

Amir Hussain

Department of Electronic Engineering and Physics,
University of Paisley, High St., Paisley, PA1 2BE, U.K.
email: huss_ee0@paisley.ac.uk

ABSTRACT

In this paper a new two-layer linear-in-the-parameters feedforward network termed the Functionally Expanded Neural Network (FENN) is presented, together with its design strategy and learning algorithm. It is essentially a hybrid neural network incorporating a variety of non-linear basis functions within its single hidden layer which emulate other universal approximators employed in the conventional Multi-Layered Perceptron (MLP), Radial Basis Function (RBF) and Volterra Neural Networks (VNN). The FENN's output error surface is shown to be uni-modal allowing high speed single run learning. A simple strategy based on an iterative pruning re-training scheme coupled with statistical model validation tests is proposed for pruning the FENN. Both simulated chaotic (Mackey Glass time series) and real-world noisy, highly non-stationary (sunspot) time series are used to illustrate the superior modeling and prediction performance of the FENN compared with other recently reported, more complex neural network based predictor models.

1. INTRODUCTION

Over the past decade, there has been an increasing interest in the use of Artificial Neural Networks (ANNs) for solving complex real-world problems [1-13]. This is mainly due to their ability to effectively deal with non-linearity, non-stationarity and non-Gaussianity [1]. The modeling and analysis of so-called chaotic processes has also recently attracted the attention of many researchers [1,4-11]. Deterministic chaos is characterized by an exponential divergence of nearby trajectories [7]. Since the problem of time series prediction is synonymous with modelling of the underlying physical mechanism responsible for its generation, there are two related consequences. Firstly, chaos precludes any long-term predictability since the uncertainty of the prediction increases exponentially with time. Secondly, on the other hand an apparently random time series might have been produced by a deterministic chaotic system and thus be predictable in the short term. A prediction algorithm for chaotic systems thus has to capture the short term structure of the time series [6].

The short-term structure of chaotic behaviour can be captured by expressing the present value of the chaotic time series sample, as a function of the previous d values of the time series: $y(k) = f(y(k-1), \dots, y(k-d))$, where the vector $(y(k-1), \dots, y(k-d))$ lies in the d -dimensional state space [5]. An

efficient method of fitting the non-linear function $f(\cdot)$ is to use a feed-forward neural network predictor with a single output [5-8]; the inputs to the network being the observation vector $(y(k-1), \dots, y(k-d))$. In real-world chaotic time-series processes, intrinsic noise will be present and the task of the predictor will be to reconstruct $f(\cdot)$ without modeling the noise.

Two well known feedforward ANNs are the Multi-Layered Perceptron (MLP) and the Radial Basis Function (RBF) networks, both of which have been shown to be capable of forming an arbitrarily close approximation to any continuous non-linear mapping [3]. Consequently, both have to-date been successfully employed for approximating $f(\cdot)$ [1,4-11]. However, the MLP has a highly non-linear in the parameters structure, and requires computationally expensive non-linear learning algorithms (such as back-propagation) which may converge to local minimum solutions [3]. The RBF network has a linear in the parameters structure giving relative advantages of ease of analysis and rapid learning. However, it suffers the drawback of requiring a prohibitively large number of basis functions to cover high dimensional input spaces [7].

The topology of the RBF can be considered to be very similar to that of a two-layered MLP. The primary difference between the two structures is in the nature of their basis functions. The hidden layer nodes in the MLP employ sigmoidal type basis functions (which are non-zero over an infinitely large region of the input space); whereas the basis functions in the RBF network cover only small localized regions. Hush [3] has recently shown that some problems, such as functional approximation can be solved more efficiently with sigmoidal type basis functions; while others such as classification problems are more amenable to localized (e.g. Gaussian type) basis functions.

This paper describes a new unified approach which combines both these types of basis functions within a *single* neural network *layer*, so that the approximating capabilities of both the MLP and the RBF networks can be employed. The approach yields a new linear-in the-parameters feedforward neural network termed the Functionally Expanded Neural Network (FENN). A general design strategy is presented for specifying the type and number of basis functions within the network's single hidden layer, for an arbitrary number of network inputs. The FENN's output error surface is shown to be uni-modal allowing high speed single run learning. An exponentially weighted recursive least squares based learning algorithm is employed for updating its output layer weights. A simple strategy based on an iterative pruning-retraining scheme coupled with statistical model validity tests has been

proposed for pruning the size of the FENN. The new structure is shown to be highly efficient in the modeling of both simulated chaotic, and real world noisy time-series processes, and its performance is compared with other recently reported neural network based predictor models. Two simulation examples are presented using the chaotic Mackey-Glass Equation, and real world sunspot data.

2. THE FENN STRUCTURE

The complete two-layer, multiple-input multiple-output FENN is shown in Figure 1. It comprises an input functional expander within its single hidden layer, and an output layer. The FENN functional expander performs a non-linear transformation which maps the n -dimensional input space onto a new non-linear hidden space of increased dimension N . The choice of basis functions to be employed in the functional expander has been discussed in [2,12] and is summarized in the design strategy below. The output layer of the $(n, M; m)$ FENN comprises a set of m linear combiners. It is interesting to note that the RBF network with fixed non-linear hidden layer basis functions or centres (and widths) can be regarded as a FENN. The linear-in-the-parameters Volterra Neural Network (VNN) which employs a purely polynomial expansion of its inputs [12] can also be considered to be a special case of the FENN, in which the number of polynomial expansion terms grow exponentially with increasing input dimensions. The conventional Functional-Link Network (FLN) [13] also resembles the above FENN in terms of employing a functionally expanded input model. However, the FLN possesses a non-linear in the parameters structure (as its output layer is essentially a Perceptron requiring the non-linear Delta learning Rule for its weight updates); and the nature of its expanded input functions also differ. The functional expansion model employed in the FENN is unique in that it employs a combination of non-linear basis functions that emulate a variety of other universal approximators such as the squashing type sigmoidal, Gaussian bell shaped and polynomial-subset activation functions. These have been shown to significantly enhance the approximation ability of the FENN in the modeling of a very wide class of non-linear dynamical processes [2,11].

2.1 Design Strategy

Expand the input vector $[x_1(k) \dots x_n(k)]$ for any number n of FENN inputs (all normalized to within the range $(+1, -1)$), using the following expansion model:

$F(k)$ = sum of the following (linear and non-linear) N components:

1. zero-order term (resulting in 1 dc term).
2. original input terms $x_1 \dots x_n$ (resulting in n terms). These terms will enable modeling of linear systems.
3. sine expansion of the n inputs, comprising $\sin(x_i)$, $\sin(2x_i)$ and $\sin(3x_i)$ terms, for $i=1, \dots, n$ (resulting in a total of $3n$ terms). These terms emulate squashing type sigmoidal basis functions.
4. cosine expansion of the n inputs comprising $\cos(x_i)$, $\cos(2x_i)$ and $\cos(3x_i)$ terms, for $i=1, \dots, n$ (resulting in a total of $3n$ terms). These functions emulate Gaussian like basis functions of fixed widths.

5. product of each input with the sine and cosine functions of other inputs comprising $x_i \sin(x_j)$ and $x_i \cos(x_j)$ terms (for $i \neq j, i, j=1, \dots, n$) giving a total of $2n(n-1)$ terms. These terms emulate multi-quadratic and sigmoidal type functions respectively
6. outer-product expansion of the n inputs resulting in total of $(P_2^n + P_3^n + \dots + P_{n-1}^n + 1)$ terms for n greater than two inputs, with $P_m^n = [n! / (n-m)!m!]$ where $!$ denotes factorial. Note that for $n=2$ inputs the outer-product expansion will result in 1 term $(x_1 x_2)$, and for $n=1$ there will be no outer product terms. These higher order outer-product terms can be considered to be a polynomial expansion of the inputs without the n -th power of the inputs [11].

Hence, in general for n inputs, the FENN functional expansion model $F(k)$ will comprise a total of:

$N = (1 + 2n^2 + 4n + \sum_{i=1}^n P_i^n)$ terms. That is for $n=1, N=8$; $n=2, N=20$; $n=3, N=38$; $n=4, N=64$ and so on. Note that the functional expansion model of the FENN is extremely flexible since virtually any function of the input such as $\tanh(\cdot)$, $\exp(\cdot)$, etc. can also be employed. In practice, physical knowledge of the non-linear system to be identified can also be incorporated within the input functional expansion model. If no *a priori* system knowledge is available, the FENN approximation can be further improved by addition of higher order polynomial terms from the Volterra series expansion in the input functional expansion model. Thus, the overall FENN structure can be seen to possess non-linear approximation ability by virtue of the input non-linear functional expander, and yet learning of its output layer weights is a linear problem. It is this latter characteristic of the FENN that provides the real motivation for exploiting its use in complex real-world non-linear dynamical system modeling applications.

2.2 Learning Algorithm

- (1) Compute the $i=1, \dots, m$ FENN outputs at time k , as

$$y_i(k) = F^T(k) W_i(k-1)$$

where $F(k)$ defines the $[N \times 1]$ hidden layer vector comprising the non-linear functional terms, and $W_i(k-1)$ is the $[N \times 1]$ weight vector of the i -th output.

- (2) The output prediction error for each FENN output is:

$$e_i(k) = d_i(k) - y_i(k)$$

where $d_i(k)$ is the i -th desired output. The Mean Squared Error (MSE) is therefore:

$$E(e_i(k)^2) = E(d_i(k)^2) - 2W_i(k-1)^T E(d_i(k)F(k)) + W_i(k-1)^T E(F(k)F(k)^T) W_i(k-1)$$

where $E(\cdot)$ denotes the expectation operator and T denotes matrix transpose. The corresponding minimum MSE (MMSE) for the FENN can thus be readily written as:

$MMSE = E(d_i(k)^2) - E(d_i(k)F(k))^T E(F(k)F(k)^T)^{-1} E(d_i(k)F(k))$ where superscript -1 denotes matrix inverse. The above MMSE also includes the best linear (Wiener) MMSE for the case of $F(k)$ =input vector $[x_1(k) \dots x_n(k)]$, that is, only linear functions of the inputs. The advantage of this particular FENN structure is that linear adaptive filter theory can be readily applied for on-line adaptation.

The quadratic form of the MSE expression above with respect to the FENN weights ensures that there will be no local minima and so fast and certain convergence may be obtained in practice. The FENN weights, for each of the m outputs, can be

updated using the exponentially weighted Recursive Least Squares (RLS) algorithm as follows:

(3) Update the inverse of the correlation matrix of the input functional expansion vector:

$$P(k) = [F(k)F(k)^T]^{-1} = 1/\lambda [P(k-1) - P(k-1)F(k)F(k)^TP(k-1)/(\lambda + F(k)^TP(k-1)F(k))]$$

where λ is the forgetting factor (≤ 1), which introduces exponential weighting into past data.

(4) Update the output layer weights using:

$$W_i(k) = W_i(k-1) + P(k) F(k) e_i(k) \quad \text{for } i=1, \dots, m$$

Numerically robust versions of the RLS can be used instead of the above. The simpler Least Mean Squares (LMS) algorithm which is a stochastic gradient algorithm can also be used for updating the output layer weights as follows:

$$W_i(k) = W_i(k-1) + \mu e_i(k) F(k)$$

where μ controls the convergence rate. However, the rate of convergence of the LMS algorithm is dependent on the spread of the eigenvalues of the input expansion model's autocorrelation matrix, $E(F(k)F(k)^T)$, with an increase in eigenvalue spread dictating a slower convergence rate [11]. The RLS algorithm will converge more rapidly but at the expense of an increased computational complexity, $O(N^2)$ compared to $O(N)$. Other Fast RLS (FRLS) algorithms of reduced complexity can also be readily applied to train the above FENN.

Thus, once the full expansion model at the input hidden layer of the FENN has been specified, the exponentially weighted RLS algorithm can then be used to provide an efficient means for real time adaptation of the network weights. This will give the FENN a significant advantage over the multi-layered neural network structures such as the MLP in recursive identification applications.

3. SIMULATION RESULTS

3.1 Modeling of Real Sunspots

Following Tong [9], Weigend [7], Svaver [10] and McDonnell [8], a (2,20;1)FENN was trained on the annual sunspot series for the years 1700-1920. The two FENN inputs having been normalized to within (+1,-1) were expanded into twenty functional terms. The FENN was pruned by employing an iterative pruning re-training scheme to successively remove the insignificant basis functions. Output error auto-correlation and Chi-squared statistic based model validation tests [11,14] were employed at each stage in order to validate the pruned FENN model over the training set. The pruned and trained (2,14;1)FENN one-step predictor model of the sunspot series which satisfied all the correlation and chi-squared model validity tests [11] is illustrated below:

$$\begin{aligned} \hat{y}(k) = & y(k-1) + 1.16y(k-2) + 1.23\sin(y(k-2)) - \\ & 2.1\sin(2y(k-1)) + 1.98\sin(2y(k-2)) - 1.14\cos(y(k-2)) \\ & - 1.5\cos(2y(k-1)) + 1.54\sin(3y(k-1)) - 1.1\sin(3y(k-2)) \\ & - 0.66y(k-1)\sin(y(k-2)) - 0.11y(k-1)\cos(y(k-2)) \\ & - 2.78y(k-2)\sin(y(k-1)) - 3.8y(k-2)\cos(y(k-1)) + 2.7 \end{aligned} \quad (3.1)$$

where $\hat{y}(k)$ is the one-step FENN prediction of the current sunspot sample $y(k)$, based on the previous two sunspot time series samples $[y(k-1) y(k-2)]$. The one-step predictions of the above evolved (2,14;1)FENN predictor model were evaluated on the sunspot series for test years 1956-1979 (which are

known to be atypical of the entire time series [8], and most difficult to predict on account of their highly non-stationary nature), in order to test its generalization ability. The average relative variance (*arv*) [8] (defined as the ratio of the Mean Squared Error to the estimated variance) achieved by the FENN one-step predictor model is compared in Table 1 with other published results. The TAR denotes a Threshold Auto-Regressive Model, and SLP represents a Single-Layered IIR type Perceptron. As can be seen from Table 1, the new FENN model outperforms the other techniques both in terms of prediction ability and relative computational requirements. An examination of the FENN predictor model (equation 3.1 above) also reveals the respective contributions of the various proposed non-linear basis functions which are primarily responsible for the superior FENN performance over the other more complex neural network models (all of which required information from at least the previous 5 sunspot samples).

Model	FENN	TAR [9]	MLP [7]	MLP [10]	SLP [8]
Inputs	2	8	12	5	6
No. of Parametr.	14	16	43	16	23
<i>arv</i> (1956-1979)	0.18	0.28	0.35	0.35	0.37

Table 1: Performance comparison of various single step predictor models on the sunspot test data.

3.2 Modeling of simulated chaotic Mackey Glass Equation:

One thousand samples were generated using the following Mackey Glass equation [8]:

$$\frac{d(y(k))}{d(k)} = \frac{0.2 y(k-30)}{1 + y^{10}(k-30)} - 0.1 y(k)$$

A one-step predictor model based on a fully expanded (2,20;1)FENN was devised and trained on the first 500 samples. The fully expanded model was then pruned iteratively to yield a (2,4;1)FENN predictor model:

$$\begin{aligned} \hat{y}(k) = & 1.99y(k-1) - 0.88\sin(y(k-2)) + 0.74y(k-1)\sin(y(k-2)) \\ & + 0.05y(k-2)\sin(y(k-1)) \end{aligned} \quad (3.2)$$

The *arv* performance measure achieved by the above FENN one-step predictor on the subsequent 500 sample Mackey Glass test data is compared in Table 2 with that reported by McDonnell [8], who used a 17-term recurrent one-step predictor model that evolved from a parent Recurrent IIR Perceptron after five thousand generations of an optimisation process incorporating a computationally expensive, complex multi-agent stochastic search technique. As can be seen from Table 2, the 4-term FENN predictor model of equation 3.2 comprising a weighted linear term, a non-linear sigmoidal shaped term, and 2 multi-quadratic shaped terms outperforms the more complex recurrent predictor model reported in [8].

Model	FENN	IIR Perceptron [9]
<i>arv</i> (1-step predictions)	0.0012 (comprising 4 parameters)	0.0025 (comprising 17 parameters)
<i>arv</i> (2-step predictions)	0.0016 (7 parameters)	0.0070 (17 parameters)

Table 2: *arv* performance comparison of one-step and two step non-linear predictor models on a 500 sample test set.

A (2,20;1)FENN two-step ahead predictor model was also devised and trained on the same 500 sample training data set. Upon pruning, an optimal (2,7;1)FENN two-step predictor model evolved [11] which is illustrated below:

$$\begin{aligned}\hat{y}(k) = & 0.96 \hat{y}(k-1) - 0.82 \sin(2 \hat{y}(k-1)) + 0.57 \sin(2y(k-2)) \\ & + 0.07 \cos(2 \hat{y}(k-1)) - 0.95 \hat{y}(k-1) \cos(y(k-2)) \\ & + 0.04 y(k-2) \sin(\hat{y}(k-1)) + 1.42 y(k-2) \cos(\hat{y}(k-1)) \quad (3.5)\end{aligned}$$

The *arv* performance of the above 7-term FENN two-step predictor model on the 500 sample test data is given in Table 2. The FENN predictor can be seen to significantly outperform the corresponding 17-term recurrent predictor model of [8].

4. CONCLUSIONS

A new feedforward two-layer linear in the parameters neural network termed the FENN has been presented, together with its design strategy and learning algorithm. It is basically a hybrid neural network incorporating a set of non-linear basis functions which have the effect of simulating sigmoidal shaped, Gaussian shaped and polynomial subset activation functions simultaneously. The use of an iterative pruning-retraining strategy coupled with statistical model validation tests was shown to result in parsimonious FENN predictor models comprising the most significant of the proposed non-linear basis functions. The final evolved FENN models outperformed other more complex neural network predictors in the modeling of both the simulated chaotic Mackey Glass time series and real-world noisy, non-stationary sunspot time series, both in terms of non-linear prediction ability and relative computational requirements. The respective contributions of the various proposed non-linear basis functions responsible for the superior FENN performance were also highlighted in the case studies. An added benefit of the new FENN, like the VNN, is that the structures of the corresponding FENN predictor models may also provide highly useful insights into the physics of the underlying unknown non-linear system dynamics. In other tests to be reported elsewhere, the design strategy presented for the FENN structure, the least squares based learning algorithm and the pruning strategy have also resulted [11] in highly efficient FENN predictor models of a variety of other complex, simulated and real-world non-linear time series processes including the chaotic logistic, Henon and Lorenz maps, SISO and MIMO NARX processes; and real-world stock market data, real laser time series and actual speech signals. Recently, a Recurrent FENN employing local output feedback has also been developed [11] and shown to outperform the above feedforward FENN in the modeling of certain types of non-linear dynamical processes. Currently, the FENN is also being investigated for performing non-linear processing in full-band and multi-band speech enhancement systems [15].

5. ACKNOWLEDGEMENTS

The author is grateful to Dr.J.J.Soraghan and Prof.T.S Durrani of the University of Strathclyde, and Prof. D.R. Campbell of the University of Paisley for numerous stimulating discussions. The author also acknowledges the financial support of U.K Engineering and Physical Sciences Research Council.

6. REFERENCES

- [1] S.Haykin, *Neural Networks expand Signal Processing's Horizons*, Signal Processing Mag., pp.25-49, Mar'1996.
- [2] A.Hussain, J.J.Soraghan, T.S.Durrani, *A New Hybrid Neural Network for Non-linear Time Series Modeling*, Neurovest Journal, (Ed.) R.B. Caldwell, Special Issue Hybrid Neural Networks for Financial Time Series Forecasting, 5(1), Finance & Technology Publishing, U.S.A, Jan'1997.
- [3] D.R.Hush, B.G.Horne, *Progress in Supervised Neural Networks: What's new since Lippmann*, IEEE Signal Processing Magazine, pp.9-39, Jan'1993.
- [4] O.E.B.Nielsen, J.L.Jensen, W.S.Kendall, (Eds.), *Networks & Chaos-Statistical and Probabilistic Aspects*, Chapman 1994.
- [5] M. Casdagli, *Non-linear Prediction of chaotic time series*, Physica D, 35, pp.335-356, 1989.
- [6] D.Lowe and A.R.Webb, *Time series prediction by adaptive networks*, IEE Proc.-F, 138, pp.17-25, 1991.
- [7] A.S.Weigend, D.E.Rumelhart, and B.A.Huberman, *Predicting the future: A Connectionist approach*, Technical Report Stanford-PDP-90-01, 1990.
- [8] J.R.McDonnell, and D.Wagen, *Evolving Recurrent Perceptrons for Time-Series Modelling*, IEEE Trans. Neural Networks, 5(1), pp.24-38, 1994.
- [9] M.B.Priestley, *Non-linear and non-stationary time series analysis*, Academic Press 1988.
- [10] C.Svaver, L.K.Hansen, J.Larsen, *On design and development of tapped delay neural network architectures*, IEEE Int. Conf. on Neural Networks, San Francisco, 1992.
- [11] A. Hussain, J.J.Soraghan, T.S.Durrani, *New ANN Architectures and Algorithms for Non-linear Dynamical System Modeling and Digital Communications Applications*, Technical Report, SPD, University of Strathclyde, U.K, 1997.
- [12] P.J.W.Rayner, M.R.Lynch, *A new connectionist model based on a non-linear adaptive filter*, ICASSP, Glasgow, 1989.
- [13] Y.H.Pao, S.M.Phillips, D.J.Sobajic, *Neural net computing and the intelligent control of systems*, Int. J. Control, 56(2), pp.263-289, 1992.
- [14] Leontaritis, S.A.Billings, *Model Selection and Validation methods for non-linear systems*, Intern. Journal of Control, Vol.41, pp.311-341, 1987.
- [15] A. Hussain, D.R.Campbell, T.J.Moir, *New Adaptive Non-Linear FIR filters for Sub-band Processing in Speech Enhancement Systems*, 16th IEE Colloquium on Digital and Analogue Filters and Filtering Systems, pp.6/1-6/6, London, U.K, 9th Dec'1996.

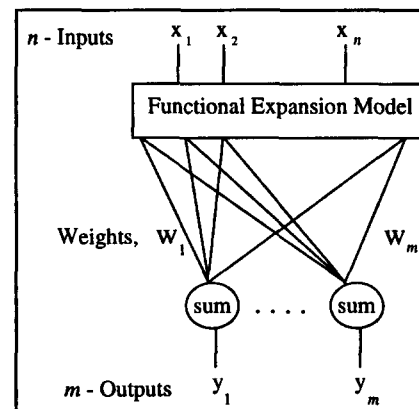


Figure 1: The Functionally Expanded Neural Network (FENN)