Chapter 4

Time Reverse RSSE

Overview: *For Maximum Likelihood Sequence Estimation (MLSE) the Euclidean path distance, and correspondingly the bit error rate (BER) performance, remain unchanged when the channel coefficients are reversed. This invariance to channel reversal does not exist for Reduced State Sequence Estimation (RSSE), and the Time Reversed RSSE (TR-RSSE) is proposed to take advantage of this observation. TR-RSSE can provide significant performance improvements over conventional RSSE of similar complexity for a well defined and practically motivated class of channels.*

4.1 Introduction

GOOD PERFORMANCE can be obtained from Reduced State Sequence Estimation (RSSE) when equalizing channels whose channel impulse response (CIR) contains an energy peak at the start (essentially minimum phase channels). In this case, past decisions used in the branch metric calculation are reliable and the effect of errors will be minimized due to the low energy in the tail of the CIR.

At the other extreme, for channels that have their energy peak at the end of the CIR (post-cursor dominant), error propagation is exaggerated and performance of RSSE is significantly degraded relative to Maximum Likelihood Sequence Estimation (MLSE). Alternatively, to guarantee sufficient performance with RSSE, the complexity needs to be unacceptable high. Proposed in this chapter is a modification to RSSE that uses *time reverse* principles and processing to improve performance for the class of post-cursor dominant channels. Such channels occur, for example, when the line of sight (LOS) path between transmitter and receiver is blocked which can occur in a variety of situations. A wireless indoor local area network (LAN) can assume such a profile whenever the direct path is attenuated by obstructions relative to the reflected paths. Ariyavisitakul [54] first proposed a time reverse decision feedback equaliser (DFE) and the results in this chapter generalize this work to RSSE.

This chapter will first provide background information on the probability of error bounds. This will provide the basis to show that the performance of MLSE equalising a specific channel is identical to performance obtained when equalising a time reversed[1] version.

It will be shown that the property of performance invariance for MLSE under channel reversal does not hold for RSSE. This observation will be used to extend the class of channels that RSSE can be applied to. *Time Reverse RSSE* (TR-RSSE) is presented in Section 4.4 and applies RSSE techniques to post-cursor dominant channels.

4.2 Analytical Determination of Error Bounds

The probability of error for a Viterbi algorithms (VA) based equalizer can be easily determined by simulation. However, there are two limitations of this technique: firstly, no insight can be gained into sources of errors; and secondly, determining low bit error rates (even using importance sampling) is not practical. An analytical technique to determine bounds on the error probability was

[1] Channel reversal is when the first element of the CIR is exchanged with the last, the second with the second last, and etc.

presented in [1] and this will be followed here.

The analysis of errors in a trellis is based on *error events*. An error event occurs when the decoded path through a trellis differs from the actual path. The probability of each error event can be determined and by weighting and summing these probabilities an overall probability of error can be calculated, for a specified channel at a particular signal-to-noise ratio (SNR). Although this work is largely analytical, it still requires extensive computer searches.

4.2.1 Error Events

An error occurs when the estimated symbol in the equalizer disagrees with the transmitted (true) symbol. When this error occurs, an incorrect path has a lower path metric than the true path. An error event is the duration of this deviation. Alternatively, a series of correlated errors can be grouped into clumps, and those clumps are error events.

An error event that extends from k_1 to k_2 has the property,

$$S_{k_1} = \hat{S}_{k_1}, \; S_{k_2} = \hat{S}_{k_2}, \text{ and } S_k \neq \hat{S}_k \text{ for } k_1 < k < k_2. \tag{4.1}$$

The length of an error event is defined as $\varsigma = k_2 - k_1 - L$ and can be as short as 1 or possibly infinite. An error event can be represented as an error vector, defined by

$$\varepsilon = [\epsilon_{k_1}, \epsilon_{k_1+1}, \epsilon_{k_1+2}, \ldots, \epsilon_{k_1+\varsigma}]. \tag{4.2}$$

The elements of the error vector are defined by

$$\epsilon_j = (a_j - \hat{a}_k) \in \mathcal{E} \tag{4.3}$$

where $\mathcal{E}$ is the set of all valid error values. For example, for M level Pulse Amplitude Modulation[2] (PAM) $\mathcal{E}$ would be

$$\mathcal{E} = \{0, \pm 2, \pm 4, \ldots, \pm 2(M-1)\}. \tag{4.4}$$

[2] Pulse Amplitude Modulation is a generalization of BPSK. It maps the coded source symbol to a transmitted symbol containing one of M possible amplitudes.

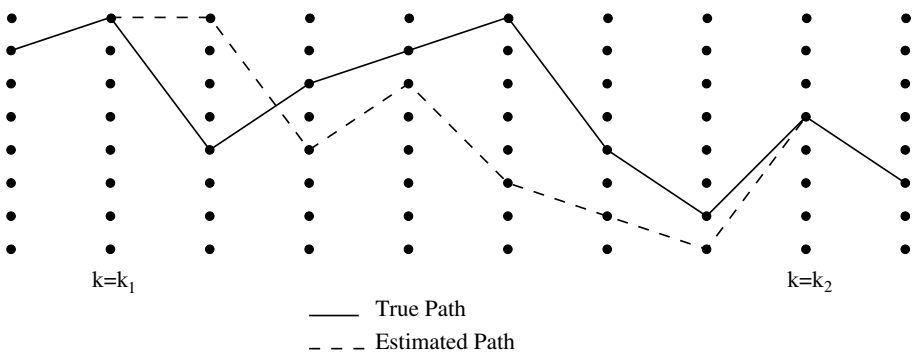

Figure 4.1: An error event extending from time k_1 to k_2 where the error vector is $\epsilon = [+2, -2, 0, -2]$.

Note, the last element of the error vector is not an index of time $k_2 - 1$. This is because there must be a sequence of length L (the channel length) of no errors in order for the paths to re-converge, i.e., for $S_{k_2} = \hat{S}_{k_2}$. This allows the statement of another property of error events, that is, there cannot be L consecutive zeros in an error vector as this would constitute two disjoint error events.

Figure 4.1 illustrates a single error event. The solid line represents the true sequence (the transmitted sequence) and the estimated path is shown as the dashed line. At time $k = k_1$ the paths diverge and do not re-converge until $k = k_2 = k_1 + \varsigma + L$ where $\varsigma = k_1 - k_2 - L = 4$. The error event for this example is $\varepsilon = [+2, -2, 0, -2]$ and, as indicated in (4.2), it does not include the zero error values which re-converge the paths.

A question can be proposed: How many types of distinct error events are there? Given infinite length transmitted sequences are possible there will be an infinite number of error events which are possibly infinite in length.

Given the multiplicity of error events (which are not equally likely) the closeness of error events to the true sequence is an important consideration. The Euclidean distance gives the appropriate measure [1] in the sense that the closer the error event (the smaller the Euclidean distance) is to the true sequence the more likely it is. Alternatively, the smaller the Euclidean distance the lesser the noise power required to force an error event. The Euclidean path distance, of an

error event ϵ, is defined by

$$d_H(\varepsilon) = \sqrt{\sum_{k=k_1}^{k_2-1} |d_k(\varepsilon)|^2} \tag{4.5}$$

where $d_k(\varepsilon)$ is the Euclidean branch distance and is defined by

$$d_k(\varepsilon) = \sum_{j=0}^{L} h_j \epsilon_{k-j}. \tag{4.6}$$

An important quantity which is often used to compare algorithms is the minimum Euclidean distance, $d_{\min}(\varepsilon)$. Given that the complete set of valid error events is denoted by $\mathfrak{E}$, the minimum Euclidean distance is defined by

$$d_{\min}(\varepsilon) = \min_{\epsilon \in \mathfrak{E}} d_H(\varepsilon). \tag{4.7}$$

Finally, the number of non-zero terms in an error event is given by the bit multiplicity or Hamming distance and is denoted and defined by

$$w_B(\varepsilon) = \text{ number of non-zero coefficients in } \epsilon. \tag{4.8}$$

In summary, an error event is characterized by the following properties:

- $S_{k_1} = \hat{S}_{k_1}, S_{k_2} = \hat{S}_{k_2}$, and $S_k \neq \hat{S}_k$ for $k_1 < k < k_2$
- The length of an error event $\varsigma = k_2 - k_1 - L$.
- There cannot be L consecutive zeros.
- Equation (4.5) defines the Euclidean path metric which gives a useful (as will be shown next) measure for the error event.

4.2.2 Probability of a Single Error Event

The probability of an error event is now considered. For any error event three sub-events must occur:

$\mathbb{E}_1$: At time k_1, it must be true that $S_{k_1} = \hat{S}_{k_1}$.

$\mathbb{E}_2$: Between k_1 to $k_1 + \varsigma$ the input sequence, a_k must be such that $a_k - \epsilon_k$ produces an allowable sequence. For example, if $a_k = +1$ and $\epsilon_k = +2$ then $\hat{a}_k = -1$ is allowable, however, if $\epsilon_k = -2$ then $\hat{a}_k = +3$ is not allowable.

$\mathbb{E}_3$: The noise terms η_k, for $k_1 \leq k < k_2$ must be such that over this segment the estimated path has a greater likelihood than the true path.

Determining the probability of $\mathbb{E}_3$ first. This condition states that the path metric of the estimated path, $\Lambda(\hat{S})$, is lower than the true path $\Lambda(S)$, hence,

$$\Lambda(S) > \Lambda(\hat{S}) \text{ for the duration } [k_1, k_2). \tag{4.9}$$

As the noise sequence is white-Gaussian, $\mathcal{N}(0, \sigma_\eta^2)$, then the natural log of the probability density function is

$$\ln p(\eta_k) = -\frac{1}{2} \ln 2\pi\sigma_\eta^2 - (y_k - C(S_k))^2/2\sigma_\eta^2, \tag{4.10}$$

where $C(S_k) = \sum_{j=0}^{L} h_j a_{k-j}$. Extending (4.9) by incorporating (4.10) then

$$\Lambda(S) - \Lambda(\hat{S}) = \frac{1}{2\sigma_\eta^2} \sum_{k=k_1}^{k_2-1} \left[(y_k - C(\hat{S}_k))^2 - (y_k - C(S_k))^2\right] > 0 \tag{4.11}$$

where $C(\hat{S}_k) = \sum_{j=0}^{L} h_j \hat{a}_{k-j}$. Equation (4.11) states that $\hat{a}_k$, over the period $k \in [k_1, k_2)$, an error is more likely if $C(\hat{S}_k)$ is closer to y_k than $C(S_k)$.

The probability of $\mathbb{E}_3$ is the probability that a single Gaussian variable of variance σ_η^2 exceeds half the distance between $C(S_k)$ and $C(\hat{S}_k)$, which is half the squared Euclidean path distance

$$d_H^2(\varepsilon) = \sum_{k=k_1}^{k_2-1} \left[C(S_k) - C(\hat{S}_k)\right]^2. \tag{4.12}$$

Consequently,

$$\Pr(\mathbb{E}_3) = \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \int_{d_H(\varepsilon)/2}^{\infty} \exp\left(-\eta^2/2\sigma_\eta^2\right) d\eta = Q\left(\frac{d_H(\varepsilon)}{2\sigma_\eta}\right), \tag{4.13}$$

where $Q(\cdot)$ is defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}\sigma} \int_x^{\infty} \exp\left(-y^2/2\sigma^2\right) dy. \tag{4.14}$$

Sub-event $\mathbb{E}_2$, which states that $\hat{a}_k$ must be an allowable sequence is dependent on the transmit alphabet and is independent of $\mathbb{E}_1$ and $\mathbb{E}_3$. This probability is different for all modulation schemes and it reflects permitted error values. For M level PAM then

$$\Pr(\mathbb{E}_2) = \prod_{i=0}^{\varsigma-L} \frac{M - |\epsilon_i|/2}{M}. \tag{4.15}$$

The longer the error event, the smaller $\Pr(\mathbb{E}_2)$ which intuitively state that they will be less likely to occur compared to shorter error events. This probability will be referred to as the *error multiplicity* and be denoted by $w_E(\varepsilon) = \Pr(\mathbb{E}_2)$.

> *Example IV.I:* For BPSK where $M = 2$ and given the error event of Figure 4.1, $\epsilon = [+2, -2, 0, -2]$. The bit multiplicity is $w_B(\epsilon) = 3$ and error multiplicity is simply, $w_E(\epsilon) = (\frac{1}{2})^{w_B(\epsilon)} = 0.125$.

Sub-event $\mathbb{E}_1$ is not easily computable as it is dependent on the noise term in $\mathbb{E}_3$. However, the probability that $\mathbb{E}_1$ is not meet is of the order of the error probability and can be over-bounded by unity.

Bringing these results together shows the probability of an error event can be bounded by

$$\begin{aligned}\Pr(\epsilon) &= \Pr(\mathbb{E}_1/\mathbb{E}_3)\Pr(\mathbb{E}_2)\Pr(\mathbb{E}_3) \\ &\lesssim \left[\prod_{i=0}^{\varsigma-L} \frac{M - |\epsilon_i|/2}{M}\right] \cdot Q\left(\frac{d_H(\varepsilon)}{2\sigma_\eta}\right).\end{aligned} \tag{4.16}$$

4.2.3 Probability of Error

Using the presented material, the probability of a particular error event can be determined. This is a step towards the overarching objective of determining the probability of a symbol error at a given SNR. This can be achieved by combining the individual error event information and, for moderately small error rates, a tight bound (with in .5 to 1dB) will result.

The union bound states that the probability of the union of events is not greater than the sum of their individual probabilities. Hence, the probability of

an error event occurring is bounded by

$$\Pr(E) = \Pr(\varepsilon_1 \cup \varepsilon_2 \cup \varepsilon_3 \cup \dots) \leq \sum_{\varepsilon \in \mathfrak{E}} \Pr(\varepsilon). \tag{4.17}$$

The probability of a symbol error, P_{se}, can be bounded by weighting each of the error events by the bit multiplicity. This leads to

$$\begin{aligned} \mathrm{P}_{se} &\leq \sum_{\varepsilon \in \mathfrak{E}} w_B(\varepsilon) \Pr(\varepsilon) \\ &\leq \sum_{\varepsilon \in \mathfrak{E}} w_B(\varepsilon) \cdot w_E(\varepsilon) \cdot Q\left(\frac{d(\varepsilon)}{2\sigma_\eta}\right) \end{aligned} \tag{4.18}$$

which is an upper bound on the true probability of error. A lower bound can be generated by using the minimum Euclidean path distance, as defined in (4.7), following the standard arguments given in [1, 55]. Calculating both the upper and lower bound gives an indication on how tight the bounds are to the true probability of error. There have been other authors who have suggested techniques for tightening the upper or lower bounds, such as Verdú [56], Swaszek [57] and Lapidoth [58].

It is not practical to determine all error events as the set is infinite. However, the probability of these very long error events occurring is negligible, consequently, the set can be truncated with very little effect. The techniques for determining the error events and their parameters is discussed in Appendix A.

4.3 Time Reversal MLSE

The bit error rate (BER) performance of MLSE depends on the channel characteristics. The best performing, albeit trivial, time dispersive channel is an ISI free one. At the other extreme, the real CIR of a given length with the worst performance is almost flat (in the sense that the impulse response has roughly the same magnitude - their zeros lie close to the unit circle.) [19, p. 601]. While the performance varies for different channels, it will be shown that the BER performance of MLSE is invariant to the time reversal of a channel.

Formally, a vector $P = [p_0, p_1, p_2, \ldots, p_m]$ is said to be *time reversed* if the order of the elements is reversed. That is, the first element is exchanged with the last, the second exchanged with the second last element, etc. Time reversal is denoted by an over-line, i.e., $\overline{P} = [p_m, \ldots, p_1, p_0]$.

The Euclidean distance will be used to support the argument that the performance is invariant to the order of the channel coefficients. Consider any arbitrary channel impulse response, H, the squared Euclidean distance for an arbitrary error event, ε, referring to (4.5), is

$$d_H^2(\varepsilon) = \sum_{k=k_1}^{k_2-1} \left| \sum_{j=0}^{L} h_j \epsilon_{k-j} \right|^2. \tag{4.19}$$

Then consider the following equations which determine the distance for a time reversed channel and time reversed error event:

$$\begin{aligned} d_{\overline{H}}^2(\overline{\varepsilon}) &= \sum_{k=k_1}^{k_2-1} \left| \sum_{j=0}^{L} \overline{h}_j \overline{\epsilon}_{k-j} \right|^2 \\ &= \sum_{k=k_1}^{k_2-1} \left| \sum_{j=0}^{L} h_{L-j} \epsilon_{k-L+j} \right|^2 \\ &= \sum_{k=k_1}^{k_2-1} \left| \sum_{j=L}^{0} h_j \epsilon_{k-j} \right|^2. \end{aligned} \tag{4.20}$$

Therefore,

$$d_{\overline{H}}^2(\overline{\varepsilon}) = d_H^2(\varepsilon). \tag{4.21}$$

It has been shown that the distance of a given error event is the same as its time reverse. Given $\mathfrak{E}$ is the set of all valid error events, then the distance spectra (which is a graph of distance verses error multiplicity) for H and $\overline{H}$ are identical because

$$\varepsilon \in \mathfrak{E} \Leftrightarrow \overline{\varepsilon} \in \mathfrak{E} \tag{4.22}$$

where $\Leftrightarrow$ implies if ε is in the set of error events, $\mathfrak{E}$, so it $\overline{\varepsilon}$.

4.4 Time Reverse RSSE

For the MLSE case, performance was shown to be invariant to time reversal of the channel. However, it will be shown next, the performance for RSSE is not invariant to time reversal of the channel. Reverse time processing has the potential to provide improved performance for RSSE operating on non-minimum phase channels.

Firstly further terminology must be introduced to describe the channel. Channels that contain most of the energy at the end of the impulse response are referred to as post-cursor dominant (non-minimum phase). Likewise, pre-cursor dominant channels contain most of their energy at the start of the impulse response. General channels will be defined as channel that have mixed phase, that is, where neither pre or post-cursor dominant and the energy peak exists in the middle of the CIR.

Conventional RSSE operates well on pre-cursor dominant channels, but, for a given complexity, tends to perform poorly for post-cursor dominant channels. To retain low complexity and apply RSSE to these channels, time reversal principles can be employed. Ariyavisitakul [54] used a DFE with time reverse structure to implement a receiver for wireless indoor LANs. This section extends that work by broadening the application from the DFE to RSSE. *Time Reverse RSSE* (TR-RSSE) is basically a conventional RSSE operating with a reversed CIR.

Before applying the technique of the previous section to RSSE, consider the following justification. Determining the probability of error for RSSE is not identical to MLSE as pointed out by Sheen [47]. Errors, unique to RSSE, are due to error propagation and early survivor merging. Error propagation occurs when incorrect decision are made which are then used via decision feedback to calculate the path metric which can possibly force the occurrence of additional errors. The number of states are reduced in RSSE compared to MLSE, hence, when an error does occur, less time is required for the error path to re-converge to the true path. This is early merging. This reduces the Euclidean path distance between the true and error paths, hence, degrading performance of RSSE. Early merging is inherently dealt with when determining error events for the reduced trellis, however,

error propagation is very difficult to analyse. Sheen suggests that for error propagation to be accommodated then the arguments of the $Q(\cdot)$ function need to be modified (see equation (14) of [47]). Luckily, the effects of error propagation is generally small, especially for systems operating at a moderate to high SNR (which is the normal region considered to produce tight bounds). Consequently, (4.18) can be used to determine adequately the probability of error for RSSE and the procedure of the previous section holds as a reasonable approximation for RSSE.

Increasing $d_{\min}(\varepsilon)$ generally implies improving performance. By containing the majority of the channel energy in the reduced state, $d_{\min}(\varepsilon)$ will be large and for post-cursor dominant channels this is what TR-RSSE attempts to achieve. The performance of TR-RSSE will be illustrated by an example. Consider the expected performance for normal RSSE and TR-RSSE, each containing a 2 state trellis, for the channel $H = [0.324, 0.583, 0.745]$. The minimum Euclidean distance differs between the normal case, $d_{\min}(\varepsilon) = 0.4453$, and time reversed case, $d_{\min}(\overline{\varepsilon}) = 0.8052$. The complete spectrum for time reversed and normal case can be observed in Figures 4.2b and 4.2c respectively. This lowering of minimum distance compared to MLSE (with $d_{\min}(\varepsilon) = 0.8679$ and the distance spectra is shown in Figure 4.2a) points to a degradation of performance, particularly for the normal RSSE case. The following paragraphs will formally state the TR-RSSE by defining: the state definition; metric calculations; and the received data sequences.

The received sequence must be time reversed via a blocking method. That is, N received symbols are stored in a block and the algorithm operates over the period $k_1 + N$ to k_1 with the time reversed channel. The received symbols that the algorithm operates with can be described by

$$\check{y}_k = y_k = \sum_{j=0}^{L} \overline{h}_j a_{k-L+j} + \eta_k \qquad \forall k \in [k_1 + N, k_1]. \tag{4.23}$$

The number of states are selected in the same way as it is for RSSE, i.e., a performance/complexity trade-off is considered in selecting l the channel partition

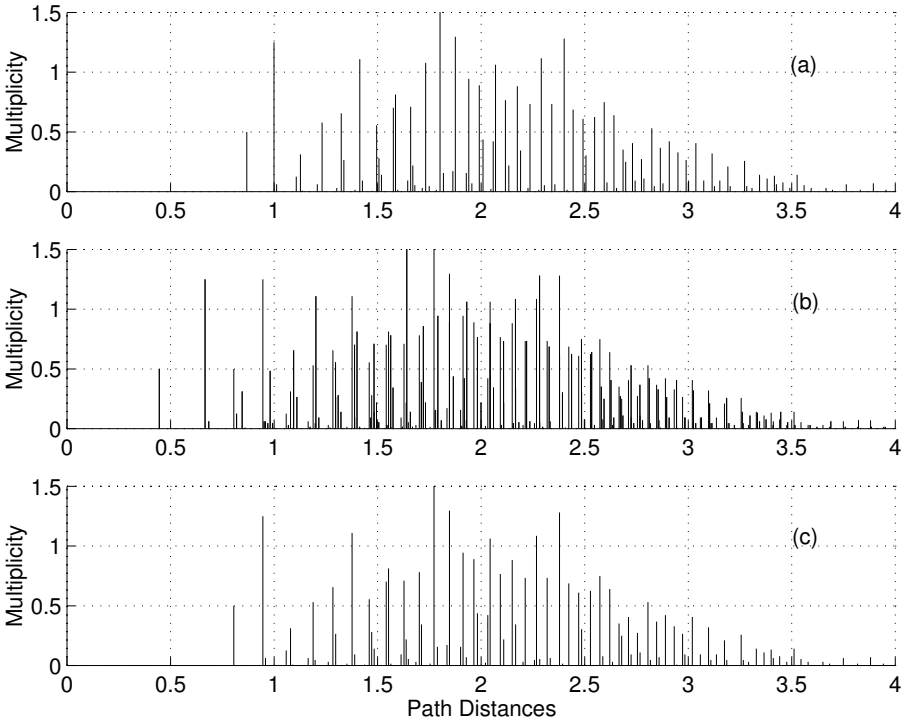

Figure 4.2: The Euclidean distance spectra (which is a plot of the error multiplicity versus Euclidean path distance) for (a) 4-state MLSE, (b) conventional 2-state RSSE and (c) 2-state time reversed RSSE for the channel $H = [0.324, 0.583, 0.745]$.

parameter. The states for the algorithm are defined according to

$$\check{S}_k = [a_{k-L+1}, a_{k-L+2}, \ldots, a_{k-L+l}]. \tag{4.24}$$

The path metric calculations are performed in the same way as for the normal RSSE, that is, the path metric is given by the previous path metric and the branch metric. The two paths leading to a node are compared and the one with the lower path metric selected. The branch metric, however, must be modified from the standard RSSE branch metric ((2.37) in Chapter 2) to take into account the time reversed nature, as follows

$$\lambda\big(\check{y}_k, a_{k-L}, \check{S}_k, \hat{\mathcal{S}}^k_{k_1+N}(\check{S}_k)\big) = \left|\check{y}_k - \bar{h}_0 a_{k-L} - \check{S}_k \cdot (\bar{H} \cdot \Psi)' + \big(\hat{\mathcal{S}}^k_{k_1+N}(\check{S}_k) \cdot \Upsilon\big) \cdot \bar{H}'\right|^2 \tag{4.25}$$

where Ψ is a $L \times l$ unit subdiagonal matrix and Υ is a $|\hat{\mathcal{S}}^k_{k_1+N}(\check{S}_k)| \times (L-l)$ unit diagonal matrix and are defined next. The unit subdiagonal matrix, Ψ, is defined as $\psi_{i,i-1} = 1 \;\; \forall i \in [1, l]$ and all other entries are zero. The unit diagonal matrix, Υ, is defined as $\upsilon_{i,i} = 1 \;\; \forall i \in [0, l-L]$ and all other entries are zero. Note that the symbol estimates are extracted from the equalizer in a time reversed order, hence, they must also be blocked and reversed prior to passing on to the next stage of the receiver.

Equations (4.23) to (4.25) describe the operation of the TR-RSSE. Alternatively, it can be thought of as a conventional RSSE with the received (input) sequence, channel and estimate (output) sequence all reversed. A straight forward but significant identity is

$$d_{\min}(\varepsilon)(\text{using TR-RSSE}) = d_{\min}(\overline{\varepsilon})(\text{using conventional RSSE}). \tag{4.26}$$

For the limiting case for $l = 0$, the proposed structure reduces to that presented by Ariyavisitakul [54], a time reversed DFE.

Ensuring $d_{\min}(\overline{\varepsilon}) > d_{\min}(\varepsilon)$ is a suitable criterion for selecting when the TR-RSSE should operate. This calculation may not be practical: it is just an indication of when the TR-RSSE outperforms RSSE.

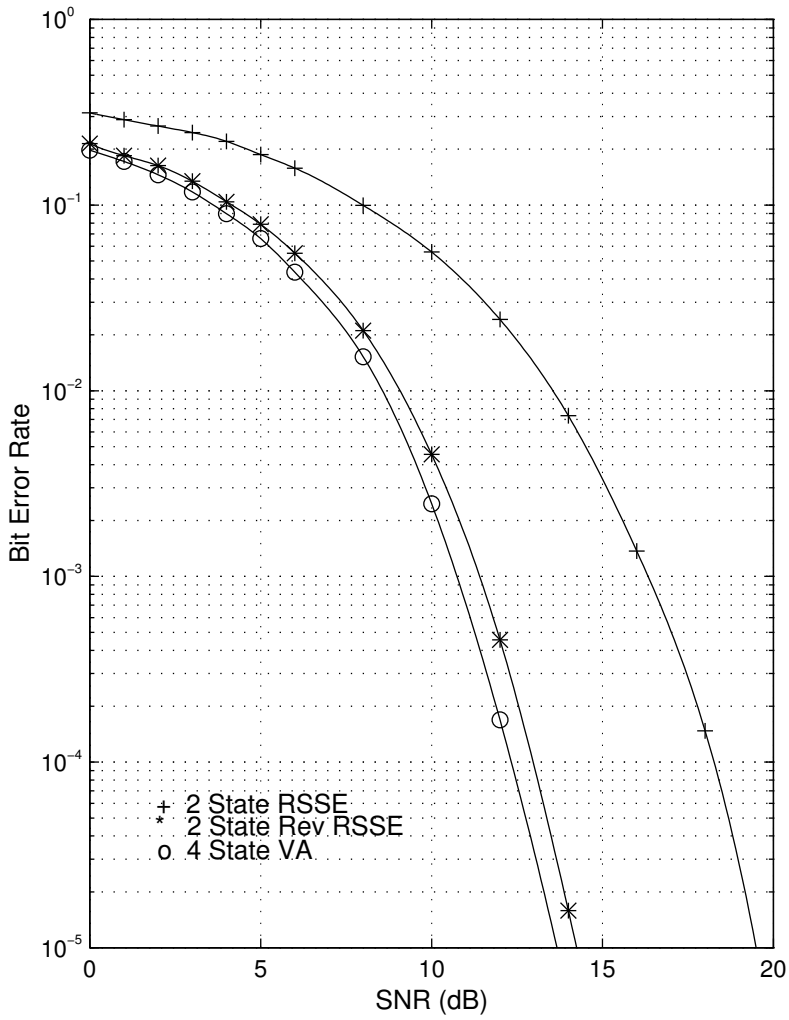

Figure 4.3: Bit Error Rate comparison between 4-state MLSE, conventional 2-state RSSE and 2-state time reversed RSSE for the channel $H = [0.324, 0.583, 0.745]$.

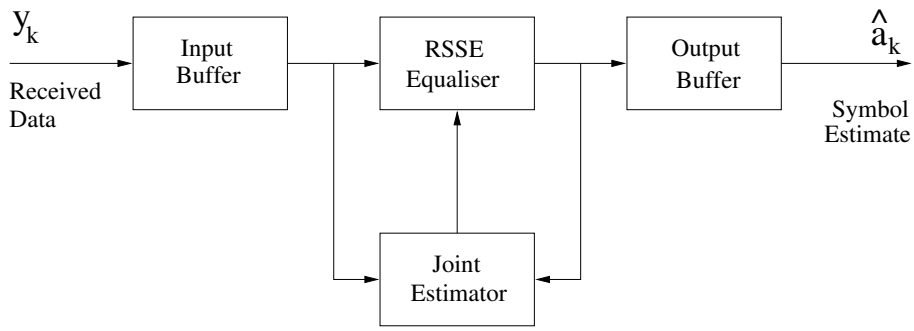

Figure 4.4: A suggested receiver structure to perform TR-RSSE.

4.4.1 Simulation

For the case of post cursor dominant channels, the TR-RSSE exhibits an improved BER versus SNR over the normal RSSE due to the greater minimum Euclidean distance. Simulations were performed as shown in Figure 4.3. The channel considered was $H = [0.324, 0.583, 0.745]$ and the simulations were performed for 4 state MLSE and 2 state RSSE and TR-RSSE. For this example TR-RSSE outperforms conventional RSSE by 5.2dB at a BER of 10^{-5}.

4.4.2 Implementation for Continuous Operation

For the TR-RSSE to equalize continuous data, blocks, of a specific length N, are used to reverse the order of the data by a last in first out (LIFO) buffering procedure. As for conventional RSSE, estimates of the transmitted sequence are extracted from the converged survivor sequences which necessitates the blocks to be overlapped. Also, as a block must be filled with data prior to processing, the blocking procedure introduces latency. Hence, the selection of N is arbitrary and must be selected according to the application. For N selected large, latency would be long and large amounts of memory would be block the data. With this case, however, the additional computations due to overlapping would be minimal. At the other extreme, short block lengths have low latency but relatively high overheads due to overlapping. Implementation issues relating to blocking of data will be further discussed in Chapter 5.

Figure 4.4 shows a general block diagram of a possible receiver design. Buffer-

ing is required at the input to allow the time reversal to be performed. The joint estimator determines factors such as phase and amplitude compensation, channel estimation, and symbol arrival times. Final output buffering is required to reverse the symbol estimates.

4.5 Summary

The performance of MLSE is invariant to channel reversal. This invariance was analyzed using error events and occurs because the state definition of MLSE involves the complete CIR.

By comparison, RSSE does not have this invariance property. Pre-cursor dominant channels are typically well equalized using standard RSSE, however, poor performance is often observed for post-cursor dominant channels with low complexity RSSE. Thus, the TR-RSSE was proposed to extend RSSE techniques to a wider class of channels. It is basically a conventional RSSE that operates in reverse time; so that a post-cursor dominant CIR becomes a pre-cursor dominant CIR. This technique typically outperforms RSSE if the majority of CIR energy is present in the time reversed reduced state. The minimum Euclidean distance can act as a guide to accessing the improvement.

To implement the TR-RSSE, the received data and estimated data is blocked and reversed. The block length needs to be selected carefully as the equaliser could have unacceptable latency. For one example, TR-RSSE outperformed conventional RSSE by 5.2dB at a BER of 10^{-5}. This new TR-RSSE compliments conventional RSSE, and allows a greater class of channels to be equalized using RSSE type techniques.