Computers and Mathematics with Applications 62 (2011) 3796–3811

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Multilayer perceptrons and radial basis function neural network methods for the solution of differential equations: A survey

Manoj Kumar, Neha Yadav *

Department of Mathematics, Motilal Nehru National Institute of Technology, Allahabad-211004 (U.P.), India

ARTICLE INFO

Article history:
Received 29 April 2011
Received in revised form 8 August 2011
Accepted 14 September 2011

Keywords:
Differential equations
Neural network
Multilayer perceptron
Radial basis functions
Backpropagation algorithm

ABSTRACT

Since neural networks have universal approximation capabilities, therefore it is possible to postulate them as solutions for given differential equations that define unsupervised errors. In this paper, we present a wide survey and classification of different Multilayer Perceptron (MLP) and Radial Basis Function (RBF) neural network techniques, which are used for solving differential equations of various kinds. Our main purpose is to provide a synthesis of the published research works in this area and stimulate further research interest and effort in the identified topics. Here, we describe the crux of various research articles published by numerous researchers, mostly within the last 10 years to get a better knowledge about the present scenario.

1. Introduction

A series of problems in many scientific fields can be modeled with the use of differential equations such as problems in physics [1–3], chemistry [4–6], biology [7,8], economics [9], etc. Due to the importance of differential equations many methods have been proposed in the relevant literature for their solution such as Runge–Kutta methods [10,11], Predictor–Corrector methods [12,13], Finite Difference methods [14–16], Finite Element methods [17–19], Splines [20–23] and other methods [24–41] etc. These methods require the discretization of domain into the number of finite elements where the functions are approximated locally. Although these methods provide good approximations to the solution, they require a discretization of domain via meshing, which may be challenging in two or higher dimension problems. Also, the approximate solution derivatives are discontinuous and can seriously impact on the stability of the solution. Furthermore, in order to obtain satisfactory solution accuracy, it may be necessary to deal with finite meshes that significantly increase the computational cost.

In spite of the above methods, approximate particular solutions can also be achieved by using multilayer perceptrons, radial basis functions, models based on genetic programming, hybrid approaches based on neural networks, etc. Advantages of these methods are that they involve a single independent variable regardless of the dimension of the problem, and the solution obtained from the neural network methods are differentiable and in closed analytic form. During the last few years much progress has been done in this area. In this article, a wide survey of solution of differential equations using multilayer perceptrons (MLP) and radial basis functions (RBF) along with our own comparative remarks have been presented. We will omit the details of techniques used to solve these models due to length constraints. Interested readers can see the references cited (along with these methods) for more details.

The rest of this article is organized as follows: in Section 2, we present a summary of Multilayer perceptron (MLP) neural network techniques for solving differential equations from various research articles. A brief description of articles published on the Radial Basis function neural network (RBFN) technique for solving differential equations is presented in Section 3. In

* Corresponding author.
E-mail addresses: manoj@mnnit.ac.in (M. Kumar), nehayad441@yahoo.co.in (N. Yadav).

0898-1221/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2011.09.028

Section 4, a comparison of MLP and RBF techniques for solving differential equations are given. Finally, Section 5 presents major conclusions and further developments.

2. Multilayer perceptron (MLP) neural network techniques to solve differential equations

In this section, we will give a brief description of multilayer perceptron (MLP) neural network techniques for solving differential equations of various kinds. Various research articles considered here are in the ascending order of their years of publication.

In [42], Meadre and Fernandez concentrated on developing a general, numerically efficient, non-iterative method in which the feed forward artificial neural network architecture can be used to model the solution of algebraic and differential equations accurately, using only the equation of interest and the boundary or initial conditions. A feed forward artificial neural network (FFANN) constructed by this non-iterative method is indistinguishable from those methods which are trained using conventional techniques. They borrowed a technique from applied mathematics, known as the method of weighted residuals (MWR), and showed how it can be made to operate directly on the network architecture. The method of weighted residuals is a generalized method for approximating functions, usually from given differential equations. A simple feed forward network using a single input and output neuron with a single hidden layer of processing elements utilizing the hard limit transfer function is constructed to approximate accurately the solution of a first and second order linear ordinary differential equation. They also predicted and controlled the error constructed by feed forward artificial neural networks. The non-iterative approach outlined in the article has allowed suitable algorithms to be devised for the synthesis of FFANNs that approximate non-linear ordinary differential equations. Also the non-iterative algorithm has allowed work to progress in the construction of FFANNs, sigma-Pi, and recurrent networks, to approximate linear and non-linear partial differential equations.

Remark 1. It can be concluded from the above article that it is possible to construct directly and non-iteratively a feed forward neural network to approximate arbitrary linear ordinary differential equations. The methods used are all linear in storage and processing time. The L_2—norm of the network approximation error decreases quadratically with the increasing number of hidden neurons. The result obtained through the output of network demonstrates the accuracy of approximation.

In the research article [43], Issac Ellias Lagaris and Aristidis Likas presented a new method for solving initial and boundary value problems using artificial neural network. To illustrate their method they took the following general differential equation:

$$G(\vec{x}, \psi(\vec{x}), \nabla\psi(\vec{x}), \nabla^2\psi(\vec{x})) = 0, \quad \vec{x} \in D \tag{1}$$

subject to the certain boundary conditions which are either Dirichlet or Neumann boundary conditions. Also $\vec{x} = (x_1, x_2, \ldots, x_n) \in R^n$ denotes the definition domain and $\psi(\vec{x})$ is the solution to be computed. Collocation method is adopted, to obtain a solution of the above differential equation (1), which assumes the discretization of the domain D and S into $\hat{D}$ and $\hat{S}$. Solve the above differential equation. They choose the form of trial function $\psi_t(\vec{x})$ such that by construction it satisfies the given boundary conditions. This is obtained by writing the trial function as a sum of two parts

$$\psi_t(\vec{x}) = A(\vec{x}) + F(\vec{x}, N(\vec{x}, \vec{p})) \tag{2}$$

where $N(\vec{x}, \vec{p})$ is a single output feed forward neural network with parameters $\vec{p}$ and n input units fed with the input vector $\vec{x}$. Term $A(\vec{x})$ contains no adjustable parameters and satisfies the boundary conditions. The second term F employs a neural network whose weights are adjusted to deal with the minimization problem and it is constructed so as not to contribute with boundary conditions. Then the network is trained to satisfy the differential equation. The authors illustrated present method by solving a variety of model problems. The solution such obtained is compared with the solution obtained using the Galerkin finite element method for several cases of partial differential equations and it is found that the method exhibits excellent generalization performance since the deviation at the test points was 'in no case' greater than the maximum deviation at the training points.

Remark 2. Method presented by the authors in [43] is general and can be applied to ordinary differential equations, the system of coupled ordinary differential equations and also to partial differential equations. It is also stressed out that the present method described in [43] can easily be used for dealing with the domains of higher dimensions. The method becomes particularly interesting with the help of neuroprocessor due to expected essential gains in the execution speed.

In article [44], the authors presented a method to solve a class of first order partial differential equation as input to state linearizable or approximate linearizable system. They proposed an extended backpropagation algorithm for training the derivative of a feed forward neural network and further, an approximate solution for a class of partial differential equation is obtained. A feedback control law has been designed to control a class of non-linear systems, based on the approximate solution. Simulation technique is used to demonstrate the effectiveness of the proposed algorithm and it is shown that the method can be very useful for practical applications in the following cases: