Information Sciences Letters
An International Journal

http://dx.doi.org/10.12785/isl/030205

Eigen Value and It′s Comparison with Gaussian RBF, Multi-Quadratic RBF and Inverse Multi-Quadratic RBF Methods

Abhisek Paul[1,*], *Paritosh Bhattacharya*[1] *and Santi Prasad Maity*[2]

[1]Department of Computer Science and Engineering, National Institute of Technology, Agartala, India
[2]Department of Information Technology, Bengal Engineering and Science University, Shibpur, India

Received: 11 Dec. 2013, Revised: 23 Mar. 2014, Accepted: 24 Mar. 2014
Published online: 1 May 2014

Abstract: For any system Eigen value is the characteristic value and It is the characteristic roots for an equation which modeled that system. In this paper we have introduce Radial Basis Function (RBF) to evaluate Eigen value approximation for any given system. We have made some approximation to get best possible output with the help of pseudo inverse technique in Radial Basis Functions. We have experimented and compared the Eigen values of matrices with respect to Gaussian RBF, Multi-Quadratic RBF and Inverse Multi-Quadratic RBF methods. In this paper we worked on several matrices and calculated their Eigen value by using different RBF methods.

Keywords: RBF, Eigen value, Gaussian RBF, Multi-Quadratic RBF, Inverse Multi-Quadratic RBF

1 Introduction

Eigen vectors and Eigen values are the characteristic roots, characteristic values and numbers for a given system of equations when the system is operated by a set of matrix equation. Eigen values are modeled by scalar quantities which are related to a square matrix. Eigen vectors are the vectors which are related to the same matrix. Therefore, a system can be characterized by the Eigen values and eigen vector. The Eigen values are used in different fields of , science and technology such as engineering system design, pattern recognition, mechanics, automatic machine, statistics, computational intelligence, geometry, economics, astronomy, image processing, computer, robotics, etc. [4,20,25].

T be a matrix whose dimension is m×m. λ is a number which is an Eigen value of matrix T when a nonzero vector v is exists such that.

$$Tv = \lambda v \quad (1)$$

Vector v is Eigen vector of matrix T while corresponding to λ. We can rewrite the above equation as,

$$(T - \lambda I)v = 0 \quad (2)$$

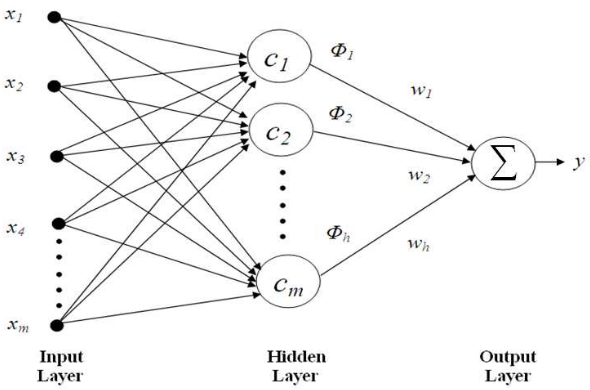

Fig. 1: RBF Neural Network.

Here I is an identity matrix whose dimension is m×m. For a nonzero vector v, $(T - \lambda I)$ cannot be invertible. So, the determinant of $(T - \lambda I)$ must be zero. Thus, $q(\lambda) = \det(T - \lambda I)$ is the characteristic polynomial of T. So, the Eigen value of T is the characteristic polynomial of T[4].

* Corresponding author e-mail: abhisekpaul13@gmail.com

The RBF neural network architecture is given here. It has three layers. First layer is input layer. The second layer is hidden layer which is different from the computational unit structure of the multilayer network. The computational units are known as radial centers which are being represented by $c_1, c_2, c_3.....c_h$ vectors. Dimension of each center for am input network is $m \times 1$. The centers have the same vector dimension as that of the input. The input space is the space where all possible data are located. The centers represent the clusters in the input space. The output of each center which is $\phi_i$is the function of the Euclidian distance between c_i and x. Output is obtain by proper choice of w_j,which is the weight of j^{th} center. The output is simply the summation of $\phi_i w_j$. Transformation from input space to hidden unit space in nonlinear and transformation from hidden unit space to output space is linear. Here $||x - c_j||$ is the Euclidian distance [1,2,7,8,9,12,13,21,22,25].

$$y = \sum_{h=1}^{n} \phi_h w_h \tag{3}$$

$$\phi = \phi(||x - c_h||) \tag{4}$$

The reminder of the paper is organized as follows. Section 2 reviews Radial Basis Function types and basics. Section 3 describes simulation, analysis,results and discussions. Section 4 gives conclusion.

2 Radial Basis Function

Radial Basis Function (RBF) gives non-zero response while input is in small localized area. Different Radial functions are available namely Gaussian RBF, Quadratic RBF and Inverse Quadratic RBF etc[14,15,16].

Function for Gaussian RBF is given below, where z is the Euclidian distance, σ is the maximum distance of center[3,24].

$$\phi(z) = e^{-z/2\sigma^2} \tag{5}$$

Function for Quadratic RBF is as follows, where r is any numerical number which is greater than zero and z is the Euclidian distance[17,19].

$$\phi(z) = (z^2 + r^2)^{1/2} \tag{6}$$

Function for Inverse Quadratic RBF is as follows, where r is any numerical number which is greater than zero and z is the Euclidian distance[6].

$$\phi(z) = (z^2 + r^2)^{-1/2} \tag{7}$$

2.1 RBF network learning process

Training of Radial Basis Function Network (RBFN) requires optimal selection of the parameter vectors c_i and w_i, $i = 1...h$. Both layers are optimized using different techniques and in different time scales[5,10,11,18].

Following techniques are used to update the weights and centers of a RBFN.

1.Pseudo-Inverse Technique.
2.Gradient Descent Learning.
3.Hybrid Learning.

2.1.1 Pseudo-Inverse Technique

This is a least square problem. Assume a fixed radial basis functions e.g. Gaussian functions. The centers are chosen randomly. The function is normalized i.e. for any x,$\sum \phi_i = 1$. The standard deviation (width) of the radial function is determined by an adhoc choice[5].

2.1.2 Gradient Descent Learning

One of the most popular approaches to update c and w, is supervised training by error correcting term which is achieved by a gradient descent technique[18].

2.1.3 Hybrid Learning

In hybrid learning, the radial basis functions relocate their centers in a self-organized manner while the weights are updated using supervised learning. When a pattern is presented to RBFN, either a new center is grown if the pattern is sufficiently novel or the parameters in both layers are updated using gradient descent [11].

$$\mathbf{A3} = \begin{bmatrix} 2 & 1 & 3 \\ 1 & 2 & 1 \\ 3 & 1 & 2 \end{bmatrix} \tag{8}$$

$$\mathbf{A4} = \begin{bmatrix} 2 & 2 & 3 & 3 \\ 4 & 2 & 2 & 2 \\ 4 & 2 & 2 & 3 \\ 2 & 2 & 3 & 2 \end{bmatrix} \tag{9}$$

$$\mathbf{A5} = \begin{bmatrix} 2 & 2 & 3 & 3 & 2 \\ 4 & 2 & 2 & 2 & 3 \\ 2 & 2 & 2 & 3 & 2 \\ 4 & 2 & 3 & 2 & 3 \\ 2 & 3 & 2 & 2 & 2 \end{bmatrix} \tag{10}$$

$$\mathbf{A6} = \begin{bmatrix} 3 & 2 & 1 & 1 & 0 & 0 \\ 2 & 3 & 2 & 3 & 2 & 2 \\ 1 & 2 & 3 & 1 & 2 & 1 \\ 1 & 3 & 1 & 3 & 1 & 2 \\ 0 & 2 & 2 & 1 & 3 & 1 \\ 0 & 2 & 1 & 2 & 1 & 3 \end{bmatrix} \tag{11}$$

$$\mathbf{A7} = \begin{bmatrix} 2 & -1 & -1 & 0 & -1 & 1 & 0 \\ -1 & 2 & 0 & 0 & 1 & -1 & 0 \\ -1 & 0 & 2 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 2 & 0 & -1 & -1 \\ -1 & 1 & 0 & 0 & 2 & -1 & -1 \\ 1 & -1 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & -1 & -1 & 0 & 2 \end{bmatrix} \quad (12)$$

$$\mathbf{A8} = \begin{bmatrix} 2 & -1 & -1 & 0 & -1 & 1 & 0 & 0 \\ -1 & 2 & 0 & 0 & 1 & -1 & 0 & -1 \\ -1 & 0 & 2 & 1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 1 & 2 & 0 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 & 2 & -1 & -1 & 0 \\ 1 & -1 & 0 & -1 & -1 & 2 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 & 0 & 2 & 1 \\ 0 & -1 & -1 & -1 & 0 & 0 & 1 & 2 \end{bmatrix} \quad (13)$$

Table 1: Comparison of Eigen Values of Matrix A3 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
-1.000000000000000	-1.000000000000001	-0.980392156862739	-0.980392156862743
1.438447187191170	1.438447187191170	1.410242340383501	1.410242340383501
5.561552812808831	5.561552812808832	5.452502757655707	5.452502757655715

Table 2: Comparison of Error of Eigen Values of Matrix A3 for Different RBF Methods.

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000001	0.019607843137261	0.019607843137257
0.000000000000000	0.028204846807669	0.028204846807669
0.000000000000001	0.109050055153124	0.109050055153116

Table 3: Comparison of Eigen Values of Matrix A4 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
10.227535796937078	10.227535796937078	10.026995879350071	10.026995879350075
-1.790729366741630	-1.790729366741630	-1.755617026217285	-1.755617026217275
0.000000000000002	0.000000000000001	0.000000000000011	0.000000000000000
-0.436806430195437	-0.436806430195437	-0.428241598230823	-0.428241598230817

Table 4: Comparison of Error of Eigen Values of Matrix A4 for Different RBF Methods

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000000	0.200539917587006	0.200539917587003
0.000000000000000	0.035112340524345	0.035112340524355
0.000000000000001	0.000000000000010	0.000000000000002
0.000000000000000	0.008564831964614	0.008564831964620

Table 5: Comparison of Eigen Values of Matrix A5 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
12.183971475785187	12.183971475785189	11.945070074299124	11.945070074299194
0.369769129398551	0.369769129398549	0.362518754312240	0.362518754312308
0.000000000000001	0.000000000000000	-0.000000000000056	-0.000000000000003
-1.553740605183741	-1.553740605183741	-1.523275103121381	-1.523275103121312
-1.000000000000000	-1.000000000000004	-0.980392156862742	-0.980392156862749

Table 6: Comparison of Error of Eigen Values of Matrix A5 for Different RBF Methods

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000002	0.238901401486062	0.238901401485993
0.000000000000002	0.007250375086311	0.007250375086243
0.000000000000001	0.000000000000057	0.000000000000004
0.000000000000001	0.030465502062360	0.030465502062429
0.000000000000004	0.019607843137258	0.019607843137251

3 Results and discussions

Six matrices are being experimented. The matrices are A3,A4,A5,A6,A7 and A8. Dimension of matrices A3,A4,A5,A6,A7,A8 are 3×3, 4×4,5×5, 6×6,7×7 and 8×8 respectively. Matrices are input. We have compared the Eigen values of these matrices with three different RBF methods namely Gaussian, Quadratic and Inverse Quadratic respectively. Fig 2.shows the comparison of errors of Eigen values for matrix A3 with respect to Gaussian, Quadratic and Inverse Quadratic methods. Here error is less in Gaussian RBF method. figure 3.shows the comparison of errors of Eigen values for matrix A4 with respect to Gaussian, Quadratic and Inverse Quadratic methods. Here error is less in Gaussian RBF method. figure 4.shows comparison of errors of Eigen values for the matrix A5. Here, error in Quadratic and Inverse Quadratic RBF is quite close but Gaussian RBF shows better result. figure 5.shows the comparison of errors of Eigen values for matrix A6 with respect to Gaussian, Quadratic and Inverse Quadratic methods. Here error is less in Gaussian RBF method. figure 6.shows comparison of Eigen values of matrix A7. Here, Gaussian RBF gives best result with respect to the other two methods. figure 7.shows comparison of Eigen values of matrix A8. Here, Gaussian RBF gives best result with respect to the other two methods. Table 1, Table 3, Table 5, Table 7, Table 9 and Table 11 show the Eigen values. While Table 2, Table 4 and Table 6, Table 8, Table 10 and Table 12 give demonstrate of the relative errors.

Table 7: Comparison of Eigen Values of Matrix A6 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
-0.516457847340464	-0.516457847340464	-0.506331222882816	-0.506331222882810
0.563873934620447	0.563873934620443	0.552817582961246	0.552817582961224
1.100739385595450	1.100739385595446	1.079156260387692	1.079156260387693
3.023694707307639	3.023694707307634	2.964406575791775	2.964406575791801
3.346716161469828	3.346716161469827	3.281094275950823	3.281094275950810
10.481433658347100	10.481433658347100	10.275915351320661	10.275915351320688

Table 8: Comparison of Error of Eigen Values of Matrix A6 for Different RBF Methods.

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000000	0.010126624457648	0.010126624457654
0.000000000000004	0.011056351659201	0.011056351659223
0.000000000000004	0.021583125207757	0.021583125207756
0.000000000000005	0.059288131515864	0.059288131515838
0.000000000000001	0.065621885519005	0.065621885519018
0.000000000000000	0.205518307026439	0.205518307026413

Table 9: Comparison of Eigen Values of Matrix A7 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
0.000000000000000	0.000000000000000	-0.000000000000066	0.000000000000002
0.000000000000001	0.000000000000000	-0.000000000000035	-0.000000000000001
0.913869802348507	0.913869802348507	0.895950786616148	0.895950786616183
1.999999999999999	2.000000000000000	1.960784313725466	1.960784313725489
2.000000000000000	1.999999999999999	1.960784313725481	1.960784313725489
3.571993268316203	3.571993268316202	3.501954184623703	3.501954184623728
5.514136929335290	5.514136929335289	5.406016597387500	5.406016597387540

Table 10: Comparison of Error of Eigen Values of Matrix A7 for Different RBF Methods

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000000	0.000000000000067	0.000000000000001
0.000000000000001	0.000000000000035	0.000000000000002
0.000000000000000	0.017919015732358	0.017919015732324
0.000000000000001	0.039215686274533	0.039215686274510
0.000000000000001	0.039215686274520	0.039215686274511
0.000000000000001	0.070039083692500	0.070039083692475
0.000000000000001	0.108120331947791	0.108120331947751

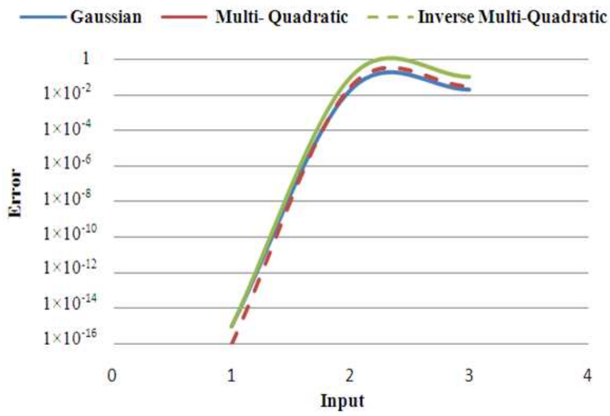

Fig. 2: Comparison of error of Eigen values of Matrix A3 for different RBF methods.

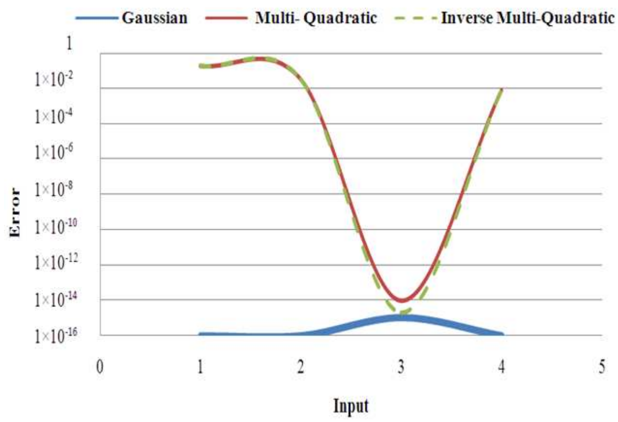

Fig. 3: Comparison of error of Eigen values of Matrix A4 for different RBF methods.

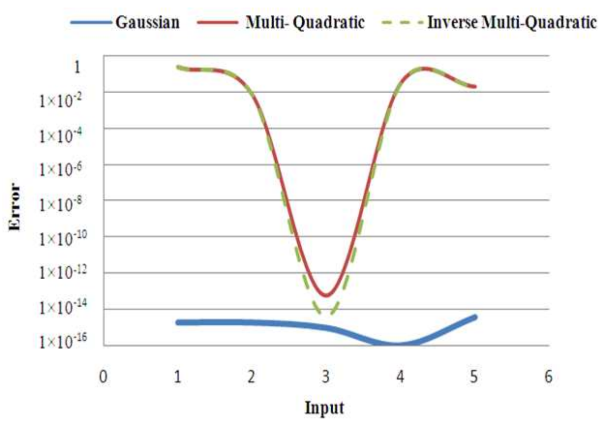

Fig. 4: Comparison of error of Eigen values of Matrix A5 for different RBF methods.

Table 11: Comparison of Eigen Values of Matrix A8 Using Normal Method and Different RBF Methods.

Theoritical	Gaussian RBF	Multi-Quadratic RBF	Inverse Multi-Quadratic RBF
0.000000000000000	0.000000000000000	0.000000000000127	0.000000000000127
0.000000000000000	-0.000000000000001	0.000000000000081	0.000000000000082
0.000000000000000	0.000000000000000	0.000000000000075	0.000000000000075
2.000000000000000	2.000000000000001	1.960784313725525	1.960784313725525
2.000000000000001	2.000000000000001	1.960784313725535	1.960784313725535
2.000000000000001	2.000000000000002	1.960784313725546	1.960784313725546
4.000000000000001	4.000000000000003	3.921568627451014	3.921568627451014
6.000000000000002	6.000000000000001	5.882352941176585	5.882352941176585

Table 12: Comparison of Error of Eigen Values of Matrix A8 for Different RBF Methods

Error in Gaussian RBF	Error in Multi-Quadratic RBF	Error in Inverse Multi-Quadratic RBF
0.000000000000001	0.000000000000127	0.000000000000001
0.000000000000001	0.000000000000081	0.000000000000001
0.000000000000001	0.000000000000075	0.000000000000001
0.000000000000001	0.039215686274475	0.039215686274512
0.000000000000001	0.039215686274465	0.039215686274511
0.000000000000001	0.039215686274455	0.039215686274510
0.000000000000002	0.078431372548986	0.078431372549022
0.000000000000002	0.117647058823417	0.117647058823532

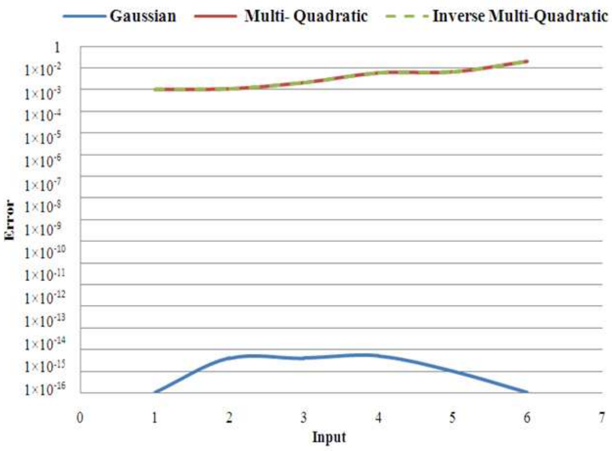

Fig. 5: Comparison of error of Eigen values of Matrix A6 for different RBF methods.

4 Conclusion

Gaussian RBF gives better result for the calculation of Eigen values of matrices. In every experiment Gaussian RBF shows better result corresponding to Quadratic and Inverse Quadratic RBF methods. The experiments are done using MATLAB 7.6.0 software tools. Choosing the proper weight in RBF neural network experimental results are satisfactory. Thus we can conclude that Gaussian RBF may be used for getting good results in the neural network.

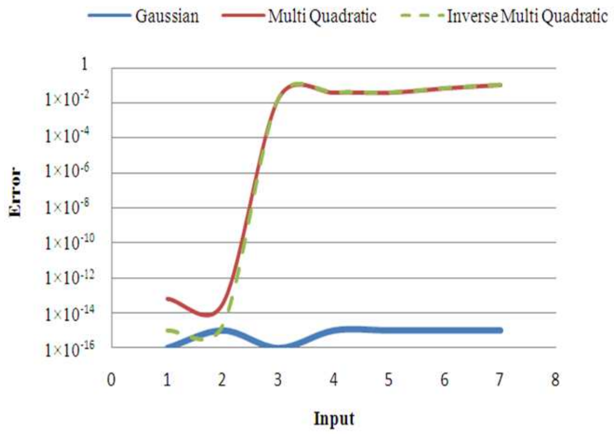

Fig. 6: Comparison of error of Eigen values of Matrix A7 for different RBF methods.

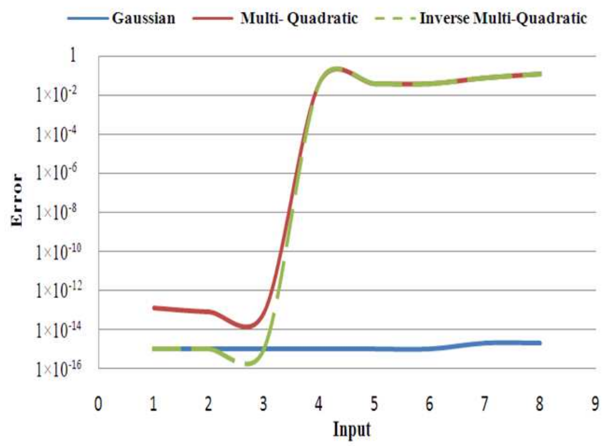

Fig. 7: Comparison of error of Eigen values of Matrix A8 for different RBF methods.

Acknowledgement

The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] K.Z. Mao, G.B. Huang, "Neuron selection for RBF neural network classifier based on data structure preserving criterion",IEEE Trans. Neural Networks vol. 16, no. 6, 1531-1540 (2005).

[2] D.S. Broomhead, D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks", Royal Signals and Radar Establishment, 8-14 (1988).

[3] B. Scholkopf, K.-K. Sung, C.J.C. Burges, F. Girosi, P. Niyogi, T. Poggio, V. Vapnik,”Comparing support vector machines with Gaussian kernels to radial basis function classifiers”,IEEE Trans. Signal Process vol. 45, 2758-2765 (1997).
[4] F.L. Luo and Y.D. Li,”Real-time computation of the eigenvector corresponding to the smallest eigen value of a positive definite matrix”,IEEE Trans. Circuits and Systems vol. 1, no. 41, 550-553 (1994).
[5] C.A. Klein, C.H. Huang,”Review of pseudo-inverse control for use with kinematically redundant manipulators”,IEEE Trans. Syst. Man Cybern vol. 13, no. 3, 245-250 (1983).
[6] Mehdi Dehghan, Ahmad Nikpour,”TNumerical solution of the system of second-order boundary value problems using the local radial basis functions based differential quadrature collocation method”, Applied Mathematical Modelling, vol. 37, 8578?8599 (2013).
[7] K.Z. Mao, ”RBF neural network center selection based on Fisher ratio class separability measure”, IEEE Trans. Neural Networks, vol. 13, no. 5, 1211?1217 (2002).
[8] G.B.Huang,P.Saratchandran,N.Sundararajan,”A genearlized growing and pruning RBF(GGAP-RBF)neural network for function approximation”,IEEETans.Neural Networks, vol. 16, no.1, 57?67 (2005).
[9] G.A.Montazer,R.Sabzevari,F.Ghorbani,”Three-phase strategy for the OSD learning method in RBF neural networks”,Neurocomputing vol.72, 1797-1802 (2009).
[10] Qu Yi, Li Zhan-ming, Li Er-chao,”Fault detection and diagnosis for non-Gaussian stochastic distribution systems with time delays via RBF neural networks”, ISA Transactions vol. 51, 786?791 (2012).
[11] J. Peng, K. Li, D. Huang,”A hybrid forward algorithm for RBF neural network construction”, IEEE Trans. Neural Networks, vol. 17 no. 6, 1439?1451 (2006).
[12] J.A. Rad, S. Kazem, K. Parand,”Optimal control of a parabolic distributed parameter system via radial basis functions”, Communications in Nonlinear Science and Numerical Simulation, vol. 19 no. 8, 2559-2567 (2014).
[13] Sung-Kwun Oh, Wook-Dong Kim, Witold Pedrycz, Kisung Seo,”Fuzzy Radial Basis Function Neural Networks with information granulation and its parallel genetic optimization”, Fuzzy Sets and Systems, vol. 237, 96-117 (2014).
[14] John P. Boyd”A Fourier error analysis for radial basis functions on an infinite uniform grid. Part 2: Spectral-plus is special”, Applied Mathematics and Computation, vol. 225, 695-707 (2014).
[15] Stefano De Marchi, Gabriele Santin,”A new stable basis for radial basis function interpolation”, Journal of Computational and Applied Mathematics, vol. 253,1-13 (2013).
[16] Carlos Valencia, Ming Yuan,”Radial basis function regularization for linear inverse problems with random noise”, Journal of Multivariate Analysis, vol. 116, 92-108 (2013).
[17] B. Fakhr Kazemi, F. Ghoreishi,”Error estimate in fractional differential equations using multiquadratic radial basis functions”, Journal of Computational and Applied Mathematics, vol. 245, 133-147 (2013).
[18] Xin Luo, Huijun Liu, Gaopeng Gou, Yunni Xia, Qingsheng Zhu”A parallel matrix factorization based recommender by alternating stochastic gradient decent”, Engineering Applications of Artificial Intelligence, vol. 25 no. 7, 1403-1412 (2012).
[19] John P. Boyd, Philip W. McCauley”Quartic Gaussian and Inverse-Quartic Gaussian radial basis functions: The importance of a nonnegative Fourier transform”,Computers & Mathematics with Applications, vol. 65 no. 1, 75-88 (2013).
[20] N.S. O’Brien, K. Djidjeli, S.J. Cox,”Solving an eigenvalue problem on a periodic domain using a radial basis function finite difference scheme”, Engineering Analysis with Boundary Elements, vol. 37 no. 12, 1594-1601 (2013).
[21] Chang-Wook Han,”Fuzzy Neural Network-based Time Delay Prediction for Networked Control Systems”, Applied Mathematics & Information Sciences, vol. 8 no. 1, 407-413(2014).
[22] Paris Mastorocostas, Constantinos Hilas, Dimitris Varsamis, Stergiani Dova,”A Recurrent Neural Network based Forecasting System for Telecommunications Call Volume”, Applied Mathematics & Information Sciences, vol. 7 no. 5, 1643-1650 (2013).
[23] Yu-Yi Fu, Chia-Ju Wu, Chia-Nan Ko, Jin-Tsong Jeng,”Radial basis function networks with hybrid learning for system identification with outliers ”, Applied Soft Computing, vol. 11 no. 3, 3083-3092 (2011).
[24] John P. Boyd, Zhengjie Xu,”Comparison of three spectral methods for the Benjamin?Ono equation: Fourier pseudospectral, rational Christov functions and Gaussian radial basis functions”, Wave Motion, vol. 48 no. 8, 702-706(2013).
[25] Alexandre Leblanc, Alain Malesys, Antoine Lavie,”Compactly supported radial basis functions for the acoustic 3D eigenanalysis using the particular integral method”, Engineering Analysis with Boundary Elements, vol. 36 no. 4, 520-526 (2012).

Abhisek Paul received the M.Tech. degree in Computer Science and Engineering at Tripura University of Agartala, India. He is the PhD Scholar in the Department of Computer Science and Engineering in National Institute of Technolohy, Agartala, India. His research interests are in the areas of wireless mesh network, mobile adhoc network, vehicular adhoc network, mathematical modeling, applied mathematics including the mathematical methods, numerical methods.

Paritosh Bhattacharya is presently working as Associate Professor in Mathematics and Head of the Department of Computer Science and Engineering at National Institute of Technology, Agartala, Tripura, India. His Twelve years of experience in higher education include his positions as a faculty member, reviewer, academic counselor, academic guide and member of academic societies. He obtained his Ph.D. (Engg) and M.Tech from Jadavpur University, West Bengal, India. He has published more than 50 papers in national and international journals and conferences. His present area of research includes Mathematical Modeling, Neural Network, Biomathematics and Computational Fluid Dynamics.

Santi Prasad Maity received the Post-Doctoral degree in Laboratoire des Signaux et Systems at France. He received the PhD degree in Engineering (Computer Science and Technology) at Bengal Engineering and Science University(BESU), Shibpur, India. His research interests are in the areas of digital image watermarking,signal processing, wavelet theory based analysis, soft-computing, access control, reversible watermarking, VLSI Architecture, cognitive radio, spectral sensing and compressive sampling, integration of cognitive radio with wireless network, wireless network, physical layer security of network, power control in information theoretic security,optical information processing , combinational and sequential circuits, codes generation for optical communication. He has published research articles in reputed international journals of engineering science. He is referee and editor of engineering journals.