
Author final version (often known as postprint)
Link to published version (if available):
10.1109/ISCAS.1996.541706
Link to publication record in Explore Bristol Research
PDF-document

University of Bristol - Explore Bristol Research
General rights
This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
http://www.bristol.ac.uk/pure/about/ebr-terms.html

Take down policy
Explore Bristol Research is a digital archive and the intention is that deposited content should not be removed. However, if you believe that this version of the work breaches copyright law please contact open-access@bristol.ac.uk and include the following information in your message:
• Your contact details
• Bibliographic details for the item, including a URL
• An outline of the nature of the complaint

On receipt of your message the Open Access Team will immediately investigate your claim, make an initial judgement of the validity of the claim and, where appropriate, withdraw the item in question from public view.
A HIGH THROUGHPUT ADAPTIVE DFE FOR HIPERLAN

R. Perry, D. R. Bull and A. Nix
Centre for Communications Research
University of Bristol, Bristol, BS8 1TR, UK.
email russ.perry@bristol.ac.uk, dave.bull@bristol.ac.uk,
Tel: +44 117 928 7740 Fax: +44 117 925 5265

Abstract
This paper describes two methods for increasing the throughput of an adaptive Decision Feedback Equaliser (DFE) using the LMS training algorithm. In the first method, a signed power-of-two number representation is used for the equaliser input data. Using this number representation, all multipliers can be replaced with barrel shifters and adders. In the second method, the Delayed Least Mean Square Algorithm (DLMS) is used to train the equaliser. A delay, equal to the feedforward filter length, is introduced in the filter coefficient update, which allows the DFE to be realised as the cascade of a series of modular sections.

1. INTRODUCTION
This paper discusses methods of improving the throughput rate of an adaptive equaliser. Such techniques are of current interest in the context of emerging high data rate wireless LANs such as HIPERLAN [1]. The HIPERLAN supports data rates of up to 23.5Mb/s which, even in indoor environments, can lead to very severe intersymbol interference (ISI). This necessitates the use of an adaptive equaliser in the receiver.

Numerous equaliser algorithms and architectures have been reported in the literature. Decision Feedback Equalisation is considered here, since the complexity of alternatives such as maximum likelihood sequence estimation is prohibitive for channel impulse response lengths greater than 5 symbols. A DFE can be realised, using either transversal filters, lattice filters or systolic arrays [6]. In [6] adaptive equalisers were considered for application to TDMA based systems, which in some cases, impose severe tracking requirements on the equaliser. However, in the case of HIPERLAN, reasonably stationary channel conditions can be assumed. The equaliser is trained using all, or part of, a 450-bit header packet and may then be fixed while the following data blocks (up to 49) are processed. The comparatively long training sequence allows low complexity (slow converging) algorithms to be used for equaliser training. For this reason the LMS algorithm or a variant is a natural choice. However, achieving a throughput of 23.5Mb/s is still problematic due to the sampling rate limitation associated with the coefficient update and the decision feedback loop, imposed by a conventional DFE structure.

Two methods for producing a reduced complexity high throughput rate DFE are described in this paper. The motivation for choosing a transversal filter is explained in section 2.1. The complexity of the modified DFE architectures and their convergence and output mean square error characteristics are discussed in sections 3 and 4 respectively.

2. ADAPTIVE TRANSVERSAL EQUALISER'S
Two methods for increasing the throughput rate of a transversal filter based DFE are described in this section.

2.1. Non-Uniform Number Representation
The first method uses non-uniform quantisation (a signed-power-of-two (SPT) approximation) of the equaliser input data [3]. This quantisation is applied to the input data, as opposed to the filter coefficients, since the performance of the equaliser is largely unaffected by this approximation (see section 4) while facilitating significant complexity savings. In addition, the non-uniform quantisation of the input data is required only once per input sample. In contrast, for SPT filter coefficients, it is necessary to quantise the coefficients following each update. This introduces additional latency and complexity within the coefficient update loop. The standard LMS algorithm can be used in both cases without modification.

A representation of a discrete-time B-bit two's complement number \( x(m) \) , in the signed power-of-two space [3] is given by

\[
x_N(m) = \sum_{r=1}^{N} s(r)2^{g(r)}, \quad s(r) = -1, 0, 1
\]

where \( g(r) \) is the power of the \( r \)-th power-of-two (POT) term and \( N \) is the number of POT terms used in the approximation. If \( x(m) \) is an integer, then for \( N = \lceil B/2 \rceil \), all integers in the range \(-2^{B-1} \ldots 2^{B-1} - 1\) can be exactly represented. However, for \( N < \lceil B/2 \rceil \), not all integer values that \( x(m) \) may take, can be represented by \( x_N(m) \).

Hereafter the term, N-SPT, will be used to denote an approximation (in some cases an exact representation) of a two's complement integer using \( N \) POT terms, each taking either positive or negative sign.

0-7803-3073-0/96/$5.00 ©1996 IEEE 301
The area and/or latency of a multiplier can be significantly reduced by using restricted-number representations for either the multiplier or multiplicand, i.e., using coefficients with a limitation on the number of non-zero digits. The multiplier can then be replaced with shift and addition elements. Using a 2-SPT representation of the input data, as described above, allows the multipliers in both the transversal filter and coefficient update modules (for the LMS algorithm) to be replaced with a pair of barrel shifters and a single adder.

The transversal filter based DFE operates directly on the input data samples (as opposed to the backward residuals with a pair of barrel shifters and a single adder. In a manner similar to [4], an output vector is defined as

\[ y_i(n-i) = \sum_{k=0}^{i-1} x_f(n-i-k)w_f^*(n-i-1) + y_b(n-i) \]  

where

\[ y_i(n-i) = y_{i-1}(n-i) + x_f(n-2i)w_f^*(n-i-1) + y_b(n-i) \]  

Initially we define \( y_b(n-i) \) as

\[ y_b(n-i) = \sum_{k=0}^{L-1} x_b(n-i-k)w_b^*(n-1-k) \]  

This can be interpreted as a transposed direct-form transversal filter. The delay \( k \), in the coefficient terms in (7a) is due to the delay in the output signal propagating along the filter structure. This is not a strict realisation of the DLMS algorithm. However, by inserting delays in the filter coefficient terms in (7a), a transposed filter structure implementing the DLMS algorithm is obtained i.e. the \( k \)th coefficient used in (7a) should be delayed by \( L-1-k \) iterations. In this case \( y_b(n-i) \) is given by

\[ y_b(n-i) = \sum_{k=0}^{L-1} w_b^*(n-k-1 -(L-1-k))x_b(n-i-k) \]

\[ y_b(n-i) = \sum_{k=0}^{L-1} w_b^*(n-L)x_b(n-i-k) \]  

The transformed block diagram for a (3,3) DFE using the DLMS algorithm is shown in figure 1; and consists of three identical processing modules (PMs). The latency in the output is \( 2L-1 \) sample periods. This is the time required for all the feedforward filter stages to fill and for the estimate of the desired response to propagate along the filter structure.
It should be noted that the input to the feedforward filter enters from the left whilst the previous decision is input to all the feedback filter sections simultaneously. Note also that the index for the feedforward filter coefficients increases left to right, but for the feedback filter coefficients, it decreases left to right.

The weight update for $w_f(n-i-1)$ required by (5) is obtained from (4) as

$$w_f(n-i) = w_f(n-i-1) + \beta e^*(n-L-i)x_f(n-L-2i)$$  \hspace{1cm} (8)

For the update of $w_h(n-1)$ there are two forms corresponding to equations (7a) and (7b). For (7a) the weight update is

$$w_h(n) = w_h(n-1) + \beta e^*(n-L)x_h(n-L-i)$$  \hspace{1cm} (9a)

For (7b) the weight update is given by

$$w_h(n-(L-1-i)) = w_h(n-L+i) + \beta e^*(n-2L+1+i)x_h(n-2L+1)$$  \hspace{1cm} (9b)

In both (9a) and (9b), global communication is required; in (9a) the same error term is fed back to all the coefficient update sections, whereas in (9b) the same data symbol is fed back. The form (9b) is attractive because the feedback data is only a complex number of the form $\pm 1 \pm j$. In addition, because of the reversed order of the feedback filter coefficients, the error term in (8) is the same as that required in (9b) and therefore this reduces the communication costs considerably. An individual PM for the DLMS DFE structure is shown in figure 2 using the update (9b). The throughout of a DFE, is limited now, by the time to perform a multiply shift and add (M6-M5-A3). An N-SPT approximation for the input data can also be used to reduce the complexity of the proposed filtering structure, as described for the LMS algorithm.

The new DFE structure has the additional advantage that different representations for the feedback and feedforward input data do not destroy the regularity of the structure. This is in contrast with the conventional LMS algorithm, where different feedforward and feedback filter structures would be required for the different number representations of the input data.

3. COMPLEXITY COMPARISON

In order to determine the potential complexity (area) savings from using 2-SPT feedforward input data, the gate counts required to implement LMS and DLMS based DFEs using both two's complement and 2-SPT input data have been estimated (figure 3). It is assumed in all cases, that the step size parameter is selected as a POT term to eliminate one full multiplier and that single POT terms are used for the feedback data. The gate counts used for each type of logic gate are based on commercial products [7,8]. Each gate-equivalent is a structure from which a 2-input NAND gate, or a 2-input NOR gate, can be constructed. A Baugh-Wooley parallel multiplier (for two's complement data) and a Barrel Shifter Multiplier (for SPT Data) are assumed. The filter length of the feedforward and feedback filter $L$ is fixed at $L=8$, as this is anticipated to be the longest filter required for a HIPERLAN equaliser.

It can be seen from figure 3 that the use of the SPT coded data reduces the gate count by 50% for an input wordlength of 8 bits, compared to a two's complement representation. The complexity of the DLMS algorithm differs from LMS algorithm only by the additional pipelining latches.
5. CONCLUSIONS

This paper has discussed methods to reduce the complexity and increase the throughput of adaptive transversal DFEs for applications such as HIPERLAN. It was shown that the use of a 2-SPT approximation of the input data allows a reduction in gate count of up to 50% for filter coefficients of 16-bits and a filter length of 8. A new modular structure for implementing a pipelined DFE using the DLMS algorithm was also described. This modified structure resulted in a throughput rate determined by a single multiplier, barrel shifter and adder. Using non-uniform quantisation of the input data in conjunction with this structure allows the throughput rate to be improved still further.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude to the members of the Centre for Communications Research, University of Bristol. We also wish to thank EPSRC and Hewlett Packard Laboratories for their support of this work.

REFERENCES