Turkish Journal of Mathematics

http://journals.tubitak.gov.tr/math/

Research Article

Turk J Math
(2017) 41: 808 – 824
© TÜBİTAK
doi:10.3906/mat-1512-61

Penalty-free method for nonsmooth constrained optimization via radial basis functions

Fardin RAHMANPOUR[1,*], **Mohammad Mehdi HOSSEINI**[1,2], **Farid Mohammad MAALEK GHAINI**[1]
[1]Department of Mathematics, Yazd University, Yazd, Iran
[2]Department of Applied Mathematics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran

Received: 13.12.2015 • **Accepted/Published Online:** 27.08.2016 • **Final Version:** 25.07.2017

Abstract: We consider a general class of nonlinear constrained optimization problems, where derivatives of the objective function and constraints are unavailable. This property of problems can often impede the performance of optimization algorithms. Most algorithms usually determine a quasi-Newton direction and then use line search techniques. We propose a smoothing algorithm without the need to use a penalty function. A new algorithm is developed to modify the trust region and to handle the constraints based on radial basis functions (RBFs). The value of the objective function is reduced according to the relation of the predicted reduction of constraint violation achieved by the trial step. At each iteration, the constraints are approximated by a quadratic model obtained by RBFs. The aim of the present work is to keep the good position for the interpolation points in order to obtain a proper approximation in a small trust region. The numerical results are presented for some standard test problems.

Key words: Exact penalty function, derivative-free method, trust-region method, nonsmooth optimization, radial basis functions, constrained optimization, nonlinear programming

1. Introduction

Consider the nonlinear optimization problem with general nonlinear constraints:

$$\begin{array}{ll} \min\limits_{x} f(x) & \\ s.t. & \\ g_p(x) \geqslant 0 & p \in I_1 = \{1,2,\ldots,P\} \\ h_t(x) = 0 & t \in I_2 = \{1,2,\ldots,T\}, \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$, $g_p, h_t : \mathbb{R}^n \to \mathbb{R}$, $p \in I_1$, $t \in I_2$ are real valued continuous functions for which there is no need for differentiability.

The penalty method needs the evaluation of the objective function and the constraints. There are several difficulties associated with the use of penalty functions. The effectiveness of these methods depends on the selection of penalty parameter. To avoid this deficiency, penalty-free methods are proposed. Moreover, computation of the objective and constraints is complicated.

The first proposed penalty-free method seems to be an algorithm presented by Yamashita [47]. In [47], a quasi-Newton method that does not use a penalty function for equality constraints is developed. In 2003,

*Correspondence: fardin_rahmanpour@yahoo.com
2010 *AMS Mathematics Subject Classification:* 65D05, 65K05, 90C30, 90C20, 90C56.

Yamashita and Yabe [48] modified this algorithm so that it can be applied to the solution of problems with equality and inequality constraints. Another early penalty-free method is the Zoppke-Donaldsons tolerance-tube algorithm, which was introduced in the author's PhD thesis [50] in 1995. On the other hand, researchers proposed some other penalty-free methods such as the SQP method by Liu and Yuan [30]. To avoid the selection of the penalty parameter, some authors worked on the technique without the penalty function, for example, see [10, 12, 26, 42]. The methods that do not use any penalty function are called the penalty-free-type ones. Several extensions of trust region methods for problems involving inequality constraints have been proposed; some of them are based on trust region methods for box-constrained problems; see, Byrd [8], Omojokun [35], Dennis et al. [22], Conn et al. [15], Dennis and Vicente [24], and Chen et al. [11]. There are other related approaches, such as Dennis et al. [23], Plantenga [37], Vicente [44], Byrd et al. [9], and Coleman and Li [14], which combine the trust region method with the interior point method.

Most of these approaches require derivatives of the objective function and the constraints. We also assume that some derivatives of the objective and constraint functions are either unavailable or are computationally too expensive to obtain.

Recently, derivative-free trust region algorithms have been used increasingly [18, 29, 39, 49]. A common approach is to combine conventional algorithms such as genetic algorithms or pattern search with surrogate models to solve expensive problems. For instance, Booker et al. [6] and Jones et al. [28] proposed methods based on Kriging basis functions. In recent years, nonlinear optimization is perhaps one of the most common reasons for using derivative-free methods. Forming surrogate models by interpolation has been proposed by Winfield [46] and reviewed by Powel [39] and Conn [18]. A derivative-free algorithm for constrained global optimization based on exact penalty functions has been proposed by Pillo et al. [36]. Regis and Shoemaker [41] constructed a surrogate model based on radial basis functions (RBFs).

The present paper gives a new derivative-free method without a penalty function for the solution of (1), which belongs to the class of trust-region methods for constrained optimization. The underlying idea of this method is towards the two goals in determining a trial point, whether it is accepted or not. One of them is to improve the feasibility and the other is to reduce the value of the objective function.

The main contribution of this paper is to given trust-region algorithm using RBF's interpolation without using any penalty function or filter. Thus, the new method does not need updating the penalty parameter in each iteration.

Our aim in this paper is to find an efficient algorithm for the global solution of optimization problems with general constraints. At each iteration instead of constraints a quadratic surrogate model is built via RBFs to obtain the suitable feasible point to have a sufficient decrease in the objective function. Thus, we have chosen the position of interpolation points within a sphere of radius $\Delta > 0$ around the trial point. The important idea is the assumption that the interpolation points exist and can well approximate constraints in small spheres. When the current trial point is not sufficiently close to a local minimum, we update the interpolation points and construct a new model by RBFs.

In the previous methods, whenever a trial point did not decrease the objective function as expected, one of the interpolation points was replaced by another evaluated point. In our approach, all the interpolation points can be changed at each iteration. Since evaluation of objective function is computationally expensive, we stress the importance of having complete knowledge of all points previously evaluated by the algorithm. This is a fundamental difference between our method and previous algorithms, where, in order to reduce linear algebraic costs, the interpolation set was allowed to change by at most one point under proper assumptions.

The proposed method will guarantee global convergence. Furthermore, models based on RBFs have been shown to be of interest for global optimization.

The paper is organized as follows. In section 2, the surrogate model is introduced. In section 3, the RBFs are described. In section 4, we present a derivative-free optimization method. Section 5 gives a summary of the surrogate model based on RBFs. In section 6, the algorithm is introduced and its convergence properties are established. Numerical results for some examples are reported in the last section.

Throughout the paper $||.||$ denotes the Euclidean norm and for simplicity we also use subscripts to denote functions evaluated at iterates, for example, $f_k = f(x_k),\ k = 1, 2, \ldots,\ \ g_p = g(x_p)$, and $h_t = h(x_t)$.

2. Surrogate method

We consider the following nonlinear subproblem:

$$\min_{x\in\mathbb{R}^n} C(x) = \sum_{t=1}^{T} h_t^2(x) + \sum_{p=1}^{P} \max(g_p(x), 0), \tag{2}$$

where $g_p : \mathbb{R}^n \to \mathbb{R}$ and $h_t : \mathbb{R}^n \to \mathbb{R}$ are functions that are not necessarily differentiable. In this paper, we propose the smooth surrogate model, which is minimized more easily than $C(x)$.

We remark that surrogate modeling is referred to as a technique that uses the sample points to build a surrogate function, which is sufficient to predict the behavior of the $C(x)$.

2.1. Quadratic surrogate model

Powell [39] and Conn et al. [17, 18] proposed the surrogate model as follows:

$$Sm(x_k + s) = C(x_k) + \nabla C(x_k)^T s + \frac{1}{2} s^T \nabla^2 C(x_k) s,$$

where $C(x)$ is twice differentiable and admits a Hessian matrix $\nabla^2 C(x)$ that is positive definite.

The goal is to construct the surrogate model $Sm(x)$ instead of the $C(x)$, which is computationally simple and inexpensive with good analytical properties. It could be used in optimization because of its simplicity and suitable algebraic form.

To build a quadratic model, we define the trust region $B_k := \{x \in \mathbb{R}^n : ||x - x_k|| \leq \Delta_k\}$. At each iteration of the surrogate method, the solution of the optimization problem inside B_k [16, 33], as

$$\min_{s} Sm(x_k + s) \quad s.t. \quad ||s|| \leq \Delta_k, \tag{3}$$

is needed for some trust region with radius $\Delta_k \geq 0$. The ratio of the actual $C(x)$ over the predicted $Sm(x)$ is as follows:

$$\rho_k = \frac{C(x_k) - C(x_k + s_k)}{Sm(x_k) - Sm(x_k + s_k)}; \tag{4}$$

given the standard trust region $0 \leq \eta_0 \leq \eta_1 < 1,\ 0 < \gamma_0 < 1 < \gamma_1,\ 0 < \Delta_k \leq \Delta_{\max}$, and $x_k \in \mathbb{R}^n$, we define a model Sm_k on B_k, and compute a step s_k such that $x_k + s_k \in B_k$, in order to sufficiently reduce the model $Sm(x_k)$.

By accepting the trial point x_k, we compute $C(x_k + s_k)$ and ρ_k by using (4), and then update the surrogate model parameters as follows:

$$x_{k+1} = \begin{cases} x_k + s_k & \rho_k \geq \eta_0 \\ x_k & o.w. \end{cases}$$

and

$$\Delta_{k+1} = \begin{cases} \Delta_k & \eta_0 \leq \rho_k < \eta_1, \\ \min\{\gamma_1 \Delta_k, \Delta_{\max}\} & \rho_k \geq \eta_1, \\ \gamma_0 \Delta_k & \rho_k < \eta_0; \end{cases}$$

the following assumptions were considered in this section:

(**A1**) $C(x)$, which is defined by (2), is a two times differentiable function.

(**A2**) $\{x_k\}$ is a bounded sequence.

(**A3**) $\frac{||\nabla C(x_k)||}{||\nabla^2 C(x_k)||} = +\infty$ when $\nabla^2 C(x_k) = 0$.

Suppose that these assumptions hold. Let s_k be a solution of subproblem (3). The following lemma, which can be obtained from the well-known result (Powel), is needed [39]:

Lemma 1 *Subproblem (3) satisfies a sufficient decrease condition of the form*

$$Sm(x_k) - Sm(x_k + s_k) \geq \frac{c}{2}||\nabla C_k|| \min(\Delta_k, \frac{||\nabla C(x_k)||}{||\nabla^2 C(x_k)||}),$$

for some constant $c \in (0, 1)$.

We implement the trust region method by using a cubic RBF model with 2-norm and we take care to distinguish between the trust region norm (at iteration k) and the standard 2-norm is used in the sequel.

Note that now the main questions are as follows: how to build surrogate models and how to evaluate the accuracy of the surrogate models?

3. Interpolation by radial basis functions

RBFs are a powerful tool to solve the multivariate scattered data interpolation problems. Their use is not common in optimization but they have a potential that we have exploited by developing a new derivative-free algorithm based on RBFs. The scattered data interpolation problem consists of finding a function that interpolates another function at some given points. Interpolation functions based on radial basis functions have some nice properties. The matrix of the system made by these interpolation constraints is nonsingular.

A radial basis function $\phi(x)$ by the linear combination of N translates of a radial function is a one variable continuous function, where $\phi(x) = \phi(||x - x_i||)$, $i = 1, \ldots, N$ for a finite set of center's $x_i \in \mathbb{R}^n$. Here the radial basis function is simply $\phi(r) = \phi(||x - x_i||)$, $i = 1, \ldots, N$ [7].

A multivariate interpolation can be stated as follows: given data $(x_i, C(x_i))$, $i = 1, \ldots, N$, with $x_i \in \mathbb{R}^n$, $C(x_i) \in \mathbb{R}$, we find a continuous function $Sm(x)$ such that $Sm(x_i) = C(x_i)$, $i = 1, \ldots, N$.

The function $Sm(x)$ is assumed to be given by a linear combination of RBFs for the interpolation points $\{y^i\}_{i=1}^N$, that is,

$$Sm(x_k + s) = \sum_{i=1}^{N} \lambda_i \varphi(||s - y^i||) + V(s), \tag{5}$$

where $\varphi(||s-y^i||)$ from $\mathbb{R}_+$ to $\mathbb{R}$ is a univariate function centered at the point s. Note that we have $V(s)=\sum_{j=1}^{M}\gamma_j\nu_j(s)$, where $\nu=\{\nu_1(s),\ldots,\nu_M(s)\}$ is an ordered basis for the linear space π^n_{M-1}, and $\Gamma=\{\gamma_1,\ldots,\gamma_M\}\in R$ are scalar coefficients to be determined. The space of n variable polynomials of total degree less than or equal to $M-1$. $\{\lambda_j\}_{j=1}^N$ are the unknown RBF's coefficients. $Sm(x)$ as defined by (5) has M degrees of freedom. To overcome additional degrees of freedom two more constraints are imposed as follows:

$$Sm(x_i+s)=C(x_i+s),\qquad i=1,\ldots,N, \tag{6}$$

$$\sum_{i=1}^{N}\lambda_i\nu_k(s)=0,\qquad k=1,\ldots,M. \tag{7}$$

We consider the equations (6) and (7) can be written in the form of the symmetric linear system:

$$\begin{bmatrix}\Phi & V\\ V^T & 0\end{bmatrix}\begin{bmatrix}\Lambda\\ \Gamma\end{bmatrix}=\begin{bmatrix}f\\ 0\end{bmatrix}, \tag{8}$$

where $\Lambda=\{\lambda_i\}_{i=1}^N$ and $\Gamma=\{\gamma_j\}_{j=1}^M$ are the undetermined coefficient vectors. For the sake of clarity, the matrix Φ is in the form:

$$\Phi=\begin{bmatrix}\varphi(||x_1-x_1||) & \cdots & \varphi(||x_1-x_N||)\\ \vdots & & \vdots\\ \varphi(||x_N-x_1||) & \cdots & \varphi(||x_N-x_N||)\end{bmatrix}_{N\times N}.$$

It can be seen (8) is well-posed if the coefficient matrix is nonsingular [7]. Moreover, RBFs can be classified by using the concept of conditionally positive definite functions. The main interest of the concept of conditionally positive definite functions is characterization of the nonsingularity of the interpolation matrix.

Definition 1 *Let ν be a basis for π^n_{M-1}, with the convention that $\pi=\emptyset$ if $M=0$. A function φ is said to be conditionally positive definite (CPD) of order M if for all distinct points in $Y\subset\mathbb{R}^n$ and all $\lambda\neq 0$, such that $\sum_{i=1}^{|Y|}\lambda_i\pi(y_i)=0$, the quadratic form $\sum_{i,j=1}^{|Y|}\lambda_i\lambda_j\varphi(|y_j-y_i||)$ is positive[7, 32, 45].*

Micchelli [32] proved that the interpolation problem in equation (8) is solvable when the following two conditions are met:
(**A1**) The points $\{x_j\}_{j=1}^N$ are distinct.
(**A2**) The RBFs used are conditionally positive definite.
Some of the most popular RBFs are shown in Table 1.

4. Derivative-free optimization

In this section, we suppose that $C(x)$ is a function from $\mathbb{R}^n$ into $\mathbb{R}$ that is not necessarily smooth. The algorithm is based on approximating the function (2) by a positive definite quadratic model. The main idea is to use the available values of $C(x)$ and building a quadratic model by interpolating within a trust region.

Suppose that in the current x_k, we have the sample points $Y_k=\{y_k^1=0,y_k^2,\ldots,y_k^N\}$, with $y_k^i\in R^n$, $i=1,\ldots,N$, which contains the points that are in the neighborhood of x_k. Furthermore, we will always enforce

Table 1. Some examples of popular RBFs and their orders of conditional positive definiteness.

$\phi(r)$	Order	Parameters	Example
r^β	2	$\beta \in (2,4)$	$Cubic, r^3$
$(c^2+r^2)^\beta$	2	$c>0,\ \beta \in (1,2)$	MqI, $(c^2+r^2)^{\frac{3}{2}}$
$-(c^2+r^2)^\beta$	1	$c>0,\ \beta \in (0,1)$	MqII, $-(c^2+r^2)^{\frac{1}{2}}$
$(c^2+r^2)^{-\beta}$	0	$c>0,\ \beta>0$	Inv.Mq, $(c^2+r^2)^{-\frac{1}{2}}$
$Exp(-c^2r^2)$	0	$c>0$	Gaussian, $Exp(-c^2r^2)$

interpolation at the current iterate x_k so that $x_k = y_k^1$. We wish to construct a quadratic model of the form

$$Sm(x_k+s) = C(x_k) + \nabla C(x_k)^T s + \frac{1}{2} s^T \nabla^2 C(x_k) s, \tag{9}$$

where $\nabla C(x_k) \in \mathbb{R}^n$, and $\nabla^2 C(x_k) \in \mathbb{R}^{n\times n}$ is a symmetric matrix [34]. We impose the interpolation condition in what follows:

$$Sm(x_k + y^j) = C(x_k + y^j), \qquad j = 1,\ldots,N. \tag{10}$$

We now need to evaluate $Sm(x_k+s)$ in $N = \frac{1}{2}(n+1)(n+2)$ points to find an approximating quadratic form, where n is the number of variables [5, 18, 20].

We consider $\{\varphi_i(.)\}_{i=1}^N$ as a basis for the linear space of N-dimensional quadratic functions. The quadratic function (9) can be expressed as

$$Sm(x_k + y^j) = \sum_{i=1}^{N} \lambda_i \varphi_i(y^j), \qquad j = 1,\ldots,N$$

for some coefficients λ_i, which should be determined from the interpolation equation (10),

$$\sum_{i=1}^{N} \lambda_i \varphi_i(y_k^j) = C(x_k + y_k^j), \qquad j = 1,\ldots,N.$$

λ_i's are unique if the matrix below is nonzero

$$\begin{bmatrix} \varphi_1(y_k^1) & \cdots & \varphi_1(y_k^N) \\ \vdots & & \vdots \\ \varphi_N(y_k^1) & \cdots & \varphi_N(y_k^N) \end{bmatrix},$$

and is invertible. Then iteratively we optimize and update the surrogate model Sm_k to reach a satisfactory solution.

5. Surrogate methods based on radial basis functions

In this section, the relevance of the surrogate methods and RBFs is considered. Suppose

$$Sm(x_k+s) = \sum_{i=1}^{N} \lambda_i \varphi_i(s) + \sum_{k=1}^{M} \gamma_k \nu_k(s);$$

the model is twice differentiable, which is important for the convergence part of our method [21, 40]. This study considers the interpolation condition at the points of Y:

$$Sm(x_k + y_k^i) = C(x_k + y_k^i), \qquad \forall y_k^i \in Y.$$

Let $\Phi \in \mathbb{R}^{N\times N}$, $V \in \mathbb{R}^{N\times M}$ be the matrices defined by $\Phi_{ij} = \varphi(||y^i - y^j||)$ and $V_{jk} = \nu_k(y^j), k = 1,\dots,M$. Then the interpolation condition can be expressed as $\Phi\Lambda + V\Gamma = C$. By using RBFs we get the following linear system:

$$\begin{bmatrix} \Phi_{N\times N} & V_{N\times M} \\ V^T_{M\times N} & 0_{M\times M} \end{bmatrix} \begin{bmatrix} \Lambda_{N\times 1} \\ \Gamma_{M\times 1} \end{bmatrix} = \begin{bmatrix} C_{N\times 1} \\ 0_{M\times 1} \end{bmatrix} \Rightarrow \begin{bmatrix} \Phi & V \\ 0_{M\times N} & -V^T\Phi^{-1}V \end{bmatrix} \begin{bmatrix} \Lambda \\ \Gamma \end{bmatrix} = \begin{bmatrix} C \\ -V^T\Phi^{-1}C \end{bmatrix}, \tag{11}$$

with the solution $\Gamma = (V^T\Phi^{-1}V)^{-1}V^T\Phi^{-1}C$, $\Lambda = \Phi^{-1}(C - V\Gamma)$.

Sufficient condition for the solvability of system (11) is that the points in Y are distinct and yield a V of full column rank.

Suppose that $V = QR$ and hence $R \in \mathbb{R}^{N\times M}$. The null space of matrix V^T, denoted $\aleph(V^T)$, is the set of all solutions to the homogeneous equation $V^TZ = 0$. Written in set notation, we have

$$\aleph(V^T) = \{Z : Z \in \mathbb{R}^N \text{ } and \text{ } V^TZ = 0\}$$

If Z is an orthonormal basis for the null space of V^T [3], using the equation (7) it follows that $\Lambda \in \aleph(V^T)$. Therefore, $\Lambda = Zw$. According to (11), $\Phi\Lambda + V\Gamma = C$. Multiplying by Z^T from the left gives $Z^T\Phi\Lambda + Z^TV\Gamma = Z^TC$. Keeping in mind that Z is an orthonormal basis for the null space V^T, we obtain $Z^TV\Gamma = 0$. Hence

$$Z^T\Phi Zw = Z^TC. \tag{12}$$

Now we can obtain w from (12) and compute Λ. If we use RBFs based on cubic spline [7, 20], which is the smooth and conditional positive definite interpolants, then $Z^T\Phi Z$ is also positive definite. By replacing the Cholesky factorization $Z^T\Phi Z = LL^T$, in which L is a nonsingular lower triangular matrix in (12), we obtain $LL^Tw = Z^TC$, which yields $w = (LL^T)^{-1}Z^TC$ and so

$$||\Lambda|| = ||Zw|| = ||Z(L^T)^{-1}L^{-1}Z^TC|| \le ||L^{-1}||^2|C|.$$

Moreover, for procure Γ, we have $\Phi\Lambda + V\Gamma = C$, which by using the QR factorization and premultiplying by Q^T, yields $R\Gamma = Q^T(C - \Phi\Lambda)$. Finally $\Lambda = Zw$ concludes

$$R\Gamma = Q^T(C - \Phi Zw). \tag{13}$$

Now we discuss a method of creating surrogate models. For this purpose Φ must be conditionally positive definite of order at least 2 (Table 1) and $V \in \pi_2^n$ linear.

The RBF coefficients λ_i and γ_i must be bounded in magnitude. Suppose y^i is the ith point in Y, that is in the vicinity of the trust region. However, for $n \ge 1$ equation (10) is not sufficient for the existence and uniqueness of the interpolant and to guarantee the good quality of the model. Some geometric conditions on the set Y are required to ensure the existence and uniqueness of the interpolant [19].

Definition 2 *Let $Y = \{y^1, \ldots,\ y^N\}$ be points in $\mathbb{R}^n$. These points are called affinely independent if there do not exist real numbers $\alpha_1, \alpha_2 \ldots \alpha_N$ that are not all zero such that $\sum_{i=1}^{N} \alpha_i y^i = 0$ and $\sum_{i=1}^{N} \alpha_i = 0$.*

The process can be summarized as follows:

The study chooses the $n+1$ affinely independent points and then generates the other interpolation points.

Definition 3 *Y is unisolvent for π_M^n if there exists a unique polynomial in π_M^n of lowest possible degree with interpolation points of Y.*

The cubic spline $\varphi(r) = r^3$ in dimension n is unisolvent on points $Y = \{y^1, \ldots,\ y^N\}$ if the matrix

$$\left[\varphi(||y^i - y^j||)\right] \qquad 1 \leq i,\ j \leq N,$$

is invertible for any choice of N distinct points $y^1, \ldots, y^N \in Y$. Unisolvent systems of RBFs are widely used in interpolation because they guarantee a unique solution to the interpolation problem. This is equivalent to the interpolation system (11), which is nonsingular if the interpolation point set Y is unisolvent.

The collection of $n+1$ distinct points will uniquely determine a polynomial of lowest possible degree in π^n. In this section, we describe an algorithm to find $n+1$ interpolation points that are affinely independent. We suppose $D := \{d_i \in \Delta_k \mid C(x_k + d_i) \textit{ is known}\}$; we note that Δ_k is the chosen radius of the current trust-region. Algorithm 1 shows how to obtain $n+1$ affinely interpolation points.

Algorithm 1 For finding n+1 affinely independent points:

Step0: Constants $0 < \gamma_0 \leq \gamma_1$, $\Delta_k \in (0, \Delta_{max}]$ and $|D| > N$.

Step1: Choose $D = \{d_1, d_2, \ldots, d_{|D|}\} \subset \mathbb{R}^n$ such that $x_i = x_k + d_i$ are close to x_k.

Step2: Let $Z = I_n$.

Step3: While $i, j \geq 1$

if $||d_i|| \leq \gamma_1.\Delta_k$, define $u = \frac{d_i}{\gamma_0 \Delta_k}$,

    if $||proj_Z^u|| \geq \gamma_0$, then $y_j = d_i$,

Using the Gram–Schmidt algorithm, we obtain an orthogonal basis for Y as $\bar{Z}$, update $Z = \bar{Z}$,

Step4: If $|Y| < n+1$

    if $||d_i|| \leq 2\Delta_{max}$, define $u = \frac{d_i}{\gamma_0 \Delta_k}$

    if $||proj_Z^u|| \geq \gamma_0$, then $y_j = d_i$

Using the Gram–Schmidt process, we obtain an orthonormal basis for Y as $\bar{Z}$. Update $Z = \bar{Z}$.

We note that the projections in Steps 3 and 4 are exactly the magnitude of the pivot that results from adding point d_i to Y. The projection of d_i onto Y is defined by $\frac{\langle Y, d_i \rangle}{||Y||^2} Y$. We can alternatively define the projection of d_i onto Y to be the vector in the space spanned by Y that is closest (in Euclidean distance) to d_i.

However, with $n+1$ points, the solution of the system (11) is just interpolation obtained for linear function and coefficient $\Lambda = 0_N$. To build the surrogate model for nonlinear functions, we must add some new points. Algorithm 2 shows how we can obtain ”well independent” additional sample points in the trust region.

Algorithm 2 Finding additional independent points

Step0: Input Y (obtained from algorithm 1), $p_{max} = \frac{(n+1)(n+2)}{2}$ and $D = \{d_1, d_2, \ldots, d_{|D|}\} \subset \mathbb{R}^n$.

While $i \geq 1$

Step1: If $|Y| < p_{max}$,

$$\Pi^T = \begin{bmatrix} y^1 = 0 & y^2 & \ldots & y^{|Y|} & d_i \\ 1 & 1 & \ldots & 1 & 1 \end{bmatrix}.$$

Step2: Find the orthogonal basis Z for null space Π.

Step3: Build the interpolation matrix by using the cubic spline function at sample points Y,

$$\Phi_{new} = \begin{bmatrix} \Phi & \Phi_{d_j} \\ \Phi_{d_j}^T & 0 \end{bmatrix}.$$

Step4: Obtain P=$Z^T \Phi_{new} Z$:

$$P = Z^T \Phi_{new} Z = \begin{bmatrix} Z^T \Phi Z & Z^T \Phi_{d_j} Z \\ Z^T \Phi_{d_j}^T Z & 0 \end{bmatrix}.$$

Step5: P is positive definite for cubic spline function $\varphi(r) = r^3$; note that for P to be positive definite, the points Y must be distinct.

Step6: Let $P = LL^T$; if all diagonal entries of L are positive, then add d_j to the set of sample points Y.

This procedure continues until $|Y| = \frac{(n+1)(n+2)}{2}$. Note that the points $D = \{d_1, \ldots, d_{|D|}\}$ are in the neighborhood of the trial point x_k with radius Δ_k by using a random process.

In a derivative-free algorithm, it is essential to guarantee whenever necessary a model for the objective function with uniformly good local accuracy can be constructed. Indeed, it is no longer guaranteed that the model $Sm(x_k)$ approximates the function locally. Therefore, it is required that we design a derivative-free method similar to derivative-based models.

The main difference between interpolation models and gradient-based models is that the former are considered as suitable approximations of the objective function only under some specific conditions. These conditions depend mainly on the geometry of the points. If they are satisfied, we say that the model is valid in the trust region. If not, new points are generated to improve the accuracy of the model. The class of algorithms based on interpolation models is called the conditional trust region method. The term conditional just means that the model is a convenient approximation of f only if some conditions are satisfied. The general framework of trust region methods guarantees convergence to a first- or second-order critical point, depending on the assumptions on the model and on the objective function. A full analysis of trust region methods can be found in [25, 39, 49].

The radial function φ must be both twice continuously differentiable and we have relatively simple analytic expressions for the gradient:

$$\nabla Sm_k(x_k + s) = \sum_{i=1}^{N} \lambda_i \varphi'(||s - y_k^i||) \frac{s - y^i}{||s - y_k^i||} + \nabla V(s),$$

and similarly for the Hessian,

$$\nabla^2 Sm_k(x_k+s)=\sum_{i=1}^{N}\lambda_i[\frac{\varphi'(||s-y_k^i||)}{||s-y_k^i||}\mathrm{I}_n+(\varphi''(||s-y_k^i||)-\frac{\varphi'(||s-y_k^i||}{||s-y_k^i||})\frac{(s-y_k^i)}{||s-y_k^i||}\frac{(s-y_k^i)^T}{||s-y_k^i||}]$$

6. Optimization surrogate based on radial basis functions (OSRB)

This section discusses the details of the derivative-free algorithm for finding a global solution of problem (1). As pointed out in the introduction, since the objective function and constraints are not necessarily smooth, the traditional methods are not sufficient to search good directions. In order to optimize problem (1) in a feasible area, similar to the penalty method, the two following aims are considered. One is to satisfy the constraints and the other is to minimize the objective function. We give the algorithm of the feasibility phase in which a surrogate model of problem (2) is solved. The algorithm proceeds until the magnitudes of the constraints become less than a natural stopping criterion.

Here we propose a derivative-free algorithm without using the penalty parameter. In this algorithm we solve the subproblem (3), which is approximated by using RBFs (as in Section5) and obtain a search direction s_k.

Given the current iterate x_k at step k, we probe the behavior of the objective function $f(x)$ along the direction s_k. In the case a sufficient reduction of the function value is obtained, a suitable optimal solution is computed and is used for the next iteration, i.e. $x_{k+1}=x_k+s_k$. If we do not obtain a sufficient reduction, then the trust region radius Δ_k is updated and interpolation points are chosen again (Algorithms 1 and 2). By solving the subproblem (3), we obtain another direction s_k at the next iteration that suitably reduces the objective function.

Given N, a set of distinct interpolation points $Y_k=\{y_k^1=0,y_k^2,\dots,y_k^N\}\in\mathbb{R}^n$, and the function values $\{C(x_k+y_k^i)\}$, we obtain the surrogate model for C on Y_k. Algorithm 3 is described as follows:

In step 1, the interpolation points set $Y=\{y^1=0,\dots,y^{p_{max}}\}$ is determined, which are linearly independent. In step 2, we consider how to construct a model and to obtain parameters of RBF model from (12) and (13). In step 4, the algorithm uses criteria for model $Sm_k(x_k+s)$ and updates the parameters trust region method. We finds the candidate step s_k by approximately solving the subproblem (3). In this paper, we solve subproblem (3) by using the Fmincon function in MATLAB software.

6.1. Convergence properties of OSRB algorithm

In this study, the trust region algorithm ensures that $C(x)$ is sampled only within the relaxed level set, $L(x):=\{y\in R^n;\ ||x-y||\le\Delta_{\max}\}$.

Theorem 1 *Let* $\{\Delta_k\}$ *and* $\{x_k\}$ *be sequences generated by the OSRB algorithm. Then* $\lim_{k\to\infty}\Delta_k=0$ *and* $\lim_{k\to\infty}\nabla C(x_k)=0$.

Proof After the last successful iteration, there is an infinite number of iterations that are not either acceptable or successful. If $x_{k+1}=x_k+s_k$ is obtained so that $C(x_{k+1})\le C(x_k)$, then Δ_k is never increased for sufficiently large k, and so Δ_k is decreased at least once every n iterations by a factor of $0<\gamma<1$; thus Δ_k convergence to zero. Secondly, for each k, after the jth iteration we have $||x_k-x_j||\to 0$ since $||x_k-x_j||\leqslant\lim_{k\to\infty}n\Delta_k$ and

Algorithm 3 Iteration k of a derivative-free surrogate model

Step0: Input $0 \leq \gamma_0 < \gamma_1 \leq 1$, $0 < \eta < 1$, $0 < \Delta_1 \leq \Delta_{max}$, $\theta > 1$ and $\epsilon, \epsilon' > 0$.

We assume that trial point x_k is given.

While $k \geq 1$

Step1: From algorithm 1 and 2 find independent points that are denoted by Y.

Step2: Obtain surrogate model $Sm(x_k + s)$ by using the RBFs described in Section 5.

Step3: If $|f(x)| \leq \epsilon'$, then terminate.

Step4: While $||\nabla Sm(x_k)|| > \epsilon$

If $Sm(x_k) - Sm(x_k + s_k) \leq \frac{\eta}{2} ||\nabla Sm(x_k)|| \min(\Delta_k, \frac{||\nabla Sm(x_k)||}{||\nabla^2 Sm(x_k)||})$

Obtain a step s_k by solving: $\min \ \{Sm(x_k + s) \ ; \ x_k + s \in B(x_k, \Delta_k)\}$.

Evaluate $f(x_k + s_k)$ and update the trial point according to the ratio ρ_k (4),

$$x_{k+1} = \begin{cases} x_k + s_k & f(x_k + s_k) \leq f(x_k) \\ x_k & o.w. \end{cases}$$

$$\Delta_{k+1} = \begin{cases} min\{\gamma_1 \Delta_k, \Delta_{\max}\} & f(x_k + s_k) \leq f(x_k) \\ \gamma_0 \Delta_k & f(x_k + s_k) > f(x_k) \end{cases}$$

If there are no adaptable direction s_k to minimize $Sm(x_k)$, otherwise update $\Delta_k = \theta.\Delta_k$ and go to Step 1.

$\lim\limits_{k \to \infty} n\Delta_k \to 0$. Now

$$||\nabla C(x_k)|| \leqslant ||\nabla C(x_k) - \nabla Sm(x_k) + \nabla Sm(x_k)|| \leqslant ||\nabla C(x_k) - \nabla Sm(x_k)|| + ||\nabla Sm(x_k)||.$$

All terms on the right-hand side are equal to zero. □

The statement of theorem 1 gives a natural stopping criterion for the OSRB algorithm. It results from the updating of the trust region at the k-th iteration.

Surrogate model Sm_k is made such that

$Sm(x_k + s_k) - Sm(x_k) = \mathcal{G}^T s_k + \frac{1}{2} {s_k}^T \mathcal{H}_k s_k$,

where $\mathcal{G}_k = \nabla C(x_k)$ and $\mathcal{H}_k = \nabla^2 C(x_k)$.

Assumption 1 *The following assumptions were considered:*

*(**A1**) The expression $C(x)$, defined by (2), is bounded below on $L(x)$.*

*(**A2**) The quadratic model $Sm(x)$, defined by (9), is twice continuously differentiable.*

Now we discuss the corresponding lemma 2 on the models realizations that we use in the algorithm. This lemma guarantees that we are able to adequately reduce the model at each iteration of our algorithm.

Lemma 2 *Suppose that assumption 1 holds. Then*

$$Sm(x_k + s_k) - Sm(x_k) \geq \frac{1}{2} ||\mathcal{G}_k|| \min\{\Delta_k, \frac{||\mathcal{G}_k||}{||\mathcal{H}_k||}\}.$$

Proof We first note that

$$Sm(x_k - \alpha_k \mathcal{G}) = Sm(x_k) - \alpha ||\mathcal{G}||^2 + \frac{1}{2} \alpha^2 \mathcal{G}^T \mathcal{H} \mathcal{G}.$$

If $s_k = -\mathcal{H}_k^{-1}\mathcal{G}_k$ such that $||s_k|| \leq \Delta_k$, then the quadratic subproblem (9) can be resolved,

$$Sm(x_k - \mathcal{H}_k^{-1}\mathcal{G}_k) = Sm(x_k) - \mathcal{H}_k^{-1}||\mathcal{G}_k||^2 + \frac{1}{2}(\mathcal{H}_k^{-1})^2\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k.$$

Knowing the cubic spline is twice continuously differentiable and $\mathcal{H}_k$ is symmetric and positive definite, $\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k > 0$, we know the model is convex along direction s_k and so a stationary point will necessarily be the global minimizer in the direction s_k. Denoting the optimal parameter by α^* we have

$$-||\mathcal{G}_k||^2 + \alpha^*\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k = 0 \Rightarrow \alpha^* = \frac{||\mathcal{G}_k||^2}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k}.$$

Thus if $||\alpha^*\mathcal{G}_k|| \leq \Delta_k$, we conclude

$$Sm(x_k + s_k) - Sm(x_k) = -\frac{||\mathcal{G}_k||^4}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k} + \frac{1}{2}\frac{||\mathcal{G}_k||^4}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k} = -\frac{1}{2}\frac{||\mathcal{G}_k||^4}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k},$$

and consequently

$$Sm(x_k) - Sm(x_k + s_k) = \tfrac{1}{2}\tfrac{||\mathcal{G}_k||^4}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k} \geq \tfrac{1}{2}\tfrac{||\mathcal{G}_k||^2}{\mathcal{G}_k^T\mathcal{H}_k\mathcal{G}_k} \geq \tfrac{1}{2}||\mathcal{G}_k|| \min\{\Delta_k, \tfrac{||\mathcal{G}_k||}{||\mathcal{H}_k||}\}.$$

□

It is worth noting that lemma 2 guarantees that the OSRB algorithm is convergent and the objective function will sufficiently decrease at iteration k. The subproblem (3) is required to satisfy a sufficient decrease condition of the form given in lemma 2 at each iteration.

7. Numerical results

In this section, we present preliminary computational results to illustrate the performance of the Algorithm 3, denoted by OSRB. For the nonlinear programming problems we use the following OSRB parameters: $\Delta_1 = \max(1, ||x_0||)$, $\Delta_{\max} = 10^3\Delta_1$, $\eta_0 = 0$, $\eta_1 = 10^{-3}$, $\gamma_0 = 0.1$, $\gamma_1 = 10$, and termination criterion $||\nabla Sm(x_k)|| < 1.e-7$.

We present the well-known engineering optimization problem. This example has linear and nonlinear constraints, and has been previously solved using a variety of other techniques, and is useful to show the validity and effectiveness of the proposed algorithm.

Example 1 *Tension/compression spring design problem*

The tension/compression spring design problem is described by Belegundu [2] and Arora [1] for which the aim is to minimize the weight ($f(x)$) of a tension/compression spring (as shown in Figure1) subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter, and design variables. The design variables are wire diameter $d(x_1)$, mean coil diameter $D(x_2)$, and number of active coils $P(x_3)$. Formally, the problem can be expressed as

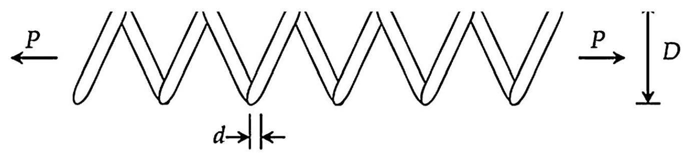

Figure 1. Tension/compression spring design problem.

$$
\begin{aligned}
&\min_{x} f(x) = (x_3+2)x_2x_1^2 \\
&s.t. \\
&\quad g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0 \\
&\quad g_2(x) = \frac{4x_2^2 - x_1x_2}{125669(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \\
&\quad g_3(x) = 1 - \frac{140.45x_1}{x_2^2 x_3} \leq 0 \\
&\quad g_4(x) = \frac{x_1+x_2}{1.5} - 1 \leq 0 \\
&\quad 0.05 \leqslant x_1 \leqslant 2 \\
&\quad 0.25 \leqslant x_2 \leqslant 1.30 \\
&\quad 2.00 \leqslant x_3 \leqslant 15.00
\end{aligned}
$$

The comparison of the best solution among several algorithms is given in Table 2. This problem has been used as a benchmark problem for testing different optimization methods, such as Coello [13], CONMIN [43], and OPTDYN [4].

We present a set of nonlinear programming problems from [27] that have been solved by the OSRB algorithm proposed to accommodate a practical experiment to show the success of the proposed method. Note that the interpolation points are chosen so that the interpolation matrix always is invertible even when the trust region is very small.

Table 2. Comparison of the results for example 1 (tension/compression spring design problem).

D. V	Best solution				
	This paper	Coello	Arora	M-5	OPTDYN
x_1	0.05031	0.051989	0.053396	0.05000	0.0644
x_2	0.35321	0.363965	0.399180	0.315900	0.7488
x_3	11.1013	10.890522	9.185400	14.25000	2.9597
$g_1(x)$	-0.0637065	-0.000013	0.000019	-0.000014	-0.005134
$g_2(x)$	-0.07027909	-0.000201	-0.000018	-0.003782	0.002609
$g_3(x)$	-4.1019	-4.061338	-4.123832	-3.938302	-4.450398
$g_4(x)$	-0.073098	-0.722698	-0.698283	-0.756067	-0.457867

We have employed the Fmincon routine from MATLAB, which corresponds to the surrogate model. The starting points are randomly chosen in our algorithm. We solve problem (1) with equality and inequality constraints to show the efficiency of the proposed method.

The numerical results for the test problems are listed in Table 3. The headers of the columns are as follows: n is number of variables, $Eq.$ and $Ineq.$ are number of equality and inequality constraints respectively, f is the final value of the objective function value at the final iteration, and n_f is the number of objective function evaluations. The problems are numbered in the same way as in Hock and Schittkowski [27]. For example, HS6 means problem 6 in Hock and Schittkowski collection [27].

Table 3 shows the comparison of the best solution of our method in terms of the value of design variables and function value. The comparison of the OSRB algorithm against the Skinny method [31] and HOPSPACK [38] shows that our algorithm is more robust, because the number function evaluations are less than in most problems; also in HOPSPACK [38] some problems have no feasible solution, whereas our algorithm is always able to find a feasible point. These overall results suggest that the proposed OSRB can be considered an effective optimization technique for solving nonsmooth constrained optimization problems.

Table 3. Comparison of the results. The function values marked with '*' are infeasible solutions.

				This paper		Algorithm in [31]		HOPSPACK [38]	
Problem	n	$Eq.$	$Ineq.$	f	n_f	f	n_f	f	n_f
HS6	2	1	0	7.7849E-17	83	8.8304E-08	97	4.8400E+00*	151
HS7	2	1	0	-1.7321E+00	98	-1.7321E+00	150	6.93147E-01	325
HS8	2	2	0	-1.0000E+00	43	-1.0000E+00	56	-1.0000E+00*	187
HS9	2	1	0	-5.0000E+00	78	-5.0000E-01	109	-5.0000E-01	26
HS14	2	1	1	1.3935E+00	112	1.3935E+00	142	1.3941E+00	202
HS26	3	1	0	9.0579E-7	26	2.1160E+01	33	2.11600E+01	585
HS27	3	1	0	4.0000E+00	31	4.0000E+00	269	4.0006E+00	1358
HS28	3	1	0	1.2998E-23	29	3.6892E-27	43	7.7034E-08	264
HS32	5	1	1	1.0000E+00	178	1.0000E+00	209	1.0000E+00	51
HS39	4	2	0	-1.0000E+00	101	-1.0000E+00	302	-1.0000E+00	830
HS40	4	3	0	-2.5000E-01	128	-2.5000E-01	215	-2.5056E-01*	897
HS41	4	1	0	1.9259E+00	116	1.9259E+00	348	1.9259E+00	292
HS42	4	2	0	1.3869E+01	173	1.3858E+01	209	1.4000E+01	779
HS48	5	2	0	9.1339E-17	52	1.4960E-16	76	1.1174E-06	497
HS49	5	2	0	6.3236E-09	184	3.7388E-05	261	1.4294E-04	1002
HS50	5	3	0	2.7840E-07	103	7.7030E-06	246	5.2937E-07	290
HS51	5	3	0	5.4689E-17	64	8.4671E-17	88	1.2537E-06	142
HS52	5	3	0	5.3268E+00	191	5.3267+00	337	5.3267E+00*	311
HS53	5	3	0	4.0930E+00	234	400930E+00	295	4.09302E+00	216
HS55	6	6	6	6.3333E+00	96	6.3333E+00	223	-	-
HS61	3	2	0	-1.43462E+00	109	-1.4365E+00	196	-1.4300E+02	621
HS63	3	2	0	9.61736E+00	58	9.6172E+02	159	9.6261E+02	317
HS71	4	1	1	1.7032E+01	189	1.7014E+01	398	1.7031E+01*	1939
HS77	5	2	0	2.41495E-01	315	2.4151E-01	598	4.6807E+00*	1904

8. Conclusions

In this paper, we have proposed a smooth penalty-free algorithm based on the trust-region method for solving nonsmooth constrained optimization problems, and have discussed global convergence. The approach has been improved by a derivative-free local search phase in which the basis of the algorithm uses the radial basis functions. This algorithm does not use any penalty function and the constraint violation of the iterate points is controlled by trust-region radius. The trial step is accepted if the value of the objective function is sufficiently reduced. At each iteration, a surrogate model is constructed instead of the constraints by RBFs. The most significant advantage of the proposed algorithm is that the interpolation points can be managed easier in the system (11) to have a unique solution. We have tested a set of problems from [27]. The preliminary numerical simulations illustrate the effectiveness of the proposed method. Study of numerical experiments, especially for large scale optimization problems, is the aim of our future research.

References

[1] Arora JS. Introduction to Optimum Design. New York, NY, USA: McGraw-Hill, 1989.

[2] Belegundu AD. A Study of Mathematical Programming Methods for Structural Optimization. PhD, Department of Civil and Environmental Engineering, University of Iowa, Iowa, IA, USA, 1982.

[3] Benzi M, Golub GH, Liesen J. Numerical solution of saddle point problems. Acta Numerica 2005; 14: 1-137.

[4] Bhatti MA, Polak E, Pister KS. OPTDYNA General Purpose Optimization Program for Problems With or Without Dynamic Constraints. Technical Report UCB/EERC-79/16, University of California, Berkeley, 1979.

[5] Bjorkman M, Holmstrom K. Global optimization of costly nonconvex functions using radial basis functions. Optim Eng 2000; 4: 373-397.

[6] Booker AJ, Frank PD, Dennis Jr JE, Moore DW. Managing Surrogate Objectives to Optimize a Helicopter Rotor Design: Further Experiments. Proceedings of 8th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 1998.

[7] Buhmann MD. Radial basis functions: theory and implementations. Cambridge Monographs on Applied and Computational Mathematics 2003; 12: 147-165.

[8] Byrd RH. Robust trust region methods for constrained optimization. Third Siam Conference on Optimization, Houston, TX, USA, 1987.

[9] Byrd RH, Gilbert JC, Nocedal J. A trust region method based on interior point techniques for nonlinear programming. Math Program 2000; 89: 149-185.

[10] Chen ZW. A penalty-free-type nonmonotone trust-region method for nonlinear constrained optimization. Appl Math Comput 2006; 173: 1014-1046.

[11] Chen ZW, Han JY, Xu DC. A nonmonotone trust region algorithm for optimization with simplebound constraints. Appl Math Opt 2001; 43: 65-85.

[12] Chin CM, Fletcher R. On the global convergence of an SLP-filter algorithm that takes EQP steps. Math Program 2003; 96: 161-177.

[13] Coello CA, Montes EM. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. Adv Eng Inform, 2002; 3: 193-203.

[14] Coleman TF, Li A. An interior trust region approach for nonlinear minimization subject to bounds. Siam J Optim 1996; 6: 418-445.

[15] Conn AR, Gould NIM, Toint PhL. Global convergence of a class of trust region algorithms for optimization with simple bounds. Siam J Numer Anal 1998; 25: 433-460.

[16] Conn AR, Guold M, Toint PhL. Trust-Region Method, MPS-SIAM Series on Optimization. Siam, 2000.

[17] Conn AR, Scheinberg K, Toint PhL. On the convergence of derivative-free methods for unconstrained optimization. Approximation theory and optimization: tributes to MJD Powell 1986; 83-108.

[18] Conn AR, Scheinberg K, Toint PhL. Recent progress in unconstrained nonlinear optimization without derivatives. Math Program 1997; 79: 345-397.

[19] Conn AR, Scheinberg K, Toint PhL, Vicente LN. Geometry of interpolation sets in derivative free optimization. Math Program 2008; 111: 141-172.

[20] Conn AR, Scheinberg K, Vicente LN. Introduction to Derivative-Free Optimization. Siam, 2009.

[21] Conn AR, Toint PhL. An algorithm using quadratic interpolation for unconstrained derivative free optimization. Nonlinear Optimization and Applications 1996; 27-47.

[22] Dennis J, EL-Alem MM, Maciel MC. A global convergence theory for general trust region based algorithms for equality constrained optimization. Siam J Optim 1997; 7: 177-207.

[23] Dennis J, Heinkenschloss M, Vicente LN. Trust-region interior-point SQP algorithms for a class of nonlinear programming problems. Siam J Control Optim 1989; 36: 1750-1794.

[24] Dennis J, Vicente LN. On the convergence theory of trust-region-based algorithms for equality constrained optimization. Siam J Optim 1997; 7: 927-950.

[25] Fasshauer GE. Meshfree approximation methods with MATLAB. Singapore: World Scientific, 2007.

[26] Gould NIM, Toint PhL. Nonlinear programming without a penalty function or a filter. Math Program 2010; 122: 155-196.

[27] Hock W, Schittkowski K. Test examples for nonlinear programming codes, Lecture Notes in Economics and Mathematics Systems, No. 187, Berlin, Germany: Springer-Verlag, 1981.

[28] Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. J Global Optim 1989; 4: 455-492.

[29] Kurokawa N. Global convergence of the derivative free trust region Algorithm using inexact information a function values. PhD, Graduate School Of Informatics Kyoto University, 2009.

[30] Liu X, Yuan Y. A sequential quadratic programming method without a penalty function or a filter for nonlinear equality constrained optimization. Siam J Optim 2011; 21: 545-571.

[31] Martnez JM, Sobral FN. Constrained derivative-free optimization on thin domains. J Global Optim 2013; 3: 1217-1232.

[32] Micchelli CA. Interpolation of scattered data: distance matrices and conditionally positive definite functions. Constr Approx 1986; 2: 11-22.

[33] Nelder JA, Mead R. A simplex method for function minimization. Comput J 1965; 7: 308-313.

[34] Nocedal J, Wright S. Numerical Optimization. New York, NY, USA: Springer Science & Business Media, 2006.

[35] Omojokun FO. Trust region algorithms for optimization with nonlinear equality and inequality constrains. PhD, University of Colorado at Boulder, USA, 1989.

[36] Pillo GD. Exact penalty methods. In Algorithms for Continuous Optimization. Springer Netherlands, 1994.

[37] Plantenga TD. A trust region method for nonlinear programming based on primal interior point techniques. Siam J Sci Comput 1999; 20: 282-305.

[38] Plantenga TD. Hopspack 2.0 user manual. Sandia National Laboratories Technical Report Sandia National Laboratories Technical Report SAND 2009: 2009-6265. Harvard

[39] Powell MJD. UOBYQA: unconstrained optimization by quadratic approximation. Math Program 2002; 92: 555-582.

[40] Qeuvray R. Trust-Region Methods Based on Radial Basis Functions with Application to Biomedical Imaging. PhD, Ecole Polytechnique Fédérale de Lausanne, 2005.

[41] Regis RG, Shoemaker CA. A stochastic radial basis function method for the global optimization of expensive functions. Informs J Comput 2007; 19: 497-509.

[42] Ulbrich S. On the superlinear local convergence of a filter-SQP method. Math Program 2004; 100: 217-245.

[43] Vanderplaats GN, Moses F. Structural optimization by methods of feasible directions. Comput Struct; 1973, 3: 739-755.

[44] Vicente LN, Vicente IN, Badgwell A, Sorensen DC. Trust-region interior point algorithms for a class of nonlinear programming problems. 1996.

[45] Wendland H. Scattered Data Approximation. Cambridge, UK: Cambridge University Press, 2005.

[46] Winfield D. Function minimization by interpolation in a data table. Ima J Appl Math 1973; 12: 339-347.

[47] Yamashita H. A globally convergent quasi-newton method for equality without a penalty function for nonlinear constrained optimization. Technical Report, Mathematical Systems Inc., Tokyo, Japan, 1979.

[48] Yamashita H, Yabe H. A globally convergent trust-region SQP method without function. Technical Report, Mathematical Systems Inc., Tokyo, Japan, 2003.

[49] Zhao Z, Meza JC, Van Hove M. Using pattern search methods for surface structure determination of nanomaterials. J Phys Condens Matter 2006; 18: 8693-8706.

[50] Zoppke-Donaldson C. A tolerance-tube approach to sequential quadratic programming with applications. PhD, Department of Mathematics and Computer Science, University of Dundee, Dundee, Scotland, UK, 1995.