Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2012, Article ID 286391, 13 pages
doi:10.1155/2012/286391

Research Article

Numerical Solution of Poisson's Equation Using a Combination of Logarithmic and Multiquadric Radial Basis Function Networks

Mohammad Mehdi Mazarei[1] and Azim Aminataei[2]

[1] *Faculty of Engineering, Islamic Azad University, Bushehr Branch, P.O. Box 7519619555, Bushehr, Iran*
[2] *Department of Mathematics, K. N. Toosi University of Technology, P.O. Box 15418-49611, Tehran, Iran*

Correspondence should be addressed to Azim Aminataei, ataei@kntu.ac.ir

Received 27 June 2011; Accepted 18 September 2011

Academic Editor: Roberto Natalini

This paper presents numerical solution of elliptic partial differential equations (Poisson's equation) using a combination of logarithmic and multiquadric radial basis function networks. This method uses a special combination between logarithmic and multiquadric radial basis functions with a parameter r. Further, the condition number which arises in the process is discussed, and a comparison is made between them with our earlier studies and previously known ones. It is shown that the system is stable.

1. Introduction

Many problems in applied sciences and engineering are reduced to a set of partial differential equations (PDEs). Analytical methods are frequently inadequate for obtaining solution, and usually numerical methods must be resorted. Radial basis function network is a well-known method to interpolate unknown functions and approximate numerical solutions. We have some radial basis functions such as "spline functions," "Gaussian functions," "multiquadric functions," and "logarithmic functions." All of researchers have tried to increase accuracy of approximate solutions while the stability of their suggested system is stable (the condition number is near to unity as possible). In multiquadric radial basis functions (MQ-RBFs), there are some parameters that influence accuracy of the solution, for instance, the width parameter of a basis function, scattered data points, and so on. In recent years, many researchers have worked on these parameters. They have tested many cases and have obtained different relations for such parameters. Kansa [1] has found that the best results are achieved by MQ

approximation scheme when the parameter r^2 (equivalent to the RBFs width parameter in this paper) is varied according to the following expansion:

$$r^2(j) = r^2_{\min}\left(\frac{r^2_{\max}}{r^2_{\min}}\right)^{(j-1)/(N-1)}, \tag{1.1}$$

where $r^2_{\min}$ and $r^2_{\max}$ are two input parameters; superscript j indexes the jth data point; N is the number of data points. However, in (1.1), Kansa [1] did not report how $r^2_{\min}$ and $r^2_{\max}$ should be chosen until later, when Moridis and Kansa [2] have stated that the ratio $r^2_{\max}/r^2_{\min}$ must be in the range of 10^1–10^9. Sharan et al. [3] have suggested some relations for these parameters to obtain more accurate solutions and having stable systems. Kansa and Hon [4] have used MQ-RBFs for ill-conditioning problems. They showed that, in using finite element method (FEM) to simulate large scale complex PDE problems, the resulting coefficient matrices even with finite band widths can still be ill conditioned, though by using MQ-RBFs, we can solve these problems. Mai-Duy and Tran-Cong [5] have developed the new methods based on the MQ radial basis function networks (RBFNs) for the approximation of both functions and their first and higher derivatives. The so-called direct RBFN (DRBFN) and indirect RBFN (IRBFN) methods were studied, and it was found that the IRBFN method yields consistently better results for both functions and their derivatives. In the IRBFN method, they have set the unknown function as the following relation:

$$u_{jj}(X) = \sum_{i=1}^{N} w_i g_i(X) = \sum_{i=1}^{N} w_i \sqrt{r_i{}^2 + a_i^2}, \tag{1.2}$$

wherein g_i is a MQ-RBF and $a_i^2 = (x - x_i)^2 + (y - y_i)^2$ is in a two-dimensional PDE. They have let the values of r_i (named as width parameter of the ith neuron (center) of MQ-RBF) as the following relation:

$$r_i = \beta d_i, \tag{1.3}$$

where d_i's are distances between the points (x_i, y_i) and the nearest point. They introduced a range for parameter β that in that range they have more accurate approximated solution and have a stable system. Fedoseyev et al. [6] improved MQ approximation scheme for elliptic partial differential equations via PDE collocation. They formulated an improved Kansa-MQ approximation scheme with the PDE collocation on the boundary. The idea of the method is to add an additional set of nodes adjacent to the boundary and, accordingly, an additional set of collocation equations obtained via collocation of the PDE on the boundary. Galperin and Kansa [7] have used MQ-RBFs with global optimization to numerical solutions of weakly singular Volterra integral equations. Mai-Duy and Tran-Cong [8] improved their results for approximation of unknown function and its derivatives. Also, Mai-Cao [9] solved transient PDEs using IRBFN method and showed that it gives better accuracy as before, but, in parabolic PDEs, this method and its ranges for opting the values of β do not work as well as elliptic PDEs, and, in some cases, they have inaccurate solution and ill-conditioned system. Buhmann [10] has discussed MQ-RBFs in solving n-dimensional PDEs and has introduced his results. Ling and Kansa [11] have experimented with different implementations of

the coupling of approximate cardinal basis functions preconditioning technique. They used MQ-RBFs with domain decomposition method. They showed that it can be used MQ-RBFs in solving problems of higher dimensions as the six-dimensional Boltzmann equation or the molecular Schrodinger equation to have better results than traditional methods such as finite difference method and FEM. Later, Aminataei and Mazarei [12] have studied on the width parameter of MQ-RBFs for one- and two-dimensional PDEs and have introduced a new range for the width parameter. Aminataei and Sharan [13] have used MQ approximation scheme on the numerical solution of ODEs with a singularity point and PDEs in one and two dimensions incorporating the domain decomposition method. Brown et al. [14] have applied the MQ-RBFs on approximate cardinal preconditioning methods for solving PDEs. Munoz-Gomez et al. [15] have proposed an overlapping domain decomposition method with RBFs for transient PDEs. Mazarei and Aminataei [16] have worked on the parameter β in DRBFN and IRBFN methods and have made a comparison between the RBFN methods (DRBFN and IRBFN methods) with Adomian double decomposition method and showed that these RBFNs yield better accuracy than Adomian double decomposition method. Recently, Aminataei and Mazarei [17] have used the DRBFN and IRBFN methods on the polar coordinate and have achieved better accuracy. In addition, MQ-RBFs collocation method has been applied with great success to obtain approximate solution for a large variety of problems such as in delay differential equations [18], differential algebraic equations [19], and integral equations [20]. In the present paper, we have combined logarithmic and MQ-RBFs with a parameter r and have improved the accuracy of our earlier works, but more important thing is about the condition number of the system that becomes equal to unity. Hence, the stability of the system in this new way is assured.

The organization of the present paper is as follows. In Section 2, we give the theories of the new method. In Section 3, we provide some numerical experiments on the two-dimensional Poisson's equation with the Dirichlet, the Neumann, and curved boundary conditions and we propose to examine the stability of the method and its behavior towards input variations. Finally, in Section 4, some conclusions are presented.

2. The New Method of the Present Study

The form of a Poisson's equation is as follows:

$$\nabla^2 u = f(X), \tag{2.1}$$

in which $f(X)$ is a known function. Also, Poisson's equation can be in the following two Dirichlet and Neumann boundary conditions:

$$u = f_1(X), \quad \text{on } \Gamma_1, \tag{2.2}$$

$$\overrightarrow{n} \cdot \overrightarrow{\nabla u} = f_2(X), \quad \text{on } \Gamma_2, \tag{2.3}$$

where Γ_1 and Γ_2 are boundaries of the domain. Also, the vector $\vec{n}$ is the outer unit normal to the boundaries and f_1, f_2 are two known functions of X. In MQ approximation scheme, we approximate the unknown function $u(X)$ by an expression. In this study, we consider two-dimensional Poisson's equation.

In this method, we use the following expression:

$$u(x,y) = \sum_{i=1}^{N} w_i \ln\left(\sqrt{(x-x_i)^2 + (y-y_i)^2 + r^2}\right). \tag{2.4}$$

The derivatives in the two-dimensional Cartesian coordinates are presented in the following:

$$\begin{aligned}
u_x &= \sum_{i=1}^{N} w_i \frac{x-x_i}{(x-x_i)^2 + (y-y_i)^2 + r^2},\\
u_{xx} &= \sum_{i=1}^{N} w_i \frac{(y-y_i)^2 - (x-x_i)^2 + r^2}{\left((x-x_i)^2 + (y-y_i)^2 + r^2\right)^2},\\
u_y &= \sum_{i=1}^{N} w_i \frac{y-y_i}{(x-x_i)^2 + (y-y_i)^2 + r^2},\\
u_{yy} &= \sum_{i=1}^{N} w_i \frac{(x-x_i)^2 - (y-y_i)^2 + r^2}{\left((x-x_i)^2 + (y-y_i)^2 + r^2\right)^2}.
\end{aligned} \tag{2.5}$$

In expression (2.4), the set of weights $\{w_i\}_{i=1}^{N}$ is to be found. In the present study, the closed form of approximation function (2.4) is first obtained from a set of training points and the derivative functions are then calculated directly by differentiation of such closed activity. The nonzero parameter r protects of having zero values inside the logarithmic function. Also, we decrease the parameter r to improve approximate solutions. In fact, this parameter controls the accuracy and the stability of the system. In each experiments, we evaluate the condition number of the system (condition number = $\|A\| \cdot \|A^{-1}\|$, that A is the coefficient matrix of the system) and try to inspect the affect of the parameter r on the stability of our system.

3. Illustrative Numerical Experiments

In this section, we present three experiments, wherein their numerical solutions illustrate some advantages of the new method with high accuracy and show that, in this new way, the system is not ill conditioned.

3.1. Stability of the Solution

A method is said to be stable when the obtained solution undergoes small variations as there are slight variations in inputs and parameters and when probable perturbations in parameters that are effective in equations and conditions prevailing them do not introduce, in comparison to the physical reality of the problem, any perturbations in what is returned. We propose here to compare the new method with other numerical methods (i.e., DRBFN and IRBFN methods) by offering experiments and examining the stability of the new method (Tables 1 and 3).

Experiment 1. Consider the following two-dimensional Poisson's equation:

$$\nabla^2 u = xe^y, \tag{3.1}$$

with the following Dirichlet boundary conditions on $0 \le x \le 2$ and $0 \le y \le 1$:

$$\begin{aligned} u(x,0) &= x, \qquad u(x,1) = xe, \\ u(0,y) &= 0, \qquad u(2,y) = 2e^y. \end{aligned} \tag{3.2}$$

The exact solution is: $u_e(x,y) = xe^y$.

We denote the root-mean-square error by the RMSE from the following relation:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(u_e^k - u^k\right)^2}, \tag{3.3}$$

where u_e^k is exact solution and u^k is approximate solution at points (x_k, y_k). We have considered those 20 points that we had used for IRBFN method on the polar coordinates [17]. As we have shown in earlier work [17], the parameter r influences on accuracy, partially. We have shown that increasing the values of this parameter causes instability, and usually can not affect on accuracy adequately (see Table 4). Although, in that method, we could improve our results by focusing on other parameters such as substituting the scattered point places, in this new way, we can improve our solutions by changing the values of parameter r without having instability. In contrast of MQ-RBFs, in this special combination, as parameter r decreases, the accuracy increases. Also, since, for values greater than 10^{-2}, the condition number is not near to unity and we will not have a stable system sufficiently, so we have considered at least $r = 0.01$. For instance, for this value of r, we have RMSE $= 1.99 \times 10^{-15}$, and, when we continue to decrease this parameter over and over, we get a better solution. For instance, in the best position RMSE $= 2.26 \times 10^{-16}$ while in IRBFN method on the polar coordinates [17] in the best position, we had RMSE $= 4.36 \times 10^{-11}$. Also, in this new way, when r reaches to about 10^{-9} and smaller values, our accuracy and condition number are almost fixed. As we are decreasing parameter r, the accuracy is going to be better (see Table 2). Further, when we decrease r values quite enough, the accuracy and condition number almost do not change (see Table 2). Note that, for smaller values of r, condition number is very near to unity (equals unity in double precision).

Experiment 2. Consider the following two-dimensional Poisson's equation:

$$\nabla^2 u = \left(\lambda^2 + \mu^2\right)e^{\lambda x + \mu y}, \tag{3.4}$$

with the following Neumann and Dirichlet boundary conditions on $0 \le x \le 1$ and $0 \le y \le 1$:

$$\begin{aligned} u_x &= \lambda e^{\lambda x + \mu y} \qquad \text{at } x = 0,\ x = 1, \\ u &= e^{\lambda x + \mu y} \qquad \text{at } y = 0,\ y = 1. \end{aligned} \tag{3.5}$$

Table 1: Comparison between exact solution and approximate solution of the new method of Experiment 1.

x_i	y_i	Exact solution	Approximate solution of the new method	Error of the new method
0.3333	0.2	0.4070935392947847	0.4070935392947850	3×10^{-16}
0.3333	0.4	0.4972251717238353	0.4972251717238351	2×10^{-16}
0.3333	0.6	0.6073121961701566	0.6073121961701570	4×10^{-16}
0.3333	0.8	0.7417727914665396	0.7417727914665393	3×10^{-16}
0.6667	0.2	0.8143092188653853	0.8143092188653852	1×10^{-16}
0.6667	0.4	0.9945995259174347	0.9945995259174348	1×10^{-16}
0.6667	0.6	1.214806604220352	1.214806604220355	3×10^{-15}
0.6667	0.8	1.483768137025928	1.483768137025920	8×10^{-15}
1.0000	0.2	1.221402758160170	1.221402758160172	2×10^{-15}
1.0000	0.4	1.491824697641270	1.491824697641277	7×10^{-15}
1.0000	0.6	1.822118800390509	1.822118800390500	9×10^{-15}
1.0000	0.8	2.225540928492468	2.225540928492466	2×10^{-15}
1.3333	0.2	1.628496297454955	1.628496297454971	1.6×10^{-14}
1.3333	0.4	1.989049869365105	1.989049869365101	4×10^{-15}
1.3333	0.6	2.429430996560666	2.429430996560662	4×10^{-15}
1.3333	0.8	2.967313719959008	2.967313719959003	5×10^{-15}
1.6666	0.2	2.035589836749739	2.035589836749740	1×10^{-15}
1.6666	0.4	2.486275041088941	2.486275041088940	1×10^{-15}
1.6666	0.6	3.036743192730822	3.036743192730820	2×10^{-15}
1.6666	0.8	3.709086511425547	3.709086511425548	1×10^{-15}

The exact solution is $u_e(x, y) = e^{\lambda x+\mu y}$, where λ and μ are, respectively, 2 and 3. This experiment was solved using MQ approximation scheme by Kansa [1]. The author used a total of 30 points, including 12 scattered data points in the interior and 18 along the boundary. The reported results showed that the norm of error is 2.25×10^{-2}. Later, Mai-Duy and Tran-Cong [5] used IRBFN method and got a greater accuracy. They reported the norm of error is 2.2×10^{-4} for this experiment. In this study, we have used those same points (see Figure 1) and have achieved better accuracy in comparison with those two previous works. By using present approach, the norm of error that we have gotten in the best position is 3.26×10^{-7} ($r = 10^{-8}$). The results are shown in Tables 5 and 6. There are the same properties and results about this new way that we have explained in Experiment 1 (more accuracy and stability).

Experiment 3. Consider the following two-dimensional Poisson's equation in the elliptical region:

$$u_{xx} + u_{yy} = -2. \tag{3.6}$$

The great diameter of the ellipse is a, and small diameter is b. The boundary condition is $u = 0$ on all of boundary points. The equation of ellipse is

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1. \tag{3.7}$$

Table 2: Values of condition number and RMSE for some values of r of Experiment 1 (for the new method).

r	Condition number	RMSE
10^{-1}	1.616697379470667	2.06×10^{-15}
10^{-2}	1.005685744274181	1.99×10^{-15}
10^{-3}	1.000056810457828	2.97×10^{-15}
10^{-4}	1.000000568099880	2.34×10^{-15}
10^{-5}	1.000000005680999	1.03×10^{-15}
10^{-6}	1.000000000056810	5.05×10^{-16}
10^{-7}	1.000000000000585	1.32×10^{-15}
10^{-8}	1.000000000000003	7.77×10^{-16}
10^{-9}	1.000000000000000	2.26×10^{-16}
10^{-10}	1.000000000000000	2.26×10^{-16}
10^{-11}	1.000000000000000	2.26×10^{-16}

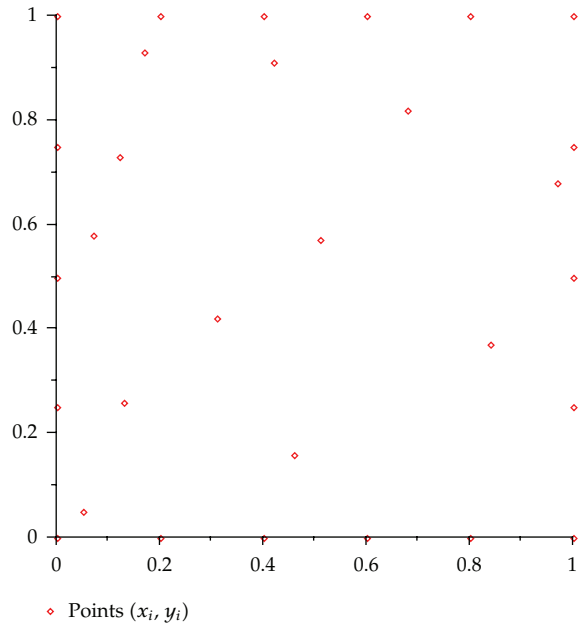

Figure 1: Location of scattered data points (12 of these points are interior points, and 18 points are boundary points) of Experiment 2.

Since the above ellipse is a symmetrical region, then we have solved this equation on the first quarter. The Dirichlet and the Neumann boundary conditions are

$$\begin{aligned} u_x &= 0 \quad \text{on the line } x = 0, \\ u_y &= 0 \quad \text{on the line } y = 0, \\ u &= 0 \quad \text{on the bound of ellipse.} \end{aligned} \tag{3.8}$$

The analytical solution is $u_e(x, y) = -[(x^2/a^2) + (y^2/b^2) - 1][a^2b^2/(a^2 + b^2)]$.

Table 3: Comparison between exact solution and approximate solution of IRBFN method on the polar coordinate of Experiment 1 [17].

x_i	y_i	Exact solution	Approximate solution of IRBFN method on the polar coordinate	Error of IRBFN method on the polar coordinate
0.3333	0.2	0.4070935392947847	0.407093539278427	1.64×10^{-11}
0.3333	0.4	0.4972251717238353	0.497225171776793	5.29×10^{-11}
0.3333	0.6	0.6073121961701566	0.607919812163175	3.95×10^{-11}
0.3333	0.8	0.7417727914665396	0.741772791449410	1.71×10^{-11}
0.6667	0.2	0.8143092188653853	0.814309218844426	2.09×10^{-11}
0.6667	0.4	0.9945995259174347	0.994599525931543	1.41×10^{-11}
0.6667	0.6	1.214806604220352	1.214806604225190	4.84×10^{-12}
0.6667	0.8	1.483768137025928	1.483768136992937	3.30×10^{-11}
1.0000	0.2	1.221402758160170	1.221402758117891	4.23×10^{-11}
1.0000	0.4	1.491824697641270	1.491824697612953	2.83×10^{-11}
1.0000	0.6	1.822118800390509	1.822118800390206	3.03×10^{-11}
1.0000	0.8	2.225540928492468	2.225540928445109	4.74×10^{-11}
1.3333	0.2	1.628496297454955	1.628496297401382	5.37×10^{-11}
1.3333	0.4	1.989049869365105	1.989049869316768	4.84×10^{-11}
1.3333	0.6	2.429430996560666	2.429430996510109	5.06×10^{-11}
1.3333	0.8	2.967313719959008	2.967313719902712	5.63×10^{-11}
1.6666	0.2	2.035589836749739	2.035711976964034	6.16×10^{-11}
1.6666	0.4	2.486275041088941	2.486424223498127	6.06×10^{-11}
1.6666	0.6	3.036743192730822	3.036925404550813	6.01×10^{-11}
1.6666	0.8	3.709086511425547	3.709309065459067	5.94×10^{-11}

Table 4: Values of condition number and RMSE for some values of r of Experiment 1 (for IRBFN method on the polar coordinate [17]).

r	Condition number	RMSE
0.005	1.471481785169896	9.17×10^{-10}
0.01	2.037248475218950	4.36×10^{-11}
0.5	4.678768168755817	4.92×10^{-11}
1.0	7.977675155476113	2.24×10^{-11}
1.5	10.67180878932467	5.17×10^{-11}
2.0	16.72051422331341	2.08×10^{-11}
2.5	17.54142375428882	3.22×10^{-11}
3.0	22.68488788104916	1.38×10^{-11}
3.5	94.83497808866327	9.66×10^{-12}
4.0	687.3099630448350	6.79×10^{-12}
4.5	2251.857925768735	7.09×10^{-11}

Table 5: Comparison between exact solution and approximate solution of the new method of Experiment 2.

x_i	y_i	Exact solution	Approximate solution of the new method
0.0	0.0	1.000000000000000	1.000000001520331
0.0	.25	2.117000016612675	2.116999962177082
0.0	.5	4.481689070338065	4.481689131011527
0.0	.75	9.487735836358526	9.487735908103340
0.0	1	20.08553692318767	20.08553702346013
1	0.0	7.389056098930650	7.389056124466515
1	.25	15.64263188418817	15.64263190711622
1	.5	33.11545195869231	33.11545243577679
1	.75	70.10541234668786	70.10541228133020
1	1	148.4131591025766	148.4131603446419
.2	0.0	1.491824697641270	1.491824709011273
.2	1	29.96410004739701	29.96410038594428
.4	0.0	2.225540928492468	2.225540978673106
.4	1	44.70118449330082	44.70118489448246
.6	0.0	3.320116922736547	3.320116658933115
.6	1	66.68633104092514	66.68633190025831
.8	0.0	4.953032424395115	4.953032677031149
.8	1	99.48431564193381	99.48431532815737
.05	.05	1.284025416687741	1.284025760755220
.13	.26	2.829217014351560	2.829217082663191
.46	.16	4.055199966844675	4.055199961753327
.31	.42	6.553504862191149	6.553504821832245
.07	.58	6.553504862191149	6.553504889746486
.12	.73	11.35888208000146	11.35888287993308
.42	.91	35.51659315162847	35.51659237035928
.51	.57	15.33288701990720	15.33288705221003
.68	.82	45.60420832084874	45.60420881755932
.84	.37	16.28101980178843	16.28101996810631
.97	.68	53.51703422749116	53.51703410219755
.17	.93	22.87397954244081	22.87397949274820

The results have been computed for $a = 10$ and $b = 8$. We have used 28 points that 17 of them are boundary points and 11 are interior points (see Figure 2) which were selected at random. In this new way, we have achieved a better accuracy in comparison with IRBFN method in the polar coordinates [17] (see Tables 7, 8, and 9). When we decrease parameter r, RMSE decreases too, and so we have better accurate solution. Also, the condition number closes to unity more and more. For $r = 10^{-9}$ (RSME $= 9.50 \times 10^{-12}$), it almost equals to unity (see Table 8).

We have shown in [17], in using of MQ-RBFs on the polar coordinate, when we have used 28 data points and we have been increasing the width parameter r, the accuracy of our

Table 6: Values of condition number and RMSE for some values of r of Experiment 2 (for the new method).

r	Condition number	RMSE
10^{-1}	15.34291952699858	8.11×10^{-4}
10^{-2}	4.566804572945806	5.72×10^{-4}
10^{-3}	2.134108059285889	1.03×10^{-4}
10^{-4}	1.082012712119965	2.06×10^{-4}
10^{-5}	1.010904486607965	6.33×10^{-5}
10^{-6}	1.003900496513122	5.70×10^{-5}
10^{-7}	1.001720933454105	4.61×10^{-7}
10^{-8}	1.001551378516199	3.26×10^{-7}
10^{-9}	1.001534906804984	3.98×10^{-7}
10^{-10}	1.001537639564036	6.78×10^{-7}
10^{-11}	1.002332833767664	4.13×10^{-7}

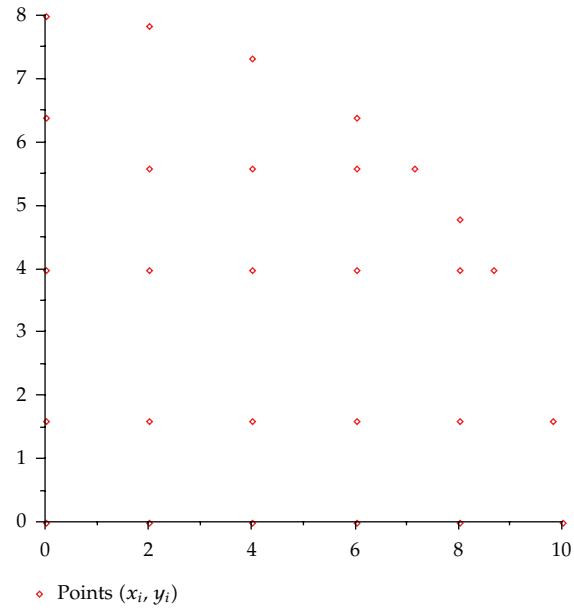

Figure 2: Location of scattered data points (11 points are interior points, and 17 points are boundary points) of Experiment 3.

solution has been increasing a little, though our system has been going to nonstability (see Table 9). In contrast, in this new way, when we are decreasing the parameter r, the accuracy of our solution is increasing too. Also, the condition number of our system is decreasing and closes to unity (see Table 8).

Here, we would like to emphasize that this experiment had been also solved by [1] and [3], and the norm of error is 3.57×10^{-4} in the DRBFN method [1], wherein the norm of errors in the DRBFN and the IRBFN methods are 1.36×10^{-5} and 5.41×10^{-7} [3], respectively.

Table 7: Comparison between exact solution and approximate solution of the new method of Experiment 3.

x_i	y_i	Exact solution	Approximate solution of the new method	Error of the new method
0.0	0.5	38.87195121951220	38.87195121952383	1.16×10^{-11}
0.0	1.5	37.65243902439024	37.65243902440016	9.92×10^{-12}
0.0	3.5	31.55487804878049	31.55487804879181	1.13×10^{-11}
0.0	5.5	20.57926829268293	20.57926829269648	1.35×10^{-11}
0.0	7.5	4.72560975609756	4.72560975611862	2.11×10^{-11}
2.0	0.0	37.46341463414634	37.46341463417008	2.37×10^{-11}
4.0	0.0	32.78048780487805	32.78048780491504	3.70×10^{-11}
6.0	0.0	24.97560975609756	24.97560975615097	5.34×10^{-11}
8.0	0.0	14.04878048780488	14.04878048786474	5.99×10^{-11}
10.0	0.0	0.00000000000000	0.00000000005427	5.43×10^{-11}
2.0	1.6	35.90243902439024	35.90243902440453	1.43×10^{-11}
4.0	1.6	31.21951219512195	31.21951219515048	2.85×10^{-11}
6.0	1.6	23.41463414634146	23.41463414638886	4.74×10^{-11}
8.0	1.6	12.48780487804878	12.48780487810293	5.41×10^{-11}
2.0	4.0	27.70731707317073	27.70731707318550	1.48×10^{-11}
4.0	4.0	23.02439024390244	23.02439024392700	2.46×10^{-11}
6.0	4.0	15.21951219512195	15.21951219516124	3.93×10^{-11}
8.0	4.0	4.292682926829268	4.29268292687660	4.73×10^{-11}
2.0	5.6	18.34146341463414	18.34146341465220	1.81×10^{-11}
4.0	5.6	13.65853658536585	13.65853658539350	2.76×10^{-11}
6.0	5.6	5.85365853658536	5.85365853662464	3.93×10^{-11}
9.798	1.6	–0.00031375609756	–0.00031375604674	5.08×10^{-11}
8.660	4.0	0.00171707317073	0.00171707321643	4.57×10^{-11}
7.141	5.6	0.00238790243902	0.00238790247911	4.01×10^{-11}
2.0	7.838	0.00350975609756	0.00350975612416	2.66×10^{-11}
4.0	7.332	0.00108292682927	0.00108292685914	2.99×10^{-11}
6.0	6.4	0.00000000000000	0.00000000003483	3.48×10^{-11}
8.0	4.8	0.00000000000000	0.00000000004390	4.39×10^{-11}

4. Conclusion

In the present paper, we have introduced a new way for numerical solution of Poisson's partial differential equation by a special combination between logarithmic and MQ-RBFs. We have showed that by this new method it does not need to control the parameter r (the width parameter) all times for preventing inaccuracy of solutions or increasing the value of

Table 8: Values of condition number and RMSE for some values of r of Experiment 3 (for the new method).

r	Condition number	RMSE
10^{-1}	2.817474760824544	1.02×10^{-10}
10^{-2}	1.590396300485768	3.62×10^{-11}
10^{-3}	1.253627364959679	1.49×10^{-10}
10^{-4}	1.062100283211657	6.82×10^{-11}
10^{-5}	1.050234416869035	4.53×10^{-11}
10^{-6}	1.030320686741992	2.75×10^{-11}
10^{-7}	1.015443720207797	4.46×10^{-11}
10^{-8}	1.005553497698158	1.76×10^{-11}
10^{-9}	1.003855850797432	9.50×10^{-12}
10^{-10}	1.002467354513745	2.02×10^{-11}
10^{-11}	1.000616708637105	3.82×10^{-11}

Table 9: Values of condition number and RMSE for some values of r of Experiment 3 (for IRBFN method on the polar coordinate [17]).

r	Condition number	RMSE
0.001	1.433292009119422	7.73×10^{-8}
0.01	1.833381468640938	$\mathbf{3.31 \times 10^{-8}}$
0.5	1.641721037366621	6.01×10^{-8}
1.0	2.368533981239234	2.91×10^{-8}
1.5	6.742734977317889	3.47×10^{-8}
2.0	7.268116615890860	1.33×10^{-8}
2.5	19.12284221789496	8.78×10^{-9}
3.0	56.72352660443333	9.62×10^{-9}
3.5	83.76935856474893	2.20×10^{-8}
4.0	101.1762326758897	9.00×10^{-9}
4.5	164.9216307945755	6.81×10^{-9}

condition number and having an ill-conditioned system. In this new way that is enough to consider the value of the parameter r smaller than 10^{-9}. In the aforesaid experiments, the accuracy is better than those before results obtained by [1, 3, 5, 16, 17] and the condition number of the systems is equal to unity. So we have some complete stable systems and more accurate solutions.

It should be noted that the computations associated with the experiments discussed above were performed by using Maple 13 on a PC, CPU 2.4 GHz.

Acknowledgments

This research paper has been financially supported by the office of vice chancellor for research of Islamic Azad University, Bushehr Branch, for the first author. The authors are very grateful to both reviewers for carefully reading the paper and for their comments and suggestions which have improved the paper. Also, authors acknowledge the editor professor Roberto Natalini, for managing the review process for this paper.

References

[1] E. J. Kansa, "Multiquadrics-a scattered data approximation scheme with applications to computational fluid-dynamics. II. Solutions to parabolic, hyperbolic and elliptic partial differential equations," *Computers & Mathematics with Applications*, vol. 19, no. 8-9, pp. 147–161, 1990.

[2] G. J. Moridis and E. J. Kansa, "The Laplace transform multiquadrics method: a highly accurate scheme for the numerical solution of linear partial differential equations," *Journal of Applied Science and Computations*, vol. 1, no. 2, pp. 375–407, 1994.

[3] M. Sharan, E. J. Kansa, and S. Gupta, "Application of the multiquadric method for numerical solution of elliptic partial differential equations," *Applied Mathematics and Computation*, vol. 10, pp. 175–302, 1997.

[4] E. J. Kansa and Y. C. Hon, "Circumventing the ill-conditioning problem with multiquadric radial basis functions: applications to elliptic partial differential equations," *Computers & Mathematics with Applications*, vol. 39, no. 7-8, pp. 123–137, 2000.

[5] N. Mai-Duy and T. Tran-Cong, "Numerical solution of differential equations using multiquadric radial basis function networks," *Neural Networks*, vol. 14, no. 2, pp. 185–199, 2001.

[6] A. I. Fedoseyev, M. J. Friedman, and E. J. Kansa, "Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary," *Computers & Mathematics with Applications*, vol. 43, no. 3–5, p. 439, 2002.

[7] E. A. Galperin and E. J. Kansa, "Application of global optimization and radial basis functions to numerical solutions of weakly singular Volterra integral equations," *Computers & Mathematics with Applications*, vol. 43, no. 3–5, pp. 491–499, 2002.

[8] N. Mai-Duy and T. Tran-Cong, "Approximation of function and its derivatives using radial basis function networks," *Applied Mathematical Modelling*, vol. 27, no. 3, pp. 197–220, 2003.

[9] L. Mai-Cao, "Solving time-dependent PDEs with a meshless IRBFN-based method," in *Proceedings of the International Workshop on Meshfree Methods*, 2003.

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

[11] L. Ling and E. J. Kansa, "Preconditioning for radial basis functions with domain decomposition methods," *Mathematical and Computer Modelling*, vol. 40, no. 13, pp. 1413–1427, 2004.

[12] A. Aminataei and M. M. Mazarei, "Numerical solution of elliptic partial differential equations using direct and indirect radial basis function networks," *European Journal of Scientific Research*, vol. 2, no. 2, pp. 2–11, 2005.

[13] A. Aminataei and M. Sharan, "Using multiquadric method in the numerical solution of ODEs with a singularity point and PDEs in one and two-dimensions," *European Journal of Scientific Research*, vol. 10, no. 2, pp. 19–45, 2005.

[14] D. Brown, L. Ling, E. J. Kansa, and J. Levesley, "On approximate cardinal preconditioning methods for solving PDEs with radial basis functions," *Engineering Analysis with Boundary Elements*, vol. 29, pp. 343–353, 2005.

[15] J. A. Munoz-Gomez, P. Gonzalez-Casanova, and G. Rodriguez-Gomez, "Domain decomposition by radial basis functions for time-dependent partial differential equations, advances in computer science and technology," in *Proceedings of the IASTED International Conference*, pp. 105–109, 2006.

[16] M. M. Mazarei and A. Aminataei, "Numerical solution of elliptic PDEs using radial basis function networks and comparison between RBFN and Adomian method," *Far East Journal of Applied Mathematics*, vol. 32, no. 1, pp. 113–126, 2008.

[17] A. Aminataei and M. M. Mazarei, "Numerical solution of Poisson's equation using radial basis function networks on the polar coordinate," *Computers & Mathematics with Applications*, vol. 56, no. 11, pp. 2887–2895, 2008.

[18] S. K. Vanani and A. Aminataei, "Multiquadric approximation scheme on the numerical solution of delay differential systems of neutral type," *Mathematical and Computer Modelling*, vol. 49, no. 1-2, pp. 234–241, 2009.

[19] S. Karimi Vanani and A. Aminataei, "Numerical solution of differential algebraic equations using a multiquadric approximation scheme," *Mathematical and Computer Modelling*, vol. 53, no. 5-6, pp. 659–666, 2011.

[20] M. A. Jafari and A. Aminataei, "Application of RBFs collocation method for solving integral equations," *Journal of Interdiscplinary Mathematics*, vol. 14, no. 1, pp. 57–66, 2011.