

# Finding Points of Importance for Radial Basis Function Approximation of Large Scattered Data

Vaclav Skala<sup>[0000-0001-8886-4281]</sup>, Samsul Ariffin Abdul Karim<sup>[0000-0001-6518-6705]</sup> and Martin Cervenka<sup>[0000-0001-9625-1872]</sup>

<sup>1</sup> Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Univerzitni 8, CZ 301 00 Plzen, Czech Republic

<sup>2</sup> Fundamental and Applied Sciences Department and Centre for Smart Grid Energy Research (CSMER), Institute of Autonomous System, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 32610 Seri Iskandar, Perak DR, Malaysia.

<sup>1</sup>skala@kiv.zcu.cz

<sup>2</sup>samsul\_ariffin@utp.edu.my

**Abstract.** Interpolation and approximation methods are used in many fields such as in engineering as well as other disciplines for various scientific discoveries. If the data domain is formed by scattered data, approximation methods may become very complicated as well as time-consuming. Usually, the given data is tessellated by some method, not necessarily the Delaunay triangulation, to produce triangular or tetrahedral meshes. After that approximation methods can be used to produce the surface. However, it is difficult to ensure the continuity and smoothness of the final interpolant along with all adjacent triangles. In this contribution, a meshless approach is proposed by using radial basis functions (RBFs). It is applicable to explicit functions of two variables and it is suitable for all types of scattered data in general. The key point for the RBF approximation is finding the important points that give a good approximation with high precision to the scattered data. Since the compactly supported RBFs (CSRBF) has limited influence in numerical computation, large data sets can be processed efficiently as well as very fast via some efficient algorithm. The main advantage of the RBF is, that it leads to a solution of a system of linear equations (SLE)  $Ax=b$ . Thus any efficient method solves the systems of linear equations that can be used. In this study is we propose a new method of determining the importance points on the scattered data that produces a very good reconstructed surface with higher accuracy while maintaining the smoothness of the surface.

**Keywords:** Meshless Methods, Radial Basis Functions, Approximation.

## 1 Introduction

Interpolation and approximation techniques are used in the solution of many engineering problems. However, the interpolation of unorganized scattered data is still a severe problem. In the one dimensional case, i.e. curves represented as  $y = f(x)$ , it is possible to order points according to the  $x$ -coordinate. However, in a higher dimensionality this is not possible. Therefore, the standard approaches are based on the tessellation of the

domain in  $x, y$  or  $x, y, z$  spaces using, e.g. Delaunay triangulation [7], etc. This approach is applicable for static data and  $t$ -varying data, if data in the time domain are “framed”, i.e. given for specific time samples. It also leads to an increase of dimensionality, i.e. from triangulation in  $E^2$  to triangulation in  $E^3$  or from triangulation in  $E^3$  to triangulation in  $E^4$ , etc. It results in significant increase of the triangulation complexity and complexity of a triangulation algorithm implementation. This is a significant factor influencing computation in the case of large data sets and large range data sets, i.e. when  $x, y, z$  values are spanned over several magnitudes.

On the contrary, meshless interpolations based on Radial Basis Functions (RBF) offer several significant advantages, namely:

- RBF interpolation is applicable generally to  $d$ -dimensional problems and does not require tessellation of the definition domain
- RBF interpolation and approximation is especially convenient for scattered data interpolation, including interpolation of scattered data in time as well
- RBF interpolation is smooth by a definition
- RBF interpolation can be applied for interpolation of scalar fields and vector fields as well, which can be used for scalar and vector fields visualization
- If the Compactly Supported RBFs (CSRBF) are used, sparse matrix data structures can be used which decreases memory requirements significantly.

However, there are some weak points of RBF application in real problems solution:

- there is a real problem for large data sets with robustness and reliability of the RBF application due to high conditionality of the matrix  $\mathbf{A}$  of the system of linear equations, which is to be solved
- numerical stability and representation is to be applied over a large span of  $x, y, z$  values, i.e. if values are spanned over several magnitudes
- problems with memory management as the memory requirements are of  $O(N^2)$  complexity, where  $N$  is a number of points in which values are given
- the computational complexity of a solution of the linear system, which is  $O(N^3)$ , resp.  $O(kN^2)$ , where  $k$  is a number of iteration if the iterative method are used, but  $k$  is relatively high, in general.
- Problems with unexpected behavior at geometrical borders

Many contributions are solving some issues of the RBF interpolation and approximation available. Numerical tests are mostly made using some standard testing functions and restricted domain span, mostly taking interval  $\langle 0,1 \rangle$  or similar. However, in many physically based applications, the span of the domain is higher, usually over several magnitudes and large data sets need to be processed. Also large data sets are to be processed.

As the meshless techniques are easily scalable to higher dimensions and can handle spatial scattered data and spatial-temporal data as well, they can be used in many engineering and economical computations, etc. Polygonal representations (tessellated domains) are used in computer graphics and visualization as a surface representation and for surface rendering. In time-varying objects, a surface is represented as a triangular mesh with constant connectivity.

On the other hand, all polygonal based techniques, in the case of scattered data, require tessellations, e.g. Delaunay triangulation with  $O(N^{\lfloor d/2+1 \rfloor})$  computational

complexity for  $N$  points in  $d$ -dimensional space or another tessellation method. However, the complexity of tessellation algorithms implementation grows significantly with dimensionality and severe problems with robustness might be expected, as well.

In the case of data visualization smooth interpolation or approximation on unstructured meshes is required, e.g. on triangular or tetrahedral meshes, when physical phenomena are associated with points, in general. This is quite a difficult task especially if the smoothness of interpolation is needed. However, it is a natural requirement in physically-based problems.

## 2 MESHLESS INTERPOLATION

Meshless (meshfree) methods are based on the idea of Radial Basis Function (RBF) interpolation [1][2][27][28], which is not separable. RBF based techniques are easily scalable to  $d$ -dimensional space and do not require tessellation of the geometric domain and offer smooth interpolation naturally. In general, meshless techniques lead to a solution of a linear system equations (LS) [4][5] with a full or sparse matrix.

Generally, meshless methods for scattered data can be split into two main groups in computer graphics and visualization:

- “implicit” –  $F(\mathbf{x}) = 0$ , i.e.  $F(x, y, z) = 0$  used in the case of a surface representation in  $E^3$ , e.g. surface reconstruction resulting into an implicit function representation. This problem is originated from the implicit function modeling [18] approach,
- “explicit” –  $F(\mathbf{x}) = h$  used in interpolation or approximation resulting in a functional representation, e.g. a height map in  $E^2$ , i.e.  $h = F(x, y)$ .

where:  $\mathbf{x}$  is a point represented generally in  $d$ -dimensional space, e.g. in the case of 2-dimensional case  $\mathbf{x} = [x, y]^T$  and  $h$  is a scalar value or a vector value.

The RBF interpolation is based on computing of the distance of two points in the  $d$ -dimensional space and it is defined by a function:

$$f(\mathbf{x}) = \sum_{j=1}^M \lambda_j \varphi(\|\mathbf{x} - \mathbf{x}_j\|) = \sum_{j=1}^M \lambda_j \varphi(r_j) \quad (1)$$

where:  $r_j = \|\mathbf{x} - \mathbf{x}_j\|_2 \stackrel{\text{def}}{=} \sqrt{(x - x_j)^2 + (y - y_j)^2}$  (in 2-dimensional case) and  $\lambda_j$  are weights to be computed. Due to some stability issues, usually a polynomial  $P_k(\mathbf{x})$  of a degree  $k$  is added [6]. It means that for the given data set  $\{(\mathbf{x}_i, h_i)\}_1^M$ , where  $h_i$  are associated values to be interpolated and  $\mathbf{x}_i$  are domain coordinates, we obtain a linear system of equations:

$$h_i = f(\mathbf{x}_i) = \sum_{j=1}^M \lambda_j \varphi(\|\mathbf{x}_i - \mathbf{x}_j\|) + P_k(\mathbf{x}_i) \quad i = 1, \dots, M \quad \mathbf{x} = [x, y, 1]^T \quad (2)$$

For a practical use, a polynomial of the 1<sup>st</sup> degree is used, i.e. linear polynomial  $P_1(\mathbf{x}) = \mathbf{a}^T \mathbf{x}$  in many applications. Therefore, the interpolation function has the form:

$$\begin{aligned}
f(\mathbf{x}_i) &= \sum_{j=1}^M \lambda_j \varphi(\|\mathbf{x}_i - \mathbf{x}_j\|) + \mathbf{a}^T \mathbf{x}_i \\
&= \sum_{j=1}^M \lambda_j \varphi_{i,j} + \mathbf{a}^T \mathbf{x}_i
\end{aligned}
\quad h_i = f(\mathbf{x}_i) \quad i = 1, \dots, M \quad (3)$$

and additional conditions are to be applied:

$$\sum_{j=1}^M \lambda_j \mathbf{x}_j = \mathbf{0} \quad \text{i.e.} \quad \sum_{j=1}^M \lambda_j x_j = 0 \quad \sum_{j=1}^M \lambda_j y_j = 0 \quad \sum_{j=1}^M \lambda_j = 0 \quad (4)$$

It can be seen that for the  $d$ -dimensional case a system of  $(M + d + 1)$  linear system has to be solved, where  $M$  is a number of points in the dataset and  $d$  is the dimensionality of data. For  $d = 2$  vectors  $\mathbf{x}_i$  and  $\mathbf{a}$  are in the form  $\mathbf{x}_i = [x_i, y_i, 1]^T$  and  $\mathbf{a} = [a_x, a_y, a_0]^T$ , we can write :

$$\begin{bmatrix}
\varphi_{1,1} & \dots & \varphi_{1,M} & x_1 & y_1 & 1 \\
\vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\
\varphi_{M,1} & \dots & \varphi_{M,M} & x_M & y_M & 1 \\
x_1 & \dots & x_M & 0 & 0 & 0 \\
y_1 & \dots & y_M & 0 & 0 & 0 \\
1 & \dots & 1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\lambda_1 \\
\vdots \\
\lambda_M \\
a_x \\
a_y \\
a_0
\end{bmatrix}
=
\begin{bmatrix}
h_1 \\
\vdots \\
h_M \\
0 \\
0 \\
0
\end{bmatrix} \quad (5)$$

This can be rewritten in the matrix form as:

$$\begin{bmatrix} \mathbf{B} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix} \quad \mathbf{Ax} = \mathbf{b} \quad \mathbf{a}^T \mathbf{x}_i = a_x x_i + a_y y_i + a_0 \quad (6)$$

For the two-dimensional case and  $M$  points given a system of  $(M + 3)$  linear equations has to be solved. If ‘‘global’’ functions, e.g.  $(\varphi(r) = r^2 \lg r)$ , are used, then the matrix  $\mathbf{B}$  is ‘‘full’’, if ‘‘local’’ functions CSRBFs are used, the matrix  $\mathbf{B}$  can be sparse.

The RBF interpolation was originally introduced by Hardy as the multiquadric method in 1971 [5], which was called Radial Basis Function (RBF) method. Since then many different RFB interpolation schemes have been developed with some specific properties, e.g. 4 uses  $\varphi(r) = r^2 \lg r$ , which is called Thin-Plate Spline (TPS), a function  $\varphi(r) = e^{-(\epsilon r)^2}$  was proposed in [28]. However, the shape parameter  $\epsilon$  might leads to an ill-conditioned system of linear equations [33].

The CSRBFs were introduced as:

$$\varphi(r) = \begin{cases} (1-r)^q P(r), & 0 \leq r \leq 1 \\ 0, & r > 1 \end{cases} \quad (7)$$

where:  $P(r)$  is a polynomial function and  $q$  is a parameter. Theoretical problems with numerical stability were solved in [4]. In the case of global functions, the linear system of equations is becoming ill conditioned and problems with convergence can be expected. On the other hand, if the CSRBFs are taken, the matrix  $\mathbf{A}$  is becoming relatively sparse, i.e. computation of the linear system will be faster, but we need to carefully select the scaling factor  $\alpha$  (which can be ‘‘tricky’’) and the final function might tend to be ‘‘blobby’’ shaped, see Table 1 and Fig.1.

The compactly supported RBFs are defined for the ‘‘normalized’’ interval  $r \in (0,1)$ , but for the practical use a scaling is used, i.e. the value  $r$  is multiplied by shape parameter  $\alpha$ , where  $\alpha > 0$ .

**Table 1.** Typical examples of “local” functions – CSRBF (“+” means – value zero out of (0,1))

ID	Function	ID	Function
1	$(1 - r)_+$	6	$(1 - r)_+^6(35r^2 + 18r + 3)$
2	$(1 - r)_+^3(3r + 1)$	7	$(1 - r)_+^8(32r^3 + 25r^2 + 8r + 3)$
3	$(1 - r)_+^5(8r^2 + 5r + 1)$	8	$(1 - r)_+^3$
4	$(1 - r)_+^2$	9	$(1 - r)_+^3(5r + 1)$
5	$(1 - r)_+^4(4r + 1)$	10	$(1 - r)_+^7(16r^2 + 7r + 1)$

Meshless techniques are primarily based on the approaches mentioned above. They are used in engineering problem solutions, nowadays, e.g. partial differential equations, surface modeling, surface reconstruction of scanned objects, reconstruction of corrupted images [26], etc. More generally, meshless object representation is based on specific interpolation or approximation techniques [1][6][28].

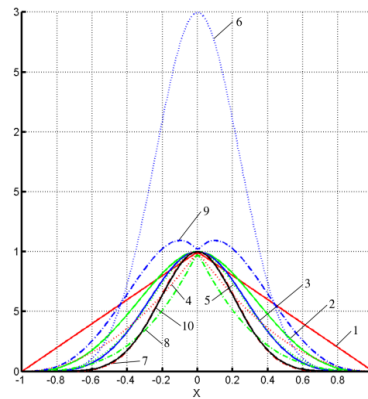
The resulting matrix  $\mathbf{A}$  tends to be large and ill-conditioned. Therefore, some specific numerical methods have to be taken to increase the robustness of a solution, like preconditioning methods or parallel computing on GPU [11][12], etc. In addition, subdivision or hierarchical methods are used to decrease the sizes of computations and increase robustness [18][20][34].

It should be noted, that the *computational complexity* of meshless methods actually covers the complexity of tessellation itself and interpolation and approximation methods. This results in problems with large data set processing, i.e. numerical stability and memory requirements, etc.

If global RBF functions are considered, the RBF matrix is full and in the case of  $10^6$  of points, the RBF matrix is of the size approx.  $10^6 \times 10^6$  ! On the other hand, if CSRBF used, the relevant matrix is sparse and computational and memory requirements are decreased significantly using special data structures [10][12][24][34].

In the case of physical phenomena visualization, data received by simulation, computation or obtained by experiments usually are oversampled in some areas and also numerically more or less precise. It seems possible to apply approximation methods to decrease computational complexity significantly by adding virtual points in the place of interest and use analogy of the least square method modified for the RBF case [21].

Due to the CSRBF representation the space of data can be subdivided, interpolation, resp. the approximation can be split to independent parts and computed more or less independently [24]. This process can be also parallelized and if appropriate computational architecture is used, e.g. GPU, etc. it will lead to faster computation as well. The approach was experimentally verified for scalar and vector data used in the visualization of physical phenomena.



**Fig.1.** Properties of CSRBFs

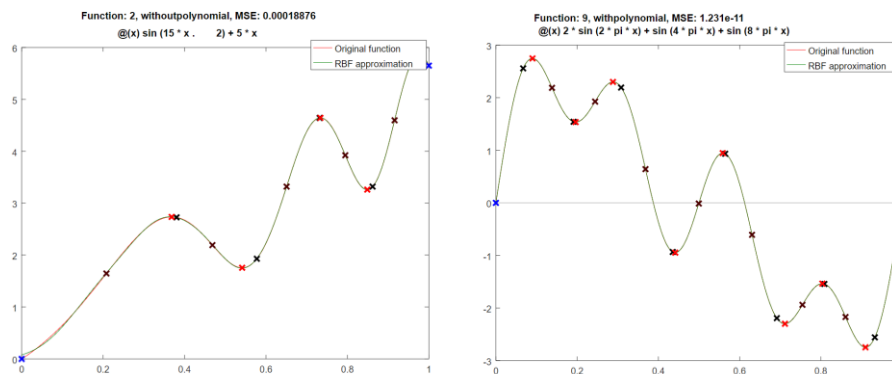
### 3 Points of importance

Algorithms developed recently were based on different specific properties of “global” RBFs or “local” compactly supported RBFs (CS-RBFs) and application areas expected, e.g. for interpolation, approximation, solution of partial differential equations, etc., expecting “reasonable” density of points. However, there are still some important problems to be analyzed and hopefully solved, especially:

- What is an acceptable compromise between the precision of approximation and compression ratio, i.e. reduction of points, if applicable?
- What is the optimal constant shape parameter, if does exist and how to estimate it efficiently ([33])?
- What are optimal shape parameters  $\alpha$  for every single  $\varphi(r, \alpha)$  ([30][33])?
- What is the robustness and stability of the RBF for large data and large range span of data with regard to shape parameters ([20][21])?

In this contribution, we will analyze a specific problem related to the first question.

Let us consider given points of a curve (samples of a signal), described by explicit function  $y = f(x)$ . According to the Nyquist-Shannon theorem, the sampling frequency should be at least double the frequency of the highest frequency of the original signal. The idea is, how “points of importance”, i.e. points of inflection and extrema can be used for smooth precise curve approximation.



**Fig. 2.** Testing functions and resulting approximation based on the points of importance (red points are extrema, black points are additional points of importance)

Let us consider sampled curves in Fig.2, i.e. a signal without noise (the blue points are values at the borders, red are maxima, the black are inflection and added points). It can be seen that the reconstruction based on radial basis functions (RBF) has to pass:

- points at the interval borders
- points at extremes, maxima and minima
- some other important points, like points of inflection etc., and perhaps some additional points of the given data to improve signal reconstruction.

However, there several factors to be considered as well, namely:

- extensibility from 2 D to 3 D for explicit functions of two variables, i.e.  $z = f(x, y)$  and hopefully to higher dimension robustness of computation as given discrete data are given.

For extrema finding, the first derivative  $f'(x)$  is to be replaced by a standard discrete scheme. At the left, resp. right margin, forward, resp. the backward difference is to be used. Inside of the interval, the central difference scheme is recommended, as it also “filters” high frequencies. The simple scheme for the second derivative estimation is shown, too. It can be seen, that this is easily extensible for the 3 D case as well.

$$\begin{aligned} f'(x) &\approx \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} & f'(x) &\approx \frac{f(x_i) - f(x_{i-1}))}{x_i - x_{i-1}} \\ f''(x) &\approx \frac{(x_{i+1}) - f(x_{i-1}))}{2(x_{i+1} - x_{i-1})} & f''(x) &\approx \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{(x_{i+1} - x_i)(x_i - x_{i-1})} \end{aligned} \quad (8)$$

So far, a finding of extrema is a simple task, now. However, due to the discrete data, the extrema is detected by

$$\text{sign}(f(x_{i+1}) - f(x_i)) \neq \text{sign}(f(x_i) - f(x_{i-1})) \quad (9)$$

as we need to detect the change of the sign, only. This increases the robustness of computation as well. The points of inflections rely on a second derivative, i.e.  $f''(x) = 0$ ; a similar condition can be derived from (8).

Now, all the important points, i.e. points at the interval borders, maxima, minima and points of inflection, are detected and found. However, it is necessary to include some more points at the interval borders (at least one on each side) to respect the local behavior of the curve and increase the precision of approximation. It is recommended to include at least one or two points which are closest to the borders to respect a curve behavior at the beginning and end of the interval. Also, if additional points are inserted ideally between extreme and inflection points, the approximation precision increases. Now, the standard RBF interpolation scheme can be applied.

$$\begin{bmatrix} \mathbf{B} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad \mathbf{a}^T \mathbf{x}_i = a_x x_i + a_0 \quad (10)$$

where:  $\mathbf{B}$  represents the RBF submatrix,  $\boldsymbol{\lambda}$  the weights of RBFs,  $\mathbf{P}$  represents points for the polynomial  $\mathbf{a}$  represents coefficients of the polynomial,  $\mathbf{f}$  given function values.

It should be noted, that in the case of scattered data, neighbors for each point are to be found, before the estimation of the derivative is made. In the 2 D case, ordering is possible, in the 3 D case computation is to be made on neighbors found. If the regular sampling in each dimension (along the axis) is given, computation simplifies significantly.

It is necessary to note that the curve reconstruction is at the Nyquist-Shannon theorem boundary and probably limits of the compression were obtained with very low relative error, which is less than 0.1 %. However, we have many more points available and if a higher precision is needed, the approximation based on Least Square Error (LSE) computational scheme with Lagrange multipliers might be used [14]. The RBF methods usually lead to an ill-conditioned system of linear equations [33]. In the case of approximation, it can be partially improved by geometry algebra in projective space [22][23] approach.

## 4 Experimental results

The presented approach was tested on several testing functions used for evaluation of errors, stability, robustness of computation, see Tab.2:

**Table 2.** Examples of testing functions

ID	Function	ID	Function
1	$y = \sin(15x^2 + 5x)$	2	$y = \cos(20x) / 2 + 5x$
3	$y = 50(0.4 \sin(15x^2) + 5x)$	4	$y = \sin(8\pi x)$
5	$y = \sin(6\pi x^2)$	6	$y = \sin(25x + 0.1) / (25x + 0.1)$
7	$y = 2 \sin(2\pi x) + \sin(4\pi x)$	8	$y = 2 \sin(2\pi x) + \sin(4\pi x) + \sin(8\pi x)$
9	$y = 2 \sin(\pi(2x - 1)) + \sin(3\pi(2x - 1/2))$	10	$y = 2 \sin(\pi(1 - 2x)) + \sin(3\pi(2x - 1/2))$
11	$y = 2 \sin(\pi(2x - 1)) + \sin(3\pi(2x - 1/2)) - x$	12	$y = 2 \sin\left(2\pi x - \frac{\pi}{2}\right) + \sin(3\pi(2x - 1/2))$
13	$y = \operatorname{atan}(10x - 5)^3 + \operatorname{atan}(10x - 8)^3 / 2$	14	$y = (4.88x - 1.88) * \sin(4.88x - 1.88)^2 + 1$
15	$y = \exp(10x - 6) * \sin(5x - 2)^3 + (3x - 1)^3$	16	$y = \tanh(9x + 1/2) / 9$

The experiments have also proven, that for large data and data with a large span of data a polynomial  $P_k(x)$  should be  $P_k(x) = a_0$ , i.e.  $k = 0$ , see [20][21].

Selected results of the approximation of some functions are presented at Fig.3. It can be seen, that the proposed approximation based actually on RBF interpolation scheme using points of importance offers good precision of approximation a with good compression ratio. The functions were sampled in 200 points approx. and 10-20 points are actually used for the proposed approximation method.

## 5 Conclusion

This contribution briefly describes a method for efficient RBF approximation of large scattered data based on finding points of importance. This leads to a simple RBF based approximation of data with relatively low error with high compression. The precision of approximation can be increased significantly by covering some additional points. The approach is easily extensible to the 3D case, especially if data are ordered. However, if data are scattered, the neighbor points must be evaluated to find points of importance.

Experiments proved relatively high precision of approximation based on RFB interpolation using found points of importance leading to high data compression as well.

In future, deep analysis of an approximation behavior at the interval borders is expected as it is a critical issue for the 3D case, i.e.  $z = f(x, y)$ , as the first already made experiments shown. Also, the discrete points of curves of inflection are to be taken into account, i.e. discrete points of implicit curves  $F(x, y) = 0$ .



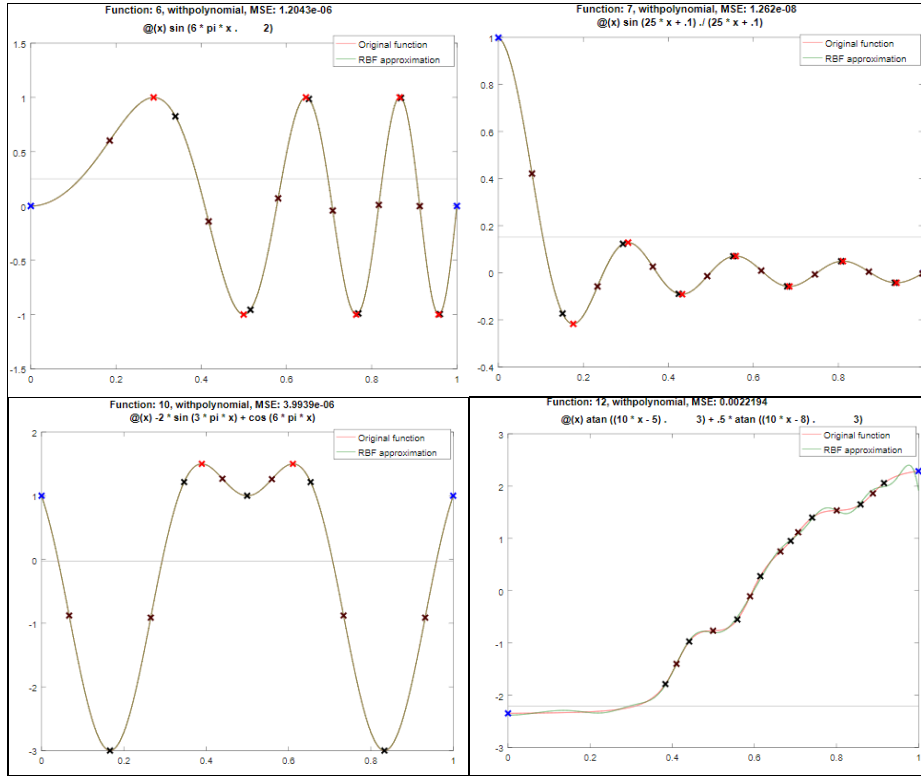


Fig. 3. Examples of approximation for selected functions.

## Acknowledgments

The authors would like to thank their colleagues and students at the University of West Bohemia and Universiti Teknologi PETRONAS for their discussions and suggestions; especially to Michal Smolik, Zuzana Majdisova and Jakub Vasta from the University of West Bohemia. Thanks belong also to anonymous reviewers for their valuable comments and hints provided.

This research was supported by the Czech Science Foundation (GACR) project GA 17-05534S and partially by SGS 2019-016.

## References

1. Biancolini, M.E.: Fast Radial Basis Functions for Engineering Applications. Springer Verlag, DOI:10.1007/978-3-319-75011-8, 2017
2. Buhmann, M.D.: Radial Basis Functions: Theory and Implementations. Cambridge Univ. Press, 2008.
3. Cervenka, M., Smolik, M., Skala, V.: A New Strategy for Scattered Data Approximation Using Radial Basis Functions Representing Points of Inflection, Computational Science and Its Application, ICSSA 2019, Part I, LNCS 11619, pp.322-226, DOI:10.1007/978-3-030-24289-3\_24, Springer, 2019.
4. Duchon, J.: Splines minimizing rotation-invariant semi-norms in Sobolev space. Constructive Theory of Functions of Several Variables, LNCS 571, Springer, 1997.
5. Hardy, L.R.: Multiquadric equation of topography and other irregular surfaces. Journal of Geophysical Research 76 (8), 1905-1915, 1971.
6. Fasshauer, G.E.: Meshfree Approximation Methods with MATLAB. World Scientific Publishing, 2007.
7. Karim, S.A.A., Saaban, A., Skala, V.: Range-restricted Interpolation Using Rational Bi-Cubic Spline Functions with 12 Parameters, SSN:2169-3536, Vol.7, pp.104992-105006, DOI:10.1109/ACCESS.2019.2931454, IEEE, 2019
8. Lazzaro, D., Montefusco, L.B.: Radial Basis functions for multivariate interpolation of large data sets. Journal of Computational and Applied Mathematics, 140, pp. 521-536, DOI:10.1016/S0377-0427(01)00485-X, 2002.
9. Macedo, I., Gois, J.P., Velho, L.: Hermite Interpolation of Implicit Surfaces with Radial Basis Functions. Computer Graphics Forum, Vol.30, No.1, pp.27-42, DOI:10.1109/SIBGRAPI.2009.11. 2011.
10. Majdisova, Z., Skala, V.: A New Radial Basis Function Approximation with Reproduction, CGVCVIP 2016, pp.215-222, ISBN 978-989-8533-52-4, Portugal, 2016
11. Majdisova, Z., Skala, V.: Radial basis function approximations: Comparison and applications. Applied Mathematical Modelling, 51:728-743, DOI:10.1016/j.apm.2017.07.033, 2017.
12. Majdisova, Z., Skala, V.: Big geo data surface approximation using radial basis functions: A comparative study. Computers & Geosciences, 109:51-58, DOI:10.1016/j.cageo.2017.08.007, 2017.
13. Majdisova, Z., Skala, V.: Radial basis function approximations: Comparison and applications. Applied Mathematical Modelling, 51:728-743, DOI:10.1016/j.apm.2017.07.033, 2017.
14. Majdisova, Z., Skala, V., Smolik, M.: Determination of stationary points and their bindings in dataset using RBF methods. CoMeSySo 2018, pp. 213-224. Springer, DOI:10.1007/978-3-030-00211-4\_20, 2018.
15. Majdisova, Z., Skala, V., Smolik, M.: Determination of Reference Points and Variable Shape Parameter for RBF Approximation, Integrated Computer-Aided Engineering, Vol.27, No.1, pp.1-15, ISSN 1069-2509, DOI: 0.3233/ICA-190610, IOS Press, 2020
16. Pan, R., Skala, V.: A two level approach to implicit modeling with compactly supported radial basis functions. Engineering and Computers, Vol.27. No.3, pp.299-307, ISSN 0177-0667, Springer, DOI:10.1007/s00366-010-0199-1, 2011.
17. Pan, R., Skala, V.: Surface Reconstruction with higher-order smoothness. The Visual Computer, Vol.28, No.2., pp.155-162, ISSN 0178-2789, Springer, DOI:10.1007/s00371-011-0604-9, 2012.
18. Ohtake, Y., Belyaev, A., Seidel, H.-P.: A multi-scale approach to 3d scattered data interpolation with compactly supported basis functions. Shape modeling, IEEE, Washington, pp 153-161, DOI:10.1109/SMI.2003.1199611, 2003.

19. Savchenko,V., Pasco,A., Kunev,O., Kunii,T.L.: Function representation of solids reconstructed from scattered surface points & contours. *Computer Graphics Forum*, 14(4), pp.181–188, DOI:10.1111/1467-8659.1440181, 1995.
20. Skala,V.: RBF interpolation with CSRBF of large data sets, *ICCS 2017, Procedia Computer Science*, 108:2433–2437, DOI:10.1016/j.procs.2017.05.081, 2017.
21. Skala,V.: RBF Interpolation and Approximation of Large Span Data Sets, *MCSI 2017 – Corfu, IEEE*, pp. 212-218, DOI:10.1109/MCSI.2017.44, 2018
22. Skala,V., Karim,S.A.A, Kadir,E.A.: Scientific Computing and Computer Graphics with GPU: Application of Projective Geometry and Principle of Duality, *International Journal of Mathematics and Computer Science*, Vol.15, No.3, pp.769-777, ISSN 1814-0432, 2020
23. Skala,V.: High Dimensional and Large Span Data Least Square Error: Numerical Stability and Conditionality, *International Journal of Applied Physics and Mathematics*, Vol.7., No.3., pp.148-156, ISSN 2010-362X, DOI:10.17706/ijapm.2017.7.3.148-156, IAP, California, USA, 2017
24. Smolik,M., Skala,V.: Large scattered data interpolation with radial basis functions and space subdivision. *Integrated Computer-Aided Engineering*, 25(1):49– 26, DOI:10.3233/ICA-170556, 2018.
25. Smolik,M., Skala,V., Majdisova,Z.. Vector field radial basis function approximation. *Advances in Engineering Software*, 123(1):117–129, DOI:10.1016/j.advengsoft.2018.06.013, 2018
26. Uhlir,K., Skala,V: Reconstruction of Damaged Images using Radial Basis Functions, *EUSIPCO 2005 conference proceedings*, ISBN 975-00188-0-X, Turkey, 2005.
27. Wenland,H.: *Scattered Data Approximation*. Cambridge University Press, DOI:10.1017/CBO9780511617539, 2010.
28. Wright,G.B.: *Radial Basis Function Interpolation: Numerical and Analytical Developments*. University of Colorado, Boulder. PhD Thesis,, 2003
29. Zhang,X., Song,K.Z., Lu,M.W., Liu,X.: Meshless methods based on collocation with radial basis functions. *Computational mechanics*, 26(4):333–343, DOI:10.1007/s004660000181, 2000.
30. Skala,V., Karim,S.A.A., Zabran,M.: Radial Basis Function Approximation Optimal Shape Parameters Estimation: Preliminary Experimental Results, accepted to *ICCS 2020 conference*, 2020
31. Vasta,J., Skala,V., Smolik,M., Cervenka,M: Modified Radial Basis Functions Approximation Respecting Data Local Features, pp.445-449, *Informatics 2019, IEEE proceedings*, ISBN 978-1-7281-3178-8, Poprad, Slovakia, 2019
32. Cervenka,M., Smolik,M., Skala,V.: A New Strategy for Scattered Data Approximation Using Radial Basis Functions Representing Points of Inflection, *Computational Science and Its Application, ICSSA 2019 proceedings, Part I, LNCS 11619*, pp.322-226, ISSN 0302-9743, ISBN 978-3-030-24288-6, [https://doi.org/10.1007/978-3-030-24289-3\\_24](https://doi.org/10.1007/978-3-030-24289-3_24), Springer, 2019
33. Cervenka,M., Skala,V.: Conditionality Analysis of the Radial Basis Function Matrix, submitted to *Int. Conf. on Computational Science and Applications ICCSA*, 2020
34. Smolik,M., Skala,V.: Efficient Speed-up of Radial Basis Functions Approximation and Interpolation Formula Evaluation, submitted to *Int. Conf. on Computational Science and Applications ICCSA*, 2020