Lattice-based equalization algorithms for an underwater acoustic modem

José Sérgio Costa Vilela  
Instituto Superior Técnico  
Lisboa, Portugal  
jose.sergio.vilela@tecnico.ulisboa.pt

Abstract—Telecommunications are increasingly vital as our society is more and more connected. In endeavors such as off-shore oil drilling and tsunami prevention underwater acoustic telecommunications are better suited due to the high level of attenuation that the media bring to common radio frequency transmissions. Besides attenuation the underwater channel also brings other detrimental phenomena like multipath, which causes Inter Symbol Interference (ISI), making the received signal exceedingly different from what was initially transmitted. In order to compensate for this issue equalization must be utilized.

It is then clear that the algorithms used for equalization are critical, which may suggest an approach of utilizing the most robust and optimal one in order to obtain a clear signal. However, as often happens in engineering problems the perfect solution is not the most useful given the trade-offs that must be made when receivers are developed. One of the most common factors that one must focus is the complexity that algorithm introduces in the problem, as this directly influences the amount of operations needed to obtain the data and the battery power needed.

This thesis studies the Least Squares Lattice (LSL) algorithm with objective of understanding if its a viable alternative to the classical Recursive Least Squares (RLS) algorithm given that it has properties that are advantageous to a more efficient equalization, namely its modularity and linear complexity. The results obtained for the two LSL approaches studied suggest that it is possible to make use of its properties as the expected trends of lower errors with increase of modules were detected. While further study into this topic is needed the research presented here is a stepping stone for future endeavors into equalization in underwater acoustic channels.

I. INTRODUCTION

The study of underwater communications is relatively recent having around a century of lifetime.

Underwater communications have been evolving through the last few years, especially since World War I with the development of technologies capable of higher data rates and higher quality. This area of study has a variety of applications in practical fields such as scientific research, Unmanned Underwater Vehicle (UUV), underwater surveillance and disaster prevention. The DART II tsunami detection system currently in use relies on acoustic communications between an acoustic modem located on the seafloor with a surface buoy and is a good example of a useful and life-saving use case of this technology.

The study of underwater communications is in many ways mirrored by the study of communications in Radio Frequency (RF) over the air with many of the phenomena in study being either similar or the same. Yet, the nature of the underwater channel poses challenges which are not as easy to address as they are when the signal is propagating in the atmosphere. Data rates within the kilobytes range, and attenuation due to the channel, which is almost always considerably higher than in RF, are examples of the challenges when operating at the sea.

Throughout the years the study of the underwater channel has evolved towards obtaining a better characterization of this medium with the objective of developing technologies capable of improving the rate of information transfer at sea. These studies can then give better insight into the development of underwater communication systems modules such as Modulation and Equalization. Given that the underwater acoustic channel is considered one of nature’s most unsavory wireless communication media[1], it is quite important to minimize all undesirable factors such as ISI. The study of equalization is important considering that it can reduce the above mentioned ISI generated by the channel and ultimately obtain a better quality signal.

There are several approaches to the development of equalizers, most of which are based on criteria such as the Least Mean Squares (LMS), or the RLS. This thesis will study the approach based on algorithms of the RLS family, specifically, lattice-based ones whose computational complexity scales linearly with the problem order, as opposed to the basic RLS algorithm whose complexity scales quadratically.

II. BACKGROUND AND LITERATURE REVIEW

In this section, in a first instance, a description of the Underwater Acoustic Channel and its main phenomena will be given. Secondly, in light of the scope of this study, a review of the evolution of equalization techniques and algorithms will be provided with a special focus the algorithms studied in this thesis.

A. Underwater Acoustic Channel

The Underwater Acoustic Channel (UAC) is a wireless communications channel and as such it shares many of the properties and phenomena that other wireless channels exhibit. However, many of these effects are more severe than in other media or are not as economically viable to solve. Phenomena such as losses due to absorption, scattering and spreading, which are discussed in Sections II-A2 to II-A4, are shared with the popular (atmospheric) wireless radio
channel. While some of the effects mentioned are common, the solutions used to mitigate or even neutralize them may or may not be reused. As an example, a possible solution to the channel absorption loss can be an increase in input power which, if high enough, will make the loss acceptable to the communication system designed. Although this solution can easily solve the issue in a point to point air atmospheric radio connection, where absorption losses are around $10^{-3}$ dB/km for a frequency of 1GHz, the same can’t be said for the UAC where at an acoustic frequency of 100KHz the attenuation can be as high as 50dB/km.

With this situation in mind it is clear that the UAC should be carefully studied before any implementation is attempted, in order to get a better understanding of future results. The current state of the art of the UAC is presented in the following sections and much more thoroughly reviewed in [2], [3], [4].

1) Underwater Propagation: For purposes of propagation the ocean can be considered as an acoustic waveguide which is limited by both the sea surface and the seafloor. As in other known waveguides the wave speed (in this case sound) is one of the defining parameters of the medium. The most well known parallel to this situation is the refractive index used in optics which is the index between the speed of light in vacuum and the speed of light within the medium. Shallow regions, which are areas in the continental shelf, have depth on the order of a few hundred meters and consequently require a good characterization of the waveguide considering the increasing possibility of both bottom and surface interactions which are almost always undesirable sources of either signal loss or temporal variability. Most of the times the sea surface is considered a simple perfectly reflecting boundary, while the seafloor is considered a lossy boundary with strongly varying topography[5]. Absorption loss is seen in the bottom of the sea and in the sea itself, while scattering is only seen in the waveguide limits. The following sections discuss the main loss phenomena in greater detail.

2) Absorption Loss: As was briefly stated in Section II-A1 the absorption loss can be seen in the seawater. Beginning with the loss caused by the sea itself, it is important to mention that it is an effect deeply dependent on frequency and that it is one of the main factors of limitation of the channel bandwidth, with attenuation reaching 50dB/km at a frequency of around 100kHz.

3) Scattering Loss: Scattering effects can be explained as added attenuation to the specularly reflected component. This means that when the acoustic energy is reflected by one of the ocean boundaries there are components that are not reflected in the same direction as the specularly reflected component and never reach the receiver therefore making them lost.

This reflection may also manifest as acoustic reverberation which is a problem given that the echoes can confuse the measurements in the receiver.

As with the absorption loss, the scattering loss grows with frequency.

4) Spreading Loss: The spreading loss is a measure of how much signal propagates from the source to the receiver, as the associated wavefront spreads its energy over an increasingly large surface. This can be studied in two different geometries, spherical and cylindrical. Spherical spreading is considered in the near-field region of the source, while cylindrical geometry is achieved only at longer distances, corresponding to propagation in a waveguide limited by the sea surface and the sea floor.

5) Ambient Noise: Beyond the factors mentioned in the previous sections there is another important effect that needs to be mentioned: ambient noise. Acoustic masking, which is the phenomenon that may cause problems at the receiver, occurs when the presence of one sound (noise) reduces the ability of perceiving a second sound (of interest).

The ocean has a lot of sources of noise such as marine life, passing ships, breaking waves and even rain [1]. From the examples mentioned above it can be seen that noise sources can be divided in two categories: man-made and natural. The first one has been growing in the last few decades with the increasing naval traffic coupled with the growing use of offshore rigs resulting in an increase of at least 20 dB in ambient noise levels compared to pre-industrial levels, while the second one has been more or less constant.

6) Doppler effect: The Doppler effect (sometimes called Doppler shift), is the change in frequency perceived by the receiver due to the combination of the possible movements of the transmitter, receiver and the medium itself.

Considering that for the UAC typical carrier frequencies are on the order of $f_c = 10^4$ Hz , whereas $v$ may be around $1 - 2 m/s$ and $c$ is close to 1500 m/s, we get Doppler shifts of a few Hz. These manifest themselves as rotations of the signal constellation that are clearly noticeable on time intervals spanning even a few tens of symbols at typical signaling rates, and must be compensated in order to get reliable communications. The key to the success of the now classic Decision Feedback Equalizer (DFE)-Phase-locked Loop (PLL) receiver architecture [2] was the realization that Doppler-induced time variations are highly structured and are best handled by a specialized component (the PLL), which greatly removes the tracking burden from the less agile equalizer. Although the exact PLL of [2] cannot be used in the lattice implementation of the DFE, where the outputs of the feedforward and feedback filters are not separable, a comparable component will be derived.

B. Evolution of Underwater Equalization

The UAC makes channel equalization a challenging problem due to its inherent properties which, in turn has stimulated interest in this topic in the past few decades.

UACs in shallow waters may exhibit long delay spreads due to numerous multipath arrivals, and high Doppler spreads caused by a variety of phenomena such as movement of transducers and internal waves[4]. These two spreads cause ISI and strong phase fluctuations of the signal which led the design of early systems to focus on non-coherent detection and low signaling rates[2]. To reduce ISI, non-coherent systems employ guard times between successive pulses, reducing the
already small data throughput to around 1 kbit/s. It can be concluded that the inefficient use of bandwidth, coupled with the limited availability of bandwidth underwater, makes incoherent systems unsuited for high data rate transmissions such as images[3].

Considering the increasing need of bandwidth efficiency it was necessary to design systems based on coherent modulation techniques such as Phase Shift Keying (PSK) and Quadrature Amplitude Modulation (QAM). To increase performance in these systems it was also necessary to implement equalizers which can, if well designed, reduce the Bit Error Rate (BER) significantly.

The coupling of PLL with equalization is a solution that first appeared in order to deal with time-varying ISI and residual phase fluctuations together, whereas previously these problems were normally dealt with separately. This approach was theoretically acceptable due to the fact that joint estimates are always at least as good as the marginal ones[6]. Initially the approach used was based on a Maximum Likelihood optimal receiver, but its complexity stimulated the search for a new receiver design that would benefit from joint estimates while decreasing the complexity as much as possible.

In [6] the algorithm for one of the most commonly known underwater DFE receivers was introduced. This architecture has influenced many designs for the underwater acoustic channel, including the approach in this thesis.

In the Adaptive filter part of the receiver there are two main families of algorithms used to update the tap coefficients, LMS and RLS.

In [2] several examples of LMS algorithm are mentioned with the systems where it is implemented having data rates within the 15 kb/s range for vertical path propagation. Still in [2] an example of a system using Differential Phase Shift Keying (DPSK) and operating with LMS in a 1km long, 10m deep channel is given with a data throughput of 600 b/s.

The relatively low data rates achieved through these early implementations based on LMS gave rise to study of other algorithms with the objective of getting better system performance. By 1996 the latest development in coherent based systems was a prototype implemented by the Woods Hole Oceanographic Institution (WHOI) in which ISI was processed using a DFE operating under an RLS algorithm reaching a data rate of 5kb/s [2].

As mentioned before, one of the most common strategies to build a coherent digital receiver has been the insertion of PLL and DFE blocks and further optimization of these blocks algorithms’ complexity. The DFE is usually composed of a set of conventional feedforward taps that sample the receiver pressure signal, a set of feedback taps that provide previous symbol decisions as well as introduce non-linearity into the filtering[3]. The PLL is then used to relieve the tap weight adaptive algorithm of the phase-tracking task by taking care of the generally most prominent time-varying feature of the signal which is the mean variable Doppler shift[3].

One of the first approaches used was based on the LMS algorithm as evidenced by several references to it in [2]. This was due to its linear complexity which is a desirable trait in these type of systems. However, its convergence time can become quite long when large adaptive filters are used (five times longer than RLS) and that is especially important considering that shallow channels tend to need a high amount of taps for its processing.

As mentioned earlier RLS has better convergence than LMS but the complexity of its standard version is quadratic, which is generally too high for large adaptive filters[2] creating the need to use less stable but linearly complex solutions. These are characteristics shared by most fast RLS algorithms developed up to date but even though standard RLS has quadratic complexity, a square-root RLS algorithm has been developed and implemented with its main advantages being excellent numerical stability and fewer periodical updates for receiver parameters which implies lower computational load per each detected symbol. This adaptability is particularly interesting for use in high transmission rates due to the need for large adaptive filters to compensate high ISI, which in turn decreases the time interval between transmitted signals and consequently eliminates the need to update parameters every symbol interval.

One study cited in [3] made a comparison between LMS and RLS strategies having found no steady-state advantage between them and another has compared a fast RLS with the classic LMS and a variation of LMS having concluded that the fast RLS was numerically unstable while the strategies based on LMS gave similar results.

One topic that has generally been considered important in research within this field is the search for algorithms that can autonomously initialize themselves without human intervention[3]. Since equalization is a very intricate problem with many sensitive parameters such as step factors, forgetting factors and filter support it is necessary that the selection of these variables be as good as possible in order to get better performance.

Lattice filters that use RLS have only been barely mentioned in the widely cited survey[2], making this thesis’ theme relevant to the growing knowledge in this field.

C. Advances in Adaptive Equalization for the UAC

1) QR-RLS Multichannel Lattice Equalization: First a brief introduction to lattice and its main characteristics will be made. This type of filter has a different working process compared to the commonly used transversal filters. The main thing to be aware in these two architectures is that to increase the order of transversal filters there is the need to recalculate all tap coefficients while in a lattice scheme only the parameters corresponding to the increased order must be calculated. This is possible due to this algorithm backward prediction errors being uncorrelated with each other.

The finding of a link between Kalman filter equations and RLS, and the realization that the many single channel RLS lattice and transversal algorithms could be written as specific cases of Kalman filter equations was made by Ali Sayed during his PhD and later published in [7]. This previous work and
the framework developed in it was critical to find a simpler multichannel case.

The algorithm proposed in [8] makes use of a complex givens rotation to pass from the prearray to the postarray which is actually a variant of the unnormalized a priori lattice[8].

In [8] the authors make a comparison between RLS, QR-RLS, and two other multichannel lattice algorithms and the results obtained show that QR-LSL is more robust to the absence of noise than RLS while managing to obtain Mean Square Error (MSE) very similar to the latter.

Considering these findings, the relevance of the study of these type of algorithms is more than justified given the fact that QR-type algorithms avoid complex inversion operations and lattice has characteristics such as modularity and linear complexity that are beneficial to systems requiring better performance.

2) Channel-Estimation-Based Adaptive Equalization: The canonical DFE mentioned in Section II-B is a common design for equalizers in wireless channels. However, it has the consequence of needing to train both the feedforward and feedback filter taps. In [9], the concept of using Channel Estimation is investigated with the objective of training only the feedforward taps during the equalization process while the feedback taps are trained by a separate mechanism, therefore gaining a reduction of complexity in the equalization with the added benefit of having a faster start-up. The results obtained in [9] motivated the study of this type of equalization in the specific case of the UAC in [10].

The use of explicit channel estimation equalization has the advantages mentioned before but can also bring new avenues of further reduction of complexity. There are two main approaches that can be taken to tackle this challenge:

- Better and more efficient algorithms.
- Reduction of the number of parameters in the receiver that need to be adjusted adaptively.

The article [10] focuses on the last one, specifically in reducing the size of the adaptive equalizer through sparsification. The objective of this technique is to use only the filter taps where the majority of the error energy is.

In [10] the authors developed an algorithm for multichannel equalization based on a different interpretation of the DFE. In this case the equivalent feedforward signal has to be calculated first, by subtracting the equivalent feedback signal from the input signal. The equivalent feedback signal is obtained from the previous decisions and the channel estimate.

Two major advantages of this approach are the fact that channel estimation is decoupled from equalization, the fact that the channel estimate may be extended to include distant multipath arrivals without influencing performance as a classical DFE would[10].

Underwater communications tend to not be reliable in a Single Input Single Output (SISO) architecture and the use of multiple receivers is common with the approach of [8] being used as a framework for this thesis.

III. ADAPTIVE EQUALIZATION THEORY

A. Least Squares Lattice Predictors

The main interest in this family of algorithms is that its complexity increases linearly with the number of adjustable filter parameters which in itself deals with one of the main disadvantages of simple RLS and gives adaptive filters a computationally efficient and modular structure[11]. One of the direct consequences of the order update property of these algorithms is the fact that should the need arise for the filter order to increase, the information gathered for the current filter order may be used and not be wasted.

A lattice based solution will have two predictions: the forward prediction and the backward prediction. The forward prediction error \( f_{m-1}(i) \) is determined by the tap inputs \( u(i), u(i-1), ..., u(i-n+1) \) while the order updated forward prediction error \( f_m(i) \) requires knowledge of \( u(i-m) \). However, this input can be obtained by delaying the backward prediction error \( b_{m-1}(i) \), which has the same inputs as those involved in \( f_{m-1}(i) \), by one time unit. In other words, \( b_{m-1}(i) \) can be seen as the input of the one-tap least squares filter which gives \( f_{m-1}(i) \) as the desired response and \( f_m(i) \) will be the residual resulting of the estimation of the least-squares estimation. The same loci can be followed to obtain \( b_m(i) \) and is explained in detail in[11].

Moreover, it is interesting to note that in the case of backward linear prediction, the update recursion for the tap-weight vector, \( \hat{w}_{h,m} \), requires knowledge of the current value \( k_m(n) \) of the gain vector, while in the case of forward linear prediction, the tap-weight vector requires information of the "past" value \( k_m(n-1) \) of the gain vector.[11]

Finally, one of the reasons that makes the study of this type of algorithms so interesting to the problem at hand is the property known as Exact Decoupling Property.

This property states that for a \( m \) stages predictor the backward prediction errors are uncorrelated (orthogonal) with each other. Considering that the transformation of the correlated input data sequence into a sequence of uncorrelated backward prediction errors is reciprocal, we can then conclude that the least squares lattice predictor preserves the full information of the input[11].

The fact that lattice algorithms bring these beneficial properties along with being able to maintain favorable characteristics of other algorithms (such as RLS) is the reason for the study of a Lattice Recursive Least Squares algorithm solution for the underwater equalization problem.

B. Kalman and LSL (Least Squares Lattice) variables

In this section the correspondences between Kalman variables and LSL variables will be presented to give a complete picture of the preliminaries necessary to the correct operation of the QRD-LSL algorithm.

The same line of thought can be followed to get similar equations for both the backward prediction and the joint-estimate. Table 1 summarizes the correspondences between these variables and Kalman variables.
Table 1: Correspondences between lattice and Kalman variables

<table>
<thead>
<tr>
<th>Kalman</th>
<th>Forward Prediction</th>
<th>Backward Prediction</th>
<th>Joint-Process Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y(n)$</td>
<td>$\alpha(n)$</td>
<td>$\beta(n)$</td>
<td></td>
</tr>
<tr>
<td>$u_H(n)$</td>
<td>$\lambda^{n/2}c_{j,m-1}(n)$</td>
<td>$\lambda^{n/2}c_{b,m-1}(n)$</td>
<td>$\lambda^{n/2}c_{m-1}(n)$</td>
</tr>
<tr>
<td>$x(n)</td>
<td><em>{Y</em>{n-1}}$</td>
<td>$-\lambda^{n/2}K_{j,m}(n - 1)$</td>
<td>$-\lambda^{n/2}K_{b,m}(n - 1)$</td>
</tr>
<tr>
<td>$K_{n-1}$</td>
<td>$\lambda^{-1}B^{-1}_{m-1}(n - 2)$</td>
<td>$\lambda^{-1}B^{-1}_{m-1}(n - 1)$</td>
<td>$\lambda^{-1}B^{-1}_{m-1}(n - 1)$</td>
</tr>
<tr>
<td>$\alpha(n)$</td>
<td>$\lambda^{n/2}\gamma_{j,m-1}(n - 1)\beta_{m}(n)$</td>
<td>$\lambda^{n/2}\gamma_{b,m-1}(n - 1)\beta_{m}(n)$</td>
<td>$\lambda^{n/2}\gamma_{m-1}(n - 1)\beta_{m}(n)$</td>
</tr>
</tbody>
</table>

The last line of the table can be obtained due to the fact that in Kalman filter theory the innovation is defined as follows:

$$\alpha(n) = y(n) - u^H(n)x(n)|_{Y_{n-1}}$$  \hspace{1cm} (1)

Finally we are in conditions of presenting the QRD-LSL algorithm for the single channel case, however, the case in study is multichannel and as such it is necessary to introduce it before presenting the algorithm used.

C. Expanding to Multichannel QRD-LSL

The experimental results are obtained with a multichannel derivation of the SISO case that was explained in Section III-A. The reason for the use of the multichannel case will now be given along with a brief explanation of the architecture itself.

There are three main reasons for the development of a multichannel case of QRD-LSL:

- Use of a feedback filter due to the DFE architecture
- Use of fractional sampling giving rise to a non-stationary signal
- Use of multiple hydrophones

Firstly, the use of a feedback filter in the architecture studied gives the advantage of being able to compensate errors by using information from a previous iteration, however, given that the feedback filter size has no relation with the feedforward filter size there is more often than not a case for different sizes, and therefore the need for a second channel. This will be most visible in the study of the case (Section III-E1) where channel estimation is mixed with a lattice filter to equalize the signal.

Secondly, the problem of non-stationary signals generated by the use of fractional sampling. If for example there is an oversampling of two then a possible solution would be to separate the samples in two stationary signals (sometimes denoted as polyphase components) which in itself gives need to a new channel.

Finally, the use of diversity by implementing multiple hydrophones, is the most common solution to mitigate the multipath effect when enhancing emitted power is not economically or technically viable. While there are several types of diversity, the one used in this study is most commonly known as spatial diversity. By using multiple receivers and combining the signals collected, the final result will be closer to what was initially emitted.

For a multichannel setup of $L$ input channels there will be $L$ signals $u(n)^{(i)}, 1 \leq i \leq L$ observed. These will then be linearly combined to approximate a reference signal with the value estimated being defined as $d(n) = w^*u(n)$.

It is important to note that the coefficient vector $w$ is the tap-weight vector and $u(n)$ is a vector of vectors in the following form:

$$u(n) = \begin{bmatrix} u^{(1)}(n) \\ \vdots \\ u^{(L)}(n) \end{bmatrix}$$  \hspace{1cm} (2)

The value $m_i$ immediately suggests that each channel may have a different filter order and this may in fact happen.

There are two main blocks in the multichannel architecture of the QRD-LSL algorithm: lattice and ladder blocks. Broadly, lattice blocks can be seen as groups of blocks of the forward and backward predictors, while ladder blocks correspond to the joint-estimation part. Further details will be given to both blocks starting with the lattice ones.

Figure 1 shows the structure of a lattice block for $L = 3$ and gives insight into how the filter order grows. This is done by increasing the order in one channel, which in turn will increase the order of another channel in the next stage until $L$ stages have passed, after which, the whole block will have increased in order[8].

![Fig. 1: Structure of a lattice block(L) for L = 3](image)

As said before each channel may have different filter orders, but after the explanation of the lattice blocks given this may not be clear, giving need for further clarification. Figure 2 shows the structure of the filter with the lattice blocks being defined as $L$ and the ladder blocks as $D$ with the size of the block in between brackets. The way to have channels with different orders is by cascading lattice blocks of increasing dimension. A detailed mathematical explanation can be seen in [8], however, intuitively it can be observed that channel one passes through all four blocks of growing order and therefore has order 4, while channel two passes only by three and for that reason has order 3, and finally channel three passes only by the last ladder block making it of order 1.
can be deduced as: the channel response \( h(t) \). The equations for the classical DFE (D-Basic DFE) are presented as multiple single channel cases summed to obtain the joint-error estimation seen in Section III-A. As seen in the section mentioned, the joint-error estimation relies on the backward error because it is a set of uncorrelated values.

In the case of multichannel lattice, lattice blocks that are higher than order 1 have more than one set of backward errors and since they are all uncorrelated they could all be used to update the ladder blocks. However, in practice the set used is the one of error \( \hat{b}_{L,l-1} \) which is associated with \( u_{L,l-1} \) that uses the same time reference for the most recent sample in all channels. Considering all of what was mentioned the joint-error recursion for order 1 can be defined as:

\[
e_l = e_{l-1} - K_l(n)\hat{b}_{L,l-1}
\]

where \( L \) is the order of the block and \( l \) is the stage of the block for which the error is calculated. In the case of the example seen in Figure 1 the joint error vector is composed of the values calculated for \( e_0 \), \( e_1 \), \( e_2 \) and \( e_3 \).

Table 2 summarizes the algorithm for the multichannel QR-RLS Lattice:

**D. Channel-Estimate-Based Adaptive Equalization**

The method that will be used is based on an adaptation of the one seen in Section II-C2. The main difference between the algorithm explained before and the one used is the lack of use of a pre-combiner. This decision was made considering that in the scenario tested there are few hydrophones making the use of a pre-combiner unnecessary.

Let the input signal be:

\[
v(t) = \sum_n d(n)h(t - nT) + w(t)
\]

This is sampled at the Nyquist or higher rate and so:

\[
v(n) = \sum_k h(k)d(n - k) + w(n)
\]

The time span \((-M_2T/2, M_1T/2)\) is chosen to capture all the channel response \( h(t) \). The equations for the classical DFE can be deduced as:

\[
d(n) = a'v(n) - \sum_{k>0} b_k^*d(n - k)
\]

where \( a \) is the array of equalizer coefficients that can be calculated by using an adaptive algorithm such as RLS or LMS. However, the decision can also be described in the following form:

\[
\hat{d}(n) = a'\left[\sum_{k<0} h(k)d(n - k) + w(n)\right] = a'v_f(n)
\]

In this case the feedforward signal, \( v_f(n) \), has to be obtained first so that the coefficients, \( a \), can be calculated and finally the estimate of the symbol sent for decision made.

The issue in this architecture is the fact that the \( v_f(n) \) can not be measured directly and must be obtained in an indirect way by subtracting the feedback signal, \( v_b(n) \), from the input signal:

\[
v_f(n) = v(n) - v_b(n)
\]

The signal \( v_b(n) \) can obtained from the previous decisions and the channel estimate:

\[
v_b(n) = \sum_{k>0} h(k)d(n - k)
\]

However, it has been proven in a study cited in [10] that \( v_b(n) \) obeys a shifting law which simplifies the operation. The relation mentioned is:

\[
\hat{v}_b(n) = \downarrow\hat{v}_b(n - 1) + \hat{h}(1)\hat{d}(n - 1)
\]

where \( \downarrow \) indicates shifting downward by as many elements as there are samples per one symbol interval.

The only variable that remains to define is the channel estimate:

\[
\hat{h}[n] = \lambda_{ch}\hat{h}[n - 1] + (1 - \lambda_{ch})v(n)d^*(n)
\]

This concludes the explanation for the single channel case but as mentioned before, in underwater communications it is common to use multiple channels to enhance the signal obtained by at the end user. The next section will summarize the multichannel case.

**E. Expanding to the Multichannel Case**

As can be seen in Figure 3 the multichannel case can be presented as multiple single channel cases summed to obtain the decided value.
A. Least Squares Lattice Equalization Results

 done by using shifting structures. Results obtained for this from the channel estimate as well as an update mechanism gaining insight into the usefulness of lattice algorithms in feedforward filter coefficients. This can be achieved by using with the use of a lattice algorithm in the processing of the algorithm studied is a a specific case of the

In chain $i = 1$ use the precomputed Givens matrix to update the second row of the backward prediction postarray above, then update the Givens matrix, prediction energy and conversion factor as:

$$
\begin{bmatrix}
\lambda^{1/2} F_{i-1}^{1/2} (n-1) \\
\lambda^{1/2} B_{L_i-1,1}^{1/2} (n-1) \\
0
\end{bmatrix}
\begin{bmatrix}
\hat{e}^f_{i-1,1-1} (n) \\
\hat{e}^b_{i-1,1-1} (n) \\
\gamma_{L_i-1,1-1} (n)
\end{bmatrix}
\Theta^f_{i,1} =
\begin{bmatrix}
\hat{f}^i_{i-1,1} (n) \\
\hat{p}^b_{i,1} (n) \\
\hat{e}^b_{i,1} (n)
\end{bmatrix}

In chain $i = 1$ the precomputed Givens matrix to update the second row of the backward prediction postarray above, then update the Givens matrix, prediction energy and conversion factor as:

In chain $i = 1$ use the precomputed Givens matrix to update the second row of the backward prediction postarray above, then update the Givens matrix, prediction energy and conversion factor as:

$$
\begin{bmatrix}
\lambda^{1/2} F_{i-1}^{1/2} (n-1) \\
\lambda^{1/2} B_{L_i-1,1}^{1/2} (n-1) \\
0
\end{bmatrix}
\begin{bmatrix}
\hat{e}^f_{i-1,1-1} (n) \\
\hat{e}^b_{i-1,1-1} (n) \\
\gamma_{L_i-1,1-1} (n)
\end{bmatrix}
\Theta^f_{i,1} =
\begin{bmatrix}
\hat{f}^i_{i-1,1} (n) \\
\hat{p}^b_{i,1} (n) \\
\hat{e}^b_{i,1} (n)
\end{bmatrix}
$$

Parameter initialization can be seen in [8]

However, some features have to be added such as phase tracking. This could have been incorporated into the channel response but having it explicitly separates the rapid time-variation of the phase from the channel which usually varies at a slower pace. These two variations can then be tracked by a PLL and the channel estimator respectively[10]. In this receiver it was decided to implement the PLL individually for all channels before the channel estimator.

All other operations are equivalent to the single channel scenario with the added detail that each channel must be equalized meaning that the same operations will have to be done for each hydrophone as seen in Figure 3. In Table 3 the summary of the algorithm used will be presented for the multichannel case.

1) Channel-Estimate-Based With Lattice Adaptive Equalization: The algorithm studied is a a specific case of the Channel Estimation one presented in the previous section with the use of a lattice algorithm in the processing of the feedforward filter coefficients. This can be achieved by using the lattice algorithm in Table 2 to update the equalizer vectors $a(n)$ presented in step 9 of Table 3.

The study of this option was done with the objective of gaining insight into the usefulness of lattice algorithms in the case of a non-classical DFE approach. In addition, this scenario provides the advantage of decoupling the equalizer from the channel estimate as well as an update mechanism done by using shifting structures. Results obtained for this algorithm will be present in Chapter IV.

IV. RESULTS AND ANALYSIS

A. Least Squares Lattice Equalization Results

In this section a more in depth explanation of the results obtained with a Least Squares Lattice Equalizer based on the canonical DFE approach will be given along with an analysis of its working procedure.

The results were obtained using data measured off the coast of Algarve during the Calcom10 sea trial. As part of the trial a Single Input Multiple Output (SIMO) configuration was tested

<table>
<thead>
<tr>
<th>Parameter initialization can be seen in [8]</th>
</tr>
</thead>
</table>

Table 3: Summary of Channel Estimation Algorithm[10]

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Form the matrix of input signals $V(n)$. This matrix is of size $K$ (Number of channels) x $P$ (Number of coefficients of the channel estimator) $V(n) = v_1(nT + MT_1T_2) \ldots v_1(nT) \ldots v_1(nT - M2T_2) \ldots v_K(nT + MT_1T_2) \ldots v_K(nT) \ldots v_K(nT - M2T_2)$</td>
</tr>
<tr>
<td>2</td>
<td>Compute the Phase-corrected signals $X_{\theta}(n) = \begin{bmatrix} e^{-j\theta_1(n)} &amp; \cdots &amp; 0 \ 0 &amp; \cdots &amp; 0 \ 0 &amp; \cdots &amp; e^{-j\theta_P(n)} \end{bmatrix} \times V(n)$</td>
</tr>
<tr>
<td>3</td>
<td>Compute $X_{\delta}(n)$ which is the post-cursor ISI matrix. This matrix is initialized with zeros and is updated at a later step</td>
</tr>
<tr>
<td>4</td>
<td>Compute $X_f(n) = X_{\theta}(n) - X_{\delta}(n)$</td>
</tr>
<tr>
<td>5</td>
<td>Compute $\alpha_p(n) = \hat{a}<em>p^T x_r$ and $\alpha</em>{pb}(n) = \hat{a}<em>p^T x</em>{pb}$</td>
</tr>
<tr>
<td>6</td>
<td>Compute the data symbol estimate $\hat{d}(n) = \sum_{p=1}^{P} \alpha_p(n) - \sum_{p=1}^{P} \alpha_{pb}(n)$</td>
</tr>
<tr>
<td>7</td>
<td>Compute the error as: $e(n) = d(n) - \hat{d}(n)$</td>
</tr>
<tr>
<td>8</td>
<td>Update the phase estimates as in [10]</td>
</tr>
<tr>
<td>9</td>
<td>Update the equalizer vectors $a(n+1) = a(n) + A_1[x_f(n), e(n)]$</td>
</tr>
<tr>
<td>10</td>
<td>Update the matrix of channel estimates $F[n] = \lambda_{ch} F[n-1] + (1 - \lambda_{ch}) X_{\theta}(n) d \ast (n)$</td>
</tr>
<tr>
<td>11</td>
<td>Compute the post-cursor ISI term $X_{\delta}^T(n+1) = X_{\theta}^T(n) + F^T[n] d^*(n)$</td>
</tr>
</tbody>
</table>

$$
\begin{bmatrix}
\lambda^{1/2} F_{i-1}^{1/2} (n-1) \\
\lambda^{1/2} B_{L_i-1,1}^{1/2} (n-1) \\
0
\end{bmatrix}
\begin{bmatrix}
\hat{e}^f_{i-1,1-1} (n) \\
\hat{e}^b_{i-1,1-1} (n) \\
\gamma_{L_i-1,1-1} (n)
\end{bmatrix}
\Theta^f_{i,1} =
\begin{bmatrix}
\hat{f}^i_{i-1,1} (n) \\
\hat{p}^b_{i,1} (n) \\
\hat{e}^b_{i,1} (n)
\end{bmatrix}
$$

Table 2: Summary of Multichannel QR-RLS Lattice Algorithm[8]

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$e_{i,0}^f = e_{i,0}^b = w^{(i)}(n)$</td>
</tr>
<tr>
<td>2</td>
<td>$e_0(n) = w^{(1)}(n)$</td>
</tr>
<tr>
<td>3</td>
<td>$e_0(n) = d(n), \gamma_{L_i,0}^{1/2}$</td>
</tr>
</tbody>
</table>
| 4 | $\lambda^{1/2} F_{i-1}^{1/2} (n-1) \\
\lambda^{1/2} B_{L_i-1,1}^{1/2} (n-1) \\
0
\begin{bmatrix}
\hat{e}^f_{i-1,1-1} (n) \\
\hat{e}^b_{i-1,1-1} (n) \\
\gamma_{L_i-1,1-1} (n)
\end{bmatrix}
\Theta^f_{i,1} =
\begin{bmatrix}
\hat{f}^i_{i-1,1} (n) \\
\hat{p}^b_{i,1} (n) \\
\hat{e}^b_{i,1} (n)
\end{bmatrix}
$$

However, some features have to be added such as phase tracking. This could have been incorporated into the channel response but having it explicitly separates the rapid time-variation of the phase from the channel which usually varies at a slower pace. These two variations can then be tracked by a PLL and the channel estimator respectively[10]. In this receiver it was decided to implement the PLL individually for all channels before the channel estimator.

All other operations are equivalent to the single channel scenario with the added detail that each channel must be equalized meaning that the same operations will have to be done for each hydrophone as seen in Figure 3. In Table 3 the summary of the algorithm used will be presented for the multichannel case.

1) Channel-Estimate-Based With Lattice Adaptive Equalization: The algorithm studied is a a specific case of the Channel Estimation one presented in the previous section with the use of a lattice algorithm in the processing of the feedforward filter coefficients. This can be achieved by using the lattice algorithm in Table 2 to update the equalizer vectors $a(n)$ presented in step 9 of Table 3.

The study of this option was done with the objective of gaining insight into the usefulness of lattice algorithms in the case of a non-classical DFE approach. In addition, this scenario provides the advantage of decoupling the equalizer from the channel estimate as well as an update mechanism done by using shifting structures. Results obtained for this algorithm will be present in Chapter IV.
with the emitter and receiver at around 10m along sea depths close to 100m and distances of approximately 1km making the data obtained useful given that multipath effects mentioned in Section II-A are more prominent in shallow areas.

The objective of the Matlab experiment is to verify that the behavior of signal passing through the lattice equalizer mirrors what was expected, which is, a gradual reduction of the error energy as the symbol passes through the modules.

1) Study of Behavior Along the Filter Modules: A series of simulations were devised with the aim of verifying that the reduction of error as the symbol passes through the modules of the filter is verified and an example of the results obtained is presented in Figure 4.

In this chapter N1 and N2 (where N2 is the limit of samples ahead of the symbol being estimated) used define the span of the feedforward filter and Nb denotes the size of the feedback filter. The span of the feedforward filter will be: \( N1 - (N2) + 1 \).

Fig. 4: MSE along filter modules for packet 1

The expected trend is generally seen giving support to the strategy envisioned for the automatic process of choosing equalizer order. The high reduction in MSE from filter modules 5 to 7 may have some relation to the channel response as most of the energy of the main impulse is within the first five taps.

Fig. 5: Channel Impulse Response for data of Packet 1

2) Study of Constellations by Varying Feedback and Feedforward Filter Size: The constellations obtained with the QR-LSL algorithm show a concentration of the symbols around the four points of the 4-PSK scheme used (-1, 1, j, -j) as was expected. However, there is still a fairly wide spread around each symbol, giving rise to errors.

In Figure 6 the constellation is clearly defined with only 54 errors detected in a packet with 5625 symbols, making the error rate less than 1%, which is acceptable given the media and the fact that the algorithm needs some symbols to converge.

The number of symbol errors obtained after equalizing gives a measure of the quality of the equalizing algorithm and is a deciding factor on whether the equalizer designed is viable or not given that the main objective of such a receiver is to compensate for channel ISI and ultimately present the information to the end user as close as possible to what was originally transmitted.

With this in mind the results obtained by equalizing the data obtained in the Calcom10 experiment were analyzed and presented in Figure 7.

Fig. 6: Constellation of LSL algorithm for a FeedForward filter size of 10 and Feedback filter size 5 for packet 1

Fig. 7: Number of Errors by increasing filter size in Packet 3

The trend mentioned is clearly visible in Figure 7, especially for spans up to size sixteen. However, if the filter span increases beyond this value, the number of errors start growing due to the appearance of gradient-noise amplification problems which stop the filter from reaching a steady-state[12]. This phenomena can be verified by the increase of the MSE (see Figure 8) after a specific span, in this case the increase
starts being visible around span sixteen which coincides with last span with no clear increase in number of errors as was expected. The span in which MSE starts increasing is related to the channel impulse response as can be seen in Figure 9 where the link between the response and the sixteen taps can be clearly seen since the response is within this span.

\[ \text{MSE by increasing filter size} \]

**Fig. 8:** MSE by increasing filter size in Packet 3

\[ \text{Channel impulse response} \]

**Fig. 9:** Channel impulse response for data of Packet 3

\[ \text{Energy Error along modules for a FeedForward Filter size 15 and Channel-Estimate size 50 for Packet 1} \]

**Fig. 10:** Energy Error along modules for a FeedForward Filter size 15 and Channel-Estimate size 50 for Packet 1

\[ \text{Constelation of CHE-LSL algorithm for a FeedForward filter size 15 and Channel-Estimate size 50 for Packet 3} \]

**Fig. 11:** Constelation of CHE-LSL algorithm for a FeedForward filter size 15 and Channel-Estimate size 50 for Packet 3

\[ \text{The span of the feedforward filter will be: } = N1 - (N2) + 1. \]

From Figure 10 it can be observed that the larger the filter is the smaller the error energy. However, the rate of error energy reduction decreases substantially after module four which gives insight into the possibility of reducing the number of modules used after a suitable period of training, therefore reducing complexity.

\[ \text{Study of Constellations by Varying Feedback and Feedforward Filter Size} \]

The study of the constellations was made with the objective of observing if there was a clear definition of the four symbols, giving a good picture of the effectiveness of the equalizer in study.

It must be noted that, even if, the constellation is clearly seen, the symbols obtained may not be correct as they could have suffered rotations. With this in mind the number of errors will also be studied to validate the constellations obtained.

The result obtained in Figure 11 presents four clearly defined symbols with barely any dots in areas of uncertainty. This is reflected in no errors found with these parameters.
The results obtained in Figure 11 show that the results obtained can be improved by increasing the size of feedforward filter, and in doing so increase the number of lattice modules. The results obtained for 11 are of good quality with a feedforward filter size of 15.

1) Study of Symbol Errors by Increasing Filter Size: The constellations obtained previously are quite good with symbols centered around the four possible cases of 4-PSK modulation. As before it must be verified if these results are correct and symbols are not rotated to be close to a different possible value.

After analyzing the data obtained from the equalization of the symbols using the channel estimate LSL algorithm it can be concluded that a small increase in the feedforward filter size improves significantly the success of process with almost no errors with sizes around 14 for even the worst case in study (packet 3) therefore reinforcing the results obtained in the constellations of Section IV-C.

The aggregated results for the Matlab experiments done can be seen in Figure 12, and present the expected trend of lowering number of errors when increasing filter size.

![Fig. 12: Errors CHE-LSL algorithm by increasing FeedForward filter size and Channel-Estimate size 50 for Packet 3](image_url)

After this study one can see that there is evidence to support the initial hypothesis of choosing filter size by following the error energy. This can be observed while comparing Figures 10 and 12 where the number of errors diminishes at the same pace of the error energy. There is also a link between these findings and channel impulse response seen in Figure 9 where a channel of around 16 taps manages to include the great majority of the impulse response.

V. CONCLUSION

In this thesis the lattice RLS based approach was studied with the aim of understanding if it was possible to bring the benefits of simple RLS, mainly fast convergence and stability, along with the added advantages of a lattice based algorithm which are mainly its linear complexity and modularity.

After running packets through the canonical DFE with a LSL algorithm several outputs were generated such as error energy through filter, constellations and number of errors achieved. These results suggest that the LSL can indeed be used for equalization as the expected decay of error through the modules was verified. The constellations obtained were acceptable with four distinct clouds centered in the 4-PSK values and finally the number of errors was found to be under 1% for a reasonable feedforward filter size.

To further reinforce the insight gained, a second approach based on the channel estimate coupled with a LSL equalizer was tested.

The same trend of lower error through the modules was observed. These findings were further verified by the constellations obtained which had even more distinct clouds. The number of symbols wrong after equalization were reduced to nearly zero from filter sizes with fifteen taps or more. This size is acceptable and is within the range observed in research compiled for the state of art.

Another insight taken from this experiment is that the channel estimate approach achieved better results than the DFE approach with this being especially visible in the number of errors achieved for both best and worst scenarios presented for each case. This seems to imply that the conjunction of the channel estimate approach proposed in [10] with a LSL algorithm is quite powerful by being capable of taking advantage of the beneficial characteristics of these separate functions.

The main focus of this thesis was to test if the LSL algorithm verified the expected behavior of reduction of error when passing through each module of the filter and since that this was now verified with real world data, all the background work has been done for the coding and final implementation of this approach to be possible.

REFERENCES


