u} \tag{7}$$

where $\mathbf{R}$ is the correlation matrix. The closed-form solution of Equation (4) is not suitable in practice, as we need to estimate the correlation matrix and perform an inversion.

Thus, an adaptive implementation of the detector is considered. In [9], the use of the Frost algorithm [11] was proposed.

We will here use the structure of the generalized sidelobe canceler [10,12], which transforms a constrained problem into an unconstrained problem by means of an orthogonal decomposition of $\mathbf{h}$. The main reason for doing this is that simpler algorithms can be applied.

The idea is to divide the weight vector $\mathbf{h}$ into two parts as:

$$\mathbf{h} = \mathbf{h}_q - \mathbf{C}_a\mathbf{h}_a \tag{8}$$

where $\mathbf{h}_q$ is a fixed vector satisfying the constraint equations, $\mathbf{C}_a$ is a $GN\times(GN\text{-}N)$ matrix which is the orthogonal complement to the constraint matrix, i.e., $\mathbf{C}_a^{\mathrm{H}}\mathbf{C} = \mathbf{0}$, and $\mathbf{h}_a$ is an adaptive filter of dimension $(GN\text{-}N)\times 1$, and unaffected by the constraints.

By choosing $\mathbf{h}_q = \mathbf{C}(\mathbf{C}^{\mathrm{H}}\mathbf{C})^{-1}\mathbf{u}$ and defining $\mathbf{x} = \mathbf{C}_a^{\mathrm{H}}\mathbf{r}$ and $d = \mathbf{h}_q^{\mathrm{H}}\mathbf{r}$, we can apply the LMS adaptive implementation for the update of the vector $\mathbf{h}_a$ [10].

3 PIPELINED ADAPTIVE RECEIVER

Consider the LMS algorithm of Equations (9) and (10) [10]:

$$\mathbf{h}_a(k) = \mathbf{h}_a(k-1) + \mu\mathbf{x}(k)z^*(k) \tag{9}$$

$$z(k) = d(k) - \mathbf{h}_a^{\mathrm{H}}(k-1)\mathbf{x}(k) \tag{10}$$

where μ is the step size parameter.

By applying the M-step look-ahead to Equation (9) we have:

$$\mathbf{h}_a(k) = \mathbf{h}_a(k-M) + \mu\sum_{i=0}^{M-1}\mathbf{x}(k-i)z^*(k-i) \tag{11}$$

For M = 1, Equation (14) represents the LMS algorithm of Eq. (9). Substituting Eq. (11) in Eq. (10) leads to:

$$\begin{aligned} z(k) = d(k) - \Big[&\mathbf{h}_a^{\mathrm{H}}(k-M-1) + \\ \mu\sum_{i=0}^{M-1}&\mathbf{x}^{\mathrm{H}}(k-i-1)z(k-i-1)\Big]\mathbf{x}(k) \end{aligned} \tag{12}$$

The above technique, however, results in a large hardware overhead since it transforms the conventional serial LMS algorithm into an equivalent pipelined algorithm. The aforementioned equivalency is in terms of the input-output behavior. For many applications the hardware overhead of Eq. (12) can not be tolerated. This is specially significant when M is large and implementation of mobile receivers are of interest. Thus, a number of approximations should be utilized.

Assuming that μ is sufficiently small, the third term on the right hand side of Eq. (12) can be approximated as zero. Finally, by replacing $\mathbf{h}_a^{\mathrm{H}}(k-M-1)$ by $\mathbf{h}_a^{\mathrm{H}}(k-M)$ [1], Equation (12) can be approximated as:

$$z(k) = d(k) - \mathbf{h}_a^{\mathrm{H}}(k-M)\mathbf{x}(k) \tag{13}$$

Equations (11) and (13) describe the pipelined LMS algorithm.

In Section 4, with the aid of simulations it is demonstrated that these approximations are reasonable. Through these approximations, the functionality of the algorithm has been maintained. However, the input-output behavior of the Equations (9) and (10) has been altered.

As a result of these relaxation techniques, the convergence condition should be checked. This problem has been addressed in [1], and it is shown that the upper bound on μ to guarantee the convergence is found to be tighter than that of the serial LMS algorithm.

Due to the approximations used when driving Equation (13), the performance of the receiver may degrade. Usually, for non-stationary signals, this would mean slight increase in the mean-squared error. In the context of our application, simulation results of Section 4 illustrate that for a moderate *M*, these approximations result in minimal performance loss.

3.1 The Architecture

Figure 2 illustrates the architecture of the pipelined LMS algorithm when *M*=3. As it can be observed from the shaded region of Figure 2, by applying the RLA technique, (*M*-1) additional adders and delay elements are required per filter tap. In the receiver of Figure 1, *N*(*G*-1) taps are needed. In the case of stationary signals, one could still apply more approximation to Eq. (11) by using techniques such as the sum and delay relaxation and further reduce the hardware overhead of Equation (11) [1].

Also, as a result of Equation (11), *M* delays have been introduced in the recursive loops. By proper distribution of these extra delays, the pipelined architecture will operate *M* times faster. This increased of throughput as a result of pipelining can be exchanged for either reducing power or reducing area on the chip. Reducing power can be done in combination with power supply scaling [4,7]. Area reduction can be achieved in combination with folding transformation [1]. Power or area reductions are of great importance when implementation of mobile communication systems are of interest.

4 SIMULATION RESULTS

Simulations have been conducted to compare the performance of the conventional serial LMS and the pipelined LMS algorithms for different levels of pipelining or speedup factors *M*. Furthermore, the above mentioned algorithms were compared for different number of antennas *N*.

In these simulations, antennas were structured as a uniform linear array with half the wavelength spacing. The direction of arrival was set to 15° and the signal-to-noise ratio at the antennas for the desired user was 8 dB. The spreading sequences were Gold codes of length 7. The number of users in these simulations was five. Figures 3 and 4 illustrate the average signal-to-interference ratio (SIR) as a function of the relative power of the interfering users when one and two antennas were used respectively. In these simulations, the interfering power of all users varies from 0 to 10 dB. The initial condition for the adaptive filter $\mathbf{h}_a$ is the zero vector, and therefore, the output of the filter $\mathbf{h}$ up to *M* iterations is the same as the output of the $\mathbf{h}_q$.

As can be seen from Figures 3 and 4, as *M* (the number of pipelining stages) increases, the SIR will decrease. This is due to the higher misadjustment as a result of the approximations. Figure 5 illustrates the bit error rate (BER) curves as a function of the relative powers of the interfering users when using one antenna for different speedup factors. The curves are obtained by averaging 20000 bits from 100 independent trials.

By comparing Figures 3 and 4, we can observe that the level of pipelining *M* should be carefully selected when more antennas are introduced. As an example, consider the case where the relative interference power is 10 dB. For a single-antenna receiver with *M*=1 roughly the same SIR can be achieved as with two antennas having *M=10*.

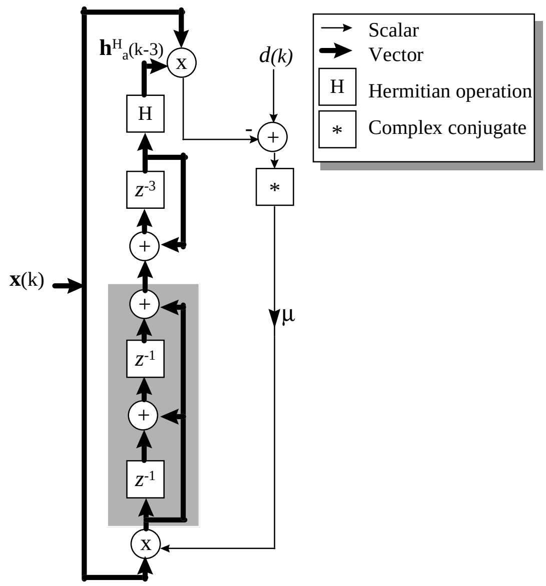

Figure 2. Implementation of Equations (11) & (13) for *M* = 3.

As it was mentioned earlier in Section 3, as a result of the relaxation techniques used, the upper bound on μ to guarantee the convergence is found to be tighter than that of the conventional serial LMS algorithm [13]. Figure 6 illustrates the SIR as a function of the number of the iterations for different levels of pipelining.

As can be seen from this figure, the convergence speed of the pipelined LMS was found to be about the same as in the case of the conventional LMS i.e., *M* = 1. The misadjustment, however, will not stay the same as *M* increases. One should carefully select the step size μ.. In order to satisfy the upper bound of convergence as *M* increases smaller μ should be selected. This, however, will result in slower convergence speed as expected.

5 CONCLUSIONS

Pipelined implementation of a DS-CDMA receiver was proposed when multiple antennas are utilized in mobile receivers. A number of approximation techniques were utilized to introduce pipelining into the conventional serial LMS algorithm. As a result, the pipelined receiver achieves a higher throughput as compared to the receiver using the serial algorithm. This increase was achieved at the expense of (*M*-1) additional adders and delay elements per filter tap. The increased throughput, however, can be traded for power reduction.

Simulations were carried out to illustrate the SIR and BER versus the relative interfering power for different number of antennas and different levels of pipelining. Also, the convergence speed for different levels of pipelining was compared.

ACKNOWLEDGMENT

This work is part of a research project of the Institute of Radio Communication (IRC) funded by Technology Development Center (TEKES), NOKIA Research Center, Telecom Finland and the Helsinki Telephone Company.

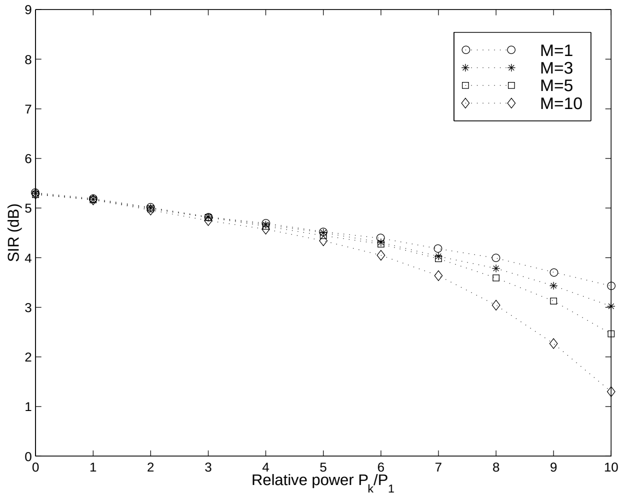

Figure 3. SIR versus the relative powers of the interfering users when using one antenna (*N*=1) for *M*=1, 3, 5, and 10.

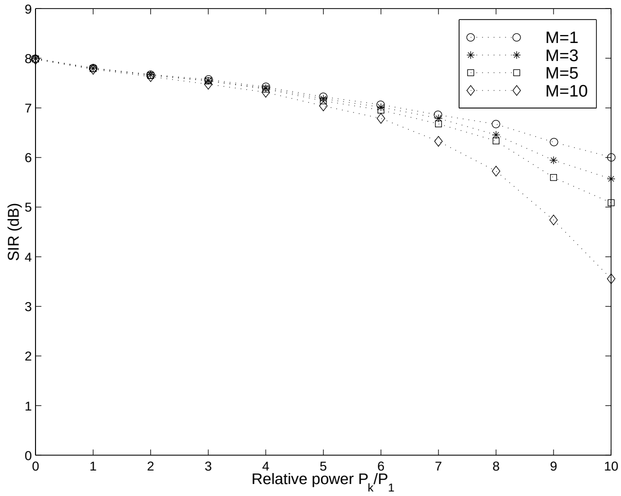

Figure 4. SIR versus the relative powers of the interfering users when using two antennas (*N*=2) for *M*=1, 3, 5, and 10.

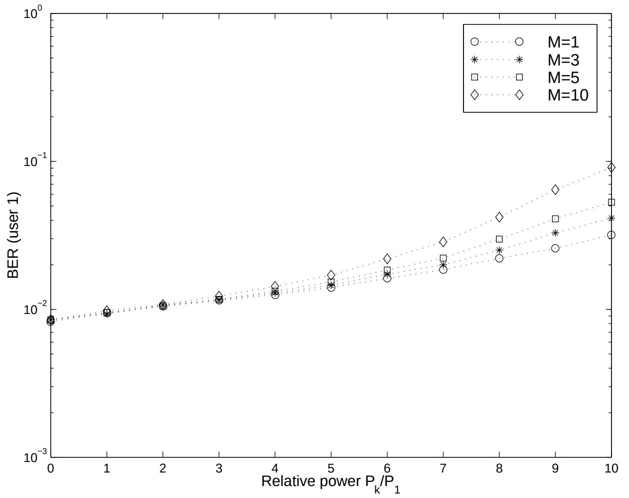

Figure 5. BER versus the relative powers of the interfering users when using one antenna (*N*=1) for *M*=1, 3, 5, and 10.

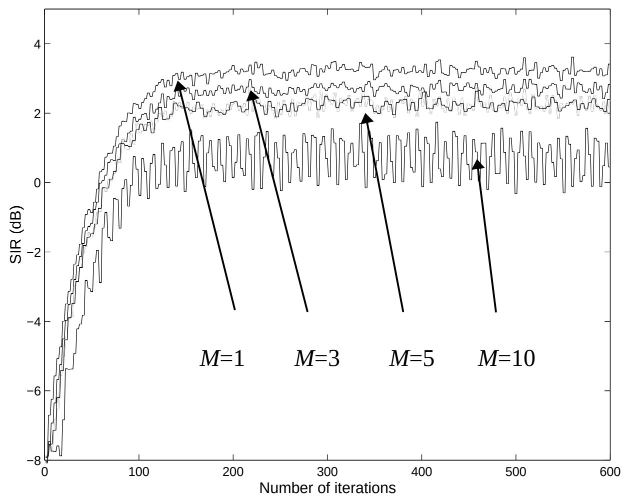

Figure 6. SIR as a function of the number of iterations (500 runs smoothed) when using one antenna and *M*=1, 3, 5, and 10.

REFERENCES

[1] N. Shanbhag and K. Parhi, *Adaptive Digital Filters*, Kluwer Academic Publishers, 1994.

[2] K. Parhi, "Algorithm transformation techniques for concurrent processors," *Proceedings of the IEEE*, vol. 77, pp. 1879-1895, Dec. 1989.

[3] K. Parhi and D. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters - Part I: Pipelining using scattered look-ahead and decomposition," *IEEE Trans.on ASSP*, vol. 37, pp. 1099-1117, July 1989.

[4] M. Chandrakasam, M. Potkonjak, R. Mehra, J. Rabaey, and R. Broderson, "Optimizing power using transformations," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, no. 1, pp. 12-31, Jan. 1995.

[5] M. Renfors and Y. Neuvo, "The maximum sampling rate of digital filters under hardware speed constraints," *IEEE Transactions on Circuits and Systems*, vol. CAS-28, no. 3, pp. 196-202, Mar. 1981.

[6] K. Matsubara, K. Nishikawa, and H. Kiya, "Pipelined LMS adaptive filter using a new look-ahead transformation," in *IEEE International Symposium on Circuits and Systems*, Hong Kong, pp. 2309-2312, June 1997.

[7] N. Shanbhag and M. Goel, "Low-Power adaptive filter architect. and their applic. to 51.84 Mb/s ATM-LAN," *IEEE Trans. on Signal Proc.*, vol. 45, pp. 1276-1290, May 1997.

[8] M. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Transactions on Information Theory*, vol. 41, pp. 944-960, July 1995.

[9] S. Werner, T. Laakso, and J. Lilleberg, "Adaptive multiple-antenna receiver for CDMA mobile reception," *to appear in ICC'98*, Atlanta, USA, June 1998.

[10] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 1996.

[11] O. Frost, "An algorithm for linearly constrained adaptive array processing," *Proceedings of the IEEE*, vol. 60, pp. 926-935, Aug. 1972.

[12] L. Griffiths and C. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. on Antennas and Propag.*, vol. AP-30, pp. 27-34, Jan. 1982.

[13] N. Shanbhag and K. Parhi, "Relaxed look-ahead pipelined LMS adaptive filters and their application to ADPCM coder," *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Proc.*, vol. 40, pp. 753-766, Dec. 1993.