Abstract— An algorithm for constructing and training the Dyadic Wavelet Neural Network is proposed combining the theory of multiresolution analysis (MRA) of wavelet transforms and the conventional neural networks. The focus is to mainly improve function approximation accuracy in terms of dilation and translation parameters of wavelets, meanwhile not increasing the number of wavelet bases leading to optimal network structure. The proposed activation functions are drawn from a family of orthonormal basis functions. The good localization characteristics of the basis functions, both in the input and frequency domain allow hierarchical, multi-resolution learning of input-output function from experimental data. Compared with Wavelet Neural Networks of previous works, the model accuracy and generalization capability of the DWN are improved by adjusting the resolution parameter as it plays a significant role in dyadic wavelet analysis and approximation of a given function. In the learning process exhaustive experimentations are conducted to illustrate the impact of learning rate with respect to the generalization of the output function. All these advantages have been reflected in our experimentations. Two benchmark functions are simulated to illustrate the effectiveness of the method.

Index Terms—Dyadic Wavelet Neural Networks (DWN), Function learning, Orthonormal scaling functions, Wavelet Transforms (WT).

I. INTRODUCTION

Wavelets are a new family of basis functions that combine powerful properties such as orthogonality, compact support, localization in time and frequency, and fast algorithms [1, 2 and 3]. Cybenkos research has made the neural network into a powerful tool for signal representation particularly in function estimation problems [5].

The wavelet theory has found many applications in numerical analysis and signal processing [4], the Wavelet Transform (WT) of which provides another novel approach towards the function approximation problem [6,7,18]. Incorporating the time-frequency localization properties of wavelets and the learning abilities of general neural network (NN), WNN has shown its advantages over the regular methods utilizing wavelets as the basis function to construct a network [8, 9,10,12, 13, 14, 15, 17, 19, 20, 21].

The WNN's are characterized based on their activation functions used. When the continuous wavelets are selected as activation functions, and the parameters can take on any continuous real value, the translation, dilations and the network weights are updated in the process of training to optimize the network output. In the case of Dyadic wavelet series decomposition the network weights are optimized through learning process while the translation and dilation parameters are discretized in prior according to the dyadic sampling of the time-frequency plane. However, in order to improve computational efficiency, the values of the shift and scale parameters are often limited to some discrete lattices and are termed as the Discrete Wavelet Transform (DWT). The proposed methodology takes the advantage of the DWT in the learning process [11].

Though functional approximators other than wavelets may have the universal approximation property, in general, they do not possess the multiresolution property needed for approximation problems. The wavelets with coarse resolution capture the global (low frequency) behavior easily, while the wavelets with fine resolution can capture the local behavior (higher frequency) of the function accurately [1, 2]. This unique characteristic leads to the wavelet based neural network to be of the advantages of achieving fast convergence, easy learning and high accuracy.

The learning algorithm in the analysis phase validates the statistical measures used for the optimal network structure and parameter compatibility. The tuning of simulation parameters (learning rate and resolution) helps to arrive at an optimal network model through a training process.

This paper is organized as follows. The notations used and the basic concepts of wavelet subspace are introduced in section II and section III respectively. The learning algorithm for constructing the DWN is introduced in section IV. Two benchmark functions are simulated to illustrate the performance of DWN in comparison with other methods reported are given in section V. Finally, a brief conclusion is drawn in section VI.
II. NOTATIONS

The notations used are as in Table I.

<table>
<thead>
<tr>
<th>Term</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L^2(R)$</td>
<td>Space in which squares of all functions are $f(.)$ integrable on real data set $R$</td>
</tr>
<tr>
<td>$c_{j,m}$</td>
<td>WNN weights</td>
</tr>
<tr>
<td>$\beta_j$</td>
<td>non-linear system</td>
</tr>
<tr>
<td>$\omega$</td>
<td>translation parameter for scaling function</td>
</tr>
<tr>
<td>$\nu_j$</td>
<td>scaling parameters for wavelet function</td>
</tr>
<tr>
<td>$\mathcal{S}_j$ &amp; closed subspace of $L^2(R)$</td>
<td></td>
</tr>
<tr>
<td>$W_0$</td>
<td>Subspace for rough approximations</td>
</tr>
<tr>
<td>$W_j$</td>
<td>Subspace for detail approximations</td>
</tr>
<tr>
<td>$\phi(.)$</td>
<td>Scaling function</td>
</tr>
<tr>
<td>$\psi(.)$</td>
<td>Wavelet function</td>
</tr>
<tr>
<td>$N$</td>
<td>Number of training patterns (Samples)</td>
</tr>
<tr>
<td>$i$</td>
<td>Index for pattern</td>
</tr>
<tr>
<td>$k^i$</td>
<td>Index for input units</td>
</tr>
<tr>
<td>$x^i_b$</td>
<td>Input in vector notation</td>
</tr>
<tr>
<td>$\phi_{i,n}^{t,b}$</td>
<td>Wavelet function at $t^{th}$ and $b^{th}$ input</td>
</tr>
<tr>
<td>$D_i^t$</td>
<td>Desired output for unit $i$ with pattern $t$</td>
</tr>
<tr>
<td>$O^i_{i,k}$</td>
<td>Output for unit $i$ with pattern $i$, hidden and input layers</td>
</tr>
<tr>
<td>$b_j$</td>
<td>Translation parameter of $j^{th}$ scaling unit</td>
</tr>
<tr>
<td>$\alpha_j$</td>
<td>Dilatation parameter of $j^{th}$ scaling unit</td>
</tr>
<tr>
<td>$L$</td>
<td>Resolution level</td>
</tr>
<tr>
<td>$\eta$</td>
<td>Learning Rate $(\eta)$</td>
</tr>
<tr>
<td>$W_{ij}$</td>
<td>Weight between $i^{th}$ output and $j^{th}$ wavelet unit</td>
</tr>
<tr>
<td>$e_i^{(t)}$</td>
<td>Error for the $i^{th}$ pattern</td>
</tr>
</tbody>
</table>

III. WAVELET SUBSPACE IN SPACE $L^2(R)$

In this section, we briefly review the theoretical results from wavelet theory that are relevant to this work.

The space $L^2(R)$ of the system $f(.)$, must be divided into a closed subspace $V_0$ and a sequence of subspaces $W_j$, $j = 1, 2, 3, ..., +\infty$. Based on this space division, the definition of the subspace $V_0$ is inherited from the theory of multi-resolution approximation [1, 2]. The subspace $V_0$ which characterizes the rough approximation modeling at Resolution-0, can be any subspace $S_i$.

The subspace $W_j$, $j = 1, 2, 3, ..., +\infty$ characterizes the modeling of the system details at Resolution-1, Resolution-2, Resolution $+\infty$, are then defined as the difference between the subspace $S_i$ and the subspace $S_{j-1}$ as defined by equation given below:

$$ W_j = S_j - S_{j-1}, \quad j = 1, 2, 3, ..., +\infty $$

The properties of the subspaces $S_j$ are specified in equations given below

$$ \forall (j,k) \in Z^2, \quad j(t) \in W_j \Leftrightarrow j(t - 2^j k) \in W_j $$

Equation (2) describes the invariant property of the subspaces $W_j$. This property characterizes the system detail modeling on the subspace $W_j$. Equation (3) represents a system modeling on the subspace $W_{j-1}$ that is carried out by dilating the wavelet bases on the subspace $W_j$ by 2. The property described in equation (4) guarantees that the system detail modeling on the subspace $W_j$ is independent from the system detail modeling on the subspace $W_s$. Equation (5) ensures that when the subspace $W_j$ goes to positive infinity, i.e., the sum of the system rough modeling on the subspace $V_0$ and the system detail modeling on subspaces $W_j$, $j = 1, 2, 3, ..., +\infty$ converges to the exact function [1,2,16].

From the above analysis, the system modeling can be carried out on the set of subspaces $\{V_0 + \sum_{j=1}^{+\infty} W_j\}$ $j \in Z$.

IV. TRAINING WITH DYADIC WAVELET NETWORK

A fixed Dyadic Wavelet Network system model is in structure similar to a wavelet multiresolution model. The network structure is illustrated in Figure 1. It forms a wavelet multiresolution decomposition similar to the conventional Multi-Resolution Analysis (MRA) which involves not only a wavelet but also the associated scaling function.

The proposed model is developed by using orthonormal bases functions yielding the potential advantage of minimum usage of wavelet activation units. The number of activation units at each resolution can be determined as by shifting the wavelet (scaling) function in parallel with a step of the full support length. The DWN construction is detailed as follows.

![Dyadic Wavelet Network](image)

The orthonormal scaling functions $\phi_{i,m}(x)$, which are generated from the scaling function $\phi(x)$ by translation and dilation constitute orthonormal bases for the subspace $S_i$, as described in Equation (6).
The system rough approximation \( \hat{f}(.)|V_0 \) is modeled through the orthonormal bases \( \phi_{i,m}(x) \) on the subspace \( V_0 \) as described in equation

\[
\hat{f}(.)|V_0 = \sum_{m} c_i,m \phi_{i,m}(.)
\]

where \( c_i,m \) are coefficients of the system rough approximation and \( 'm' \) are translation parameters of the orthonormal bases on the subspace \( V_0 \). The orthonormal wavelet functions \( \Psi_{j+1,n} \), which are generated from the wavelet function \( \Psi(x) \) by translation and dilation, represent the orthonormal bases for the subspace \( W_j \). The orthonormal property of the wavelet function and the scaling function ensures that the inner dot product among any two different WNN's is always zero. The well trained WNN parameters at low resolution can be directly used in higher resolutions without any change. The learning algorithm of the DWN is given in the following section. The construction of DWN for multiresolution representation corresponds to wavelet decomposition and as a consequence the proposed representation of a system presents an advantage of resolution levels that can be separately trained. Consequently we can also show that the resolution levels are independent.

A. Learning Algorithm

The algorithm consists of two phases, the initialization phase and training phase.

Network Initialization

- Form vector \( \theta \) of network parameters \( W_{i,j}, b_j, a_j \)
- Assign weights \( W_{i,j} = 0 \) to links between hidden layer units and output layer.
- Choose a resolution level \( L \)
- Calculate the dyadic position \( b_j \) of wavelon within the given resolution.
- Set \( a_j = 2^L \)
- Finally initialize a wavelon with \( b_j, a_j \)

Training

- For each input output pattern \( (x^l, D^l) \) i.e., \( (x, f(x)) \)
- Compute the output of the DWN by
  \[
  O_{i,j}^l = \sum_{j=1}^{L_{max}} w_{i,j} \phi_{ij,b_j} \]
  where \( \phi_{ij,b_j} \) as given in equation (A)
- Compute the output with parameter vector \( \theta \) given by
  \[
  O_{\theta}^l = \sum_{j=1}^{L_{max}} w_{i,j} \sqrt{a_i,b_j} \phi(a_{i,j}x - b_{i,j})
  \]
- Compute error using
  \[
  e^l = |D^l - O^l|
  \]
- if \( e^l > e_{max} \)
  - Weights \( W_{ij} \) in vector \( \theta \) are modified in the opposite direction of the gradient
  \[
  C(\theta, x_k, f(x_k)) = \frac{\partial C}{\partial \phi} \frac{\partial \phi}{\partial Z} \frac{\partial Z}{\partial k}
  \]
- Compute the Gradient using
  \[
  \frac{\partial C}{\partial \phi} = c_k \sqrt{a_j} \phi(Z_{k,j})
  \]
- Where \( e_k = D_k^l - O_k^l \)
- Update weights by \( \frac{\partial C}{\partial \phi} \eta \)
- Repeat till error for all the patterns fall below the minimum prescribed value.
- Else save weights, network parameters and exit.

V. EXPERIMENTATION RESULTS AND DISCUSSION

In this section, to investigate the feasibility and efficiency of the proposed algorithm for Dyadic Wavelet Network, two benchmark datasets for non-linear one-dimensional function approximations are considered. To assess the approximation results, the figure of merit \( \delta \) is applied.

\[
\delta = \sqrt{\frac{\sum_{i=1}^{P} (|D_i - \tilde{D}_i|^2)}{\sum_{i=1}^{P} (D_i - \bar{y})^2}}
\]

Where
\[
\bar{y} = \frac{1}{P} \sum_{i=1}^{P} D_i
\]

A. Experiment (1) Benchmark dataset of function 1

The benchmark data function chosen is a piecewise function defined as follows:

\[
\begin{cases}
  -2.186x - 12.864 & \text{if } -10 \leq x < -2 \\
  4.246x - 2 & \text{if } -2 \leq x < 0 \\
  10 \exp(-0.05x - 0.5) \sin((0.03x + 0.7)x) & \text{if } 0 \leq x < 10
\end{cases}
\]

over the domain \( D = [-10,10] \).

We sampled 200 points, distributed uniformly over \([-10, 10]\) as training dataset to demonstrate the Dyadic wavelet network for approximation of static functions in terms of performance evaluation and efficiency of function learning.

To implement the wavelet decomposition network, scaling function from quadratic B-splines as given by eqn (A) is used

\[
\phi(x) = \begin{cases}
  \frac{1}{2}(x+1)^2 & \text{for } -1 \leq x < 0 \\
  \frac{1}{2} - (x-1/2)^2 & \text{for } 0 \leq x < 1 \\
  \frac{1}{2}(x-2)^2 & \text{for } 1 \leq x < -2 \\
  0 & \text{else}
\end{cases}
\]

During network learning, several experiments have been conducted by varying the network simulation parameters, i.e. learning rate and iterations at different resolution levels. The Performance index \( \delta \) of the network output function approximates to 0.0342 with less hidden units. Figures 2 to 5 are the results of the desired output and the proposed model output through the learning process of 1000 training epochs for approximation of the piecewise defined function for different Resolutions (L). The generalization of output function at different levels has greater impact on the accuracy level of the output function.

The optimal network output is obtained with good generalization for 23 wavelons and is as shown in Figure 5 for Resolution L = 0. The network parameter values obtained for the optimal wavelon is given in Table II.
experimentations learning rate $\eta$ of 0.1 has been chosen and for analysis the figure of merit $\delta$ for each approximation is computed on a uniformly sampled test set of 200 points [8, 21].

Table III depicts the network performance comparative results of DWNN and other reported WNN models [8, 16]. Figures 6 to 9 and from Figures 10 to 13 shows the network output for the optimal wavelength ‘23’ by varying number of iterations and learning rate respectively. Exhaustive experimentations demonstrated that the proposed DWN method resulted in an enhanced performance in terms of reducing the number of iterations and hidden nodes when compared with conventional neural networks and WNN [8]. It is observed that the change in error is negligible on increasing the number of iterations greater than 1000, has not resulted in any considerable improvement. Even varying learning rate parameter has least impact on the generalization of the network as indicated in the respective plots. This clearly reveals that the proposed learning algorithm for DWN is considerably robust and effective for function learning applications.

Fig. 2. Network output with 13 units and $L = -1$

Fig. 3. Network output with 15 units and $L = -0.75$

Fig. 4. Network output with 17 units and $L = -0.5$

Fig. 5. Network output with 23 units and $L = 0$

Fig. 6. Network output with 23 units and Learning Iterations = 50

Fig. 7. Network output with 23 units and Learning Iterations = 100

Fig. 8. Network output with 23 units and Learning Iterations = 200

Fig. 9. Network output with 23 units and Learning Iterations = 500
To demonstrate the validity of the proposed Dyadic Wavelet Network for approximating a single variable function [13], training points are obtained from the function defined by

\[ f(x) = \sin\left(\frac{2\pi x}{e^x}\right), \quad x \in (0, 10) \quad (10) \]

The approximation results for function (2) with increase in number of wavelons and Resolution (L) with fixed learning rate are shown in Table IV. In general the observations from the plots shown in Figures 14 to 19 indicate that with choice of decomposition level for network model, the low frequency values are approximated better than the high frequency values. Figure 14 and 15 shows the learning characteristics with less number of wavelons and different resolution values indicate poor generalization. With resolution L= 0 and 13 wavelons in the hidden layer the high frequency values are generalized better which is observable in Figure 16. Table IV provides the statistical results with choice of wavelons from 5 to 69. Figures 17 to 19 show that the network output for 25, 36 and 69 wavelons results in better generalization for low frequency values with decrease in variance. The generalization response obtained attributes to the characteristics of orthonormal scaling function chosen, which has better smoothness and convergence properties.
For function (2) experimentations are conducted considering 400 samples. Table V shows the various statistical measures i.e., mean, variance and standard deviation for 49 wavelons with constant Resolution L=2 and varying learning rates. Observations indicate that there is a subtle difference in the statistical measures obtained by varying the learning rate. The network responses are shown in Figures 20 to 22. Figure 23 indicates poor generalization performance of network with increase in wavelons, and reduction in learning rate. Similarly the network output for fixed Resolution 0 and 13 wavelons with different learning rates are shown in Figures 24 to 27. The observation of plots clearly indicates the effect of varying learning rate on the generalization characteristics and also the choice of optimal learning rate for training the network. The experimentations indicate that smaller values of learning rate results in poor generalization.

Table VI shows the Weight matrix i.e., translation and dilation parameters for the network model with 13 wavelons and resolution L=0. Since, it is a dyadic network the scaling function parameters remain constant with changes only in the weights.
VI. CONCLUSIONS

Wavelet Neural Networks have been successfully applied to the problem of function representations. Nevertheless, the analysis and design of WNN still remains a difficult task, since it requires a clear analysis of simulation parameters that are crucial to obtain a good approximation. The success of the proposed algorithm resides on the choice of generic activation function that keeps a balance between the outputs of the target function and the coordinates of the input vector space. In this paper the analysis of DWN is done considering the various statistical measures tuning with the simulation parameters (learning rate and resolution) so as to ensure the convergence capabilities. The study results also indicate that the orthonormal scaling functions considered resulted in smoothness and regularity on the convergence of network model.

In our work we have experimented considering orthonormal basis function such as Battle-Lemarie function, as the orthonormal wavelet-based networks can provide a unique and efficient representation for the given function, this also means that the approximation accuracy of this kind of DWN highly relies on the selected wavelet bases. We can also choose wavelet frames and Daubechies wavelets to construct the proposed DWN. Further study on extending of the proposed algorithm on DWN under additive noise is an interesting area under investigation in terms of generalization abilities.

In summary, the presented DWN not only reserves the multiresolution capability of WNN, but also has the advantages of high approximation accuracy and good generalization performance.

REFERENCES


[18] N Chihab, A Zergainoh, P Duhamel, J P Astruc " Orthonormal Non-Uniform B-Spline scaling and wavelet bases on non-equally spaced knot sequence for multiresolution signal approximations .


Pushpalatha M P received her BE degree in Computer Science and Engineering and MTech from University of Mysore, India in 1989 and 1997. Presently she is working as faculty in the department of Computer Science and Engineering, SJ College of Engineering, Mysore, India and a research scholar in Visvesvaraya Technological University, Belgaum, India. Her research interests are in soft computing.

Nalini N received the BE degree in Computer Science and Engineering from Kuvempu University, Karnataka, India in 1996, MS in Software Engineering from BITS, Pilani, India in 1999, PhD from Visvesvaraya Technological University, Belgaum, India in 2007. Presently she is working as Professor and Head of the department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India. Her research interests are in soft computing and security aspects in sensor networks. She has authored/co-authored more than 30 papers on these topics. She is an Associate Editor of Journal of Information Technology www. maxwellsci.com.