Abstract

Digital communication using an acoustic wave transmitted through water in a single metal pipe or a network of metal pipes is considered. Echoes, multi-path and channel fading arise which can severely distort or corrupt the data transmitted. To counter the effects of echoes and multi-path fading two deconvolution algorithms, iterative reweighted least squares (IRLS) and residual steepest descent (RSD), are used to solve the following system identification problem: Given a record of data received, estimate the impulse response of the medium.

1. Introduction

Communication through a water pipe is conducted in the presence of acoustic echoes. An acoustic echo may be unnoticeable or distinct, depending on time delay involved. If the delay between the transmitted signal and its echo is short, the echo is unnoticeable, but perceived as a form of spectral distortion referred as reverberation. If on the other hand, the delay exceeds a few tens of milliseconds, the echo is distinctly noticeable [1].

Since it is practically difficult to generate and propagate an impulse at the input of a channel, often a system is instead excited by a narrow time domain pulse. The output is recorded and then a numerical deconvolution is often done to extract the impulse response of the channel. In the past, the fast Fourier transform technique has been applied with much success to the above deconvolution problem. However, when the signal-to-noise ratio (SNR) becomes small, sometimes one encounters instability with the FFT approach. Deconvolution attenuates reverberations and short-period multiples [2]. This serves as our motivation for studying deconvolution.

Deconvolution integrates in a natural way the presence of multiple echoes in the received signal. The deconvolution becomes difficult to analyse when the input and the impulse response of the system are both unknown.

To overcome this difficulty, one of the following two procedures may be used [1], [3]:
1) Predictive deconvolution where the procedure is based on linear prediction theory.
2) Blind deconvolution, which accounts for phase information contained in the data received and this information is ignored in predictive deconvolution.

Predictive deconvolution is achieved by designing processing filters, which minimize a measure of residuals, i.e. the difference between the desired and predicted response. Predictive deconvolution rests on two hypotheses [4]:
1) The feedback hypothesis, which treats the channel model to be autoregressive; the implication of this hypothesis is that the medium is minimum phase.
2) The random hypothesis, according to which the result of the deconvolution is assumed to have the properties of white noise, at least within certain time intervals.

Linear prediction has proven to be adequate when modelling a signal as the response of an all-pole system. Its advantage over other identification methods is that for signals well matched to the model it provides an accurate representation with a small number of easily computed parameters. However, in situations where spectral zeros are important, linear prediction is less satisfactory. It has been applied in the analysis of seismic data, although restrained by the fact that such data often involve a substantial mixed phase component [5].

One form of predictive deconvolution, based on $l_p$ norm analysis is called $l_p$ deconvolution or seismic deconvolution. In this paper two $l_p$ deconvolution methods are compared.

2. Model of the transmission channel

A message encoded in an acoustic wave is sent through a municipal pipe water network from a sender and retrieved...
at the receiver. The transmission medium is the water flowing in a pipe network.

The parameters of the system must be estimated, amongst other the impulse response of the system.

The channel was modelled as a causal Finite Impulse Response (FIR) filter and the length of the impulse response was estimated using one of the information theoretic criteria - minimum description length (MDL) criterion [6].

\[ \text{MDL}(k) = -(p - k)Q \ln \frac{G(\lambda_{k+1}, \ldots, \lambda_{p})}{A(\lambda_{k+1}, \ldots, \lambda_{p})} + \frac{1}{2} k(2p - k) \ln Q \]

where \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \) denote the eigenvalues of the autocorrelation function of the output vector \( E[\gamma(n)\gamma^T(n)] \) estimated as \( \frac{1}{Q} \sum_{n=1}^{Q} \gamma(n)\gamma^T(n) \), \( G \) is the geometric mean of the arguments, and \( A \) is the arithmetic mean of the arguments. The dimension of the signal space is taken to be the value of \( k = 0, \ldots, p - 1 \), for which \( \text{MDL}(k) \) is minimized. \( Q \) is the length of the output vector. An equalizer to remove the inter-symbol interference is not considered in this paper. The main goal is to obtain the model of the system and to undo the influence of the channel by finding its stable inverse.

### 3. \( l_p \) Deconvolution

Deconvolution is a process used to reverse the effects of convolution. The aim of the deconvolution is to find a solution of a convolution equation of the form \( x_n * a_n = y_n \). In real life, the process is usually modelled by \((x_n * a_n) + e = y_n \) where \( e \) is noise. Here, \( y_n \) is the data received, \( a_n \) is the ‘unknown’ convolution filter. The deconvolution matrix can be written in matrix form as

\[ Xa = y \]

with \( a = (a_1, \ldots, a_M)^T \), \( y = (y_1, \ldots, y_Q)^T \) and

\[
X = \begin{bmatrix}
    x_1 & 0 & \cdots & 0 \\
    x_2 & x_1 & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    x_N & \cdots & x_2 & x_1 \\
    0 & x_N & \cdots & x_2 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & x_N \\
\end{bmatrix}
\]

The advantages of \( l_p \) norm deconvolution are higher resolution and robustness to outlier noise; however, convergence problems and considerable use of memory and processing time resources are issues which must be regarded in their implementation [7].

The \( l_p \) norm estimators are the maximum likelihood estimators when the probability density function of the residual is the generalized \( p \) Gaussian. If the distribution of residual is unknown, a value of \( p \) can be found for \( l_p \) norm estimator to approach corresponding maximum likelihood estimation [8]. The performances have been demonstrated to be better than other linear parameter estimation methods [9].

For \( l_p \) norm, \( 1 \leq p \leq 2 \), values of \( p \) close to 1 produce deconvolution filters with less sensitivity to aberrant noise than those close to 2 (see [7]). For \( p \) greater than 2, filters design via \( p \) norm will be more sensitive to aberrant data. When \( 1 \leq p < 2 \), \( l_p \) is not a normed linear space and standard filter design is impossible.

In \( l_p \) norm deconvolution, the problem is to find \( a_n \) given \( x_n \) to minimize the error

\[ E_p = \sum |y_n - x_n * a_n|^p \] when \( y_n = x_{n+k} \) (1)

The \( l_2 \) solution to the filter design problem is

\[ a = (X^T X)^{-1} X^T y \] (2)

where \((X^T X)\) is a Toeplitz matrix. For \( 1 \leq p \leq 2 \) where \( p \) is real-valued the method used to find the solution of \( a \) are iterative reweighted least squares (IRLS) and residual steepest descent (RSD).

### 4. Algorithms

#### 4.1 IRLS Algorithm

The IRLS algorithm provides a means by which linear systems can be solved by minimizing the \( l_p \) norm of the residuals \( (1 \leq p \leq 2) \) [7]. In IRLS algorithm, \( l_p \) problem is solved by iteratively computing

\[ a(k + 1) = (X^T W(k) X)^{-1} X^T W(k) y \] (3)

where the weight \( W(k) \) is a diagonal matrix whose \((i,i)\)th element

\[ (W(k))_{ii} = W_i(k), \quad i = 1, 2, \ldots, Q \] (4)

is calculated from the residual vector

\[ r_i(k) = (y - Xa(k))_i \] (5)

with

\[ W_i(k) = \begin{cases} 
    |r_i(k)|^{p-2}, & \text{if } |r_i(k)| > \varepsilon \\
    \varepsilon^{p-2}, & \text{if } |r_i(k)| \leq \varepsilon 
\end{cases} \] (6)
for a small positive number \( \varepsilon \). This will help avoid a singularity for \( p = 1 \), which can result in small residual having the same order as the higher residual. The signal values predicted accurately will be given large weights in the next iteration. On the first iteration, \( a(1) \) is an \( l_2 \) solution in (2).

\( l_1 \) deconvolution with IRLS algorithm is especially efficient and robust in the presence of high-amplitude noise bursts [10].

### 4.2 RSD algorithm

The RSD algorithm also provides a way of solving linear systems by minimizing the \( l_p \) norm of the residual \((1 \leq p \leq 2)\) [7] and is computationally less complex than IRLS. The RSD algorithm solves the \( l_p \) problem by iteratively computing

\[
a(k + 1) = a(k) - \Delta_k (X^T X)^{-1} X^T y(k) \tag{7}
\]

where

\[
\Delta_k = (A^T W(k) A) A^T W(k) r \tag{8}
\]

(8) an IRLS solution of \( E(k) = \| r(k) - A_k A(k) \|_p \) , the initial value \( \Delta_0 \) is the \( l_2 \) solution in (8) and \( W(k) \) is computed as in (4) – (6) by replacing the residual vector \( r \) with \( E \) given by

\[
E(k) = r(k) - A_k A(k) \tag{9}
\]

with

\[
r(k) = X a(k) - y \tag{10}
\]

and

\[
A(k) = X (X^T X)^{-1} X^T y_k \tag{11}
\]

The column vector \( y(k) \) the gradient of the cost function

\[
y_k = [y_1(k), y_2(k), \cdots, y_Q(k)]
\]

with

\[
y_i(k) = \| X a - y \|^{p-1}_2 \text{sgn}(X a(k) - y) \tag{12}
\]

The condition for the stability of steepest descent algorithm depends on these three quantities [1]:

1) The starting point which is specified by the initial value \( a(0) \).
2) The gradient vector, which, at a particular point on the error-performance surface (i.e. a particular value of \( a(k) \), is uniquely determined by \( y \) and \( X \).
3) The step size parameter \( \mu \) controls the incremental change.

The necessary and sufficient condition for the convergence or stability of the steepest descent algorithm is

\[0 < \mu < \frac{2}{\lambda_{\text{max}}} \]

where \( \lambda_{\text{max}} \) is the largest eigenvalue of the correlation matrix of the output vector. The algorithm may require a large number of iterations for the algorithm to converge to a point sufficiently close to the optimum solution. The limitation is due to the fact that the steepest-descent algorithm is based on the straight-line (i.e. first-order) approximation of the error-performance surface around the current point.

### 5. Simulation results

All simulations were done using MATLAB. In the following simulation ten iterations were performed to update the residual for each of the IRLS and RSD algorithms. \( \varepsilon \) was taken equal to one hundredth of the maximum value of the convolution vector for both RSD and IRLS. Two signals are convolved and spiky noise is added to the convolution vector. A low pass filter, minimum phase signal \([-1 + e^{j/2\pi}]^2 [2 + e^{j/2\pi}]^{30} \Pi(r)\) (Figure 1) with a sampling interval \( T \) of 500ms is used to model the channel impulse response where “\( \Pi(r) \)” denotes a rectangular frequency function centred at 0 and with a width of 1. A random spiky signal is used as the input to the channel (Figure 2). A spiky noise signal (two spikes) is added to the convolution of the input signal samples and the channel impulse response. The noisy signal is shown in Figure 3.

Figure 1: Minimum phase signal- to model the channel impulse response.

The deconvolution between the noisy signal and the channel impulse response gives the signals shown in Figure 4 for the IRLS algorithm and in Figure 5 for the RSD algorithm. \( l_1 \) deconvolution totally eliminates spiky noise whereas with \( l_2 \) deconvolution, the recovered signal has been slightly changed.
Figure 2: A random input signal.

Figure 3: Convolution signal (the top) and the same signal added with spiky noise (bottom).

Figure 4: Deconvolved signal using IRLS.

Figure 5: Deconvolved signal using RSD.

Figure 4 and 5 show that the $l_2$ residual is much more perturbed than the $l_1$ residual. The $l_2$ filter attempted to remove the noise bursts, and in the process has of course transformed the rest of the signal. It is clear that $l_1$ being insensitive to spikes, is more reliable than $l_2$ deconvolution.

In the simulation for Bit Error Rate (BER) analysis, we varied the channel output SNR from 99dB down to 1dB by adding additive Gaussian noise to the convolution vector $y$. The input to the channel is a random sequence bits generated using 1245 random state generator. The modulation scheme is synchronous binary frequency-shift keying (FSK) chosen with the water pipe communication problem in mind. Bit ‘1’ was mapped to a time-limited 42kHz sine wave and bit ‘0’ to a time-limited 28kHz sine wave. In one symbol interval, there are 81 samples. The channel was modelled by the normalized version of the signal in Figure 1 (it was normalized to its maximum amplitude) which has a length of 100 samples. Additive Gaussian noise was added at the channel output signal. The convolution vector is deconvolved with the channel impulse response. For any single signal-to-noise ratio (SNR) value, the number of iterations used in implementing IRLS and RSD algorithms is 10 - the residual is updated at each iteration. Coherent detection was performed after the deconvolution. BER was calculated after demodulation. The digital data received was then compared with the digital input data to get the bit error rate. The $l_1$ deconvolution itself does not improve much on the result of $l_2$ deconvolution. The way to suppress noise is to use large damping factors, which are not a typical characteristic of the $l_1$ deconvolution. Using IRLS and RSD, there were no obvious difference between $l_1$ and $l_2$ (Figure 7 for IRLS and Figure 8 for RSD).
6. Application of $l_p$ deconvolution algorithms to a water pipe network

Next we consider $l_p$ deconvolution applied to a water pipe network. An electrical signal is converted to an acoustic signal by a transducer which propagates through flowing or still water. The acoustic wave is converted back to an electrical signal by a sensor at the far end of the pipe. Two 40kHz ultrasound piezoelectric transducers were used at the transmitter and the receiver ends. The received signal was sampled and 2500 samples were recorded at a sampling frequency of 1MHz. The input signal was a sine wave with amplitude 10kVolts and frequency of 40kHz. The frequency spectrum (frequency normalized to the sampling frequency $F_s$) of the received signal is shown in Figure 9. The flowing water and the environmental noise contributed mainly to the noise that appeared in the frequency spectrum.

The order of the channel was estimated using the MDL criterion in (1). The order of the channel was 2420. The $l_1$ and $l_2$ deconvolutions between the received signal and the channel input signal was performed using RSD. Two data sets of 2500 samples recorded were used. To perform the channel identification one set of data was used. The impulse response in Figure 10 was found using $l_1$. The reverberation appears in the impulse response of the signal. After the identification of the channel, another set of data was used and the output signal was deconvolved with the impulse response calculated to recover the signal sent at the transmitter input. Figure 11 shows the recovered signal using $l_1$ and Figure 12 depicts the deconvolution using $l_2$. 

![Figure 9: Frequency spectrum of the signal received](Image)
7. Conclusion

Using \( l_1 \) deconvolution is shown to be robust in presence of spiky noisy and it works moderately well in presence of additive Gaussian noise. The application of the \( l_1 \) deconvolution to a water pipe network for the identification of the channel showed positive results. Future work will focus on real-time DSP implementations of \( l_p \) deconvolution algorithms.

8. Acknowledgement

Failsafe, an on-campus INCENTIF company is thanked for their support and the use of their equipment.

9. References


---

Figure 10: Impulse response of the medium (channel order = 2420).

Figure 11: Recovered signal at the output (solid line) and the signal sent at the input of the channel (dashed line) using \( l_1 \) (RSD algorithm was used).

Figure 12: Recovered signal at the output (solid line) and the signal sent at the input of the channel (dashed line) using \( l_2 \). (RSD algorithm was used).

In \( l_1 \) deconvolution the signal is recovered with the same frequency and approximately the same amplitude as the input signal.