УДК 004.032.26

L. Avdiyenko, Ye. Bodyanskiy[1]

Kharkiv National University of Radio Electronics, Kharkiv

MODULAR OPTIMIZATION-MEMORY BASED ARTIFICIAL NEURAL NETWORK

In this article an architecture and learning algorithm for the modular neural network, where the hidden layer is formed by a general regression and radial basis function neural networks, have been proposed. These networks are parallel connected to the input layer and trained independently, following which the optimization with respect to the network output accuracy is performed. Formed by neural network models based on memory and optimization, the proposed modular neural network provides a high accuracy both in early learning stages and when data set could grow in real time.

Keywords: *radial basis function network, general regression neural network, optimization-based neural network, memory-based neural network, modular neural network.*

Introduction

Nowadays artificial neural networks have become widespread for solving problems of identification, emulation, prediction and nonlinear system control under uncertain conditions of both plant and environment features. First of all, their efficiency is provided by universal approximating capabilities and ability to learn using observations of the system inputs and outputs.

This situation is complicated when data comes continuously in real time and it should be processed online. It is obvious that in this case a conventional multilayer perceptron is not effective and as alternative a radial basis function network (RBFN) can be used, which is also a universal approximator [1-4]. Since the input signal of RBFN linearly depends on adjustable synaptic weights, its learning may be performed using the least squares method both in batch and recurrent modes. The recurrent least squares method, which is an optimization second-order procedure, has a high speed of convergence but requires a large amount of training data for effective adjustment of RBFN.

A radial basis function network, like a lot of other neural networks whose learning process is associated with optimization of a priori set learning criterion, belongs to a wide class of so-called “optimization-based neural networks”. The main drawback of such networks is defined by a low learning accuracy with a small training set.

An effective alternative of the optimization-based neural networks is so-called “memory-based neural networks“ whose typical representative is a general regression neural network (GRNN) proposed by D.F. Specht [5]. The basis of this network is an idea of Parzen windows [6] and Nadaraya-Watson kernel estimates [7-9]. Its learning comes to setting multidimensional radial basis functions in the points where coordinates are estimated by the plant input signals. Thereby, the GRNN and similar networks are called “just-in-time models” [4] and the learning principle – “neurons in data points” [10]. Having the architecture similar to RBFN, GRNN uses a completely different learning algorithm providing a high accuracy in the early learning stage. Problems can occur as the training set begins to grow due to a necessity of an additional solution to the subsidiary clustering problem.

It may be reasonable to develop a network which combines the advantages of RBFN and GRNN and guarantees a high accuracy in all learning stages in both batch and real-time modes. We suppose that a proposed modular general regression radial basis function network (GRRBFN), where learning is simultaneously based on optimization and memory, can meet these requirements.

GRRBFN architecture

The GRRBFN architecture is illustrated in fig. 1.

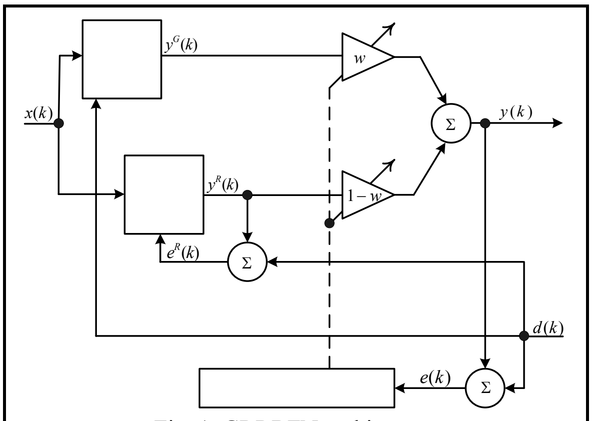

Fig. 1. GRRBFN architecture

The input signal $x(k)$ ($x(k) \in R^n$, $k = 1,2,3...$ is the current discrete time) is fed to the hidden layer formed by GRNN and RBFN. Their output signals $y^G(k)$ and $y^R(k)$ are fed to the output network layer formed by an adaptive linear associator with the adjustable synaptic weights w and $1-w$. The output network signal $y(k)$ is compared with the learning signal $d(k)$ (the output signal of the system) and as a result the learning error $e(k) = d(k) - y(k)$ used for adjusting the output layer is calculated. The neural networks in the hidden layer are trained using the learning signal $d(k)$ (GRNN) and error $e^R(k) = d(k) - y^R(k)$ (RBFN).

Batch-mode learning

Let assume a training set

$$x(1), d(1), x(2), d(2), ..., x(N), d(N)$$

is given. The Gaussians with fixed width parameter σ are used as activation functions for the hidden layer.

It has been already noticed that learning of GRNN turns into setting the Gaussian centers in points $x(k)$, $k = 1, 2, 3..., N$. Thus, a network response to the random signal x which does not belong to the training set is calculated according to the following expression:

$$y^G(x) = \sum_{k=1}^{N} d(k) \exp\left(-\frac{\|x - x(k)\|^2}{2\sigma^2}\right) \Bigg/ \sum_{k=1}^{N} \exp\left(-\frac{\|x - x(k)\|^2}{2\sigma^2}\right). \quad (1)$$

For RBFN with h neurons in the hidden layer the $((h+1)\times 1)$-vector of the synaptic weights are evaluated using the least squares method

$$w^R(N) = \left(\sum_{k=1}^{N} \varphi(k)\varphi^T(k)\right)^{-1} \sum_{k=1}^{N} \varphi(k) d(k),$$

where $\varphi(k) = (1, \varphi_1(k), ..., \varphi_h(k))^T$;

$$w^R(N) = \left(w_0^R(N), w_1^R(N), ..., w_h^R(N)\right)^T;$$

$$\varphi_l(k) = \exp\left(-\|x(k) - c_l\|^2 \big/ \left(2\sigma^2\right)\right);$$

$l = 1, 2, 3..., h$; c_l - $(n \times 1)$ – vector of the centres of the l-th activation function in the RBFN hidden layer) and a response to the signal x can be expressed as

The output layer of GRRBFN combines the

$$y^R(x) = w_0^R(N) + \sum_{l=1}^{h} w_l^R(N) \exp\left(-\|x(k) - c_l\|^2 \big/ \left(2\sigma^2\right)\right).$$

signals $y^G(k)$ and $y^R(k)$ as follows

$$y(x) = w(N) y^G(x) + (1-w)(N) y^R(x),$$

so that the signal $y(x)$ is unbiased and not worse than $y^G(k)$ and $y^R(k)$.

Let us take into consideration a set of $(N \times 1)$-vectors

$$y_N^G = \left(y^G(1), y^G(2), ..., y^G(N)\right)^T,$$

$$y_N^P = \left(y^P(1), y^P(2), ..., y^P(N)\right)^T,$$

$$y_N = \left(y(1), y(2), ..., y(N)\right)^T,$$

$$d_N = \left(d(1), d(2), ..., d(N)\right)^T,$$

$$e_N^G = \left(e^G(1), e^G(2), ..., e^G(N)\right)^T = d_N - y_N^G,$$

$$e_N^P = \left(e^P(1), e^P(2), ..., e^P(N)\right)^T = d_N - y_N^R,$$

$$e_N = \left(e(1), e(2), ..., e(N)\right)^T = d_N - y_N,$$

and write the obvious proportion

$$e_N = w e_N^G + (1-w) e_N^R,$$

after that solving the differential equation $\frac{\partial \|e_N\|}{\partial w} = 0$,

we obtain
$$\begin{cases} w(N) = (e_N^R)^T \dfrac{e_N^R - e_N^G}{\|e_N^R - e_N^G\|^2}, \\ 1 - w(N) = (e_N^G)^T \dfrac{e_N^G - e_N^R}{\|e_N^G - e_N^R\|^2}. \end{cases}$$

It is not difficult to prove validity of the inequalities

$$\begin{cases} \|e_N\|^2 - \|e_N^G\|^2 = -\dfrac{\left(\|e_N^G\|^2 - (e_N^G)^T e_N^R\right)^2}{\|e_t^P - e_t^G\|^2} \le 0, \\ \|e_N\|^2 - \|e_N^R\|^2 = -\dfrac{\left(\|e_N^R\|^2 - (e_N^G)^T e_N^R\right)^2}{\|e_N^G - e_N^R\|^2} \le 0. \end{cases}$$

Thus it follows that accuracy of the GRRBFN output signal is not worse than accuracy of the GRNN and RBFN output signals. The synaptic weight $w(N)$ defines a contribution of $y^G(k)$ to $y(k)$, i.e. the proximity of $y^G(k)$ and $y^R(k)$ to the training signal $d(k)$.

On-line mode learning

In order to operate in real-time mode the above-mentioned procedures should be rewritten in recurrent notation considering the appearance of new training samples $x(N+1), d(N+1)$.

For GRNN writing the expression (1) as follows

$$y^G(x) = NG(N) / DG(N),$$

it is quite easy to consider influence of new data

$$y^G(x) = \frac{NG(N) + d(N+1)\exp\left(-\|x - x(N+1)\|^2 \big/ \left(2\sigma^2\right)\right)}{DG(N) + \exp\left(-\|x - x(N+1)\|^2 \big/ \left(2\sigma^2\right)\right)}.$$

The synaptic weights of RBFN can be refined using either recurrent least squares method

$$\begin{cases} w^R(N+1) = w^R(N) + P(N+1) \times \\ \times \left(d(N+1) - w^{RT}(N)\varphi(N+1)\right)\varphi(N+1); \\ P(N+1) = P(N) - \dfrac{P(N)\varphi(N+1)\varphi^T(N+1)P(N)}{1 + \varphi^T(N+1)P(N)\varphi(N+1)}, \end{cases}$$

or quite the effective and simple Kaczmarz-Widrow-Hoff algorithm

$$w^R(N+1) = w^R(N) + \frac{d(N+1) - w^{RT}(N)\varphi(N+1)}{\|\varphi(N+1)\|^2}\varphi(N+1).$$

For the weight coefficients of the output layer considering the additional error $e^D(k) = e^R(k) - e^G(k)$, the following expressions can be obtained

$$\begin{cases} w(N+1) = \dfrac{\eta(N)}{\eta(N+1)} w(N) + \dfrac{e^R(N+1)e^D(N+1)}{\eta(N+1)}, \\ \eta(N+1) = \eta(N) + (e^D(N+1))^2. \end{cases} \quad (2)$$

Taking into consideration these obvious relations

$$e^D(N) = d(N) - y^R(N) - d(N) + y^G(N) = y^G(N) - y^R(N),$$

$$e^D(N+1) = y^G(N+1) - y^R(N+1),$$

the expression (2) may be rewritten

$$\begin{cases} w(N+1) = \dfrac{\eta(N)}{\eta(N+1)} w(N) + \dfrac{e^R(N+1)\left(y^G(N+1) - y^R(N+1)\right)}{\eta(N+1)}, \\ \eta(N+1) = \eta(N) + (y^G(N+1) - y^R(N+1))^2. \end{cases}$$

Conclusions and directions of the future investigations

In this article the simple and effective modular general regression radial basis function network, which allows solving problems of identification, emulation and prediction etc. under the considerable uncertainty of the system, has been proposed. The network can be trained in both batch and real-time modes.

It is supposed to apply the proposed neural model to solving a practical problem of predicting the growth of bacteria count in poultry which depends on temperature, chemical properties of the environment and time. The choice of the problem is grounded on the fact that the process of growing is characterized by a small amount of experimental data. It makes a usage of the standard RBFN problematic since it requires a quite large amount of training data for solving an approximation task.

Assuming that in real time this process could be described by a growing learning set that in turn makes a usage of single GRNN non-effective, the modular neural network model is formed by RBFN and GRNN should solve the formulated task.

Also in the future it is assumed to apply the obtained results to non-stationary plants, improve robust properties of the algorithm, provide a possibility of adjusting architecture of the network hidden layer and receptive fields of the radial basis functions, synthesize a fuzzy-neural system based on optimization and memory simultaneously.

References

1. *Moody J., Darken C.J. Fast learning in networks of locally-tuned processing units // Neural Computation. – 1989. – 1. – P. 281-294.*
2. *Park J., Sandberg I.W. Universal approximation using radial-basis-function networks // Neural Computation. – 1991. – 3. – P. 246-257.*
3. *Schilling R.J., Carrol J.J., Al-Ajlouni A.F. Approximation of nonlinear systems with radial basis function neural networks // IEEE Trans. on Neural Networks. – 2001. – 12. – P. 1-15.*
4. *Nelles O. Nonlinear System Identification. – Berlin: Springer, 2001. – 785 p.*
5. *Specht D.E. A general regression neural network // IEEE Trans. on Neural Networks. – 1991. – 2. – P. 568-576.*
6. *Parzen E. On the estimation of a probability density function and the mode // Ann. Math. Stat. – 1962. – 38. – P. 1065-1076.*
7. *Nadaraya E.A. About nonparametric probability and regression estimates // Probability Theory and its Application. – 1965. – 10. – #1. – P. 199-203.*
8. *Bishop C.M. Neural Networks for Pattern Recognition. – Oxford: Clarendon Press, 1995. – 482 p.*
9. *Friedman J., Hastie T., Tibshirani R. The Elements of Statistical Learning. Data Mining, Inference and Prediction. – Berlin: Springer, 2003. – 552 p.*
10. *Zahirniak D.R., Chapman R., Rogergs S.K., Suter B.W., Kabrisky M., Pyatti V. Pattern recognition using radial basis function network // Proc. 6-th Ann. Aerospace Application of AI Conf-Dayton, OH, 1990. – P. 249-260.*

Поступила в редколлегию 25.07.2008

Рецензент: д-р техн. наук, проф. В.М. Илюшко, Национальный аэрокосмический университет «ХАИ», Харьков.

МОДУЛЬНАЯ ИСКУССТВЕННАЯ НЕЙРОННАЯ СЕТЬ, ОСНОВАННАЯ НА ПАМЯТИ И ОПТИМИЗАЦИИ

Л.С. Авдиенко, Е.В. Бодянский

В статье предложены архитектура и алгоритм обучения модульной нейронной сети, скрытый слой которой образован обобщенной регрессионной и радиально-базисной нейронными сетями, параллельно подключенными ко входящему слою и обучаемыми независимо друг от друга, а в выходном слое производится оптимизация по точности выхода сети относительно обучающего сигнала. Предлагаемая модель, объединяя в себе нейронные сети, основанные на памяти и оптимизации, обеспечивает высокую точность алгоритма как на начальных этапах обучения, так и при росте выборки данных в реальном времени.

Ключевые слова: *радиально-базисная нейронная сеть, обобщенная регрессионная нейронная сеть, нейронная сеть, основанная на оптимизации, нейронная сеть, основанная на памяти, модульная нейронная сеть.*

МОДУЛЬНА ШТУЧНА НЕЙРОННА МЕРЕЖА, ЩО БАЗУЄТЬСЯ НА ПАМ'ЯТІ ТА ОПТИМІЗАЦІЇ

Л.С. Авдієнко, Є.В. Бодянський

У статті запропоновані архітектура та алгоритм навчання модульної нейронної мережі, прихований шар якої сформований узагальненою регресійною та радиально-базисною нейронними мережами, що паралельно підключені до вхідного шару та навчаються незалежно одна від одної, а у вихідному шарі відбувається оптимізація по точності виходу мережі відносно навчального сигналу. Пропонуєма модель, поєднуючи у собі нейронні мережі, засновані на пам'яті та оптимізації, забезпечує високу точність алгоритму як на початкових етапах навчання, так і з ростом вибірки даних у реальному часі.

Ключові слова: *радиально-базисна нейронная мережа, узагальнена регресійна нейронна мережа, нейронна мережа, заснована на оптимізації, нейронная мережа, заснована на пам'яті, модульна нейронна мережа.*