Journal of Applied Mathematics and Computing (2022) 68:2151–2172
https://doi.org/10.1007/s12190-021-01602-0

ORIGINAL RESEARCH

Adaptive trust region scheme for multi-objective optimization problem using Geršgorin circle theorem

Nantu Kumar Bisui[1] · Geetanjali Panda[1]

Received: 26 October 2020 / Revised: 10 July 2021 / Accepted: 14 July 2021 /
Published online: 2 August 2021

Abstract

In this article, a trust region algorithm is proposed to solve multi-objective optimization problem. A sequence of points is generated using Geršgorin Circle theorem with a modified secant equation. This sequence converges to a critical point of the problem. At every iteration, a common positive definite matrix is considered to take care of all the objective functions simultaneously and the radius of the trust region is obtained in an explicit form. Global convergence of the method is established with numerical support using a set of test problems.

Keywords Trust region method · Multi-objective optimization · Geršgorin Circle theorem · Pareto optimal solution · Critical point

Mathematics Subject Classification 49M37 · 65K05 · 65K10 · 90C26 · 90C29

1 Introduction

A general unconstrained multi-objective optimization problem is stated as

$$(MOP)\ \min_{x\in\mathbb{R}^n} F(x),$$

where $m \geq 2$, $F(x) = (f_1(x), f_2(x)\ldots, f_m(x))^T$, $f_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, 2, \ldots, m$. A point $x \in \mathbb{R}^n$ that minimizes all the objective functions simultaneously is an ideal minimum point of (MOP). This type of solution exists in rare cases when all the objective functions have the same local minimum point. In general, improvement in one objec-

✉ Geetanjali Panda
geetanjali@maths.iitkgp.ac.in

Nantu Kumar Bisui
nantukrbisuimath@gmail.com

[1] Department of Mathematics, Indian Institute of Technology Kharagpur, Kharagpur, India

tive function at any point results in other objective functions, causing trade-off among various objectives. Visualizing and resolving this trade-off is a key issue of (MOP). Due to the conflicting nature of the objective functions, the concept of optimality is replaced by the concept of Pareto optimality or efficiency. Scalarization methods and heuristic methods are traditional approaches to solve (MOP). The scalarization processes (see [6,16]) transform (MOP) to a single objective optimization problem with the help of pre-determined parameters. So these methods are user dependent, and often have difficulties in finding an approximation to the Pareto front. Heuristic methods [5] provide approximate Pareto front but do not guarantee the convergence property.

In recent years, many researchers have developed numerical approximation algorithms for (MOP) to overcome these difficulties and proved convergence property under different assumptions. The first few numerical approximation techniques for (MOP) are steepest descent method by Fliege and Svaiter [9], Newton's method by Fliege et al. [8]. The first one uses the linear approximation and the latter one uses the quadratic approximation of all objective functions. Newton's method requires the convexity assumption of each objective function. To avoid the convexity criterion, Qu et al. [17] proposed an extension of Quasi-Newton method for multi-objective optimization and Ansary and Panda ([1–3]) proposed a modified Quasi-Newton method for vector optimization.

Each of these algorithms is based on the line search techniques. Line search techniques and trust region techniques generate steps with the help of a subproblem in which the quadratic approximations of the objective functions are considered in different ways. Line search techniques generate a search direction by solving the subproblem, and then search for a suitable step length along this direction. In the trust region techniques, a region around the current iterate is defined within which one may trust the approximations as the sufficient depiction of the objective functions. This region is called the trust region. Then the step is chosen to be the approximate minimizer in this region. If a step is not acceptable, the size of the region is reduced and a new step is computed by solving the subproblem within the reduced region.

Recently, the classical trust region method for single objective optimization problems is extended to the multi-objective case by Qu, Goh, and Liang [18], in which a sequence $\{x^k\}$ is generated in $\mathbb{R}^n$, where $x^{k+1} = x^k + d^k,\ d^k \in \mathbb{R}^n$ with $||d^k|| \leq \Delta^k$, Δ^k is the radius of trust region, and $\{x^k\}$ converges to a critical point of (MOP). In this iterative process, the initial trust region radius is independent of Hessian of the objective functions. At every x^k, m number of positive definite matrices are computed corresponding to m objective functions using the BFGS update formula. As a result, there is a chance of an increase in the computational difficulty in the case of a very large number of objective functions. In this article, the following improvements are made to address these issues.

A common positive definite matrix is constructed at every x^k in place m positive definite matrices in the light of the concept of the author's previous work [1] in a different scenario. The BFGS update formula is modified accordingly and Geršgorin Circle theorem is used to ensure the positive definiteness of the common matrix, which is a new concept. The trust region radius at x^k is derived in explicit form. This does not depend on the trust region radius of the previous step directly.

This article is organized as follows: Some preliminaries on (MOP) are discussed in the next section. The methodology is developed and an algorithm is proposed in Sect. 3. In Sect. 4, global convergence of the algorithm is established. Numerical illustrations of the algorithm is provided in Sects. 5 and 6 has some concluding remarks. The differences and advantages of this methodology over the existing method are discussed towards the end of this section.

2 Preliminaries

In this section, some notations and preliminaries which will be used in the rest of the article are introduced. Denote $\mathbb{R}$ and $\mathbb{R}_+$ as the set of real numbers and the set of non-negative real numbers respectively. For $p, q \in \mathbb{R}^m$, the vector inequalities are considered as:

$$\begin{gathered} p = q \Leftrightarrow p_i = q_i \ \forall i \in \Lambda_m, \\ p \leq q \Leftrightarrow p_i \leq q_i \ \forall \, i \in \Lambda_m, \ \ p \neq q, \\ p < q \Leftrightarrow p_i < q_i \ \forall i \in \Lambda_m. \end{gathered}$$

Denote the non-negative orthant and the positive orthant of $\mathbb{R}^m$ as

$$\mathbb{R}^m_{++} := \{p \in \mathbb{R}^m | \ p > 0\} \text{ and } \mathbb{R}^m_{+} := \{p \in \mathbb{R}^m | \ p \geq 0\},$$

respectively and the index set as $\Lambda_m := \{1, 2, \dots, m\}$. For $x \in \mathbb{R}^n$, $||x||$ is the Euclidean norm of x and $||A|| = \max\limits_{x \in \mathbb{R}^n} \frac{||Ax||}{||x||}$, $x \neq 0$ is the norm of the matrix A of order $m \times n$. Throughout the article, assume that $F \in C^2(\mathbb{R}^n, \mathbb{R}^m)$, that is, F is twice continuously differentiable on $\mathbb{R}^n$. The symbols $\nabla f_j(x) \in \mathbb{R}^m$, $JF(x) \in \mathbb{R}^{m \times n}$ and $\nabla^2 f_j(x) \in \mathbb{R}^{n \times n}$ are used to denote the gradient of f_j, the Jacobian of F and the Hessian of f_j respectively at x. Associated with (MOP), some definitions and theorems that are used in this article are provided below.

Definition 1 A vector $d \in \mathbb{R}^n$ is said to be a descent direction of F at x if for any $j \in \Lambda_m$, $f_j^{'}(x; d) < 0$, where $f_j^{'}(x; d)$ is the directional derivative of f_j in the direction d, defined as $f_j^{'}(x; d) = \lim_{h \to 0} \frac{f_j(x+hd) - f_j(x)}{h}$

Since $F \in C^2(\mathbb{R}^n, \mathbb{R}^m)$, so it follows from the above definition that d is a descent direction of F at x if $\nabla f_j(x)^T d < 0 \ \ \forall j$.

Definition 2 (Ehrgott [6]) A point x^* is called (weak)Pareto optimal solution of (MOP) if there does exist any other x such that $F(x) \leq F(x^*)$ and $F(x) \neq F(x^*) (F(x) < F(x^*))$.

If E is the set of all Pareto optimal solutions then $F(E)$ is said to be the Pareto front of (MOP). Most of the numerical algorithms search for a local solution. The following definition deals with the local (weak)Pareto optimal.

Definition 3 (Fleige et al. [8]) A point x^* is said to be a local (weak)Pareto optimal solution of (MOP) if there is a neighborhood N of x^* such that there does not exist any other point $x \in N$ satisfying $F(x) \leq F(x^*)$ and $F(x) \neq F(x^*)(F(x) < F(x^*))$.

A necessary condition for x to be a local weak Pareto optimal solution of (MOP) is $Range(JF(x)) \cap (-\mathbb{R}^m_{++}) = \phi$. The following definition is motivated from this fact.

Definition 4 (Fleige et al. [8,9]) A point x^* is said to be a critical point of F if $Range(JF(x)) \cap (-\mathbb{R}^m_{++}) = \phi$

From the above definition, it is clear that if x^* is not a critical point of F then there exists a direction $d \in \mathbb{R}^n$ such that $JF(x^*)d \in (-\mathbb{R}^m_{++})$. In other words, $\nabla f_j(x^*)^T d < 0$ for every $j \in \Lambda_m$ ensures the existence of a descent direction at x^*. If x^* is critical point of F then $\forall\, d \in \mathbb{R}^n$, there exists at least one $j \in \Lambda_m$ such that $\nabla f_j(x^*)^T d \geq 0$. Note that criticality does not imply Pareto optimality (see [8]). There is a relation between criticality and local (weak) Pareto optimality, which is presented in the following theorem.

Theorem 1 (Fleige et al. [8]) *Let $F \in C^1(Q, \mathbb{R}^m)$, where $Q \subseteq \mathbb{R}^n$*

1. *If x^* is a locally weak Pareto optimal, then x^* is a critical point for F.*
2. *If Q is convex, F is $\mathbb{R}^m$- convex and $x^* \in Q$ is critical for F, then x^* is weak Pareto optimal.*
3. *If Q is convex, $F \in C^2(Q, \mathbb{R}^m)$, $\nabla^2 f_j(x)$ are positive definite for all $j \in \Lambda_m$ and for all $x \in Q$, and if $x^* \in Q$ is critical for F, then x is Pareto optimal.*

Theorem 2 *(Gordan's Theorem of alternative) Let A be an $m \times n$ matrix. Then exactly one of the following two systems has a solution:*

1. *$Ax < 0$ for some $x \in \mathbb{R}^n$*
2. *$A^T y = 0, \;\; y \geq 0$ for some non zero y.*

3 Methodology

In this section, a methodology is developed to generate an iterative sequence of the form $x^{k+1} = x^k + d^k$, which will converge to a critical point. The direction vector d^k is determined through a subproblem, associated with a region of trust.

3.1 Determination of a trial step

To compute a trust region trial step d, the following subproblem is considered at every iteration.

$$\min_{d} \max_{j \in \Lambda_m} \nabla f_j(x)^T d + \frac{1}{2} d^T B(x) d$$
$$\text{subject to } \; ||d|| \leq \Delta, \tag{1}$$

where $g_j(x) = \nabla f_j(x)$, $B(x)$ is assumed to be a symmetric positive definite matrix and $\Delta > 0$ is the trust region radius. This subproblem is equivalent to

$$\begin{gathered} P_{TR}(x, \Delta): \ \min_{d,t} \ t + \tfrac{1}{2} d^T B(x) d \\ \text{subject to } g_j(x)^T d \leq t, \ j \in \Lambda_m \\ ||d|| \leq \Delta, \end{gathered}$$

Let $(t(x, \Delta); d_{TR}(x, \Delta))$ be the optimal solution of $P_{TR}(x, \Delta)$ with optimal value $\theta(x, \Delta)$,

$$\begin{aligned} \theta(x, \Delta) &= \min_{t,d} \ \{t + \frac{1}{2} d^T B(x) d \mid g_j(x)^T d \leq t, \ ||d|| \leq \Delta\} \\ &= \min_{||d|| \leq \Delta} \max_{j \in \Lambda_m} g_j(x)^T d + \frac{1}{2} d^T B(x) d \end{aligned}$$

Since $B(x)$ is a positive definite matrix, so $P_{TR}(x, \Delta)$ is a convex programming problem with a unique solution. This implies that there exist Lagrange multipliers $\lambda \in \mathbb{R}^m_+$, $\alpha \in \mathbb{R}_+$ satisfying Karush-Kuhn-Tucker(KKT) optimality conditions. The Lagrange function of this problem is given by

$$L(t, d; \lambda, \alpha) = t + \frac{1}{2} d^T B(x) d + \sum_{j \in \Lambda_m} \lambda_j (g_j(x)^T d - t) + \alpha (d^T d - \Delta^2)$$

The KKT optimality conditions for $P_{TR}(x, \Delta)$ are

$$\sum_{j \in \Lambda_m} \lambda_j g_j(x) + B(x) d + 2\alpha d = 0, \tag{2}$$

$$\sum_{j \in \Lambda_m} \lambda_j = 1, \tag{3}$$

$$\lambda_j \geq 0, \ \ g_j(x)^T d \leq t, \tag{4}$$

$$\lambda_j (g_j(x)^T d - t) = 0 \ \forall j \in \Lambda_m, \tag{5}$$

$$\alpha \geq 0, \ d^T d \leq \Delta^2, \alpha (d^T d - \Delta^2) = 0. \tag{6}$$

For the sake of simplicity the following notations are used for developing theoretical results.

$$t := t(x, \Delta), \ d := d_{TR}(x, \Delta), \ \theta := \theta(x, \Delta)$$

Next two lemmas provide some properties of $P_{TR}(x, \Delta)$.

Lemma 1 *Suppose (t, d) be a solution of $P_{TR}(x, \Delta)$. If $d = 0$, then x is a critical point of (MOP).*

Proof Let $d = 0$. Then from (2) and (3) we have,

$$\sum_{j \in \Lambda_m} \lambda_j g_j(x) = 0, \ \sum_{j \in \Lambda_m} \lambda_j = 1$$

Now consider a matrix A of order $m \times n$, whose row vectors are $g_j(x), \ j \in \Lambda_m$. Then from the above relation, one can say that there exists $\lambda \in \mathbb{R}^m$ with $\lambda \geq 0$ such that $A^T\lambda = 0$ holds. Hence by Gordan's theorem of alternative, $Ad < 0$ has no solution d. This implies that for all $d \in \mathbb{R}^n$, $g_j(x)^T d \geq 0$ holds. Therefore x^* is a critical point of (MOP). □

Lemma 2 *Suppose* (t, d) *is a solution of* $P_{TR}(x, \Delta)$ *and there exists* $a > 0$ *such that* $z^T Bz \geq a||z||^2$ *holds for every* $z \in \mathbb{R}^n$*. Then* $t \leq -(a + 2\alpha)||d||^2$*, where* α *is a Lagrange multiplier. If* $d \neq 0$*, then* d *is a descent direction of* F*.*

Proof Since (t, d) is a solution of $P_{TR}(x, \Delta)$, the KKT optimality conditions (2)-(6) hold at (t, d). Taking the sum of (5) over j and using (3),

$$\sum_{j \in \Lambda_m} \lambda_j g_j(x)^T d - t = 0.$$

Operating both sides of (2) with d and then using $t = \sum_{j \in \Lambda_m} \lambda_j g_j(x)^T d$,
we have $d^T Bd + 2\alpha d^T d + t = 0$.
Since $z^T Bz \geq a||z||^2 \ \ \forall z$, so $t = -d^T Bd - 2\alpha d^T d \leq -(a + 2\alpha)||d||^2$. In this expression if $d \neq 0$ then $t < 0$. Hence from (4), $g_j(x)^T d < 0 \, \forall j$, which ensures that d is a descent direction. □

3.2 Adaptive BFGS Update using Geršgorin Circle Theorem

Here, a sequence of positive definite matrices $\{B(x^k)\}$ is generated with an initial symmetric positive definite matrix $B(x^0)$ starting at the initial point x^0. At the k^{th} iteration, denote

$$x^k := (x_1^k, x_2^k, \ldots, x_n^k)^T, \ d^k := d_{TR}(x^k, \Delta^k)$$
$$B^k := B(x^k), \ g_j^k := g_j(x^k), \ f_j^k := f_j(x^k)$$

Consider the following quadratic form in x.

$$f_j{}^{k+1} + \nabla f_j^{k+1^T} d + \frac{1}{2} d^T B^{k+1} d, \ \ j \in \Lambda_m$$

From (3) we have $\sum_{j \in \Lambda_m} \lambda_j = 1$. Consider the summation over j in the above form. Denote

$$m^{k+1}(x) \triangleq \sum_{j\in\Lambda_m} \lambda_j {f_j}^{k+1} + (x - x^{k+1})^T {g_j}^{k+1} + \frac{1}{2}(x - x^{k+1})^T B^{k+1}(x - x^{k+1})$$

Instead of $f_j,\ j \in \Lambda_m$, consider the function ${f_j}^k = f_j(x) + \frac{1}{2}(x - x^k)^T A^k(x - x^k)$ for all j at the k^{th} iteration, where A^k is some symmetric positive definite matrix. Taking the summation over j in ${f_j}^k$ and denote

$$m^k(x) \triangleq \sum_{j\in\Lambda_m} \lambda_j(f_j(x) + \frac{1}{2}(x - x^k)^T A^k(x - x^k)) \text{ (using(3))}$$

As in general BFGS update concept, B^{k+1} is a good approximation when $\nabla m^k(x^k) = \nabla m^{k+1}(x^k)$ and $\nabla m^k(x^{k+1}) = \nabla m^{k+1}(x^{k+1})$ These two relations can be simplified to

$$B^{k+1}(x^{k+1} - x^k) = \sum_{j\in\Lambda_m} \lambda_j(\nabla f_j^{k+1} - \nabla f_j^k)$$

and

$$\sum_{j\in\Lambda_m} \lambda_j \nabla f_j^{k+1} = \sum_{j\in\Lambda_m} \lambda_j \nabla f_j^{k+1} + A^k(x^{k+1} - x^k)$$

respectively. Hence

$$B^{k+1}d^k = v^k + A^k d^k = \gamma^k \text{ (say)}, \tag{7}$$

where $d^k = x^{k+1} - x^k$, $v^k = \sum\limits_{j\in\Lambda_m} \lambda_j(\nabla f_j^{k+1} - \nabla f_j^k)$. This is a new modified secant equation for (MOP). Suppose ${d^k}^T \gamma^k > 0$. Then using new modified secant equation (7) and following the steps of BFGS update formula for single objective optimization problem, a BFGS-type update formula is obtained as follows:

$$B^{k+1} = B^k - \frac{B^k d^k {d^k}^T B^k}{{d^k}^T B^k d^k} + \frac{\gamma^k {\gamma^k}^T}{{d^k}^T \gamma^k}$$

At the kth iteration, the functions f_j^k for all $j \in \Lambda_m$ are used with the following nice properties.

(i) At x^k, for all $j \in \Lambda_m$, ${f_j}^k(x^k) = f_j(x^k)$ and $\nabla {f_j}^k(x^k) = \nabla f_j(x^k)$. Hence $\nabla f_j(x^k)^T d^k < 0$ for all $j \in \Lambda_m$ implies $\nabla {f_j}^k(x^k)^T d^k < 0$. That is, if d^k is a descent direction of f_j for all j, then d^k is also a descent direction of ${f_j}^k$ for all $j \in \Lambda_m$ at x^k.

(ii) If $f_j \in C^2(\mathbb{R}^n, \mathbb{R}^m)$ and A^k is a symmetric positive definite matrix then for any $z \in \mathbb{R}^n$ with $z \neq 0$ we have $z^T(\nabla^2 {f_j}^k(x^k) - \nabla^2 f_j(x^k))z = z^T A^k z > 0$ for all j. Thus, at the iterative point x^k, ${f_j}^k$ has better curvature than that of f_j for every j.

Next, it is necessary to select a suitable matrix A^k so that the new matrix B^{k+1} is positive definite. Using the Taylor series expansion for each f_j about x^{k+1}

$$f_j(x) \simeq f_j^{k+1} + (x - x^{k+1})^T g_j^{k+1} + \frac{1}{2}(x - x^{k+1})^T \nabla^2 {f_j}^{k+1}(x - x^{k+1}).$$

At x^k,

$$f_j(x^k) = f_j^k \simeq f_j^{k+1} + (x^k - x^{k+1})^T g_j^{k+1} + \frac{1}{2}(x^k - x^{k+1})^T \nabla^2 {f_j}^{k+1}(x^k - x^{k+1}).$$

That is, ${d^k}^T \nabla^2 {f_j}^{k+1} d^k \simeq 2(f_j^k - f_j^{k+1}) + 2{d^k}^T g_j^{k+1}$, where $d^k = x^{k+1} - x^k$. Hence using (3), we get

$$ {d^k}^T \sum_{j \in \Lambda_m} \lambda_j^k \nabla^2 {f_j}^{k+1} d^k \simeq 2 \sum_{j \in \Lambda_m} \lambda_j^k (f_j^k - f_j^{k+1}) + 2 \sum_{j \in \Lambda_m} \lambda_j^k {d^k}^T g_j^{k+1} \tag{8}$$

Operating both sides of (7) by ${d^k}^T$,

$$ {d^k}^T B^{k+1} d^k = {d^k}^T \gamma^k = {d^k}^T \sum_{j \in \Lambda_m} \lambda_j^k (g_j^{k+1} - g_j^k) + {d^k}^T A^k d^k. \tag{9}$$

Both (8) and (9) indicate that a reasonable choice for A^k should satisfy the following relation.

$$ {d^k}^T A^k d^k = 2 \sum_{j \in \Lambda_m} \lambda_j^k (f_j^k - f_j^{k+1}) + \sum_{j \in \Lambda_m} \lambda_j^k {d^k}^T (g_j^{k+1} + g_j^k) = u^k \ (say).$$

At the kth iteration, if x^k is not the critical point then x^k can be updated to x^{k+1}. So $d^k = x^{k+1} - x^k \neq 0$. In that case, one possible solution of ${d^k}^T A^k d^k = u^k$ is $A^k = \frac{u^k}{{d^k}^T d^k} I$, where I is the identity matrix.

The positive definite condition ${d^k}^T \gamma^k > 0$ holds if ${d^k}^T v^k > 0$ and A^k is positive definite. $A^k = \frac{u^k}{{d^k}^T d^k} I$ is positive definite if $u^k > 0$, which is not always true. For $u^k \leq 0$, a matrix is constructed using Geršgorin Circle theorem so that B^{k+1} is positive definite.

Theorem 3 (Geršgorin Circle theorem [19]) *Let G be a complex matrix of order n, with entries a_{ij}. For $i \in \{1, 2, 3, \ldots, n\}$, let $R_i = \sum_{j \neq i} |a_{ij}|$ and $D(a_{ij}, R_i)$ be the closed disc centered at a_{ii} with radius R_i. Such a disc is known as Geršgorin disc. Every eigenvalue λ of G lies within at least one of the Geršgorin disc $D(a_{ij}, R_i)$, that is, $|\lambda - a_{ii}| \leq R_i$ for some i.*

Lemma 3 *At the kth iteration consider the matrix* $G^k = (a_{ij})_{n\times n}$, *where* $\delta > 0$ *a small number,*

$$a_{ij} = \begin{cases} \frac{u^k}{||d^k||^2}, & \textit{if } i \neq j \textit{ and } |i-j| = 1 \\ 2\frac{|u^k|}{||d^k||^2} + \delta, & \textit{if } i = j \\ 0, & \textit{otherwise.} \end{cases} \tag{10}$$

Then G^k *is positive definite.*

Proof Let $R_i = \sum_{j\neq i} |a_{ij}|, \forall i, j = 1, 2, \ldots, n$. From the construction of G^k, we have, $a_{ii} - R_i > 0$. Therefore by Geršgorin Circle theorem, the eigenvalues (say λ) of the matrix G^k satisfies the condition $|\lambda - a_{ii}| \leq R_i$, which implies $-(\lambda - a_{ii}) \leq R_i$. This implies that $\lambda \geq a_{ii} - R_i > 0$. Therefore all the eigenvalues of G^k are positive. Hence G^k is positive definite. □

Using this new positive definite matrix G^k, the updated matrix B^{k+1} can be determined as follows.

$$B^{k+1} = \begin{cases} B^k - \frac{B^k d^k d^{k^T} B^k}{d^{k^T} B^k d^k} + \frac{\gamma^k \gamma^{k^T}}{d^{k^T}\gamma^k}, & \text{if } d^{k^T} v^k > 0 \text{ and } u^k > 0 \\ (1-c_k)G^k + c_k B^k, & \text{otherwise,} \end{cases} \tag{11}$$

where $\{c_k\}$ is a real positive monotonically convergent sequence such that $\{c_k\} \to 1$ and $0 < c_0 < 1$.

3.3 Determination of trust region radius

Consider the sub-problem $P_{TR}(x, \varDelta)$. If $P_{TR}(x, \varDelta)$ is free from the restriction on trust region radius, that is, if $||d|| < \varDelta$ is replaced by $d \in \mathbb{R}^n$, then condition (2) becomes $\sum\limits_{j\in\varLambda_m} \lambda_j g_j(x) + B(x)d = 0$. So at the k^{th} iteration, $d^k = -B^{k^{-1}} \sum\limits_{j\in\varLambda_m} \lambda_j^k g_j^k$. In that case,

$$\begin{aligned} ||d^k|| &\leq ||B^{k^{-1}}|| || \sum_{j\in\varLambda_m} \lambda_j^k g_j^k || \\ &\leq ||B^{k^{-1}}||(\max_{j\in\varLambda_m} ||g_j^k||) \sum_{j\in\varLambda_m} \lambda_j^k \\ &= ||B^{k^{-1}}||(\max_{j\in\varLambda_m} ||g_j^k||) \end{aligned}$$

This relation provides information about an upper bound of $||d^k||$. So we may consider $\varDelta^k = ||B^{k^{-1}}||(\max\limits_{j\in\varLambda_m} ||g_j^k||)$.

The trust region trial step d^k is a solution to the subproblem $P_{TR}(x, \varDelta)$. After the computation of d^k, the improvement on the objective function can be measured through its actual reduction and predicted reduction from the subproblem. At the k^{th} iteration, the actual reduction is given by

$$Ared(d^k) = \max_{j \in \Lambda_m} \{f_j(x^k) - f_j(x^k + d^k)\}.$$

Let $\phi^k(d^k) = \max_{j \in \Lambda_m} g_j^{k^T} d^k + \frac{1}{2} d^{k^T} B^k d^k$. Then the predicted reduction, that is, the reduction according to our sub-problem $P_{TR}(x, \Delta)$ is

$$Pred(d^k) = \phi^k(0) - \phi^k(d^k) = -(\max_{j \in \Lambda_m} g_j^{k^T} d^k + \frac{1}{2} d^{k^T} B^k d^k).$$

Let $\rho_k(d^k) = \frac{Ared(d^k)}{Pred(d^k)}$. If $\rho_k(d^k)$ is close to 1, there is a good agreement between $Ared^k$ and $Pred^k$ over this step. In this situation, it is safe to expand the trust region for the next iteration. If $\rho_k(d^k)$ is positive but significantly smaller than 1, then the trust region remains unaltered. But if it is close to zero or negative then the trust region can be shrunk by reducing Δ^k at the next iteration. Using this logic, the trust region radius Δ^k can be updated, which is explained in the algorithm of the next subsection.

Lemma 4 *Suppose* $F \in C^2(\mathbb{R}^n, \mathbb{R}^m)$, *for all* $j \in \Lambda_m$, $g_j(x)$ *and* $B(x)$ *are bounded, then at the* k^{th}*iteration,* $|Ared(d^k) - Pred(d^k)| \le \delta_1 ||d^k|| + \delta_2 ||d^k||^2$ *for some* $\delta_1, \ \delta_2 > 0$.

Proof

$$\begin{aligned} Ared(d^k) - Pred(d^k) &= \max_{j \in \Lambda_m} (f_j(x^k) - f_j(x^k + d^k)) + \max_{j \in \Lambda_m} g_j^{k^T} d^k \\ &\quad + \frac{1}{2} d^{k^T} B^k d^k \\ \Rightarrow |Ared(d^k) - Pred(d^k)| &= |\max_{j \in \Lambda_m} (f_j(x^k) - f_j(x^k + d^k))| + |\max_{j \in \Lambda_m} (g_j^{k^T} d^k)| \\ &\quad + \frac{1}{2} ||B^k|| ||d^k||^2 \\ &\le \max_{j \in \Lambda_m} |(f_j(x^k) - f_j(x^k + d^k))| + \max_{j \in \Lambda_m} |(g_j^{k^T} d^k)| \\ &\quad + \frac{1}{2} ||B^k|| ||d^k||^2 \qquad (12) \end{aligned}$$

Using Taylor series expansion of first order, we have,

$$f_j(x^k + d^k) = f_j(x^k) + g_j(\xi)^T d^k,$$

where ξ is an interior point in the line segment joining x^k and $x^k + d^k$. Hence from (12), we get,

$$\begin{aligned}|Ared(d^k) - Pred(d^k)| &\leq \max_{j\in\Lambda_m} |g_j(\xi)^T d^k| + \max_{j\in\Lambda_m} |g_j^{k^T} d^k| + \frac{1}{2}||B^k||||d^k||^2 \\ &\leq (\max_{j\in\Lambda_m} ||g_j(\xi)|| + \max_{j\in\Lambda_m} ||g_j^k||)||d^k|| + \frac{1}{2}||B^k||||d^k||^2\end{aligned}$$

Hence the lemma follows, since $g_j(x)$ and $B(x)$ are bounded for all $j \in \Lambda_m$. □

Define

$$v(x, \Delta) = \sup_{||d||\leq\Delta} \{-\max_{j\in I_m} g_j(x)^T d\} \tag{13}$$

Then the following lemma holds.

Lemma 5 *1.* $v(x, \cdot)$ *is non negative.*
2. x *is a critical point of* (MOP) *if and only if* $v(x, \Delta) = 0$ *for some* $\Delta > 0$.

Proof 1. From the definition of $v(x, \Delta)$, the result is obvious.
2. Let $v(x, \Delta) = \sup_{||d||\leq\Delta} \max_{j\in\Lambda_m} g_j(x)^T d = 0$.
Then for all d satisfying $||d|| \leq \Delta$, $\max_{j\in\Lambda_m} g_j(x)^T d = 0$. This implies that there exists a $j \in \Lambda_m$ such that $g_j(x)^T d = 0$. Hence x is a critical point.
Conversely let x is a critical point of (MOP). Then by the definition of critical point, there exists a $j \in \Lambda_m$ such that $g_j(x)^T d \geq 0$ for all d. This implies that there exists some Δ such that $\sup_{||d||\leq\Delta} \{-\max_{j\in I_m} g_j(x)^T d\} \leq 0$. That is, $v(x, \Delta) \leq 0$.
But from the first part, we have, $v(x, \Delta) \geq 0$. Hence the result follows.

□

Lemma 6 *Suppose there is a constant* $M > 0$ *such that* $||B(x)|| \leq M$. *Then at the* k^{th} *iteration* $Pred(d^k) \geq \frac{1}{2}\frac{\{v(x^k,\Delta^k)\}^2}{M{\Delta^k}^2}$.

Proof From the definition of predicted reduction, we have at the k^{th} iteration,

$$\begin{aligned}Pred(d^k) &= -\max_{j\in\Lambda_m} g_j^{k^T} d^k - \frac{1}{2}d^{k^T} B^k d^k \\ &\geq -\max_{j\in\Lambda_m} g_j^{k^T} d^k - \frac{M||d^k||^2}{2}\end{aligned}$$

Let $\bar{d}^k$ be the maximum in (13). Then for any $\eta \in [0, 1]$, we have that

$$\begin{aligned}Pred(d^k) &\geq \max_{\eta\in[0,1]} [-\max_{j\in\Lambda_m} g_j(x^k)^T (\eta\bar{d}^k) - \frac{M||\eta\bar{d}^k||^2}{2}] \\ &\geq \max_{\eta\in[0,1]} [\eta(-\max_{j\in\Lambda_m} g_j(x^k)^T \bar{d}^k) - \eta^2\frac{M{\Delta^k}^2}{2}]\end{aligned}$$

$$\geq \max_{\eta\in[0,1]} [\eta v(x^k, \Delta^k) - \frac{\eta^2}{2} M {\Delta^k}^2]$$
$$\geq \frac{1}{2} \frac{\{v(x^k, \Delta^k)\}^2}{M {\Delta^k}^2}$$

Hence the lemma. □

3.4 Algorithm

The concepts developed in Subsections 3.1 to 3.3 are summarized in the following steps.
Step 1. Let $x^0 \in \mathbb{R}^n$, $0 < c_0 < 1$, error of tolerance $\epsilon > 0$. Choose the parameters $0 < \alpha_1 \leq \alpha_2 < 1, 0 < \beta_1 < 1 < \beta_2$, a positive definite matrix. Set $k = 0$. Compute Δ^k
Step 2. Solve the sub-problem $P_{TR}(x^k, \Delta^k)$. Find the value of d^k, t^k.
Step 3. If $||d^k|| < \epsilon$ then stop. Otherwise compute

$$\rho_k(d^k)) = \frac{Ared(d^k)}{Pred(d^k)} = \frac{\max_{j\in\Lambda_m} \left\{f_j(x^k) - f_j(x^k + d^k\right\}}{\phi^k(0) - \phi^k(d^k)}$$

Step 4. If $\rho_k(d^k) < \alpha_1$, set $\Delta^k = \beta_1 \Delta^k$ and go to Step 2. Otherwise set $x^{k+1} = x^k + d^k$.
Step 5. Generate a positive definite matrix B^{k+1} using the formula (11). Compute Δ^k as

$$\Delta^{k+1} = \begin{cases} \beta_1 ||{B^{k+1}}^{-1}|| \max_{j\in\Lambda_m} ||{g_j}^{k+1}||, & \text{if } \alpha_1 \leq \rho_k(d^k) \leq \alpha_2 \\ \beta_2 ||{B^{k+1}}^{-1}|| \max_{j\in\Lambda_m} ||{g_j}^{k+1}||, & \text{otherwise.} \end{cases} \tag{14}$$

Set $k = k + 1$, compute c_{k+1}, Go to Step 2.

4 Convergence analysis

The following theorem guarantees that Algorithm 3.4 does not cycle infinitely in the inner cycle.

Theorem 4 *Suppose that $\{B^k\}$ is bounded and the sequence $\{x^k\}$ is generated by Algorithm 3.4. Then Algorithm 3.4 does not cycle infinitely in the inner cycle Step 2-Step 3-Step 4-Step 2.*

Proof Suppose at the kth iteration, the inner cycle of the algorithm goes infinitely. Then the algorithm will not terminate at x^k, that is, $v(x^k, \Delta^k) > 0$. Let n_k be the number of iterative steps of the inner iteration. Then $\rho_k(d^k_{n_k}) < \alpha_1$. In this situation, $\Delta^k_{n_k}$ is updated by Step 5 of the algorithm. Since at every step in the inner cycle, the trust region is shrinking, so $\Delta^k_{n_k} \rightarrow 0$ as $n_k \rightarrow \infty$. Hence $||d^k_{n_k}|| \leq \Delta^k_{n_k} \rightarrow 0 \ \ as \ n_k \rightarrow \infty$. From Lemma 4 and Lemma 6,

$$|\rho_k(d^k_{n_k}) - 1)| = |\frac{Ared^k - Pred^k}{Pred^k}| \leq \frac{\delta_1||d^k_{n_k}|| + \delta_2||d^k_{n_k}||^2}{\frac{1}{2}\frac{\{v(x^k, \Delta^k)\}^2}{M{\Delta^k}^2}}$$

As $n_k \rightarrow \infty$, $|\rho_k(d^k_{n_k}) - 1)| \rightarrow 0$. Hence $\rho_k(d^k_{n_k}) \rightarrow 1$. This implies that there exists α_1 such that for sufficiently large n_k, $\rho_k(d^k_{n_k}) \geq \alpha_1$. This contradicts the relation $\rho_k(d^k_{n_k}) < \alpha_1$. Hence the lemma follows. □

Theorem 5 *Suppose that $\Omega_0 = \{x \in \mathbb{R}^n | F(x) \leq F(x^0)\}$ is bounded, $x^0 \in \mathbb{R}^n$ is the initial point, and $F \in C^1(\mathbb{R}^n, \mathbb{R}^m)$. $\{x^k\}$ is generated by Algorithm 3.4 and the sequence $\{B^k\}$ is bounded. Then every accumulation point of $\{x^k\}$ is a critical point of (MOP).*

Proof Let d^k be the accepted step at the kth iteration. Then from Algorithm 3.4, we have $\rho_k(d^k) \geq \alpha_1$. Denote $\Gamma = \{k \mid \rho_k(d^k) \geq \alpha_1\}$.
Since Ω_0 is a bounded set, $\{x^k\}$ has at least one accumulation point. Without loss of any generality, suppose that the subsequence $\{x^k\}_{k \in \Gamma}$ converges to x^*.

Since $\rho_k(d^k) \geq \alpha_1$, so

$$\alpha_1 Pred(d^k) \leq Ared(d^k) = \max_{j \in \Lambda_m}\{f_j(x^k) - f_j(x^k + d^k)\}.$$

Hence

$$\begin{aligned}\sum_{k \in \Gamma} \alpha_1 Pred(d^k) &\leq \sum_{k \in \Gamma} \max_{j \in \Lambda_m}\{f_j(x^k) - f_j(x^{k+1})\} \\ &= \sum_{k \in \Gamma} (f_{j_0}(x^k) - f_{j_0}(x^{k+1})) \text{ (for some } j_0 \in \Lambda_m) \\ &= f_{j_0}(x^0) - f_{j_0}(x^1)) + f_{j_0}(x^1) - f_{j_0}(x^2) + \ldots + f_j(x^k) \\ &- f_j(x^{k+1})\} + \ldots \\ &\leq \max_{j \in \Lambda_m}[f_j(x^0) - f_j(x^*)] < +\infty \qquad (15)\end{aligned}$$

From Lemma 6, we have for all $\Delta > \Delta^k$, $k \in \Gamma$,

$$\sum_{k \in \Gamma} \frac{1}{2} \frac{\{v(x^k, \Delta)\}^2}{M\Delta^2} \leq \sum_{k \in \Gamma} Pred(d^k)$$

Using the above inequality along with (15) we get,

$$\sum_{k \in \Gamma} \frac{\alpha_1}{2} \frac{\{v(x^k, \Delta)\}^2}{M\Delta^2} < +\infty \qquad (16)$$

Lemma 5 indicates that it is enough if we show that x^* is a solution of $v(x, \Delta)$ for some $\Delta > 0$. If possible let $v(x^*, \Delta) > 0$. This implies, there exists $\mu, \varepsilon_0 > 0$ such that for all $0 < \varepsilon \leq \varepsilon_0$ and for all $||x^k - x^*||_{k\in\Gamma} \leq \varepsilon$, $v(x^k, \Delta) \geq \mu > 0$.
Therefore

$$\sum_{k\in\Gamma} \frac{\{v(x^k, \Delta)\}^2}{M\Delta^2} \geq \sum_{k\in\{k|||x^k - x^*||_{k\in\Gamma} \leq \varepsilon\}} \frac{\mu^2}{M\Delta^2} = +\infty$$

This contradicts (16). Therefore $v(x^*, \Delta) = 0$, and x^* is a critical point. □

5 Numerical experiment and result analysis

In this section, MATLAB implementation of Algorithm 3.4 is executed on many test problems with bound constraints $lb \leq x \leq ub$, where $lb,\ ub \in \mathbb{R}^n$. Details of these test problems are summarized in Table 1. In that table, m and n denote the number of objective functions and number of variables respectively. The results obtained by Algorithm 3.4(in short term ATRMO) are compared with the Algorithm 1(in short TRMO)of [18]. Solution of (MOP) is a well-distributed set of Pareto optimal solutions. Spreading out an approximation to a Pareto set is a difficult problem. There is no single spreading technique that can work in a satisfactory manner for (MOP). In this article, a set of initial points is selected with the strategy *rand* to generate an approximated Pareto front. In this strategy, 100 random initial point are selected, which are uniformly distributed in lb and ub. Every test problem of Table 1 is executed 10 times with these random initial points.

The number of iterations, number of function counts, and number of gradient counts are recorded to compare ATRMO with TRMO, which are summarized in Table 2. Comparison between the performance of the algorithms is done using these factors.

Performance profiles Performance profiles are defined by a cumulative function $\bar{\rho}_s(\tau)$ representing a performance ratio with respect to a given metric, for a given set of algorithms. Given a set of problems P and a set of algorithms S, let the metric $\zeta_{p,s}$ be the performance of solver s on solving problem p. The performance ratio is then defined as $r_{p,s} = \zeta_{p,s}/\min\limits_{s\in S} \zeta_{p,s}$. The cumulative function $\bar{\rho}_s(\tau)$ $(s \in S)$ is defined as the percentage of problems whose performance ratio is less than or equal to τ, that is,

$$\bar{\rho}_s(\tau) = \frac{|\{p \in P : r_{p,s} \leq \tau\}|}{|P|}.$$

In multi-objective optimization, the output is a set of non-dominated points providing as an approximation of the full Pareto set. In this article, the following metrics are considered to compare between ATRMO and TRMO: the *Purity* metric, two *Spread* metrics.

Purity metric The purity metric is used to measure how many non-dominated points an algorithm is able to compute. The approximated Pareto front of problem p obtained by algorithm s is denoted by $F_{p,s}$. By considering the union of all individual Pareto

Table 1 Multi-objective test problems with bound constraints

Problem	Source	m	n	Problem	Source	m	n
BK1	[11]	2	2	Jin2	[13]	2	2
Deb41	[4]	2	2	Jin3	[13]	2	3
Deb513	[4]	2	2	Jin4	[13]	2	2
Deb521aa	[4]	2	2	lovison1	[15]	2	2
Deb521b	[4]	2	2	lovison2	[15]	2	2
DG01	[11]	2	1	lovison3	[15]	2	2
ex005	[12]	2	2	lovison4	[15]	2	2
LTDZ	[14]	3	5	LRS1	[11]	2	10
Fonseca	[10]	2	2	MHHM1	[11]	3	1
GE2	[7]	2	2	MHHM2	[11]	3	2
GE5	[7]	3	3	MLF1	[11]	2	1
IKK1	[11]	2	2	MLF2	[11]	2	2
IM1	[11]	2	2	SP1	[11]	2	2
TKLY1	[11]	2	2	MOP1	[11]	2	1
Far1	[11]	2	2	MOP3	[11]	2	2
FDS2	[8]	3	3	MOP5	[11]	3	2
FDS3	[8]	3	5	MOP6	[11]	2	2
KW2	[7]	2	2	MOP7	[11]	3	2
PNR1	[8]	2	2	SSFYY1	[11]	2	4
PNR2	[8]	2	2	SSFYY2	[11]	2	1
PNR3	[8]	2	2	VU1	[11]	2	2
PNR4	[8]	2	2	VU2	[11]	2	2
PNR5	[8]	2	2	ZDT1	[2]	2	2
SK1	[11]	2	1	ZDT2	[2]	2	2
SK2	[11]	2	4	ZDT3	[2]	2	2
VFM1	[11]	3	2	ZDT4	[2]	2	2
Jin1	[13]	2	2	ZDT6a	[2]	2	2

approximation $\bigcup\limits_{s\in S} F_{p,s}$, where the dominated points are removed, an approximation to the Pareto front F_p is achieved. The purity metric for algorithm s and problem p is defined by the ratio $\tilde{r}_{p,s} = \frac{|F_{p,s}\cap F_p|}{|F_p|}$. When computing the performance profiles for the purity metric it is required to set $r_{p,s} = 1/\tilde{r}_{p,s}$, since higher values of $\tilde{r}_{p,s}$ indicates that the algorithm gives a higher percentage of non-dominated points for problem p.

Spread metric In order to examine whether the points generated by algorithm s for problem p are well distributed over the approximated Pareto front, two spread metrics $\varGamma$ and $\varDelta$ are considered. The largest gap in the approximated Pareto front is measured by the $\varGamma$ metric, while the scaled deviation from the average gap in the approximated Pareto front is measured by the $\varDelta$ metric. In the following, how to compute these metrics is described. Let the set of N points $x^1, x^2, \ldots, x^N$ forms the approximated Pareto front obtained by algorithm s for problem p. Furthermore, let x^0 and x^{N+1} be the extreme points for objective j, computed over all the approximated

Table 2 Computation details of the test problems

ATRMO				TRMO		
Problem	It	F_c	G_c	It	F_c	G_c
BK1	2.38	6.22	3.37	4.50	6.49	5.49
Deb521b	2.25	3.34	3.24	2.06	4.18	3.05
Deb513	2.18	6.99	3.17	1.90	3.40	2.89
Deb521aa	2.78	3.68	3.77	2.76	6.13	3.75
Deb41	4.64	9.47	5.63	7.67	10.58	8.66
DG01	1.88	4.14	2.87	1.95	3.95	2.94
ex005	2.36	9.61	3.35	2.67	4.65	3.66
Fonseca	35.53	38.51	36.52	39.20	47.56	40.19
GE2	34.03	35.91	34.96	51.99	63.14	52.98
GE5	2.54	10.52	3.53	5.65	12.13	6.64
IKK1	2.22	4.06	3.20	6.58	10.05	7.57
IM1	2.67	4.62	3.66	2.44	3.94	3.43
Jin1	2.00	3.99	2.99	3.07	5.06	4.06
Jin2	1.89	3.10	2.88	1.89	3.36	2.88
Jin3	2.08	7.33	3.07	1.83	3.22	2.82
Jin4	1.95	3.56	2.94	1.84	3.16	2.83
lovison1	2.50	6.59	3.49	2.25	4.91	3.24
lovison2	2.45	3.91	3.44	2.38	3.98	3.37
lovison3	1.95	3.51	2.94	2.82	4.72	3.81
lovison4	2.78	4.37	3.77	4.01	5.70	5.00
LRS1	3.11	6.34	4.09	12.87	33.79	13.86
MHHM1	2.33	3.69	3.32	1.55	3.54	2.54
MHHM2	2.88	4.87	3.87	2.72	4.72	3.71
MLF1	0.68	2.67	1.67	1.07	3.06	2.06
MLF2	8.72	11.08	9.71	17.19	22.13	18.18
MOP1	1.25	3.43	2.24	51.00	53.04	51.99
MOP3	3.52	5.46	4.51	4.63	6.40	5.62
MOP5	12.56	19.69	13.51	24.43	31.59	25.42
MOP6	1.96	3.98	2.94	1.92	3.30	2.91
MOP7	9.02	13.29	10.01	28.15	37.81	29.14
VFM1	1.84	3.83	2.83	1.76	3.75	2.75
SP1	5.18	8.86	6.17	4.30	6.16	5.29
SSFYY1	3.20	10.0	3.88	16.82	37.99	17.81
SSFYY2	2.53	4.79	3.52	13.87	15.83	14.85
SK1	0.18	2.11	1.14	0.47	2.42	1.44
SK2	11.25	17.07	12.24	13.08	15.79	14.07

Table 2 continued

ATRMO				TRMO		
Problem	It	F_c	G_c	It	F_c	G_c
VU1	11.67	13.41	12.66	78.27	91.58	79.26
VU2	12.50	115.93	13.49	4.40	6.04	5.39
KW2	10.22	13.15	11.21	12.72	15.67	13.71
LTDZ	2.24	3.93	3.23	1.98	3.62	2.97
PNR1	4.10	9.82	5.09	4.83	7.26	5.82
PNR2	3.90	7.39	4.89	5.30	7.31	6.29
PNR3	3.80	9.09	4.79	4.95	6.88	5.94
PNR4	4.18	7.98	5.17	5.39	7.52	6.38
PNR5	4.17	7.92	5.16	5.60	7.36	6.59
TKLY1	5.28	9.59	6.27	9.16	13.72	10.15
FDS2	6.96	9.27	7.95	6.46	9.41	7.45
FDS3	14.03	20.16	15.02	11.48	17.29	12.47
FAR1	14.60	17.85	15.59	57.48	66.55	58.47
ZDT1	2.27	4.26	3.26	1.88	3.52	2.87
ZDT2	1.89	3.24	2.88	1.87	3.25	2.86
ZDT3	2.13	3.48	3.12	1.99	3.39	2.98
ZDT4	1.91	3.05	2.90	1.76	3.42	2.75
ZDT6a	6.53	9.54	7.52	8.29	11.23	9.28

Pareto fronts obtained by different algorithms. Suppose that those N points are sorted by the j^{th} objective function, that is, $f_j(x^i) \leq f_j(x^{i+1})$ $(i = 1, 2, \ldots, N)$. Define $\delta_{i,j} = |f_j(x^{i+1}) - f_j(x^i)|$, and suppose that $\tilde{\delta_j}(j \in \Lambda_m)$ be the average of the distances $\delta_{i,j}$. Then $\Gamma > 0$ and $\Delta > 0$ spread metrics are defined as

$$\Gamma_{p,s} = \max_{j \in \Lambda_m} \max_{i \in \{0,1,\ldots N\}} \delta_{i,j}$$

and

$$\Delta_{p,s} = \max_{j \in \Lambda_m} \left(\frac{\delta_{0,j} + \delta_{N,j} + \sum_{i=1}^{N-1} |\delta_{i,j} - \tilde{\delta_j}}{\delta_{0,j} + \delta_{N,j} + (N-1)\tilde{\delta_j}}\right)$$

Implementation details MATLAB(2016a) code is developed for both the algorithms ATRMO and TRMO. Implementation of the algorithms are described below:

- Initial points selection strategy *rand* is used for every test problem. An initial matrix $B^0 = I$, parameters $c_0 = 0.5, \alpha_1 = 0.2, \alpha_2 = 0.5, \beta_1 = 0.5, \beta_2 = 1.2$ are accepted and the sequence $\{c_k\}$, where $c_{k+1} = 1 - c_k^i,$ i is the number iteration, is considered.
- Subproblems of both the algorithms are solved using MATLAB function ‘fmincon’ with -‘Algorithm’,‘interior-point’, -‘initial guess’,0^{n+1}.

- For both the algorithms, stopping criteria is used as $||d^k|| \leq 10^{-5}$ or maximum 200 iterations.
- Every test problem is executed 10 times with random initial points and average is considered for performance profile. Computational details of these average values for both algorithms are presented in Table 2. In this table, 'It' corresponds to the number of iterations, 'F_c' corresponds to the number of function counts and 'G_c' corresponds to the number of gradient counts.
- The gradient is computed using forward difference formula, which needs n additional function evaluations. Therefore the total function evaluations(F_e) for 100 initial points for a test problem of m objective functions are obtained as $F_e = mF_c + mnG_c$.

Explanation with one test problem Steps of Algorithm 3.4 are explained in one numerical example. Consider the following bi-objective optimization problem in two dimensions.

$$\min_{x \in \mathbb{R}^n} \{f_1(x), f_2(x)\},$$

where

$$f_1(x) = \frac{1}{4}[(x_1-1)^4 + 2(x_2-2)^4], \quad f_2(x) = (x_2 - x_1^2)^2 + (1-x_1)^2.$$

Here f_2 is not a convex function.
Consider the initial point $x^0 = (0, 5)^T$, the initial matrix $B^0 = I_2$, parameters $c_0 = 0.5$, $\alpha_1 = 0.2$, $\alpha_2 = 0.5$, $\beta_1 = 0.5$, $\beta_2 = 1.2$. Stopping condition is $||d^k|| < 10^{-5}$. $f(x^0) = (40.75, 26.00)$ and $\Delta^0 = \max\{||g_1(x^0)||, ||g_2(x^0)||\} = 54.0095$. Now clearly assumptions of Theorem 5 are satisfied at $x^0 = (0, 5)^T$. So the sub-problem $P_{TR}(x^0, \Delta^0)$ is solved using MATLAB (R2016a) function "fmincon" with -'Algorithm', 'interior-point', -'initial guess',0^{n+1}, and the solution is obtained as $t^0 = -104.0006$, $d^0 = (2.0001, -10.0000)^T$. Since $||d^0|| > 10^{-5}$, so $\rho_0(d^0) = \frac{Ared(d^0)}{Pred(d^0)} = -1.0770$ is computed, which is less than α_2. So according to Algorithm 3.4, the subproblem is solved again with $\Delta^0 = 27.0048$. After solving the subproblem three times, $\rho_0(d^0) = 0.3152$ is obtained. So we proceed to Step 5. Hence the matrix B^1 is computed using (11).

$$B^1 = \begin{pmatrix} 3.6891 & -2.4898 \\ -2.4898 & 2.6758 \end{pmatrix}$$

Since $\rho_0(d^0) = 0.3152 < \alpha_2$, so trust region radius is updated as

$$\Delta^1 = \beta_1 ||B^{1^{-1}}|| \max\{||g_1(x^1)||, ||g_2(x^1)||\} = 30.7087.$$

Then the sub-problem $P_{TR}(x^1, \Delta^1)$ is solved. Repeating the above process, the critical point of this problem is obtained after $15th$ iteration and the solution is $(1, 1)^T$.

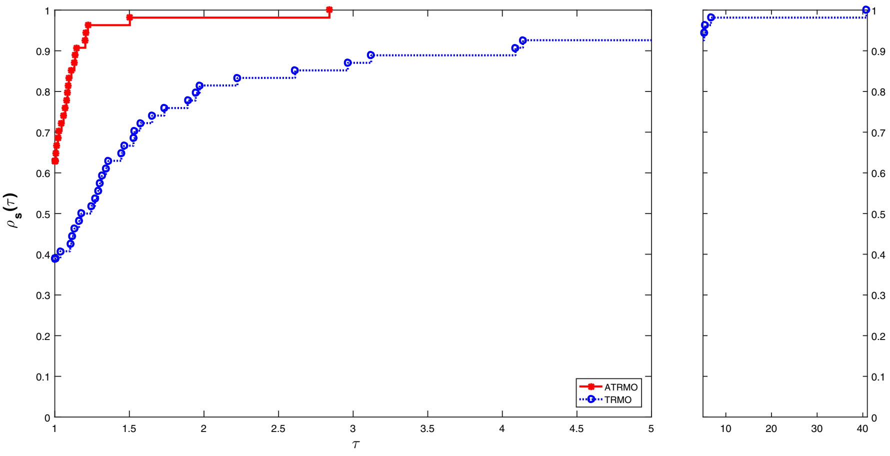

Fig. 1 Performance profile for It

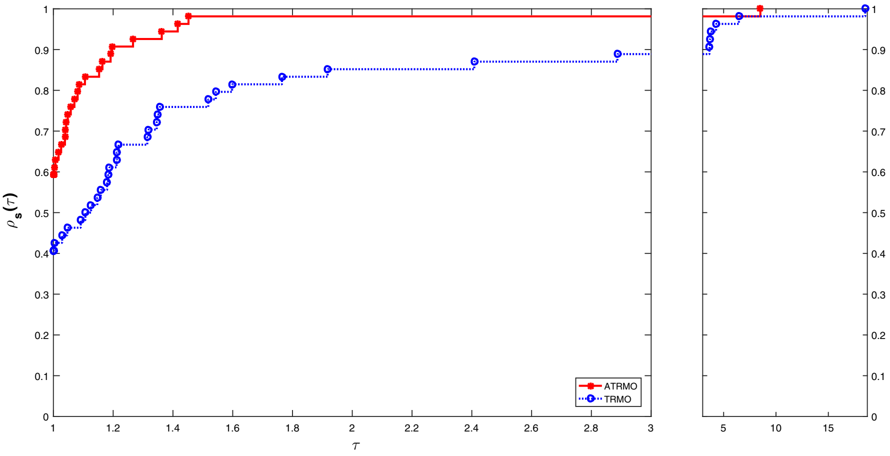

Fig. 2 Performance profile for F_e

Comparison using performance profiles Comparison of ATRMO and TRMO using performance profiles for the number of iterations(It) and the number of function evaluations(F_e) are provided in Figs. 1 and 2. In *rand*, an average of 10 purity metric values, obtained in 10 different runs is denoted by the average purity metric value. Similarly, average Γ and Δ metric values are computed. The average purity performance profile between ATRMO and TRMO is provided in Fig. 3. Figures 4 and 5 represent the average Γ and Δ performance profile respectively between ATRMO and TRMO.

Result Analysis Figures 1-4 indicate that the proposed scheme ATRMO provides better performance than TRMO in most test problems. Also one may observe from Table 2 that ATRMO takes fewer iterations and function evaluations and gradient evaluations than TRMO in the most number of cases.

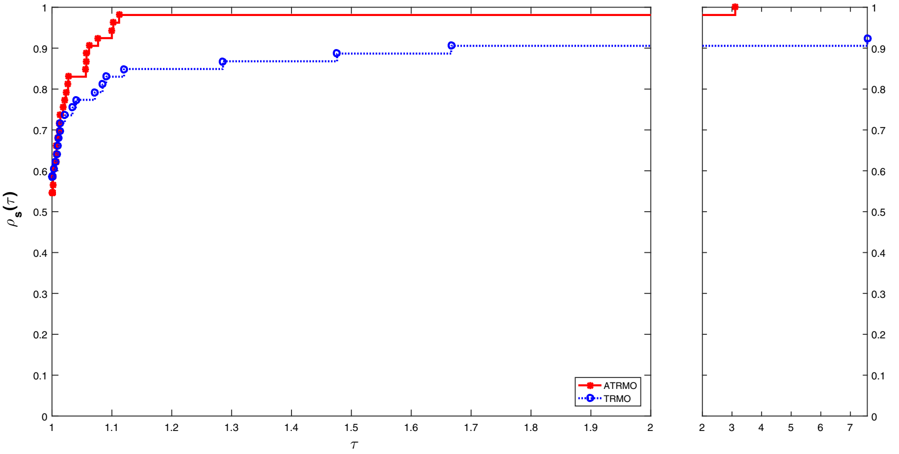

Fig. 3 Average purity performance profile between ATRMO and TRMO

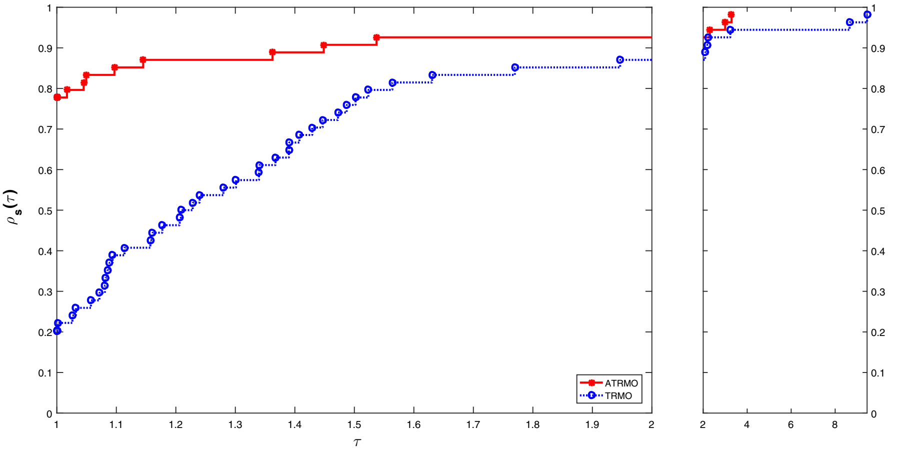

Fig. 4 Average Γ performance profile between ATRMO and TRMO

6 Conclusion

In this article, a new algorithm is proposed for (MOP) and the global convergence of the proposed algorithm is proved. The differences and advantages of this method over the previous existing method are summarized below.

(i) In the existing method [18], the positive definite matrix B_j^k is computed for every objective function, whereas the proposed scheme computes a common positive definite matrix B^k at every iteration for all objective functions. This provides a new BFGS-type update formula.

(ii) Geršgorin Circle theorem is used to ensure the positive definiteness of the matrix in BFGS update formula, which is a new concept in multi-objective programming.

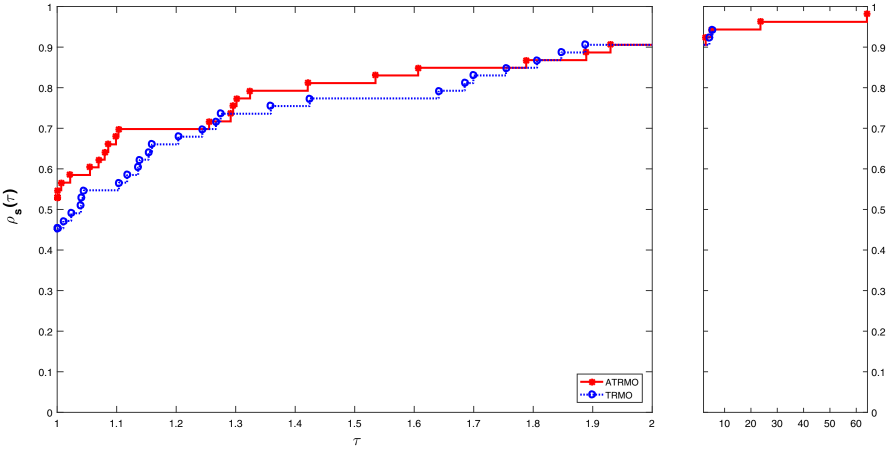

Fig. 5 Average Δ performance profile between ATRMO and TRMO

(iii) The trust region radius of this method is explicitly expressed at each iterative point, which contains both first and second order information, whereas in [18] the trust region radius is independent of first and second order information.

From the numerical illustrations, one may observe the numerical advantages from performance profiles and computation table. Application of this concept in constrained multi-objective optimization problems is the future scope of this article.

References

1. Ansary, M.A.T., Panda, G.: A modified quasi-newton method for vector optimization problem. Optimization 64, 2289–2306 (2015)
2. Ansary, M.A.T., Panda, G.: A sequential quadratically constrained quadratic programming technique for a multi-objective optimization problem. Engineering Optimization 51(1), 22–41 (2019)
3. Ansary, M.A.T., Panda, G.: A sequential quadratic programming method for constrained multi-objective optimization problems. J. Appl. Math. Comput. 64, 379–397 (2020)
4. Deb, K.: Multi-objective genetic algorithms: Problem difficulties and construction of test problems. Wiely India Pvt. Ltd., New Delhi, India (2003)
5. Deb, K.: Multi-objective optimization using evolutionary algorithms. Wiely India Pvt. Ltd., New Delhi, India (2003)
6. Ehrgott, M.: Multicriteria optimization. Berlin: Springer publication (2005)
7. Eichfelder, G.: An adaptive scalarization method in multiobjective optimization. SIAM J. Optim. 19(4), 1694–1718 (2009)
8. Fliege, F., Drummond, L.M.G., Svaiter, B.F.: Newton's method for multiobjective optimization. SIAM J. Optim **20**(2), 602–626 (2009)
9. Fliege, J., Svaiter, F.V.: Steepest descent methods for multicriteria optimization. Math. Methods Oper. Res. 51(3), 479–494 (2000)
10. Fonseca, C.M., Fleming, P.J.: Multiobjective optimization and multiple constraint handling with evolutionary algorithms.i.a unified formultion. IEEE Trans. Syst. Man Cybern. A Syst. Hum. **28(1)**, 26–37 (1998)
11. Huband, S., Hingston, P., Barone, L., While, L.: A review of multiobjective test problems and a scalable test problem toolkit. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION 10(5), 477–506 (2006)

12. Hwang, C.L., Yoon, K.: Multiple attribute decision making: methods and applications a state-of-the-art survey. Springer , NewYork (1981)
13. Jin, Y., Olhofer, M., Sendhoff, B.: Dynamic weighted aggregation for evolutionary multi-objective optimization: Why does it work and how? In Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation pp. 1042–1049 (2001)
14. Laumanns, M., Thiele, L., Deb, K., Zitzler, E.: Combining convergence and diversity in evolutionary multi-objective optimization. Evolutionary Computation 10, 263–282 (2002)
15. Lovison, A.: A synthetic approach to multiobjective optimization. arXiv preprint arXiv:1002.0093 (2010)
16. Miettinen, K.: Nonlinear multiobjective optimization. Springer Boston: Kluwer (1999)
17. Qu, S., Goh, M., Chan, F.T.S.: Quasi-newton methods for solving multiobjective optimization. Oper. Res. Lett. 39, 397–399 (2011)
18. Qu, S., Goh, M., Liang, B.: Trust region methods for solving multiobjective optimisation. Optim Methods Softw 28(4), 796–811 (2013)
19. Varga, R.S.: Geršgorin and his circles. Springer , NewYork (2010)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.