Journal of Mathematical Extension
Vol. 14, No. 2, (2020), 91-112
ISSN: 1735-8299
URL: http://www.ijmex.com

Performance Evaluation of ORBIT Algorithm to Some Effective Parameters

M. Ahmadvand
Hamedan Branch, Islamic Azad University

M. Esmaeilbeigi*
Malayer University

F. Yaghoobi
Hamedan Branch, Islamic Azad University

A. Kamandi
University of Science and Technology of Mazandaran

Abstract. ORBIT is a derivative-free trust-region framework that employs a radial basis function (RBF) interpolation to solve the computationally expensive optimization problems. The accuracy and stability of RBFs depend on a so-called shape parameter and number of data points. So, it is more appropriate to determine these parameters properly. In this paper, we evaluate the performance of ORBIT algorithm by different types of RBFs, different numbers of data points and different shape parameter values. We utilize Dolan-Moré performance profile and Moré-Wild data profile to investigate the performance of algorithms. Finally, based on this numerical study we propose some recommendations for the type of RBF, the number of data points and the shape parameter value.

AMS Subject Classification: 65D05; 65K10; 90C30; 90C20; 90C56
Keywords and Phrases: Radial basis function, derivative-free optimization, trust-region framework, shape parameter, data points

1. Introduction

Consider the following unconstrained minimization problem

$$\min f(x) \qquad x \in \mathbb{R}^n, \tag{1}$$

Received: June 2018; Accepted: April 2019
*Corresponding author

where $\mathbb{R}^n$ denotes an n-dimensional Euclidean space and $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ is a nonlinear, real-valued, and continuous function that is bounded from below. We are interested in the case that the derivatives of $f(x)$ are not available or are computationally expensive. These kinds of problems occur relatively frequently in the industry when the value of $f(x)$ is obtained from some physical, chemical or econometrical experiment or measurement, see [2, 11].

An important class of optimization algorithms to solve the unconstrained optimization problem (1) work based on the trust-region framework which is a prominent class of iterative methods [23, 8, 10, 19]. In this framework, the objective function is approximated by a model, and this model is minimized in a neighborhood of the current iterate. We regard the neighborhood as a trust-region. The trust-region methods have been used by Levenberg [18] and Marquardt [15] for nonlinear least-squares problems and by Goldfeld et al. [12] for unconstrained optimization. In 1970, trust-region frameworks have been revived by Powell for unconstrained optimization [20].

Some of these trust-region algorithms employ a radial basis function (RBF) model, see for example [22]. One of the efficient algorithms of this class is ORBIT, which was presented in 2008 by Wild et al. [28]. The RBF interpolations are derivative-free, so they are effective when all derivatives of the objective function are unavailable, unreliable or impractical to obtain or to approximate numerically. In ORBIT the RBF models are often considered to interpolate a nonlinear function using fewer function evaluations than required by nonlinear polynomial models used by other techniques.

As we expected, the RBF interpolants can affect the efficiency of the ORBIT algorithm. At the same time, the RBF interpolation depends on the type of RBF and the number of data points. Also, some RBFs used by ORBIT such as Gaussian and Multiquadric depend on a so-called shape parameter. These parameters have a significant effect on the interpolation model and therefore on the algorithm efficiency [1, 29]. Accordingly, it is more efficient to use a proportional shape parameter and a proper number of data points to solve an optimization problem. In this paper, we evaluate the performance of ORBIT algorithm by different types of RBF, different numbers of data points and different shape parameter values. So, based on this numerical study we propose some recommendations to select the appropriate options.

To investigate the performance of algorithms we utilize Dolan-Moré performance profiles [4]. In addition, ORBIT uses a gradient-based convergence test. Therefore, it may never be able to provide a high-accuracy solution for some expensive functions [16]. In order to get solutions with different accuracy, we apply a gradient-free convergence test to ORBIT which is proposed by Elster and Neumaier [5]. Using this convergence test we again evaluate the performance of ORBIT. Here, to show the performance of different solvers we utilize Moré-Wild data profiles [16].

The rest of this paper is organized as follows. Section 2 gives a brief description of RBF models and their properties. We review the framework of ORBIT algorithm in Section 3. In Section 4, we investigate the performance of ORBIT by different options. In addition, preliminarily numerical results and some recommendations are provided in this section. Finally, some conclusions are made in Section 5.

2. Radial Basis Functions

The RBF interpolants are truly meshless and simple enough tools to model smooth and high-dimensional functions [6]. In scattered data interpolation using RBFs, the approximation of a function $f(X)$ at the centers $\chi = \{X_1, \ldots, X_N\}$ may be written as a linear combination of N radial functions. It usually takes the following form:

$$s_{f,\chi}(X) = \sum_{j=1}^{N} \alpha_j \phi(X - X_j) + \sum_{k=1}^{Q} \beta_k p_k(X). \tag{2}$$

Table 1: Some well-known functions that generate RBFs

Name of function	Definition	Smoothness
Cubic	$\phi\,(x) = \|x\|_2^3$	infinitely smooth
Multiquadric (MQ)	$\phi\,(x) = -\sqrt{\|x\|_2^2 + \varepsilon^2}$	infinitely smooth
Inverse Multiquadric (IMQ)	$\phi\,(x) = -\left(\sqrt{\|x\|_2^2 + \varepsilon^2}\right)^{-1}$	infinitely smooth
Gaussian (GA)	$\phi\,(x) = exp(-\frac{\|x\|_2^2}{\varepsilon^2})$	infinitely smooth
Thin plate spline (TPS)	$\phi\,(x) = (-1)^{k+1}\|x\|_2^{2k}\log\|x\|_2\ ,\ k \in \mathbb{N}$	piecewise smooth
Conical spline	$\phi\,(x) = \|x\|_2^{2k+1},\ k \in \mathbb{N}$	infinitely smooth

$\pi_{m-1}(\mathbb{R}^d)$ corresponds to the linear space of d variables polynomials provided that their total degree is less than or equal to $m-1$. It should be noted that $\{p_1, \ldots, p_Q\}$ denotes a basis for polynomial space $\pi_{m-1}(\mathbb{R}^d)$, $X = (x_1, x_2, \ldots, x_d)$ and d is the dimension of problem. α's and β's are coefficients that should be determined and ϕ is an RBF. Some well-known RBFs are listed in Table 1.

To cope with additional degrees of freedom, the interpolation conditions

$$s_{f,\mathcal{X}}(X_j) = f(X_j), \qquad 1 \leqslant j \leqslant N, \tag{3}$$

are completed by the additional conditions

$$\sum_{j=1}^{N} \alpha_j p_k(X_j) = 0, \qquad 1 \leqslant k \leqslant Q. \tag{4}$$

Solvability of this system is therefore equivalent to solvability of the system:

$$\begin{pmatrix} A_{\phi,\mathcal{X}} & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{b} \end{pmatrix} = \begin{pmatrix} f\,|_{\mathcal{X}} \\ 0 \end{pmatrix}, \tag{5}$$

where $A_{\phi,\mathcal{X}} = (\phi(X_j - X_k)) \in \mathbb{R}^{N\times N}$, $P = (p_k(X_j)) \in \mathbb{R}^{N\times Q}$, $f\,|_{\mathcal{X}} = (f(X_j)) \in \mathbb{R}^N$, $\mathbf{c} = [c_1, ..., c_N]^T \in \mathbb{R}^N$ and $\mathbf{b} = [\beta_1, \beta_2, .., \beta_Q]^T$. Also, P^T denotes the transpose of matrix P. This last system is obviously solvable if the coefficient matrix on the left-hand side is invertible [26].

Definition 2.1. *A radial basis function ϕ is called positive definite if*

$$\sum_{j=1}^{N}\sum_{k=1}^{N} c_j c_k \phi(X_j - X_k) > 0$$

for any N pairwise distinct points $X_1, ..., X_N \in \mathbb{R}^d$, and for all nonzero $C = [c_1, ..., c_N]^T \in \mathbb{R}^N$. In addition, ϕ is called conditionally positive definite of order m on $\mathbb{R}^d$, if $\sum_{j=1}^{N} c_j p(X_j) = 0$ for any polynomial p of degree at most $m - 1$.

Equation (2) can be written without the polynomial $\sum_{k=1}^{Q} \beta_k p_k(X)$. In that case, to ensure the solvability of the resulted system, ϕ must be *unconditionally positive definite* i.e.,

$$C^T A_{\phi,\chi} C > 0, \tag{6}$$

(e.g. ϕ can be Gaussian or Inverse multiquadric). However $\sum_{k=1}^{Q} \beta_k p_k(X)$ is required when ϕ is conditionally positive definite. For instance, suppose that ϕ belongs to the Thin plate splines.

Definition 2.2. *If $\phi : \mathbb{R}^d \longrightarrow \mathbb{R}$ is a radial basis function then support of ϕ is defined as*

$$supp(\phi) = \overline{\{X \in \mathbb{R}^d; \phi(X) \neq 0\}}.$$

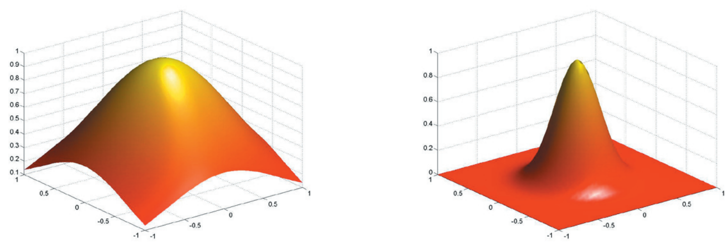
If the $supp(\phi)$ is a compact set then ϕ is called compactly supported RBF (CSRBF). Otherwise, ϕ is called globally supported RBF.

It must be noted that radial basis functions that are conditionally positive definite of order one (such as Multiquadric) can be used without appending the constant term to solve the scattered data interpolation problem [7]. Moreover, since positive definite or conditionally positive definite functions are usually globally supported, the interpolation matrix obtained by some RBFs is full and may be very ill-conditioned.

Although for improving the conditioning of the system of collocation equations CSRBFs have been applied, the CSRBFs vanish beyond an user's defined threshold distance σ. Therefore, only the entries in the collocation matrix corresponding to collocation nodes lying closer than σ to a given CSRBF center are nonzero, and thus lead to a sparse matrix. In fact, the interest in CSRBFs waning as it became evident that, in order to obtain a good accuracy, the overlap distance σ should cover most nodes in the points set, results in a populated matrix again [14].

The accuracy and stability of infinitely smooth $\phi(x)$ depend on the number of data points p and the shape parameter value ε [29]. For a fixed ε, as the number of data points increases, the RBF interpolation converges to a sufficiently smooth function. For a fixed number of data points, Madych [13] has shown that the accuracy of RBF interpolant can often be significantly improved by increasing the shape parameter value.

However, increasing the shape parameter value or increasing the number of data points has a severe effect on the stability of linear system (5). For a fixed shape parameter the condition number of the matrix in the linear system grows exponentially as the number of data points increases and for a fixed number of data points, as the shape parameter becomes large the condition number of the linear system grows and therefore causes the function to become flatter, so its like a length scale [17, 24]. A smaller ε leads to a more peaked RBF model. For example, see Figure 1.



(a) $\varepsilon = 1$ **(b)** $\varepsilon = \frac{1}{3}$

Figure 1. A Gaussian model using different shape parameters

Note that some other types of RBFs such as Cubic and Thin plate spline are shape-parameter-free. The advantage of using this category of RBFs is that we are not more worried about choosing ε. Anyway in using of RBFs, three important questions arise. Which radial basis function ϕ should be used? which ε is the adequate shape parameter value? and which p is the proper number of data points?

There are several strategies to answer these questions. For example, a first attempt at providing guidelines for the selection of appropriate basic functions can be found in [25]. Also, some works has been done in response to choose the appropriate shape parameter ε, see [9, 3].

If we already know $f(x)$, the simplest strategy to search the appropriate value for the shape parameter (ε_{opt}) and the proper number of data points (p_{opt}) is to perform a series of interpolation experiments with different ε and p, and then to pick the best one. In this paper we utilize this simple strategy to determine the appropriate options.

3. The ORBIT Algorithm

In this section we review the ORBIT algorithm. To this aim, we need the following definitions and assumptions which are mostly borrowed from [28].

Definition 3.1. *A collection of vectors* $\{v_i\}_{i\in I}$ *is called affinely dependent if there exists a collection* $v_1, ..., v_k$ *and scalars* a_i *such that* $\sum a_i = 0$ *and* a_i *are not all* 0*, such that*

$$\sum_1^k a_i.v_i = 0;$$

it is called affinely independent if no such scalars a_i *exist.*

Definition 3.2. *For a (center, radius) pair* $(x_k, \Delta_k > 0)$ *the trust-region is defined as,*

$$\mathcal{B}_k = \{x : \|x - x_k\|_k \leqslant \Delta_k\}, \tag{7}$$

where $\|.\|_k$ *at iteration* k *can be the standard 2-norm or other norms.*

Definition 3.3. *In all of the trust-region algorithms at the iteration* k *we should solve the minimization problem,*

$$\min_s\{m_k(x_k + s) : x_k + s \in \mathcal{B}_k\}, \tag{8}$$

that is so-called trust-region subproblem.

Given an approximate solution s_k to (8), the pair (x_k, Δ_k) is updated according to the ratio,

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}. \tag{9}$$

Remark 3.4. *In the framework of ORBIT if $\rho_k \geqslant \eta_1$, then the new iterate is accepted and the trust-region radius is retained or increased. We call such iterations successful.*

Definition 3.5. *Suppose that $f \in C^1[\mathcal{B}_k]$. For fixed $\kappa_f, \kappa_g > 0$, a model $m \in C^1[\mathcal{B}_k]$ is said to be fully linear on $\mathcal{B}_k$ if for all $x \in \mathcal{B}_k$,*

$$|f(x) - m(x)| \leqslant \kappa_f \Delta^2 \tag{10}$$

and

$$\|\nabla f(x) - \nabla m(x)\| \leqslant \kappa_g \Delta. \tag{11}$$

In order to avoid geometric conditions and forming the model in only $n+1$ function evaluations, ORBIT rely on fully linear interpolation models. This definition ensures that first-order Taylor-like bounds exist for the model within the compact neighborhood $\mathcal{B}_k$.

It is enough for the purpose of global convergence to find an approximate solution s_k, we consider a sufficient reduction in the model as follows.

Assumption 3.6. *For the model m_k which $\nabla^2 m_k \neq 0$ and $\|\nabla^2 m_k\| \leqslant \kappa_H$, and for some prespecified constant $\kappa_d \in (0,1)$, we assume that the trust-region step s_k satisfies a sufficient decrease condition,*

$$m_k(x_k) - m_k(x_k + s_k) \geqslant \frac{\kappa_d}{2} \|\nabla m_k(x_k)\| min \left\{ \frac{\|\nabla m_k(x_k)\|}{\kappa_H}, \frac{\|\nabla m_k(x_k)\|}{\|\nabla m_k(x_k)\|_k} \Delta_k \right\}. \tag{12}$$

Considering mentioned the definitions and assumptions, now we can present the iteration k of ORBIT algorithm. Given inputs $\kappa_f, \kappa_g, \epsilon_g > 0$, $0 \leqslant \eta_0 \leqslant \eta_1 < 1$, $0 < \gamma_0 < 1 < \gamma_1$, $0 < \Delta_0 \leqslant \Delta_{max}$, initial point $x_0 \in \mathbb{R}^n$ and the number of points $p > n+1$ to obtain a well-conditioned nonlinear RBF model. An outline of the kth iteration of the ORBIT algorithm is provided in Algorithm 1.

Step1: Find $n+1$ affinely independent points for obtaining fully linear models.

Step2: Add up to $p-n-1$ additional points to our interpolation set in order to take advantage of the nonlinear benefits of RBFs.

Step3: Obtain RBF model.

Step4: While $\|\nabla m_k(x_k)\| \leqslant \frac{\epsilon_g}{2}$:
If m_k is fully linear in $\mathcal{B}_k{}^g = \{x \in \mathbb{R}^n : \|x_k - x\|_k \leqslant \frac{\epsilon_g}{2\kappa_g}\}$, return.
Else, Obtain a model m_k that is fully linear in $\mathcal{B}_k{}^g$,
Set $\Delta_k = \frac{\epsilon_g}{2\kappa_g}$.

Step5: Approximately solve trust-region sub problem (8) to obtain a step s_k satisfying in sufficient decrease condition,
Evaluate $f(x_k + s_k)$ and ρ_k by (9).

Step6: Update trust-region parameters:

$$\triangle_{k+1} = \begin{cases} min\{\gamma_1\Delta_k, \Delta_{max}\}, & \text{if } \rho_k \geqslant \eta_1; \\ \Delta_k, & \text{if } \rho_k < \eta_1 \text{ and } \ m_k \text{ is not fully linear on } \mathcal{B}_k, \\ \gamma_0\Delta_k & \text{if } \rho_k < \eta_1 \text{ and } \ m_k \text{ is fully linear on } \mathcal{B}_k. \end{cases}$$

$$x_{k+1} = \begin{cases} x_k + s_k, & \text{if } \rho_k \geqslant \eta_1; \\ x_k + s_k, & \text{if } \rho_k \geqslant \eta_0 \text{ and } m_k \text{ is fully linear on } \mathcal{B}_k, \\ x_k, & \text{else.} \end{cases}$$

Step7: Evaluate a model-improving point if $\rho_k \geqslant \eta_0$ and m_k is not fully linear on $\mathcal{B}_k$.

Remark 3.7. *For more detail about **Step1** and **Step2**, see algorithms AffPoints and AddPoints respectively in [28].*

Remark 3.8. *If the model is not fully linear, the function is evaluated at an additional so-called model-improving point in **Step 7**, to ensure that the model is at least one step closer to being fully linear on $\mathcal{B}_{k+1} = \mathcal{B}_k$.*

It should be noted that, the global convergence of ORBIT algorithm follows the global convergence of fully linear derivative-free trust-region algorithms, by taking into account the necessary changes in the intermediate lemmas [27].

4. Numerical Results and Analysis

In this section, we investigate the performance of ORBIT with different types of RBFs, different numbers of data points and different shape parameter values. To save the convergence conditions described in [27], we focus on a specific set

$$\mathcal{M} = \{Cubic, Gaussian, Multiquadric\} \tag{13}$$

of RBF interpolants, and try to perform a series of interpolation experiments with different shape parameter values and different numbers of data points, and then try to pick the best ε, p. Our experiments are listed on a set of 60 nonlinear unconstrained problems from [2] with different dimensions ranging from 3 to 12 in Table 2.

Table 2: Test problems

Test problem	Dim	Test problem	Dim
Generalized Rosenbrock	4	Dixmaanb	6
Extended White	6	Dixmaanc	3
Perturbed Quadratic	5	Dixmaand	6
Diagonal4	6	Dixmaane	3
Generalized White	10	Dixmaanf	9
Extended BD1	8	Dixmaang	12
Perturbed Quadratic Diagonal	7	Dixmaanh	9
Extended Wood	8	Dixmaani	9
Extended Hieberd	6	Dixmaanj	3
Fletchcr	3	Dixmaank	3
Nondia	3	Dixmaanl	3
Nondquar	9	Fletcbv3	5
Dqdrtic	9	Bdqrtic	8
Almrost Pertubed Quadratic	6	Tridia	4
Perturbed Tridiagonal Quadratic	8	Arglinb	7
Staircase1	4	Arwhead	5
Stair case2	4	BG2	4
Liarwhd1	3	Engval1	6
Power	9	Cragglvy	4
Cube	7	Edensch	5
Nonscomp	11	Indef	4
Vardim	4	Explin1	3
Quartc	3	Explin2	3
Sinquad	5	Arglinc	5
Liarwhd2	6	Bdexp	7
Dixon3dq	8	Harkerp2	7
Generalized Quadratic	6	Genhumps	8
Himmelbg	8	Mccormck	4
Partial Perturbed Quadratic	4	Extended Denschnb	4
Dixmaana	6	Extended Denschnf	6

All tests are performed at standard starting points on a laptop(Intel Core i5 2.60GHz, 4GRAM) under Windows 7 Ultimate and the MATLAB compiler (version 8.2) with default options.

We obtained the last version of ORBIT for box constrained optimization at `www.mcs.anl.gov/~wild/orbit` and modified it for unconstrained optimization. It should be noted here that we used the trust-region 2-norm, $\Delta_{max} = 50$, $\epsilon_g = 10^{-5}$, $\eta_0 = 0$, $\eta_1 = 0.2$, $\gamma_0 = 0.5$, $\gamma_1 = 2$ and $\kappa_g = 10^{-3}$. Moreover, we declare that an algorithm is failed when the number of iterations exceeds 600. The other parameters have been set as in ORBIT.

As the Cubic is a shape-parameter-free interpolant, here we currently choose the RBFs from the set:

$$\mathcal{M}' = \{Gaussian, Multiquadric\}. \tag{14}$$

To our knowledge, using a large shape parameter value causes the surrogate model to be very flat and this causes the algorithm to lose some minimas. Therefore, we select the shape parameter value from the reasonable set:

$$\mathcal{H} = \{.5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30\}. \tag{15}$$

Also, we investigate four different numbers of data points, one with $p = 2n+1$ which is recommended by Powell for NEWUOA [21], another with $p = n + 2$ which is the minimum number of data points, still another with $p = \frac{(n+1)(n+2)}{2}$ which is the maximum number of data points being interpolated corresponds to the number of points needed to uniquely fit an interpolating quadratic model and the other with $p = 3n$ which is a new recommendation. Therefore, we select the number of data points from the set,

$$\mathcal{P} = \{n+2, 2n+1, 3n, \frac{(n+1)(n+2)}{2}\}. \tag{16}$$

Tables 3, 4 contain average number of function evaluations (ANF) and average minimum of function values (AFmin) for 60 tests using Gaussian and Multiquadric respectively. Each element of these tables corresponds to pair (ε, p), where $\varepsilon \in \mathcal{H}$ and $p \in \mathcal{P}$.

Table 3: Average number of function evaluations and average minimum of function values using Gaussian

	ANF				AFmin			
ε	$n+2$	$2n+1$	$3n$	$\frac{(n+1)(n+2)}{2}$	$n+2$	$2n+1$	$3n$	$\frac{(n+1)(n+2)}{2}$
0.5	279	277	273	242	2085.73	2249.28	2077.67	2032.31
1	266	266	261	242	2102.94	2064.77	2041.56	2029.70
2	254	245	237	187	2117.83	2084.66	2052.94	2058.00
3	253	237	234	208	2100.13	2035.32	2172.95	2156.02
4	250	238	222	188	2078.75	2068.64	2072.25	2087.44
5	247	237	238	223	2121.35	2022.02	2068.38	2102.45
6	246	236	235	215	2151.73	2141.51	2146.62	2134.72
7	242	237	223	208	2070.81	2096.06	2002.92	2101.34
8	242	232	228	231	2098.87	2069.52	2091.77	2105.34
9	242	221	222	219	2118.78	2090.70	2042.07	2090.98
10	244	223	227	238	2078.45	2056.83	2086.05	2086.38
15	239	219	231	230	2080.83	2023.41	2045.13	2092.81
20	247	220	230	231	1994.88	2099.52	2059.40	2065.21
25	243	218	220	244	2086.02	2087.99	2104.16	2238.40
30	239	215	238	245	2163.00	2099.82	2052.67	2137.17

Table 4: Average number of function evaluations and average minimum of function values using Multiquadric

	ANF				AFmin			
ε	$n+2$	$2n+1$	$3n$	$\frac{(n+1)(n+2)}{2}$	$n+2$	$2n+1$	$3n$	$\frac{(n+1)(n+2)}{2}$
0.5	264	247	250	227	2090.32	2043.44	2023.44	2037.36
1	245	245	237	219	2161.19	2070.93	2109.40	2070.32
2	249	234	229	220	2024.68	2060.58	2084.82	2068.89
3	249	234	235	230	2137.28	2059.74	2094.29	2087.07
4	242	226	222	225	2103.10	2080.95	2106.97	2095.36
5	243	225	227	224	2119.93	2072.40	2038.46	2040.28
6	236	227	223	224	2147.79	2104.37	2077.85	2113.39
7	245	222	223	220	2123.82	2078.61	2068.38	2143.63
8	238	230	224	239	2052.08	2124.81	2008.79	2069.55
9	241	222	221	218	2086.31	2132.98	2121.05	2093.46
10	243	226	223	241	2156.46	2087.06	2081.58	2109.71
15	243	223	225	230	2196.49	2055.86	2089.17	2136.16
20	240	221	227	242	2010.92	2096.60	2077.81	2068.88
25	245	223	231	239	2011.03	2126.26	2191.40	2115.36
30	240	221	229	245	2207.23	2117.72	2042.78	2143.71

Table 5: The best and worst (ε, p)

	Gaussian		Multiquadric	
	ANF	AFmin	ANF	AFmin
Best	$(2, \frac{(n+1)(n+2)}{2})$	$(20, n+2)$	$(9, \frac{(n+1)(n+2)}{2})$	$(8, 3n)$
Second	$(4, \frac{(n+1)(n+2)}{2})$	$(7, 3n)$	$(1, \frac{(n+1)(n+2)}{2})$	$(20, n+2)$
Third	$(7, \frac{(n+1)(n+2)}{2})$	$(1, \frac{(n+1)(n+2)}{2})$	$(2, \frac{(n+1)(n+2)}{2})$	$(25, n+2)$
Worst	$(.5, n+2)$	$(25, \frac{(n+1)(n+2)}{2})$	$(.5, n+2)$	$(15, n+2)$

In Table 5 we listed the three best and the worst ANF and AFmin. If we consider the success of an algorithm to have a fewer number of function evaluations then it is easy to see $p = \frac{(n+1)(n+2)}{2}$ is the best number of data points.

Also, Figures 2 and 4 represent ANF for Gaussian and Multiquadric respectively and Figures 3 and 5 show AFmin for Gaussian and Multiquadric respectively. Based on these figures the reported results in Tables 3 and 4 can easily be interpreted. These results are presented as some recommendations in Sub Sect. 4.1.

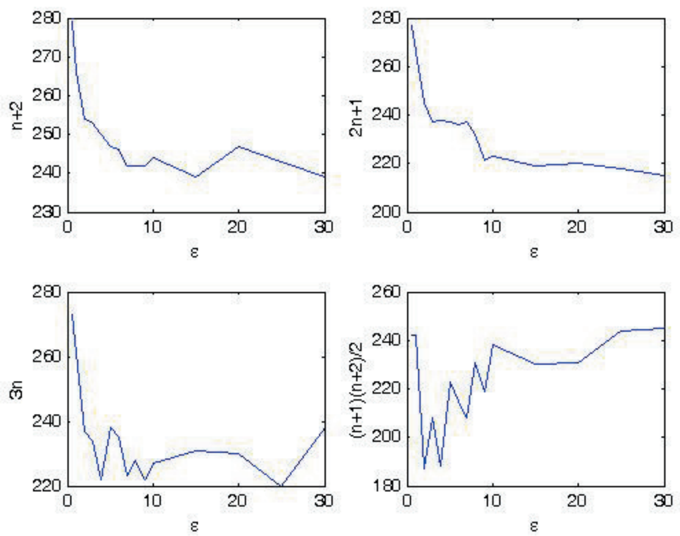


Figure 2. ANF corresponding to Gaussian

We also show the performance of ORBIT using Cubic for 60 tests of Table **??** with different numbers of data points by Dolan-Moré [4] technique. In this technique, we can choose a performance index as measure of comparison among the considered algorithms and can illustrate the results with a performance profile. Having a set $\mathcal{K}$ of benchmark problems and a set $\mathcal{S}$ of optimization algorithms the *performance profile* of algorithm $s \in \mathcal{S}$ is defined as follows:

$$\rho_s(\alpha) = \frac{1}{|\mathcal{K}|}\text{size}\{k \in \mathcal{K} : r_{k,s} \leqslant \alpha\}, \tag{17}$$

where $|\mathcal{K}|$ is the cardinality of $\mathcal{K}$ and $r_{k,s}$ denotes the performance ratio, and is defined by

$$r_{k,s} = \frac{t_{k,s}}{\min\{t_{k,s} : s \in \mathcal{S}\}}, \tag{18}$$

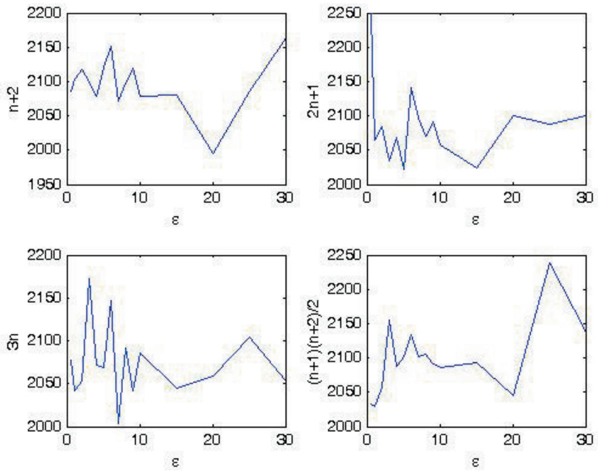


Figure 3. AFmin corresponding to Gaussian

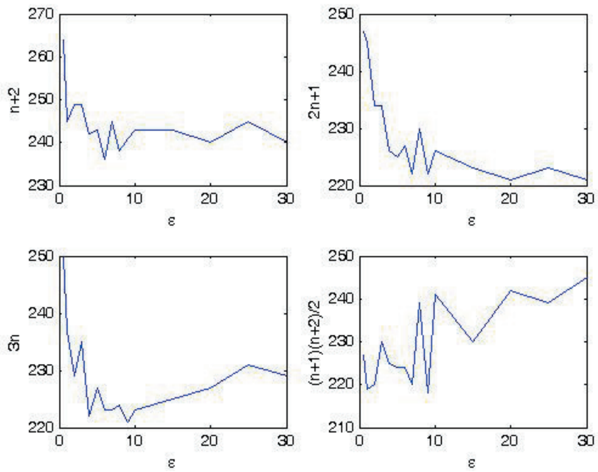


Figure 4. ANF corresponding to Multiquadric

where $t_{k,s} > 0$ is a performance measure resulted for $k \in \mathcal{K}$ and $s \in \mathcal{S}$.

This measure includes items such as the number of function evaluations or the number of iterations. $\rho_s(1)$ denotes to the fraction of problems solved by $s \in \mathcal{S}$ perfoms the best and for α sufficiently large, $\rho_s(\alpha)$ denotes the fraction of problems solved by $s \in \mathcal{S}$. So, we can conclude that a performance profile is the probability distribution for the ratio $r_{k,s}$, and in general the algorithms with high values for $\rho_s(\alpha)$ are preferable [16].

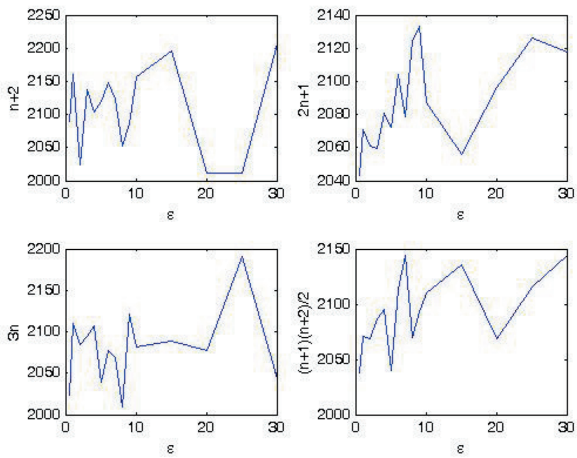


Figure 5. AFmin corresponding to Multiquadric

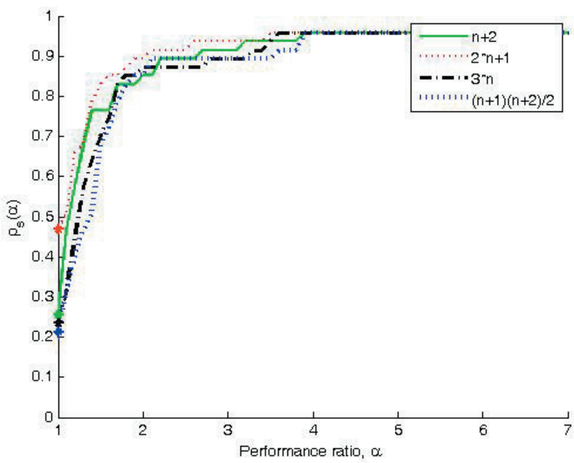


Figure 6. Performance profiles for the number of function evaluations using Cubic with different numbers of data points

Figure 6 shows this process for the number of function evaluations by Cubic with different numbers of data points. From this figure, $p = 2n+1$ is the best number of data points.

We eventually compare Gaussian, Multiquadric and Cubic. For this purpose we equipped Gaussian and Multiquadric with the best (ε, p) and Cubic with the best p, so we utilize the Dolan-Moré technique. It was noted that the shape-parameter-based RBFs Gaussian and Multiquadric are more successful, see Figure 7.

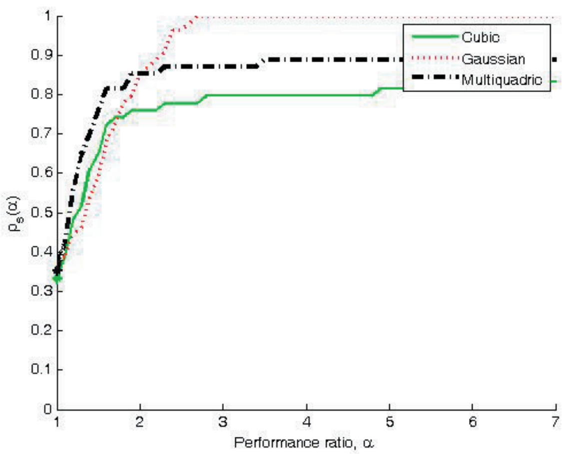


Figure 7. Performance profiles for the number of function evaluations

In addition, to get solutions with different accuracy we apply the gradient-free convergence test,

$$f(x_0) - f(x) \geqslant (1-\tau)(f(x_0) - f_L) \tag{19}$$

to ORBIT. Where $\tau > 0$ is a tolerance, x_0 is the starting point, and f_L is the best function value found by any of the solvers used to solve f. In other words, we obtained the minimum values at each problem by applying only a limit on the number of function evaluations at all the algorithms. Then we set the better minimum obtained by the algorithms as f_L. The convergence test (19) was used by Elster and Neumaier [5]. The tolerance $\tau \in [0, 1]$ refers to the percentage decrease from the starting value $f(x_0)$. By decreasing τ the value obtained for $f(x)$ is closer to f_L, and so we can hope to get a high-accuracy solution.

Here, applying the new convergence test (19) we investigate the performance of ORBIT by Cubic, Gaussian and Multiquadric models. For this purpose using the convergence test (19) with $\tau = 10^{-5}$ we consider the 60 test problems of Table 2 and calculate the number of function evaluations. To compare the RBFs we utilize Moré-Wild *data profile* [16] as follows,

$$d_s(\kappa) = \frac{1}{|\mathcal{K}|}\text{size}\{k \in \mathcal{K} : \frac{t_{k,s}}{n_k + 1} \leqslant \kappa\}, \tag{20}$$

where for $k \in \mathcal{K}$ and $s \in \mathcal{S}$, $t_{k,s}$ is the number of function evaluations required to satisfy (19) for a given τ and n_k is the number of variables in k. In this definition we can interpret κ as the number of *simplex gradient estimates*, so $d_s(\kappa)$ denotes to the percentage of problems that can be solved by the solver s with the equivalent of κ simplex gradient estimates, corresponding to $\kappa(n+1)$ function evaluations [16]. Setting $\kappa \in [0, 100]$ Figure 8 illustrates the data profiles for different RBFs. According to this figure, for example if we have a budget of 50 gradients then ORBIT can solve over 60%, 70% and 80% of the problems by using Gaussian, Cubic and Multiquadric respectively.

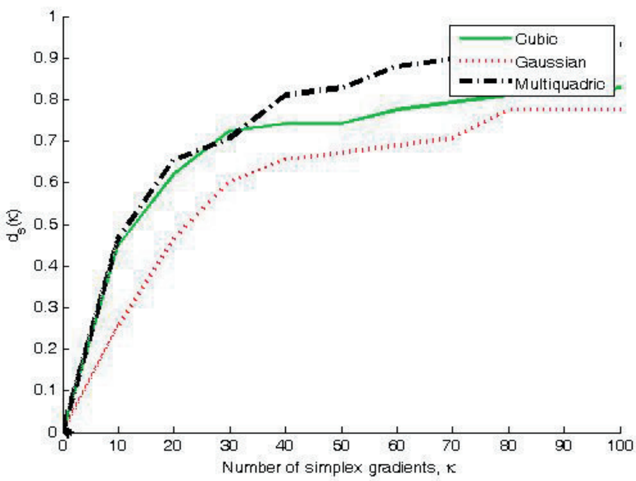


Figure 8. Data profiles

4.1 Recommendations

According to Figure 2 and Table 5, the best number of data points for Gaussian is $p = \frac{(n+1)(n+2)}{2}$. In addition, the first three top ANF are obtained when $\varepsilon = \{2, 4, 7\}$, and according to Table 3 the first three top AFmin by this data points number are obtained when $\varepsilon = \{1, .5, 2\}$ respectively. So, we can recommend that the reasonable ε may be one of the values $\{.5, 1, 2, 4, 7\}$ or in general in the interval $[.5, 7]$.

On the other hand, according to Figure 4 and Table 5 the best number of data points for Multiquadric is $p = \frac{(n+1)(n+2)}{2}$. Also, the first three top ANF are obtained when $\varepsilon = \{9, 1, 2\}$, and according to Table 4 the first three top AFmin by this data points number are obtained when $\varepsilon = \{.5, 5, 2\}$ respectively. So, we can recommend that the reasonable ε in this case may be one of the values $\{.5, 1, 2, 5, 9\}$ or in general in the interval $[.5, 9]$.

We should here note that there are other methods to determine a good shape parameter. In addition, many systematic approaches have been suggested in the statistics literature such as LOOCV and MLE to determine a good shape parameter. Using these statistical approaches can be very useful and can be considered for ORBIT as future work.

In addition, according to Figures 7 and 8 we recommend the shape-parameter-based RBFs such as Gaussian and Multiquadric as long as an appropriate ε is selected.

5. Conclusions

ORBIT is one of the most famous trust-region algorithms that uses an RBF interpolation in trust-region framework. Numerical results show that the type of RBF, the number of data points, and the shape parameter value can affect the RBF interpolation models and thus the efficiency of ORBIT. In this paper we have investigated the effect of these options on the performance of ORBIT. To show the performance of different solvers we utilize Dolan-Moré performance and Moré-Wild data profiles. Based on the results, some recommendations were given on choosing the appropriate shape parameter and the number of data points. Also, with respect to the performance profiles shape-parameter-based RBFs are recommended provided that an appropriate shape parameter is selected. Although some suggestions are given in this paper to determine a good shape parameter, there are many other approaches (e.g. statistical approaches such as LOOCV and MLE) to determine a good shape parameter. These approaches can be very useful and can be considered for ORBIT as future work.

References

[1] M. Ahmadvand, M. Esmaeilbeigi, A. Kamandi, and F. M. Yaghoobi, A novel hybrid trust region algorithm based on nonmonotone and LOOCV techniques, *Comput. Optim. Appl.*, 72 (2) (2019), 499-524.

[2] N. Andrei, An unconstrained optimization test functions collection, *Adv. Model. Optim.*, 10 (1) (2008), 147-161.

[3] M. Bozzini, L. Lenarduzzi, and R. Schaback, Adaptive interpolation by scaled multiquadrics, *Adv. Comput. Math.*, 16 (2002), 375-387.

[4] E. Dolan and J. Moré, Benchmarking optimization software with performance profiles, *Math. Program.*, 91 (2002), 201-213.

[5] C. Elster and A. Neumaier, A grid algorithm for bound constrained optimization of noisy function, *IMA J. Numer. Anal.*, 15 (1995), 585-608.

[6] M. Esmaeilbeigi and M. M. Hosseini, Dynamic node adaptive strategy for nearly singular problems on large domains, *Eng. Anal. Bound. Elem.*, 36 (9) (2012), 1311-1321.

[7] G. Fasshauer, *Meshfree Approximation Methods with MATLAB*, Interdisciplinary Mathematical Sciences, vol. 6. World Scientific Publishing Company, Singapore, (2007).

[8] P. S. Ferreira, E. W. Karas, and M. Sachine, A globally convergent trust-region algorithm for unconstrained derivative-free optimization, *Comput. Appl. Math.*, 34 (3) (2015), 1075-1103.

[9] B. Fornberg and J. Zuev, The Runge phenomenon and spatially variable shape parameters in RBF interpolation, *Comput. Math. Appl.*, 54 (2007), 379-398.

[10] J. Fu and W. Sun, Nonmonotone adaptive trust-region method for unconstrained optimization problems, *Appl. Math. Comput.*, 163 (2005), 489-504.

[11] N. I. M. Gould, D. Orban, and Ph. L. Toint, CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization, *Comput. Optim. Appl.*, 3 (2014) 4-12.

[12] S. Goldfeld, R. Quandt, and H. Trotter, Maximization by quadratic hill climbing, *Econometrica*, 34 (1966) 541-551.

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

[14] F. M. B. Martìnez, *Meshless Methods for Elliptic and Free-Boundary Problems*, Ph.D thesis, (2008).

[15] D. W. Marquardt, An algorithm for least squares estimation of nonlinear parameters, *SIAM J. Appl. Math.*, 11 (1963), 431-441.

[16] J. Moré and S. M. Wild, Benchmarking derivative-free optimization algorithms, *SIAM J. Optim.*, 20 (1) (2009), 172-191.

[17] F. J. Narcowich and J. D. Ward, Norm estimates for the inverses of a general class of scattered-data radial-function interpolation matrices, *J. Approx. Theory.*, 69 (1992), 84-109.

[18] K. Levenberg, A method for the solution of certain nonlinear problems in least squares, *Quarterly. Appl. Math.*, 2 (1944), 164-168.

[19] J. Liu, X. Xu, and X. Cui, An accelerated nonmonotone trust region method with adaptive trust region for unconstrained optimization, *Comput. Optim. Appl.*, 69 (1) (2018), 77-97.

[20] M. J. D. Powell, *A New Algorithm for Unconstrained Optimization*, in: J. B. Rosen, O. L. Mangassarian, K. Ritter (Eds.), Nonlinear Programming, Academic Press, New York, (1970), 31-66.

[21] M. J. D. Powell, The NEWUOA software for unconstrained optimization without derivatives, In G. Di Pillo and M. Roma, editors, *Large-Scale Nonlinear Optimization,* Springer, (2006), 255-297.

[22] F. Rahmanpour and M. M. Hosseini, A Smooth Method for Solving Non-Smooth Unconstrained Optimization Problems, *J. Math. Ext.*, 10 (3) (2016), 11-33.

[23] Z. Saeidian, M. Peyghami, M. Habibi, and S. Ghasemi, A new trust-region method for solving systems of equalities and inequalities, *Comput. Appl. Math.*, 36 (1) (2017), 769-790.

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

[25] R. Schaback and H. Wendland, Kernel techniques: From machine learning to meshless methods, *Acta. Numer.*, 15 (2006), 543-639.

[26] H. Wendland, *Scattered Data Approximation,* Cambridge monographs on applied and computational mathematics, Vol. 17, Cambridge University Press, Cambridge, (2005).

[27] S. M. Wild and C. A. Shoemaker, Global convergence of radial basis function trust region derivative-free algorithms, *SIAM J. Optim.*, 21 (3) (2011), 761-781.

[28] S. M. Wild, R. G. Regis, and C. A. Shoemaker, Optimization by radial basis function interpolation in trust-region, *SIAM J. Sci. Comput.*, 30 (6) (2008), 3197-3219.

[29] G. B. Wright, *Radial Basis Function Interpolation: Numerical and Analytical Developments*, Ph.D thesis, (2003).

Mohammad Ahmadvand
Ph.D Student of Mathematics
Department of Mathematics, College of Basic Sciences
Hamedan Branch, Islamic Azad University
Hamedan, Iran
E-mail: ahmadvand@malayeriau.ac.ir

Mohsen Esmaeilbeigi
Assistant Professor of Mathematics
Department of Mathematical Sciences
Malayer University
Malayer, Iran
E-mail: m.esmaeilbeigi@malayeru.ac.ir

Farajollah Yaghoobi
Assistant Professor of Mathematics
Department of Mathematics, College of Basic Sciences
Hamedan Branch, Islamic Azad University
Hamedan, Iran
E-mail: yaghoobi@iauh.ac.ir

Ahmad Kamandi
Assistant Professor of Mathematics
Department of Mathematics
University of Science and Technology of Mazandaran
Behshahr, Iran
E-mail: ahmadkamandi@mazust.ac.ir