J. Nonlinear Var. Anal. 7 (2023), No. 5, pp. 785-809
Available online at http://jnva.biemdas.com
https://doi.org/10.23952/jnva.7.2023.5.08

NEWTON'S METHOD FOR UNCERTAIN MULTIOBJECTIVE OPTIMIZATION PROBLEMS UNDER FINITE UNCERTAINTY SETS

SHUBHAM KUMAR[1], MD ABU TALHAMAINUDDIN ANSARY[2], NIHAR KUMAR MAHATO[1], DEBDAS GHOSH[3], YEKINI SHEHU[4,*]

[1]*Pandit Dwarka Prasad Mishra, Indian Institute of Information Technology, Design & Manufacturing Jabalpur, Jabalpur, India*
[2]*Department of Mathematics, Indian Institute of Technology Jodhpur, India*
[3]*Department of Mathematical Sciences, Indian Institute of Technology (BHU) Varanasi, Varanasi, India*
[4]*School of Mathematical Sciences, Zhejiang Normal University, Jinhua 321004, China*

Abstract. In this paper, we develop Newton's method for robust counterpart of an uncertain multiobjective optimization problem under an arbitrary finite uncertainty nonempty set. Here the robust counterpart of an uncertain multiobjective optimization problem is the minimum of objective wise worst case, which is the nonsmooth deterministic multiobjective optimization problem. To solve this robust counterpart with the help of Newton's method, a suproblem is constructed and solved to find a descent direction for robust counterpart. An Armijo type inexact line search technique is developed to find a suitable step length. With the help of the descent direction and step length, we present the Newton's algorithm for the robust counterpart. The convergence of the Newton's algorithm for the robust counterpart is obtained under some usual assumptions. We also prove that the algorithm converges with super linear and quadratic rate under different assumptions. Finally, we verify the algorithm and compare with the weighted sum method via some numerical problems.
Keywords. Line search techniques; Multiobjective optimization problem; Newton's method; Robust efficiency; Uncertainty; Robust optimization.

1. Introduction

Consider the deterministic unconstrained optimization problem

$$OP: \min_{x\in\mathbb{R}^n} \Upsilon(x),$$

where $\Upsilon : \mathbb{R}^n \to \mathbb{R}$ is the objective function. In real world applications, the objective function may depend on uncertain parameters. Uncertainty in the objective function leads to parameter uncertainty. Different types of uncertainty affect the solution of optimization problems [1, 2]. In order to handle such uncertainties, instead of *OP* the following parameterized family of

*Corresponding author.
E-mail address: kumarshubham3394@gmail.com (S. Kumar), md.abutalha2009@gmail.com (M.A.T. Ansary), nihar@iiitdmj.ac.in (N.K. Mahato), debdas.mat@iitbhu.ac.in (D. Ghosh), yekini.shehu@zjnu.edu.cn (Y. Shehu).
Received December 2, 2022; Accepted March 25, 2023.

problems can be considered:

$$OP(U) = \{(OP(\xi)) : \xi \in U\},$$

where for any fixed $\xi \in U$,

$$OP(\xi) : \min_{x \in \mathbb{R}^n} \Upsilon(x, \xi), \text{ is a deterministic optimization problem.}$$

On the other hand, an optimization problem which has more than one objective functions is called multiobjective optimization problem. The multiobjective optimization problem (MOP) can be defined as

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

where $\Psi : \mathbb{R}^n \to \mathbb{R}^m$, $\Psi(x) = (\Psi_1(x), \Psi_2(x), \ldots, \Psi_m(x))$, and $\Psi_j : \mathbb{R}^n \to \mathbb{R}$ for all $j \in \Lambda = \{1, 2, \ldots, m\}$.

In multiobjective optimization problem, all objective functions cannot be optimize simultaneously. A point $x^* \in \mathbb{R}^n$ is an efficient solution tothis MOP if there is no $x \in \mathbb{R}^n$, $x \neq x^*$, such that $\Psi(x) \leq \Psi(x^*)$ and $\Psi(x) \neq \Psi(x^*)$. Also a point $x^* \in \mathbb{R}^n$ is an weakly efficient solution for MOP if there is no $x \in \mathbb{R}^n$ such that $\Psi(x) < \Psi(x^*)$. Clearly, every efficient solution is an weakly efficient solution. The image of the solution of MOP is called nondominated point in objective space. The meaning of " $<$ " and " $\leq$ " in the above discussion is the vector ordering between two vectors which can be understood in the following way: for any $v, u \in \mathbb{R}^n$,

- $v \geq u \iff v - u \in \mathbb{R}^n_{\geq} \iff v_j - u_j \geq 0$, for each j
- $v > u \iff v - u \in \mathbb{R}^n_{>} \iff v_j - u_j > 0$, for each j,

where $\mathbb{R}^n = \underbrace{\mathbb{R} \times \ldots \times \mathbb{R}}_{n \text{ times}}$, $\mathbb{R}_{>} = \{x \in \mathbb{R} : x > 0\}$, $\mathbb{R}_{\geq} = \{x \in \mathbb{R} : x \geq 0\}$, $\mathbb{R}^n_{>} = \underbrace{\mathbb{R}_{>} \times \ldots \times \mathbb{R}_{>}}_{n \text{ times}}$, and $\mathbb{R}^n_{\geq} = \underbrace{\mathbb{R}_{\geq} \times \ldots \times \mathbb{R}_{\geq}}_{n \text{ times}}$.

Similar to uncertain single objective optimization problem, an uncertain multiobjective optimization problem can be considered as

$$MOP(U) = \{MOP(\xi) : \xi \in U\},$$

where for any fixed $\xi \in U$,

$$MOP(\xi) = \min_{x \in \mathbb{R}^n} \Psi(x, \xi), \text{ is a deterministic multiobjective optimization problem.}$$

In particular, if we take $m = 1$ in $MOP(U)$, then the problem is uncertain single objective optimization problem [1].

In literature, three types of scalarization methods are there to solve MOP [3, 4, 5]. With the help of scalarization method, we can find the solution of robust counterpart of $MOP(U)$ which gives the proper robust efficient or proper robust weakly efficient or proper robust strictly efficient solution. The choice of parameter in scalarization method is not known in advance, which is the disadvantage of scalarization methods. Apart from these techniques, numerical approximation techniques are also developed by many researchers to calculate the critical point and weakly efficient solution or efficient solution or strictly efficient solution for unconstrained MOP; see, e.g., [6, 7, 8, 9, 10]. To compute the critical point, descent type method are developed for MOP that do not rely on scalarization approaches; see, e.g., [5, 11, 12, 13, 14, 15]. Two main features of these methods at every iteration are as follows:

(i) with the help of a tractable subproblem, a descent direction can be generated;

(ii) to find the feasible point which dominates the current one, a line search method is conducted along the obtained direction.

Fliege and Svaiter [11] proposed a steepest descent method to find the critical point of MOP. After that, Drummond et al. [5, 15] gave the idea to find the critical point for unconstrained *MOP* with Newton's method.

Motivated by the above literature we generalize the idea of Drummond et al. [5] of the critical point for the robust counterpart of $MOP(U)$, which gives the robust efficient or robust weakly efficient or robust strictly efficient solution to $MOP(U)$. For $MOP(U)$, a solution x^* is called

- robust weakly efficient if there is no $x \in \mathbb{R}^n - \{x^*\}$ such that $\Psi_U(x) \subset \Psi_U(x^*) - \mathbb{R}^k_>$.
- robust efficient if there is no $x \in \mathbb{R}^n - \{x^*\}$ such that $\Psi_U(x) \subset \Psi_U(x^*) - \mathbb{R}^k_\geq$.
- robust strictly efficient if there is no $x \in X - \{x^*\}$ such that $\Psi_U(x) \subset \Psi_U(x^*) - \mathbb{R}^k_{\geqq}$,

where $\Psi_U(x) = \{\Psi(x,\xi) : \xi \in U\}$. With the help of above definitions, we can easily understand that

strictly efficient $\Longrightarrow$ robust efficient $\Longrightarrow$ robust weakly efficient.

As we mentioned above, to solve $MOP(U)$, first we transform it into a deterministic multiobjective optimization problem, which is known as robust counterpart of $MOP(U)$. Here we use objective wise worst case (OWC) type robust counterpart which can be written as follows

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

where $\phi(x) = (\phi_1(x), \phi_2(x), ..., \phi_m(x))$ and $\phi_j(x) = \max_\xi \Psi_j(x,\xi)$. Therefore, it can be write as:

$$\min_{x\in\mathbb{R}^n} \phi(x) = \big(\max_{\xi\in U} \Psi_1(x,\xi), \max_{\xi\in U} \Psi_2(x,\xi), ..., \max_{\xi\in U} \Psi_m(x,\xi)\big).$$

In this paper, we consider that the uncertainty set U is a finite subset of $\mathbb{R}^k$ containing p elements i.e., $U = \{\xi_i : i \in \bar{\Lambda} = \{1,2,\dots,p\}\} \subset \mathbb{R}^k$. From now on, we consider $MOP(U)$ as $P(U)$ and its robust counterpart as RP respectively. So the problem becomes

$$P(U) = \{P(\xi_i) : \xi_i \in U, i \in \bar{\Lambda}\} \tag{1.1}$$

such that $P(\xi_i) := \min_{x\in\mathbb{R}^n} \Psi(x,\xi_i)$, where $\Psi(x,\xi_i) = (\Psi_1(x,\xi_i), \Psi_2(x,\xi_i), \dots, \Psi_m(x,\xi_i))$, $\Psi : \mathbb{R}^n \times U \to \mathbb{R}^m$, $\xi_i \in U = \{\xi_1, \xi_2, \dots, \xi_p\}$.

$$RP: \min_{x\in\mathbb{R}^n} \phi(x) = (\phi_1(x), \phi_2(x), ..., \phi_m(x)), \text{ (Robust counterpart)}$$

where $\phi_j(x) = \max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i)$, $j = 1,2,\dots,m$. Therefore, $P(U)$ is a uncertain multiobjective optimization problem with finite uncertainty sets. To solve $P(U)$, we solve RP with the help of the Newton's method. The solution of RP is the solution to $P(U)$ i.e., the weakly efficient\ efficient\ strictly efficient solution for RP will be the robust weakly efficient\ robust efficient\ robust strictly efficient solution for $P(U)$. RP is a specific type nonsmooth multiobjective optimization problem. To solve this problem, we generalized the idea of Fliege et al. [5], which is defined for smooth multiobjective optimization problems.

We organize this paper as follows. Some important results, basic definitions, and theorems which are related to our problem are presented in section 2. We given the Newton's direction for finding subproblems and its solutions in subsection 3.1. After finding the Newton's descent direction in Section 3.1, which is a solution to Newton's direction finding subproblem, we given the Armijo type line search rule to find the suitable step length size in Subsection 3.2. With the step size, which satisfies the Armijo type line search rule, it is ensures that the function value will decrease in the Newton's descent direction. To find the critical point, we write the Newton's algorithm for *RP* in Subsection 3.3. With the help of this algorithm, we generate a sequence, and in Subsection 3.4, we prove that the sequence converges to a critical point. In Subsection 3.5, we prove that, under some assumptions, the sequence generated by the Newton's algorithm converges with superlinear and quadratic rate. In Subsection 3.6, numerically, we verify the Newton's algorithm for *RP* with the help of suitable examples. In Section 4, we conclude with some comments related to the presented algorithm.

2. PRELIMINARIES

In smooth *MOP*, a point x^* is critical point for Ψ if $R(D\Psi(x^*)) \cap (-\mathbb{R}^m_>) = \emptyset$ (i.e., for all $v \in \mathbb{R}^n$, there exists a $j^0 \in \Lambda$ such that $\nabla\Psi_{j^0}(x^*)^T v \geq 0$). This is the necessary condition for Pareto optimality for *MOP* (Fliege and Svaiter [11]). In other words, if x^* is critical point for G, then, for all $v \in \mathbb{R}^n$, there exists $j^0 = j^0(v) \in \Lambda$ such that $\nabla\Psi_{j^0}(x^*)^T v \geq 0$. In general, for unconstrained smooth *MOP*, efficiency (Pareto optimality) is not equivalent to criticality (i.e., critical points need not be efficient solutions). Both (criticality and efficiency) are related as follows;

(a) If $x^* \in \mathbb{R}^n$ is weak efficient solution or efficient solution, then x^* is a critical point for Ψ.
(b) If Ψ is $\mathbb{R}^m-$convex and x^* is critical point for G, then x^* is a weak efficient solution.
(c) If Ψ is $\mathbb{R}^m$ twice continuously differentiable and for each $j \in \Lambda$ and $x \in \mathbb{R}^n$, $\nabla^2\Psi_j(x)$ is positive definite, and if x^* is critical point, then x^* is an efficient solution.

In *RP*, generally, $\phi_j(x) = \max_{i\in\bar{\Lambda}} \Psi_j(x,\xi)$ is not differentiable. Then, the subdifferential of $\phi_j(x)$ is given by $\partial\phi_j(x) = conv\{\nabla\Psi_j(x,\xi_i) : i \in I_j(x)\}$, where $\Psi_j(x,\xi_i)$ is convex and continuously differentiable function for each x and ξ_i [16]. Hessian of $\Psi_j(x,\xi_i)$ is given by $\nabla^2\Psi_j(x,\xi_i)$ and $\nabla^2\Psi_j(x,\xi_i)$ is said to be positive definite if $d^T\nabla^2\Psi_j(x,\xi_i)d > 0$ for all $0 \neq d \in \mathbb{R}^n$.

Definition 2.1. (Directional derivative) The directional derivative of $f : \mathbb{R}^n \to \mathbb{R}$ at x in direction d is given as

$$f'(x;d) = \lim_{q\to 0} \frac{f(x+qd) - f(x)}{q}.$$

If f is differentiable, then $f'(x;d) = \nabla f(x)^T d$. In particular, the directional derivative for $\phi_j(x) = \max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i)$ at x in direction v is given as $\phi_j'(x,v) = \max_{i\in I_j(x)} \nabla\Psi_j(x,\xi_i)^T v$, where $\Psi_j(x,\xi_i)$ is continuously differentiable function for each $x \in \mathbb{R}^n$, $\xi_i \in U$, and $I_j(x) = \{i \in \bar{\Lambda} : \Psi_j(x,\xi_i) = \phi_j(x)\}$ denotes the set of active index for ϕ_j.

Definition 2.2. [17] (Descent direction) Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function. Then a vector v is said to be descent direction for f at x if $f'(x;v) = \nabla f(x)^T v < 0$. In particular, for *RP*, a vector v is said to be descent direction for ϕ at x if

$$\phi_j'(x,v) = \max_{i\in I_j(x)} \nabla\Psi_j(x,\xi_i)^T v < 0, \ \forall j \in \Lambda,$$

$$\text{i.e., } \nabla\Psi_j(x,\xi_i)^T v < 0, \ \forall j \in \Lambda \text{ and } i \in I_j(x),$$

equivalently, if v is a descent direction for $\phi(x)$ at x, then there exists $\varepsilon > 0$ such that

$$\phi_j(x+\alpha v) < \phi_j(x) \ \forall \ j \in \Lambda \text{ and } \alpha \in (0,\varepsilon].$$

Definition 2.3. [18] A function $\phi : \mathbb{R}^n \to \mathbb{R}^m$ is said to be $\mathbb{R}^m-$convex$\backslash\mathbb{R}^m-$strctly convex$\backslash\mathbb{R}^m-$ strongly convex if each component of ϕ is convex $\backslash$ strictly convex $\backslash$ strongly convex.

Definition 2.4. [18] $\phi : \mathbb{R}^n \to \mathbb{R}^m$ is $\mathbb{R}^m-$strongly convex if and only if there exists $\omega > 0$ such that $\lambda_{\min}(\nabla^2\phi_j(x)) \geq \omega$ for all $x \in \mathbb{R}^n$ and $j = 1,2,...,m$, where $\lambda_{\min} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ denotes the minimum eigenvalue function.

Now we give the theorem which relates the solution of $P(U)$ to the solution of *RP*. With help of this theorem, instead of solving $P(U)$ we solve *RP*.

Theorem 2.1. [19] *Let $P(U)$ be an uncertain multiobjective optimization problem with finite uncertainty non empty set and RP be the robust counterpart of $P(U)$. Then,*

(a) if $x^* \in \mathbb{R}^n$ is a strictly efficient solution to *RP*, then x^* is robust strictly efficient solution for $P(U)$;
(b) let x^* be a weakly efficient solution to *RP*. If $\max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i)$ exist for all $j \in \Lambda_m$ and all $x \in \mathbb{R}^n$, then x^* is robust weakly efficient solution for $P(U)$.

3. The Newton's Method for *RP*

We solve *RP* with the help of the Newton's method and assume that in *RP* the function $\phi : \mathbb{R}^n \to \mathbb{R}^m$ is such that $\phi(x) = (\phi_1(x), \phi_2(x), \ldots, \phi_m(x))$, where $\phi_j(x) = \max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i),\ j \in \Lambda$ and $\Psi_j : \mathbb{R}^n \times U \to \mathbb{R}$ is a twice continuously differentiable and strictly convex function for each x and $\xi_i \in U$. First, we start by introducing the necessary condition for Pareto optimality for ϕ. A point x^* is critical point for ϕ if $R(Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}) \cap (-\mathbb{R}^m_>) = \emptyset$ i.e., for at least one $j^0 \in \Lambda$, we have $\nabla\Psi_{j_0}(x^*,\xi_i)^T v \geq 0$ for all $v \in \mathbb{R}^n$ and for all $i \in I_{j_0}(x)$. Here by $R(Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\})$ we mean the range of $Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$. The above condition which is satisfied by the critical point is called a necessary condition for Pareto optimality for ϕ. If x^* is a critical point for *RP*, then there does not exists any $v \in R^n$ such that $\nabla\Psi_j(x^*,\xi_i)^T v < 0\ \forall i \in I_j(x)$ and $j \in \Lambda$.

Lemma 3.1. *If x^* is critical point for ϕ, then $0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$.*

Proof. Since x^* is critical point for ϕ, then there must exists $d \in \cup_{j\in\Lambda}\partial\phi_j(x^*)$ such that

$$v^T d \geq 0,\ \forall v \in \mathbb{R}^n. \tag{3.1}$$

On the contrary, if we assume $0 \notin Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$, since $Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$ and $\{0\}$ are closed and convex sets, then, with the help of the separation theorem, there exists $v \in \mathbb{R}^n$ and $b \in \mathbb{R}$ such that $v^T 0 \geq b$ and $v^T d < b\ \forall d \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$. Combining the two inequalities contradicts (3.1). Hence, $0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$. □

From Lemma 3.1 above and the definition of critical points, it is clear that if x^* is critical point, then both condition (a) $R(Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}) \cap (-\mathbb{R}^m_>) = \emptyset$ i.e., for at least one $j^0 \in \Lambda$, we have $\nabla\Psi_{j_0}(x,\xi_i)^T v \geq 0$ for all $v \in \mathbb{R}^n$ and, for all $i \in I_{j_0}(x)$ and (b) $0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$ is satisfied. Thus, we can use both condition as necessary condition for Pareto optimality for ϕ in *RP*.

The following theorem gives us necessary condition for Pareto optimality or efficiency for *RP*.

Theorem 3.1. *Let $\Psi_j(x,\xi_i)$ be continuously differentiable and convex function for each j and $\xi_i \in U$. If x^* is weakly efficient solution for RP, then $0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$.*

Proof. Let x^* be a weakly efficient solution to ϕ. We have to show $0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}$. Since $\Psi_j(x,\xi_i)$ is continuously differentiable and convex for each j, and $\xi_i \in U$, then $\Psi_j(x,\xi_i)$ is locally Lipschitz continuous for all $i \in \bar{\Lambda}$. Then, by [20, Theorem 4.3], one has

$$0 \in Conv\{\cup_{j\in\Lambda}\partial\phi_j(x^*)\}.$$

Thus, this theorem gives us necessary condition for Pareto optimal for ϕ. □

In the following theorem, we see that the necessary condition is sufficient for ϕ if ϕ is $\mathbb{R}^m$-convex.

Theorem 3.2. *Suppose ϕ is $\mathbb{R}^m$-convex function (i.e., each ϕ_j is convex) and if x^* is critical point for RP then x^* is weak efficient solution.*

Proof. Since $\Psi_j(x,\xi_i)$ is convex, then $\phi_j = \max_{i\in\bar{\Lambda}}\Psi_j(x,\xi_i)$ is convex. Assume that x^* is critical point to ϕ, then, for at least one j^0, we have $\phi'_{j^0}(x^*,d) \geq 0\ \forall d \in R^n$. Thus

$$\nabla\Psi_{j_0}(x,\xi_i)^T d \geq 0\ \forall d \in R^n,\ i \in I_{j_0}(x). \tag{3.2}$$

By convexity of ϕ_j and $\Psi_j(x,\xi_i)$, we obtain

$$\Psi_{j^0}(x,\xi_i) \geq \Psi_{j^0}(x^*,\xi_i) + \nabla\Psi_{j^0}(x,\xi_i)^T(x-x^*),\ \forall i \in I_{j_0}(x).$$

It follows from(3.2) that $\Psi_{j^0}(x,\xi_i) \geq \Psi_{j^0}(x^*,\xi_i)$ for all $i \in I_{j_0}(x)$, which implies $\phi_{j^0}(x) \geq \phi_{j^0}(x^*)$, that is, x^* is weakly efficient solution. □

Next, we solve a subproblem to find the Newton's descent direction.

3.1. **Subproblem to find a descent direction for *RP*.** To find the Newton's descent direction for *RP*, we e consider the following real-valued minimization subproblem:

$$\min_{v\in\mathbb{R}^n}\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x)\}. \tag{3.3}$$

We assume that

$$\rho(t,v) = t = \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x)\}, \tag{3.4}$$

and, for every j, $\nabla^2\Psi_j(x,\xi_i)$ is positive definite for all $x \in \mathbb{R}^n$ and $i \in \bar{\Lambda}$.

Solution of subproblem (3.3): Since the given objective function in subproblem (3.3) is strictly convex as being a maximum of strictly convex functions. Therefore, subproblem (3.3) has a unique solution which is the Newton's descent direction for *RP*. Let $v(x)$ and $\theta(x)$ be the optimal solution and optimal value of subproblem (3.3), respectively. Then

$$v(x) = \arg\min_{v\in\mathbb{R}^n}\rho(t,v)$$
$$\theta(x) = \rho(t,v(x)).$$

Equivalently, (3.3) can be written as

$$P(x):\ \min_{v\in\mathbb{R}^n,t\in\mathbb{R}}\rho(t,v)$$

$$s.t.\ \Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x) \leq t,\ \forall i \in \bar{\Lambda} \text{ and } j \in \Lambda.$$

Give the constraints in the above problem satisfying the Slater's constraints qualification condition at $t = 1$ and $v = (0,0,...,0)$, so the solution of the given convex programming problem is given by the KKT optimality condition. The Lagrangian function is given by

$$L(t,v,\lambda) = t + \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}(\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x) - t).$$

Then the KKT optimality conditions are

$$\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij} = 1, \tag{3.5}$$

$$\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}\nabla\Psi_j(x,\xi_i) + \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}\nabla^2\Psi_j(x,\xi_i)^T v = 0, \tag{3.6}$$

$$\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x) - t \leq 0,\ \forall i \in \bar{\Lambda} \text{ and } j \in \Lambda,$$

$$\lambda_{ij} \geq 0,\ \lambda_{ij}(\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v - \phi_j(x) - t) = 0,\ \forall i \in \bar{\Lambda} \text{ and } j \in \Lambda.$$

From (3.6), one has

$$v(x) = -\left(\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}\nabla^2\Psi_j(x,\xi_i)\right)^{-1} \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}\nabla\Psi_j(x,\xi_i). \tag{3.7}$$

Since, for every j, $\nabla^2\Psi_j(x,\xi_i)$ is positive definite for all $x \in \mathbb{R}^n$ and $i \in \bar{\Lambda}$, which ensures that the existence of the inverse of $\nabla^2\Psi_j(x,\xi_i)$. Since x is not a critical point, there must exists at least one $\lambda_{ij} > 0$ and hence the inverse of $\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}\nabla^2\Psi_j(x,\xi_i)$ exists. Therefore, $v(x)$ is well defined. Then the optimal value of subproblem (3.3) is

$$\theta(x) = \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2}v(x)^T\nabla^2\Psi_j(x,\xi_i)v(x) - \phi_j(x)\}. \tag{3.8}$$

In Theorem 3.3 below, we give the relation between critical point and $v(x)$ and we also prove that $v(x)$ given in (3.7), is a descent direction.

Theorem 3.3. *Let $v(x)$ and $\theta(x)$ be the solution and optimal value of subproblem (3.3), respectively. Then the following results will hold:*

- *(1) $v(x)$ is bounded on compact subset C of $\mathbb{R}^n$ and $\theta(x) \leq 0$.*
- *(2) The following conditions are equivalent:*
  - *(a) The point x is non-critical point.*
  - *(b) $\theta(x) < 0$.*
  - *(c) $v(x) \neq 0$.*
  - *(d) $v(x)$ is a descent direction for ϕ at x for the problem RP .*

In particular, x is critical point iff $\theta(x) = 0$.

Proof. (1) Let C be a compact subset of $\mathbb{R}^n$. Since, for each $j \in \Lambda$, $\Psi_j(x,\xi_i)$ is twice continuously differentiable for all $x \in \mathbb{R}^n$ and $\xi_i \in U$, then its first and second order derivative is bounded on compact set C. Thus, for all $x \in \mathbb{R}^n$, $j \in \Lambda$, $\xi_i \in U$ and by (3.7), $v(x)$ is bounded on compact set C. Since $t = 1$ and $v = \bar{0} = (0,0,\ldots,0) \in \mathbb{R}^n$ lies in the feasible region, then we have

$$\theta(x) \leq \max_{j\in\Lambda}\{\max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i) + \max_{i\in\bar{\Lambda}} \nabla\Psi_j(x,\xi_i)^T\bar{0} + \max_{i\in\bar{\Lambda}} \frac{1}{2}\bar{0}^T\nabla^2\Psi_j(x,\xi_i)\bar{0} - \max_{i\in\bar{\Lambda}} \phi_j(x)\} = 0.$$

Hence $\theta(x) \leq 0$.

(2): $(a) \implies (b)$

If x is not a critical point, then there exists $\bar{v}$ such that $\nabla\Psi_j(x,\xi_i)^T\bar{v} < 0$ for all $j \in \Lambda$ and $i \in I_j(x)$. Since $\theta(x)$ is the optimal value for subproblem (3.3), then, for all $\delta > 0$,

$$\theta(x) \leq \max_{j\in\Lambda}\{\max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i) + \max_{i\in\bar{\Lambda}}\{\nabla\Psi_j(x,\xi_i)^T(\delta\bar{v}) + \frac{1}{2}\bar{(}\delta\bar{v})^T\nabla^2\Psi_j(x,\xi_i)(\delta\bar{v})\} - \max_{i\in\bar{\Lambda}} \phi_j(x)\}\}.$$

It follows that

$$\theta(x) \leq \delta\left(\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\nabla\Psi_j(x,\xi_i)^T\bar{v} + \delta\frac{1}{2}\bar{v}^T\nabla^2\Psi_j(x,\xi_i)\bar{v}\}\right).$$

For small enough $\delta > 0$, the right-hand side of the inequality above is negative because of $\nabla\Psi_j(x,\xi_i)^T\bar{v} < 0$, $\frac{1}{2}\bar{v}^T\nabla^2\Psi_j(x,\xi_i)\bar{v} > 0$ and also $\theta(x) \leq 0$. Thus $\theta(x) < 0$.

$(b) \implies (c)$

Since $\theta(x)$ is the optimal value of subproblem (3.3), it is from (b) strictly negative. We obtain $v(x) \neq 0$. If $v(x) = 0$, then $\theta(x)$ is zero which is not possible from (b). Hence if $\theta(x) < 0$, then $v(x) \neq 0$.

$(c) \implies (d)$

Let $v(x) \neq 0$, so $\theta(x) \neq 0$. Since $\theta(x) \leq 0$ and $v(x) \neq 0$ then $\theta(x) < 0$. Thus,

$$\theta(x) \leq \max_{j\in\Lambda}\{\max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i) + \max_{i\in\bar{\Lambda}}\{\nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2}v^T(x)\nabla^2\Psi_j(x,\xi_i)v(x)\} - \max_{i\in\bar{\Lambda}} \phi(x)\} < 0$$

$$\implies \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2}v^T(x)\nabla^2\Psi_j(x,\xi_i)v(x)\} < 0$$

$$\implies \nabla\Psi_j(x,\xi_i)^T v(x) < 0,\ \forall\ j \in \Lambda \text{ and } i \in \bar{\Lambda}$$

$$\implies \nabla\Psi_j(x,\xi_i)^T v(x) < 0,\ \forall\ j \in \Lambda \text{ and } i \in I_j(x)$$

$$\implies v(x) \text{ is a descent direction for } \phi \text{ at } x \text{ for problem } RP.$$

$(d) \implies (a)$

Since $v(x)$ is a descent direction for ϕ at x, then

$$\nabla\Psi_j(x,\xi_i)^T v(x) < 0,\ \forall\ j \in \Lambda \text{ and } i \in I_j(x). \text{ Hence } x \text{ is not a critical point.}$$

Also if $\theta(x) < 0$, then $v(x) \neq 0$; and if $v(x) \neq 0$, then $\theta(x) < 0$. Thus, we have that x is critical point if and only if $\theta(x) = 0$. □

In Theorem 3.3, the characterization between the critical point and the descent direction can be used as a stopping criteria in Newton's algorithm. Next, we prove that $\theta(x)$ is continuous for every $x \in \mathbb{R}^n$.

Theorem 3.4. *Let $\theta : \mathbb{R}^n \to \mathbb{R}$ be a function which is defined in (3.8). Then, $\theta(x)$ is continuous.*

Proof. To prove the continuity of $\theta(x)$, it is suffice to prove that $\theta(x)$ is continuous in any arbitrary compact subset C of $\mathbb{R}^n$. Since, by Theorem 3.3, $\theta(x) \leq 0$, then, for every $j \in \Lambda$ and $i \in \bar{\Lambda}$,

$$\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2} v(x)^T \nabla^2 \Psi_j(x,\xi_i) v(x) - \phi_j(x) \leq 0.$$

Since, at the active indices, $\Psi_j(x,\xi_i)$ attains its maximum value, i.e., $\Psi_j(x,\xi_i) = \phi_j(x)$. It follows that $\nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2} v(x)^T \nabla^2 \Psi_j(x,\xi_i) v(x) \leq 0$, which implies

$$\frac{1}{2} v(x)^T \nabla^2 \Psi_j(x,\xi_i) v(x) \leq -\nabla\Psi_j(x,\xi_i)^T v(x), \text{ for every } j \in \Lambda,\ i \in \bar{\Lambda}. \tag{3.9}$$

Since, for every $j \in \Lambda$ and $i \in \bar{\Lambda}$, $\Psi_j(x,\xi_i)$ is twice continuously differentiable, and the Hessian of that is positive definite for every $x \in \mathbb{R}^n$. Also, C is a compact subset of $\mathbb{R}^n$, and then eigenvalues of Hessian of $\Psi_j(x,\xi_i)$ are bounded on C. So, there exists $\mu > 0$ and $\bar{\mu}$ such that

$$\bar{\mu} = \max_{x \in C,\ j \in \Lambda} \max_{i \in \bar{\Lambda}} \|\Psi_j(x,\xi_i)\| \tag{3.10}$$

and

$$\mu \|w\|^2 \leq w^T \nabla^2 \Psi_j(x,\xi_i) w, \text{ for every } j \in \Lambda,\ i \in \bar{\Lambda} \text{ and } \forall x \in C. \tag{3.11}$$

Now from (3.9), (3.10), (3.11), and Cauchy-Schwartz inequality, we have

$$\mu \|v(x)\|^2 \leq \|\nabla\Psi_j(x,\xi_i)\| \|v(x)\| \leq \bar{\mu} \|v(x)\| \ \forall y \in C,\ j \in \Lambda \textit{ and } i \in \bar{\Lambda},$$

which implies that $\|v(x)\| \leq \frac{\bar{\mu}}{\mu}$ for all $x \in C$ i.e., $v(x)$, the Newton's directions are bounded on C.

Now, for $x \in C$, $j \in \Lambda$ and $i \in \bar{\Lambda}$, we define

$$E_{x,j,i} : C \to \mathbb{R}$$

$$s.t.\ z \to \Psi_j(z,\xi_i) + \nabla\Psi_j(z,\xi_i) v(x) + \frac{1}{2} v(x)^T \nabla^2 \Psi_j(z,\xi_i) v(x) - \phi_j(z).$$

Thus the family $\{E_{x,j,i}\}_{x \in C,\ j \in \Lambda, i \in \bar{\Lambda}}$ is equicontinuous. Therefore, $\{\Delta_x = \max_{j \in \Lambda} \max_{i \in \bar{\Lambda}} E_{x,j,i}\}_{x \in C}$ is also equicontinuous. Now if we take $\varepsilon > 0$, then there exists $\delta > 0$ such that, for all $u, w \in C$,

$$\|u - w\| < \delta \implies |\Delta_x(u) - \Delta_x(w)| < \varepsilon, \forall x \in C.$$

Hence, for $\|u - w\| < \delta$,

$$\begin{aligned}
\theta(w) &\leq \Psi_j(w,\xi_i) + \nabla\Psi_j(w,\xi_i)^T v(u) + v(u)^T \nabla^2 \Psi_j(w,\xi_i)^T v(u) - \phi_j(w) \\
&= \Delta_u(w) \\
&\leq \Delta_u(u) + |\Delta_u(w) - \Delta_u(u)| \\
&\leq \theta(u) + \varepsilon.
\end{aligned}$$

Thus we obtain $\theta(u) - \theta(w) < \varepsilon$; if we interchange u and w, then we obtain $\theta(u) - \theta(w) > -\varepsilon$. Therefore, $|\theta(u) - \theta(w)| < \varepsilon$ whenever $\|u - w\| < \delta$, and then $\theta(x)$ is continuous in C. Since C is any arbitrary compact subset of $\mathbb{R}^n$, then $\theta(x)$ is continuous. □

Next, we establish the Armijo type inexact line search technique to find the step length size for Newton's method for *RP*.

3.2. **Calculation of step length size.** By Theorem 3.3, we have $\theta(x) \leq 0$, and we know

$$\theta(x) = \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v(x) + \frac{1}{2}v(x)^T\nabla^2\Psi_j(x,\xi_i)v(x) - \phi_j(x)\}.$$

To find the step length size, we assume an auxiliary function $\phi_j''^*(x,\alpha v)$, which is given by

$$\phi_j''^*(x,\alpha v) = \max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v\} - \phi_j(x) - \phi_j'(x,\alpha v), \tag{3.12}$$

where $\alpha \in [0,\varepsilon]$ and $\varepsilon < 1$. For every $j \in \Lambda$, (3.12) can be written as

$$\phi_j''^*(x,\alpha v) + \phi_j'(x,\alpha v) \leq \max_{i\in\bar{\Lambda}}\Psi_j(x,\xi_i) + \max_{i\in\bar{\Lambda}}\{\alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v\} - \phi_j(x)$$

and also

$$\phi_j''^*(x,\alpha v) + \phi_j'(x,\alpha v) \leq \max_{i\in\bar{\Lambda}}\{\alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v\}. \tag{3.13}$$

Since $\Psi_j(x,\xi_i)$ is upper uniformly twice continuously differentiable and convex, then there exists $k_i^j > 0$ such that

$$\Psi_j(x+v,\xi_i) \leq \Psi_j(x,\xi_i) + \nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)^T v + \frac{1}{3!}k_i^j\|v\|^3.$$

Since the inequality above holds for every $v \in \mathbb{R}^n$, one has

$$\begin{aligned}\Psi_j(x+\alpha v,\xi_i) &\leq \Psi_j(x,\xi_i) + \alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v + \frac{1}{3!}k_i^j\|\alpha v\|^3\\ &\leq \max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v + \frac{1}{3!}k_i^j\|\alpha v\|^3\}\\ &\leq \max_{i\in\bar{\Lambda}}\{\Psi_j(x,\xi_i) + \alpha\nabla\Psi_j(x,\xi_i)^T v + \frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v\} + \frac{1}{3!}\alpha^3 K^j\|v\|^3,\end{aligned}$$

where $K^j = \max_{i\in\bar{\Lambda}} k_i^j$. Now, from equation (3.12), we have

$$\Psi_j(x+\alpha v,\xi_i) \leq \phi_j'(x,\alpha v) + \phi_j(x) + \phi_j''^*(x,\alpha v) + \frac{1}{3!}\alpha^3 K^j\|v\|^3.$$

It holds for each $i \in \bar{\Lambda}$. Therefore,

$$\max_{i\in\bar{\Lambda}}\Psi_j(x+\alpha v,\xi_i) \leq \phi_j'(x,\alpha v) + \phi_j(x) + \phi_j''^*(x,\alpha v) + \frac{1}{3!}\alpha^3 K^j\|v\|^3$$

and

$$\phi_j(x+\alpha v) \leq \phi_j'(x,\alpha v) + \phi_j(x) + \phi_j''^*(x,\alpha v) + \frac{1}{3!}\alpha^3 K^j\|v\|^3. \tag{3.14}$$

It follows from (3.13) that

$$\begin{aligned}
&\phi_j''^*(x,\alpha v)+\phi_j'(x,\alpha v)\\
&\leq \max_{i\in\bar{\Lambda}}\{\alpha\nabla\Psi_j(x,\xi_i)^T v+\frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v\}\\
&\leq \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\alpha\Psi_j(x,\xi_i)+\alpha\nabla\Psi_j(x,\xi_i)^T v+\frac{1}{2}\alpha^2 v^T\nabla^2\Psi_j(x,\xi_i)^T v-\alpha\phi_j(x)\}\\
&\leq \alpha\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x)+\nabla\Psi_j(x,\xi_i)^T v+\frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v-\phi_j(x)\}\\
&+\frac{\alpha(\alpha-1)}{2}\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}} v^T\nabla^2\Psi_j(x,\xi_i)^T v\\
&\leq \alpha\theta(x)+\frac{1}{2}\alpha(\alpha-1)\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{v^T\nabla^2\Psi_j(x,\xi_i)^T v\}. \qquad (3.15)
\end{aligned}$$

Since x is not a critical point, we obtain by Theorem 3.3, (3.13), (3.14), and (3.15) that

$$\phi_j(x+\alpha v)\leq\phi_j(x)+\alpha\theta(x)+\frac{1}{2}\alpha(\alpha-1)\max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{v^T\nabla^2\Psi_j(x,\xi_i)^T v\}+\frac{1}{3!}\alpha^3K^j\|v\|^3. \quad (3.16)$$

Since $\nabla^2\Psi_j(x,\xi_i)$ is positive definite for all $x\in\mathbb{R}^n$ and $i\in\bar{\Lambda}$, then $v^T\nabla^2\Psi_j(x,\xi_i)^T v>0$, $v(\neq 0)\in\mathbb{R}^n$. Now, the third term of R.H.S of (3.16) is negative as $\alpha<1$. Then (3.16) can be written as

$$\phi_j(x+\alpha v)\leq\phi_j(x)+\alpha\theta(x)+\alpha^3\frac{1}{3!}K^j\|v\|^3.$$

If $\alpha>0$ is sufficiently small, then $\alpha^3\approxeq 0$. It follows that $\phi_j(x+\alpha v)\leq\phi_j(x)+\alpha\theta(x)$. Since $\theta(x)<0$, then, for some $\beta\in(0,1)$, $\theta(x)<\beta\theta(x)$. Since α is sufficiently small and positive, we have

$$\phi_j(x+\alpha v)\leq\phi_j(x)+\alpha\beta\theta(x). \quad (3.17)$$

Equation (3.17) represents the step length size rule for Newton's descent algorithm for the *RP*. So, we obtain Newton's descent direction $v(x)$ at x and step length size given in (3.17). Therefore, we are in position to write Newton's algorithm for *RP*.

Algorithm 1 Newton's algorithm for *RP*

Step 1. Choose $\varepsilon>0$, $\beta\in(0,1)$, and $x^0\in\mathbb{R}^n$. Set $k:=0$

Step 2. Solve $P(x^k)$ for v^k at x^k.

Step 3. Compute $\theta(x^k)=\min_{t\in\mathbb{R},v\in R^n}\rho(t,v^k)$, where $\rho(t,v^k)$, is defined in (3.4) . If $|\theta(x^k)|<\varepsilon$, then stop. Otherwise proceed to *Step* 4.

Step 4. Choose α_k as largest $\alpha\in\{\frac{1}{2^r}:r=1,2,3,...\}$ such that

$$\phi_j(x^k+\alpha v)\leq\phi_j(x^k)+\alpha\beta\theta(x^k), \quad (3.18)$$

where $\theta(x^k)<0$.

Step 5. Set $x^{k+1}:=x^k+\alpha_k v^k$, and go to step 2.

3.3. **Newton's algorithm for RP.** The well-definedness of Algorithm 1 depends on Step 2, Step 3, Step 4, and Step 5. In Step 2, we have to compute a minimizer of a function $v\mapsto t=$

$\rho(t,v)$. As this function is strictly convex, evidently there exists one minimizer of it. Therefore, at x^k, v^k always exists, and hence Step 2 is well defined. With the help of v^k, we can find $\theta(x^k)$, and hence Step 3 is well defined. Note that, by Theorem 3.3, the stopping criteria $\|\theta(x^k)\| < \varepsilon$ can be replaced by $\|v^k\| < \varepsilon$. It should be noted that if, at iteration k, Algorithm 1 does not reach Step 4, i.e., if it stops at Step 3, then, by Theorem 3.3, x^k is an approximately critical point for ϕ. If Step 4 is reached at iteration k, we choose α_k as the largest $\alpha \in \{\frac{1}{2^r} : r = 1,2,3,\dots\}$ which satisfies (3.18). Also, note that, by equation (3.17), the objective function values always decrease in the component-wise partial order, since x^k is a noncritical point of ϕ, by Theorem 3.3., $\theta(x^k) < 0$. Therefore, from Step 4, we see that $\phi(x_{k+1}) < \phi(x_k)$. Thus, Step 4 is well defined. With the help of v^k, α_k, and current iteration point x^k, we calculate x^{k+1} in Step 5. Then, we go to Step 2. Repeating this process, until we reach the stoping criteria at Step 3.

In the following subsection, we discuss the convergence analysis of Newton's algorithm for RP.

3.4. **Convergence analysis of Newton's algorithm (Algorithm 1) for problem RP.**

Lemma 3.2. *For all $k = 0,1,2,\dots$ and $j \in \Lambda$, we have $\sum_{r=0}^{k} \alpha_r |\theta(x^r)| \leq \beta^{-1}(\phi_j(x^0) - \phi_j(x^{k+1}))$, where $\phi_j(x) = \max_{i\in\bar{\Lambda}} \Psi_j(x,\xi_i)$.*

Proof. By the Step 4 of Algorithm 1, one has $\phi_j(x^{k+1}) \leq \phi_j(x^k) + \alpha_k \beta \theta(x^k)$ and

$$-\alpha_k \theta(x^k) \leq \beta^{-1}(\phi_j(x^k) - \phi_j(x^{k+1})).$$

Thus $-\sum_{r=0}^{k} \alpha_r \theta(x^r) \leq \beta^{-1}(\phi_j(x^0) - \phi_j(x^{k+1}))$, and then

$$\sum_{r=0}^{k} \alpha_r \theta(x^r) \geq -\beta^{-1}(\phi_j(x^0) - \phi_j(x^{k+1})). \tag{3.19}$$

Since $\theta(x^r) < 0$, we find from equation (3.19) that

$$\beta^{-1}(\phi_j(x^0) - \phi_j(x^{k+1})) \geq \sum_{r=0}^{k} \alpha_r \theta(x^r). \tag{3.20}$$

From (3.19) and (3.20), we conclude that $\sum_{r=0}^{k} \alpha_r |\theta(x^r)| \leq \beta^{-1}(\phi_j(x^0) - \phi_j(x^{k+1}))$. □

Theorem 3.5. *Let $\{x_k\}$ be a sequence which is produced by Algorithm 1. Then the accumulation point of $\{x_k\}$ is a Pareto optimum for ϕ.*

Proof. Let x^* be an accumulation point of the sequence $\{x_k\}$, which is generatd by Newton's descent Algorithm 1. Then there exists a subsequence $\{x^{k_l}\}$ such that $\lim_{l\to\infty} x^{k_l} = x^*$. Using Lemma 3.2 with $l = k_l$ and taking the limit $l \to \infty$ from the continuity of ϕ_j, we have $\sum_{l=0}^{\infty} \alpha_l \|\theta(x^l)\| < \infty$. Therefore, $\lim_{k\to\infty} \alpha_k \theta(x^k) = 0$. In particular, we have

$$\lim_{l\to\infty} \alpha_{k_l} \theta(x^{k_l}) = 0. \tag{3.21}$$

If x^* is a non-critical point, then

$$\theta(x^*) < 0 \textit{ and } v(x)^* \neq 0. \tag{3.22}$$

Define

$$g : \mathbb{R}^m \to \mathbb{R} \text{ by } g(u) = \max_{j\in\Lambda} u_j.$$

Using the definition of $\theta(.)$ and descent direction $v(x)^*$, we conclude that there exists a non negative integer q such that $g(\phi(x^* + 2^{-q}v(x)^*) - \phi(x^*)) < \beta 2^{-q}\theta(x^*)$. Since $g(.)$ and $\theta(x)$ are continuous, then $\lim_{l\to\infty} v^{k_l} = v(x)^*$ and $\lim_{l\to\infty}\theta(x^{k_l}) = \theta(x^*)$. Also, for l large enough,

$$g(\phi(x^{k_l} + 2^{-q}v^{k_l}) - \phi(x^{k_l})) < \sigma 2^{-q}\theta(x^{k_l}), \tag{3.23}$$

which implies

$$\phi_j(x^{k_l} + 2^{-q}v^{k_l}) - \phi_j(x^{k_l}) < \sigma 2^{-q}\theta(x^{k_l}). \tag{3.24}$$

By Step 4 in Algorithm 1, we have

$$\phi_j(x^{k_l} + \alpha_{k_l}v^{k_l}) \le \phi_j(x^{k_l}) + \sigma\alpha_{k_l}\theta(x^{k_l}). \tag{3.25}$$

Since α_{k_l} is the largest of $\{\frac{1}{2^n} : n = 1,2,3,....\}$, and $\sigma \in (0,1)$, we find from (3.24) and (3.25) that $\sigma\alpha_{k_l}\theta(x^{k_l}) \le \sigma 2^{-q}\theta(x^{k_l})$, which implies $\alpha_{k_l} \ge 2^{-q} > 0$ for l large enough. Hence $\lim_{l\to\infty}\alpha_{k_l}\theta(x^{k_l}) > 0$, which contradicts (3.21). Therefore, the condition of (3.22) is not true. Thus x^* is a critical point. Since ϕ is strictly convex, one sees that x^* is a Pareto optimum. □

Theorem 3.6. *Let x^0 be in a compact level-set of ϕ, and $\{x^k\}$ be the sequence generated by Algorithm 1. Then $\{x^k\}$ converges to a Pareto optimum $x^* \in \mathbb{R}^n$ for ϕ for the problem RP.*

Proof. Let γ_0 be the $\phi(x^0)$-level set of ϕ, that is,

$$\gamma_0 = \{x \in \mathbb{R}^n : \phi(x) \le \phi(x^0)\}.$$

Since $\{\phi(x^k)\}$ is $\mathbb{R}^m$-decreasing , then $x^k \in \gamma_0$ for all k. Therefore $\{x^k\}$ is bounded, and all of its accumulation points lie in γ_0. By using the theorem above, we conclude that they are all Pareto optima for ϕ. Since, for every $x^k \in \gamma_0$, we have $\phi(x^{k+1}) \le \phi(x^k)$ for all k, we conclude by [21, lemma 3.8] that, for any accumulation point x^* of $\{x^k\}$, $\phi(x^*) \le \phi(x^k)$ and $\lim_{k\to\infty}\phi(x^k) = \phi(x^*)$ and also $\phi(x)$ is constant in the set of accumulation point of $\{x^k\}$. As ϕ is strongly convex, i.e., strictly $\mathbb{R}^m$-convex, we see that there exists one accumulation point of $\{x^k\}$ say x^*. Hence the proof is complete. □

3.5. **Rate of convergence of Algorithm 1.** We investigate the convergence rate of any infinite sequence which is generated by Algorithm 1. First, we assume a bound for $v(x)$, where $v(x)$ is the descent direction for ϕ at x. Then we provide a bound for $\theta(x^{k+1})$ with the help of information of former iteration point x^k. Also, we show that full Newton's steps are performed when k is large enough, that is, $\alpha_k = 1$. After using this, we prove that $\{x^k\}$ is converge to x^* with superlinear rate. At the end under some additional assumptions, we prove that $\{x^k\}$ is converge to x^* with quadratic rate.

Lemma 3.3. *If, for any k, there exists λ_{ij} such that $\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij} = 1$, then*

$$\theta(x^{k+1}) \ge -\frac{1}{2\omega}\left\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)\right\|^2,$$

where $\omega > 0$ and defined in 2.4.

Proof. Observe that

$$\theta(x^{k+1}) = \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x^{k+1},\xi_i) + \nabla\Psi_j(x^{k+1},\xi_i)^T v(x^{k+1}) + \frac{1}{2}v(x^{k+1})^T\nabla^2\Psi_j(x^{k+1},\xi_i)v(x^{k+1}) - \phi_j(x^{k+1})\}.$$

On account of equation (3.5), we obtain

$$\begin{aligned}\theta(x^{k+1}) &= \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x^{k+1},\xi_i) + \nabla\Psi_j(x^{k+1},\xi_i)^T v(x^{k+1}) \\ &\quad + \frac{1}{2}v(x^{k+1})^T\nabla^2\Psi_j(x^{k+1},\xi_i)v(x^{k+1}) - \phi_j(x^{k+1})\} \\ &\geq \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\{\nabla\Psi_j(x^{k+1},\xi_i)^T v(x^{k+1}) + \frac{1}{2}v(x^{k+1})^T\nabla^2\Psi_j(x^{k+1},\xi_i)v(x^{k+1})\} \\ &\geq \min_{v\in\mathbb{R}^n}\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\{\nabla\Psi_j(x^{k+1},\xi_i)^T v + \frac{1}{2}v^T\nabla^2\Psi_j(x^{k+1},\xi_i)v\}.\end{aligned}$$

Therefore,

$$\theta(x^{k+1}) \geq \min_{v\in\mathbb{R}^n}\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\big(\nabla\Psi_j(x^{k+1},\xi_i)^T v + \frac{\omega}{2}\|v\|^2\big). \tag{3.26}$$

After solving $\min_{v\in\mathbb{R}^n}\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\big(\nabla\Psi_j(x^{k+1},\xi_i)^T v + \frac{\omega}{2}\|v\|^2\big)$, we arrive at

$$\min_v \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\big(\nabla\Psi_j(x^{k+1},\xi_i)^T v + \frac{\omega}{2}\|v\|^2\big) = -\frac{1}{2\omega}\left\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)\right\|^2. \tag{3.27}$$

From (3.26) and (3.27), we have the desired conclusion immediately. □

Theorem 3.7. *Assume, for any $x\in\mathbb{R}^n$,*

$$\|v(x)\|^2 \leq \frac{2}{\omega}\mid\theta(x)\mid, \tag{3.28}$$

where $\omega > 0$ and is defined in 2.4. Let x^0 be the initial point in compact level-set ϕ and the sequence generated by Algorithm 1 is $\{x^k\}$. Then $\{x^k\}$ converges to a Pareto optimum point x^*. Moreover if $\alpha_k = 1$ for k large enough, then convergence rate of $\{x^k\}$ to x^* is superlinear.

Proof. Let$\{x^k\}$ be the sequence which is generated by Algorithm 1 whose initial point x^0 belongs to compact level set of ϕ. By Theorem 3.6, sequence $\{x^k\}$ converges to x^*. Next we prove that $\{x^k\}$ converges to x^* with the superlinear rate. Since x^* is a critical point, then there is no descent direction at x^* by the definition, i.e., $v(x^*) = 0$. Since $\Psi_j(x,\xi_i)$ is twice continuously differantiable for each x and ξ_i, one has

$$\begin{aligned}\Psi_j(x^k+v^k,\xi_i) &\leq \Psi_j(x^k,\xi_i) + \nabla\Psi_j(x^k,\xi_i)^T v^k + \frac{1}{2}(v^k)^T\nabla^2\Psi_j(x^k,\xi_i)(v^k) + \frac{\varepsilon}{2}\|v^k\|^2 \\ &\leq \max_{i\in\bar{\Lambda}}\{\Psi_j(x^k,\xi_i) + \nabla\Psi_j(x^k,\xi_i)^T v^k + \frac{1}{2}(v^k)^T\nabla^2\Psi_j(x^k,\xi_i)(v^k) + \frac{\varepsilon}{2}\|v^k\|^2\} \\ &\leq \max_{i\in\bar{\Lambda}}\Psi_j(x^k,\xi_i) + \max_{j\in\Lambda}\max_{i\in\bar{\Lambda}}\{\Psi_j(x^k,\xi_i) + \nabla\Psi_j(x^k,\xi_i)^T v^k \\ &\quad + \frac{1}{2}(v^k)^T\nabla^2\Psi_j(x^k,\xi_i)(v^k) - \phi_j(x^k)\} + \frac{\varepsilon}{2}\|v^k\|^2.\end{aligned}$$

Thus $\Psi_j(x^k+v^k,\xi_i) \le \phi_j(x^k)+\theta(x^k)+\frac{\varepsilon}{2}\|v^k\|^2$ holds for each i. It follows that

$$\max_{i\in\bar{\Lambda}} \Psi_j(x^k+v^k,\xi_i) \le \phi_j(x^k)+\theta(x^k)+\frac{\varepsilon}{2}\|v^k\|^2$$

and

$$\phi_j(x^k+v^k)-\phi_j(x^k) \le \theta(x^k)+\frac{\varepsilon}{2}\|v^k\|^2.$$

By (3.28), one has

$$\begin{aligned}\phi_j(x^k+v^k)-\phi_j(x^k) &\le \sigma\theta(x^k)+(1-\sigma)\theta(x^k)+\frac{\varepsilon}{2}\|v^k\|^2\\ &\le \sigma\theta(x^k)+(\varepsilon-\omega(1-\sigma))\frac{\|v^k\|^2}{2} \text{ for all } k\ge k_\varepsilon,\end{aligned}$$

which demonstrates that if $\varepsilon<\omega(1-\sigma)$, then by Algorithm 1, in Step 4, $\alpha_k=1$ is acceptable for $k\ge k_\varepsilon$. For superlinear convergence, we suppose that $\varepsilon<\omega(1-\sigma)$. Using the Taylor's first order expansion of $\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)$ and (3.6), we conclude that, for any $k\ge k_\varepsilon$,

$$\begin{aligned}\Big\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)\Big\| &= \Big\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)\\ &\quad -\big(\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x,\xi_i)+\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla^2\Psi_j(x,\xi_i)^T v\big)\Big\|\\ &\le \Big\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^k+v^k,\xi_i)-\big(\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^k,\xi_i)\\ &\quad +\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla^2\Psi_j(x^k,\xi_i)^T v^k\big)\Big\|\\ &\le \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\|\varepsilon v^k\|.\end{aligned}$$

Thus $\Big\|\sum_{j\in\Lambda}\sum_{i\in i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^{k+1},\xi_i)\Big\|\le\varepsilon\|v^k\|$, which in turn implies that $\theta(x^{k+1})\ge-\frac{1}{2\omega}(\varepsilon\|v^k\|)^2$ and

$$|\,\theta(x^{k+1})\,|\le\frac{1}{2\omega}(\varepsilon\|v^k\|)^2. \tag{3.29}$$

Since $\alpha_k=1$ and $x^{k+1}=x^k+v^k$, from (3.28) and (3.29), we have $\|x^{k+1}-x^{k+2}\|=\|v^{k+1}\|\le\frac{\varepsilon}{\omega}\|v^k\|$. Thus if $k\ge k_\varepsilon$ and $j\ge1$, then

$$\|x^{k+j}-x^{k+j+1}\|\le\left(\frac{\varepsilon}{\omega}\right)^j\|x^k-x^{k+1}\|. \tag{3.30}$$

To prove the superlinear convergence rate, take $\zeta\in(0,1)$ and define

$$\varepsilon^*=\min\left\{1-\sigma,\frac{\zeta}{1+2\zeta}\right\}\omega.$$

If $\varepsilon < \varepsilon^*$ and $k \geq k_\varepsilon$ then, by (3.30) and the convergence of $\{x^k\}$,

$$\|x^* - x^{k+1}\| \leq \sum_{j=1}^{\infty} \|x^{k+j} - x^{k+j+1}\| \leq \sum_{j=1}^{\infty} \left(\frac{\zeta}{1+2\zeta}\right)^j \|x^k - x^{k+1}\|.$$

Then

$$\|x^* - x^{k+1}\| \leq (\frac{\zeta}{1+\zeta})\|x^k - x^{k+1}\|. \tag{3.31}$$

Now with the help of triangle inequality and (3.31), we obtain

$$\begin{aligned}\|x^* - x^k\| &\geq \|x^k - x^{k+1}\| - \|x^{k+1} - x^*\| \\ &\geq \|x^k - x^{k+1}\| - (\frac{\zeta}{1+\zeta})\|x^k - x^{k+1}\|,\end{aligned}$$

so

$$\|x^* - x^k\| \geq (\frac{1}{1+\zeta})\|x^k - x^{k+1}\|. \tag{3.32}$$

By (3.31) and (3.32), we conclude if $\varepsilon < \varepsilon^*$ and $k \geq k_\varepsilon$, then $\|x^* - x^{k+1}\| \leq \zeta\|x^* - x^k\|$ and

$$\frac{\|x^* - x^{k+1}\|}{\|x^* - x^k\|} \leq \zeta.$$

Since ζ is arbitrary in $(0,1)$, one has that the above quotient tends to zero and hence the proof is complete. □

Next, we prove the quadratic convergence of the Newton's algorithm, Algorithm 1).

Theorem 3.8. *Let $x^0 \in \mathbb{R}^n$ be the initial point in a compact level-set of ϕ. Let $\{x^k\}$ be the sequence generated by Algorithm 1 and $\nabla^2\Psi_j(x,\xi_i)$ be Lipschitz continuous on $\mathbb{R}^n$ for each ξ_i and $j \in \Lambda$. Then $\{x^k\}$ converges to x^* with quadratic rate.*

Proof. Since $\nabla^2\Psi_j(x,\xi_i)$ is Lipschitz continuous on $\mathbb{R}^n$ for each ξ_i $j \in \Lambda$, then

$$\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{lj}^k \nabla^2\Psi_j(x^k,\xi_i)$$

is also Lipschitz continuous. If we take $\alpha_k = 1$ for large k, then $x^{k+1} - x^k = v^k$,

$$\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla\Psi_j(x^k,\xi_i) + \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla^2\Psi_j(x^k,\xi_l)^T v = 0$$

$$\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla\Psi_j(x^k,\xi_i) = -\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla^2\Psi_j(x^k,\xi_i)^T v$$

$$\begin{aligned}\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla\Psi_j(x^k,\xi_i)\| &= \|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \nabla^2\Psi_j(x^k,\xi_i)^T v\| \\ &\leq \sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}} \lambda_{ij}^k \frac{\bar{L}}{2}\|v^k\|^2, \text{ where } \bar{L} \text{ is Lipschitz constant} \\ &\leq \frac{\bar{L}}{2}\|v^k\|^2.\end{aligned}$$

So $\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{ij}^k\nabla\Psi_j(x^k,\xi_i)\leq\frac{\bar{L}}{2}\|v^k\|^2$. From Lemma 3.3, we have

$$\theta(x^{k+1})\geq-\frac{1}{2\omega}\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{lj}^k\nabla\Psi_j(x^{k+1},\xi_i)\|^2$$

$$-\theta(x^{k+1})\leq\frac{1}{2\omega}\|\sum_{j\in\Lambda}\sum_{i\in\bar{\Lambda}}\lambda_{lj}^k\nabla\Psi_j(x^{k+1},\xi_i)\|^2$$

and

$$-\theta(x^{k+1})\leq\frac{1}{2\omega}(\frac{\bar{L}}{2}\|v^k\|^2)^2.$$

Thus

$$|\,\theta(x^{k+1})\,|\leq\frac{1}{2\omega}(\frac{\bar{L}}{2}\|v^k\|^2)^2. \tag{3.33}$$

From (3.28), we have

$$\|v(x^{k+1})\|^2\leq\frac{2}{\omega}\,|\,\theta(x^{k+1})\,|\,. \tag{3.34}$$

Using (3.33) and (3.34), we have $\|v(x^{k+1})\|^2\leq\frac{1}{\omega^2}(\frac{\bar{L}}{2}\|v^k\|^2)^2$ and $\|v(x^{k+1})\|\leq\frac{\bar{L}}{2\omega}\|v^k\|^2$ for k large enough. Take $\zeta\in(0,1)$. Since $\{x^k\}$ converges to x^* with superlinear rate, then there exists k_0 such that $k\geq k_0$, and $\|x^*-x^{k+1}\|\leq\zeta\|x^*-x^k\|$. If we apply the triangle inequality in $\|x^*-x^s\|$ and $\|x^{s+1}-x^s\|$ for $s\geq k_0$, then

$$\|x^*-x^s\|\leq\|x^*-x^{s+1}\|+\|x^s-x^{s+1}\|\leq\zeta\|x^*-x^s\|+\|x^s-x^{s+1}\|,$$

which implies

$$(1-\zeta)\|x^*-x^s\|\leq\|x^s-x^{s+1}\|. \tag{3.35}$$

Observe that

$$\|x^s-x^{s+1}\|\leq\|x^*-x^s\|+\|x^*-x^{s+1}\|\leq\|x^*-x^s\|+\zeta\|x^*-x^s\|,$$

which in turn implies

$$\|x^s-x^{s+1}\|\leq(1+\zeta)\|x^*-x^s\|. \tag{3.36}$$

From inequalities (3.35) and (3.36), we see that

$$(1-\zeta)\|x^*-x^s\|\leq\|x^s-x^{s+1}\|\leq(1+\zeta)\|x^*-x^s\|. \tag{3.37}$$

For $s=k\geq k_0$, we obtain from (3.36) that

$$\|x^k-x^{k+1}\|\leq(1+\zeta)\|x^*-x^k\| \tag{3.38}$$

and while using the inequality (3.35) for $s=k+1$ yields

$$(1-\zeta)\|x^*-x^{k+1}\|\leq\|x^{k+1}-x^{k+2}\|=\|v^{k+1}\|.$$

In view of

$$(1-\zeta)\|x^*-x^{k+1}\|\leq\|x^{k+1}-x^{k+2}\|=\|v^{k+1}\|\leq\frac{\bar{L}}{\sqrt{2}\omega}\|v^k\|^2,$$

we have

$$(1-\zeta)\|x^*-x^{k+1}\|\leq\frac{\bar{L}}{\sqrt{2}\omega}\|v^k\|^2=\frac{\bar{L}}{\sqrt{2}\omega}\|x^{k+1}-x^k\|^2.$$

By (3.38), we have

$$(1-\zeta)\|x^*-x^{k+1}\| \leq \frac{\bar{L}}{\sqrt{2}\omega}(1+\zeta)^2\|x^*-x^k\|^2. \tag{3.39}$$

Since $\zeta \in (0,1)$ was arbitrary, we can find from (3.39) that $\{x^k\}$ converges to x^* quadratically. □

In the following subsection, we discuss some numerical examples.

3.6. **Numerical examples.** In this subsection, we provide some numerical examples and compare with weighted sum method. Python code is written for Algorithm 1. The subproblem $P(x)$ is solved using *cvxopt.solvers.qp*. In our calculations, $\|v^k\| < 10^{-4}$ or $\alpha_k < 10^{-5}$ or maximum iteration 5000 is considered as stopping criteria. The solutions of a multiobjective function are not isolated minimum points but a set of efficient solutions. To generate a well distributed approximation of Pareto front, we e use the multi-start technique. Here 100 uniformly distributed random points are chosen between lb and ub (where $lb, ub \in \mathbb{R}^n$ and $lb < ub$) and Algorithm 1 is executed separately. The nondominated set of the collection of critical points is considered as an approximate Pareto front. We compare the approximate Pareto fronts obtained by Algorithm 1 with the approximate Pareto fronts obtained by scalarization methods. In weighted sum method, we solve the following single objective optimization problem

$$\min_{x\in\mathbb{R}^n} \left(w_1\phi_1(x)+w_2\phi_2(x)+\cdots+w_m\phi_m(x)\right),$$

where $(w_1,w_2,\ldots,w_m) = w \geq 0$, and $w \neq 0$ is based on the technique developed in [22] with initial approximation $x^0 = \frac{1}{2}(lb+ub)$. For bi-objective optimization problems, we consider weights $(1,0)$, $(0,1)$, and 98 random weights uniformly distributed in $(0,0)$ and $(1,1)$. For three objective optimization problems, we consider weights $(1,0,0)$, $(0,1,0)$, $(0,0,1)$ and 97 random weights uniformly distributed in $(0,0,0)$ and $(1,1,1)$.

In state of $P(x)$, we solve the following subproblem in given examples:

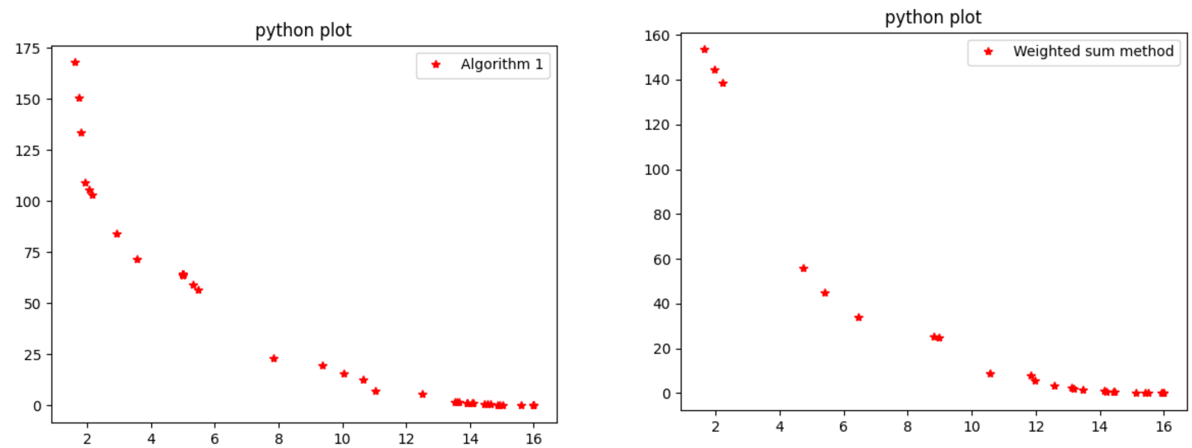
$$P(x): \min_{v\in\mathbb{R}^n, t\in\mathbb{R}} \rho(t,v) = t$$

$$s.t.\ \Psi_j(x,\xi_i)+\nabla\Psi_j(x,\xi_i)^T v+\frac{1}{2}v^T\nabla^2\Psi_j(x,\xi_i)v-\phi_j(x) \leq t,\ \forall i \in \bar{\Lambda} \text{ and } j \in \Lambda$$
$$lb \leq x+v \leq ub.$$

In Example 3.1, we explain the steps of Algorithm 1 by using one initial approximation, and demonstrate that the sequence generated by Algorithm 1 converges to the approximate weak efficient solution. At the end of the example, the comparison of approximate Pareto front generated by Newton's method (Algorithm 1) with the approximate Pareto front generated by weighted sum method is demonstrated.

Example 3.1. (Two dimensional bi-objective convex problem under uncertainty set of two elements) Consider the uncertain bi-objective optimization problem $P(U) = \{P(\xi) : \xi \in U\}$ such that $P(\xi) : \min_{x\in\mathbb{R}^2}\Psi(x,\xi)$, where $\Psi(x,\xi) = (\Psi_1(x,\xi),\Psi_2(x,\xi))$, $\Psi : \mathbb{R}^2\times U \to \mathbb{R}^2$, $\xi \in U = \{(2,2),\ (0,4)\} \subset \mathbb{R}^2$, and $\Psi(x,\xi) = ((x_1-\xi_1)^2+(x_2+\xi_2)^2$, and $(\xi_1x_1+\xi_2x_2)^2)$, where $\xi = (\xi_1,\xi_2)$. Here $\Psi_1(x,\xi) = (x_1-\xi_1)^2+(x_2+\xi_2)^2$ and $\Psi_2(x,\xi) = (\xi_1x_1+\xi_2x_2)^2$.

The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP : \min_{x\in\mathbb{R}^2}\phi(x)$, where $\phi(x) = (\phi_1(x),\phi_2(x))$, $\phi_1(x) = \max_{\xi\in U}\Psi_1(x,\xi) = \max_{\xi\in U}\{(x_1-2)^2+(x_2+2)^2, x_1^2+$



(A) The Approximate Pareto front of Example 3.1 via the Newton's Method (Algorithm 1).

(B) The Approximate Pareto front of Example 3.1 via the weighted sum method

FIGURE 1. The comparison of approximate Pareto fronts generated by 'Newton's method' with 'weighted sum method' of the Example 3.1

$(x_2+4)^2$ }, and $\phi_2(x) = \max_{\xi\in U}\Psi_2(x,\xi) = \max\{(2x_1+2x_2)^2, 16x_2^2\}$. Consider $x^0 = (2.522138$ $83,\ 4.24413089)$. Then $\Psi_1(x^0,\xi_1) = 41.30607718$, $\Psi_1(x^0,\xi_2) = 74.3268784$, $\Psi_2(x^0,\xi_1) = 183.12962\ 375$, $\Psi_2(x^0,\xi_2) = 288.20235207$, $\phi_1(x^0) = 74.3268784$, $\phi_2(x^0) = 288.20235207$, and $\phi(x^0) = (74.3268784, 288.20235207)$. Also, we can observe that $I_1(x^0) = \{2\}$ and $I_2(x^0) = \{2\}$. The solution of $P(x^0)$ is obtained as $v^0 = (-1.52211667, -6.24402446)^T$, $t^0 = -41.30541$ 769. We see that $\alpha_0 = 1$ satisfies (3.17). Hence, next iterating point is $x^1 = x^0 + \alpha_0 v^0 = (1.00002216, -1.99989357)^T$. One can observe that $\phi(x^1) = (5.00047007, 63.99318849)^T < \phi(x^0) = (74.3268784, 288.20235207)$. Using the stopping criteria $\|v^k\| < 10^{-4}$, the final solution is obtained as $x^* = (1.00002216, -1.99989357)^T$ after one iterations.

Next, we show that $x^* = (1.00002216,\ -1.99989357)^T$ is a week efficient solution to this problem. Observe that $\Psi_1(x^*,\xi_1) = 1.18191374e^{-08}$, $\Psi_1(x^*,\xi_2) = 5.00047007$ and $\Psi_2(x^*,\xi_1) = 3.99897129$, $\Psi_2(x^*,\xi_2) = 63.9931885$, and $\phi(x^*) = (5.00047007,\ 63.99318849)$. These imply $I_1(x^*) = \{2\}$ and $I_2(x^*) = \{2\}$. Hence,

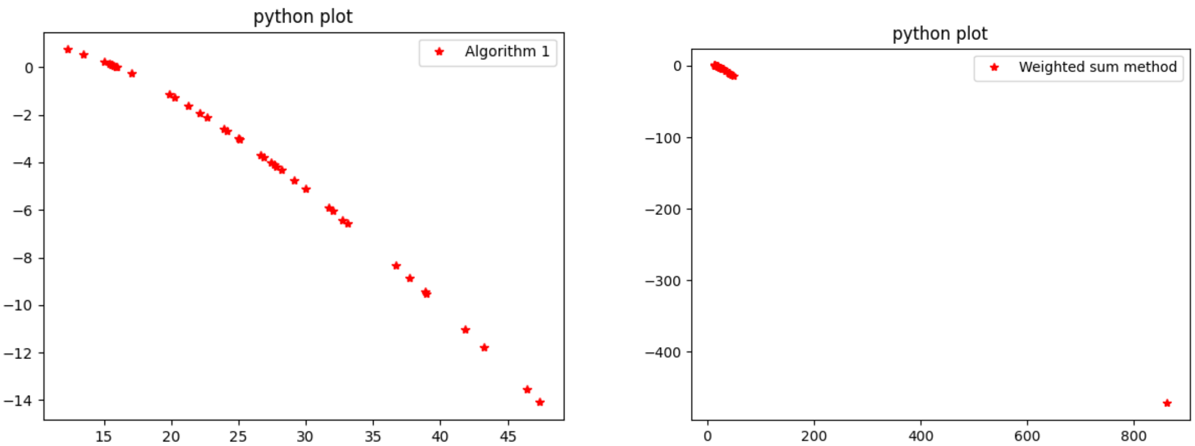
$$\begin{aligned}\partial\phi_1(x^*) &= conv\{\nabla\psi_1(x,\xi_i) : i \in I_1(x))\}\\ &= conv\{\nabla\psi_1(x,\xi_2)\}\\ &= conv\{(2.00004432,\ 4.00021286)^T\}\end{aligned}$$

and

$$\partial\phi_2(x^*) = conv\{(0,\ 63.993187)^T\}.$$

Clearly, x^* is a approximate critical point for *RP*. Thus, 0 is approximately lie in the convex combination of $\partial\phi_1(x^*)$ and $\partial\phi_2(x^*)$. Hence, by Theorem 3.2, x^* is a weak efficient solution $\phi(x)$. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.1 is given in Figure 1.

Example 3.2. (One dimensional bi-objective non-convex problem under uncertainty set of two elements) Consider the uncertain bi-objective optimization problem $P(U) = \{P(\xi) : \xi \in U\}$ such that $P(\xi) : \min_{x\in\mathbb{R}}\Psi(x,\xi)$, where $\Psi(x,\xi) = (\Psi_1(x,\xi), \Psi_2(x,\xi))$, $\Psi : \mathbb{R}\times U \to \mathbb{R}^2$,



(A) The Approximate Pareto front of Example 3.2 via the Newton's method

(B) The Approximate Pareto front of Example 3.2 via the weighted sum method

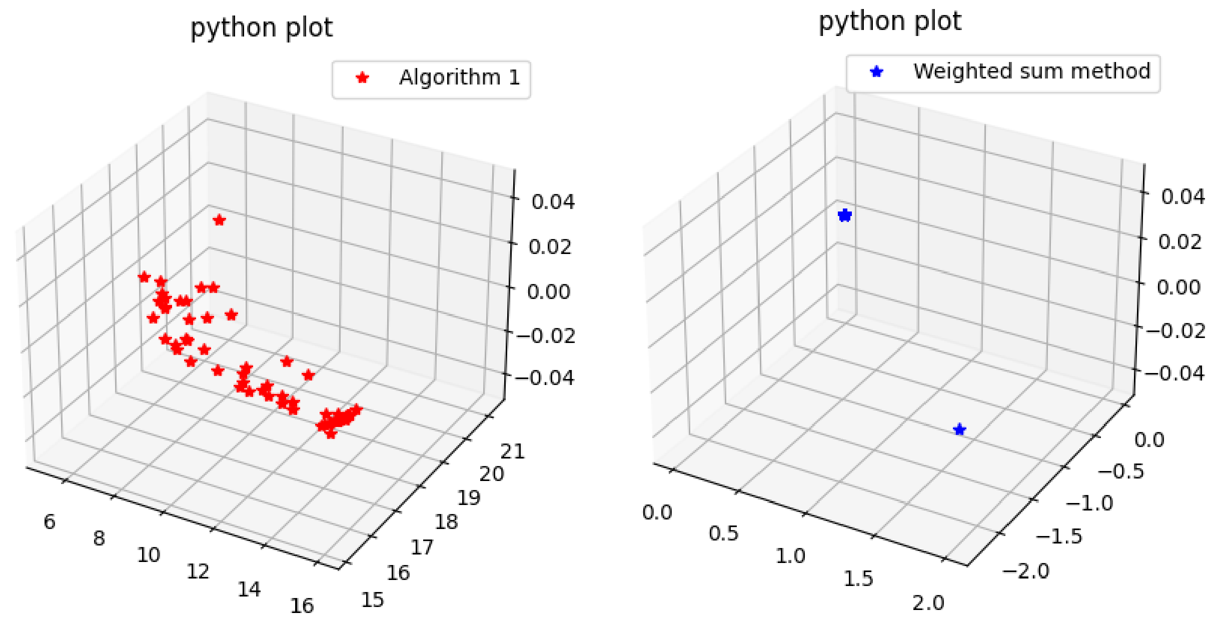
FIGURE 2. The Comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.2

$\xi \in U = \{-2, 5\} \subset \mathbb{R}$, $\Psi(x,\xi) = ((x-\xi)^2,\ -x^2+\xi x)$, where $\xi \in U$. Here $\Psi_1(x,\xi) = (x-\xi)^2$ and $\Psi_2(x,\xi) = -x^2 - \xi x$. The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP : \min_{x\in\mathbb{R}^2} \phi(x)$, where $\phi(x) = (\phi_1(x), \phi_2(x))$, $\phi_1(x) = \max_{\xi\in U} \Psi_1(x,\xi) = \max\{(x+2)^2, (x-5)^2\}$, and $\phi_2(x) = \max_{\xi\in U} \Psi_2(x,\xi) = \max\{-x^2+2x, -x^2-5x\}$. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.2 is given in Figure 2.

Example 3.3. (Three dimensional three objective non-convex problem under uncertainty set of three elements) Consider the uncertain three-objective optimization problem $P(U) = \{P(\xi) : \xi \in U\}$ such that $P(\xi) : \min_{x\in\mathbb{R}^3} \Psi(x,\xi)$, where $\Psi(x,\xi) = (\Psi_1(x,\xi), \Psi_2(x,\xi), \Psi_3(x,\xi))$, $\Psi : \mathbb{R}^3 \times U \to \mathbb{R}^3$, $\xi \in U = \{(4,1),\ (0,2),\ (1,0)\}$, and $\Psi(x,\xi) = (x_1^2 + (x_2-\xi_1)^2 - \xi_2 x_3^2,$ $\xi_1 x_1 + \xi_2 x_2^2 + x_3 + 4\xi_1\xi_2$, and $\xi_1 x_1^2 + 6x_2^2 + 25(x_3 - \xi_2 x_1)^2)$, where $\xi = (\xi_1, \xi_2)$, $\Psi_1(x,\xi) = x_1^2 + (x_2-\xi_1)^2 - \xi_2 x_3^2$, $\Psi_2(x,\xi) = \xi_1 x_1 + \xi_2 x_2^2 + x_3 + 4\xi_1\xi_2$, and $\Psi_3(x,\xi) = \xi_1 x_1^2 + 6x_2^2 + 25(x_3 - \xi_2 x_1)^2$. The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP : \min_{x\in\mathbb{R}^3} \phi(x)$, where $\phi(x) = (\phi_1(x),\ \phi_2(x), \phi_3(x))$, $\phi_1(x) = \max_{\xi\in U} \Psi_1(x,\xi) = \max\{x_1^2 + (x_2-4)^2 - x_3^2,\ x_1^2 + x_2^2 - 2x_3^2,\ x_1^2 + (x_2-1)^2\}$, $\phi_2(x) = \max_{\xi\in U} \Psi_2(x,\xi) = \max\{4x_1 + x_2^2 + x_3 + 16,\ 2x_2^2 + x_3,\ x_1 + x_3\}$, and $\phi_3(x) = \max_{\xi\in U} \Psi_2(x,\xi) = \{4x_1^2 + 6x_2^2 + 25(x_3 - x_1)^2, 6x_2^2 + 25(x_3 - 2x_1)^2 + x_1^2 + 25x_3^2\}$. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.3 is given in Figure 3.

Example 3.4. (Two dimensional three-objective non-convex problem under uncertainty set of three elements) Consider the uncertain three objective optimization problem $P(U) = \{P(\xi) : \xi \in U\}$ such that $P(\xi) : \min_{x\in\mathbb{R}^2} \Psi(x,\xi)$, where $\Psi(x,\xi) = (\Psi_1(x,\xi), \Psi_2(x,\xi), \Psi_3(x,\xi))$, $\Psi : \mathbb{R}^2 \times U \to \mathbb{R}^3$, $\xi \in U = \{(2,3),\ (4,5),\ (2,0)\}$, $\Psi(x,\xi) = (x_1^2 + \xi_1 x_2^4 + \xi_1\xi_2 x_1 x_2, 5x_1^2 + \xi_1 x_2^2 + \xi_2 x_1^4 x_2, e^{-\xi_1 x_1 + \xi_2 x_2} + x_1^2 - \xi_1 x_2^2)$, where $\xi = (\xi_1, \xi_2)$. Here $\Psi_1(x,\xi) = x_1^2 + \xi_1 x_2^4 + \xi_1\xi_2 x_1 x_2$, $\Psi_2(x,\xi) = 5x_1^2 + \xi_1 x_2^2 + \xi_2 x_1^4 x_2$, and $\Psi_3(x,\xi) = e^{-\xi_1 x_1 + \xi_2 x_2} + x_1^2 - \xi_1 x_2^2$.

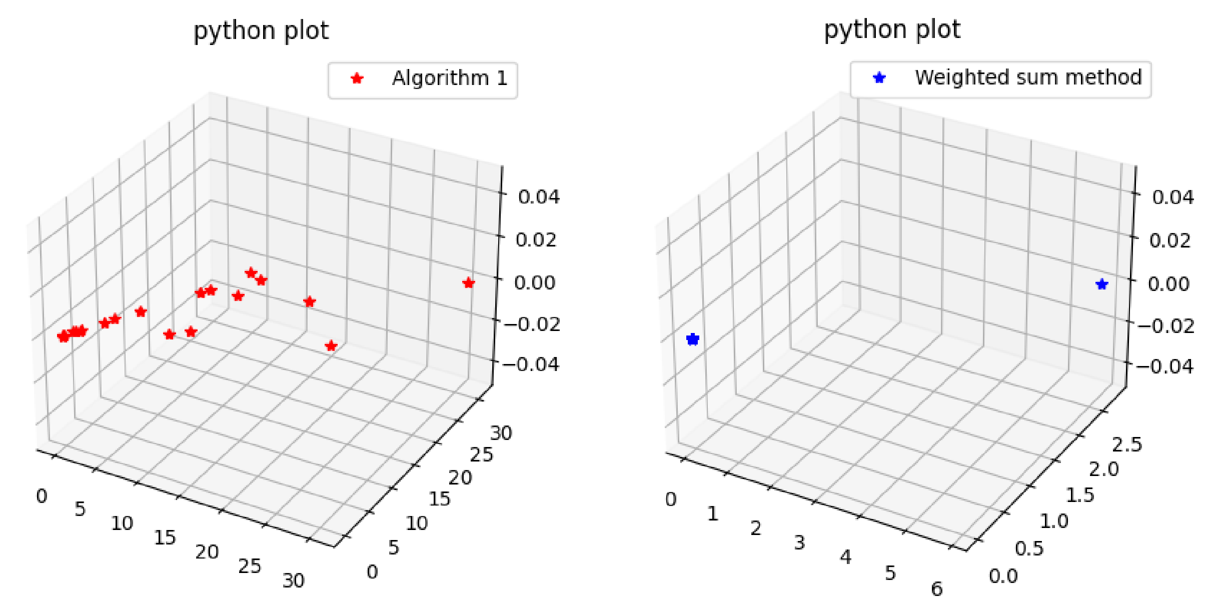
The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP : \min_{x\in\mathbb{R}^2} \phi(x)$, where $\phi(x) = (\phi_1(x), \phi_2(x))$, $\phi_1(x) = \max_{\xi\in U} \Psi_1(x,\xi) = \max_{\xi\in U}\{x_1^2 + 2x_2^4 + 6x_1x_2,\ x_1^2 + 4x_2^4 +$



(A) The Approximate Pareto front of Example 3.3 via the Newton's Method.

(B) The Approximate Pareto front of Example 3.3 via the weighted sum method

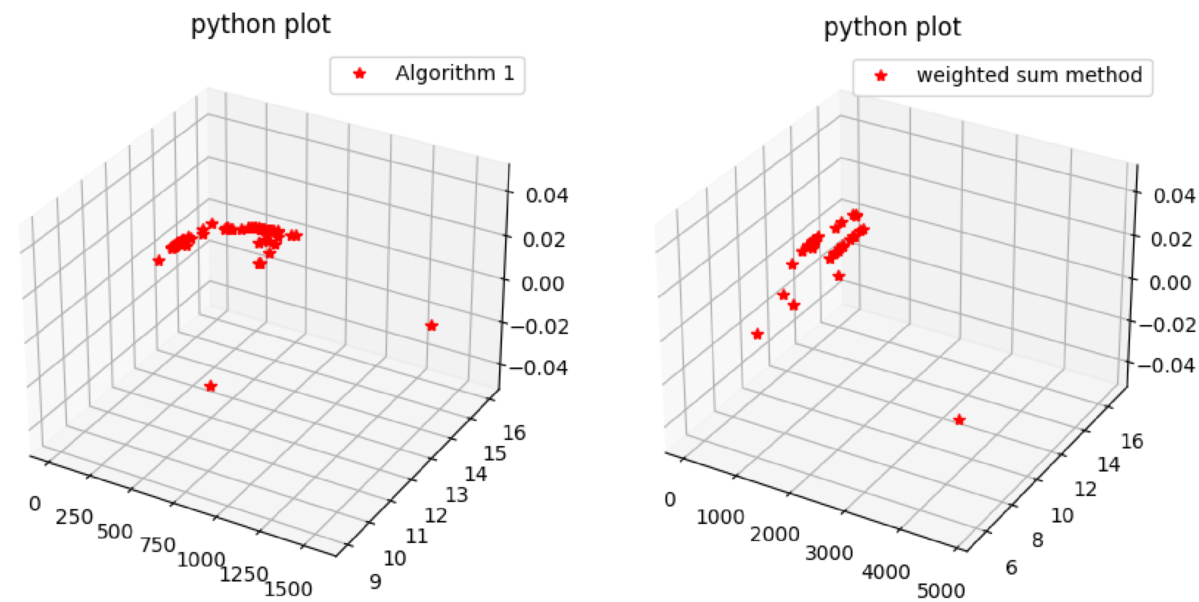
FIGURE 3. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of Example 3.3



(A) The Approximate Pareto front of Example 3.4 via the Newton's method

(B) The Approximate Pareto front of Example 3.4 via the weighted sum method

FIGURE 4. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.4



(A) The Approximate Pareto front of Example 3.5 via the Newton's method

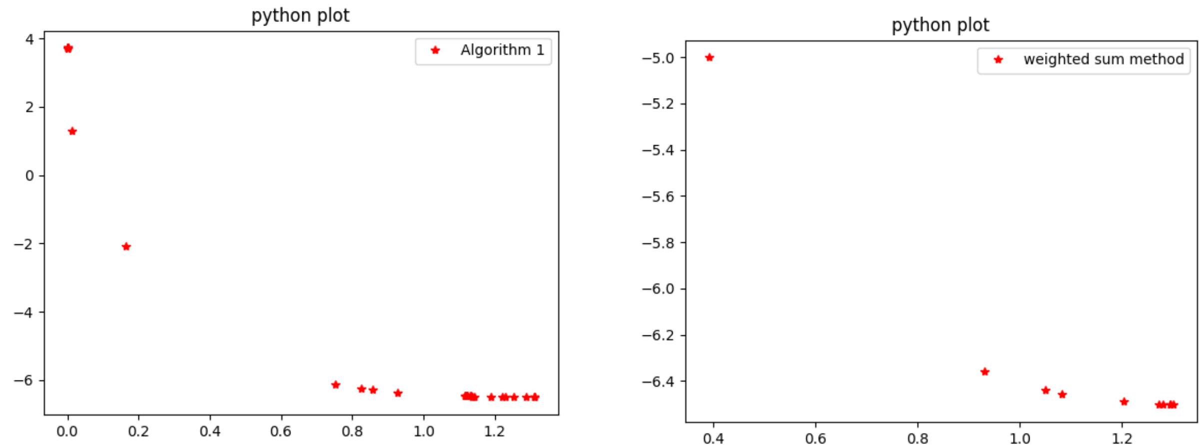
(B) The Approximate Pareto front of Example 3.5 via the weighted sum method

FIGURE 5. The Comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of Example 3.5

$20x_1x_2,\ x_1^2+2x_2^4\}$, $\phi_2(x)=\max_{\xi\in U}\Psi_2(x,\xi)=\max\{5x_1^2+2x_2^2+3x_1^4x_2,\ 5x_1^2+4x_2^2+5x_1^4x_2,\ 5x_1^2+2x_2^2\}$, $\phi_3(x)=\max_{\xi\in U}\Psi_3(x,\xi)=\max\{e^{-2x_1+3x_2}+x_1^2-2x_2^2,\ e^{-4x_1+5x_2}+x_1^2-4x_2^2,\ e^{-2x_1}+x_1^2-2x_2^2\}$. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.4 is given in Figure 4.

Example 3.5. (Two dimensional three objective convex problem under uncertainty set of three elements) Consider the uncertain bi-objective optimization problem $P(U)=\{P(\xi):\xi\in U\}$ such that $P(\xi):\min_{x\in\mathbb{R}^2}\Psi(x,\xi)$, where $\Psi(x,\xi)=(\Psi_1(x,\xi),\Psi_2(x,\xi))$, $\Psi:\mathbb{R}^2\times U\to\mathbb{R}^3$, $\xi\in U=\{(2,3),\ (1,2),\ (4,5)\}$, $\Psi(x,\xi)=(100\xi_1(x_2-x_1^2)^2+\xi_2(1-x_1)^2,(x_2-\xi_1)^2+\xi_2x_1^2,\ \xi_1x_1^2+3\xi_2x_2^2)$, where $\xi=(\xi_1,\xi_2)$, $\Psi_1(x,\xi)=100\xi_1(x_2-x_1^2)^2+\xi_2(1-x_1)^2$, $\Psi_2(x,\xi)=(x_2-\xi_1)^2+\xi_2x_1^2$, and $\Psi_3(x,\xi)=\xi_1x_1^2+3\xi_2x_2^2$. The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP:\min_{x\in\mathbb{R}^2}\phi(x)$, where $\phi(x)=(\phi_1(x),\phi_2(x),\phi_3(x))$, $\phi_1(x)=\max_{\xi\in U}\Psi_1(x,\xi)=\max\{200(x_2-x_1^2)^2+3(1-x_1)^2,\ 100(x_2-x_1^2)^2+2(1-x_1)^2,\ 400(x_2-x_1^2)^2+5(1-x_1)^2\}$, $\phi_2(x)=\max_{\xi\in U}\Psi_2(x,\xi)=\max\{(x_2-2)^2+3x_1^2,\ (x_2-1)^2+2x_1^2,\ (x_2-4)^2+5x_1^2\}$, and $\phi_3(x)=\max_{\xi\in U}\Psi_3(x,\xi)=\max\{2x_1^2+9x_2^2,\ x_1^2+6x_2^2,\ 4x_1^2+20x_2^2\}$. The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.5 is given in Figure 5.

Example 3.6. (Two dimensional bi-objective convex problem under uncertainty set of two elements) Consider the uncertain bi-objective optimization problem $P(U)=\{P(\xi):\xi\in U\}$ such that $P(\xi):\min_{x\in\mathbb{R}^2}\Psi(x,\xi)$, where $\Psi(x,\xi)=(\Psi_1(x,\xi),\Psi_2(x,\xi))$, $\Psi:\mathbb{R}^2\times U\to\mathbb{R}^2$, $\xi\in U=\{(1,2,2),\ (1,3,0)\}\subset\mathbb{R}^3$, and $\Psi(x,\xi)=(\xi_1x_1^2+\xi_2x_2^2+\xi_1x_1+\xi_1\xi_3x_2,\ (\xi_1+\xi_2x_2)^2+\xi_1x_1+x_2+10(x_1+\xi_3x_2)+e^{(1+\xi_1x_1+\xi_2x_2)^2}$, where $\xi=(\xi_1,\xi_2,\xi_3)$, $\Psi_1(x,\xi)=\xi_1x_1^2+\xi_2x_2^2+$



(A) The Approximate Pareto front of Example 3.6 via the Newton's method

(B) The Approximate Pareto front of Example 3.6 via the weighted sum method

FIGURE 6. The Comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of Example 3.6

$\xi_1 x_1 + \xi_1\xi_3 x_2$, and $\Psi_2(x,\xi) = (\xi_1+\xi_2 x_2)^2 + \xi_1 x_1 + x_2 + 10(x_1+\xi_3 x_2) + e^{(1+\xi_1 x_1+\xi_2 x_2)^2}$. The objective wise worst case cost type robust counterpart to $P(U)$ is given by $RP : \min_{x\in\mathbb{R}^2} \phi(x)$, where $\phi(x) = (\phi_1(x), \phi_2(x))$ and

$$\phi_1(x) = \max_{\xi\in U} \Psi_1(x,\xi) = \max_{\xi\in U}\{x_1^2 + 2x_2^2 + x_1 + 2x_2,\ x_1^2 + 3x_2^2 + x_1\}$$

and

$$\phi_2(x) = \max_{\xi\in U} \Psi_2(x,\xi)$$

$$= \max_{\xi\in U}\{(1+2x_2)^2 + x_1 + x_2` + 10(x_1+2x_2) + e^{(1+x_1+2x_2)^2}, (1+3x_2)^2 + 11x_1 + x_2 + e^{(1+3x_1)^2}\}.$$

The comparison of approximate Pareto front generated by 'Newton's method' with 'weighted sum method' of the Example 3.6 is given in Figure 6.

With the help of the figures presented above, we observe that the Newton's method (Algorithm 1) generates a good approximate Pareto front for both convex and non-convex problems whereas the weighted sum method fails to generate good approximate Pareto front for the non-convex problem.

4. CONCLUSIONS

In this paper, we solved a deterministic nonsmooth multiobjective optimization problem *RP*, which is a robust counterpart to the uncertain multiobjective optimization problem (1.1) namely $P(U)$ with finite uncertainty sets. We investigated the critical point for *RP*, which is a non-smooth multiobjective optimization problem. To do that, we developed a Newton's algorithm and solved a subproblem whose solution gives us the Newton's descent direction, and then we developed the Armijo type inexact line search technique for step length sizes. With the help of Newton's algorithm (Algorithm 1), we defined a sequence that converges to a critical point, and that point is the weakly efficient solution or efficient solution to *RP*, and robust efficient solution to $P(U)$. Under some assumptions, we proved that sequence generated by Newton's algorithm (Algorithm 1) converges to a critical point with a superlinear and quadratic rates. At the end of this paper, we verified the Newton's algorithm (Algorithm 1) with the help of some numerical examples, and also we compared the approximate Pareto front generated by Newton's method (Algorithm 1) with the weighted sum method. It is observed that Newton's method (Algorithm 1) generates a good approximate Pareto front for both convex and non-convex problems whereas the weighted sum method fails to generate good approximate Pareto front for the non-convex problem.

Acknowledgments

The authors are grateful to the reviewers for useful suggestions which improved the contents of this paper. Debdas Ghosh acknowledges the research grants MATRICS (MTR/2021/000696) and Core Research Grant (CRG/2022/001347) from Science and Engineering Research Board, India, to carry out this research work.

REFERENCES

[1] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust optimization, Princeton Series in Applied Mathematics, Princeton University Press, Princeton, 2009.

[2] A. Ben-Tal, A. Nemirovski, Selected topics in robust convex optimization, Math. Program. 112 (2008), 125-158.

[3] M. Ehrgott, Multicriteria Optimization, Springer, 2005.

[4] K. Miettinen, Nonlinear multiobjective optimization, International Series in Operations Research & Management Science, 12. Kluwer Academic Publishers, Boston, 1999.

[5] J. Fliege, L. G. Drummond, B. F. Svaiter, Newton's method for multiobjective optimization, SIAM J. Optim. 20 (2009), 602-626.

[6] K. Kumar, D. Ghosh, A. Upadhayay, J.C. Yao, X. Zhao, Quasi-Newton methods for multiobjective optimization problems: A systematic review, Appl. Set-Valued Anal. Optim. 5 (2023), 291-321.

[7] J. Thomann, G. Eichfelder, Representation of the Pareto front for heterogeneous multi-objective optimization, J. Appl. Numer. Optim. 1 (2019), 293-323.

[8] B. Gebken, S. Peitz, An efficient descent method for locally Lipschitz multiobjective optimization problems, J. Optim. Theory Appl. 188 (2021), 696-723.

[9] S. K. Mishra, G. Panda, M. A. T. Ansary, B. Ram, On q-Newton's method for unconstrained multiobjective optimization problems, J. Appl. Math. Comput. 63 (2020), 391-410.

[10] M. A. T. Ansary, G. Panda, A modified quasi-Newton method for vector optimization problem, Optimization 64 (2015), 2289-2306.

[11] J. Fliege, B. F. Svaiter, Steepest descent methods for multicriteria optimization, Math. Oper. Res. 51 (2000), 479-494.

[12] M. A. T. Ansary, G. Panda, A globally convergent SQCQP method for multiobjective optimization problems, SIAM J. Optim. 31 (2021), 91-113.

[13] M. A. T. Ansary, G. Panda, A sequential quadratic programming method for constrained multi-objective optimization problems, J. Comput. Appl. Math. 64 (2020), 379-397.

[14] M. A. T. Ansary, G. Panda, A sequential quadratically constrained quadratic programming technique for a multi-objective optimization problem, Eng. Optim. 51 (2019), 22-41.

[15] L. G. Drummond, B. F. Svaiter, A steepest descent method for vector optimization, J. Comput. Appl. Math. 175 (2005), 395-414.

[16] V. F. Demyanov, V. N. Malozemov, Introduction to minimax, Courier corporation, (1990), 1-25.

[17] J. Nocedal, S. Wright, Numerical optimization, Second edition, Springer Series in Operations Research and Financial Engineering, Springer, New York, 2006.

[18] W. Sun, Y.X. Yuan, Optimization theory and methods. Nonlinear programming, Springer Optimization and Its Applications, Springer, New York, 2006.

[19] M. Ehrgott, J. Ide, A. Schobel, Minmax robustness for multi-objective optimization problems, European J. Oper. Res. 239 (2014), 17-31.

[20] M. M. Makela, V. P. Eronen, N. Karmitsa, On nonsmooth multiobjective optimality conditions with generalized convexities, Optimization in Science and Engineering, pp. 333-357, Springer, New York, 2014.

[21] E. H. Fukuda, L. M. G. Drummond, A survey on multiobjective descent methods, Pesq. Oper. 34 (2014), 585-620.

[22] M. S. Bazaraa, J. J. Goode, An algorithm for solving linearly constrained minimax problems, European J. Oper. Res. 11 (1982), 158-166.