

1 Maximum-Length Sequences

A Maximum-Length Sequence (MLS) is a periodic two-level signal of length $P = 2^N - 1$, where N is an integer and P is the periodicity, which yields the impulse response of a linear system under *circular* convolution. The impulse response is extracted by the deconvolution of the system's output when excited with an MLS signal. This chapter presents the underlying theory and how the Fast Hadamard Transform can provide a very efficient means of analysing an MLS sequence. A MATLAB implementation of the algorithm presented in this chapter was written for the project (see Appendix 1). In doing so some difficulties arose from lack of emphasis of some important points in the available literature, so this description is aimed at the engineer interested in writing his own program.

Maximum-Length Sequences have a number of attractive properties. The most important is that with the exception of a DC error, the autocorrelation is a perfect impulse. They share many statistical properties with white noise despite being entirely deterministic, and may be generated very easily by use of shift-registers. Cross-correlation is computationally efficient as binary signals may be analysed using only additions and subtractions; no multiplications are necessary.

1.1 Basic MLS Theory

The MLS signal may be calculated recursively using the following relation:

$$n(k + 3) = n(k) \oplus n(k + 2) \quad (3.1)$$

Where \oplus denotes an XOR (modulo-2 sum) operation. It can be implemented with a set of shift registers (usually initialised to all 1s) and an XOR gate; an example sequence with $N=3$ is shown in *Figure 3.1*. Note that if the shift registers were initialised to consist entirely of zeros then it would be impossible for a 1 to ever occur and the circuit would remain frozen. It is for this reason that MLS signals are $2^N - 1$ and not 2^N in length.

The binary MLS is converted to a signal by the mapping:

$$\begin{aligned} 0 &\rightarrow 1 \\ 1 &\rightarrow -1 \end{aligned} \quad (3.2)$$

Let the MLS signal be $x[n]$ and the impulse response of the system be $h[n]$. The output, $y[n]$, of the system stimulated with $x[n]$ will therefore be:

$$\begin{aligned} y[n] &= h[n] * x[n] \\ &= \sum_{k=-\infty}^{+\infty} x[k]h[n-k] \end{aligned} \quad (3.3)$$

Where $*$ denotes discrete linear convolution. The

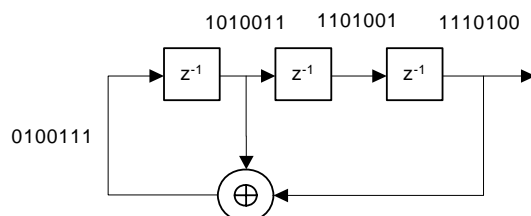


Figure 3.1: Binary feedback shift register for generating a Maximum-Length Sequence, $n=3$, $P=7$

assumption of linear convolution is not mathematically sound as it requires the MLS to be aperiodic. By definition this cannot be true unless $N=\infty$. Circular convolution is therefore more applicable in this case. Let the periodic impulse response be $h'[n]$ and let the Dirac delta $\delta[n]$ be defined as:

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

And the periodic Dirac delta be $\delta'[n]$:

$$\delta'[n] = \begin{cases} 1, & n \bmod P = 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.5)$$

Then,

$$\begin{aligned} h'[n] &= \sum_{k=-\infty}^{+\infty} \delta'[k] h[n-k] \\ &= \sum_{k=-\infty}^{+\infty} h[n+kP] \end{aligned} \quad (3.6)$$

Qualitatively, the periodic impulse response $h'[n]$ is the true impulse response $h[n]$ repeated at period P . If the periodic input is $x^p[n]$ and the periodic output is $y^p[n]$,

$$\begin{aligned} y^p[n] &= x^p[n] \otimes h'[n] \\ &= \sum_{k=0}^{P-1} x^p[k] h'[n-k] \end{aligned} \quad (3.7)$$

Where \otimes denotes circular convolution. $[n-k]$ is calculated modulo P and $h'[n]$, $x^p[n]$ and $y^p[n]$ are periodic in P .

P must be chosen to be sufficiently large that the transients from previous impulse responses $h_k[n]$ have died down enough not to cause time-aliasing. *Figure 3.2* demonstrates time-aliasing graphically.

MLS necessarily measures the periodic impulse response and not the true impulse response. However, when applied to a real system circular convolution cannot be achieved as real systems apply linear convolution, with an unexcited initial state. Circular convolution may be approximated by stimulating the system with the MLS sequence *twice* and analysing the second sequence; again the sequence must be long enough to prevent time aliasing.

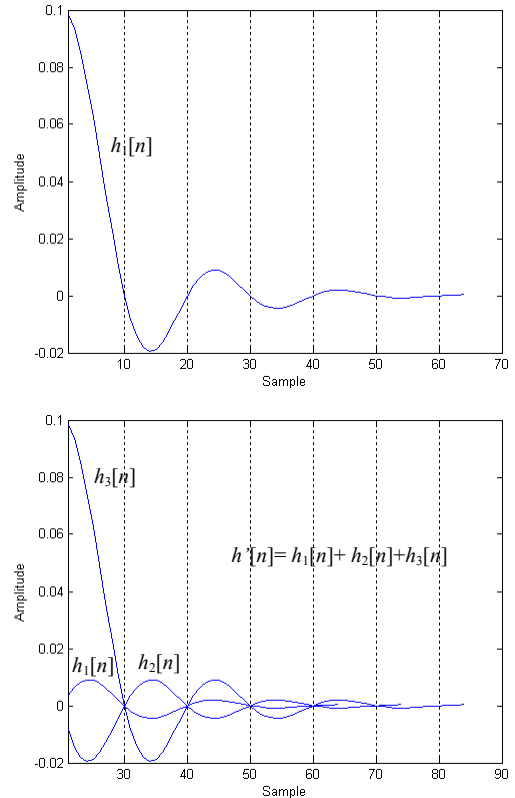


Figure 3.2: Time-aliasing in periodic impulse $h'[n]$ caused by insufficiently long MLS. Upper figure is $h[n]$ and lower, $h'[n]$, is $h[n]$ repeated every P samples.

1.2 Deconvolution

Many texts on MLS present the deconvolution operation in terms of cross-correlation [1], [2]. I shall present my discussion in terms of convolution in a similar fashion to [3].

Recall that

$$y[n] = h[n] * x[n] \quad (3.8)$$

Then

$$y[n] * z[n] = h[n] * x[n] * z[n] = h'[n] \quad (3.9)$$

Iff $x[n] * z[n] = \delta'[n]$ and there is negligible time-aliasing.

If $z[n] = x[-n]$ then $x[n] * z[n]$ is the autocorrelation function of $x[n]$ and $x[n] * z[n] = \delta'[n]$ is fulfilled except for a DC offset.

By deconvolving, the signal $h'[n]$ is given by:

$$h'[n] = \frac{1}{P+1} \left(\sum_{k=1}^P y[k] x[k-n] \right) \quad (3.10)$$

(Recall $h'[n]$ is a periodic version of $h[n]$).

In the case when $P=7$:

$$\begin{bmatrix} h[0] \\ h[1] \\ h[2] \\ h[3] \\ h[4] \\ h[5] \\ h[6] \end{bmatrix} = \frac{1}{8} \begin{bmatrix} x[0] & x[1] & x[2] & x[3] & x[4] & x[5] & x[6] \\ x[1] & x[2] & x[3] & x[4] & x[5] & x[6] & x[0] \\ x[2] & x[3] & x[4] & x[5] & x[6] & x[0] & x[1] \\ x[3] & x[4] & x[5] & x[6] & x[0] & x[1] & x[2] \\ x[4] & x[5] & x[6] & x[0] & x[1] & x[2] & x[3] \\ x[5] & x[6] & x[0] & x[1] & x[2] & x[3] & x[4] \\ x[6] & x[0] & x[1] & x[2] & x[3] & x[4] & x[5] \end{bmatrix} \begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \\ y[6] \end{bmatrix}$$

In DSP applications, the matrix containing $x[n]$ is called the M -sequence matrix and shall be referred to as such from this point onwards.

Correlation detection with the M -sequence matrix can become prohibitively complex to compute as P becomes large. Applications in room acoustics can involve impulse responses of 5 seconds or more, which at a sample rate of 48 kHz requires P to be in the order of 2^{18} samples long. Correlation detection requires P^2 operations, which on a 100MHz DSP processor would take approximately 11 minutes to calculate.

Computational complexity can be dramatically reduced by exploiting similarities between the M -sequence matrix and the Walsh-Hadamard matrix, reducing the number of additions to $P \log_2(P)$ by use of the Fast Walsh-Hadamard Transform.

The M -sequence matrix may be described by:

$$m_{i,j} = n_{i+j-2(\text{modulo-}P)} \quad i, j = 1, 2, \dots, P$$

Or successive circular left shifts of the top row. For $P=7$:

$$\mathbf{M} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

From the construction of \mathbf{M} , every row (column) of \mathbf{M} can be expressed as a linear, modulo-two combination of the first N rows of \mathbf{M} .

$$\mathbf{M} = \mathbf{L}\mathbf{S} = \mathbf{S}'\mathbf{L}' \quad (3.11)$$

Where \mathbf{L} is a binary matrix order $P \times N$ and \mathbf{S} is order $N \times P$ and is formed by the first N rows of \mathbf{M} .

Since all the rows of \mathbf{M} are distinct, all the rows of \mathbf{L} must be distinct, therefore every nonzero binary N -vector must appear as some row in \mathbf{L} . Also, the first N rows of \mathbf{L} form an identity matrix of N .

Let $\boldsymbol{\sigma}$ be the square matrix of order N , formed by the first N columns of \mathbf{S} . Then,

$$\mathbf{L}\boldsymbol{\sigma} = \mathbf{S}' \quad (3.12)$$

Every nonzero binary N -vector appears as a row in both \mathbf{L} and \mathbf{S}' , $\boldsymbol{\sigma}$ is necessarily non-singular, so,

$$\mathbf{L} = \mathbf{S}'\boldsymbol{\sigma}^{-1} \quad (3.13)$$

Using the above example,

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (3.14)$$

$$\boldsymbol{\sigma} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad (3.15)$$

$$\boldsymbol{\sigma}^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (3.16)$$

former on top with a row of all zeros and the latter on the left by a column of all zeros. The bordered matrix $\hat{\mathbf{M}}$ is then given by:

$$\hat{\mathbf{M}} = \hat{\mathbf{L}}\hat{\mathbf{S}} \quad (3.21)$$

The factors $\hat{\mathbf{L}}$ and \mathbf{B} , $\hat{\mathbf{S}}$ and \mathbf{B}' each exhaust all N -vectors in their rows and columns respectively. They differ only in the order in which the vectors occur, so we can write:

$$\hat{\mathbf{L}} = \mathbf{P}_L \mathbf{B} \quad (3.22)$$

$$\hat{\mathbf{S}} = \mathbf{B}' \mathbf{P}_S \quad (3.23)$$

Where \mathbf{P}_L and \mathbf{P}_S are $2^N \times 2^N$ permutation matrices describing the N -vector orderings in $\hat{\mathbf{L}}$ and $\hat{\mathbf{S}}$. This simple relation is due to the fact that the rows of \mathbf{B} and columns of \mathbf{B}' occur in natural numerical order. Using previous relations,

$$\hat{\mathbf{M}} = \hat{\mathbf{L}}\hat{\mathbf{S}} = \mathbf{P}_L \mathbf{B} \mathbf{B}' \mathbf{P}_S = \mathbf{P}_L \mathbf{H} \mathbf{P}_S \quad (3.24)$$

Relation (3.24) holds regardless of whether values $\{0,1\}$ or $\{1, -1\}$ are used. An M -sequence transform can be summarised by:

1. Reorder the received data from the device under test after stimulation with an MLS signal according to \mathbf{P}_S .
2. Perform a Walsh-Hadamard transform.
3. Reorder the transformed data according to \mathbf{P}_L and divide by $P+1$.

Note that the Walsh-Hadamard transform by matrix multiplication gains no computational advantage; instead the Fast Walsh-Hadamard transform should be used.

1.4 The Fast Walsh-Hadamard Transform

There are many different algorithms for calculating a 'fast' Walsh-Hadamard Transform. The following is one of the most elegant:

1. Construct $N+1$ columns, each with 2^N rows.
2. Place the input vector $x[n]$ in the 1st column.
3. Moving left to right, fill in the next column as follows:
 - a. In the top half of this column, place the pairwise, mutually exclusive sums of the previous column.

| x | a | b | y |
|-------------|------------------|------------------|------------------|
| x[0] | $a[0]=x[0]+x[1]$ | $b[0]=a[0]+a[1]$ | $y[0]=b[0]+b[1]$ |
| x[1] | $a[1]=x[2]+x[3]$ | $b[1]=a[2]+a[3]$ | $y[1]=b[2]+b[3]$ |
| x[2] | $a[2]=x[4]+x[5]$ | $b[2]=a[4]+a[5]$ | $y[2]=b[4]+b[5]$ |
| x[3] | $a[3]=x[6]+x[7]$ | $b[3]=a[6]+a[7]$ | $y[3]=b[6]+b[7]$ |
| x[4] | $a[4]=x[0]-x[1]$ | $b[4]=a[0]-a[1]$ | $y[4]=b[0]-b[1]$ |
| x[5] | $a[5]=x[2]-x[3]$ | $b[5]=a[2]-a[3]$ | $y[5]=b[2]-b[3]$ |
| x[6] | $a[6]=x[4]-x[5]$ | $b[6]=a[4]-a[5]$ | $y[6]=b[4]-b[5]$ |
| x[7] | $a[7]=x[6]-x[7]$ | $b[7]=a[6]-a[7]$ | $y[7]=b[6]-b[7]$ |

Figure 3.3: A Fast Hadamard Transform

- b. In the bottom half, place the pairwise differences.
- c. Repeat (Step 3) for each of the remaining columns.

The last column contains the Hadamard transform, $y[n]$, of the input vector, $x[n]$.

1.5 Alternative Permutation

The matrices \mathbf{P}_S and \mathbf{P}_L are presented here merely to clarify the underlying theory. In practice the permutation operation may be calculated by finding the sum of the columns of \mathbf{S} and the rows of \mathbf{L} considered as a binary number, then reordering according to these values as follows:

$$\begin{array}{r}
 \times 4 \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \\
 \times 2 \phantom{\begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}} \\
 \times 1 \phantom{\begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}} \\
 \hline
 (1 \ 2 \ 5 \ 3 \ 7 \ 6 \ 4)
 \end{array}
 \quad
 \begin{array}{r}
 \times 4 \quad \times 2 \quad \times 1 \\
 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}
 \begin{array}{l}
 (4) \\
 (2) \\
 (1) \\
 (6) \\
 (3) \\
 (7) \\
 (5)
 \end{array}
 \end{array}$$

$$\begin{array}{l}
 (y[1] \ y[2] \ y[3] \ y[4] \ y[5] \ y[6] \ y[7]) \rightarrow \\
 (y[1] \ y[2] \ y[4] \ y[7] \ y[3] \ y[6] \ y[5])
 \end{array}
 \quad
 \begin{array}{l}
 (h[4] \ h[2] \ h[1] \ h[6] \ h[3] \ h[7] \ h[5]) \rightarrow \\
 (h[1] \ h[2] \ h[3] \ h[4] \ h[5] \ h[6] \ h[7])
 \end{array}$$

1.6 DC Coupling

Some papers [2], [6], have wrongly suggested methods for recovering DC components with MLS analysis. It is shown in [3] that if a system is AC coupled (as is the case in almost audio applications), sum of $h[n]$ over all n is zero:

$$\sum_n h[n] = 0 \tag{3.25}$$

It is suggested in [2] that in order to recover a DC component then equation (x) should be re-written with a DC offset term as follows:

$$h'[n] = \frac{1}{P+1} \left(\sum_{k=1}^P y[k]x[k-n] - \sum_{k=1}^P y(k) \right) \tag{3.26}$$

Which is equivalent to bordering \mathbf{L} and \mathbf{S} with $-\sum_n y[n]$ instead of zeros to form $\hat{\mathbf{L}}$ and $\hat{\mathbf{S}}$.

A simple argument against doing this is that in the case of a DC coupled system, the DC component may only be extracted either by infinitely long or asymmetrical MLS stimuli, both of which are outside the scope of this project. I ignore the DC term entirely as it is inconsequential to loudspeaker equalisation.

1.7 SNR

Let the system's impulse response be $h(t)$ and the noise $n(t)$. Noise sources include quantisation (shot noise), thermal noise in the amplifiers and microphone and background noise. The signal received at the analyser is $g(t)=h(t)+n(t)$.

Averaging successive MLS bursts (again ignoring the first burst), we obtain the signal $\bar{g}(t)$. Assume that the noise process is ergodic, zero mean and white (it has an autocorrelation function of 0 everywhere but $k=0$):

$$E\{n(t)\}=0$$

$$R[k]=\frac{1}{P}\sum_{t=0}^{P-1-k}n(t)n(t+k)=\sigma_{n(t)}^2\delta(k) \quad (3.27)$$

Assume also that the system's impulse response is time-invariant.

$\bar{g}(t)=\frac{1}{P}\sum_{i=1}^Pg_i(t)=\frac{1}{P}\sum_{i=1}^Ph(t)+n_i(t)=h(t)+\frac{1}{P}\sum_{i=1}^Pn_i(t)$ The variance of the noise is given by:

$$\begin{aligned} \sigma_{n(t)}^2 &= E\{n^2(t)\} = E\left\{\left(\frac{1}{P}\sum_{i=1}^Pn_i(t)\right)^2\right\} = \frac{1}{P^2}E\left\{\left(\sum_{i=1}^Pn_i(t)\right)^2\right\} \\ &= \frac{1}{P^2}E\left\{\left(\sum_{i=1}^Pn_i^2(t)\right)\right\} + \frac{1}{P^2}\left\{E\left\{\sum_{i=1}^P\sum_{j=1}^P(n_i(t)n_j(t))\right\}\right\} \\ &= \frac{1}{P^2}E\left\{\left(\sum_{i=1}^Pn_i^2(t)\right)\right\} + \frac{1}{P^2}\sum_{i=1}^P\sum_{j=1}^PE\{n_i(t)n_j(t)\} \\ &= \frac{1}{P^2}\sum_{i=1}^P\sigma^2 + 0 \\ &= \frac{1}{P}\sigma^2 \end{aligned} \quad (3.28)$$

Therefore as $P \rightarrow \infty$, $\sigma_{n(t)}^2 \rightarrow 0$. As signal power is proportional to its variance, for every doubling in the number of averaged sequences, the SNR is improved by 3dB.

1.8 Practical Measurements

The frequency response at different amplitudes is obtained by analysing bursts of MLS at increasing amplitude. In order to correctly time-align the analysis software, an impulse of one sample is used as a sync pulse and the alignment of the MLS bursts is referenced to this point. Details of the issues of correct time-alignment and general information about the practical implementation of MLS code are given in Appendix 1.

1.9 Summary

MLS analysis is a computationally efficient method for extracting the impulse response of linear systems. The steps for conducting an MLS analysis are as follows:

1. Determine the approximate length of a system's impulse response. Generate an MLS sequence at least as long as the impulse response using a binary feedback register and XOR gate as described in equation (1). Convert the sequence using the mapping $0 \rightarrow 1$, $1 \rightarrow -1$.
2. Stimulate the system with at least two successive MLS bursts and record the output. The SNR is improved by 3dB for a doubling in the number of repetitions.
3. Take the mean of all but the first burst.
4. Add a zero to the first element and reorder the averaged data according to \mathbf{P}_s .
5. Perform a Fast Walsh-Hadamard transform.
6. Reorder the transformed data according to \mathbf{P}_L and divide by $P+1$.
7. The result is an estimation of a system's impulse response of length $P+1$. Perform an FFT to find the frequency-domain representation.

The importance of approximating circular convolution is often overlooked and can yield highly inaccurate results if it is ignored.

See the user guide in Appendix 1 and the Matlab code in Appendix 3 for further information about how this algorithm is implemented.

[1] J. Borish & J. B. Angell, "An Efficient Algorithm for Measuring the Impulse Response Using Pseudorandom Noise," *J. Audio Eng. Soc.*, vol. 31, pp. 478-489 (1983)

[2] D. D. Rife & J. Vanderkooy, "Transfer-Function Measurement with Maximum-Length Sequences," *J. Audio Eng. Soc.*, vol. 37, pp. 419-444 (1989)

[3] M. Cohn & A. Lempel, "On Fast M-Sequence Transforms," *IEEE Trans. Inf. Theory*, IT-23 pp. 135-137 (1977)