Review Article **Open Access**

Ch. Sanjeev Kumar Dash, Ajit Kumar Behera, Satchidananda Dehuri*, and Sung-Bae Cho

Radial basis function neural networks: a topical state-of-the-art survey

DOI 10.1515/comp-2016-0005

Received October 12, 2014; accepted August 24, 2015

Abstract: Radial basis function networks (RBFNs) have gained widespread appeal amongst researchers and have shown good performance in a variety of application domains. They have potential for hybridization and demonstrate some interesting emergent behaviors. This paper aims to offer a compendious and sensible survey on RBF networks. The advantages they offer, such as fast training and global approximation capability with local responses, are attracting many researchers to use them in diversified fields. The overall algorithmic development of RBF networks by giving special focus on their learning methods, novel kernels, and fine tuning of kernel parameters have been discussed. In addition, we have considered the recent research work on optimization of multi-criterions in RBF networks and a range of indicative application areas along with some open source RBFN tools.

Keywords: neural network; radial basis function networks; multi-criterions optimization; learning; classification; clustering; approximation

1 Introduction

Multi-layer perceptron (MLP) network models are the popular network architectures used in most of the application areas [1]. In an MLP, the weighted sum of the inputs and bias term are passed to activation level through a transfer function to produce the output (i.e., $p(\vec{x}) = f_s\left(w_0 + \sum_{i=1}^{D} w_i x_i\right)$, where $\vec{x}$ is an input vector of dimension D, $w_i, i = 1, 2, \ldots, D$ are weights, w_0 is the bias weight, and $f_s(o) = \frac{1}{1+e^{-ao}}$, a is the slope) and the units are arranged in a layered feed-forward topology called Feed Forward Neural Network [2]. The network is then trained by back-propagation learning scheme. An MLP with back-propagation learning is also known as Back-Propagation Neural Networks (BPNNs). In BPNNs a common barrier is the training speed (i.e., the training speed increases as the number of layers, and number of neurons in a layer grow) [3]. To circumvent this problem a new paradigm of simpler neural network architectures with only one hidden layer has been penetrated to many application areas with a name of Radial Basis Function Neural Networks (RBFNs) [4, 6–14]. RBFNs were first introduced by Powell [15–18] to solve the interpolation problem in a multi-dimensional space requiring as many centers as data points. Later Broomhead and Lowe [19] removed the 'strict' restriction and used less centers than data samples, so allowing many practical RBFNs applications in which the number of samples is very high. An important feature of RBFNs is the existence of a fast, linear learning algorithm in a network capable of representing complex non-linear mapping. At the same time it is also important to improve the generalization properties of RBFNs [20, 21]. Today RBFNs have been a focus of study not only in numerical analysis but also in machine learning researchers. Being inherited from the concept of biological receptive field [19] and followed, Park and Sandberg prove, "RBFNs were capable to build any non-linear mappings between stimulus and response" [22]. The growth of RBFNs research have been steadily increased and widespread in different application areas (c.f., Sections 5, 6, and 7 for more detailed discussion along with a number of indicative cited works). The research in RBFNs are grouped into three categories [23]: i) development of learning mechanism (include heuristics and non-heuristics), ii) design of new kernels, and iii) areas of application.

Recently, Patrikar [24] has studied that RBFNs can be approximated by multi-layer perceptron with quadratic inputs (MLPQ). Let the MLPQ with one hidden layer consist-

Ch. Sanjeev Kumar Dash: Department of Computer Science and Engineering, Silicon Institute of Technology, Silicon Hills, Patia, Bhubaneswar, 751024, India, E-mail: sanjeev_dash@yahoo.com
Ajit Kumar Behera: Department of Computer Application, Silicon Institute of Technology, Silicon Hills, Patia, Bhubaneswar, 751024, India, E-mail: ajit_behera@hotmail.com
***Corresponding Author: Satchidananda Dehuri:** Department of Systems Engineering, Ajou University, San 5, Woncheon-dong, Yeongtong-gu, suwon 443-749, South Korea, E-mail: satchi.lapa@gmail.com
Sung-Bae Cho: Sung-Bae Cho, Soft Computing Laboratory, Department of Computer Science, Yonsei University, 134 Shinchon-dong, Sudaemoon-gu, Seoul 120-749, South Korea, E-mail: sbcho@yonsei.ac.kr

ing of M units and the output unit with linear activation function. Let $w_0(o)$ be the bias weight for the output unit and let $w_i(o)$ be the weights associated with the connection between hidden unit i and output unit o. the output of the MLPQ can be computed as:

$$Q(\vec{x}) = w_0(o) + \sum_{i=1}^{M} \frac{w_i(0)}{1 + \exp\left(-w_{i0} - \sum_{Lij} x_j - \sum_{j=1}^{D} w_{Qij} x_j^2\right)}, \tag{1}$$

where for each hidden unit i, w_{io} is the bias weight, w_{Lij} are the weights associated with linear terms x_j, and w_{Qij} are the weights associated with quadratic terms x_j^2. For an RBFN with same hidden units, the output is given by:

$$R(\vec{x}) = \sum_{i=1}^{M} w_i \exp\left(-\beta \left(\vec{x} - \vec{\mu}_i\right)^T \left(\vec{x} - \vec{\mu}_i\right)\right), \tag{2}$$

where w_i are the weights of unit i, μ_i is the center vector of unit i, and β is the parameter. Using approximation, Equation (2) can be approximated as

$$R(\vec{x}) = \sum_{i=1}^{M} w_i \left(H/\left(1 + \exp\left(-c\beta \left(\vec{x} - \vec{\mu}_i\right)^T \left(\vec{x} - \vec{\mu}_i\right)\right) - d\right)\right), \tag{3}$$

where H, c, and d are constants.

This paper is set out as follows. Section 2 gives overview RBFN architecture. Section 3 of this paper is devoted to the multi-criteria issues of RBFNs and discussed some potential contributions. Different RBFN tools are discussed in Section 4. Sections 5-7 discusses the various application of RBFNs. Summary along with future research direction are discussed in Section 8.

2 RBFNs architecture

The idea of RBFNs is derived from the theory of function approximation. The Euclidean distance is computed from the point being evaluated to the center of each neuron, and a radial basis function (RBF) (also called a kernel function) is applied to the distance to compute the weight (influence) for each neuron. The radial basis function is so named because the radius distance is the argument to the function. In other words, RBFs represent local receptors; its output depends on the distance of the input from a given stored vector. That means, if the distance from the input vector $\vec{x}$ to the center $\vec{\mu}_i$ of each RBF φ_j i.e., $||\vec{x} - \vec{\mu}_j||$ is equal to 0 then the contribution of this point is 1, whereas the contribution tends to 0 if the distance $||\vec{x} - \vec{\mu}_j||$ increases.

RBF networks broadly consist of three layers (see Figure 1)[26] *Input layer* - The input layer can have more than one predictor variable where each variable is associated with one independent neuron. The output of the input layer neurons, then feed the values to each of the neurons in the hidden layer. 2) *Hidden layer* - The hidden layer can have multiple numbers of neurons. The selection of neurons in this layer is a challenging task. Mao and Huang [25], have suggested a data structure preserving criterion technique for selection of neurons in the hidden layer.

Each neuron in the hidden layer consists of an RBF centered at a point, depending on the dimensionality of the input/output predictor variables. The hidden unit activations are given by the basis functions $\varphi_j\left(\vec{x}, \vec{\mu}_j, \sigma_j\right)$ (e.g., Gaussian basis functions), which depend on the parameters $\{\vec{\mu}_j, \sigma_j\}$ and input activations $\{\vec{x}\}$ in a non-standard manner.

$$\varphi_j\left(\vec{x}\right) = \exp\left(-\frac{||\vec{x} - \vec{\mu}_j||^2}{2\sigma_j^2}\right). \tag{4}$$

The spread (radius) of the RBF function may be different for each dimension. The centers and spreads are determined by the training process of the network.

3. *Summation layer*- Each neuron is associated with weights $(w_1, w_2, \ldots, w_N)$. The value coming out of a neuron in the hidden layers is multiplied by the weight associated with the neuron and passed to the summation which adds up the weighted values and presents this sum as the output of the network. A bias value is multiplied by a weight w_0 and fed into the summation layer.

It is interesting to note that RBFNs are closely related to Gaussian Mixture Model (GMM) if Gaussian basis functions are used [**?**]. RBFNs with N hidden units with one output neuron in the output layer can be represented as (Equation (2)) with a simple replacement of M by N. A Gaussian mixture density is a weighted sum of component densities given by:

$$g(\vec{x}|\lambda) = \sum_{i=1}^{N} \alpha_i b_i(\vec{x}), \text{ where } \sum_{i=1}^{N} \alpha_i = 1.$$
$$b_i(\vec{x}) = \frac{1}{(2\pi)^{D/2} |\sum_i|^{1/2}} \exp\left(-1/2 \left(\vec{x} - \vec{\mu}\right)^T \sum_i^{-1} \left(\vec{x} - \vec{\mu}\right)\right) \tag{5}$$

with mean vector $\vec{\mu}_i$ and covariance matrix $\sum_i$.

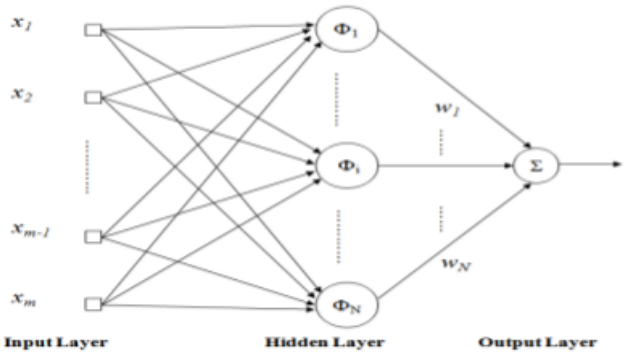

Figure 1: Architecture of RBF network.

2.1 Learning in RBFNs

Learning or training a neural network is a process by means of which the network adapts itself to a stimulus by making proper parameter adjustment, resulting in the production of desired response. Hence, in order to achieve the similar approximation/classification accuracy and in addition to the required number of RBF units, the following parameters are determined by the training process of RBFNs [27]:

1. The number of neurons in the hidden layer. [Ideally the number of neurons (M) in the hidden layer should be much less than data points (N)];
2. The coordinates of the center of each hidden-layer RBF, determined by the training algorithm;
3. The radius (spread) of each RBF in each dimension, determined by the training algorithm; and
4. The weights applied to the RBF outputs as they are passed to the summation layer.

2.2 Kernels in RBFN

The Gaussian kernel is a usual choice for kernel functions. Equation (5) is the most generic form of Gaussian kernel. The inverse of the covariance matrix is used to capture the correlations between different features, providing to each kernel an n-dimensional ellipsoid shape. It is generally more versatile than using the simple distance to kernel centroid that assumes strict variable independence. However, if the covariance matrix is singular or very ill conditioned, the use of the inverse can produce meaningless results or strong numerical instability. Therefore, a spectral decomposition suggested in [28] can be applied to covariance matrix, producing the Eigen system $\sum = P\Lambda P^t$ in which P is the matrix composed of the eigenvectors and Λ the respective Eigen values in a diagonal matrix format [125].

$$\varphi_j(\vec{x}) = \frac{1}{\left(1 + \frac{||\vec{x}-\vec{\mu}_j||}{\sigma_j^2}\right)} \tag{6}$$

$$\varphi_j(\vec{x}) = \frac{1}{\left(1 + \frac{||\vec{x}-\vec{\mu}_j||}{\sigma_j^2}\right)^{1/2}}. \tag{7}$$

Equations (6) and (7) have larger tails than Equation (4) i.e., their activations for patterns far from the centroid of the RBF which is greater than the activation of the standard form (Equation (4)) for these patterns. For sufficiently large distance norm, the decay of the Equations (6) and (7) is very slow. In addition, Equations (4), (6), and (7) do not fall asymptotically to zero. Authors have presented q-Gaussian RBFs as an alternative to Gaussian RBF [29]. The q-Gaussian RBF for the i^{th} unit is defined as:

$$\varphi_j(\vec{x}) = \begin{cases} \left(1 - (1-q)\left(\frac{||\vec{x}-\vec{\mu}_j||^2}{\sigma_j^2}\right)\right)^{1/1-q}, \\ \qquad \text{if } 1-(1-q))\left(\frac{||\vec{x}-\vec{\mu}_j||^2}{\sigma_j^2}\right) \geq 0 \\ 0 \qquad \text{otherwise,} \end{cases} \tag{8}$$

where q is a real valued parameter. The q-Gaussian RBF allows different RBFs to be represented by updating the new parameter q. In [30] the different properties of Gaussian functions have been discussed. In addition to standard Gaussian kernels, a range of other basis functions can be used in RBFNs. The commonly used RBFs are expressed in Table 1.

In multi-quadric function the matrix representation of basis function has an important spectral property: it is almost negative definite. Franke [31] has found that this radial basis function provides the most accurate interpolation surface in two dimensions. Also he found that the inverse multi-quadric basis function can provide excellent approximations, even when the number of centers is small. However, author presents that sometimes a large value of σ can be useful [32]. In contrast, there is no good choice of σ known at present in the case of multi-quadric basis function.

The thin plate spline basis function has more global nature than the Gaussian function i.e., a small perturbation of one of the control points always affect the coefficients corresponding to all other points as well. Similarly, the polynomial basis functions like cubic and linear has some degree of influence in certain applications. An overview of RBFs and its corresponding models are described in [4]. The use of above list of kernels along with

Table 1: Other Kernels used in RBFN.

Name of the Kernel	Mathematical Representation
Generalized Multi-Quadric Functions	$\Phi(r) = \left(r^2 + c^2\right)^{\beta},\ c > 0,\ 1 > \beta > 0$
Generalized Inverse Multi-Quadric Functions	$\Phi(r) = \left(r^2 + c^2\right)^{-\alpha},\ c > \alpha > 0$
Thin Plate Spline Function	$\Phi(r) = r^2 \ln(r)$
Cubic Function	$\Phi(r) = r^3$
Linear Function	$\Phi(r) = r$

Gaussian kernels can be obtained in [5, 33–36]. However, we observe in most of the neural network literature that the Gaussian RBFs is widely used in diversified domain like medical/biological science, computational finance, defense systems, engineering, etc.

2.3 Learning of kernel parameters and weights

One major advantage of RBF networks of choosing suitable hidden unit/basis function parameters without having to perform a full non-linear optimization of the whole network. The coordinates of center of each hidden-layer in RBF function can be calculated by using any of the following unsupervised methods:

2.3.1 Fixed centers selected at random

This is a simple and fast approach for setting the RBF parameters, where the centers are kept fixed at M points selected at random from the N data points. Specifically, we can use normalized RBFs centered at $\{\mu_j\}$ defined by

$$\phi_j(x) = \exp\left(-\frac{||x - \mu_j||^2}{2\sigma_j^2}\right), \quad \text{where } \{\mu_j\} \subset \{X^p\},\ \sigma_j \text{ is spread.} \tag{9}$$

2.3.2 Clustering

Clustering techniques can be used to find a set of centers which more accurately reflect the distribution of the data points. The K-means clustering algorithm [37] selects the number K of centers in advance, and then follows a simple re-estimation procedure to divide the data points $\{X^p\}$ into K disjoint sub-sets S_j and N_j data points in order to minimize the sum of squared clustering function.

$$J = \sum_{j=1}^{K} \sum_{p \in S_j} ||X^p - \mu_j||^2, \tag{10}$$

where, μ_j is the mean/centroid of the data points in set S_j given by the Equation (11):

$$\mu_j = \frac{1}{N_j} \sum_{p \in S_j} X^p. \tag{11}$$

There are, however, two intrinsic disadvantages associated with the use of K-means. The first is due to its iterative nature, which can lead to long convergence times, and the second originates from its inability to automatically determine the number of RBF centers, thus resulting in a time-consuming trial-and-error procedure for establishing the size of the hidden layer.

A multitude of alternative techniques have been proposed to tackle these disadvantages. One way is to use some improved unsupervised methods [38] such as: fuzzy clustering [39–43], self organizing map (SOM) [44], particle swarm optimization (PSO)-based subtractive clustering [45, 46], dynamic K-means clustering algorithm [47], improved K-means algorithm [48], K-harmonic clustering [49], and self-additive clustering [50], have been used for center selection in RBFNs.

The other way includes a significant portion of these methodologies use a constructive approach, building the hidden layer incrementally until a criterion is met. Within this context, the application of the orthogonal least squares algorithm has been thoroughly explored [51–54].

2.3.3 Orthogonal Least Squares (OLS)

One of the fine tuned approaches to selecting a sub-set of data points as the basis function centers is based on the technique of orthogonal least squares. OLS is a forward stepwise regression procedure, where OLS sequentially selects the center that results in the largest reduction

of sum-of-square-error at the output. OLS constructs a set of orthogonal vectors Q for the space spanned by the candidate centers. In this orthogonal subspace, computation of pseudo-inverse is avoided since $Q^{'}Q$ becomes diagonal. OLS construct a set of orthogonal vectors Q for the space spanned by basis vectors ϕ_k such that $\Phi = QA$ where, A is an upper triangular matrix. Using this orthogonal representation, the RBF solution is expressed as:

$$T = \Phi W = QG \tag{12}$$

and the LS solution for the weight vector G in the orthogonal space is given as:

$$G = (Q^{'}Q)^{-1}Q^{'}T. \tag{13}$$

The above discussions are restricted with center and spread selection mechanism except the one in respect to Gaussian kernel. However, there are abundant of proposals developed with their own merits and demerits such as constructive decay [29], resource allocating networks [55], and the minimum description length principle [56]. Recently, Alexandridis *et al.*, [57], introduced a nonsymmetric approach for partitioning the input space. Their experimental outcomes have shown that the nonsymmetric partition can lead to the development of more accurate RBF models, with a smaller number of hidden layer nodes. More elaborate methods have been suggested [58–61] for optimizing the RBF widths in order to improve approximation accuracy.

Taking advantage of the linear connection between the hidden and output layer, most training algorithms calculate the synaptic weights of RBF networks by applying linear regression of the output of the hidden units on the target values. Alternative approaches for calculating the weights include gradient descent methods [62], fuzzy logic [63], and the expectation-maximization algorithm [64].

A few algorithms aspiring to determine all the RBF training parameters in one step have also been proposed in the literature. In [65], a hierarchical Bayesian model is introduced for training RBFs. The model treats all the training parameters as unknown random variables and Bayesian calculation is performed through a reversible-jump Markov chain Monte Carlo method, whereas the networks are optimized using a simulated annealing algorithm. In [66], RBF parameters are determined in a one-step algorithm in interpolation problems with equally spaced nodes, after replacing the Euclidean norm associated to Gaussian RBF with a Mahalanobis norm. In [67], all the RBF network parameters, including input weights on the connections between input and hidden layers, are adjusted by a second-order update rule.

It should be noted, however, that calculating optimal values for all the RBF parameters is a rather cumbersome task. Viewing the RBF network training procedure as an optimization problem, one realizes that the objective function usually presents some rather unwelcome properties including, multimodality, non-differentiability and high levels of noise. As these characteristics make use of standard optimization methods inefficient, it is no surprise that a significant number of studies have focused on optimizing the RBF training procedure through the use of alternative approaches, such as evolutionary-based computation techniques [68]. The resulting methodologies include a genetic algorithm for optimizing the number and coordinates of RBF centers [69], a hybrid multi-logistic methodology applying evolutionary programming for producing RBFs with simpler structures [29], a multi-objective evolutionary algorithm to optimize RBF networks including some new genetic operators in the evolutionary process [71], and an evolutionary algorithm that performs feature and model selection simultaneously for RBF classifiers in reduced computational times [72]. Similarly, PSO is a powerful stochastic optimization algorithm that has been used successfully in conjunction with other computational intelligence tools [73, 74]. A PSO-aided orthogonal forward regression algorithm based on leave-one-out criteria is developed in [75] to construct parsimonious RBF networks with tunable nodes. A recursive orthogonal least squares algorithm has been combined with PSO in a novel heuristic structure optimization method for RBF probabilistic networks [76]. The architecture of k-means clustering-based polynomial RBFNs has been introduced in [77], using PSO and differential evolution (DE) [78].

In most practical applications, especially in medical diagnosis, the complete training data describing the input-output relationship may not available *a priori*. For these problems, classical batch-learning algorithms are rather infeasible and instead sequential learning is employed. In a sequential learning framework [79], the training samples arrive one-by-one and the samples are discarded after the learning process. Hence, it requires less memory and computational time for the learning process. In addition, sequential learning algorithms automatically determine the minimal architecture that can accurately approximate the true decision function described by stream of the training samples [80]. Radial basis function (RBF) networks have been extensively used in a sequential learning framework because of their universal approximation ability and simplicity of architecture [81–83]. Recently, there has been renewed interest in single hidden-layered RBF networks with least-square error training criterion, partly due to their modeling ability and partly due to the ex-

istence of efficient learning algorithms such as extreme learning machine (ELM) [84], and second-order training methods [67]. In recent years, researchers have been focusing on sequential learning algorithms for RBF networks through streams of data.

Taking into consideration of learning weights, spreads, and centers, Table 2 summarizes few more proposals for training RBFNs.

3 Multi-criteria optimization in RBFNs

Generally, an RBFN has a problem of model complexity. If a model is complex, then it can fit well to the input data. Because, it has low bias error and high variance error, hence, the model generalization ability becomes worse. Moreover, a complex model is not desirable as it is not very easy to handle. On the other hand, if a model is simple then variance error is less. Such a model can prevent overlearning and then can be corrected easily. The automatic optimization of an RBFN model from a given set of samples is a problem of recent studies- in which two identified competing objectives must be satisfied [70]. The model's prediction error must be minimized in order to achieve a well fitted model, while the number of RBFs should be as low as possible to obtain a reliable interpolator/regressor or a classifier/approximator [71]. The problem of optimization of these two objectives simultaneously is known as multi-criteria optimization problem in RBFNs. They are conflicting because improving one of them will worsen the other. Their solutions are usually sub-optimal for each objective in particular but acceptable taking all the objectives into accounts, where acceptable is totally subjective and problem dependent. In this connection Subsection 3.1 discusses definitions and some basic concepts of multi-objective optimization problem (MOP). Subsection 3.2 discusses various approaches for solving MOP in RBFNs.

3.1 Definitions and basic concepts of MOP

A multi-objective optimization problem can be stated in the following general form:

Minimize/Maximize $f_m(x)$, $m = 1, 2, \ldots M$;
subject to $g_j(x) \geq 0$, $j = 1, 2, \ldots J$;
$h_k(x) = 0$ $k = 1, 2, \ldots K$;
$x_i^{(L)} \leq x_i \leq x_i^{(U)}$ $i = 1, 2, \ldots m$.

A solution x is a vector of n decision variables: $x = (x_1, x_2, \ldots, x_n)^T$. The last set of constraints is called variable bounds, restricting each decision variable x_i to take a value within a lower $x_i^{(L)}$ and an upper $x_i^{(U)}$ bounds. These bounds constitute a decision variable space D, or simply the decision space. Associated with the problem are J inequality and K equality constraints and the terms $g_j(x)$ and $h_k(x)$ are called constraint functions.

In multi-objective optimization, the M objective functions $f(x) = (f_1(x), f_2(x), \ldots, f_M(x))^T$ can be either minimized or maximized or both. Many optimization algorithms in particular RBFNs are developed to solve only one type of optimization problems, such as e.g., minimization problems [85]. When an objective is required to be maximized by using such an algorithm, the duality principle can be used to transform the original objective for maximization into an objective for minimization by multiplying objective function by -1. It is to be noted that for each solution x in the decision variable space, there exists a point in the objective space, denoted by $f(x) = z = (z_1, z_2, \ldots, z_M)^T$. There are two goals in a multi-objective optimization: firstly, to find a set of solutions as close as possible to the Pareto-optimal front; secondly, to find a set of solutions as diverse as possible. Multi-objective optimization involves two search spaces i.e. the decision variable space and the objective space [86]. Although these two spaces are related by a unique mapping between them, often the mapping is non-linear and the properties of the two search spaces are not similar. In any optimization algorithm, the search is performed in the decision variable space. However, the proceedings of an algorithm in the decision variable space can be traced in the objective space. In some algorithms, the resulting proceedings in the objective space are used to steer the search in the decision variable space. When this happens, the proceedings in both spaces must be coordinated in such a way that the creation of new solutions in the decision variable space is complementary to the diversity needed in the objective space. Figure 2 illustrate a typical scenario of decision versus objective space.

Most multi-objective optimization algorithms use the concept of domination [87]. In these algorithms, two solutions are compared on the basis of whether one dominates the other solution or not. The concept of domination is described in the following definitions (assuming, without loss of generality, the objective functions to be minimized).

Definition 1. *Given two decision or solution vectors x and y, we say that decision vector x weakly dominates (or simply dominates) the decision vector y (denoted by $x \preceq y$) if and only if $f_i(x) \leq f_i(y) \forall i = 1, \ldots, M$ (i.e., the solution x is no worse than y in all objectives) and $f_i(x) < f_i(y)$ for at least*

Table 2: Methods used for Training Kernel Parameters and Weights of RBFNs (Gaussian Kernels is Assumed).

Authors	Weights Optimization Procedure	Spread and Center Optimization Procedure
Wetterschereck *et al.*, 1992 [188]	Pseudo-Inverse Method	Unsupervised learning (k-means)
Chun-Tao *et al.*, 2009 [189]	Particle Swarm Optimization	Particle Swarm Optimization
Yang *et al.*, 2009 [45]	Particle Swarm Optimization	Subtractive Clustering
Senapati *et al.*, 2007 [190]	Particle Swarm Optimization	Particle Swarm Optimization and Genetic Algorithms
Billings *et al.*, 1995 [191]	Genetic Algorithms	Genetic Algorithms
Yuan *et al.*, 2008 [192]	Genetic algorithms	Genetic Algorithms
Yu and He, 2006 [193]	Differential Evolution	Differential Evolution
Shen, *et al.*, 2008 [194]	Chaos Immune System	Logistic equation
Yunna, *et al.*, 2008 [195]	Ant Colony Algorithm	Orthogonal least square
Ziyang, *et al.*, 2008 [43]	Least Mean Square Method	Fuzzy c-means
Zhao, *et al.*, 2007 [196]	GA with Hybrid Learning Algorithm	Hybrid Learning Algorithm
Chen, *et al.*, 1991, 1992 [197**?**]	Orthogonal least square methods	Linear Regression
Fatemi, *et al.*, 2005 [199]	Mean pf Parameters	Adaptive growing technique and Sorting
Tan, *et al.*, 1993 [200]	Stable Updating Rule	Stable Updating Rule
De Lacerda *et al.*, 1994 [201]	Genetic Algorithms	Genetic Algorithms

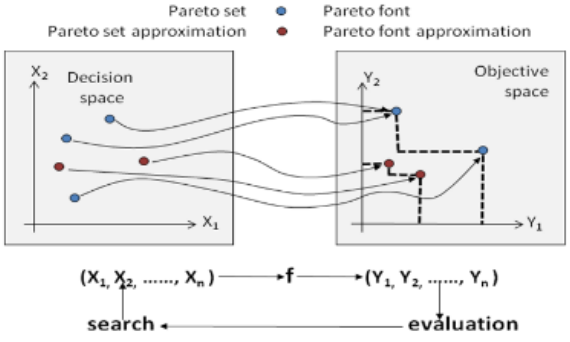

Figure 2: Decision vs. Objective Space in a multi-objective optimization problem.

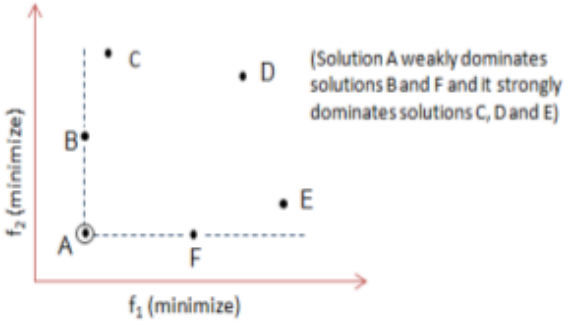

Figure 3: Dominance relation in two objective functions.

one $i \in \{1, 2, \ldots, M\}$ (i.e., the solution x is strictly better than y in at least one objective).

Definition 2. *A solution x strongly dominates a solution y (denoted by $x \prec y$), if solution x is strictly better than solution y in all M objectives.*

Figure 3 illustrates a particular case of the dominance relation in the presence of two objective functions.

However, if a solution x strongly dominates a solution y, the solution x also weakly dominates solution y, but not vice versa.

Definition 3. *The decision vector $x \in P$ (where P is the set of solution or decision vectors) is non-dominated with respect to set P, if there does not exit another $x^t \in P$ such that $f(x^t) \leq f(x)$.*

Definition 4. *Among a set of solution or decision vectors P, the non-dominated set of solution or decision vectors P' are those that are not dominated by any member of the set P.*

Definition 5. *A decision variable vector $x \in P$ where P is the entire feasible region or simply the search space, is Pareto-Optimal if it is non-dominated with respect to P.*

Definition 6. *When the set P is the entire search space, the resulting non-dominated set P' is called the Pareto-Optimal set. Mathematically, $P' = \{x \in P | x is Pareto - Optimal\}$.*

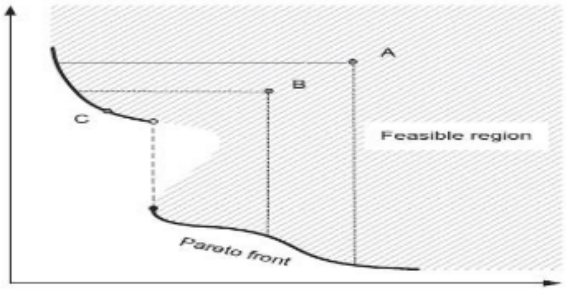

Figure 4: Pareto optimal front- the solution C is strictly better than B and then B is strictly better than A (source: [85]).

The non-dominated set P' of the entire feasible search space P is the globally Pareto-Optimal set.

Definition 7. *All Pareto-Optimal solutions in a search space can be joined with a curve (in two-objective space) or with a surface (in more than two-objective space). This curve or surface is termed as Pareto optimal front or simply Pareto front. In other words,* $PF = \{f(x)|x \in P^{'}\}$.

Figure 4 illustrates a particular case (e.g., when the feasible region is an uncountable set) of the Pareto front in the presence of two objective functions and is a non-increasing curve in R^2 and some arbitrary function may contain non-convex intervals and discontinuities.

It is to be noted that in practice, the complete Pareto Optimal set is not normally desirable (e.g., it may not be desirable to have different solutions that map to the same values in objective function space) or achievable. Thus a preferred set of Pareto optimal solutions should be obtained from practical point of view.

3.2 Multi-objective problem solving approaches in RBFNs

A multi-objective problem can be handled as a single objective optimization problem by using various classical methods such as weighted sum approach, ϵ-constraint method, weighted metric methods, value function method and goal programming methods [88]. In the weighted sum approach, multiple objectives are weighted and summed together to create a composite objective function. Optimization of this composite objective results in the optimization of individual objective functions. Goal programming methods suggest minimizing a weighted sum of deviations of objectives from user-specified targets. These conversion methods result in a single-objective optimization problem, which must be solved by using a single-objective optimization algorithm. These classical multi-objective optimization algorithms are having some difficulties particularly if the user wants to find multiple Pareto-optimal solutions [87]. First, only one Pareto-optimal solution can be expected to be found in one simulation run of a classical algorithm. Second, not all Pareto-optimal solutions can be found by some algorithms in non-convex MOPs. Although only one solution is needed for implementation, the knowledge of such multiple optimal solutions may help a designer to compare and choose a compromised optimal solution. A multi-objective optimization is, in general, more complex than a single-objective optimization, but the avoidance of multiple simulations runs, no artificial fix-ups, availability of efficient population-based optimization algorithms, and above all, the concept of dominance helps to overcome some of the difficulties and give a user the practical means to handle multiple objectives. Nevertheless, evolutionary algorithms (EAs) [87, 88] and swarm intelligence (SI) [89] approaches maintain a pool of potential solutions for the problem, thus making it easier to adapt them to solve MOPs.

The authors in [71, 90] have presented a multi-objective evolutionary algorithm for function approximation by optimizing the size, shape, and position parameters of RBFNs. While minimizing the bi-objectives such as model's prediction error and number of RBFs, a set of new mutation operators specially designed to evolve RBFNs. These new operators are based on two well-known matrix transformations: singular value decomposition (SVD) and orthogonal least squares (OLS), which have been used to define new mutation operators that produce local or global modifications in the radial basis functions (RBFs) of the networks (the individuals in the population in the evolutionary procedure). The authors in [91] have proposed a multi-objective structure selection method for RBF networks based on multi-objective genetic algorithm. The structure of RBF networks is encoded to the chromosomes in GA then evolves towards Pareto optimum for multi-objective functions concerned with *model accuracy* and *complexity*.

Methods aimed at efficiently controlling the greenhouse climate environment and optimizing the crop must take into account the influences of the outside weather, the actuators and the crop, which is achieved by the use of models. Improving prediction performance while reducing model complexity easily becomes conflicting, and giving rise to a multi-objective optimization problem. Ferreira *et al.* [92], have used a *multi-objective genetic algorithm* for the identification of *RBFNs* coupled models of humidity and temperature in a greenhouse.

Yen [93] has developed a model of hierarchical rank density genetic algorithm to evolve topology and parame-

ters of RBFN and then optimize two criteria like prediction error and RBFs in the hidden layer for effectively prediction of chaotic time series.

Guillen, *et al.* [94], have presented a parallel multi-objective GA that designs RBFNs to approximate functions by simultaneously minimizing network with the smallest number of neurons along with smallest error. The results confirm that the specialization on the different aspects of the design of RBFNs through the parallel approach could lead to obtain better results.

Ensemble learning has been an area of active research in machine learning community for improving the performance of classifier. In this context, Kondo *et al.* [95] have presented a method in which RBFNs ensemble has been constructed from Pareto-optimal set obtained by multi-objective evolutionary computation. Further, they have extended this idea [96] and applied to solve nonlinear dynamic system identification problem.

Despite of high capabilities of evolutionary multi-objective optimization certain multi-objective problems can be effectively solved in a deterministic way, taking advantages of non-linear programming. Two such works of Kokshenev and Braga [85, 97] are indeed effective approaches for the problem of inductive supervised learning within the context of multi-objective optimization. Two objectives such as empirical risk and model complexity have been optimized to get a harmonious RBFN. In particular, the work proposed in [97] is the idea of decomposition of the multi-objective problem into a set of convex sub problems led to a development of the multi-objective algorithm for finding Pareto-optimal solutions within a small class of hypotheses of RBF networks. Such an approach allows to approximate Pareto sets arbitrary well with the numbers of exact solutions of convex subproblems. However in [85], a deeper study of the previous results [97] has been presented and extends their application to larger classes of hypotheses. However, the computational complexity of these algorithms is high in comparison with other state-of-the-art machine learning methods.

Although there are few studies on the implementation of multi-objective RBF network training, but research on training of RBF network with multi-objective swarm intelligence is still new. In this direction, the work by [98] presents an adaptive evolutionary radial basis function (RBF) network algorithm to evolve accuracy and connections (centers and weights) of RBF networks simultaneously. The problem of hybrid learning of RBF network is discussed with the multi-objective optimization methods to improve classification accuracy for medical disease diagnosis. Here authors have used time variant multi-objective PSO to optimize objectives such as: i) accuracy: mean square error on training set $f_1 = \frac{1}{N}\sum_{j=1}^{N}(t_j - o_j)^2$, where N is the number of training samples, t_j is the actual output, and o_j is the desired output; and ii) complexity: ($f_2 = 1/2\sum_{j=1}^{M} w_j^2$, where M is the number of hidden neurons). Further, a few hybrid (mixture of local and global) learning strategy has been developed by [99], specifically memetic Pareto particle swarm optimization, memetic elitist Pareto non-dominated sorting genetic algorithm, and memetic elitist Pareto non-dominated sorting differential evolution to achieve compact RBFN model with both good prediction accuracy and prominent structure simultaneously.

Table 3 summarizes an attempt of RBFNs from the multi-objective perspective. To obtain Pareto set the approaches they have adapted vary from deterministic to non-deterministic. However, the criteria they are optimizing are same in number (i.e., two) with a minor variation in terminology. The application domain considered to evaluate their methods is also catering to the demand of present scenario.

4 RBFNs tools

In this Section, we have highlighted some state-of-the-art tools for implementing RBFNs. Except MATLAB, all other tools discussed here are open source.

KEEL[1]: Knowledge Extraction based on Evolutionary Learning (KEEL) is an open source (GPLv3) Java software tool which empowers the user to assess the behavior of evolutionary learning and soft computing based techniques for different kinds of DM problems: regression, classification, clustering, pattern mining, and so on [100]. Table 4 illustrates the different versions of RBFNs implementations along with their reference.

WEKA[2]: In this open source software a normalized Gaussian radial basis function network has been implemented in Java. It uses the k-means clustering algorithm to provide the basis functions and learns either a logistic regression (discrete class problems) or linear regression (numeric class problems). Symmetric multivariate Gaussians are fit to the data from each cluster. If the class is nominal it uses the given number of clusters per class. It standardizes all numeric attributes to zero mean and unit variance.

1 KEEL, (2013). http://sci2s.ugr.es/keel/algorithms.php#neural-networksforclassification

2 WEKA, (2013): http://weka.sourceforge.net/doc/weka/classifiers/functions/package-summary.html)

Table 3: Summary of RBFNs based on multi-objective approaches.

Sl. No.	Authors, Years	Approach	Benchmark Problems
1	Gonzalez *et al.*, 2001 [90]	Multi-objective Evolutionary Algorithm	Time Series Forecasting
2	Gonzalez *et al.*, 2003 [71]	Multi-objective Evolutionary Algorithm with New Genetic Operators	Function Approximation and Time Series Forecasting
3	Hatanaka *et al.*, 2003 [91]	Multi-objective Genetic Algorithms	Function Approximation
4	Ferreira *et al.*, 2005 [92]	Multi-objective Genetic Algorithms	Greenhouse Environmental Control
5	Yen, 2006 [93]	Hierarchical Rank Density Genetic Algorithm	Mackey-Glan Chaotic Time Series
6	Guillen *et al.*, 2006 [94]	Parallel Multi-objective Genetic Algorithm	Function Approximation
7	Kondo *et al.*, 2006 [202]	Multi-objective Evolutionary Algorithms	Classification
8	Kondo *et al.*, 2007 [203]	Multi-objective Evolutionary Algorithms	Non-linear Dynamic System Identification
9	Kokshenev and Braga, 2008 [97]	Inductive Supervised Learning (Non-linear Programming)	"Sinc function (Approximation) Wisconsin Breast Cancer Dataset (Classification)"
10	Kokshenev and Braga, 2010 [85]	Non-linear Programming	Twin Spiral, Noised Sinc Regression (artificial data) Wisconsin Breast Cancer Dataset, Abalone Dataset (Real Life data)
11	Qasem and Shamsuddin, 2011 [98]	Time Variant Multi-objective Particle Swarm Optimization	UCI Machine Learning Medical Related Dataset
12	Qasem and Shamsuddin, 2012 [99]	Multi-objective Hybrid Evolutionary Algorithms	UCI Machine Learning Dataset

Table 4: RBFNs Implementations in KEEL.

Acronyms	Name	Reference
RBFN-C	Radial Basis Function Neural Network for Classification Problems	[19]
Incr-RBFN-C	Incremental Radial Basis Function Neural Network for Classification Problems	[204]
Decr-RBFN-C	Decremental Radial Basis Function Neural Network for Classification Problems	[19]
EvRBFN-C	Evolutionary Radial Basis Function Neural Networks	[100]

MATLAB[3]: Two functions have been implemented in MATLAB for radial basis networks. The function *newrb* adds neurons to the hidden layer of a radial basis network until it meets the specified mean squared error goal. The syntax and meaning each argument has been described below.

net = newrb(P,T,goal,spread,MN,DF).

The larger *spread* is, the smoother the function approximation. Too large a spread means a lot of neurons are required to fit a fast-changing function. Too small a spread means many neurons are required to fit a smooth function, and the network might not generalize well. Call *newrb* with different spreads to find the best value for a given problem.

Similarly *newrbe* function of MATLAB very quickly designs a radial basis network with zero error on the design vectors. The syntax and meaning of arguments are described below.

net = newrbe(P,T,spread)

The larger the *spread* is, the smoother the function approximation will be. Too large a spread can cause numerical problems.

DTREG[4]: Software for Predictive Modeling and Forecasting implements the most powerful predictive modeling methods that have been developed including, Tree-Boost and Decision Tree Forests as well as Neural Networks, Support Vector Machine, Gene Expression Programming and Symbolic Regression, K-Means Clustering, Linear Discriminant Analysis, Linear Regression models and Logistic Regression models. Benchmarks have shown these methods to be highly effective for analyzing and modeling many types of data.

NeuralMachine[5]: NeuralMachine is a general purpose neural network modeling tool with the executable code generator. There are two versions of NeuralMachine - for MS-Windows and a Web-based version (runs in Internet Explorer across the Web). NeuralMachine allows creation of artificial neural networks (ANNs) with one hidden layer. Two types of networks, namely Multilayer perceptron and Radial Basis Function networks are supported in this version. In the case of Radial Basis Network the mapping function is of Gaussian type for the hidden layer and linear for the output layer.

NeuroXL[6]: The NeuroXL software is easy-to-use and intuitive, does not require any prior knowledge of neural networks, and is integrated seamlessly with Microsoft Excel. NeuroXL brings increased precision and accuracy to a wide variety of tasks, including: cluster analysis, stock price prediction, sales forecasting, sports prediction, and much more.

Netlab[7]: The Netlab toolbox is designed to provide the central tools necessary for the simulation of theoretically well founded neural network algorithms and related models for use in teaching, research, and applications development. It consists of a toolbox of MATLAB functions and scripts based on the approach and techniques described in [1], but also including more recent developments in the field. The Netlab library includes software implementations of a wide range of data analysis techniques, many of which are not yet available in standard neural network simulation packages. Netlab works with MATLAB version 5.0 and higher but only needs core MATLAB (i.e., no other toolboxes are required). It is not compatible with earlier versions of MATLAB.

5 RBFNs in approximation and interpolation

Approximations and interpolation is inevitable in science, engineering, and medical; indeed all around us in day-to-day life [101]. As we can see, the applications of general purpose methods for function approximations are manifold and important. The Weierstrass approximation theorem states that a continuous function $f(x)$ over a closed interval [a, b] can be approximated by a polynomial $\varphi_n(x)$ of degree n, such that:

$$|f(x) - \varphi_n(x)| \leq \varepsilon,\ x \in [a, b], \tag{14}$$

where, $\varepsilon > 0$ is a small quantity and n is sufficiently large [102].

However, RBFNs are particularly interesting when functions to be approximated: i) depend on many variables or parameters, ii) are defined by possibly very many data points, and iii) the data are scattered in their domain.

Interpolation of a given set of points is an important problem especially in higher-dimensional domains. Although polynomials are very powerful tool for interpolating a given set of points in one dimension, the use of these functions leads to difficulties in higher-dimensional domains. When we employ these functions, the arrangement

3 MATLAB, (2013): http://www.mathworks.co.kr/products/matlab/

4 DTREG, (2013). http://www.dtreg.com/index.htm

5 NeuralMachine, (2013). http://www.data-machine.nl/neuralmachine.htm

6 Neuroxl, (2013). http://www.neuroxl.com/

7 Netlab, (2013). http://www1.aston.ac.uk/eas/research/groups/ncrg/resources/netlab/

Table 5: Description of the Arguments.

Name	Description
P	R-by-Q matrix of Q input vectors
T	S-by-Q matrix of Q target class vectors
Goal	Mean squared error goal (default = 0.0)
Spread	Spread of radial basis functions (default = 1.0)
MN	Maximum number of neurons (default is Q)
DF	Number of neurons to add between displays (default = 25)

of the points in the domain of the problem should have a certain form. However, this limits us when interpolation of a scattered set of points is needed. Radial basis functions are very efficient instruments for interpolating a scattered set of points which have been used in the last 20 years [103–105]. The interpolation problem is the construction of a curve $y(x)$ which passes through a given set of data points (x_i, y_i), for $i = 0, 1, \ldots, n$ where the data points are such that $a = x_0 < x_1 < \ldots < x_n = b$. The constructed curve $y(x)$ can then be used to estimate the values of y at positions x which are between the end points a and b (interpolation) or to estimate the value of y for x exterior to the end points (extrapolation).

Let us discuss some of the contributions under these two categories. In [106], RBFNs for polynomial approximation in term of 1-D and 2-D function approximation have been used and done a comparative study among Beta wavelet, classical wavelet, and polynomial network. Their simulation result shows a good generalization ability. In [107], authors have studied the properties of RBF approximation near the ends of an interval in 1-D and towards edges of 2-D. Lendasse *et al.* [108], have proposed a method of function approximation by RBFN. The improvement they have made consists of weighting of inputs by the coefficients obtained through a linear model. These methods have then been tested for the determination of the price of a call option. The weighted RBFN gives 97% where as the classical RBFN gives 93% accuracy. Chen *et al.* [109] have explored representation capability of RBFNs. The capability of approximation to nonlinear functional and operators by RBFNs is revealed using sample data either in frequency domain or in time domain. Dyn and Ron [110] have provided a general tool for extending approximation schemes that use integer translates of a basis function to the non-uniform case. They introduce a single, relatively simple, conversion method that preserves the approximation orders provided by a large number of schemes. Ron [111] devoted his work towards L_2 approximation orders with principal spaces generated by an RBF with a thorough analysis of least-squares approximation orders. He has applied the results in different functions like polyharmonic splines, multiquadric, Gaussian kernel, and other functions. He has shown that Sobolev space can be approximated to a better rate.

Fasshauer [112] has proposed the use of smoothing operation at each step of multilevel approximation algorithm to improve the convergence rate of algorithm. In his work he suggested a different approach to smoothing, i.e., the use of a pre-computed hierarchy of smooth function by which the cost of smoothing reduces to zero. Li and Micchelli [113], have studied approximation by radial basis functions including Gaussian, multiquadric, and thin plate spline functions, and derive order of approximation under certain conditions.

Golberg *et al.* [114] have given information on the use of RBFs in dual respiratory method (DRM), particularly in thin plates splines. They have pointed out that the omission of the linear terms could have biased the numerical results. They also showed that a full understanding of convergence behavior of the DRM requires one to consider both interpolation and BEM errors [115], since the later can offset the effect of data approximation [116]. Karur *et al.* [117] have used DRM in their work. The DRM is a class of boundary element techniques where the domain integral resulting from the non homogeneous terms in Poisson type equations is transferred to equivalent boundary integral by suitable approximation function. They have used RBF as approximating functions for interpolation techniques. In their work they examined convergence property of RBF for two dimensional problems. Then they have used RBF for approximation in DRM to solve non-linear Poisson type equations.

Hua *et al.* [118] have presented a new no-reference perceptual blur metric by using the RBFN which is based on orthogonal least squares learning algorithm. In their work they transform the problem of quality estimation to a problem of function approximation and then solve the problem by using OLS-RBF network. OLS-RBF network uses an orthogonal least squares learning algorithm to select suitable centers for the RBF, which makes the training pro-

Table 6: Description of Arguments.

Name	Description
P	R-by-Q matrix of Q input vectors
T	S-by-Q matrix of Q target class vectors
Spread	Spread of radial basis functions (default = 1.0)

cedure simpler. The performance obtained from experiment is consistent with subjective evaluation. Iske [119] in his work provides a new approximation scheme which achieves to combine well known features from RBF interpolation with developments concerning scattered data filtering. He has provided an efficient algorithm for the purpose of selecting a suitable basis of approximation space automatically. The methodology of calibration based on the use of cubic B-splines for total least-squares approximation of forward static characteristics of measurements channels and on the use of RBFNs for approximation of the inverse static characteristics is developed and examined by Kluk *et al.* [120]. Li and Micchelli [113], have studied approximation by RBFs including Gaussian, multi-quadric, and thin plate spline functions. Schaback [121], in his work compares RBF interpolants on different spaces. The spaces are generated by other RBFs. The result gives new idea for further research. Zhou *et al.* [122] have investigated an approximation method based on a class of RBFNs for solving the regulator equations. They have shown that the RBF neural networks can solve the regulator equations up to a prescribed arbitrarily small error, and this small error can be translated into a guaranteed steady-state tracking error for the closed-loop system.

Some of the pioneering works in the context of interpolation is summarized as follows. For solving interpolation equations an iterative procedure can be used [123]. In this work k^{th} iteration calculates the element in a k-dimensional linear subspace of radial functions that is closest to the required interpolant, the subspace being generated by a Krylov construction that employs a self-adjoint operator A. Foley, *et al.* [124] presented a localized approach by decomposing the domain into an arbitrary triangulation that forms overlapping regions. For each region, a radial basis method is applied to a much smaller number of points and the local interpolants are blended using $C^{'}$ rational hybrid cubic Bezier triangle functions. [126] gave a theoretical justification for combining the compactly supported RBF with numerical technique in space decomposition, which is based on Schwarz domain decomposition. The method overcomes the ill-conditioning problem resulted from using RBF as a global interpolant.

Wu *et al.* [127] have introduced a suitable variation formulation for the local error of scattered data interpolation by RBFNs, the error can be bounded by a term depending on the Fourier transform of the interpolated function f and a certain "Kriging function", which allows a formulation as an integral involving the Fourier transform of OE. Jackson [128] proved that maximum difference between a sufficiently smooth function and its quasi-interpolant is bounded by a constant multiple of (hn+1). This is shown that such a quasi-interpolation formula can reproduce polynomials of degree n.

Wendland [129] has combined the theory of RBF interpolation with a partition of unity method to solve large-scale, scattered data problems. Further, Wendland [130] has given an overview on numerical aspects of multivariate interpolation and approximation by RBFNs. It is well known that for some basic functions Φ, hierarchical and fast multipole-like methods can greatly reduce the storage and operation counts for fitting and evaluating RBFs. Beatson *et al.* [131], have developed the mathematics required by methods of these types for polyharmonic splines. For faster evaluation of RBFN many authors like Beatson, *et al.* [132], have used domain decomposition methods for solving the RBF interpolation equations. In their work they have discussed different problems in three sections. First section provides efficient ways of setting up and solving small-to-medium sized RBF interpolation problems. The second section describes a natural domain decomposition method for the interpolation equation. And finally last section describes some algorithmic details and numerical results of a domain decomposition interpolatory code for polyharmonic splines in 2 and 3 dimensions. Oliveira, *et al.* [133] and Rippa [134] have discussed about the accuracy of interpolating scattered data with RBFs. The accuracy depends on a shape parameter c of the RBF. They have shown numerically that, the optimal value of c (the value of c that minimizes the interpolation error) depends on the number and distribution of data points, on the data vector, and on the precision of the computation. They present an algorithm for selecting a good value for c.

6 RBFNs in classification and prediction

Classification and prediction are two methods of data analysis that can be used to extract models describing classes or to predict future trends of underlying data. Such analysis can help provide us with a better understanding of the data at large. Broadly, classification predicts categorical (discrete, unordered) labels (that uses training data to generate a single complex rule or mathematical equation that assigns data items to one of several distinct categories), *prediction* models continuous-valued functions. For example, we can build a classification model to categorize bank loan applications as either safe or risky, a patient has cancer (yes, no) based on various medical data, whereas a prediction model to predict the expenditures in dollars of potential customers on computer equipment given their income and occupation. Furthermore, classification and prediction have numerous applications, including fraud detection, target marketing, performance prediction, manufacturing, and medical diagnosis. Many classification and prediction methods [137] have been proposed by researchers in neural networks community [135, 136]. However, RBFNs has attracted a lot of attention in last couple of years [138, 139]. Some of the proposals in this direction are discussed below.

For monitoring the drinking water quality Bouamar [140] has used RBFNs and support vector machine (SVM) and then evaluated their performance on real data, corresponding to the criterions recognition rate, the training time, and robustness. Geuzouri, *et al.* [141], have presented an extended form of RBFN called temporal-radial basis function, which can be used for decision rules and classification in spatio-temporal domain e.g., speech recognition, robotic applications, economic fluctuations, etc. Wang *et al.* [142], have presented a modular neural classifier for protein sequences with improved classification criteria. The architecture of the proposed model is a modular RBF neural network with a compensational combination at the transition of output layer. The connection weights between the final output layer and the transition output layer are optimized by delta rule, which serve as an integrator of the local neural classifiers. To enhance the classification reliability, they have presented two heuristic rules to apply to decision making. Experimental results with performance comparisons are carried out between single neural classifiers and the proposed modular neural classifier.

Bruzzone *et al.* [143] have proposed a supervised technique for training RBFNs classifiers. Their experimental results confirm that the overall classification error made by the classifier is reduced and have shown a more stable behavior of the classification error versus variations in both the number of hidden units and the initial parameters of the training process. Krzyzak *et al.* [144] have considered two approaches: the selection of the RBF classifier via nonlinear function estimation and the direct method of minimizing the empirical error probability. Oliveira *et al.* [133], have suggested a dynamic decay adjustment algorithm for training RBFNs and probabilistic neural networks (PNNs). They have argued that parameters of their algorithms will not heavily influence classification performance. Chang *et al.*, [145] have proposed a Self-Organizing Map (SOM) neural network to select more appropriate centers for RBFN and he has also proposed a Modular radial basis function (MRBF) neural network to improve the classification rate and speed up of the training time. Daqi *et al.*, [146] have proposed a cascade RBF-LBF networks for classification problems which is able to optimally determine the structures and parameters of the RBF-LBF networks with the characteristic of sample distributions. It has higher convergence rate and classification precision compared to feed-forward two-layered LBF and RBF networks. Dybowxi [147],has described the use of RBFNs with Gaussian basis functions to classify incomplete feature vectors. The method lies on the basis that any marginal distribution of a Gaussian distribution can be determined from the mean vector and covariance matrix of the joint distribution. A novel paradigm has been proposed by El-Zooghby *et al.* [148] where data information is encapsulated in determining the structure and initial parameters of the RBFN classifier before learning takes place. A hybrid learning algorithm [149, 150] is used to train the RBFNs so that the dimension of the search space is drastically reduced in the gradient paradigm. Fu *et al.* [151], have proposed a modification to the training algorithm for the construction and training of the RBFN on unbalanced data [152] by increasing bias towards the minority classes. The weights inversely proportional to the number of patterns of classes are given to each class in the mean square error function. Their experimental results show that the proposed method is effective in improving the classification accuracy of minority classes while maintaining the overall classification performance.

Training technique can be formulated as an optimization problem that is used to minimize prediction error [153]. An evolutionary algorithm such as DE has been used for optimizing the parameters of RBFN as discussed in [154]. They have used different kernels of RBFN for classifying UCI data sets. They have shown that DE-RBF gives better result than GA-RBFN and simple RBFNs. Further,

Dash *et al.* [155] have used DE-RBFN for classifying weblog dataset. In their work they have shown that DE-RBFN gives better result than simple RBFN. Furthermore, Dash *et al.* [156] have developed a novel DE+RBFNs classifier with a special attention to removal of irrelevant features and data inconsistency [157] during the process of building the model.

Nagabhushan *et al.* [158] have presented two simple novel approaches for the classification of symbolic data. In their first step, they show the representation of symbolic data in binary form and then use a simple hamming distance measure to obtain the clusters from binaries symbolic data. This gives the class label and the number of samples in each cluster. In the second part they pick a specific percentage of significant data samples in each cluster and use them to train the adaptive auto-configuring neural network. Pulido *et al.* [159] have described how to apply a neural network based on RBFNs to classify multivariate data. The classification strategy was automatically implemented in a sequential injection analytical system. RBFNs have some advantages over Counter Propagation Neural Networks (CPNNs). In their work they have reduced the classification error from 20% to 13%. Qin *et al.* [160] have presented a PSO learning algorithm to automate the design of RBFNs, to solve pattern classification problems. Baragada *et al.* [161] have contributed a work based on the combination of polynomial vector with Fisher's discriminant function using the information of bit-plane and radial basis function. Each set of pixel is preprocessed to obtain interpolated pixels using PVD. This is further trained by Fisher's discriminant method that transforms once again into 2-D vector. A processing of training the RBF is adopted to obtain set of final weights. During implementation, the final weights are used to classify the presence of hidden information. Kurban and Besdok [162], have made a comparison of training algorithms of RBFNs for classification purposes. For training they have used Artificial Bee Colony (ABC) algorithm, Genetic algorithm, Kalman filtering algorithm, and gradient descent algorithm. Their experimental results show that the use of ABC algorithm results in better learning than those of others. Li and Tufts [170] have presented the concept of structures and algorithms of principal feature classification (PFC) based on RBFNs. PFC is intended to solve complex classification problems with large data sets.

For Tibetan speech recognition, Pan *et al.* [163] have designed classifier based on RBFN with the property of supervised learning based on gradient descent. Niranjan *et al.* [164] have compared three non-linear pattern classifiers in the recognition of static speech patterns. Two of these classifiers are neural networks (MLP and the modified Kanerva model). The third one is the method of RBFNs. The class boundaries generated by the different methods are compared on simple two-dimensional examples.

Shan [165] has presented a new method based on the vehicle license location. The segment method of vertical projection information with prior knowledge is used to split characters, and extract the statistical features. Then the RBFN is used to recognize characters with the feature vector as input. The results show that this method can recognize characters and improve the ability of license plate character recognition effectively. Sitamahalakshmi *et al.* [166], have proposed two classification methods, RBFN and probabilistic neural network to recognize hand written Telugu characters. They have shown that RBFN is better technique than PNN. Lee [167] has demonstrated RBFNs, BPNNs and k-nearest-neighbor (KNN) classifiers for large handwritten digit database. All provide similar low error rates. BPN is overall superior in memory usage. The RBF classifier requires more memory and more classification time, but less training time. The simple KNN classifier can also perform handwritten digit recognition, but requires a prohibitively large amount of memory and is much slower at classification. Hwang *et al.* [168] described a method to construct RBFN classifier efficiently and effectively. The method determines neurons of middle layer by a fast clustering algorithm and computes the optimal weights between middle and output layers. The authors applied the proposed method to construct an RBFN classifier for unconstrained handwritten digit recognition. Baboo *et al.* [169] have reported the results of recognition of handwritten Tamil characters. They have experimented with two different approaches. One is SOM based method wherein the interactions between the features in the classification are done using unsupervised learning. In the second approach, a combination of RBFN and SOM has been taken to investigate its dynamic training principles in their classification network. The classification ability of RBFN-SOM is compared to SOM network.

Li and Tufts [170], have presented the concept of structures and algorithms of principal feature classification (PFC) based on RBFNs. PFC is intended to solve complex classification problems with large data sets. Subashini *et al.* [171] have compared the use of polynomial kernel of SVM and RBFNs in ascertaining the diagnostic accuracy of cytological data obtained from the Wisconsin breast cancer database. Their research demonstrates that RBFNs outperformed the polynomial kernel of SVM for correctly classifying the tumors. Chu [172] has applied a novel RBFNs for cancer classification. He has taken three data sets and the results shows that RBFN is able to achieve 100% accuracy with much fewer genes.

Kumar *et al.* [173] have extended their work on the application of RBFNs for the unsupervised classification of images. An RBF like a spherical Gaussian, is a function that is symmetrical about a given spread or center point in a multidimensional space. The RBFN has been implemented on IRS 1C LISS-3 image of Kanpur and adjoining regions of India. Thakur *et al.* [174] have presented an efficient method for face recognition using principal component analysis (PCA) and RBFNs. PCA reduces the dimensionality of the image, and also retains some of the variations in the image data. After performing PCA, the hidden layer neurons of the RBFNs have been modeled by considering intra-class discriminating characteristics of the training images. Dhanalaksmi *et al.* Dhanalaksmi *et al.* [175], have used SVM and RBFNs algorithms to automatically classify audio clips into one of the six classes: news, sports, music, movies, advertisement, and cartoon. For these categories a number of acoustic features such as linear predictive coefficient, mel-frequency cepstral coefficients are extracted to characterize the audio content. Rosenblum *et al.* [176] have developed an RBF architecture that learns the correlation of facial feature motion patterns and human expressions. Each expression network was trained by viewing a set of sequences of one expression for many subjects. The trained neural network is then tested for retention, extrapolation, and rejection ability. Success rates are 88% for retention, 88% for extrapolation, and 83% for rejection. Using only 2-D face images and a small number of anchor points, Arad *et al.* [177] have shown that the method of RBFNs provides a powerful mechanism for processing facial expressions.

The performance of RBFNs with its variants on UCI data sets is listed in Table 7.

Tables 8 and 9 present a summary of algorithms which are performing better and worst with respect to different datasets.

As a summary note we provide the simple statistical analysis of the performance of classifiers. The motivation behind the statistical analysis is to provide a kind of road map by which one can compare the performance of his or her own proposed classifier with the algorithms so far been developed. Classification results are summarized in terms of the classification accuracy with respective datasets. We provide the range of the classification accuracy (i.e., maximum and minimum classification accuracy so far reported) with respect to different UCI datasets. Also we compare each classifier based on mean and standard deviation which is listed in Table 10.

Fanhui *et al.* [178] have presented a prediction model based on RBFNs for a time sequence data, then RBFN model and BP model is applied to the prediction of container handling capacity at Sanghai port. The result shows that prediction through RBFN model is faster and more accurate than BP model. Leonard *et al.*, have presented a novel network called the validity index network (VI net) [179]. The VI net, derived from RBFNs, fits functions and calculates confidence intervals for its predictions, indicating local regions of poor fit and extrapolation.

7 RBFN in differential equations

Many problems of science and engineering can be mapped to a set of differential equations (DEs) through a process of mathematical modeling. It is not easy to obtain their exact solutions, so numerical methods must be resorted to. Usually, in practice, only lower order approximations are employed resulting in a continuous approximation of the function across the mesh but not its partial derivatives. The discontinuity of the approximation of the derivative can adversely affect the stability of the solution. While higher-order schemes are necessary for more accurate approximations of the spatial derivatives, they usually involve additional computational cost. To increase the accuracy of the low-order scheme, it is required that the computational mesh be refined with a higher density of elements in the regions near the contours. This, however, is also achieved at the expense of increased computational cost. Alternative method to find an approximate particular solution is achieved by using radial basis function (RBF) [180].

Since [181], derived a modified RBFs method suitable for solving parabolic, hyperbolic, and elliptic partial differential equations (PDE), the method has been successfully applied for solving PDEs of various types. Later, Franke and Schaback [31], Wendland [130], Wu (1998) and Wu and Schaback [182], contributed some theories on the solvability and error bounds in applying the RBFs to solve the PDEs. The advantage of RSFs is that they involve a single independent variable regardless of the dimension of the problem. They prove particularly attractive when the domain cannot be expressed as product domains of lower dimensions. In all the interpolation methods for scattered data sets, RBFs outperforms all the other methods regarding accuracy, stability, efficiency, memory requirement, and simplicity of the implementation. In a similar study, Stead (1984) examined the accuracy of partial derivative approximations over scattered data sets, also concluding that RBFs performed more accurately compared to other considered methods.

Table 7: The Classification Accuracy of Classifiers Based on RBFNs.

Sl. No.	Authors /Reference	Dataset	Accuracy in Percentage
1	Huang and Du, 2008 [76]	Glass	65.14
		Satellite Image	89.37
2	Lampariello and Sciandrone, 2001 [205]	Breast cancer	99.43
3	Wang *et al.*, 2002 [142]	Protein Sequences	87
4	Sing *et al.*, 2003 [48]	Iris	92.5
		New thyroid	95.17
		Ionosphere	96.03
5	Barra *et al.*, 2006 [206]	Iris	95.58
		Wine	95.84
		Bupa	64.02
		Spiral	99.14
6	Yu and He, 2006 [193]	Iris	97.33
		Wine	96.31
		New-thyroid	94.44
		Glass	86.2
7	Falcao *et al.*, 2006 [28]	Satellite image	90.35
8	Nagabhushan and Padma, 2007 [263]	Soybean	98.5
		Zoo	98
9	Nagabhushan and Padma, 2007 (Platt, 1991 (RAN)) [158]	Soybean	100
		Zoo	100
10	Chen *et al.*, 2008 (SC) [46]	Wine	97.71
		WBCD	95.26
		New thyroid	93.9
		Ionosphere	91.45
		Iris	96
11	Chen *et al.*, 2008 (PSOSC) [46]	Wine	97.22
		WBCD	95.37
		New thyroid	94.39
		Ionosphere	91.68
		Iris	96.67
12	Subashini *et al.*, 2009 [171]	Breast cancer	96.57
		Iris	93.9
		Wine	96.7
		Glass	90.7
13	Kurban and Besdok, 2009 GD [162]	Iris	91.8
		Wine	94.4
		Glass	91.5
14	Kurban and Besdok, 2009 (Genetic Algorithm) [162]	Iris	96.1
		Wine	97.3
		Glass	91.9
15	Kurban and Besdok, 2009 (Artificial Bee Colony) [162]	Iris	96.3
		Wine	97.9
		Glass	92.7

Table 7: Contd.

Sl. No.	Authors /Reference	Dataset	Accuracy in Percentage
16	Zainuddin and Lye, 2010 [49]	Iris	95.19
		Diabetes	65.91
		Breast Cancer	92.36
		Hepatitis	79.31
		Lung cancer	49.03
17	Gutierrez *et al.*, 2011 [29]	Hepatitis	85.38
		Glass	70.02
		Sonar	80.9
		Ionosphere	93.95
		Iris	96.8
		New thyroid	96.59
		Balance	94.24
19	Kokshenev *et al.*, 2010 [85]	Breast Cancer	98.3
20	Qasem *et al.*, 2010 [98]	Breast Cancer	96.53
		Diabetes	78.02
		Hepatitis	82.26
21	Oh *et al.*, 2011 [77]	Iris	99.36
22	Qasem *et al.*, 2012 [99]	Breast cancer	97.66
		Diabetes	77.34
		Heart	82.2
		Hepatitis	85.79
		Liver	74.26
		Iris	89.11
		Wine	75.67
		Lung cancer	67.78
23	Qasem *et al.*, 2012 [99]	Iris	83.78
		Breast cancer	96.78
		Diabetes	72.78
		Wine	72.18
		Heart	79.07
		Hepatitis	80.04
		Liver	62.63
		Lung cancer	66.67
24	Qasem *et al.*, 2012 [99]	Breast cancer	96.93
		Diabetes	77.07
		Heart	87.54
		Hepatitis	90.33
		Liver	67.87
		Wine	77.11
		Iris	86
		Lung cancer	71.39

Table 8: Algorithms with Best Accuracy.

Serial Number	Data Set	Name of the Algorithm	Accuracy
1	Iris	PSO	99.36
2	Wine	ABC	97.9
3	Glass	ABC	92.7
4	New thyroid	RBF	96.59
5	Diabetes	PSO	78.02
6	Hepatitis	MPPSON	90.33
7	Heart	MPPSON	87.54
8	Liver	MPPSON	74.26
9	Breast cancer	RBF	99.43
10	Lung cancer	MPPSON	67.78
11	Soybean	RBF	100
12	Zoo	RBF	100
13	Satellite Image	RBF	90.35
14	WBCD	PSO	95.37

Table 9: Algorithm with Worst Accuracy.

Serial No.	Data Set	Name of Algorithm	Accuracy
1	Iris	MPPSON	83.78
2	Wine	MPPSON	72.18
3	Glass	RBPNNs	65.14
4	New Thyroid	SOM	93.9
5	Diabetes	K-Harmonic Means	65.91
6	Hepatitis	K-Harmonic Means	79.31
7	Heart	MPPSON	79.07
8	Liver	MPPSON	62.63
9	Breast Cancer	K-Harmonic Means	92.36
10	Lung Cancer	K-Harmonic Means	49.03
11	Ionosphere	SOM	91.45
12	Soybean	ACRBFNN	98.5
13	Zoo	ACRBFNN	98.02
14	Satellite Image	RBPNN	89.37
15	WBCD	SOM	95.26

Table 10: Statistical Analysis.

Sl.No.	Dataset	Minimum	Maximum	Average	Standard Deviations
1	Iris	83.78	99.36	91.57	4.3997
2	Wine	72.18	97.9	85.04	10.238
3	Glass	65.14	92.7	78.92	11.5135
4	New thyroid	93.9	96.59	95.24	1.0489
5	Diabetes	65.91	78.02	71.96	5.0849
6	Hepatitis	79.31	90.33	84.82	4.1413
7	Heart	79.07	87.54	83.3	4.2828
8	Liver	62.63	74.26	68.44	5.8245
9	Breast cancer	92.13	99.43	95.78	2.4087
10	Lung cancer	49.03	67.78	57.85	10.5195
11	Ionosphere	91.45	97.46	94.45	2.6411
12	Soybean	98.5	100	99.25	1.0607
13	Zoo	98.02	100	99.01	1.4001
14	Satellite image	89.368	90.35	89.86	0.6944
15	WBCD	95.26	95.37	95.31	0.0778

Fasshauer [112] discussed the difference between globally and locally supported methods for numerical solution of partial differential equations. Basically for locally supported methods the important role of smoothing within a multilevel framework is demonstrated. A possible connection between multigrid finite elements and multilevel RBF methods with smoothing has been explored. Further, Fasshauer [183, 184] dealt with the numerical solution of differential equations by RBFs by two different experiments, namely 1) the solution of a two-point boundary value problem, 2) the solution of a two-dimensional Poisson equation. In the second method a multilevel collocation algorithm based on locally supported basis function is applied.

Frank *et al.* [31] studied meshless collocation methods using RBFs to approximate regular solution of system of equations with linear differential equations or integral operators. Power *et al.* [185] have presented a thorough numerical comparison between unsymmetric and symmetric RBF collocation methods for the numerical solution of boundary value problems for partial differential equations. Schaback *et al.* [186] have used RBF for solving partial differential equation and also they have generated sparse and well conditioned matrices.

Wendland [129, 130] has combined the theory of RBF network with the field of Galerkin methods to solve partial differential equations. Galerkin method helps to solve elliptic partial differential equations. This approach can be seen as mesh-less method. He restricted his work to second order partial differential equations. Jianyu *et al.* [187] used neural network for solving differential equations. In their work a new growing RBF-node insertion strategy is used in order to improve the net performance. The learning strategy is able to save computational time and memory space, resulting improved approximation results. A recent survey of the application of RBFNs in differential equations is presented in (Kumar and Yadav). In [16] authors have presented a novel method i.e., known as DE-FLANN.

In addition to the aforementioned tasks, RBFNs have been applied in diversified domain, which are enlisted in Table 11.

8 Discussion and future research directions

This paper presents a focused survey on the attempt of designing RBFNs over decades by researchers along with its numerous usage in interdisciplinary fields. First we presented the architecture of RBFNs, where the central problem is based on the selection of an appropriate kernel (or basis function), number of hidden neurons (or RBFs), and the optimization of weights associated between hidden and output layer neurons. As far as the selection of kernels is concerned, Gaussian kernel has been used widely for solving problems from approximation to classification. However, RBFNs using Gaussian kernel is highly sensitive to center and spread. Hence, many proposals under the umbrella of unsupervised learning have been developed to select the parameters appropriately. Similarly, there are lots of independent proposals have been discussed for fix-

Table 11: RBFNs in Domain Specific Applications.

Sl. Number	Application Areas	References
1	Power Transmission	[207, 208]
2	Facial Expressions	[177]
3	Steganlysis	[161]
4	Handwritten Tamil Character Recognition	[169]
5	Ship Course Changing Control	[209]
6	Digital Communication Systems	[210]
7	Fractals	[211]
8	Inverse Filter	[212]
9	Optimal Equalization Solution	[197]
10	Channel Equalization	[213]
11	Tracking, Time Variant Environments	[213]
12	Audio Signal Analysis	[214]
13	Gaseous Fuel Engine	[215]
14	Neural Spike Sorting	[216]
15	Power System	[217]
16	Plants	[218]
17	Surface Recovery	[219, 264]
18	Fault Detection	[220]
19	Direction of Arrival Estimation Computations in Antenna Arrays	[148]
20	Telegraph Equation	[221]
21	Non-linear Filtering Problem	[222]
22	Irregular Terrain	[223]
23	Artificial Intelligence	[224]
24	Channel Equalization	[267]
25	Multi-focus Image	[225]
26	Knowledge Enhancement	[226]
27	Dipole Localization	[227]
28	Traditional Proportional Navigation Guidance Laws	[220]
29	Non-linear System Identification	[228, 229]
30	Permanent Magnet Synchronous Motor	[40]
31	Automatic Covariate Selection	[29]
32	Gas Sensor	[230]
33	Options Pricing Modeling	[231]
34	Hydrologic System	[232]
35	Health Monitoring	[233]
36	Real-Time Operations	[44]
37	GSM	[228, 268]
38	Knowledge Extraction	[234]
39	DC Motor	[235]
40	Non-linear Function Estimation	[236]
41	Sensor	[162]
42	Maxwell's equations	[237]
43	Three-dimensional scalar Helmholtz equation	[226]
44	Rainfall-off Model	[238]
45	Hydrodynamics	[239]

Table 11: Contd.

Sl. Number	Application Areas	References
46	Water Quality Simulation	[240]
47	Mobile Robot	[241]
48	Particleboard Glue Mixing & Dosing System	[239, 269]
49	Dynamic Decay Adjustment Algorithm	[133]
50	Rainfall Forecasting	[242]
51	Knowledge Extraction	[234]
52	Regulation Network	[243]
53	Mechanism Analysis	[244]
54	Turbo Equalization (TEQ) Scheme	[245]
55	Human Expressions	[176]
56	Forecasting of Monthly Sardines Catches	[246]
57	ECG Signal Reduction	[247]
58	Digital Communication Channels	[248, 249]
59	Toeplitz operators	[32]
60	Interpolation matrix	[250]
61	Defect datasets of NASA	[251]
62	Computational Vision	[252]
63	Signal Estimation	[212]
64	Signal Processing	[253]
65	Blind Equalization	[254, 270]
66	Mackey-Glass Chaotic Time Series	[255]
67	Thin and Thick Sandwich Plates	[256]
68	Logistics Service Providers	[41]
69	NP-complete Problem	[257]
70	Laser Sintering Process	[42]
71	Tank Reactor	[258]
72	Power System Restoration	[259]
73	Kernel Regression Estimator	[260]
74	Flapping Wings, Aerospace Engineering	[261]
75	Bioinformatics	[29]
76	Pattern Recognition	[262, 265, 266]

ing the problem of selection of hidden layer neurons. Since RBF expansions are linearly dependents on the weights, therefore, a globally optimum least squares interpolation of non-linear maps can be achieved. However, singular value decomposition (SVD) is widely used to optimize the weights.

Secondly, we have exhaustively discussed the multi-criteria problem of RBFNs. In contrast to single objective optimization of RBFNs, the multi-objective formulation provides flexibility to user for choosing a solution of his/her interest from a pool of Pareto optimal set in one execution. In other words, when the algorithm completes, it returns a complete set of solution with different compromises between the two objectives (empirical risk and number of RBFs), while other approaches, which obtain only one solution per execution have to be executed several times with different configurations to obtain separate solution.

Thirdly we have highlighted many open source software tools for testing RBFNs. However, we found that no tools have been addressed the multi-criteria issues of RBFNs. To the best of our knowledge, there is no such report/document has been reported for a fair comparison of tools in connection to the successful application of RBFNs.

Finally we have discussed the potential use of RBFNs in classification, prediction, interpolation, approximation, and for solving partial differential equations. Additionally some domain specific usage of RBFNs is presented. In specific, RBFNs demonstrated to be competitive alternative for many approximation and classification problems. However, RBF networks still open avenues for new kernels along many issues still remain unsolved. In addition RBF networks need to be systematically evaluated and compared with other new and traditional tools. We believe that the multidisciplinary nature of RBF networks will generate more research activities and bring about more fruitful outcomes in the future.

Acknowledgement: Authors gratefully acknowledge the support of the Original Technology Research Program for Brain Science through the National Research Foundation (NRF) of Korea (NRF: 2010-0018948) funded by the Ministry of Education, Science, and Technology.

References

[1] Bishop C., Neural Networks for Pattern Recognition. Oxford University Press, USA, 1995.

[2] Callan R., The Essence of Neural Networks. Prentice Hall PTR Ur Saddle River, NJ, USA, 1998.

[3] Haykin S., Neural Networks: A Comprehensive Foundation Neural Networks. 2nd Edition, Prentice Hall, Upper Saddle River, NJ, 1999.

[4] Acosta F.M.A., Radial Basis Functions and Related Models: An Overview. Signal Processing, 45(1), 37-58, 1995.

[5] Bors A.G., Introduction of the Radial Basis Function Networks. Online Symposium for Electronics Engineers, 1(1), 1-7, 2001.

[6] Buhmann M.D., Multivariate Cardinal Interpolation with Radial Basis functions. Constructive Aoximation, 6(3), 225-255, 1990.

[7] Buhmann M.D., On Quasi-Interpolation with Radial Basis Functions. J. Aox.Theory, 72(1), 103-130, 1993a.

[8] Buhmann M. D., New Developments in the Theory of Radial Basis Function Interpolation. In Jette, K. and Utreras, F. (eds.), Multivariate Aoximation: From CAGD to Wavelets, World Scientific Publishing, Singapore, 1993b, 35-75.

[9] Buhmann M.D., Mehaute A.L., Knot Removal with Radial Basis Function Interpolation. C R Acad Science Paris Ser I Math., 320, 501-506, 1995.

[10] Buhmann M.D., Radial Functions on Compact Surt. Proceeding of Edinburgh Mathematical Society, 41(1), 33-46, 1998.

[11] Buhmann M.D., Radial Basis Functions. Acta Numerica Cambridge University Press, 9, 1-38, 2000.

[12] Buhmann M.D., Radial Basis Functions: Theory and Implementations. vol.12, Cambridge University Press University Press, Cambridge, 2003.

[13] Caiti A., Magenes G., Panisini T., Simpson R., Smooth Aoximation by Radial Basis Functions: three case studies. J. A. Sci. Comp, 1, 88-113, 1994.

[14] Wu S., Chow T.W.S., Induction Machine Fault Detection Using SOM-based RBF Neural Networks. IEEE Transactions on Industrial Electronics, 51(1), 183-194, 2004.

[15] Powell M.J.D., Radial Basis Functions for Multivariable Interpolation: A Review. In: J.C. Mason, M.G. Cox (Eds.), Algorithms for the Aoximation, Clarendon Press, 1987a, 143-167.

[16] Powell M.J.D., Radial Basis Function Aoximations to Polynomials. In: D.F. Griths, G.A. Watson (Eds.), Numerical Analysis 87, Longman Publishing Group, Chicago, 1987b, 223-241.

[17] Powell M.J.D., The Theory of Radial Basis Function Aoximation in 1990. Advances in Numerical Analysis, 2, 105-210, 1992.

[18] Powell M.J.D., Recent Research at Cambridge on Radial Basis Functions. New Developments in Aoximation Theory, International Series of Numerical Mathematics, 215-232, 1999.

[19] Broomhead D.S., Lowe D., Multi-variable Functional Interpolation and Adaptive Networks. Complex Systems, 2, 321-355, 1988.

[20] Gyorfi L., Kohler M., Krzyzak A., Walk H., A Distribution Free Theory of Non-Parametric Regression. Springer Verlag, New York Inc., 2002.

[21] Bishop C., Improving the Generalization Properties of Radial Basis Function Neural Networks. Neural Computation, 3(4), 579-588, 1991.

[22] Park J., Sandberg I.W., Universal Aoximation Using Radial Basis Function Networks. Neural Computation, 3(2), 246-257, 1991.

[23] Beatson R.K., Cherrie J.B., Mouat C.T., Fast Fitting of Radial Basis Functions: Methods based on Preconditioned GMRES Iteration. Advances in Computer Mathematics, 11(2), 253-270, 1999.

[24] Patrikar A.M., Aoximating Gaussian Mixture Model on Radial Basis Function Networks with Multilayer Perceptron. IEEE Transactions on Neural Networks, 24(7), 1161-1166, 2013.

[25] Mao K.Z., Huang G., Neuron Selection for RBF Neural Network Classifier Based on Data Structure Preserving Criterion. IEEE Transactions on Neural Networks, 15(6), 15531-1540, 2005.
[26] Beatson R.K., Newsam G.N., Fast Evaluation of Radial Basis Functions: I. Computers Math., 24(12), 7-19. 34. Bishop, C. (1991). Improving the Generalization Properties of Radial Basis Function Neural Networks. Neural Computation, 3(4), 579-588, 1992.
[27] Schwenker F., Kestler H.A., Palm G., Höher M., Similarities of LVQ and RBF Learning - A Survey of Learning Rules and the Aication to the Classification of Signals from High-Resolution Electrocardiography. Proceedings IEEE SMC, 1, 646-651, 1994.
[28] Falcao A., Langlois T., Wichert A., Flexible Kernels for RBF Networks. Neurocomputing, 69, 2356-2359, 2006.
[29] Fernandez-Navarro F., Hervas-Martinez C., Cruz-Ramirez M., Gutierrez P.A., Valero A., Evolutionary q-Gaussian Radial Basis Function Neural Network to Determine the Microbial Growth/no Growth Interface of Staphylococcus Aureus. Aied Soft Computing, 11(3), 3012-3020, 2011.
[30] Yamada T., Wrobel L.C., Properties of Gaussian Radial Basis Functions in the Dual Reciprocity Boundary Element Method. Zeitschrift for Angewandte Mathematik und Physik (ZAMP), 44(6), 1054-1067, 1993.
[31] Franke C., Schaback, R., Convergence Orders of Meshless Collocation Methods using Radial Basis Functions. Advances in Computational Mathematics, 8(4), 381-399, 1998.
[32] Baxter B.J.C., The Interpolation Theory of Radial Basis Functions. University of Cambridge, Cambridge, 1992.
[33] Bors A.G., Pitas, I., Object Classification in 3-D images using Alpha-trimmed Mean Radial Basis Function Network. IEEE Transaction on Image Processing, 8(12), 1744-1756, 1999.
[34] Bors A.G., Pitas I., Median Radial Basis Functions Neural Network. IEEE Transactions on Neural Networks, 7(6), 1351-1364, 1996.
[35] Hon Y.C., Schaback R., Zhou, X., An Adaptive Greedy Algorithm for Solving Large RBF Collocation Problems. Numerical Algorithms, 32, 13-25, 2003.
[36] Cherrie J.B., Beatson R.K., Newsam G.N., Fast Evaluation of Radial Basis Functions: Methods for Generalized Multi-quadrics in Rn. SIAM J Science Computation, 23(5), 1549-1571, 2002.
[37] Lim E.A., Zainuddin Z., An Improved Fast Training Algorithm for RBF Networks using Symmetry-Based Fuzzy C-Means Clustering. Matematika, 24(2), 141-148, 2008.
[38] Uykan Z., Guzelis C., Celebi M.E., Koivo H.N., Analysis of Input Output clustering for Determining Centers of RBFNs. IEEE Transactions on Neural Networks, 11(4), 851-858, 2000.
[39] Hunt K.J., Haas R., Murray-Smith R., Extending the Functional Equivalence of Radial Basis Function Networks and Fuzzy Inference Systems. IEEE Trans Neural Network, 7, 776-781, 1996.
[40] Gu Z., Li H., Sun Y., Chen Y., Fuzzy Radius Basis Function Neural Network Based Vector Control of Permanent Magnet Synchronous Motor. In Proceedings of IEEE International Conference on Mechatronics and Automation (ICMA 2008), 224-229, 2008.
[41] Kong F., Zhang Z., Liu Y., Selection of Logistics Service Provider Based on Fuzzy RBF Neural Networks. In Proceedings of IEEE International Conference on Automation and Logistics (ICAL 2008), 1189-1192, 2008.
[42] Li X., Dong J., Zhang Y., Modeling and Aying of RBF Neural Network Based on Fuzzy Clustering and Pseudo-Inverse Method. In Proceedings of International Conference on Information Engineering and Computer Science (ICIECS 2009), 1-4, 2009.
[43] Ziyang Z., Zhisheng W., Yong H., Mingzhi G., Learning Method of RBF Network Based on FCM and ACO. In Proceedings of Control and Decision Conference (CCDC), 102-105, 2008.
[44] Kamalabady A.S., Salahshoor K., New SISO and MISO Adaptive Nonlinear Predictive Controllers Based on Self Organizing RBF Neural Networks. In Proceedings of 3rd International Symposium on Communications, Control, and Signal Processing (ISCCSP 2008), 703-708, 2008.
[45] Yang P., Zhu Q., Zhong X., Subtractive Clustering Based RBF Neural Network Model for Outlier Detection. J. Comput., 4(8), 755-762, 2009.
[46] Chen J.Y., Qin Z., Jia J., A PSO-Based Subtractive Clustering Technique for Designing RBF Neural Networks. In Proceedings of IEEE World Congress on Computational Intelligence (Evolutionary Computation' 2008), 2047-2052, 2008.
[47] Liu H., He J., The Aication of Dynamic K-means Clustering Algorithm in the Center Selection of RBF Neural Networks. 3rd International Conference on Genetic and Evolutionary Computing, 2009. WGEC'09, 488-491, 2009.
[48] Sing J.K., Basu D.K., Nasipuri M., Kundu M., Improved k-means Algorithm in the Design of RBF Neural Networks. In Proceedings of TENCON Con Convergent Tech Asia-Pacific Region, 2, 841-845, 2003.
[49] Zainuddin Z., Lye W.K., Improving RBF Networks Classification Performance by using K-Harmonic Means. World Academy of Science, Engineering and Technology, 62, 983-986, 2010.
[50] Huilan J., Ying G., Dongwei L., Jianqiang X., Self-adaptive Clustering Algorithm Based RBF Neural Network and its Aication in the Fault Diagnosis of Power Systems. In Proceedings of IEEE/PES Transmission and Distribution Conference and Exhibition: Asia and Pacific, Dalian, 1-6, 2005.
[51] Chen S., Hong X., Luk B.L., Harris C.J., Construction of Tunable Radial Basis Function Networks Using Orthogonal Forward Selection. IEEE Trans. Syst. Man. Cybern. B Cybern., 39(2), 457–466, 2009.
[52] Chen S., Wang X.X., Hong X., Harris C.J., Kernel Classifier Construction Using Orthogonal Forward Selection And Boosting With Fisher Ratio Class Separability Measure. IEEE Trans. Neural Netw., 17(6), 1652–1656, 2006.
[53] Hong X., A Fast Identification Algorithm for Box-Cox Transformation Based Radial Basis Function Neural Network. IEEE Trans. Neural Network, (17)4, 1064–1069, 2006.
[54] Wong Y.W., Seng K.P., Ang L.M., Radial Basis Function Neural Network with Incremental Learning for Face Recognition. IEEE Trans. Syst. Man. Cybern. B Cybern., 41(4), 940–949, 2011.
[55] Yingwei L., Sundararajan N., Saratchandran P., A Sequential Learning Scheme for Function Aoximation using Minimal Radial Basis Function Neural Networks. Neural Computation, 9(2),461-478, 1997.
[56] Leonardis A., Bischof H., An Efficient MDL-Based Construction of RBF Networks. Neural Networks, 11(5), 963-973, 1998.
[57] Alexandridis A., Chondrodima E., Sarimveis H., Radial Basis Function Network Training using a Nonsymmetric Partition of the Input Space and Particle Swarm Optimization. IEEE Transactions on Neural Networks and Learning Systems, 24(2), 219-230, 2013.
[58] Orr, M., Optimising the Widths of Radial Basis Functions. In Proc. 5th Brazilian Symp. Neural Netw., 26–29, 1998.

[59] Yao W., Chen X., Zhao Y., Tooren M.V., Concurrent subspace width optimization method for RBF neural network modeling, IEEE Trans. Neural Netw. Learn. Syst., 23(2), 247–259, 2012.
[60] Yao W., Chen X., Tooren M.V., Wei Y., Euclidean Distance and Second Derivative Based Widths Optimization of Radial Basis Function Neural Networks. In Proc. Int. Joint Conf. Neural Netw., 18-23 July 2010, Barcelona, Spain, 1–8, 2010.
[61] Orr M.J., Regularization in the Selection of Radial Basis Function Centers. Neural Computation, 7(3), 606-623, 1995.
[62] Su F.S., Chuang C.C., Tao C.W., Jeng J.J., Hsiao C.C., Radial Basis Function Networks with Linear Interval Regression Weights for Symbolic Interval Data. IEEE Trans. Syst. Man. Cybern. B Cybern., 42(1), 69–80, 2012.
[63] Zhang D., Deng L.F., Cai K.Y., So A., Fuzzy Nonlinear Regression with Fuzzified Radial Basis Function Network. IEEE Trans. Fuzzy Syst., 13(6), 742–760, 2005.
[64] Langari R., Wang L., Yen J., Radial Basis Function Networks, Regression Weights, and the Expectation-Maximization Algorithm. IEEE Trans. Syst., Man., Cybern. A, Syst. Humans, 27(5), 613–623, 1997.
[65] Andrieu C., Freitas N.D., Doucet A., Robust Full Bayesian Learning for Radial Basis Networks. Neural Comput., 13(10), 2359–2407, 2001.
[66] Huan H.X., Hien D.T.T., Tue H.H., Efficient Algorithm for Training Interpolation RBF Networks with Equally Spaced Nodes. IEEE Trans. Neural Netw., 22(6), 982–988, 2011.
[67] Xie T., Yu H., Hewlett J., Rozycki P., Wilamowski B., Fast and Efficient Second-Order Method for Training Radial Basis Function Networks. IEEE Trans. Neural Netw. Learn. Syst., 23(4), 609–619, 2012.
[68] Gutiérrez P.A., Hervás-Martínez C., Martínez-Estudillo F.J., Logistic Regression by Means of Evolutionary Radial Basis Function Neural Networks. IEEE Trans. Neural Netw., 22(2), 246–263, 2011.
[69] Sarimveis H., Alexandridis A., Mazarakis S., Bafas G., A New Algorithm for Developing Dynamic Radial Basis Function Neural Network Models Based on Genetic Algorithms. Comput. Chem. Eng., 28(1–2), 209–217, 2004.
[70] González J., Rojas I., Pomares H., Ortega J., RBF Neural Networks, Multi-objective Optimization and Time Series Forecasting. LNCS, Springer-Verlag, 2084, 498–505, 2001.
[71] Gonzalez J., Rojas I., Ortega J., Pomares H., Fernandez F., Javier D., Antonio F., Multi-objective Evolutionary Optimization of the Size, Shape, and Position Parameters of Radial Basis Function Networks for Function Aoximation. IEEE Transactions on Neural Networks, 14(6), 1478-1495, 2003.
[72] Buchtala O., Klimek M., Sick B., Evolutionary Optimization of Radial Basis Function Classifiers for Data Mining Aications. IEEE Trans. Syst. Man. Cybern. B Cybern., 35(5), 928–947, 2005.
[73] Valdez F., Melin P., Castillo O., Evolutionary Method Combining Particle Swarm Optimisation and Genetic Algorithms Using Fuzzy Logic for Parameter Adaptation and Aggregation: The Case Neural Network Optimisation for Face Recognition. Int. J. Artif. Intell. Soft Comput., 2(1–2), 77–102, 2010.
[74] Valdez F., Melin P., Castillo O., An Improved Evolutionary Method with Fuzzy Logic for Combining Particle Swarm Optimization and Genetic Algorithms. A. Soft Comput., 11(2), 2625–2632, 2011.
[75] Chen S., Hong X., Harris C.J., Particle Swarm Optimization Aided Orthogonal Forward Regression for Unified Data Modeling. IEEE Trans. Evol. Comput.,14(4), 477–499, 2010.
[76] Huang D.S., Du J.X., A Constructive Hybrid Structure Optimization Methodology for Radial Basis Probabilistic Neural Networks. IEEE Trans. Neural Networks, 19(12), 2099–2115, 2008.
[77] Oh S.K., Kim W.D., Pedrycz W., Joo S.C., Design Of K-Means Clustering-Based Polynomial Radial Basis Function Neural Networks (pRBF NNS) Realized with the Aid of Particle Swarm Optimization and Differential Evolution. Neurocomputing, 78(1), 121–132, 2012.
[78] Hassan F.R., Koh S.P., Tiong S.K., Chong K.H., Abdalla A.N., Investigation of Induction Motor Parameter Identification using Particle Swarm Optimization based RBF Neural Network. International Journal of the Physical Sciences, 6, 4564-4570, 2011.
[79] Liang N.-Y., Huang G.-B., Saratchandran P., Sundararajan N., A Fast and Accurate Online Sequential Learning Algorithm for Feed-forward Networks. IEEE Trans. Neural Netw., 17(6), 1411–1423, 2006.
[80] Cheu E.Y., Quek C., Ng S.K., ARPOP: An Atitive Reward-based Pseudo-Outer-Product Neural Fuzzy Inference System Inspired From The Operant Conditioning of Feeding Behavior in Aplysia. IEEE Trans. Neural Netw. Learn. Syst., 23(2), 317–329, 2012.
[81] Suresh S., Sundararajan N., Saratchandran P., A Sequential Multicategory Classifier Using Radial Basis Function Networks. Neurocomputing, 71(1), 1345–1358, 2008.
[82] Huang G.B., Saratchandran P., Sundararajan N., An Efficient Sequential Learning Algorithm for Growing and Pruning RBF (GAP-RBF) Networks. IEEE Transactions on Systems, Man, Cybernetics, 34(6), 2284-2292, 2004.
[83] Huang G., Saratchandran P., Sundararajan N., A Generalized Growing and Pruning RBF (GGAP-RBF) Neural Network for Function Aoximation. IEEE Transactions on Neural Networks, 16, 57-67, 2005.
[84] Huang G.-B., Zhou H., Ding X., Zhang R., Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. B Cybern., 42(2), 513–529, 2012.
[85] Kokshenev I., Braga A.P., An Efficient Multi-objective Learning Algorithm for RBF Neural Network, Neurocomputing, 73, 2799-2808, 2010.
[86] Dehuri S., Cho S.-B., Multi-Criterion Pareto Based Particle Swarm Optimized Polynomial Neural Network for Classification: A Review and State-of-the-Art. Computer Science Review, 3(1), 19-40, 2009.
[87] Deb K., Evolutionary Multi-criteria Optimization. In Proceedings of GECCO (Companion), 1155-1162, 2010.
[88] Coello C.A.C., Evolutionary Multi-objective Optimization. Wiley Interdisc. Rew.: Data Mining and Knowledge Discovery, 1(5), 444-447, 2011.
[89] Coello C.A.C., Dehuri S., Ghosh S., Swarm Intelligence for Multi-objective Problems in Data Mining (Eds.). Springer-Verlag, Berlin, Heidelberg, 2009.
[90] Gonzalez J., Rojas I., Pomares H., Ortega J., RBF Neural Networks, Multi-objective Optimization and Time Series Forecasting Connectionist Models of Neurons. Learning Processes, and Artificial Intelligence, 2084, 498-505, 2001.
[91] Hatanaka T., Kondo N., Uosaki K., Multi-objective Structure Selection for Radial Basis Function Networks Based on Genetic Algorithm. In Proceedings of Evolutionary Computation (CEC'03), 2, 1095-1100, 2003.
[92] Ferreira P.M., Ruano A.E., Fonseca C.M., Evolutionary Multi-objective Design of Radial Basis Function Networks for Green-

house Environmental Control. In Proceedings of the 16 the IFAC World Congress, 16(1), 870-870, 2005.

[93] Yen G.G., Multi-Objective Evolutionary Algorithm for Radial Basis Function Neural Network Design. Studies in Computational Intelligence (SCI), Springer-Verlag, 16, 221-239, 2009.

[94] Guillen A., Rojas I., Gonzalez J., Pomares H., Herrera L.J., Paechter B., Improving the Performance of Multi-objective Genetic Algorithm for Function Aoximation through Parallel Islands Specialization. In Proceedings of AI 2006, 1127-1132, 2006.

[95] Kondo N., Hatanaka T., Uosaki K., Pattern Classification by Evolutionary RBF Networks Ensemble Based on Multi-Objective Optimization. In Proceedings of International Joint Conference on Neural Networks, 2919-2925, 2006.

[96] Kondo N., Hatanaka T., Uosaki K., Nonlinear Dynamic System Identification Based on Multi objectively Selected RBF Networks. In Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multi criteria Decision Making, 112-127, 2007.

[97] Kokshenev I., Braga A.P., A Multi-objective Aoach to RBF Network Learning. Neurocomputing, 71, 1203-1209, 2008.

[98] Qasem S.N., Shamsuddin S.M., Radial Basis Function Network Based on Time Variant Multi-objective Particle Swarm Optimization for Medical Diseases Diagnosis. Aied Soft Computing, 11(1), 1427-1438, 2011.

[99] Qasem S.N., Shamsuddin S.M., Zain A.M., Multi-objective Hybrid Evolutionary Algorithms for Radial Basis Function Neural Network Design. Knowledge-based System, 27, 475-497, 2012.

[100] Alcalá-Fdez J., Sánchez L., García S., del Jesus M.J., Ventura S., Garrell J.M., Otero J., Romero C., Bacardit J., Rivas V.M., Fernández J.C., Herrera F., KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems. Soft Computing, 13, 307-318, 2009.

[101] Li X., On Simultaneous Aoximations by Radial Basis Function Neural Networks. A. Math. Comput., 95(1), 75-89, 1998.

[102] Light W.A., Wayne H., Error Estimates for Aoximation by Radial Basis Functions. Aoximation Theory, Wavelets, and Aications, 454, 215-246, 1995.

[103] Light W., Wayne H., On Power Functions and Error Estimates for Radial Basis Function Interpolation. Journal of Aoximation Theory, 92(2), 245-266, 1998.

[104] Light W.A., Cheney E.W., Interpolation by Piecewise-Linear Radial Basis Functions II. Journal of Aoximation Theory, 64(1), 38-54, 1991.

[105] Light W.A., Cheney E.W., Interpolation by Periodic Radial Basis Functions. Journal of Mathematical Analysis and Aications, 168(1), 111-130, 1992.

[106] Bellil W., Amar C.B., Alimi A.M., Comparison between Beta Wavelets Neural Networks, RBF Neural Networks and Polynomial Aoximation for 1D, 2D Functions Aoximation. Trans. Eng. Comp. Tech., 13, 102-107, 2006.

[107] Fornberg B., Driscoll T.A., Wright G., Charles R., Observations on the Behavior of Radial Basis Function Aoximations near Boundaries. Computers and Mathematics with Aications, 43(3), 473-490, 2002.

[108] Lendasse A., Lee J., Bodt E., Wertz V., Verleysen M., Aoximation by Radial Basis Function Networks-Aication to Option Pricing. Connectionist Aoaches in Economics and Management Sciences, 201-212, 2003.

[109] Chen T., Chen H., Aoximation Capability to Functions of Several Variables, Nonlinear Functional and Operators by Radial Basis Function Neural Networks. IEEE Trans Neural Networks, 6(4), 904-910, 1995.

[110] Dyn N., Ron A., Radial Basis Function Aoximation: from Gridded Centers to Scattered Centers. Proc London Math Soc, 71, 76-108, 1995.

[111] Ron A., The L2-aoximation Orders of Principal Shift-Invariant Spaces Generated by a Radial Basis function. In: D. Braess, L.L. Schumaker (Eds.), Numerical Methods in Aoximation Theory, Verilag, Basel, 9, 245-268, 1992.

[112] Fasshauer G.E., Solving Differential Equations with Radial Basis Functions: Multilevel Methods and Smoothing. Adv. Comp. Math., 11, 139-159, 1999c.

[113] Li X., Micchelli C.A., Aoximation by Radial Bases and Neural Networks. Numerical Algorithms, 25(1), 241-262, 2000.

[114] Golberg M.A., Chen C.S., Bowman H., Power H., Some Comments on the Use of Radial Basis Functions in the Dual Reciprocity Method. Computational Mechanics, 21(2), 141-148, 1998.

[115] Golberg M.A, Chen C.S., Bowman H., Some Recent Results and Proposals for the use of Radial Basis Functions in the BEM. Engineering Analysis with Boundary Elements, 23(4), 285-296, 1999.

[116] Golberg M.A., Chen C.S., The Theory of Radial Basis Functions Aied to the BEM for Inhomogeneous Partial Differential Equations. Boundary Elements Comm., 5, 57-61, 1994.

[117] Karur S.R., Ramachandran P.A., Radial Basis Function Aoximation in the Dual Reciprocity Method. Math. Comput. Model, 20, 59-70, 1994.

[118] Hua Z., Wei Z., Yaowu C., A No-reference Perceptual Blur Metric by Using OLS-RBF Network. In Proceedings of Pacific-Asia Workshop on Computational Intelligence and Industrial Aication (PACIIA'08), 1, 1007-1011, 2008.

[119] Iske A., Reconstruction of Smooth Signals from Irregular Samples by using Radial Basis Function Aoximation. In Proceeding of International Workshop on Sampling Theory and Aications, 82-87, 1999.

[120] Kluk P., Misiurski G., Morawski R.Z., Total Least Squares Vs. RBF Neural Networks in Static Calibration of Transducers. In Proceedings of Instrumentation and Measurement Technology Conference (IMTC/97), 424-427, 1997.

[121] Schaback R. (1993). Comparison of Radial Basis Function Interpolants in Multivariate Aoximation: From CAGD to Wavelets. In: K. Jetter, F. Utreras (Eds.), World Scientific Publishing, Singapore, 1993, 293-305.

[122] Zhou G., Wang C., Su W., Nonlinear Output Regulation Based on RBF Neural Network Aoximation. In Proceedings of International Conference on Control and Automation (ICCA'05), 2, 679-684, 2005.

[123] Faul A.C., Powell M.J.D., Krylov Subspace Methods for Radial Basis Function Interpolation. Numerical Analysis, Chapman & Hall/CRC, Boca Raton, FL, 200, 115-141.

[124] Foley T.A., Dayanand S., Zeckzer D., Localized Radial Basis Methods using Rational Triangle Patches in Geometric Modeling. Computing Suement, 10, 163-176, 1995.

[125] Lai S.J., Wang B.Z., Duan Y., Eigen Value Analysis of Spherical Resonant Cavity Using Radial Basis Functions, Progress in Electromagnetic Research Letters, 24, 69-76, 2011.

[126] Hon Y.C., Schaback R., On UnSymmetric Collocation by Radial Basis Functions. A Math Comput., 119, 177-186, 2001.

[127] Wu Z., On the Convergence of the Interpolation with Radial Basis Function. Chinese Ann. Math. Ser. A, 14, 480-486, 1993.
[128] Jackson I.R.H., Convergence Properties of Radial Basis Functions. Constr Aox, 4, 243-264, 1988.
[129] Wendland H., Numerical Solution of Variational Problems by Radial Basis Functions. Aoximation Theory IX, 2, 361-368, 1999.
[130] Wendland H., Meshless Galerkin Methods using Radial Basis Functions. Math. Comp., 68, 1521-1531, 1999.
[131] Beatson R.K., Light W.A., Fast Evaluation of Radial Basis Functions: Methods for two-dimensional polyharmonic splines. IMA Journal Numerical Analysia, 17(3), 343-372, 1997.
[132] Beatson R.K, Light W.A., Billings S., Fast Solution of the Radial Basis Function Interpolation Equations: Domain Decomposition Methods. SIAM J. Sci. Comput, 22(5), 1717-1740, 2000.
[133] Oliveira A., Medeiros E., Rocha T., Bezerra M., Veras R., A Study on the Influence of Parameter on Performance of RBF Neural Networks Trained with the Dynamic Decay Adjustment Algorithm. In Proceedings of Fifth International Conference on Hybrid Intelligent Systems (HIS'05), 512-514, 2005.
[134] Ri S., An Algorithm for Selecting a Good Value for the Parameter c in Radial Basis Function Interpolation. Adv. Comput. Math., 11, 193-210, 1999.
[135] Schwenker F., Dietrich C., Initialization of Radial Basis Function networks by Means of Classification Trees. Neural Network World, 10 473-482, 2000.
[136] Kubat M., Decision Trees can Initialize Radial Basis Function Networks. IEEE Transactions on Neural Networks, 9, 813-821, 1998.
[137] Karri V., Frost F., Effect of Altering the Gaussian Function Receptive Field Width in RBF Neural Networks on Aluminum Fluoride Prediction in Industrial Reduction Cells. In Proceedings of 6th International Conference on Neural Information Processing (ICONIP'99), 1, 101-106, 1999.
[138] Musavi M.T., Ahmed W., Chan K.H., Faris K.B., Hummels D.M., On the Training of Radial Basis Function Classifiers. Neural Networks, 5(4), 595-603, 1992.
[139] Oyang Y., Hwang S., Ou Y., Chen C., Chen Z., Data Classification with Radial Basis Function Networks Based on a novel Kernel Density Estimation Algorithm. IEEE Transactions on Neural Networks, 16(1), 225-236, 2005.
[140] Bouamar M., Ladjal M., A Comparative Study of RBF Neural Network and SVM Classification Techniques Performed on Real Data for Drinking Water Quality. Systems Signals and Devices. In Proceedings of 5th IEEE International Multi-Conference on SSD, 1-5, 2008.
[141] Guezouri M., Mesbahi L., A New Aoach Using Temporal Radial Basis Function in Chronological Series. Int. Arab J. Inf. Technol., 6(1), 85-90, 2009.
[142] Wang D. *et al.*, Extraction and Optimization of Fuzzy Protein Sequences Classification Rules using GRBF Neural Networks. Neural Information Processing-Letters and Reviews, 1(1), 53-57, 2003.
[143] Bruzzone L., Prieto D.F., A Technique for the Selection of Kernel-Function Parameters in RBF Neural Networks for Classification of Remote-Sensing Images. IEEE Transactions on Geoscience and Remote Sensing, 37(2), 1179-1184, 1999.
[144] Krzyzak A., Linder T., Lugosi C., Nonparametric Estimation and Classification using Radial Basis Function Nets and Empirical Risk Minimization. IEEE Trans. Neural Networks, 7(2), 475-487, 1996.
[145] Chang C.Y., Fu S.Y., Image Classification using a Module RBF Neural Network. In Proceedings of the 1st International Conference on Innovation Computing, Information and Control (ICICIC'06), 2, 270-273, 2006.
[146] Daqi G., Genxing Y., Adaptive RBF Neural Networks for Pattern Classifications. In Proceedings of Int. Joint Con. on Neural Network (IJCNN 2002), 1, 846-85, 2002.
[147] Dybowski R., Classification of Incomplete Feature Vectors by Radial Basis Function Networks. Pattern Recognition Letters, 19(14), 1257-1264, 1998.
[148] El Zooghby A., Christodoulou C., Georgiopoulos M., Performance of Radial Basis Function Networks for Direction of Arrival Estimation with Antenna Arrays. IEEE Trans. Antennas and Propagation, 45(11), 1611-1617, 1997.
[149] Peng J., Li K., Huang D., A Hybrid Forward Algorithm for RBF Neural Network Construction. IEEE Transactions on Neural Networks, 17(6), 1439-1451, 2006.
[150] Suresh S., Babu R.V., Kim H.J., No-reference image quality assessment using modified extreme learning machine classifier. A. Soft Comput., 9(2), 541–552, 2009.
[151] Fu X., Wang L., A GA-Based RBF Classifier with Class-Dependent Features.In the Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC'02, 2, 18890-1894, 2002.
[152] Fu X., Wang L., Chua K., Chu F., Training RBF Neural Networks on Unbalanced Data. In Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02), 1016-1020, 2002.
[153] Peng H., Ozaki T., Haggan-Ozaki V., Toyoda Y., A Parameter Optimization Method for Radial Basis Function Type Models. IEEE Transactions on Neural Networks, 14(2), 432-438, 2003.
[154] Dash C.S.K., Behera A.K., Dehuri S., Cho Sung-Bae, Differential Evolution-Based Optimization of Kernel Parameters in Radial Basis Function Networks for Classification. Int. Journal of Aied Evolutionary Computation (IJAEC), 4(1), 56-80, 2013.
[155] Dash C.S.K., Behera A.K., Pandia M.K., Dehuri S., Neural Networks Training Based on Differential Evolution in Radial Basis Function Networks for Classification of Web Logs. In Proceedings of International Conference on Distributed Computing and Internet Technology, 183-194, 2013.
[156] Buhmann M.D., Pre-Wavelet on Scattered Knots and from Radial Function Spaces: A Review. In: Martin R.R. (Ed.), 6th IMA Conference on the Mathematics of Surfaces, Clarendon Press, Oxford, 1995, 309-324.
[157] Dash C.S.K., Dash A.P., Dehuri S., Cho S.-B., Wang G.-N., DE+RBFNs Based Classification: A Special Attention to Removal of Inconsistency and Irrelevant Features. Engineering Aications of AI, 26(10), 2315-2326, 2013.
[158] Nagabhushan T.N., Ko H., Park J., Padma S.K., Nijagunarya Y.S., Classification of Symbolic Objects Using Adaptive Auto-Configuring RBF Neural Networks. In Proceedings of International Symposium on Information Technology Convergence (ISITC 2007), 22-26, 2007.
[159] Pulido A. *et al.*, Radial Basis Functions aied to the classification of UV-visible spectra1. Analytica Chimica Acta, 388(3), 273-281, 1999.
[160] Qin Z., Chen J., Liu Y., Lu J., Evolving RBF Neural Networks for Pattern Classification. Computational Intelligence and Security, LNAI 3081, 957-964, 2005.
[161] Baragada S.R., Ramakrishna S., Rao M.S., Purushothaman S., Polynomial Discriminant Radial Basis Function for Steganalysis.

IJCSNS, 9(2), 209-218, 2009.

[162] Kurban T., Besdok E., A Comparison of RBF Neural Network Training Algorithms for Inertial Sensor Based Terrain Classification. Sensors, 9, 6312-6329, 2009.

[163] Pan X., Lu Y., Cao Y., Zhang H., Zhao Y., Xu X., Research on the Algorithm of Noisy-Robust Tibetan Speech Recognition Based on RBF. In Proceedings of International Symposium on Intelligent Information Technology Aication Workshops (IITAW'08), 416-419, 2008.

[164] Niranjan M., Fallside F., Neural Network and Radial Basis Functions in Classifying Static Speech Patterns. Computer Speech and Language, 4(3), 275-289, 1990.

[165] Shan B., License Plate Character Segmentation and Recognition Based on RBF Neural Network. In Proceedings of 2nd International Workshop on Education Technology and Computer Science (ETCS), 2, 86-89, 2010.

[166] Sitamahalakshmi T., Babu D., Jagadeesh M., Mouli K., Performance Comparison of Radial Basis Function Networks and Probabilistic Neural Networks for Telugu Character Recognition. Global Journal of Computer Science and Technology, 11(4), 9-15, 2011.

[167] Lee Y., Hand Written Digit Recognition Using K-Nearest Neighbor, Radial Basis Function and Back propagation Neural Networks. Neural Computation, 3, 440-449, 1991.

[168] Hwang Y.S., Bang S.Y., An Efficient Method to Construct a Radial Basis Function Classifier Neural Networks. Neural Networks, 10(8), 1495-1503, 1997.

[169] Baboo S.S., Subashini P., Krishnaveni M., Combining Self-Organizing Maps and Radial Basis Function Networks for Tamil Handwritten Character Recognition. ICGST International Journal on Graphics, Vision, and Image Processing, 9, 49-56, 2009.

[170] Li Q., Tufts D.W., Principal Feature Classification. IEEE Transactions Neural Networks, 8(1), 155-160, 1997.

[171] Subashini T.S., Ramalingam V., Palanivel S., Breast Mass Classification Based on Cytological Patterns using RBFNN and SVM. Expert Systems with Aications, 36, 5284-5290, 2008.

[172] Chu F., Wang L., Aying RBF Neural Networks to Cancer Classification Based on Gene Expressions. In Proceedings of Int. Joint Conference on Neural Networks (IJCNN'06), 1930-1934, 2006.

[173] Kumar M., Srinivas D.S., Unsupervised Image Classification by Radial Basis Function Neural Network. In Proceedings of 22nd Asian Conference on Remote Sensing, 5, 9, 2001.

[174] Thakur S., Sing J.K., Basu D.K., Nasipuri M., Kundu M., Face Recognition using Principal Component Analysis and RBF Neural Networks. In Proceedings of First International Conference on Emerging Trends in Engineering and Technology (ICETET'08), 695-700, 2008.

[175] Dhanalakshmi P., Palanivel S., Ramalingam V., Classification of Audio Signals Using SVM and RBFNN. Expert Systems with Aications, 36(3), 6069-6075, 2009.

[176] Rosenblum M., Yacoob Y., Davis L.S., Human Expression Recognition from Motion using a Radial Basis Function Network Architecture. IEEE Trans. Neural Networks, 7, 1121-1138, 1996.

[177] Arad N., Dyn N., Reisfeld D., Yeshurun Y., Image Warping by Radial Basis Functions: Aications to Facial Expressions. CVGIP: Graphical Models and Image Processing, 56(2), 161-172, 1994.

[178] Fanhui X., Lixin S., Zhan Y., A Container Handling Capacity Prediction Model Based on RBF Neural Networks and its Simulation. In Proceedings of IEEE International Conference on Control and Automation (ICCA'2007), 1098-1100, 2007.

[179] Leonard J.A., Kramer M.A., Ungar J.H., Using Radial Basis Functions to Aoximate a Function and its Error Bounds. IEEE Trans. on Neural Networks, 3(4), 624-627, 1992.

[180] Mai-Duy N., Tran-Cong T., Numerical Solution of Differential Equations Using Multiquadric Radial Basis Function Networks. Neural Networks, 14(2), 185-199, 2001.

[181] Ling L., Kansa E.J., A Least-Squares Preconditioner for Radial Basis Functions Collocation Methods. Advances in Computational Mathematics, 23(1-2), 31-54, 2005.

[182] Schaback R., Wendland H., Using Compactly Surted Radial Basis Functions to Solve Partial Differential Equations. Boundary Element Technology, 13, 311-324, 1999.

[183] Fasshauer G.E., On Smoothing for Multilevel Aoximation with Radial Basis Functions. Aoximation Theory IX, 2, 55-62, 1999a.

[184] Fasshauer G.E., On the Numerical Solution of Differential Equations with Radial Basis Functions. Boundary Element Technology, 13, 291-300, 1999b.

[185] Power H., Barraco V., A Comparison Analysis between Unsymmetric and Symmetric Radial Basis Function Collocation Methods for the Numerical Solution of Partial Differential Equations. Computers and Mathematics with Aications, 43(3-5), 551-583, 2002.

[186] Schaback R., Wendland H., Using Compactly Surted Radial Basis Functions to Solve Partial Differential Equations. Boundary Element Technology, 13, 311-324, 1999.

[187] Jianyu L., Siwei L., Yingjian Q., Yaping H., Numerical Solution of Elliptic Partial Differential Equation using Radial Basis Function Neural Networks. Neural Networks, 16(5), 729-734, 2003.

[188] Wetterschereck D., Dietterich T., Improving the Performance of Radial Basis Function Networks by Learning Center Locations. Advances in Neural Information Processing Systems, 4, 1133-1140, 1992.

[189] Chun-tao M., Kun W., Li-yong Z., A New Training Algorithm for RBF Neural Network Based on PSO and Simulation Study. In Proceedings of WRI World Congress on Computer Science Info. Engineering, 4, 641-645, 2009.

[190] Senapati M.R., Vijaya I., Dash P.K., Rule Extraction from Radial Basis Functional Neural Networks by using Particle Swarm Optimization. Journal of Computer Science, 3(8), 592-599, 2007.

[191] Billings S.A., Zheng G.L., Radial Basis Function Network Configuration using Genetic Algorithms. Neural Networks, 8(6), 877-890, 1995.

[192] Yuan J.L., Li X.Y., Zhong L., Optimized Grey RBF Prediction Model Based on Genetic Algorithm. In Proceedings of International Conference on Computer Science and Software Engineering, 1, 74-77, 2008.

[193] Yu B., He X., Training Radial Basis Function Networks with Differential Evolution. In Proceedings of World Academy of Science Engineering and Technology, 11, 157-160, 2006.

[194] Shen Y., Bu Y., Yuan M., Study on Weigh-in-Motion System Based on Chaos Immune Algorithm and RBF Network. In Proceedings of Pacific-Asia Workshop on Computational Intelligence and Industrial Aication (PACIIA'08), 2, 502-506, 2008.

[195] Yunna W., Zhaomin S., Aication of RBF Neural Network Based on Ant Colony Algorithm in Credit Risk Evaluation of Construction Enterprises. In Proceedings of International Conference on Risk Management & Engineering Management (ICRMEM'08), 653-658, 2008.

[196] Zhao Z.Q., Huang D.S., A Mended Hybrid Learning Algorithm for Radial Basis Function Neural Networks to Improve General-

ization Capability. Aied Mathematical Modeling, 31, 1271-1281, 2007.

[197] Chen S., Cowan C.F.N., Grant P.M., Orthogonal Least Squares Learning Algorithm Basis Function Networks. IEEE Trans. Neural Networks, 2(2), 302-309, 1991.

[198] Liu Y., Zheng Q., Shi Z., Chen J., Training Radial Basis Function Networks with Particle Swarms. Advances in Neural Networks–ISNN, 3174, 317-322, 2004.

[199] Fatemi M., Roopaei M., Shabaninia F., New Training Methods for RBF Neural Networks. In Proceedings of International Conference on Neural Networks and Brain (ICNN&B'05), 3, 1322-1327, 2005.

[200] Tan S., Hao J., Vandewalle J., Nonlinear Systems Identification using RBF Neural Networks. In Proceedings of International Joint Conference on Neural Networks (IJCNN'93), 2, 1833-1836, 1993.

[201] De Lacerda E., de Carvalho A., Ludermir T., Model Selection via Genetic Algorithms for RBF Networks. Journal of Intelligent and Fuzzy Systems, 13, 111-122, 2003.

[202] Kondo N., Hatanaka T., Uosaki K., Pattern Classification by Evolutionary RBF Networks Ensemble Based on Multi-Objective Optimization. In Proceedings of International Joint Conference on Neural Networks, Korea, 2919-2925, 2006.

[203] Kondo N., Hatanaka T., Uosaki K., Non-linear Dynamic System Identification based on Multi-objectively Selected RBF Networks. In Proceedings of the IEEE Symposium on Computational Intelligence in Multi-criteria Decision Making, 112–127, 2007.

[204] Platt J.C., A Resource Allocation Network for Function Interpolation. Neural Comput., 3(2), 213–225, 1991.

[205] Lampariello F., Sciandrone M., Efficient Training of RBF Neural Networks for Pattern Recognition. IEEE Transactions on Neural Networks, 12(5), 1235-1242, 2001.

[206] Barra T.V., Bezerra G.B., de Castro L.N., Von Zuben F.J., An Immunological Density-Preserving Aoach to the Synthesis of RBF Neural Networks for Classification. In Proceedings of International Joint Conference on Neural Networks (IJCNN'06), 929-935, 2006.

[207] Ahmad R., Jamaluddin H., Radial Basis Function (RBF) for Non-Linear Dynamic System Identification. Journal Teknologi, 36, 39-54, 2002.

[208] Onsivilai A., Saichoomdee S., Distance Transmission Line Protection Based on Radial Basis Function Network. World Academy of Science, Engineering and Technology, 60, 81-84, 2009.

[209] Bi G., Dong F., A Sequential Learning Algorithm for RBF Networks and its Aication to Ship Course-Changing Control. In Proceedings of Control and Decision Conference (CCDC), Chinese, 3779-3784, 2008.

[210] Bing H., Gang L., Cun G., Jiang G., Modulation Recognition of Communication Signal Based on Wavelet RBF Neural Network. In Proceedings of 2nd International Conference on Computer Engineering and Technology (ICCET), 2, V2-490, 2010.

[211] Blu T., Unser M., Wavelets, Fractals, and Radial Basis Functions. IEEE Transactions on Signal Processing, 50(3), 543-553, 2002.

[212] Cha I., Kassam S.A., Interference Cancellation using Radial Basis Function Networks. EURASIP, Signal Processing, 47(3), 247-268, 1995.

[213] Cid-Sueira J., Artes-Rodriguez A., Figueiras-Vidal A.R., Recurrent Radial Basis Function Networks for Optimal Symbol-by-Symbol Equalization. EURASIP Signal Processing, 40, 53-63, 1994.

[214] Dhanalakshmi P., Palanivel S., Ramalingam V., Classification of Audio Signals using SVM and RBFNN. Expert Systems with Aications, 36, 6069-6075, 2008.

[215] Cui H., Research of Optimizing Ignition Control System in Gaseous Fuel Engine Based on RBF Neural Network. In Proceedings of International Conference on Intelligent Computation Technology and Automation (ICICTA), 1, 399-403, 2008.

[216] Dai J., Liu X., Zhang S., Zhang H., Yu Y., Zheng X., Neuronal Spike Sorting Based on 2-stage RBF Networks. In Proceedings of 2nd Int. Con. on Future Generation Communication and Networking (FGCN 2008), 3, 47-50, 2008.

[217] Dash P.K., Mishra S., Panda G., A Radial Basis Function Neural Network Controller for UPFC. IEEE Trans Power Syst, 15, 1293-1299, 2000.

[218] Ding J., Wang L., Forecast of RBF Neural Networks to Weak Electrical Signals in Plant. In Proceedings of International Conference on Artificial Intelligence and Computational Intelligence, 1, 621-625, 2009.

[219] Ding S., Jia W., Su C., Chen J., Research of Neural Network Algorithm Based on FA and RBF. In Proceedings of 2nd International Conference on Computer Engineering and Technology (ICCET), 7, V7—228, 2010.

[220] Dongli Y., Jianguo Y., Qingbiao X., A Study of Fault Detection and System Reconfiguration for UAV Navigation System bon RBF Neural Network. In Proceedings of 7th World Congress on Intelligent Control and Automation (WCICA 2008), 55-58, 2008.

[221] Esmaeilbeigi M., Hosseini M., Mohyud-Din S., A New Aoach of the Radial Basis Functions Method for Telegraph Equations. Int J Physical Sciences, 6(6), 1517-1527, 2011.

[222] Fan G., Dai Y., Wang H., Gaussian Sum Particle Filtering Based on RBF Neural Networks. In Proceedings of 7th World Congress on Intelligent Control and Automation (WCICA'2008), 3071-3076, 2008.

[223] Fraile R., Rubio L., Cardona N., Aication of RBF Neural Networks to the Prediction of Propagation Loss over Irregular Terrain. In Proceedings of 52nd IEEE VTS-Fall VTC Conference, 2, 878-884, 2000.

[224] Girosi F., Some Extensions of Radial Basis Functions and their Aications in Artificial Intelligence. Comput. Math. A., 24, 61-80, 1992.

[225] Gao L., Wang R., Yang S., Chai Y., An Image Fusion Algorithm Based on RBF Neural Networks. In Proceedings of International Conference on Machine Learning and Cybernetics, 8, 5194-5199, 2005.

[226] Ghosh J., Nag A., Knowledge Enhancement and Reuse with Radial Basis Function Networks. In Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02), 2, 1322-1327, 2002.

[227] Guanglan Z., Abeyratne U.R., Saratchandran P., Comparing RBF and BP Neural Networks in Dipole Localization. In Proceedings of First Joint BMES/EMBS Conference, 2, 939, 1999.

[228] Guo J., Gong J., Xu J., Improved RBF Neural Network for Nonlinear Identification System. In Proceedings of the 2009 International Workshop on Information Security and Aication, 137-140, 2009.

[229] Chen S., Billings S.A., Grant P.M., Recursive Hybrid Algorithm for Nonlinear System Identification using Radial Basis Function Networks. Int. J. Control, 55(5), 1051-1070, 1992.

[230] Hao W., Li X., Zhang M., Aication of RBF Neural Network to Temperature Compensation of Gas Sensor. In Proceedings of 2008

International Conference of Computer Science and Software Engineering, 4, 839-842, 2008.

[231] Hon Y.C., Mao X.Z., A Radial Basis Function Method for Solving Options Pricing Model. Journal of Financial Engineering, 8, 31-49, 1999.

[232] Jayawardena A.W., Fernando D.A.K., Use of Radial Basis Function Type Artificial Neural Networks for Runoff Simulation. Computer-Aided Civil and Infrastructure Engineering, 13, 91-99, 1997.

[233] Jianwei Z., Yu Z., Yina Z., Longfei Z., Damage Diagnosis of Radial Gate Based on RBF Neural networks. In Proceedings of 2009 International Forum on Computer Science-Technology and Aications, 3, 399-402, 2009.

[234] McGarry K.J., Wermter S., MacIntyre J., Knowledge Extraction from Radial Basis Function Networks and Multilayer Perceptrons. In Proceedings of International Joint Conference on Neural Networks (IJCNN'99), 4, 2494-2497, 1999.

[235] KeKang L.L., Zhang S., Research and Aication of Compound Control Based on RBF Neural Network and PID. In Proceedings of 3rd International Conference on Intelligent System and Knowledge Engineering (ISKE 2008), 1, 848-850, 2008.

[236] Krzyzak A., Linder T., Radial Basis Function Networks and Complexity Regularization in Function Learning. IEEE Trans. Neural Networks, 9, 247-256, 1997.

[237] Lai S., Wang B., Duan Y., Aication of the RBF-based Meshless Method to Solve 2-D Time Domain Maxwell's Equations. In Proceedings of International Conference on Microwave and Millimeter Wave Technology (ICMMT 2008), 2, 749-751, 2008.

[238] Lin G.F., Chen L., A Non-Linear Rainfall-Runoff Model using Radial Basis Function Network", Journal of Hydrology, 289, 1-8, 2004.

[239] Liu H., Gong Z., Li M., Sliding Mode Control of ROV Based on RBF Neural Networks Adaptive Learning. In Proceedings of 3rd International Conference on Intelligent System and Knowledge Engineering (ISKE'08), 1, 590-594, 2008.

[240] Ma L., Xin K., Liu S., Using Radial Basis Function Neural Networks to Calibrate Water Quality Model. International Journal of Intelligent Systems and Technologies, 3(2), 90-98, 2008.

[241] Liu S., Yu Q., Lin W., Yang S.X., Tracking Control of Mobile Robots Based on Improved RBF Neural Networks. In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 1879-1884, 2006.

[242] El Shafie A., El-Shafie A., Almukhtar A., Taha M., El Mazoghi H., Shehata A., Radial Basis Function Neural Networks for Reliably Forecasting Rainfall. Journal of Water and Climate Change, 3(2), 125-138, 2012.

[243] Neruda R., Vidnerov P., Learning Errors by Radial Basis Function Neural Networks and Regularization Networks. International Journal of Grid and Distributed Computing, 1(2), 49-57, 2008.

[244] Yang X., Dai H., Sun Y., A Hybrid Modeling Method Based on Mechanism Analysis, Identification, and RBF Neural Networks. In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, 2, 1310-1315, 2008.

[245] Yee MS., Yeap B.L., Hanzo L., Radial Basis Function-Assisted Turbo Equalization. IEEE Transactions on Communications, 51, 664-675, 2003.

[246] Rodriguez N., Yaez E., Crawford B., Wavelet Based Autoregressive RBF Network for Sardines Catches Forecasting. In Proceedings of Third International Conference on Convergence and Hybrid Information Technology (ICCIT'08), 2, 808-811, 2008.

[247] Sadr A., Mohesenifar N., Okhovat R.S., Comparison of MLP and RBF neural networks for Prediction of ECG Signals. IJCNS Int J Comput Sci Network Security, 11(11), 124-128, 2011.

[248] Arnott R., (1993). An Adaptive Radial Basis Function Diversity Combiner. IEEE Electronics Letters, 29(12), 1092-1094, 1993.

[249] Chen S., Gibson G.J., Cowan C.F.N., Grant P.M., Reconstruction of Binary Signals using an Adaptive Radial-Basis-Function Equalizer. EURASIP Signal Processing, 22, 77-93, 1991.

[250] Baxter B.J.C., (2002). Preconditioned Conjugate Gradients, Radial Basis Functions, and Toeplitz Matrices. Computers & Math with Aications, 43(3), 305-318, 2002.

[251] Bezerra M.E.R., Oliveira A.L.I., Meira S.R., A Constructive RBF Neural Network for Estimating the Probability of Defects in Software Modules. In Proceedings of International Joint Conference on Neural Networks (IJCNN 2007), 2869-2874, 2007.

[252] Chakravarthy S.V., Ghosh J., Scale-Based Clustering using the Radial Basis Function Network. IEEE Trans. Neural Networks, 7(5), 1250-1261, 1996.

[253] Chen S., McLaughlin S., Mulgrew B., Complex-valued Radial Basis Function Network, Part I: Network Architecture and Learning Algorithms. Signal Processing, 35(1), 19-31, 1994.

[254] Gomes J., Barroso V., Using an RBF Network for Blind Equalization: Design and Performance Evaluation. IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP-97, 4, 3285-3288, 1997.

[255] Chang E.S., Chen S., Mulgrew B., Gradient radial basis function networks for nonlinear and nonstationary time series prediction. IEEE Trans. Neural Networks, 7(1), 190-194, 1996.

[256] Ferreira A., Joaquim M., da Cunha Roque C., Carrera E., Cinefra M., Polit O., Analysis of Sandwich Plates by Radial Basis Functions Collocation, According to Murakami's Zig-Zag Theory. Journal of Sandwich Structures and Materials, 14(5), 505-524, 2012.

[257] Kattan A., Galvan E., Evolving Radial Basis Function Networks via GP for Estimating Fitness Values using Surrogate Models. In Proceedings of IEEE Congress on Evolutionary Computation (CEC'2012), 1-7, 2012.

[258] Pottmann M., Peter Jorgl H., Radial Basis Function Networks for Internal Model Control. Aied Mathematics and Computation, 70(2-3), 283-298, 1995.

[259] Sadeghkhani I., Ketabi A., Feuillet R., Radial Basis Function Neural Network Aication to Power System Restoration Studies. Computational Intelligence and Neuroscience, 2012, 654895, 2012.

[260] Xu L., Kryzak A., Yuille A., On Radial Basis Function Nets and Kernel Regression Statistical Consistency, Convergence Rates, and Receptive Field Sizes. Neural Networks, 7, 609-628, 1994.

[261] Bos F.M., Van Oudheusden B.W., Bijl H., Radial Basis Function Based Mesh Deformation Aied to Simulation of Flow Around Flang Wings. Computers & Fluids, 79, 167-177, 2013.

[262] Dybowski R., Classification of Incomplete Feature Vectors by Radial Basis Function Networks. Pattern Recognition Letters, 19, 1257-1264, 1998.

[263] Nagabhushan T.N., Padma S.K., On the Generalization of Incremental Learning RBF Neural Networks Trained with Significant Patterns. In Proceedings of 6th International Conference on Information, Communications, & Signal Processing, 2, 1-5, 2007.

[264] Ding Y., Zhao J., Yuan Z., Zhang Y., Long C., Xiong L., Constrained Surface Recovery using RBF and its Efficiency Improve-

ments. Journal of Multimedia, 5(1), 55-66, 2010.

[265] Er M.J., Chen W., Wu S., High-speed Face Recognition Based on Discrete Cosine Transform and RBF Neural networks. IEEE Trans Neural Networks, 16, 679-691, 2005.

[266] Er M., Wu S., Lu J., Toh H., Face Recognition with Radial Basis Function (RBF) Neural Networks. IEEE Transactions on Neural Network, 33, 697-710, 2002.

[267] Gan Q., Saratchandran P., Sundararajan N., Subramanian K.R., A Complex Valued Radial Basis Function Network for Equalization of Fast Time Varying Channels. IEEE Transactions on Neural Networks, 10(4), 958-960, 1999.

[268] Kantsila A., Lehtokangas M., Saarinen J., On Radial Basis Function Network Equalization in the GSM System. In Proceedings of European Symposium on Artificial Neural Networks, Bruges, 179-184, 2003.

[269] Liu Y., Zhang X., Hua J., Fault Diagnosis Based on Radial Basis Function Neural Network in Particleboard Glue Mixing and Dosing System. In Proceedings of Control and Decision Conference (CCDC 2008), 774-778, 2008.

[270] Cid-Sueiro J., Figueiras-Vidal A.R., Recurrent Radial Basis Function Networks for Optimal Blind Equalisation. In Proceedings of the 1993 IEEE-SP Workshop on Neural Networks for Signal Processing, 562-571, 1993.