PERFORMANCE OF ADAPTATION ALGORITHMS IN MULTIPATH CHANNEL EQUALIZATION FOR CDMA SYSTEMS

A. F. C. Vliese, S. L. Netto, M. L. R. de Campos, and P. S. R. Diniz

COPPE/DEL-EE/Federal University of Rio de Janeiro,
PO Box 68504, Rio de Janeiro, RJ, 21945-970, Brazil
vliese@lps.ufrj.br, sergioln@lps.ufrj.br, campos@lps.ufrj.br, diniz@lps.ufrj.br

ABSTRACT

In this work, performance of several adaptation algorithms are investigated for the reception of CDMA signals. Two adaptive receiving frameworks are considered: the standard channel equalization with training sequence, and the fractionally-spaced equalization. Amongst the several widely known adaptation algorithms, we consider the LMS, the RLS, the binormalized data-reusing (BNDR), and the quasi-Newton (QN) algorithms. The overall system behavior is verified in a multiuser environment (15 users), and for several values of the signal-to-noise ratio (SNR). Algorithm performances are measured by the resulting receiving bit error-rate.

1. INTRODUCTION

Due to their intrinsic advantages over the fixed systems, e.g., mobility and facility of system expansion, mobile cellular phones have become very popular in the past decade. Code division multiple access (CDMA) has become one standard technology for digital mobile telephony as it is less amenable to channel imperfections than previously used techniques, and it allows a large number of users for a given bandwidth. Other advantages of the CDMA technology include reduced signal distortion due to multipath propagation, reduced missed calls during handoff, higher data secrecy, reduced costs for system operation and expansion, reduced electromagnetic interference to other electronic devices, and (possible) reduced risks to human health due to the attenuation of the average transmission power [1,2,3].

In this work we analyze the ability of CDMA systems on dealing with the problem of uplink multipath propagation. Such phenomenon typically occurs when the signal, leaving the mobile station, reaches the base station following several different paths: a more direct one and others due to obstacles such as buildings, trees, hills, etc. To reduce the effect of multipath propagation we consider two commonly employed techniques: adaptive channel equalization and adaptive fractionally-spaced equalization [4,5]. In these frameworks we evaluate the performances of four adaptation algorithms: two gradient-type schemes, namely the LMS [4] and the binormalized data-reusing (BNDR) [6] algorithms, and two Newton-type schemes, the RLS [4] and the quasi-Newton (QN) algorithm [7].

This paper is organized as follows: in the following section, we describe reception of CDMA signals with adaptive equalization. In Section 3, we briefly describe the LMS, BNDR, RLS, and QN adaptation algorithms, listing their respective characteristics with respect to computational complexity, convergence speed, robustness, etc. In Section 4 we include computer experiments comparing the performances of the adaptation algorithms mentioned in Section 3 when receiving CDMA signals in a variety of setups. Section 5 closes the paper listing conclusions drawn from the experiments carried out.

2. ADAPTIVE CDMA SIGNAL RECEPTION

A basic CDMA system is depicted in Fig. 1. A common drawback for this type of system is that several copies of the transmitted signal reach the receiver after following a distinct path each. This is the so-called multipath problem. Such issue can be greatly reduced with a proper channel-equalization procedure, as explained below.

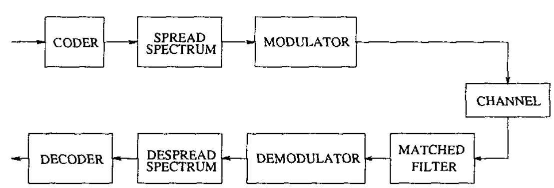

Figure 1: Basic CDMA system.

2.1 Standard Adaptive Equalizer

The block diagram of a standard adaptive channel equalizer is given in Fig. 2. In such scheme an input signal, also known to the receiver, is initially transmitted and corrupted by the multipath channel. An adaptive filter is then used to remove channel distortion, in such a way that the adaptive-filter output signal resembles the original transmitted signal after the adaptive-filter convergence, i.e., when the error

0-7803-5682-9/99/$10.00©1999 IEEE.

signal approximates zero. The delay block that appears in this configuration is used to compensate for the processing delay introduced by both the channel and the adaptive filter.

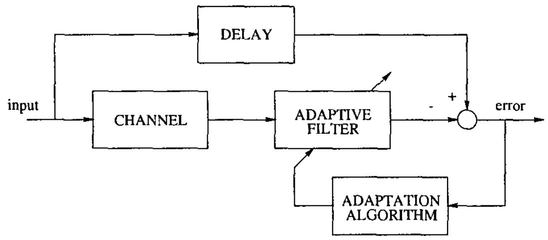

Figure 2: Standard adaptive channel equalizer.

2.2 Fractionally-Spaced Adaptive Equalizer

In the fractionally-spaced equalizer, as described in [5], the idea is to work on a rate M times faster than the original sampling rate. In that manner, a large number of iterations is required, but each iteration takes just $1/M$ seconds of the original sampling period. This can be achieved by interpolating the original training sequence introducing $M-1$ zeros between the original samples of the input sequence. A low-pass filter is then used to smooth the resulting interpolated signal, followed by an energy normalization procedure that adjusts the resulting energy level. Such scheme is used to generate the new input signal that will be used in the same fashion as described for the standard adaptive channel equalizer, but on a time scale M times higher than the previous one.

Figure 3: Generating the input and reference signals for the adaptive fractionally spaced equalizer.

3. ADAPTATION ALGORITHMS

The block diagram of a general adaptive system is seen in Fig. 4, where $x(n)$ represents the input signal, $y(n)$ the adaptive-filter output signal, $d(n)$ the desired output signal, and $e(n)$ the error signal.

In this work, we will assume that the adaptive filter is implemented by a time-varying FIR filter, the transfer function of which is given by

$$H(z,n) = w_0(n) + w_1(n)z^{-1} + \ldots + w_N(n)z^{-N} \tag{1}$$

where N is the filter order and the w_i's are the adaptive filter coefficients. We also use the notation $\boldsymbol{w}(n)$, for the adaptive coefficient vector, and

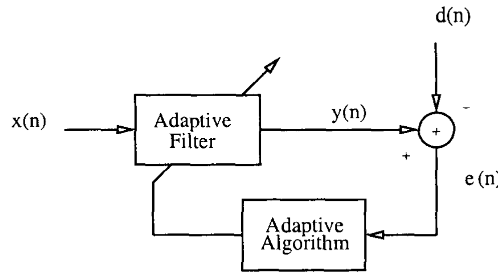

Figure 4: Adaptive-filter block diagram.

$\boldsymbol{x}(n)$, for the input signal vector, defined as

$$\begin{aligned} \boldsymbol{w}(n) &= [\ w_0(n) \quad w_1(n) \quad \ldots \quad w_N(n)\ ]^T \\ \boldsymbol{x}(n) &= [\ x(n) \quad x(n-1) \quad \ldots \quad x(n-N)\ ]^T \end{aligned} \tag{2}$$

respectively.

3.1 The LMS and RLS algorithms

The LMS and RLS are standard procedures and their descriptions, derivations, and analyses can be found in many standard books on adaptive signal processing, such as [4]. For the sake of completeness, their pseudo-code descriptions are included in Table 1 and Table 2, respectively.

Meanwhile, the BNDR and QN algorithms are somewhat new and not as widely known as the other two algorithms. They are then discussed in more depth below.

TABLE I
LMS ALGORITHM DESCRIPTION

Given $\boldsymbol{x}(n)$, $d(n)$, and $\boldsymbol{w}(n)$, compute:
$y(n) = \boldsymbol{x}^T(n)\boldsymbol{w}(n)$
$e(n) = d(n) - y(n)$
$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu e(n)\boldsymbol{x}(n)$

3.2 The BNDR algorithm

The BNDR is a variation of the LMS algorithm where at each iteration the algorithm is forced to adapt in two orthogonal directions. Such scheme can be visualized as each set of data is used twice in the algorithm convergence, once at iteration n, and then at iteration $n+1$, thus the *data-reusing* terminology. The binormalized term comes from the proper choice of the convergence parameter for each of the two directions followed by the algorithm at each iteration. It can be shown that the BNDR algorithm performs extremely well, although it only

TABLE II
RLS ALGORITHM DESCRIPTION

Given $\boldsymbol{x}(n)$, $d(n)$, $\boldsymbol{w}(n)$, and $\boldsymbol{R}(n)$, compute:

$$y(n) = \boldsymbol{x}^T(n)\boldsymbol{w}(n)$$
$$e(n) = d(n) - y(n)$$
$$\boldsymbol{t}(n) = \boldsymbol{R}^{-1}(n)\boldsymbol{x}(n)$$
$$\tau(n) = \boldsymbol{x}^T(n)\boldsymbol{t}(n)$$
$$\boldsymbol{R}^{-1}(n+1) = \frac{1}{\lambda}\left\{\boldsymbol{R}^{-1}(n) - \frac{\boldsymbol{t}(n)\boldsymbol{t}^T(n)}{\lambda+\tau(n)}\right\}$$
$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \frac{e(n)}{\lambda+\tau(n)}\boldsymbol{t}(n)$$

involves vector operations, even in cases of colored input signal, as opposed to the standard LMS algorithm. Table 3 includes a summary for the BNDR algorithm as given in [6].

TABLE III
BNDR ALGORITHM DESCRIPTION

Given $\boldsymbol{x}(n)$, $\boldsymbol{x}(n-1)$, $d(n)$, $d(n-1)$, and $\boldsymbol{w}(n)$, compute:

$$a = \boldsymbol{x}^T(n)\boldsymbol{x}(n-1)$$
$$b = \boldsymbol{x}^T(n)\boldsymbol{x}(n)$$
$$c = \boldsymbol{x}^T(n-1)\boldsymbol{x}(n-1)$$
$$d = \boldsymbol{x}^T(n)\boldsymbol{w}(n)$$
$$e = \boldsymbol{x}^T(n-1)\boldsymbol{w}(n)$$
$$den = bc - a^2 + \epsilon$$
$$A = (d(n)c + ea - dc - d(n-1)a)/den$$
$$B = (d(n-1)b + da - eb - d(n)a)/den$$
$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + A\boldsymbol{x}(n) + B\boldsymbol{x}(n-1)$$

3.3 The QN algorithm

The QN algorithm [7] was derived as a stable alternative for the RLS algorithm when highly correlated signals are present, even in cases of nonpersistent excitation. The algorithm was shown to maintain stability even when implemented with finite-precision arithmetic. Table 4 gives a summary for the QN algorithm.

4. COMPUTER EXPERIMENTS

4.1 Simulation Descriptions

Transmitter

To compare the adaptive algorithm performances a system model for a CDMA uplink was developed. We then considered 15 simultaneous users, not necessarily synchronized, each one with an independent

TABLE IV
QN ALGORITHM DESCRIPTION

Given $\boldsymbol{x}(n)$, $d(n)$, $\boldsymbol{w}(n)$, and $\boldsymbol{R}(n)$, compute:

$$y(n) = \boldsymbol{x}^T(n)\boldsymbol{w}(n)$$
$$e(n) = d(n) - y(n)$$
$$\boldsymbol{t}(n) = \boldsymbol{R}^{-1}(n)\boldsymbol{x}(n)$$
$$\tau(n) = \boldsymbol{x}^T(n)\boldsymbol{t}(n)$$
$$\mu(n) = 1/(2\tau(n))$$
$$\boldsymbol{R}^{-1}(n+1) = \boldsymbol{R}^{-1}(n) + \frac{[\mu(n)-1]}{\tau(n)}\boldsymbol{t}(n)\boldsymbol{t}^T(n)$$
$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \frac{e(n)}{\tau(n)}\boldsymbol{t}(n)$$

source at a rate of 9,600 bits per second. 10,000 bits were transmitted in 100 bursts of 100 bits of data for each user. Source bits were spectrally spread by direct sequences with processing gain equal to 32 using Hadamard codes. The symbols were BPSK modulated before transmission.

Source bits were saved for further comparison to measure resulting bit error rate (BER) for each user and for each algorithm. To simulate different signal paths, each user had a distinct time-invariant channel impulse response. Each channel received interference of an AWGN process to simulate a hostile practical environment. Power normalization was implemented such that signals from all users reached the receiver with equal power. The signal to noise ratio was calculated looking upon signal power of user number 1 soon after modulation and before transmission through the channel. Detection and equalization were performed with respect to user number 1.

Receiver

The adaptive filter used in the equalization procedure had 11 taps (order equal to 10) and the number of training samples was 40. The adaptation algorithms were adjusted such that the resulting BER was as low as possible. For that matter, the following algorithm parameters were used: $\mu_{\text{LMS}} = 0.047$, $\mu_{\text{BNDR}} = 0.5$, $\lambda_{\text{RLS}} = 0.95$.

Equalization was performed after matched filtering and demodulation and before despreading.

4.2 Simulation Results

As we can see in Fig. 5, for the standard adaptive equalization, it is verified that the gradient-type algorithms, such as the LMS and BNDR algorithms, are outperformed by the Newton-type algorithms, such as the RLS and QN algorithms.

In the fractionally-spaced equalization, as depicted in Figs. 6 and 7, the BNDR and QN algorithms are outperformed by the LMS and RLS al-

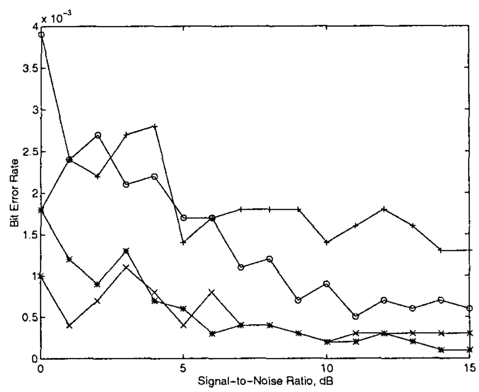

Figure 5: Bit error rate for the standard adaptive channel equalizer for several values of the signal-to-noise ratio: LMS (+); BNDR (o); QN (*); RLS (x).

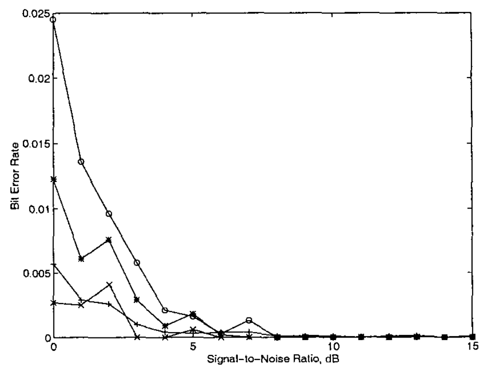

Figure 6: Bit error rate for the adaptive fractionally spaced equalizer for several values of the signal-to-noise ratio: LMS (+); BNDR (o); QN (*); RLS (x).

gorithms for low SNR values. For high SNR all algorithms performed similarly.

5. CONCLUSIONS

Adaptive equalization is an important segment of a CDMA system as it greatly reduces distortion introduced by the propagation channel if a proper adaptation algorithm is used. In this work we verified the performance of four adaptation algorithms, namely the LMS, RLS, BNDR, and QN algorithms, in an adaptive equalization framework. Two distinct adaptive equalizers were considered: the fractionally-spaced equalizer and the standard equalizer.

For the standard equalizer, the Newton-type algorithms performed very well, as expected due to its natural ability of processing colored signals. The BNDR algorithm had an intermediate performance

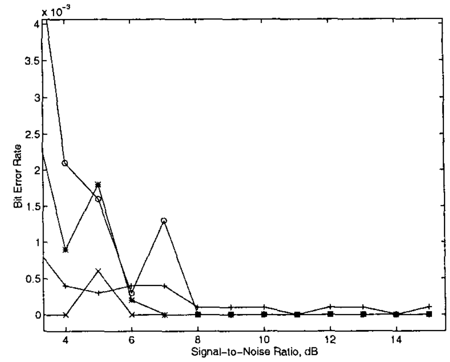

Figure 7: Detailed bit error rate for the adaptive fractionally spaced equalizer for several values of the signal-to-noise ratio: LMS (+); BNDR (o); QN (*); RLS (x).

and remained as a good alternative with low computational complexity for situations with high SNR.

For the fractionally-spaced equalizer, the normalized algorithms were outperformed by the LMS and RLS algorithms for low SNR values, say SNR< 6 dB. For high SNR values, all algorithms presented similar performance.

REFERENCES

[1] P. G. Flikkema, "Spread spectrum techniques for wireless communication," *IEEE Signal Processing Magazine*, pp. 26–36, May 1997.

[2] J. D. Laster and J. H. Reed, "Interference rejection in digital wireless communications," *IEEE Signal Processing Magazine*, pp. 37–60, May 1997.

[3] B. Sklar, *Digital Communications – Fundamentals and Applications*," Prentice-Hall, Englewood-Cliffs:PA, 1988.

[4] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Kluwer, Boston:MA, 1997.

[5] J. R. Treichler, I. Fialkow, and C. R. Johnson, Jr., "Fractionally space equalizers," *IEEE Signal Processing Magazine*, pp. 65–81, May 1996.

[6] J. A. Apolinário, Jr., M. L. R. de Campos, and P. S. R. Diniz, "The binormalized data-reusing LMS algorithm," submitted to *IEEE Trans. on Signal Processing*, 1998.

[7] M. L. R. de Campos and A. Antoniou, "A new quasi-Newton adaptive filtering algorithm," *IEEE Trans. Circuits and Systems: Analog and Digital Signal Processing*, vol. 44, no. 11, pp. 924–934, Nov. 1997.