Learning Neural Networks for Detection and Classification of Synchronous Recurrent Transient Signals

Erol Gelenbe *, Kerem Harmancı,
and Jeffrey Krolik †
Department of Electrical & Computer Engineering
Duke University
Durham, NC 27708-0291
{erol,harmanci}@ee.duke.edu

August 16, 1997

Abstract

This paper proposes a neural network solution to the classical signal processing problem of detection of a synchronous recurrent transient signal in noise. If a signal exists, it is assumed to be one of M known signals which may sometimes occur (probabilistically) in successive intervals. Several neural network configurations are applied to this problem and compared with each other and with the optimum adaptive sequential detector. A novel efficient neural network detector is proposed using an $XOR-Tree$ configuration with learning. Tests with synthetic and real noise, show the excellent performance of this aproach as compared to the optimum adaptive detector and to other neural network techniques. With real (non-white) noise obtained from sonar data, the $XOR-Tree$ network widely outperforms the likelihood ratio detector. We also discuss the learning time complexity of the $XOR-Tree$ network and compare it to that of standard three layer network architectures.

1 Introduction

The objective of this paper is to address a classical problem in detection theory using neural networks. The problem, which will be described in greater detail below, is that of the detection of *a synchronous recurrent transient signal in noise.* The prior knowledge is that if a signal exists, it will be just *one of M known signals* which may occur in specific intervals. Each of the M possible signals is represented by a distinct specific (and in general non-standard) waveform. A classical approach for this problem uses an adaptive sequential version of the likelihood ratio [5], which is also capable of classifying which of the waveforms or signals occur if it exists by the use of *Maximum A Posteriori Estimation.* This receiver gives the optimum detection/classification performance, provided the M signals are orthogonal and the statistics of the signal and of the noise are known exactly.

In many cases, characteristics of the noise are not available except through observations during the actual classification process. Real noise will often be non-white and it will have been observed over

*E. Gelenbe's work was supported by the Office of Naval Research under ONR Contract N00014-97-1-0112
†J. Krolik and K. Harmancı were supported in part by the Office of Naval Research under ONR Contract N66001-95-C-6032.

a relatively short interval. In many applications the signal detection will have to be carried out very rapidly and accurately. Thus it is of interest to consider non-parametric techniques based on learning neural networks.

Neural networks have long been applied to signal processing applications, and excellent surveys exist in this area [4, 19]. Applications of neural networks to image processing problems are quite common, including [10, 11, 13, 14, 20] where non-standard biologically inspired pulsed recurrent networks are used [7, 8, 12].

In this paper neural networks are applied to the detection and classification of synchronous recurrent transiet signal in noise, and evaluated under the same conditions as the optimum detector/classifier. Different neural network configurations are compared with each other, and with the adaptive sequential detector which is used as a benchmark. A novel $XOR-Tree$ neural network architecture, together with appropriate learning algorithms, is proposed as an effective tool for the detection problem.

We show that the $XOR-Tree$ network is a natural approach to the problem since it can solve the problem exactly when noise is very low. We prove that its learning time complexity with the backpropagation algorithm [4] is $O(M^2)$, whereas a three layer standard $M \times N \times 1$ network architecture can have a larger learning time complexity of $O(N.M)$ when $N \geq M$. The $XOR-Tree$ architecture, as well as the other neural networks previously suggested, provide satisfactory performance with respect to the known optimum likelihood ratio (LR) detector/classifier for high noise levels when the noise is an "ideal" Gaussian White noise, with the $XOR-Tree$ coming out ahead of other neural net detectors tested. The tests show that the $XOR-Tree$ network has superior performance with respect to the LR algorithm with real non-White noise taken from sonar data.

1.1 Problem Statement

First let us introduce some notation. An *observation frame* $\mathbf{F_L}$ is made up of a finite sequence of observation intervals of the same duration T denoted by $\mathbf{O}_k^L,\ k = 1, 2, ..., K$:

$$\mathbf{F}_L = \{\mathbf{O}_k^L, k = 1, ..., K\}. \tag{1}$$

$\mathbf{O}_k^L$ begins at $(L-1)KT + (k-1)T$ and ends at $(L-1)KT + kT$. Within any of these intervals there may or may not be some real valued signal $\mathbf{c}_m(t),\ t \in [0, T[,\ m = 1, ..., M$. It is assumed that each of the M signals has unit norm: $\int_0^T \mathbf{c}_m^2(t)dt = 1$, and furthermore that the signals are orthogonal:

$$\int_0^T \mathbf{c}_m(t)\mathbf{c}_l(t)dt = 0, for\ m \neq l. \tag{2}$$

The extension to complex valued signals is straightforward. An example of four such signals is shown in Figure 2.

To describe the signal observed in $\mathbf{O}_k^L$ let us use the notation $\mathbf{x}_k^L = \{x_k^L(t),\ t \in [(k-1)T, kT[\}$ where $\mathbf{x}_k^L(t),\ t \in [(K-1)T + (k-1)T, (K-1)T + kT[$ is the signal measured or observed in $\mathbf{O}_k$. $\mathbf{X}_k^L$, will denote the total observation or measurement in $\mathbf{F}_L$, up to and including the subinterval k, is defined as

$$\mathbf{X}_k^L = [\mathbf{x}_1^L, ..., \mathbf{x}_i^L, ...\mathbf{x}_k^L], 1 \leq k \leq K. \tag{3}$$

For convenience, we take $\mathbf{X}_0^L$ to be the empty measurement.

The setting for the synchronous recurrent transient detection problem is stated with two hypotheses defined for a frame composed of K intervals. The binary hypothesis format is formulated as,

$$\begin{aligned} H_1 : &\quad x_k^L(t) = \mathbf{s}_k^L(t) + \mathbf{n}_k(t), \\ H_0 : &\quad x_k^L(t) = \mathbf{n}_k(t), \end{aligned}$$

for $k = 1, ..., K$ and $t \geq 0$, where

- $\mathbf{s}_k^L(t) \in \{a.\mathbf{c}_m(t - [(L-1)K + (k-1)]T),\ m = 1, ..., M\}$, and a is a real positive constant,
- $\mathbf{n}_k(t)$ is zero-mean stationary White Gaussian Noise (WGN) with variance σ_n^2.

If no signal exists (H_0), $x_k(t)$ will be zero-mean White Gaussian Noise (WGN) with variance σ_n^2. If a signal does exist (H_1), it will be only one of the M signals $\mathbf{s}_k^L(t) = a\mathbf{c}_m((t - [(L-1)K + (k-1)]T)$ which will be received within each observation period of the entire frame in a probabilistic manner: the same signal occurs *independently* in each of the K intervals of the $L-th$ frame with probability $\nu_{L,k}$ in interval k of the $L-th$ frame. Note that this probability does not depend on m the signal number, but does depend on the observation period. The Signal-to-Noise-Ratio (SNR) for the $k-th$ interval is then

$$SNR = \frac{a^2 \nu_{L,k}^2}{\sigma_n^2} \tag{4}$$

since all the m signals have unit norm. In the sequel we will drop the dependency of ν on L, k. A somewhat different metric we will use is the "occupancy" Signal-to-Noise-Ratio (SNR_{occ}) which is the SNR in an interval in which the signal does exist:

$$SNR_{occ} = \frac{a^2}{\sigma_n^2} \tag{5}$$

The objective of the *problem of detecting synchronous recurrent transient signals* is to determine:

- Whether there is a signal in a given frame,
- And if so, to identify which of the m known signals it is.

Detection schemes need to be evaluated according to some useful metrics. Typically one can use the *Probability of Detection (P_D)* which is the probability of a correct decision given that the signal exists. Another useful metric is P_{fa}, the probability of error given that the signal does not exist, i.e. the probability that the detection scheme concludes that the signal exists, despite the fact that it does *not* exist. P_{fa} is often called the probability of "false alarm".

In the optimal detector, the last step is to threshold the likelihood ratio for a decision of whether there is a signal (H_1) or not (H_0). This threshold is chosen so that the detector has a certain probability of false alarm (P_{fa}); in turn the threshold determines the value of P_d. However in this paper these measures are not used. Instead the measure we use is the probability of a correct decision (or Probability of Correct Detection) P_{CD} given that the absence and presence of the signal are equally likely. However, the correct terminology for the complement of this measure is the "probability of error" P_e^D, given that the absence and presence of the signal are equally likely):

$$P_e^D = 1 - P_{CD}. \tag{6}$$

To be consistent we also define the "probability of error" P_e^C for the classification problem, in which we use the "probability of correct classification" P_c^C given that the absence and presence of a signal are equally likely, and that if there is a signal, each waveform is equally likely. Here

$$P_e^C = 1 - P_e^C. \tag{7}$$

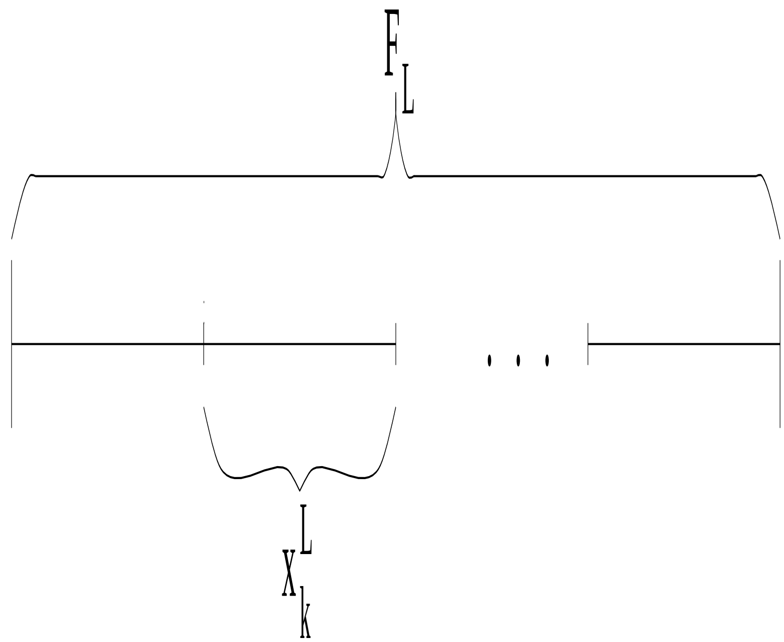

Figure 1: Synchronous Recurrent Waveform time format

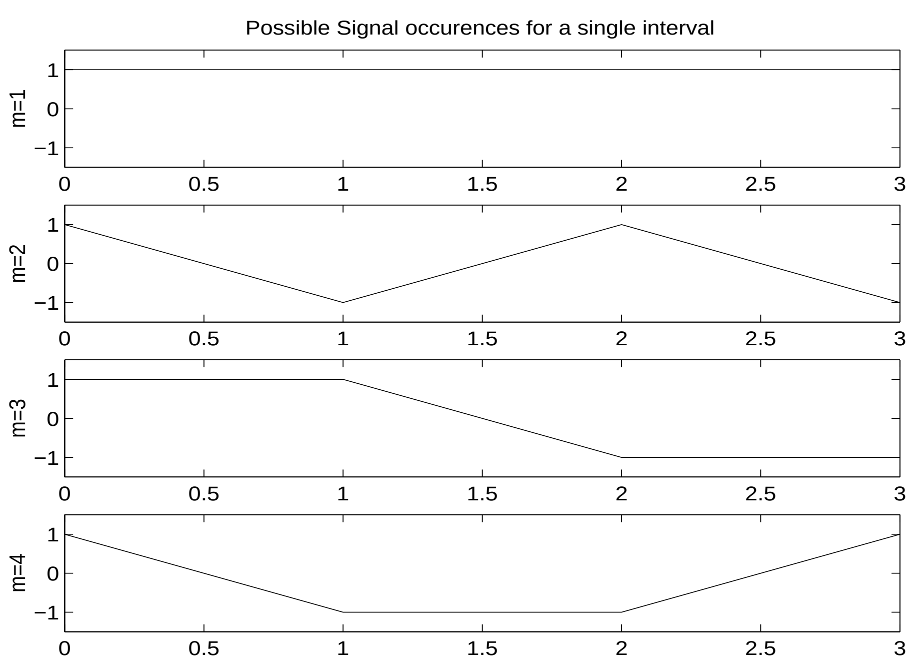

Figure 2: Waveforms of the M=4 possible signals. (Single interval)

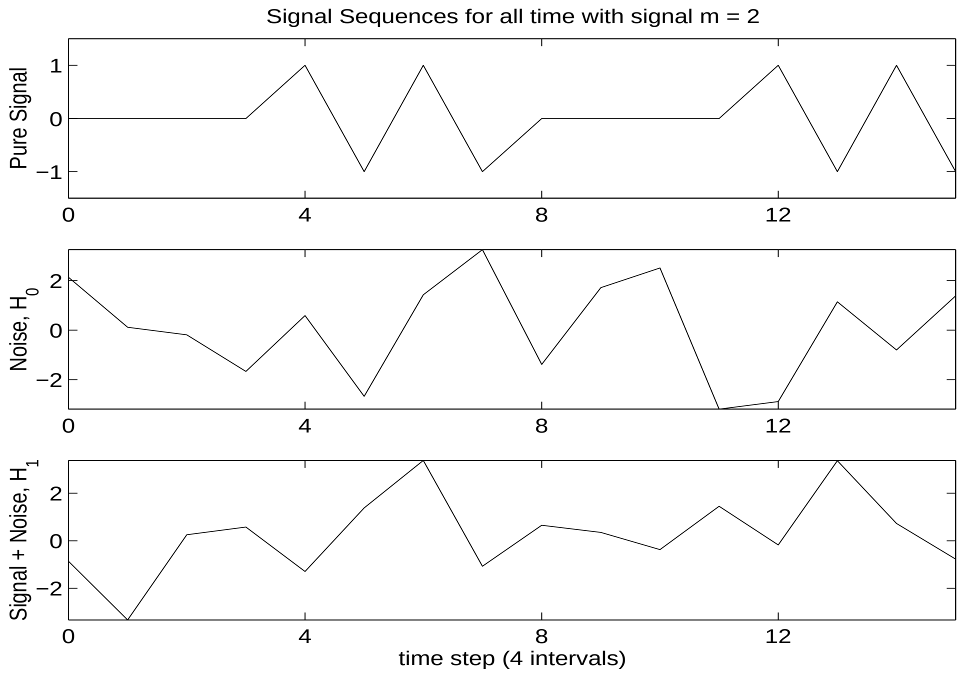

Figure 3: One observation of the synchronous recurrent transient, 4 intervals, each with 4 time steps, $\nu = 0.5$, $SNR_{occ} = 0$ dB

1.2 Adaptive Sequential Optimal Detector

Let us briefly recall the classical approach to optimal detection as applied to the problem at hand. Here we concentrate on a single specific frame L, and therefore drop this index. All observations and computations will relate to this frame, and the rest of the notation is as defined above.

Detection theory provides the optimum detector for this problem using the Likelihood Ratio [2]:

$$\Lambda(\mathbf{X}_k) = \frac{P(\mathbf{X}_k|H_1)}{P(\mathbf{X}_k|H_0)}. \tag{8}$$

The decision threshold for detection with the likelihood ratio 8 is set to $\frac{P(H_1)}{P(H_0)}$ in order to minimize the probability of detection error, as discussed in [2, 1].

In general, one assumes that the statistics $P(\mathbf{X}_k|H_1)$ and $P(\mathbf{X}_k|H_0)$ are known exactly and can be computed (for instance) directly from the hypotheses. The presence/absence decision concerning a signal is then made by obtaining the likelihood ratio (8) as described below, and then simply comparing it to a threshold. This threshold is selected according to the detection criterion used and the prior knowledge on the probability of the presence of signal. Let us use the following notation:

$$\lambda(\mathbf{x}_k|m) = \frac{P(\mathbf{x}_k|H_1\& \; The\ Signal\ is\ \mathbf{c}_m)}{P(\mathbf{x}_k|H_0\ \&\ The\ Signal\ is\ \mathbf{c}_m)}, \tag{9}$$

$$P(\mathbf{x}_k|m, H_1) = P(\mathbf{x}_k|The\ Signal\ is\ \mathbf{c}_m\ \&\ H_1), \tag{10}$$

$$P(m|\mathbf{X}_{k-1}, H_1) = P(The\ Signal\ is\ \mathbf{c}_m|\mathbf{X}_{k-1}\ \&\ H_1). \tag{11}$$

Since the direct implementation of the likelihood ratio is computationally costly, a sequential version was proposed in [5] based on updating the likelihood ratio at each time interval k. The equations of the adaptive sequential detector are:

$$\Lambda(\mathbf{X}_k) = \lambda(\mathbf{x}_k|\mathbf{X}_{k-1})\Lambda(\mathbf{X}_{k-1})$$

where

$$\begin{aligned}
\lambda(\mathbf{x}_k|\mathbf{X}_{k-1}) &= \sum_{m=1}^{M} \lambda(\mathbf{x}_k|m)P(m|\mathbf{X}_{k-1}, H_1) \\
\lambda(\mathbf{x}_k|m) &= \frac{P(\mathbf{x}_k|m, H_1)}{P(\mathbf{x}_k|H_0)} \\
P(m|\mathbf{X}_k, H_1) &= \frac{\lambda(\mathbf{x}_k|m)}{\lambda(\mathbf{x}_k|\mathbf{X}_{k-1})} P(m|\mathbf{X}_{k-1}, H_1)
\end{aligned}$$

These equations update the *a posteriori* probability distribution $P(m|\mathbf{X}_k, H_1)$ for the M orthogonal signals as well as the likelihood ratio $\lambda(\mathbf{X}_k)$. Therefore not only is it possible to detect the presence of a signal, but it is also possible to classify the incoming signal by selecting the signal m for which this probability is maximized.

In the results reported in this paper, idealized signal statistics are chosen. Hence the adaptive sequential receiver described above provides maximum achievable performance under ideal Gaussian noise conditions, and it is used as a benchmark for neural network performance. It should be noted however, that in general real data statistics can only be estimated empirically. Therefore the use of the likelihood ratio which results in complex receivers is not necessarily optimal with real data.

2 Neural Network Configurations

In this research, multilayer feedforward neural networks with sigmoid transfer function are used. The conventional backpropagation algorithm [3] is used for training the network. As can be observed from the adaptive sequential detector, it is important to use prior knowledge about data and incorporate it into the neural network scheme. This can be done by first processing the data using prior knowledge, and then feeding the outcome to the neural network that performs detection/classification. The M possible waveforms are known exactly prior to observation. Hence we use a correlator bank to preprocess data that will be fed into the neural network.

In the $k-th$ subinterval, the correlator output $\mathbf{y}_{k,m}$ for the recevied signal $\mathbf{x}_k$ is given as,

$$\mathbf{y}_{k,m}(t) = \int_{[(L-1)K+(k-1)]T}^{[(L-1)K+k]T} \mathbf{c}_m((t - [(L-1)K + (k-1)]T)\mathbf{x}_k(t)dt, \ m = 1, 2, ..., M. \tag{12}$$

for $m = 1, ..., M$.

Prior knowledge states that when the signal occurs, it does so at random in certain intervals of the frame; secondly noise is zero-mean AWG. Hence the data can be averaged over the subintervals before being processed by the neural network. Thus we may also use the following averaged data from the whole $L-th$ frame:

$$\mathbf{Y}_m(t) = \frac{1}{K}\sum_{k=1}^{L} \mathbf{y}_{k,m}, \ m = 1, ..., M. \tag{13}$$

so that instead of using (12) to calculate the correlator output, we will use a frame based correlator $\mathbf{Y}_m$.

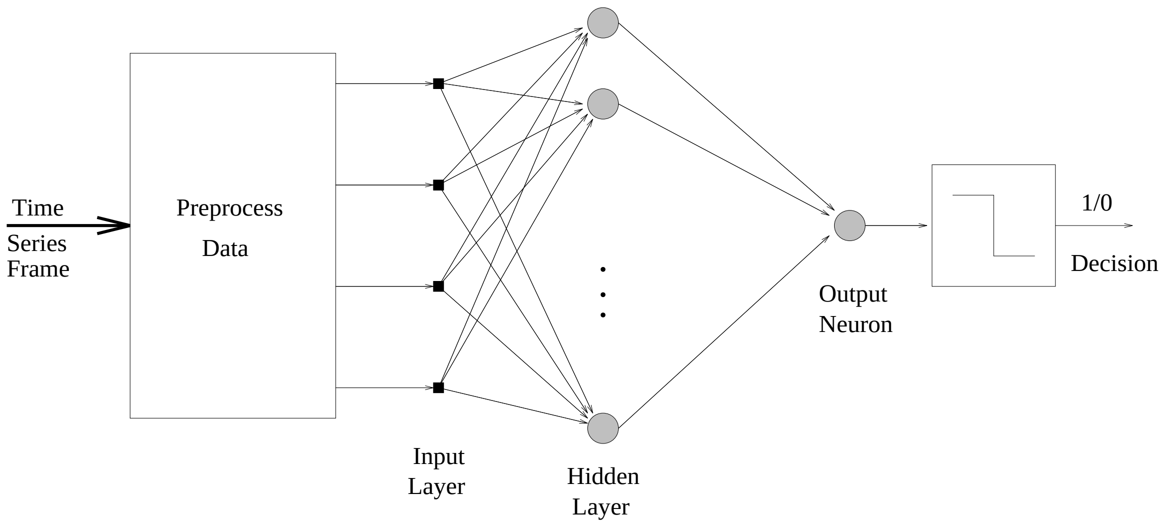

Figure 4: Multilayer feedforward detector

2.1 Feedforward Networks with One Hidden Layer

In this section we will discuss two conventional neural network configurations with a single hidden layer. The first one performs only detection, while the second performs joint detection/classification.

The detection scheme uses the network with a single output and M inputs which are the averaged correlator outputs (12). The network is trained to have a "high" output (1) when a single signal is present and "low" output (-1) for no signal. The neural network has one hidden layer with the assumption that it suffices to designate a decision region convex in the input vector space, due to the observation that when there is no signal, the input vector is unimodally distributed (WGN). The detection network is shown in Figure 4.

The classification network, which is shown in Figure 5, has $M+1$ output neurons. M of these correspond to each one of the possible signals, and one corresponding to the absence of any signal. The network is trained such that the $m-th$ output neuron is "high" when the $m-th$ signal is present, and "low" when the $m-th$ signal is absent. The $(M+1)^{st}$ neuron gives "high" when all signals are absent and "low" when a signal is present. The maximum of the output neuron values is chosen for decision. The weights of the network are trained so that the output neuron values approximate the *a posteriori* probabilities $\Pr(m, H_1|\mathbf{X}_k)$ and $\Pr(H_0|\mathbf{X}_k)$. The classifier has a single hidden layer for simplicity, implying that the decision region for each neuron is convex, as in detection, and the inputs are again the KM correlated outputs (12). Each forms a weighted sum of its inputs and produces an output which is a sigmoid function of this sum x of the form:

$$\mathrm{sg}(x) = \frac{2\alpha}{1+\exp(-\beta x)} - \alpha, \ \alpha > 0, \ \beta > 0, \tag{14}$$

as shown in Figure 6.

2.2 The XOR-Tree Configuration

A multilayer feed-forward neural network can also be designed with inspiration from a digital logic approach for this problem. The intuition behind this approach is the fact that if the data were completely

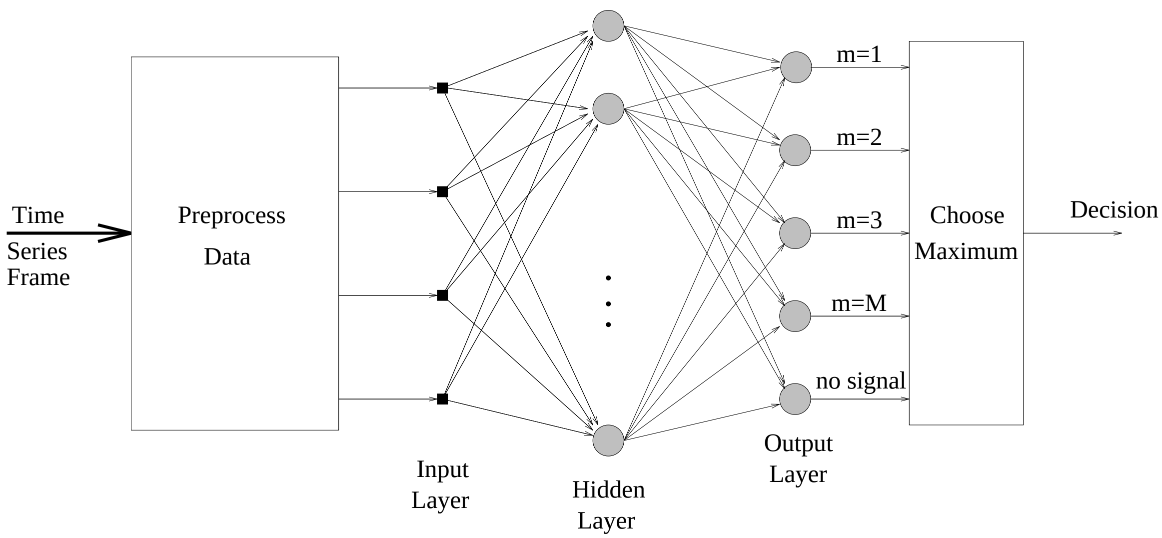

Figure 5: Multilayer feedforward classifier

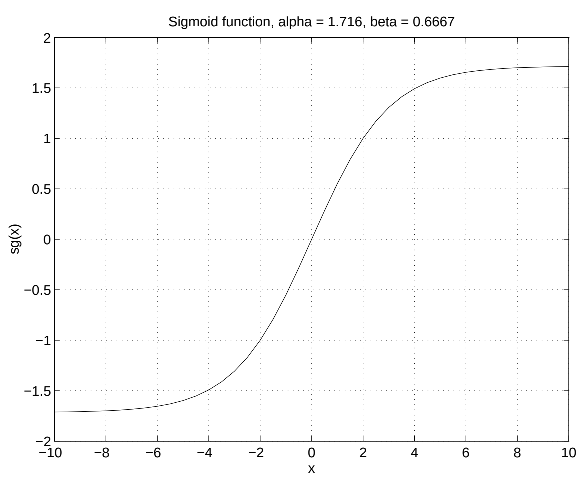

Figure 6: Sigmoid function with $\alpha = 1.716$ and $\beta = 2/3$, used in the multilayer neural networks of this research.

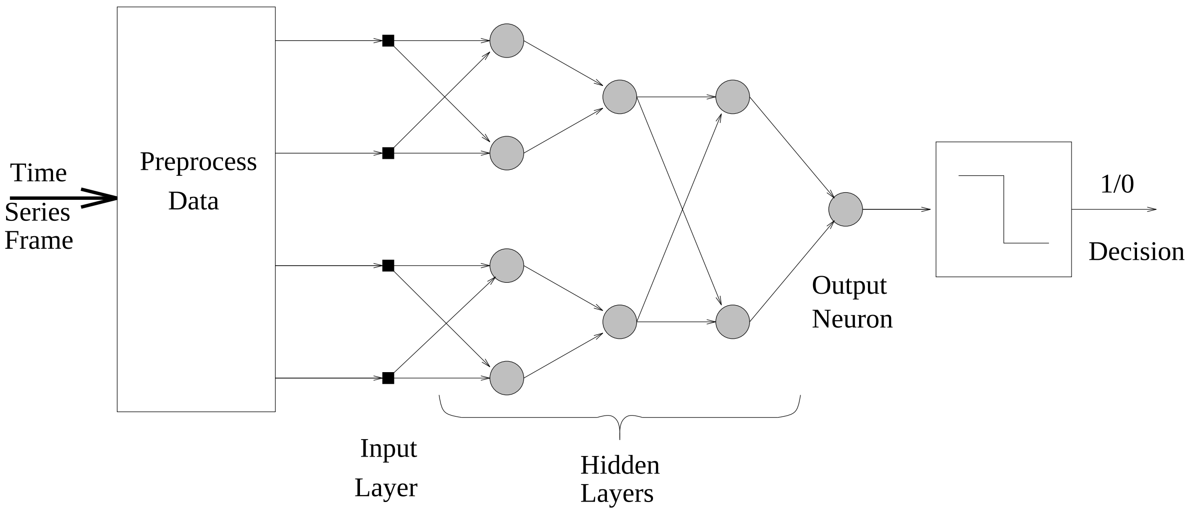

Figure 7: The XOR configuration for the feedforward neural network

deterministic (i.e. with no noise), then the output of the detection algorithm would be "1" if and only if there were a single high level in the set of averaged correlators (13), and the identity of that input would simply solve for the classification problem. Although various feed-forward structures of logical gates could have been used to implement this logical function, using the XOR (exclusive OR) function as the basic building bloc is particularly attractive.

When M is some power of 2, i.e. $M = 2^W$, we have a tree of XOR units of depth $W = log_2 M$ with M leaves and 2^{W+1} XOR gates to implement the logical function. As an example a two-stage structure composed of three XOR functions is illustrated in Figure 7 for $M = 4$. From a neural network perspective, although this is a multi-layer (rather than the commonly used three-layer) structure, it is attractive because of its tree structure which limits dependencies between successive levels of the network, while eliminating dependencies between gates in adjacent sub-trees. Furthermore, our idea is to implement the XOR gates as neural sub-networks so that the network as a whole can be trained to deal with the non-parametric noisy data. The weights and biases used in the ("small") neural networks to perform the XOR operation are used as initial weights for this network when it is trained with backpropagation learning.

We summarize the advantages of the $XOR - Tree$ as follows:

- If all inputs are deterministic and the M input signals are orthogonal, the output of the output will be 1 if and only if exactly one of the previously known signals is detected at the input. This would not be possible with another basic building bloc such as an AND or OR gate.
- As we show below, the learning complexity is comparable (and may be higher or lower depending on the relation between N and M) than that of a standard $M \times N \times 1$ three-layer network, since N (the size of the intermediate layer) needs to be larger than M in order to provide adequate non-linear mixing of the M input signals.
- The number of learning exemplars needed for the XOR network seems to be substantially lower than that of the other approaches. This has been experimentally observed, but it is intuitively evident because the XOR will solve the problem exactly when there is no noise at all.
- Finally, the XOR logical function viewed as a neural network has been extensively studied in the neural network literature (see [4] pp. 157-160, or [3, 8]).

2.2.1 Learning Time Complexity of the Neural Network Detectors

In this section we evaluate the time complexity of learning with the neural network architectures comparedAs a first step let us determine the learning complexity of the $XOR - Tree$ network.

Proposition 1 The time complexity of the learning algorithm for the $XOR - Tree$ network to detect the presence of one out of M input signals is $O(M^2)$.

Proof The $XOR - Tree$ architecture for M signals will have a depth of $W =\sim log_2 M$ (i.e. the depth is the maximum number of XOR units on a path running from an input leaf to the final output decision node). Consider now such a tree, and let C_l denote the time complexity of learning, for one input exemplar, of the subtree of depth l. Our problem is to compute C_W. Consider the standard feedforward backpropagation learning step [3] for some weight ω and some output neuron p', and let p be an immediate predecessor neuron of p'. Let $\omega_{p,p'}$ denote the weight linking neuron p to neuron p'. The algorithm step is:

$$\frac{\partial O_{p'}}{\partial x}|_{x=x_{p'}} \; if \; \omega = \omega_{p,p'} \quad (15)$$

$$\frac{\partial O_{p'}}{\partial \omega} = \omega_{p,p'} \frac{\partial O_p}{\partial x}|_{x=x_p} \; if \; \omega \neq \omega_{p,p'} \; and \; \omega \; is \; a \; predecessor \; of \; neuron \; p', \quad (16)$$

$$0, \quad otherwise, \quad (17)$$

where x_p and $x_{p'}$ refer to the input values to neuron p and p', respectively. As a consequence, we can relate the complexity C_l of learning all weights impacting the output of any $l - th$ level XOR unit, to the complexity C_{l-1} of learning all weights impacting the output of a $(l-1) - th$ unit as follows:

$$C_l = 2 + 2[2 + 2C_{l-1}], l \geq 1, \quad (18)$$

where the time complexity of operation (15) and (16) is taken to be 1, since the $(l-1) - th$ level XOR unit's output is the input to the following layer. As a consequence:

$$C_l = 6 + 4C_{l-1}, \quad (19)$$

or equivalently:

$$C_l = \sum_{i=0}^{l+1} 6.4^i$$

$$= 6\frac{1-4^{l+1}}{1-4} \sim 6.4^l$$

where we have set $C_0 = 0$ since the inputs to the first level XOR units will simply be the M signals, and will therefore not require any computation. Hence setting $W \sim log_2 M$, and $log(C_W) \sim log_2 6 + 2log_2 M$, we end up with$C_W \; 6M^2$, or $C_W = O(M^2)$, completing the proof.

Proposition 2 The time complexity of the learning algorithm for the $M \times N \times 1$ network to detect the presence of one out of M input signals is $O(MN)$.

Proof We follow the same lines as previously using equations (15), (16), but now we call $K(M)$ the learning complexity of the detector network single utput for M inputs, and notice that: $K(M) = N + NM$, completing the proof.

2.3 Learning Error Criteria

Although it is the most commonly used, the Mean Squared Error criterion is not the only metric that can be applied to backpropagation learning and it is possible to apply others that enhance the algorithm performance. Several criteria have been proposed in [9, 16, 17] and tested for signal detection in white Gaussian noise [18]. In particular:

- Least Mean Squares (LMS) formulated as,

$$\epsilon = \frac{1}{P}\sum_{p=1}^{P}\frac{1}{2}(o_p - \hat{o}_p)^2. \tag{20}$$

  Here, o_p is the desired output and $\hat{o}_p$ is the actual neural network output which is an estimate of the desired one. P is the number of training pairs. This criterion is widely used in backpropagation. It can be generalized for use in classification by summing all the squared outputs over all training pairs.

- El-Jaroudi & Makhoul Criterion (JM) [9] formulated as,

$$\epsilon = -\frac{1}{P}\sum_{p=1}^{P}\ln(1 - |o_p - \hat{o}_p|) \tag{21}$$

  Here, o_p is the desired output and $\hat{o}_p$ is the actual neural network output, P is the number of training pairs. With this criterion, small errors $|o_p - \hat{o}_p| << 1$ add linearly while large errors are deemphasised as opposed to LMS. It can also be generalized for use in classification by summing all log terms of output errors over all training pairs.

In the sequel we will test both of these error criteria, using the standard backpropagation algorithm [3, 4] for training the networks.

3 Performance evaluation with ideal and real noise

In this section we test the performance of the different detection classification algorithms both with ideal Gaussian noise, and with real noise collected from sonar signals provided by NATO. We will first describe the general testing scheme we have used, and then detail the results obtained for the two types of noise. Our results will show in particular that the XOR-nn provides significant improvement over the LR detector for the real noise. We consider the four different detection/classification algorithms for synchronous recurrent transients discussed earlier: the $XOR - Tree$ network (XOR-nn), the $2 - layer$ neural network with least mean squares error function (LMS-nn), the 2-layer network with the Jaroudi-Makhoul criterion (JM-nn), and the adaptive sequential likelihood ratio (LR) receiver.

The LMS-nn uses 40 neurons in the hidden layer. Notice here that $N = 40$ so that the learning complexity is much higher than that of the $XOR - Tree$ ($40M$ rather than $24M$). This layout has been used for detection with just one output neuron and for classification with $M + 1$ output neurons. A similar network architecture also been trained with the El Jaroudi-Makhoul criterion (JM-nn) and compared with conventional LMS learning. For the JM-nn, the hidden layer contains 20 neurons, so that its learning complexity is comparable to that of the $XOR - Tree$ network. The $XOR - Tree$ neural network (XOR-nn) has been trained with the conventional least mean squares algorithm for detection. Finally, the adaptive sequential likelihood ratio (LR) receiver has been used as a benchmark.

This receiever is based on the Bayesian approach and determines upper bounds for performance in this problem since statistics are known a priori.

Figure 2 shows the four ($M = 4$) signals that are used in these tests. Also there are four intervals in each frame ($K = 4$). These signals are obtained by using the Hadamard matrix which generates binary-valued orthogonal functions. In Figure 3 we show an example frame. The first plot shows how one of the M signals occurs in each sub-interval with the same probability, say $\nu = 0.5$. The second plot shows sample WGN. If H_0 occurs, this is the actual received signal. The last plot, on the other hand, shows a superposition of signal and noise which is observed in the case of H_1. In the simulations we report here, the absence or presence of a signal are equally likely. Furthermore, if a signal occurs the occurence of each waveform is equally likely throughout the simulations.

In detection, the last step of the algorithm is the choice of a threshold for the decision of presence/absence of the signal, where the objective is to minimize the probability of detection error. The best choice of the threshold for the LR receiver is $\frac{P(H_1)}{P(H_0)} = 1$ since both cases are equally likely. In the case of neural network receivers, the threshold is chosen as 0, since the networks are trained so that the output neuron value is 1 when the signal is present and -1 when the signal is absent, and because both cases are equally likely.

All neural network configurations, and the likelihood ratio receiver, have been tested on three different environmental settings, one for high SNR_{occ} (20 dB) and low probability of signal occurence (ν= 0.1), another with lower-range SNR_{occ} (10 dB) and low probability of occurence (ν= 0.1), and finally with mid-range SNR_{occ} (0 dB) and mid-range probability of occurence (ν= 0.5). The number of possible signals was $M = 4$ and these signals were generated using the Hadamard matrix, each interval was of duration $T = 4$, and each frame contained $K = 40$ intervals.

3.1 Testing with ideal Gaussian noise

Table 1 shows detection performance (i.e. the Probability of Detection Error P_e^D) for all the configurations and settings which were tested. For the LMS-nn and JM-nn networks, training was carried out 26 times with random weight initialization, and each were tested with 1100 frames. The initial weights for $XOR - Tree$ were chosen as the XOR-function weights [4] and once convergence was observed in training, the network was tested with 26100 intervals. The results can be summarized as follows:

- We see that the $XOR - Tree$ offers a large improvement over the LMS-nn, and that the JM-nn outperforms the LMS-nn due to its improved error criterion.
- The JM-nn takes considerably longer to train than the other networks.
- All the neural network configurations' performance is close to the upper bound provided by the Likelihhod Ratio for the "lowest-SNR_{occ}, highest probability of occurence" settings ($SNR_{occ} = 0$ dB, $\nu = 0.5$), while they exhibit poor performance when the probability of occurence is low even when the SNR_{occ} is higher (SNR_{occ}=10 dB, $\nu = 0.1$). This appears to be due to properties of the preprocessor of the neural network, since the averaging technique is efficient for low SNR_{occ}, but is less effective elsewhere.

On the bottom part of Table 1 we also show the performance data of each neural network with respect to the criterion it has been trained for. These results parallel those obtained for P_e^D. It can be seen that the values obtained for these errors are not as low as would be expected. This is a consequence of the random occurence of the waveform as is implied by the decrease in LMS or JM error with increasing probability of signal occurence.

Probability of Detection Error P_e^D			
	SNR_{occ}=20 dB, ν= 0.1	SNR_{occ}=10 dB, ν= 0.1	SNR_{occ} = 0 dB, ν = 0.5
LR	0.0072	0.0842	0.0993
XOR-nn	0.1025	0.3036	0.1769
LMS-nn	0.1563	0.3433	0.2612
JM-nn	0.1576	0.3231	0.1695
Root Mean Squared Error			
XOR-nn	0.2775	0.4484	0.3670
LMS-nn	0.6896	0.9284	0.3843
Jaroudi Makhoul Error			
JM-nn	0.3661	0.5981	0.3834

Table 1: Comparisons of detector performance. The two-layer network trained for LMS has 40 hidden layer neurons. The network trained for JM has 20 neurons.

Probability of Classification Error P_e^C			
	SNR_{occ}=20 dB, ν= 0.1	SNR_{occ}=10 dB, ν= 0.1	SNR_{occ} = 0 dB, ν = 0.5
LR	0.0072	0.0876	0.1063
LMS-nn	0.1311	0.3644	0.2178
JM-nn	0.1364	0.3855	0.1609
Root Mean Squared Error			
LMS-nn	0.4634	0.6539	0.5667
Jaroudi-Makhoul Error			
JM-nn	0.1741	0.3455	0.1752

Table 2: Comparison of classifier performance. The 2-layer network trained for LMS has 40 hidden layer neurons, while the one trained for JM has 20 neurons.

Table 2 shows performance data P_e^C (Probability of Classification Error) close to those of the detection performance. Moreover the trends observed in classification are similar to the trends in detection. Given that one of the classes of classifiers ("no signal" class) is the same as the H_0 hypothesis, and that all the M signal classes are included in the H_1 hypothesis, the results confirm that differentiating the "no signal" class from the others is generally harder than differentiating the M signals from each other.

3.2 Testing the LR and the XOR-Tree with Measured Real Non-White Noise

We now evaluate our proposed XOR-nn with real noise, and the simulated signal discussed earlier. The noise data has been retreived from a sonar experiment named "SACLANT" which can be obtained from the web site with URL "http://spib.rice.edu/spib/saclant.html", and has been extracted from time waveform of sensor 17. More specifically, the time waveform of sensor 17 of SACLANT data named 'P2701' measured on October 27, 1993 has been used. Only sections of this time waveform labeled as noise have been used. The signal format is the exact same synchrounous recurrent waveform format used in the previous simulations. The statistical characteristics of this noise are shown in Figures 8 and 9, which indicate that although the noise appears to be Gaussian, its autocorrelation function demonstrates periodic properties indicating that it is not white.

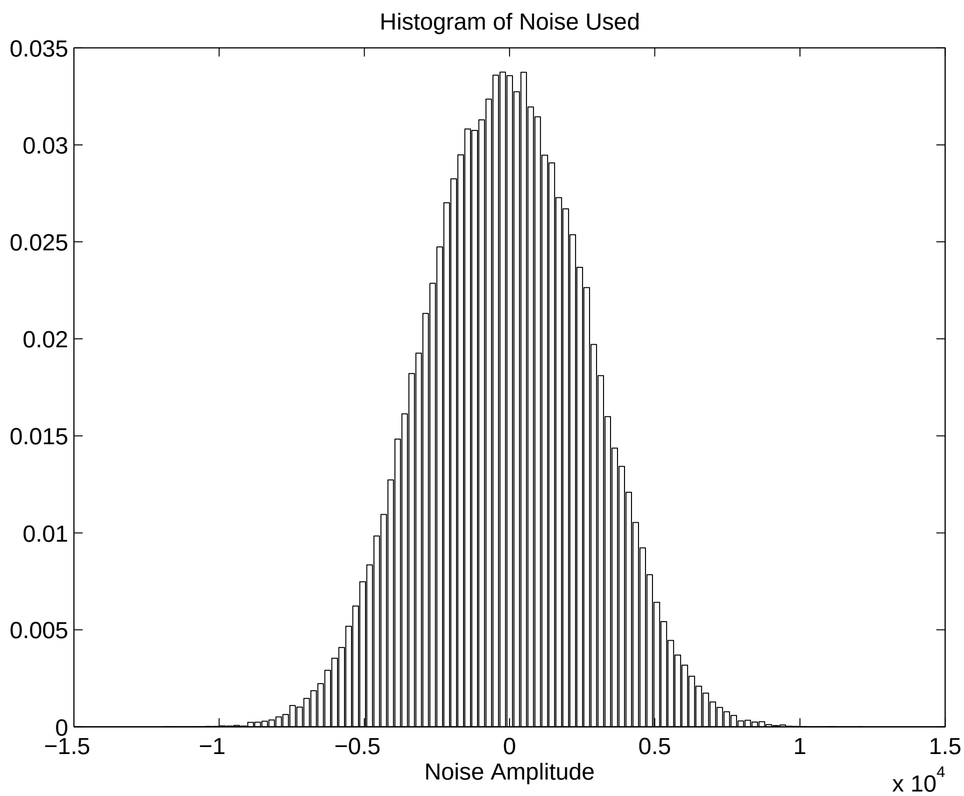

Figure 8: Histogram of real noise exhibiting Gaussian distribution

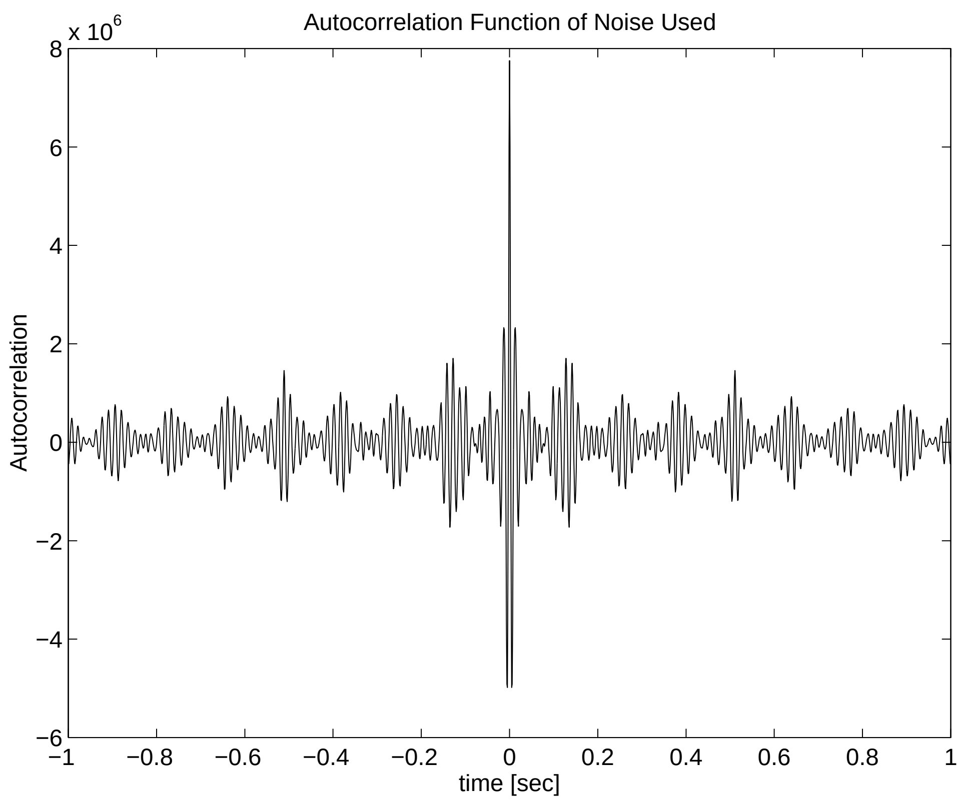

Figure 9: Autocorrelation function of real noise exhibiting non-white periodic time dependencies

Probability of Error in Detection			
	SNR=20 dB, ν= 0.1	SNR=10 dB, ν= 0.1	SNR = 0 dB, ν = 0.5
LR	0.0045	0.3199	0.1868
XOR-nn	0.0660	0.1018	0.0000
Root Mean Squared Error			
XOR-nn	0.3542	0.3865	0.4348

Table 3: Comparison of detector performance for LR and XOR-nn with experimentally collected real noise. The LR test used the noise variance estimated from the data. For the XOR-nn, 894 observations of the signal plus noise were used for testing, after training with 956 observations of the real noise.

The XOR-nn receiver was trained with an initial part of the data, and tested using the second part. The training data consisted of 956 observations, and the testing data consisted of 894 observations. The SNR of the data was controlled by scaling the real noise input to control its measured variance.

We do not report results on this real noise data for the other neural network techniques discussed earlier, simply because the available training sequence was not sufficiently long to get useful results for the other network models. This shows clearly that the XOR-nn outperforms the previous neural network methods not only in accuracy (as was shown with simulated noise), but also in its need for significantly shorter training sequences.

The Likelihood Ratio (LR) receiver was also tested with the second half of the data which consisted of 894 observations. However this likelihood ratio test assumes a white gaussian noise model, which does not accurately describe the nature of real noise used. Histogram analysis on noise data shows that it is almost of zero-mean Gaussian nature. However the autocorrelation function shows that this noise is not actually white since the measured autocorrelation is consistently non-zero at regular intervals, while autocorrelation for white noise would consist of a single peak at t=0.

The results obtained from these tests are summarized in Table 3. They show that the XOR-nn network outperforms the Likelihood Ratio when the SNR is low (10 *db* and 0 *db*), while the opposite is observed for a higher SNR of 20 *db*. However, in the case of SNR=20 *db* the probabilities of detection error are sufficiently small in both cases (0.0054 for the LR detector and 0.0660 for the XOR-nn) that the difference is less significant than it appears. Indeed, since we are dealing with a relatively small (894) number of tests, dictated by the duration of the real noise data, the confidence intervals associated with estimating these probabilities will also be quite low. Thus as we would have expected, the XOR-nn which does not does not require a model of the noise and uses training data instead, is robust and effective both for high and low SNR values.

4 Conclusions and Further Work

In this paper three neural network configurations have been tested for the detection of recurrent transient signals in noise: an $XOR - Tree$ structure, and a conventional three layer structure trained with the Least Means Squared Error criterion and with the Jaroudi-Makhoul criterion. It has been shown that the $XOR - -tree$ configuration of a multilayer feedforward neural network offers considerable improvement over conventional neural network designs, both in terms of performance (due to the number of hidden layers) and simplicity of structure (due to the locally connected network structure). This design results from an initial logic design, and offers significant improvements. Of the two error criteria which have been tested for the learning algorithm's convergence, the Jaroudi-Makhoul (JM) criterion appears to yield higher detection/classification performance over the conventional Least Mean Squared Error (LMS) criterion, as previously shown in [18]. However, the networks are harder to train using the JM crietrion especially when the error magnitude is large; hence the algorithm needs improvement in terms of its initialization.

As a result of this research an efficient neural network detector for synchronous recurrent networks can be proposed using the $XOR - Tree$ configuration which would be trained in two stages, initially with LMS and then with JM. Because of the loss in performance due to random signal occurence for all the neural networks, and in view of the structure of the classical likelihood ratio detector/classifier, we suggest that an extension of our approach where the neural network structure does not average the intervals over a frame (i.e observation interval), but instead processes each interval sequentially and perhaps also uses feedback, is worth pursuing in future work.

References

[1] H.L. Van Trees, *Detection, Estimation and Modulation Theory: Part I*, John Wiley & Sons, New York, 1968

[2] A.D. Whalen *Detection of Signals in Noise*, Academic Press, 111 Fifth Avenue, New York, NY, 1971.

[3] D.E. Rumelhart and J.L. McClelland (1986) *"Parallel Distributed Processing"*, Volumes 1 & 2. MIT Press, 1986.

[4] S. Haykin, *Neural Networks, a Comprehensive Foundation*, Macmillan Coll. Publ. Co., 866 Third Ave., New York, NY, 10022, 1994.

[5] L.W. Nolte, "Adaptive optimum detection: synchronous recurrent transients", *Journal of Acoustical Society of America*, Vol. 44 No 1, pp 224-239, July 1968.

[6] C.P. Hatsell and L.W. Nolte, "On transient and periodic signal detection in time-varying noise power", *IEEE Trans. on Aerospace and Electronic Systems*, Vol. AES-7, No 6, November 1971.

[7] E. Gelenbe "Réseaux stochastiques ouverts avec clients négatifs et positifs, et réseaux neuronaux", *Comptes-Rendus Acad. Sci. Paris*, t. 309, Série II (1989), 979-982.

[8] E. Gelenbe "Random neural networks with positive and negative signals and product form solution", *Neural Computation*, Vol. 1, No. 4, pp 502-510, 1989.

[9] A. El-Jaroudi and J. Makhoul, "A new error criterion for posterior probability estimation with neural nets", *Proceedings of the Int. Joint Conf. on Neural Networks*, Vol. 1, No 5, pp 185-192, 1990.

[10] V. Atalay, E. Gelenbe, N. Yalabik "The random neural network model for texture generation", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 6, No. 1, pp 131-141, 1992.

[11] V. Atalay, E. Gelenbe "Parallel algorithm for colour texture generation using the random neural network model", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 6, No. No. 2 & 3, pp 437-446, 1992.

[12] E. Gelenbe "Learning in the recurrent random network", *Neural Computation*, 5, pp 154-164, 1993.

[13] E. Gelenbe, T. Feng, K.R.R. Krishnan "Neural network methods for volumetric magnetic resonance imaging of the human brain," *Proceedings of the IEEE*, Vol. 84, No. 10, pp. 1488–1496, October 1996.

[14] C. Cramer, E. Gelenbe, H. Bakircioglu "Low bit rate video compression with neural networks and temporal subsampling," *Proceedings of the IEEE*, Vol. 84, No. 10, pp. 1529–1543, October 1996.

[15] E. Gelenbe, A. Ghanwani, V. Srinivasan "Improved neural heuristics for multicast routing", *IEEE Journal of Selected Areas of Communications*, Vol. 15, No. 2, pp. 147-155, February 1997.

[16] S. Amari, "Backpropagation and stochastic gradient methods", *Neurocomputing*, Vol. 5, pp 185-196, 1993.

[17] B.A. Telfer and H.H. Szu, "Energy functions for minimizing misclassification error with minimum complexity networks", *Proceedings of the Int. Joint Conf. on Neural Networks*, Vol. 4, pp 214-219, 1992.

[18] D. Andina, J.L. Sanz-Gonzalez, J.A. Jimenez-Pajares, "A comparison of criterion functions for a neural network applied to binary detection", *Proceedings of the ICNN*, Vol. 1, pp 329-333, November 1995.

[19] F.-L. Luo, R. Unbehauen "Applied Neural Networks for Signal Processing", Cambridge Univ. Press, 1997.

[20] H. Bakircıoglu, E. Gelenbe, T. Koçak, "Image processing with the Random Neural Network model", Invited Paper, *Proc. 13th IEEE International Conference on Digital Signal Processing (DSP97)*, Santorini, Greece, July 1997.