Journal of the Nigerian Association of Mathematical Physics
Volume **25 (November, 2013), pp 147 – 156**

On an Adaptive Method for Radial Basis Function Interpolation

Terhemen Aboiyar and Tersoo Luga

**Department of Mathematics/Statistics/Computer Science,
University of Agriculture, PMB 2373, Makurdi, Nigeria**

Abstract

In this paper, we have used two radial basis functions namely, multiquadrics and the thin plate splines to implement the adaptive residual subsampling method in one dimension. Two functions with localized features were chosen to establish the efficiency of the adaptive method and illustrate the advantages of the adaptive radial basis function interpolation over radial basis function interpolation on uniform grids in one dimension. The numerical results show that the adaptive radial basis function interpolation method performed better with the thin plate splines than the multiquadrics and also the adaptive interpolation method yields a better approximation to functions that have localized features than radial basis function interpolation on uniform grids.

***Keywords*:** radial basis functions, adaptive interpolation, multiquadrics, thin plate splines.

1.0 Introduction

Radial basis function (RBF) methods are well-known traditional and powerful tools for multivariate interpolation from scattered data. According to Buhmann [1], they have nowbecome a viable choice as a method for numerical solution of partial differential equations as shown in [2-4]. Radial basis function methods offer numerous advantages such as no need for mesh or triangulation, simple implementation and dimension independence, no need for polygonization for boundaries and are preferred to low order methods such as finite difference, finite volumes and finite elements, see Cheng *et al* [5] and Buhmann&Dyn [6]

According to Driscoll &Heryudono [7], fixed grid radial basis function methods are effective tools for interpolation and approximating scattered data problems but they cannot be conveniently used in problems that exhibit high degree of localization such as steep gradients, corners and topological changes resulting from nonlinearity, thus the adaptive methods are preferred. Many researchers in recent years have incorporated RBF methods in several adaptive schemeswhich have produced good results (see Examples in [8-12]).

Driscoll &Heryudono [7] constructed a new algorithm called adaptive residual subsampling method and applied it to interpolation problems, initial value problems and boundary value problems with localized features. Although the method performed well on the tested problems, they only utilized the multiquadric radial basis function which contains a free parameter. The choice of the free parameter plays a critical role in the performance of the method.

In this paper, we will apply the adaptive residual subsampling method of Driscoll &Heryudono [7]to the thin plate spline radial basis function which is parameter free and solve some interpolation problems with localized features in one dimension. The aim is to compare the adaptive interpolation method with the multiquadric radial basis function with adaptive interpolation with thin plate splines and also demonstrate the superiority of the adaptive grid interpolation over interpolation on uniform grid.

The rest of the paper is organized as follows: In Section 2, interpolation by radial basis function is considered. In Section 3, we report our numerical results and findings by considering two examples and finally a brief conclusion is presented in Section 4.

2.0 Methods

In this section, we will discuss the general theory of radial basis function interpolation.Our description in here is based on the paper of Iske [13]

Corresponding author: ***Terhemen Aboiyar,*** E-mail: t_aboiyar@yahoo.co.uk, Tel.: +2347069121825

2.1 Radial Basis Function

A radial basis function (RBF) is a real valued function whose value depends only on the distance from the origin, so that$\Phi(x) = \phi(\|x\|)$; or alternatively on the distance from some other point$c$, called a centre, so that$\Phi(x, c) = \phi(\|x - c\|)$. Any function Φthat satisfies the property $\Phi(x) = \phi(\|x\|)$ is a radial basis function. The norm is usually the Euclidean norm, although the use of other norms is also possible. Sums of radial basis functions are typically used to approximate given functions. Radial basis functions (RBFs) are means to approximate multivariable functions by linear combinations of terms based on a single univariate function. Commonly used types of radial basis functions include:

i. Gaussian,$\phi(r, \varepsilon) = exp(-\varepsilon r)^2$
ii. Linear radial basis function, $\phi(r, \varepsilon) = \varepsilon r$
iii. Multiquadrics, $\phi(r, \varepsilon) = \sqrt{1 + (\varepsilon r)^2}$
iv. Inverse multiquadrics, $\phi(r, \varepsilon) = \frac{1}{\sqrt{1+(\varepsilon r)^2}}$
v. Polyharmonic splines, $\phi(r) = r^k,\ k = 1,3,5, \ldots$
$\phi(r) = r^k \log r\,, k = 2,4,6, \ldots$
vi. Thin plate splines (a special polyharmonic splines when$k = 2$), $\phi(r) = r^2 \log r$.

Radial basis function interpolation using scattered data requires a data vector $f\big|_X = (f(x_1), f(x_2), \ldots, f(x_n))^T \in \mathbb{R}^n$ of function values sampled from unknown function $f: \mathbb{R}^d \longrightarrow \mathbb{R}$ at a scattered data set $X = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^d, d \geq 1$. Scattered interpolation requires computing a suitable interpolant $s: \mathbb{R}^d \longrightarrow \mathbb{R}$ satisfying $s\big|_X = f\big|_X$ i.e.

$$s(x_j) = f(x_j), \qquad \text{for } 1 \leq j \leq n \tag{2.1}$$

The radial basis function interpolation scheme works with a fixed radial function$\phi: [0, \infty) \longrightarrow \mathbb{R}$, and the interpolant s in (3.1) is assumed to have the form

$$s(x) = \sum_{j=1}^{n} c_j \phi(\|x - x_j\|) + p(x), \qquad p \in P_m^d \tag{2.2}$$

where$\|\cdot\|$ is the Euclidean norm on$\mathbb{R}^d$. P_m^ddenotes the linear space containing all real valued polynomials in d variables of degree at most $m - 1$, where $m \equiv m(\phi)$ is said to be the order of the basis function ϕ.

To evaluate (2.2), we consider two cases; when $m = 0$ (positive definite radial basis functions)and $m > 0$ (conditionally positive radial basis functions).

2.2 The Linear System Associated with Radial Basis Functions with m=0

Radial basis functions that have a polynomial part $p(x), p \in P_m^d$in (2.2) omitted have order $m = 0$. Examplesare Gaussians, inverse quadrics and inverse multiquadrics. If the polynomial part in (2.2) is omitted then the interpolant(2.2) becomes

$$s(x) = \sum_{j=1}^{n} c_j \phi(\|x - x_j\|) \tag{2.3}$$

Weconsider the interpolant (2.3) and the interpolating condition (2.1). The unknown coefficients$c = (c_1, \ldots, c_n)^T \in \mathbb{R}^n$ of s can be computed from the linear system

$$A_{\phi,X}.\, c = f\big|_X \tag{2.4}$$

where

$$A_{\phi,X} = \begin{bmatrix} \phi(\|x_1 - x_1\|) & \phi(\|x_1 - x_2\|) & \cdots & \phi(\|x_1 - x_n\|) \\ \phi(\|x_2 - x_1\|) & \phi(\|x_2 - x_2\|) & \ldots & \phi(\|x_2 - x_n\|) \\ \vdots & \vdots & & \vdots \\ \phi(\|x_n - x_1\|) & \phi(\|x_n - x_2\|) & \ldots & \phi(\|x_n - x_n\|) \end{bmatrix};\ c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \text{ and } f\big|_X = \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix} \tag{2.5}$$

2.3 The Linear System Associated with Radial Basis Functions with $m > 0$

We consider the interpolant(2.2) and the interpolating condition (2.1). In this case, the interpolant s contains a nontrivial polynomial part yielding $q$additonal degree of freedom, where $q = \binom{m-1+d}{d}$ is the dimension of the space P_m^d. These additional degrees of freedom are usually eliminated by requiring q vanishing moment conditions

$$\sum_{j=1}^{n} c_j\, p(x_j) = 0 \quad \text{for all} p \in P_m^d \tag{2.6}$$

Altogether, this amounts to solving the linear system

$$\begin{bmatrix} A_{\phi,X} & P_X \\ P_X^T & 0 \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} f|_X \\ 0 \end{bmatrix} \tag{2.7}$$

where we let

$$P_X = \left((x_j)^{\alpha}\right)_{1\le j\le n,\ |\alpha|<m} \in \mathbb{R}^{n\times q} \text{ where } d = (d_\alpha)_{|\alpha|<m} \in \mathbb{R}^q \tag{2.8}$$

for the coefficients of the polynomial part in (3.2). An example are the polyharmonic splines.

2.4 Existence and Uniqueness of the Solution ofLinear Systems Associated with RBFs

According to Iske [13], for the linear system (2.4) associated with positive definite RBFs to exist and be unique, the coefficient matrix $A_{\phi,X}$ must be positive definite.

On the other hand, for the solution of (2.7) to exist and be unique then

$$c^T A_{\phi,X} \cdot c > 0, \qquad \text{for all } X \text{and} c \in \mathbb{R}^n \backslash \{0\} \text{with} P_X^T \cdot c = 0 \tag{2.9}$$

2.5 Interpolation Matrices for Multiquadrics and Thin Plate Splines RBFs

The basis function of the multiquadrics is given by

$$\phi(r) = (1 + (\varepsilon r)^2)^{\nu} r \in \mathbb{R}, \nu = \frac{1}{2}. \tag{2.10}$$

Therefore, the interpolant of the multiquadrics is of the form

$$s(x) = \sum_{j=1}^{n} c_j\, (1 + \left(\varepsilon \|x - x_j\|\right)^2)^{\nu} \tag{2.11}$$

Considering (2.11) and the interpolating condition (2.1) with $d = 1$ and ν=$\frac{1}{2}$ leads us to the linear system

$$\begin{bmatrix} \sqrt{1+(\varepsilon|x_1-x_1|)^2} & \sqrt{1+(\varepsilon|x_1-x_2|)^2} & \cdots & \sqrt{1+(\varepsilon|x_1-x_n|)^2} \\ \sqrt{1+(\varepsilon|x_2-x_1|)^2} & \sqrt{1+(\varepsilon|x_2-x_2|)^2} & \cdots & \sqrt{1+(\varepsilon|x_2-x_n|)^2} \\ \vdots & \vdots & & \vdots \\ \sqrt{1+(\varepsilon|x_n-x_1|)^2} & \sqrt{1+(\varepsilon|x_n-x_2|)^2} & \cdots & \sqrt{1+(\varepsilon|x_n-x_n|)^2} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix} \tag{2.12}$$

where the $n \times n$matrix on the left hand side of the equation is the interpolation matrix for the multiquadric in one dimension.

Similarly, the basis function of the thin plate splines is given by

$$\phi(r) = r^2 \log r,\ r \in \mathbb{R}$$

Therefore, the interpolant of the thin plate splines is given by

$$s(x) = \sum_{j=1}^{n} c_j \left\|x - x_j\right\|^2 \log\left\|x - x_j\right\| + \ p(x), \qquad p \in P_m^d \tag{2.13}$$

where$p(x)$ is a polynomial of degree one since the thin plate splines is a RBF of order $m = 2$.

If $d = 1$, we have

$$p(x) = d_0 + d_1 x$$

and so

$$s(x) = \sum_{j=1}^{n} c_j \left|x - x_j\right|^2 \log\left|x - x_j\right| + d_0 + d_1 x.$$

The vanishing moment condition is

$$\sum_{j=1}^{n} c_j \, p(x_j) = \sum_{j=1}^{n} c_j \left(d_0 + d_1 x_j\right) = 0$$

which means that

$$\sum_{j=1}^{n} c_j \, p(x_j) = d_0(c_1 + c_2 + \cdots + c_n) + d_1(c_1 x_1 + c_2 x_2 + \cdots + c_n x_n) = 0$$

but d_0 and $d_1 \neq 0$, therefore

$c_1 + c_2 + \cdots + c_n = 0$, and

$c_1 x_1 + c_2 x_2 + \cdots + c_n x_n = 0.$

Therefore, the linear system associated with the thin splines for d=1 is given by

$$\begin{bmatrix} |x_1 - x_1|^2 \log|x_1 - x_1| & |x_1 - x_2|^2 \log|x_1 - x_2| & \cdots & |x_1 - x_n|^2 \log|x_1 - x_n| & 1 & x_1 \\ |x_2 - x_1|^2 \log|x_2 - x_1| & |x_2 - x_2|^2 \log|x_2 - x_2| & \cdots & |x_2 - x_n|^2 \log|x_2 - x_n| & 1 & x_2 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ |x_n - x_1|^2 \log|x_n - x_1| & |x_n - x_1|^2 \log|x_n - x_1| & & |x_n - x_1|^2 \log|x_n - x_1| & 1 & x_n \\ 1 & 1 & \cdots & 1 & 0 & 0 \\ x_1 & x_2 & & x_n & 0 & 0 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ d_0 \\ d_1 \end{bmatrix} = \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \\ 0 \\ 0 \end{bmatrix} \quad (2.14)$$

The $(n+2) \times (n+2)$ matrix on the left hand side of (2.14) is the interpolation matrix for the thin plate splines in one dimension.

2.6 Residual Subsampling Method for One Dimensional Interpolation Problems

Below is a description of the residual subsampling method algorithm developed by Driscoll &Heryudono [7]for one dimensional interpolation problems.

Algorithm 2.1

(i) Generate an initial discretization using n equally spaced points and find the RBF approximation of the function.

(ii) Compute the interpolation error at points halfway between the nodes.

(iii) Points at which the error exceeds a threshold θ_r are to become centres, and centres that lie between two points whose error is below a small threshold θ_c are removed.

(iv) The two end points are always left intact.

(v) The adaptive process follows the solve-estimate-refine/coarsen until a stopping criterion is satisfied.

3. Results and Discussion

In this section, we present the numerical results and discuss our findings obtained from MATLAB programmes in one dimension. To this end, we consider two functions

$f_1(x) = \tanh(60x - 0.01)$and

$f_2(x) = |x + 0.04|$.

We will interpolate all the functions on the interval $[-1,1]$. All the programmes are run in MATLAB 7.14 on Windows XP operating system. We choose our refine threshold to be $\theta_r = 2 \times 10^{-5}$and our coarsen threshold to be $\theta_c = 1 \times 10^{-8}$. We take the values of the shape parameter to be $\varepsilon = 0.25$ and $\varepsilon = 0.50$ when interpolating with the multiquadrics. On the other hand, the thin plate splines does not have a shape parameter. The adaptive radial basis function method is demonstrated by testing two functions $f_1(x)$and$f_2(x)$ using two radial basis functions:multiquadrics and the thin plate splines. In each case, we record the number of iterations It, number of centres N, number of centres to be added N_r, number of centres to be removed $N_c$and the maximum error of interpolation on both the adaptive and uniform grids denoted by $\|\cdot\|_{\infty,\boldsymbol{a}}$ and $\|\cdot\|_{\infty,\boldsymbol{u}}$ respectively. Errors from uniform grids are provided so that they can be compared with errors on the adaptive grid to test the efficiency of the adaptive algorithm.

Example 1
We consider the function $f_1(x)$. The results are recorded in Tables 1-3 and the corresponding graphs in Figures 1-6.

Table 1: Adaptive interpolation with multiquadricsperformed on $f_1(x)$for $\varepsilon = 0.25$.

				$ItNN_rN_c\|\cdot\|_{\infty,a}\|\cdot\|_{\infty,u}$
1	11	10	0	$7.5308 \times 10^{-1} 7.5308 \times 10^{-1}$
2	21	20	0	$5.7700 \times 10^{-1} 5.7700 \times 10^{-1}$
3	41	40	0	$3.2684 \times 10^{-1} 3.2684 \times 10^{-1}$
4	81	41	0	$8.2495 \times 10^{-2}\ 8.2495 \times 10^{-2}$
5	122	30	26	$3.7189 \times 10^{-3} 7.7969 \times 10^{-3}$
6	126	4	45	$1.2531 \times 10^{-4}\ 6.4877 \times 10^{-3}$
7	85	4	0	$1.0287 \times 10^{-2}\ 7.0909 \times 10^{-2}$
8	89	6	0	$3.2040 \times 10^{-3}\ 6.1154 \times 10^{-2}$
9	95	4	0	$2.8030 \times 10^{-4}\ 4.8639 \times 10^{-2}$
10	99	8	0	$5.6683 \times 10^{-5}\ 4.1791 \times 10^{-2}$
11	107	0	0	$9.6826 \times 10^{-6}\ 3.0720 \times 10^{-2}$

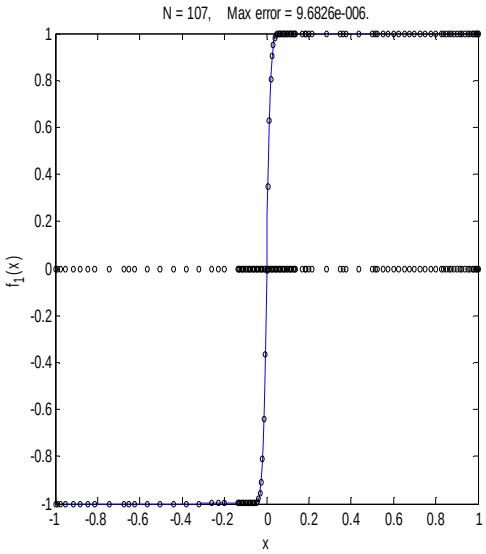

Figure1: Plot of $f_1(x)$with final node distribution on the adapted grid using multiquadrics for $\varepsilon = 0.25$.

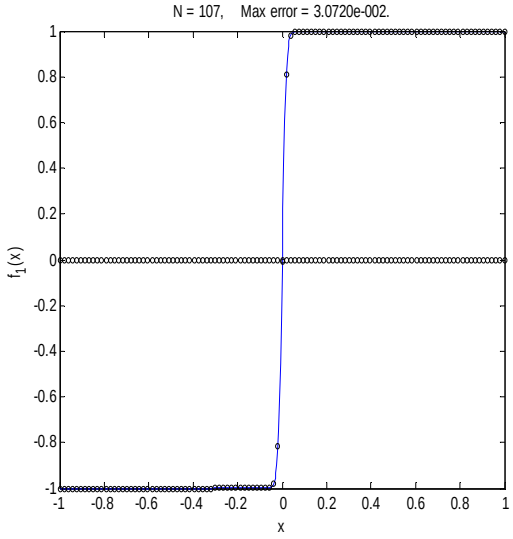

Figure2: Plot of $f_1(x)$with final node distribution on uniform grid using multiquadrics for $\varepsilon = 0.25$.

Table 2 : Adaptive interpolation with multiquadrics performed on $f_1(x)$ for$\varepsilon = 0.50$

				$ItNN_rN_c\|\cdot\|_{\infty,a}\|\cdot\|_{\infty,u}$
1	11	10	0	$7.5411 \times 10^{-1} 7.5411 \times 10^{-1}$
2	21	20	0	$5.8178 \times 10^{-1} 5.8178 \times 10^{-1}$
3	41	28	0	$3.3305 \times 10^{-1} 3.3305 \times 10^{-1}$
4	69	26	6	$8.7240 \times 10^{-2} 1.3324 \times 10^{-1}$
5	89	21	22	$4.5743 \times 10^{-3} 6.5325 \times 10^{-2}$
6	88	1	28	$1.4362 \times 10^{-4} 3.0294 \times 10^{-2}$
7	61	14	0	$4.7110 \times 10^{-4}\ 1.7526 \times 10^{-1}$
8	75	0	7	$1.9772 \times 10^{-5}\ 1.0793 \times 10^{-1}$

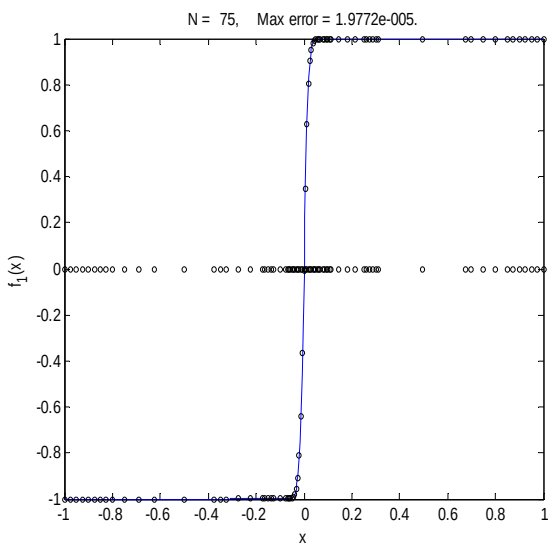

Figure 3: Plot of $f_1(x)$with final node distribution on the adapted grid using multiquadrics for $\varepsilon = 0.50$.

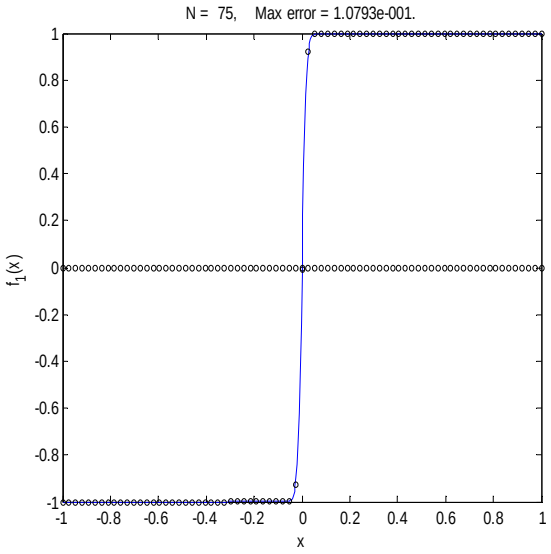

Figure 4: Plot of $f_1(x)$with final node distribution on uniform grid using multiquadrics for $\varepsilon = 0.50$.

Table 3: Adaptive interpolation with thin Plate splines performed on $f_1(x)$

				$ItNN_rN_c\|\|\cdot\|\|_{\infty,a}\|\|\cdot\|\|_{\infty,u}$
1	11	10	0	$7.7744 \times 10^{-1}7.7744 \times 10^{-1}$
2	21	18	0	$6.1737 \times 10^{-1}6.1737 \times 10^{-1}$
3	39	26	0	$3.7889 \times 10^{-1}3.9866 \times 10^{-1}$
4	65	24	0	$1.2494 \times 10^{-1}\ 1.9674 \times 10^{-1}$
5	89	25	0	$1.5759 \times 10^{-2}9.9512 \times 10^{-2}$
6	144	26	0	$1.5247 \times 10^{-3}\ 2.1254 \times 10^{-2}$
7	140	15	0	$1.7443 \times 10^{-4}\ 1.7058 \times 10^{-2}$
8	155	8	0	$2.2736 \times 10^{-5}\ 1.8041 \times 10^{-2}$
9	163	4	0	$1.8612 \times 10^{-5}\ 1.5063 \times 10^{-2}$
10	167	1	0	$1.8696 \times 10^{-5}\ 1.3760 \times 10^{-2}$
11	168	0	0	$1.8696 \times 10^{-5}\ 1.1805 \times 10^{-2}$

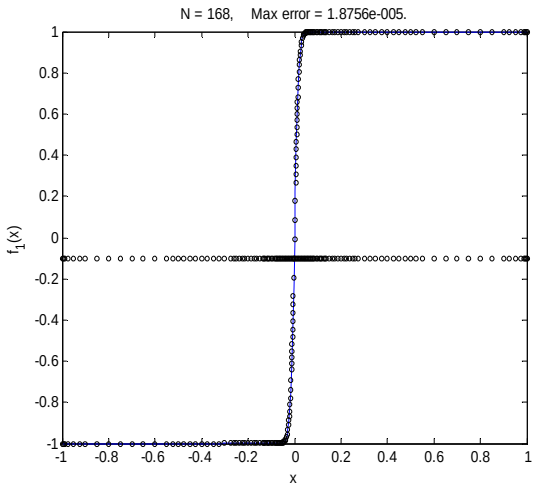

Figure 5: Plot of $f_1(x)$with final node distribution on the adapted grid using thin plate splines.

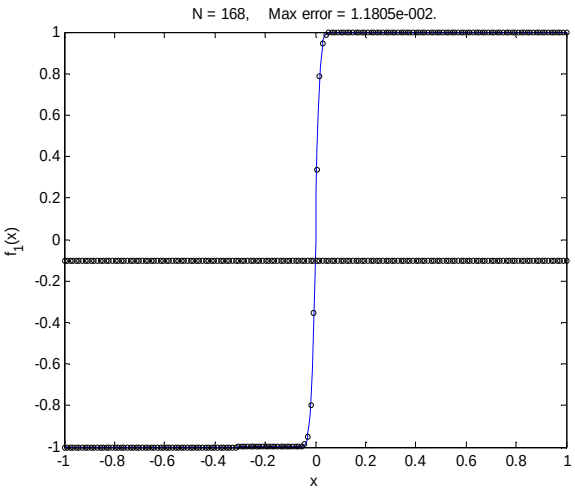

Figure 6: Plot of $f_1(x)$with final node distribution on uniform grid using thin plate splines.

$f_1(x)$has two sharp corner features. We see in Tables 1-3 where the numerical results are recorded that the adaptive interpolation with the multiquadrics and thin plate splines yield good results and show the superiority of the results on adapted grids over that on uniform grids. For example, in Tables 1 and 2, the multiquadrics converged in 11 and 8 iterations for the values of $\varepsilon = 0.25$ and $\varepsilon = 0.50$ and we end up with 107 and 75 centres respectively. The errors on the adapted and uniform grids at the end of the iterations for $\varepsilon = 0.25$ are 9.6826×10^{-6} and 3.0720×10^{-2} respectively, while for $\varepsilon = 0.50$, we have 1.9972×10^{-5} and 1.0793×10^{-1} respectively. Likewise, in Table 3, the adaptive interpolation with the thin plate splines converged in 11 iterations with the final error on adapted and uniform grids as 1.8696×10^{-5} and 1.1805×10^{-2} respectively. Figures 1, 3 and 5 show how nodes are concentrated at the two localized features and consequently making the error on adaptive grids far less than error on the uniform grids.

Example 2

We now consider the function $f_2(x)$. The results are recorded in Tables 4-6 and the corresponding graphs in Figures 7-10.

Table 4: Adaptive interpolation with multiquadrics performed on $f_2(x)$ for $\varepsilon = 0.25$.

It	NN_r	N_c	$\|\cdot\|_{\infty,a}$	$\|\cdot\|_{\infty,u}$
1	11	10	0	3.7599×10^{-2} 3.7599×10^{-2}
2	21	20	0	3.0429×10^{-2} 3.0429×10^{-2}
3	41	30	0	9.3860×10^{-3} 9.3860×10^{-3}
4	71	14	0	7.6060×10^{-3} 8.6926×10^{-3}
5	85	42	0	2.7473×10^{-3} 6.4279×10^{-3}
6	127	19	7	1.9018×10^{-3} 5.0687×10^{-3}
7	139	34	12	8.2866×10^{-3} 3.1699×10^{-3}
8	161	9	24	9.0104×10^{-4} 2.3465×10^{-3}
9	146	85	0	4.7550×10^{0} 4.1964×10^{-3}
10	231	7	53	1.3225×10^{-4} 2.6456×10^{-3}

TABLE 5: Adaptive interpolation with multiquadrics performed $f_2(x)$ for $\varepsilon = 0.50$.

It	NN_r	N_c	$\|\cdot\|_{\infty,a}$	$\|\cdot\|_{\infty,u}$
1	11	10	0	3.7843×10^{-2} 3.7843×10^{-2}
2	21	18	0	3.0780×10^{-2} 3.0780×10^{-2}
3	39	17	0	9.4590×10^{-3} 1.1614×10^{-2}
4	56	13	0	7.6950×10^{-3} 1.1193×10^{-2}
5	69	13	9	2.3647×10^{-3} 8.6072×10^{-3}
6	73	9	19	1.9238×10^{-3} 8.8347×10^{-3}
7	63	11	0	5.9121×10^{-4} 7.1182×10^{-3}
8	74	5	11	4.8094×10^{-4} 4.8000×10^{-3}
9	68	17	0	3.1726×10^{-3} 4.6202×10^{-3}
10	85	3	7	1.2085×10^{-4} 6.4961×10^{-3}
11	81	15	0	1.9537×10^{-4} 7.6950×10^{-3}
12	96	1	2	2.5136×10^{-5} 6.4800×10^{-3}
13	95	0	1	2.5362×10^{-5} 3.2785×10^{-3}

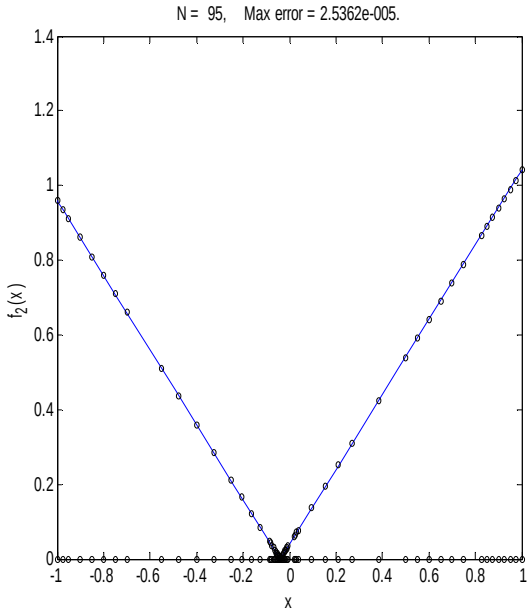

Figure 7: Plot of $f_2(x)$with final node distribution on the adapted grid using multiquadrics for $\varepsilon = 0.50$.

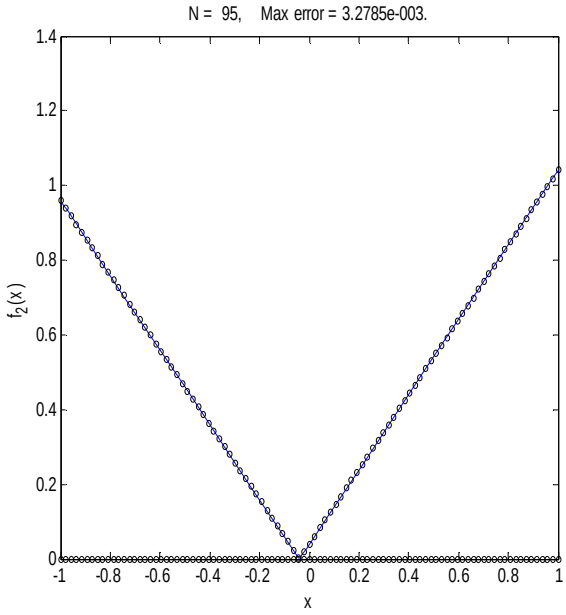

Figure 8: Plot of $f_2(x)$with final node distribution on uniform grid using multiquadric RBF for $\varepsilon = 0.50$.

Table 6: Adaptive interpolation with thin Plate splines Performed on $f_2(x)$

				$ItNN_rN_c\|\cdot\|_{\infty,a}\|\cdot\|_{\infty,u}$
1	11	10	0	$4.2796 \times 10^{-2} 4.2796 \times 10^{-2}$
2	21	17	0	$3.4922 \times 10^{-2} 3.4922 \times 10^{-2}$
3	38	14	0	$1.0694 \times 10^{-2} 1.3521 \times 10^{-2}$
4	52	11	0	$8.7293 \times 10^{-3}\ 1.4371 \times 10^{-2}$
5	63	8	0	$2.6734 \times 10^{-3} 8.0689 \times 10^{-3}$
6	71	8	0	$2.1823 \times 10^{-3}\ \ 9.9764 \times 10^{-3}$
7	79	4	0	$6.6834 \times 10^{-4}\ \ 9.2479 \times 10^{-3}$
8	83	3	0	$5.4558 \times 10^{-4}\ \ 8.1003 \times 10^{-3}$
9	86	2	0	$1.6709 \times 10^{-4}\ \ 5.0322 \times 10^{-3}$
10	88	2	0	$1.3638 \times 10^{-4}\ \ 5.7502 \times 10^{-3}$
11	90	1	0	$4.1785 \times 10^{-5}\ \ 6.3425 \times 10^{-3}$
12	91	0	0	$3.8421 \times 10^{-5}\ \ 4.7526 \times 10^{-3}$

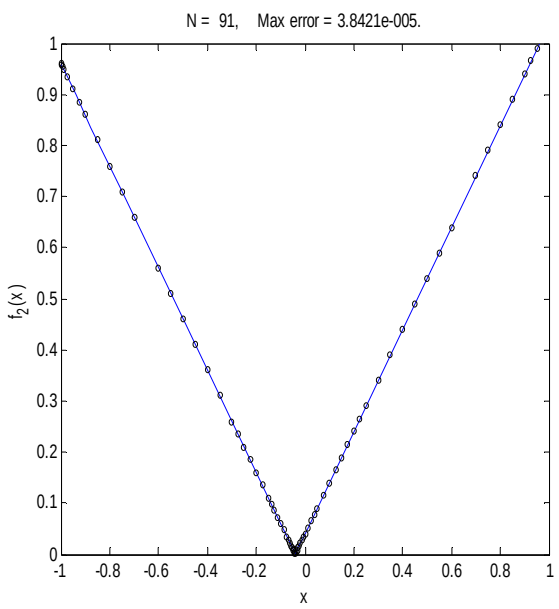

Figure 9: Plot of $f_2(x)$with final node distribution on the adaptive gridusing the thin plate splines.

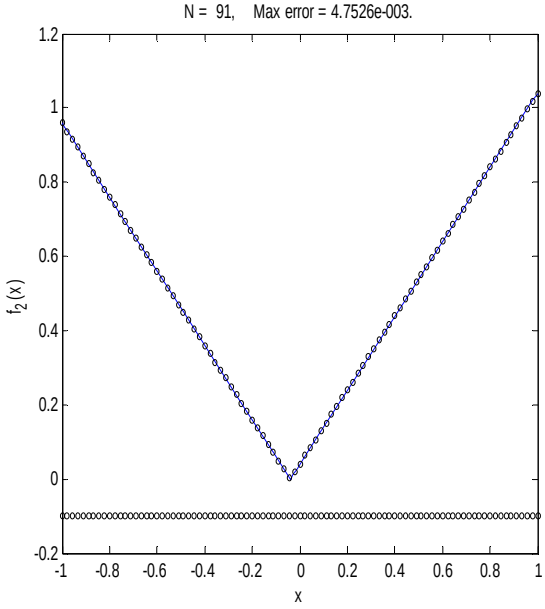

Figure 10: Plot of $f_2(x)$with final node distribution on uniform grid using the thin plate splines.

The results for $f_2(x)$is recorded in Tables 4 - 6. Figures 7 and 9 show that $f_2(x)$ has one sharp localized feature and the adaptive algorithm allocated more centres there. In our numerical experiments for $f_2(x)$, we observed from Table 4 that the adaptive algorithm did not converge with multiquadrics for $\varepsilon = 0.25$. Once again, Tables 5 and 6 show the advantage of interpolation using the adaptive algorithm over interpolation on uniform grid for functions with sharp gradients or localized features. We see that the errors on the adapted grid just like $f_1(x)$ are smaller than errors on the uniform grid. The adaptive interpolation with multiquadrics for the value of $\varepsilon = 0.50$ yields the error 2.5362×10^{-5} in 13 iterations with a total of 95 centres on the adapted grid while on the uniform grid of 95 centres, the error is 3.2785×10^{-2}. Similarly, adaptive interpolation with the thin plate splines gives the errors on the adapted and uniform grids of 91 centres as 3.8421×10^{-5} and 4.7526×10^{-3} respectively.

Table7: Comparison of Errors on Adapted and Uniform Grids for $f_1(x)$ and $f_2(x)$ at the Final Iteration.

		Multiquadric RBF				Thin Plate Splines RBF			
		Final				Final			
Function	ε	It	N	$\|\cdot\|_{\infty,a}$	$\|\cdot\|_{\infty,u}$	It	N	$\|\cdot\|_{\infty,a}$	$\|\cdot\|_{\infty,u}$
$f_1(x)$	*0.25*	11	107	9.6826×10^{-6}	30720×10^{-2}				
$f_1(x)$	*0.50*	8	75	1.9772×10^{-5}	1.0793×10^{-1}				
$f_1(x)$	ε-free					11	168	1.8696×10^{-5}	1.1805×10^{-1}
$f_2(x)$	*0.25*	10	231	1.3325×10^{-4} *	2.6456×10^{-3} *				
$f_2(x)$	*0.50*	13	95	2.5362×10^{-5}	3.2785×10^{-3}				
$f_2(x)$	ε-free					12	91	3.8421×10^{-5}	4.7526×10^{-3}

KEY:* Error message after this iteration: Matrix is close to singular or badly scaled, result may be inaccurate.

Once again, we use Table 7 to compare the adaptive interpolation with multiquadrics and the thin plate splines. We observed that both yield small maximum errors at the end of each iteration for the two functions. We also noted that the adaptive interpolation errors at the end of each iteration for the thin plate splines and the multiquadrics for the value of ε = 0.50 were almost the same. In summary, from Table 7 and Figures 1, 3, 5, 7 and 9 show that the adaptive algorithm performed well with the multiquadrics and the thin plate splines for $f_1(x)$ and $f_2(x)$ that contain sharp corner features. Table 4 shows that the adaptive algorithm did not yield the desired results with the multiquadricsfor ε = 0.25, thus the thin plate splines which is parameter free yielded the best results. The thin plat splines require fewer iterations for $f_2(x)$

4. 0 Conclusion

We implemented an adaptive radial basis functions interpolation algorithm developed in[7] known as residual subsampling method and applied it to the thin plate splines. Our results show that the adaptive radial basis function interpolation yields a better approximation to functions with localized features than interpolation on uniform grid. Also, the thin plate splines which is parameter free yielded better results compared to the multiquadrics which contains a shape parameter and its accuracy depends on the shape parameter. Nevertheless, the interpolation matrix of the thin plate splines needs more computational efforts than that of the multiquadrics and hence its implementation may not be trivial especially in higher dimensions.

References

[1]Buhmann, M. D. (2003). Radial Basis Functions: Theory and Implementations, Cambridge University Press, Cambridge.

[2]Kansa, E. J. (1990). Multiquadrics – A Scattered Data Approximation Scheme with Applications to Computational Fluid Dynamics- II: solution to Parabolic and Elliptic Partial Differential Equations, *Computers and Mathematics with Applications* Vol. 19, No. 8-9, pp. 147-161.

[3]Franke, R. and Schaback, R. (1998). Solving Partial Differential Equations by Collocation using Radial Basis Functions, *Applied Mathematics and Computation*, Vol.1, No. 2, pp. 73 – 82.

[4]Larsson, E. and Fornberg, B. (2003). A Numerical Study of Some Radial Basis Function Based Solution Methods For Elliptic PDEs, *Computational Mathematics with Applications*, Vol. 46, No. 5-6, pp. 891-902.

[5]Cheng, A. H.- D., Golberg, E. Kansa, E. J. and Zammito, G. (2003). Exponential Convergence and $h -$ □ Multiquadric Collection Method for Partial Differential Equations, *NumericalMethods Partial Differential Equations*, Vol. 19, No. 5, pp. 571 – 594.

[6]Buhmann, M. D. and Dyn, N. (1993). Spectral Convergence of Multiquadric Interpolation, *Proceedings of the Edinburgh Mathematical Society*, Vol.2, No. 36, pp. 319 – 333.

[7]Driscoll, T. A. and Heryudono, A. R. H. (2007). Adaptive Residual Subsampling Methods for Radial Basis Function Interpolation and Collocation Problems, *Computer with Mathematics and Applications*, Vol. 53, pp. 927-939.

[8]Behrens, J. and Iske, A. (2002). Grid-free Adaptive Semi-Lagrangian Advection Using Radial Basis Function, *Computational Mathematics with Applications*, Vol.3, No.43, pp. 319-327.

[9]Behrens, J., Iske, A. and Kaser, M. (2003). Adaptive Meshfree Method of Backward Characteristics for Nonlinear Transport Equations: In Meshfree Methods for Partial Differential Equations, (Bonn, 2001), in: Lecture Notes in Computer Science and Engineering, Springer Berlin, Vol. 26, pp. 21 – 36.

[10]Munoz-Gomez, J. A. (2006). Adaptive Node Refinement Collocation Method for Partial Differential Equations, *Computer Science*, pp. 70-80.

[11]Schaback, R. and Wendland, H. (2000). Adaptive Greedy Algorithm Techniques for Approximate Solution of Large RBF Systems, *Numerical Algorithms*, Vol.3, No.32, pp. 239 – 254.

[12]Hon, Y. C., Schaback, R. and Zhou, X. (2003). An Adaptive Greedy Algorithm for Solving Large RBF Collocation Problems, *Numerical Algorithms*, Vol.1, No.32 pp. 13 – 25.

[13]Iske, A. (2003). Radial Basis Functions: Basics Advanced Topics and Meshfree Methods for Transport Problems, *RendicotidelSeminarioMatematica*, Vol. 61 No. 3, pp. 247 – 286.