International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:4, No:9, 2010

A localized interpolation method using radial basis functions

Mehdi Tatari

Abstract—Finding the interpolation function of a given set of nodes is an important problem in scientific computing. In this work a kind of localization is introduced using the radial basis functions which finds a sufficiently smooth solution without consuming large amount of time and computer memory. Some examples will be presented to show the efficiency of the new method.

Keywords—Radial basis functions, Local interpolation method, Closed form solution.

I. Introduction

INTERPOLATION of a given set of points is an important problem in scientific computing. Polynomial interpolation method is efficient way but it is not convergent always. Radial basis functions are very efficient instruments for interpolating of a set of points which have been used in last 20 years. Convergence analysis of radial basis functions interpolation method has been carried out by several researchers, see, e.g. [7], [16], [8]. To find more about radial basis functions see [2], [9].

The main problem in the use of these functions is the instability behavior of this procedure. In fact the use of the radial basis functions arise a linear system of equations which its coefficient matrix is nonsingular and ill-conditioned [14]. Choice of interpolation nodes is important in the interpolation by the radial basis functions. Under certain conditions, the radial basis functions interpolation is equivalent to polynomial interpolation [12]. In this work a localization idea has been proposed for decreasing the size of the interpolation matrix. By this idea, the solution is found in a short time and using of more digits in floating point arithmetics is not needed. In the new method it is possible to construct a sufficiently smooth solution.

The organization of this paper is as follows:

In Section 2, the globally supported radial basis functions are introduced. A new efficient method for interpolation have been introduced in Section 3. Advantages of the present method over other existing methods are explained in this section. Also the solution of the resulted linear system of equations have been investigated. To present a clear overview of the method, in Section 4, it has been examined by several examples. A conclusion is drown in Section 5.

II. Globally supported radial basis functions

In this section RBFs methods have been introduced for interpolation of scattered data. Some well-known globally supported radial basis functions (RBFs) are listed in Table I. Let r be the Euclidean distant between $x^* \in \mathbb{R}^d$ and any $x \in \mathbb{R}^d$ i.e. $\|x - x^*\|_2$. A radial function $\phi^* = \phi(\|x - x^*\|_2)$ depends only on the distance between $x \in \mathbb{R}^d$ and fixed point $x^* \in \mathbb{R}^d$. This property results that radial basis function ϕ^* is radially symmetric about x^*. It is clear that functions in Table I are globally supported, infinitely differentiable and depends to a free parameter c. Let $x_1, x_2, ..., x_N$ be a given set of distinct points in $\mathbb{R}^d$. The idea behind the use of RBFs is interpolation by translations of a single function i.e. the interpolating RBFs approximation is considered as

TABLE I
Some well-known functions that generate RBFs.

Name of function	Definition
Gaussian (GA)	$\phi(r) = \exp(-cr^2)$
Inverse Multiquadric (IMQ)	$\phi(r) = (\sqrt{r^2 + c^2})^{-1}$
Inverse Quadric (IQ)	$\phi(r) = (r^2 + c^2)^{-1}$

$$F(x) = \sum_{i=1}^{N} \lambda_i \phi_i(x), \quad (1)$$

where $\phi_i(x) = \phi(\|x - x_i\|)$ and λ_i are unknown scalers for $i = 1, ..., N$. Assume that we want to interpolate the given values $f_i = f(x_i),\ i = 1, ..., N$. The unknown scalers λ_i are chosen so that $F(x_j) = f_j$ for $j = 1, ..., N$ which results the following linear system of equations

$$\mathbf{Az} = \mathbf{f}, \quad (2)$$

where $\mathbf{A} = [a_{i,j}]$ with $a_{i,j} = \phi_i(x_j)$ for $1 \le i, j \le N$, $\mathbf{z} = [\lambda_1, ..., \lambda_N]^T$ and $\mathbf{f} = [f_1, ..., f_N]^T$. Since all ϕ of the interest have global support, this method produces a dense matrix $\mathbf{A}$. The matrix $\mathbf{A}$ can be shown to be positive definite (and therefore nonsingular) for distinct interpolation points for GA, IMQ and IQ by the Schoenberg's theorem [15].

Although Matrix $\mathbf{A}$ is nonsingular in the above cases, usually it is very ill-conditioned i.e. the condition number of $\mathbf{A}$

$$\kappa_s(\mathbf{A}) = \|\mathbf{A}\|_s \|\mathbf{A}^{-1}\|_s, \quad s = 1, 2, \infty, \quad (3)$$

is a very large number. Therefore, a small perturbation in initial data may produce large amount of perturbation in the solution. Thus we have to use more precision arithmetics than standard floating point arithmetic in our computations.

Despite researches are done by many scientists to develop algorithms for selecting the values of c which produces most accurate interpolation (e.g. see [3], [11]), the optimal choice of shape parameter is still an open question.

Spectral accuracy is obtained in interpolating smooth data using global, infinitely differentiable RBFs [1], [2], [6], [7].

Department of Mathematical Sciences, Isfahan University of Technology, Isfahan, 84156-83111, Iran, Email: mtatari@cc.iut.ac.ir, (M. Tatari).

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:4, No:9, 2010

More information about the accuracy of the approximations made by the radial basis functions can be found in [4], [17]. Node distribution near the boundaries in the use of the radial basis functions is investigated for example in [5], [10].

III. The new method

Our strategy for overcoming to the instability is decreasing the size of matrix $\mathbf{A}$ by a localization method. Set

$$D = \left[\frac{N-1}{n-1}\right],$$

and

$$\Omega_i = \{(x_{in-i-n+2}, y_{in-i-n+2}), ..., (x_{in-i+1}, y_{in-i+1})\},$$

$$i = 1, ..., D,$$

$$\Omega_{D+1} = \{(x_{Dn-D+1}, y_{Dn-D+1}), ..., (x_N, y_N)\}.$$

In this case we have

$$\Omega = (\cup_{i=1}^{D} \Omega_i) \cup \Omega_{D+1},$$

$$\Omega_i \cap \Omega_{i+1} = \{(x_{in-i+1}, y_{in-i+1})\}, \; i = 1, ..., D.$$

For interpolation of the points in Ω_1 consider

$$F_1^n(x) = \sum_{i=1}^{n} \lambda_{1,i}\phi_i(x), \quad x_1 \leq x \leq x_n,$$

By interpolation of the first n points we have

$$\mathbf{A}\mathbf{z}_1 = \mathbf{f}_1, \tag{4}$$

where $\mathbf{A} = [a_{i,j}]$ with $a_{i,j} = \phi_i(x_j)$ for $i, j = 1, ..., n$, $\mathbf{z}_1 = [\lambda_{1,1}, ..., \lambda_{1,n}]^T$ and $\mathbf{f}_1 = [y_1, ..., y_n]^T$. In fact, the interpolation of points in Ω_1 has been presented using radial basis functions. Notice that it is easy to find the solution of system (4) because the size of its coefficient matrix is small enough. For finding the interpolation on the Ω_2 let

$$F_2^{n,k}(x) = \sum_{i=n}^{2n-1} \lambda_{2,i}\phi_i(x) +$$

$$\sum_{i=0}^{k-1} \gamma_{2,i} \frac{(x-x_n)^i \prod_{j=n}^{2n-1}(x-x_j)}{(i+1)! \prod_{j=n+1}^{2n-1}(x_n-x_j)}, \quad , x_n \leq x \leq x_{2n-1},$$

for $k = 1, 2, ...$ The unknowns $\lambda_{2,1}, ..., \lambda_{2,n}$ and $\gamma_{2,0}, ..., \gamma_{2,k-1}$ are found in a way that $F_2^{n,k}(x)$ satisfy the following conditions

$$F_2^{n,k}(x_i) = y_i, \quad (x_i, y_i) \in \Omega_2,$$

$$\frac{d^s}{dx^s} F_2^{n,k}(x_n) = \frac{d^s}{dx^s} F_1^n(x_n), \quad s = 1, ..., k,$$

which results a linear system of $n+k$ equations

$$\mathbf{B}\mathbf{z}_2 = \mathbf{f}_2,$$

where $\mathbf{z}_2 = [\lambda_{2,1}, ..., \lambda_{2,n}, \gamma_{2,0}, ...\gamma_{2,k-1}]^T$, $\mathbf{f}_2 = [y_n, ..., y_{2n-1}, \frac{d}{dx}F_1^n(x_n), ..., \frac{d^k}{dx^k}F_1^n(x_n)]^T$ and $\mathbf{B}$ is given by

$$\mathbf{B} = \begin{pmatrix} \phi_1(x_1) & . & . & . & \phi_n(x_1) & 0 & . & . & . & . & 0 \\ . & & & & . & . & & & & & . \\ . & & & & . & . & & & & & . \\ . & & & & . & . & & & & & . \\ \phi_1(x_n) & . & . & . & \phi_n(x_n) & 0 & . & . & . & . & 0 \\ \phi_1'(x_n) & . & . & . & \phi_n'(x_n) & d_{1,1} & 0 & . & . & . & 0 \\ \phi_1''(x_n) & . & . & . & \phi_n''(x_n) & . & d_{2,2} & . & & & . \\ . & & & & . & . & & . & . & & . \\ . & & & & . & . & & & . & . & . \\ . & & & & . & . & & & & . & 0 \\ \phi_1^{(k)}(x_n) & . & . & . & \phi_n^{(k)}(x_n) & d_{k,1} & . & . & . & . & d_{k,k} \end{pmatrix}, \tag{5}$$

where

$$d_{p,q} = g^{(p)}(x_n),$$

$$g(x) = \frac{(x-x_n)^{q-1} \prod_{j=n}^{2n-1}(x-x_j)}{q! \prod_{j=n+1}^{2n-1}(x_n-x_j)},$$

for $1 \leq q \leq p \leq k$, which results

$$d_{p,p} = 1, \quad p = 1, ..., k.$$

The last k conditions guarantee the continuity of the $k-$th differentiation of the interpolation function at $x = x_n$. In fact $F_2^{n,k}$ have been forced to have first k derivatives equal with of F_1^n at $x = x_n$.

Interpolation on the other points is performed similarly. Generally for finding the interpolation on the Ω_m we set

$$F_m^{n,k}(x) = \sum_{i=mn-m-n+2}^{mn-m+1} \lambda_{m,i}\phi_i(x) +$$

$$\sum_{i=0}^{k-1} \gamma_{m,i} \frac{(x-x_{mn-m-n+2})^i \prod_{j=mn-m-n+2}^{mn-m+1}(x-x_j)}{(i+1)! \prod_{j=mn-m-n+3}^{mn-m+1}(x_{mn-m-n+2}-x_j)},$$

with conditions

$$F_m^{n,k}(x_i) = y_i, \quad (x_i, y_i) \in \Omega_m,$$

$$\frac{d^s}{dx^s} F_m^{n,k}(x_{mn-m-n+2}) = \frac{d^s}{dx^s} F_{m-1}^{n,k}(x_{mn-m-n+2}), \quad s = 1, ..., k,$$

for finding unknowns $\lambda_{m,1}, ..., \lambda_{m,n}$ and $\gamma_{m,0}, ..., \gamma_{m,k-1}$. The resulted linear system in this case is

$$\mathbf{B}\mathbf{z}_m = \mathbf{f}_m,$$

where

$$\mathbf{z}_m = [\lambda_{m,1}, ..., \lambda_{m,n}, \gamma_{m,0}, ..., \gamma_{m,k-1}]^T,$$

and

$$\mathbf{f}_m = [y_{mn-m-n+2}, ..., y_{mn-m+1}, \frac{d}{dx}F_{m-1}^{n,k}(x_{mn-m-n+2}),$$

$$..., \frac{d^k}{dx^k}F_{m-1}^{n,k}(x_{mn-m-n+2})]^T.$$

Notice that the matrix $\mathbf{B}$ have been appeared again as coefficient matrix. In fact the value of the elements of the matrix $\mathbf{B}$ depends on the $x_i - x_j$ for certain i and j not on the values of the x_i and x_j.

Also the Matrix $\mathbf{B}$ is invertible because by a simple computation we find

$$\det(\mathbf{A}) = \det(\mathbf{B}).$$

Therefore, a unique interpolation function exists for each $\Omega_i,\ i = 1, ..., D$.

For finding the interpolation of the points of Ω_{D+1}, let $v = |\Omega_{D+1}|$ and consider

$$F_{D+1}^{v,k}(x) = \sum_{i=Dn-D+1}^{N} \lambda_{D+1,i}\phi_i(x)$$

$$+\sum_{i=0}^{k-1} \gamma_{D+1,i} \frac{(x - x_{Dn-D+1})^i \prod_{j=Dn-D+1}^{N}(x - x_j)}{(i+1)! \prod_{j=Dn-D+2}^{N}(x_{Dn-D+1} - x_j)},$$

with conditions

$$F_{D+1}^{v,k}(x_i) = y_i, \quad (x_i, y_i) \in \Omega_{D+1},$$

$$\frac{d^s}{dx^s}F_{D+1}^{v,k}(x_{Dn-D+1}) = \frac{d^s}{dx^s}F_{D}^{v,k}(x_{Dn-D+1}), \quad s = 1, ..., k,$$

for finding unknowns $\lambda_{D+1,1}, ..., \lambda_{D+1,N}$ and $\gamma_{D+1,0}, ..., \gamma_{D+1,k-1}$. The resulted linear system in this case is

$$\mathbf{B}_v\mathbf{z}_{D+1} = \mathbf{f}_{D+1},$$

where

$$\mathbf{z}_{D+1} = [\lambda_{D+1,1}, ..., \lambda_{D+1,N}, \gamma_{D+1,0}, ..., \gamma_{D+1,k-1}]^T,$$

$$\mathbf{f}_{D+1} = [y_{Dn-D+1}, ..., y_N, \frac{d}{dx}F_D^{v,k}(x_{Dn-D+1}),$$

$$..., \frac{d^k}{dx^k}F_D^{v,k}(x_{Dn-D+1})]^T,$$

and $\mathbf{B}_v$ is a matrix which is resulted from ignoring of the $(v+1)-$th to $n-$th rows and columns of the Matrix $\mathbf{B}$.

It is easy to verify that

$$\det(\mathbf{B}_v) = \det(\mathbf{A}_v),$$

where $\mathbf{A}_v$ is a $v \times v$ matrix which has been resulted from $\mathbf{A}$ by ignoring of the last $n - v$ rows and columns of matrix $\mathbf{A}$. By the Schoenberg's theorem [15] $\mathbf{A}_v$ is invertible. Therefore, the matrix $\mathbf{B}_v$ is a nonsingular matrix.

At the end we consider the piecewise function $F^{n,k}$ as the interpolation function for the points of the set Ω as

$$F^{n,k}(x) = \begin{cases} F_1^n(x) & \text{if } x_1 \le x \le x_n, \\ F_i^{n,k}(x) & \text{if } x_{in-i-n+2} \le x \le x_{in-i+1} \\ i = 1, ..., D, & \\ F_{D+1}^{v,k}(x) & \text{if } x_{Dn-D+1} \le x \le x_N. \end{cases}$$

Remark 1. The globally interpolation method is an special case of the new presented method with $n = N$.

Remark 2. Since in the new method computations have been based on n nodes, the parameter c should be chosen in a way that be compatible with n. Furthermore, in the present method despite of the global interpolation method use of more digits in floating point arithmetics is not needed.

Now we present an efficient method for solving of the resulted linear systems

$$\mathbf{A}\mathbf{z}_1 = \mathbf{f}_1, \tag{6}$$

$$\mathbf{B}\mathbf{z}_i = \mathbf{f}_i, \quad i = 2, ..., D. \tag{7}$$

$$\mathbf{B}_v\mathbf{z}_{D+1} = \mathbf{f}_{D+1} \tag{8}$$

Since the matrix $\mathbf{A}$ is positive definite, we can decompose it via Choleski method as

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T,$$

where $\mathbf{L} = [l_{ij}]$ is a lower triangular matrix. This factorization needs $\mathcal{O}(n^3)$ number of operations.

Therefore the linear system (6) can be written as

$$\mathbf{L}\mathbf{L}^T\mathbf{z}_1 = \mathbf{f}_1,$$

If we set $\mathbf{p}_1 = \mathbf{L}^T\mathbf{z}_1$ then using the forward substitution with $\mathcal{O}(n^2)$ number of operations, the solution of the

$$\mathbf{L}\mathbf{p}_1 = \mathbf{f}_1,$$

is founded with

$$p_{1,i} = \frac{1}{l_{ii}}\left(y_i - \sum_{j=1}^{i-1} l_{ij}p_{1,j}\right), \quad i = 1, ..., n.$$

Also we find $\mathbf{z}_1$ by solving

$$\mathbf{L}^T\mathbf{z}_1 = \mathbf{p}_1,$$

using the backward substitution which needs $\mathcal{O}(n^2)$ number of operations with

$$\lambda_{1,i} = \frac{1}{l_{ii}}\left(p_{1,i} - \sum_{j=i+1}^{n} l_{ji}\lambda_{1,j}\right), \quad i = n, ..., 1,$$

and the solution of the linear system (6) is founded.

For solving the linear systems (7), according to the (5) we rewrite them as

$$\mathbf{A}\mathbf{z}_{i,1} = \mathbf{f}_{i,1}, \tag{9}$$

$$\mathbf{B}_{n+j}.\mathbf{z}_i^T = \gamma_{i,j-1}, \quad j = 1, ..., k, \tag{10}$$

where $\mathbf{z}_{1,i} = [\lambda_{i,1}, ..., \lambda_{i,n}]^T$, $\mathbf{f}_{i,1} = [y_{in-i-n+2}, ..., y_{in-i+1}]$ and $\mathbf{B}_{n+j}$ is the $(n+j)-$th row of $\mathbf{B}$.

Now since we have computed the Choleski decomposition of $\mathbf{A}$, by an only forward and backward substitutions the solution of (9) will be found. Also according to (5) and (10) the unknowns $\gamma_{i,0}, ..., \gamma_{i,k-1}$ are found as

$$\gamma_{i,j} = \frac{d^{j+1}F_{i-1}^{n,k}(x_{in-i-n+2})}{dx^{j+1}} -$$

$$\sum_{p=1}^{n} \phi_p^{(j+1)}(x_{in-i-n+2})\lambda_{i,p} + \sum_{q=1}^{j} d_{j+1,q}\gamma_{i,q-1}, \quad j = 0, ..., k-1.$$

Also for finding the solution of the linear system (8) we rewrite it as

$$\mathbf{A}_v\mathbf{z}_{D+1,1} = \mathbf{f}_{D+1,1},$$

$$\mathbf{B}_{v+j}.\mathbf{z}_{D+1}^T = \gamma_{D+1,j-1}, \quad j = 1, ..., k.$$

Notice that the Choleski decomposition of the matrix $\mathbf{A}_v$ is in the form

$$\mathbf{A}_v = \mathbf{L}_v \mathbf{L}_v^T,$$

where the matrix $\mathbf{L}_v$ obtain by ignoring the last $(n-v)$ rows and columns of matrix $\mathbf{L}$. In fact any computations for finding Choleski decomposition for matrix $\mathbf{A}_v$ is not needed. Again the solution obtain by only a forward and backward substitutions similar the previous cases.

In this method, we find the solution of the problem in hole of the domain by Choleski decomposition, forward and backward substitutions. Notice n and k are chosen as small integers and so the size of matrices $\mathbf{A}$ and $\mathbf{B}$ is small. This method is specially efficient for finding the interpolation in the large domains.

IVNUMERICAL EXAMPLES

AExample 1

In this example we consider the problem of finding interpolation function of the points

$$(x_i, f(x_i)), \quad i = 1, ..., N,$$

where $x_i = (i-1)h$ and $f(x) = \sin(x)$.

In Table II, the results of the new method have been shown using the Gaussian radial basis function with shape parameter $c = 0.1$, $n = 4$, $k = 2$ $h = 0.1$ and $\delta = 10$ digits in floating points arithmetics by the RMS error

$$E^2 = \left(\frac{1}{N-1} \sum_{i=1}^{N-1} \left| f(x_i + \frac{h}{2}) - F^{n,k}(x_i + \frac{h}{2}) \right| \right)^{\frac{1}{2}},$$

and the max error

$$E^\infty = \max_{1 \le i \le N-1} \left| f(x_i + \frac{h}{2}) - F^{n,k}(x_i + \frac{h}{2}) \right|,$$

also we propose the time which have been consumed for finding the solution in this table. In this case the matrix $\mathbf{B}$ is

$$\mathbf{B} = \begin{pmatrix} 1 & 0.999 & 0.996 & 0.991 & 0 & 0 \\ 0.999 & 1 & 0.999 & 0.996 & 0 & 0 \\ 0.996 & 0.999 & 1 & 0.999 & 0 & 0 \\ 0.991 & 0.996 & 0.999 & 1 & 0 & 0 \\ 0 & 0.0199 & 0.0398 & 0.0594 & 1 & 0 \\ -0.2 & -0.199 & -0.197 & -0.194 & -36.666 & 1 \end{pmatrix}.$$

In Table III, the global interpolation method has been used for this problem using the Gaussian radial basis function with shape parameter $c = 0.1$, and $h = 0.1$. It is clear from the contents of Table II that the solution will obtain using the new method in a short time and without loss of accuracy in the large domains.

In Table IV, the results have been for $N = 241$, $c = 0.1$, $k = 2$, $\delta = 10$ and some values of n.

Table V shows the results of the new method for $\delta = 10$, $k = 2$ and consumed time for $N = 241$ and some values of c.

In the Table VI, the results of the new method have been shown for $k = 2$, $n = 7$ and consumed time for $N = 241$ and

TABLE II
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $\delta = 10$, $k = 2$, $n = 4$ AND CONSUMED TIME FOR SOME VALUES OF N USING THE NEW METHOD.

N	E^2	E^∞	time(s)
31	0.202×10^{-2}	0.184×10^{-6}	0.047
61	0.191×10^{-2}	0.434×10^{-7}	0.094
91	0.215×10^{-2}	0.219×10^{-7}	0.141
121	0.212×10^{-2}	0.471×10^{-8}	0.188
151	0.205×10^{-2}	0.582×10^{-8}	0.234
181	0.210×10^{-2}	0.269×10^{-7}	0.282
211	0.204×10^{-2}	0.633×10^{-8}	0.312
241	0.205×10^{-2}	0.295×10^{-7}	0.374
271	0.206×10^{-2}	0.542×10^{-8}	0.406

TABLE III
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $\delta = 10$, AND CONSUMED TIME FOR SOME VALUES OF N USING THE GLOBAL INTERPOLATION METHOD.

N	E^2	E^∞	time(s)
31	0.109×10^{-1}	0.155×10^{-4}	0.047
61	0.161×10^{-1}	0.113×10^{-5}	0.391
91	0.238×10^{-1}	0.133×10^{-4}	1.344
121	0.189×10^{-1}	0.115×10^{-5}	3.438
151	0.193×10^{-1}	0.259×10^{-5}	7.078
181	0.514×10^{-2}	0.220×10^{-6}	13.326
211	0.744×10^{-2}	0.252×10^{-8}	22.312
241	0.136×10^{-1}	0.236×10^{-6}	36.796
271	0.521×10^{-2}	0.190×10^{-6}	56.219

TABLE IV
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $\delta = 10$, $k = 2$ AND CONSUMED TIME FOR $N = 241$ USING THE NEW METHOD.

n	E^2	E^∞	time(s)
2	0.358×10^{-1}	0.314×10^{-5}	1.861
3	0.427×10^{-2}	0.352×10^{-6}	0.436
4	0.205×10^{-2}	0.295×10^{-7}	0.374
7	0.226×10^{-2}	0.811×10^{-8}	0.343
13	0.817×10^{-2}	0.258×10^{-7}	0.437

TABLE V
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $\delta = 10$, $k = 2$, $n = 7$ AND CONSUMED TIME FOR $N = 241$ USING THE NEW METHOD.

c	E^2	E^∞	time(s)
1	0.100×10^{-2}	0.371×10^{-8}	0.484
5	0.610×10^{-2}	0.628×10^{-7}	0.390
10	0.164×10^{-1}	0.112×10^{-5}	0.375
15	0.286×10^{-1}	0.137×10^{-4}	0.376
20	0.414×10^{-1}	0.128×10^{-4}	0.374

TABLE VI
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $k = 2$, $n = 7$, $c = 1$ AND CONSUMED TIME FOR $N = 241$ USING THE NEW METHOD.

δ	E^2	E^∞	time(s)
10	0.100×10^{-2}	0.371×10^{-8}	0.484
16	0.459×10^{-3}	0.306×10^{-8}	0.390
24	0.459×10^{-3}	0.306×10^{-8}	0.390
32	0.459×10^{-3}	0.306×10^{-8}	0.390

International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:4, No:9, 2010

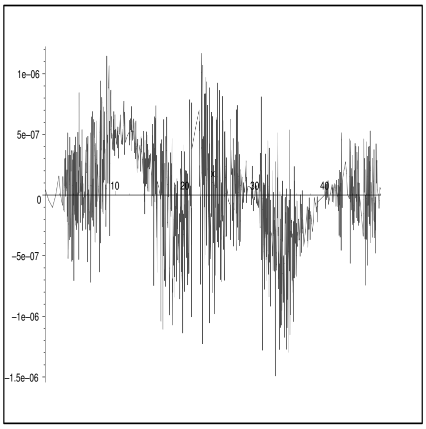

Fig. 1. Plot of $u(x) - \sin(x)$ for global interpolation function using the Gaussian radial basis functions with $\delta = 16$, $c = 0.1$ and $N = 481$.

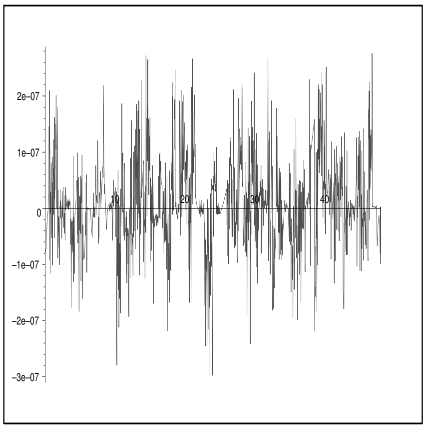

Fig. 2. Plot of $F^{7,2}(x) - \sin(x)$ for local interpolation function $F^{7,2}(x)$ ($n = 7$ and $k = 2$) using the Gaussian radial basis functions with $\delta = 16$, $c = 0.1$ and $N = 481$.

some values of δ. It is clear from the contents of this table that the use of the more precision is not needed.

In Figure 1, the error $u(x) - \sin(x)$ is plotted for global interpolation function $u(x)$ with $\delta = 16$, $c = 0.1$ and $N = 481$. Also in Figure 2 the error $F^{7,2}(x) - \sin(x)$ is plotted for the local interpolation function with $n = 7$, $k = 2$, $\delta = 16$, $c = 0.1$ and $N = 481$. As an interesting points, the consumed time for finding the global interpolation function is $t = 620.595$ seconds while for local interpolation this is only $t = 1.031$ second.

BExample 2

In this example consider the function

$$f(x) = \frac{1}{1 + x^2},$$

for finding the interpolation function in $[-6, 6]$.

In Table VII, the values of E^2 and E^∞ has been presented for $c = 0.1$, $n = 4$, $\delta = 10$, $k = 2$ and some values of h (distance of two adjacent points). Also in Table VIII, the corresponding results have been shown using global interpolation method.

TABLE VII
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $k = 2$, $n = 4$, $\delta = 10$ AND CONSUMED TIME USING THE NEW INTERPOLATION METHOD.

h	E^2	E^∞	time(s)
0.05	0.900×10^{-2}	0.913×10^{-8}	0.203
0.1	0.308×10^{-2}	0.264×10^{-7}	0.109
0.2	0.721×10^{-2}	0.479×10^{-6}	0.062

TABLE VIII
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $\delta = 10$ AND CONSUMED TIME FOR SOME VALUES OF h (DISTANCE OF TWO ADJACENT POINTS) USING THE GLOBAL INTERPOLATION METHOD.

h	E^2	E^∞	time(s)
0.05	1.151	0.435×10^{-2}	35.467
0.1	0.468	0.295×10^{-3}	3.390
0.2	0.767	0.111×10^{-1}	0.391

In Table IX, the results have been shown for $c = 0.1$, $k = 2$, $h = 0.1$, $\delta = 10$ and some values of n. Using the same parameters in the global interpolation method we have

$$E^2 = 2.658,$$

$$E^\infty = 0.437 \times 10^{-1},$$

with the consumed time $t = 35.312$. These experimental results shows the efficiency and accuracy of the new method against the global interpolation scheme.

In the Figures 3 and 4, the $F^{7,2}(x) - \frac{1}{1+x^2}$ (error of the new method) and $u(x) - \frac{1}{1+x^2}$ (error of the global interpolation method using the Gaussian radial basis functions) have been plotted respectively for $\delta = 32$, $c = 0.1$ and $N = 121$.

In Figure 5, the error $s(x) - \frac{1}{1+x^2}$ is plotted for cubic natural spline function $s(x)$ with $\delta = 32$ and $N = 121$. It is clear from the Figures 4 and 5 that the new method, without any need of solving large system of linear equations provides an accurate approximation comparing with natural cubic splines.

V. CONCLUSION

The problem of interpolation of a given set of points is a hot subject of research in applied mathematics. The use of the global interpolation schemes usually occurs some problems. In this work a local interpolation method has been proposed using the globally supported radial basis functions. The method has been designed in a way that the interpolation function will be sufficiently smooth in the domain of the problem. The computational cost of the method is lower than the that of computational effort which is needed for finding the global

TABLE IX
THE ERRORS E^2, E^∞ OF INTERPOLATION FUNCTION USING GAUSSIAN RADIAL BASIS FUNCTIONS WITH $c = 0.1$, $\delta = 10$, $h = 0.1$, $k = 2$ AND CONSUMED TIME USING THE NEW METHOD.

n	E^2	E^∞	time(s)
2	0.953×10^{-2}	0.304×10^{-5}	2.080
3	0.201×10^{-2}	0.169×10^{-7}	0.235
4	0.308×10^{-2}	0.264×10^{-8}	0.1009
7	0.379×10^{-2}	0.580×10^{-8}	0.188
13	0.329×10^{-1}	0.580×10^{-8}	0.249

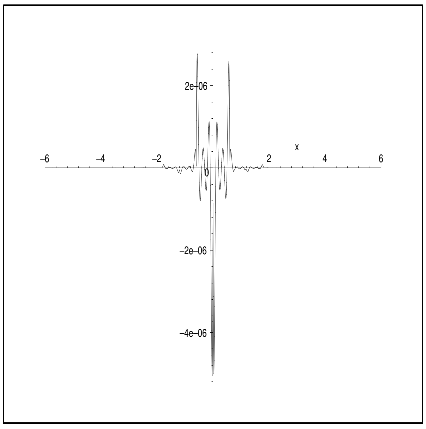

Fig. 3. Plot of $F^{7,2}(x) - \frac{1}{1+x^2}$ for local interpolation function ($n = 7$ and $k = 2$) using the Gaussian radial basis functions with $\delta = 32$, $c = 0.1$ and $N = 121$.

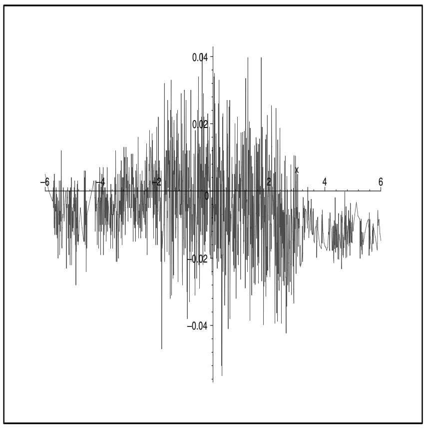

Fig. 4. Plot of $u(x) - \frac{1}{1+x^2}$ for global interpolation function $u(x)$ using the Gaussian radial basis functions with $\delta = 32$, $c = 0.1$ and $N = 121$.

interpolation method. In fact we need to solve only an small system of linear equations and the solution of other resulted systems will find explicitly with only backward and forward substitutions. Also the provided solution is accurate. The method is more efficient when we are interested of solving a problem in large domains.

REFERENCES

[1] M. D. Buhmann, Spectral convergence of multiquadric interpolation, Proc. Edinburg Math. Soc. 36 (1993) 319-333.

[2] M. D. Buhmann, Radial basis functions, Combridge University Press, Combridge, 2003.

[3] R. E. Carlson, T. A. Foley, The parameter r^2 in multiquadric interpolation, Proc. Edinburg Math. Soc. 36 (1993) 319-333.

[4] B. Fornberg, N. Flyer, Accuracy of radial basis function interpolation and derivative approximations on 1-D infinite grids, Adv. Comput. Math. 23 (2005) 5-20.

[5] B. Fornberg, T. Driscoll, G. Wright, Charles, Observations on the behavior of radial basis function approxiamtions near boundaries, Comput. Math. Appl. 43 (2002) 473-490.

[6] W. R. Madych, S. A. Nelson, Error bounds for multiquadric interpolation, in: C. Chui, L. Schumaker, J. Ward(Eds.), Approximation Theory VI, Academic Press, New York, 1989, pp. 413-416.

[7] W. R. Madych, S. A. Nelson, Multivariate interpolation and conditionally positive definite functions, ii, Math. Comp. 4 (1990) 211-230.

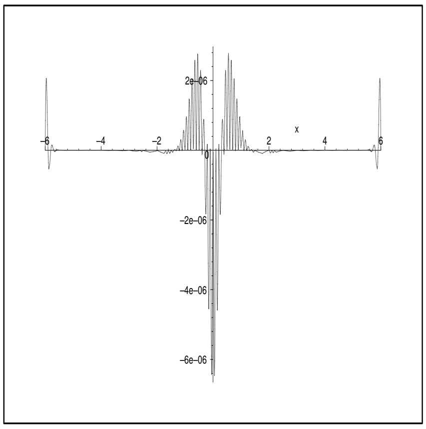

Fig. 5. Plot of $s(x) - \frac{1}{1+x^2}$ for cubic natural spline function $s(x)$ with $\delta = 32$ and $N = 121$.

[8] W. R. Madych, Miscellaneous error bounds for multiquadric and related interpolators, Comput. Math. Appl. 24 (1992) 121-138.

[9] M. Powell, The theory of radial basis function approximation in 1990, in: W. Light(Ed.), Advances in Numerical Analysis, vol. II: Wavelets, Subdivision Algorithms and Radial Functions, 1990.

[10] R. Platte, T. Driscoll, Computing eigenmodes of elliptic operators using radial basis functions, Comput. Math. Appl. 48 (2004) 561-576.

[11] S. Rippa, An algorithm for selecting a good parameter c in radial basis function interpolation, Adv. Comput. Math. 11 (1999) 193-210.

[12] R. Platte, T. Driscoll, Polynomials and potential theory for Gaussian radial basis function interpolation, SIAM J. Numer. Anal. 43 (2005) 750-766.

[13] S. A. Sarra, Adiptive radial basis function method for time dependent partial differential equations, Applied Numerical Mathemetics 54 (2005) 79-94.

[14] R. Schaback, Error estimates and condition numbers for radial basis function interpolation, Adv. Comput. Math. 3 (1995) 251-264.

[15] I. J. Schoenberg, Metric spaces and comletely monotone functions, Ann. Math. 39 (1938) 811-841.

[16] H. Wendland, Gaussian interpolation revisited, in trend in Approximation Theory, K.Kopotun, T. Lyche, and N. Neamtu, eds., Vanderbilt University Press, nashville, TN, 2001, 1-10.

[17] J. Yoon, Spectral approximation orders of radial basis function interpolation on the Sobolov space, SIAM J. Math. Anal. 33 (2001) 946-958.

Mehdi Tatari received a B.S. degree from Isfahan University in 2001, an M.S.C degree from Amirkabir University of Technology in 2003, and a Ph.D. degree from Amirkabir University of Technology in 2007 all in applied mathematics. He joined the Isfahan University of Technology in 2007 and is currently assistant professor of numerical analysis. His research interest include finite difference, finite element and meshless methods for numerical solution of PDEs.