Dual decision-feedback equalizer with variable detection delay
Bergmans, J.W.M.

Published in:
Electronics Letters

DOI:
10.1049/el:20000389

Published: 01/01/2000

Citation for published version (APA):
Dual decision-feedback equaliser with variable detection delay

J.W.M. Bergmans

The use of a variable detection delay is proposed to improve the bit error rate and error propagation characteristics of a dual decision-feedback equaliser.

Decision-feedback equaliser: Digital transmission and recording systems often employ feedback-type detectors such as the decision-feedback equaliser (DFE) [1], fixed-delay tree search with decision feedback (FDTSDF) [2], and the dual DFE (DDFE) [3]. Among these detectors, the DDFE stands out because it achieves close-to-optimum performance while having a remarkably low complexity. The performance of the DDFE depends on its detection delay, which in the basic DDFE is fixed and determined by two internal shift registers. In this Letter the use of a variable detection delay is proposed as a means for improving the bit error rate and error propagation characteristics of the DDFE.

Dual DFE: The DFE is suboptimum because it exploits only the correct decision variables. Error propagation constitutes a second (and typically much smaller) loss factor. The DDFE represents an improvement on the DFE in both respects (Fig. 2). It uses two DFEs that run independently of each other and most of the time deliver the same decisions. Both DFEs are identical except that the threshold levels of their bit-detectors have antisymmetric offsets (a) with \( \alpha = 0.2 \) to 0.3. Most of the time, noise is small and the decisions \( d_k \) and \( d'_k \) of both DFEs are correct and identical. Occasionally, however, the decision variables \( d_k \) and \( d'_k \) of the DFEs fall within the erasure zone \([-\alpha, \alpha] \). In this event the decisions are uncertain and are, moreover, different. Since both decisions are applied to a shift register with \( \delta \) stages, the detector can permit itself a total of \( \delta \) symbol intervals to determine which of both DFE outputs to settle on. Throughout this erasure period detection thresholds are zeroed so as to maximise the probability that subsequent DFE decisions will be correct. The DFE with the correct decision \( d_k \) is likely to produce small error samples \( e_k = d_k - d_k \) in the remainder of the erasure period, while the error samples of the other DFE are likely to be relatively large because the erroneous decision causes imperfect cancellation of trailing ISI. A comparison of the energy of both errors across the erasure period is used to select between both DFEs. The switch in Fig. 2 is set accordingly, and both DFEs are realigned by transferring the register contents of the 'selected' DFE to the other one. Detection thresholds then re-assume their default values \( \pm \alpha \), and erasure detection recommences.

We first present some DFE basics. The replay signal \( r(t) \) in Fig. 1 is a linearly distorted and noisy version of the transmitted data sequence \( a_k \) of data rate \( 1/T \). The function \( h(t) \) represents the response of the system to a single transmitted bit. Noise \( n(t) \) may or may not be white. The forward filter in Fig. 1 has impulse response \( h(t) \). It serves to suppress prescursive intersymbol interference (ISI) at the decision instants \( t_k = kT \) to reject out-of-band noise, and to whiten in-band noise. Ideally the scaled and sampled system response \( q_k = (h * w)(kT) \) is a minimum-phase function so that a maximum fraction of the energy of \( q_k \) is concentrated near the time origin \( k = 0 \). Minimum-phase functions are all causal \( q_k = 0 \) for \( k < 0 \), i.e. prescursive ISI is indeed absent. Without loss of generality we may set \( q_0 = 1 \). Post-cursive ISI is due to the 'tail' \( q_1, q_2, \ldots, q_6 \). This tail serves as the impulse response of a feedback filter (FBF) that is excited by past decisions \( d_k, d_{k-1}, d_{k-2}, \ldots \). The FBF output is subtracted from the forward filter output so as to cancel post-cursive ISI. If past decisions affecting the FBF output are all correct then the decision variable \( d_k \) is just a noisy version of \( a_k \), and a new bit-decision \( d_k \) can be formed by means of a slicer with zero threshold. The difference \( e_k = d_k - d_k \) is a measure of the decision quality, and is commonly used to drive the read-path control loops.

**Fig. 1** Baseline system with decision feedback equaliser (DFE)

**Fig. 2** Schematic model of DDFE

**Fig. 3** Bit error rate against erasure threshold \( \alpha \) for various values of \( D \) at normalised information density \( 15T \times 2.85 \)

(i) \( \text{SNR} = 15 \text{dB}, \text{fixed delay} \)
(ii) \( \text{SNR} = 15 \text{dB}, D = 3 \)
(iii) \( \text{SNR} = 15 \text{dB}, D = 1.5 \)
(iv) \( \text{SNR} = 16 \text{dB}, \text{fixed delay} \)
(v) \( \text{SNR} = 16 \text{dB}, D = 3 \)
(vi) \( \text{SNR} = 16 \text{dB}, D = 1.5 \)

The performance of the DDFE is basically equal to that of near-optimum restricted-delay schemes such as FDTSDF [3]. The complexity, however, is much lower. Bursts of errors tend to be shorter than for the DFE. Erasures occur much more frequently than bit errors and can be counted to obtain an accurate prediction of the bit error rate.

**Variable detection delay:** In the basic scheme of Fig. 2, the DDFE has a fixed detection delay of \( \delta \) symbol intervals. For \( \delta = 0 \) the DDFE degenerates into a DFE. As \( \delta \) is increased beyond this point, the performance initially improves rapidly because the DDFE exploits an increasing fraction of 'useful' data energy (namely the energy that stems from the 'tail' \( q_1, q_2, \ldots, q_6 \)). Once this tail is covered, the performance gradually declines because the DDFE dwells during an increasing fraction of the time within the

**Authorized licensed use limited to: Eindhoven University of Technology. Downloaded on January 30, 2009 at 03:54 from IEEE Xplore. Restrictions apply.**
erasure periods. Throughout these periods it effectively functions as a DFE, and as \(d \to \infty\) its net performance again approaches that of the DFE.

The basic insight into the use of a variable detection delay is that most DDFE decisions can be reliably made shortly after the beginning of an erasure period. More specifically, if the erasure period commences at the instant \(k = k_0\) and if \(d_k\) denotes the accumulated error-energy-difference between the two DFEs from the start of the erasure period (i.e. \(d_k = \sum_{k_0}^{k} (e_k^2 - (\bar{e}_k^2))\), then a large positive value of \(d_k\) indicates that decisions \(d_{k+1}\) are much more likely than \(d_{k-1}\), and vice versa for a large negative value. This suggests that we end the erasure period as soon as \(|d_k|\) exceeds a prescribed reliability threshold \(D\), and no later than \(\delta\) symbol intervals after the commencement of the erasure period (i.e. no later than for \(n = \delta\)). In all cases the final DDFE decision is based on the sign of \(d_{\delta}\) at the end of the erasure period. Both DFEs are then realigned, and erasure thresholds re-assume their default values \(\delta\).

In many cases \(|d_\delta|\) will cross the reliability threshold \(D\) within a few symbol intervals from the beginning of the erasure period, so that the average detection delay \(\delta\) will be strictly smaller than the maximum delay \(\delta\). Clearly \(\delta\) is a monotonically increasing function of \(D\), ranging between 0 for \(D = 0\) and \(\delta = \delta\) for \(D = \infty\). As a rule of thumb, if the bulk of the energy of the system response \(q_k\) is contained within the first \(M\) samples, then \(D\) should be selected in order for \(\delta\) to be somewhat larger than \(M\).

The total number of additional arithmetic operations introduced by the conversion is less than 5\(N\).

**References**


**Fast computation of discrete W transform through discrete Hartley transform (DHT)**

L.Z. Cheng

A new algorithm is presented for the type-II, -III and -IV discrete W transforms which involves the simple conversion of a length-\(N\) discrete W transform into a length-\(N\) discrete Hartley transform. The total number of additional arithmetic operations introduced by the conversion is less than 5\(N\).

**Introduction**

As a generalisation of the discrete Hartley transform (DHT), the discrete W transform (DWT) has become a useful tool in signal processing [1-3]. The development of fast algorithms for performing DWTs is, therefore, an active area of research. To represent the fast algorithm for the DWT, we restate the definition of the type-I, -II, -III and -IV DWTs of a real-valued sequence \(x(n)\) as follows:

\[
X_{I}(k) = \sum_{n=0}^{N-1} x(n)\cos\left(\frac{2\pi nk}{N}\right)
\]

\[
X_{II}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi nk}{N}\right)
\]

\[
X_{III}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{(2n+1)k\pi}{N}\right)
\]

\[
X_{IV}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi nk}{N} + \frac{(2n+1)k\pi}{N}\right)
\]

Where \(k = 0, 1, 2, \ldots, N-1\), and subscripts \(I, II, III\) and \(IV\) indicate the type-I, -II, -III and -IV DWTs, denoted by DWT-I, -II, -III and -IV, respectively. Obviously, DWT-I is just the discrete Hartley transform (DHT) [4-6]. To coincide with the usually used notation, in this Letter DWT-I is still referred to as DHT.

Introduction: As a generalisation of the discrete Hartley transform (DHT), the discrete W transform (DWT) has become a useful tool in signal processing [1-3]. The development of fast algorithms for performing DWTs is, therefore, an active area of research. To represent the fast algorithm for the DWT, we restate the definition of the type-I, -II, -III and -IV DWTs of a real-valued sequence \(x(n)\) as follows:

\[
X_{I}(k) = \sum_{n=0}^{N-1} x(n)\cos\left(\frac{2\pi nk}{N}\right)
\]

\[
X_{II}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi nk}{N}\right)
\]

\[
X_{III}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{(2n+1)k\pi}{N}\right)
\]

\[
X_{IV}(k) = \sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi nk}{N} + \frac{(2n+1)k\pi}{N}\right)
\]

Where \(k = 0, 1, 2, \ldots, N-1\), and subscripts \(I, II, III\) and \(IV\) indicate the type-I, -II, -III and -IV DWTs, denoted by DWT-I, -II, -III and -IV, respectively. Obviously, DWT-I is just the discrete Hartley transform (DHT) [4-6]. To coincide with the usually used notation, in this Letter DWT-I is still referred to as DHT.

Both the fast cosine transform (FCT) algorithm for the DWTs developed by Wang [1] and the fast polynomial transform (FPT) algorithm for the multi-dimensional DWT reported by Zeng [2] require the transform lengths to be a power of 2. Whereas with