AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 5(6): 5521–5540.
DOI: 10.3934/math.2020354
Received: 28 April 2020
Accepted: 22 June 2020
Published: 28 June 2020

Research article

On q-steepest descent method for unconstrained multiobjective optimization problems

Kin Keung Lai[1,*], **Shashi Kant Mishra** [2], **Geetanjali Panda** [3], **Md Abu Talhamainuddin Ansary**[4] **and Bhagwat Ram** [5]

[1] College of Economics, Shenzhen University, Shenzhen 518060, China
[2] Department of Mathematics, Institute of Science, Banaras Hindu University, Varanasi 221005, India
[3] Department of Mathematics, Indian Institute of Technology, Kharagpur 721302, India
[4] Department of Economic Sciences, Indian Institute of Technology Kanpur, Kanpur 208016 India
[5] DST-Centre for Interdisciplinary Mathematical Sciences, Institute of Science, Banaras Hindu University, Varanasi 221005, India

* **Correspondence:** Email: mskklai@outlook.com.

Abstract: The q-gradient is the generalization of the gradient based on the q-derivative. The q-version of the steepest descent method for unconstrained multiobjective optimization problems is constructed and recovered to the classical one as q equals 1. In this method, the search process moves step by step from global at the beginning to particularly neighborhood at last. This method does not depend upon a starting point. The proposed algorithm for finding critical points is verified in the numerical examples.

Keywords: multiobjective; q-calculus; steepest descent; pareto optimality; critical point; algorithms
Mathematics Subject Classification: 90C29, 05A30, 34M60, 58E17, 90C53, 11Y16

1. Introduction

In multiobjective problems, several objective functions have to be minimized or maximized simultaneously. There are conflicts between the objective functions. It is not possible to get a single point that minimizes all objective functions simultaneously. Therefore, the concept of optimality is replaced by the concept of Pareto optimality, a measure of efficiency in multiobjective problems [1]. A point is called Pareto-optimal or efficient, if there does not exist a different point with same or smaller objective function values such that there is a decrease in at least one objective function value. Applications of such optimization problems can be found in space exploration [2, 3], engineering [4], truss optimization [5, 6], design [7, 8], environmental analysis [9, 10], statistics [11], management science [12–14], economic sciences [15], etc.

There are many solution strategies for finding a Pareto optimal point of unconstrained multiobjective optimization problems (UP). One popular approach is to reformulate (UP) as a scalar optimization problem whose objective function is a weighted linear combination of all objective functions. However, this approach may yield unbounded scalar problems [20], if the weighted values are not properly chosen. Another popular approach is the evolutionary algorithm which searches for a Pareto optimum in a set of candidate solutions with some genetic operator. No convergence proofs are known in this case, and empirical results show that convergence is quite slow [16]. Another general solution strategy for multiobjective optimization problems is ϵ-constrained method [17]. A drawback of this method is that ϵ may be selected so that the feasible region becomes empty. Other parameter-free multiobjective optimization techniques use the ordering of the different criteria, that is, an ordering of the objective functions based on priority. In this case, the ordering has to be prespecified. Moreover, the corresponding optimization process is usually augmented by an interactive procedure [18]. Another kind of approach is descent methods, which extend the traditional descent methods for scalar optimization to solve (UP), such as the gradient descent method [19], Newton's method [10, 20], etc. A remarkable property of this method is that it is a parameter free approach. This is quite different from the weighted approach for (UP). With this method, there is no need to analyze the prior information including the relationships and conflicting between the objectives. Note that such information is very important for choosing the weights for the weighted method. Specifically, an example is shown in [20] that the weighted method fails for a large range of weights which leads to unbounded weighted objective, but a descent method works well with any starting point.

The steepest descent method is one of the oldest and simplest descent methods to minimize the real-valued functions proposed by Baron Augustin-Louis Cauchy (1789–1857) [21] in 1847. Cauchy developed the gradient method to solve nonlinear equations. The idea of this method is that the continuous function value always decreases if the negative descent direction is considered. Almost, all optimization books [26] discuss this method to learn advanced optimization algorithms. However, this method is used to solve system of ridge regression or regularized least squares [22], also see [23–25]. The steepest descent method for single-objective problems was extended for unconstrained multiobjective optimization problems (UP) [19, 27]. Further, the method is presented to find critical points of (UP) using maps from a Euclidean space to a Banach space [28]. Recently, the descent method for (UP) decreases the gradient at the rate of $\frac{1}{\sqrt{k}}$ regardless of the starting point. This rate of convergence is derived for obtaining some point satisfying weak Pareto-optimality [29].

On the other hand, quantum calculus (briefly, q-calculus) is dedicated to the study of calculus regardless of the limits. In fact, by introducing the parameter q, we consider q-analogue of calculus concepts such that they can be recaptured for the case when $q \to 1$. Euler was the founder of this branch of mathematics, by using the parameter q in Newton's work of infinite series. The q-calculus is one of the research interests in the field of mathematics and physics for the last few decades. The q-analogue of ordinary derivative was first proposed by F.H. Jackson [30, 31]. Its wide applications can be seen in several areas such as operator theory [32], mean value theorems of q-calculus [33], q-Taylor formula and its remainder [34, 35], fractional integrals and derivatives [36], integral inequalities [37], variational calculus [38], transform calculus [39], sampling theory [40], etc. Recently, a qFunctions Mathematica package is developed for q-series, and partition theory applications [41].

A q-version of the classical steepest descent method has been developed to solve single objective unconstrained optimization problems [42–45]. In this method, the quantum value q acts as a dilation parameter that controls the balance between global and local search direction in the context of q-gradient. The q-LMS (Least Mean Square) algorithm is proposed which utilizes the q-gradient to compute the secant of the cost function instead of the tangent [46]. The algorithm takes larger steps towards the optimum solution and achieves a higher convergence rate. A new class of stochastic gradient algorithm based on q-calculus is developed to enhance the q-LMS algorithm. The proposed approach utilizes a parameterless concept of error correlation energy, and normalization of signal for high convergence, stability, and low steady-state error. The q-Taylor formula for the functions of several variables and mean value theorems in q-calculus are utilized for solving the systems of equations [48]. In multiobjective optimization, Newton's method using q-calculus [49] is extended for unconstrained multiobjective optimization problems [50]. Following this trend, the Quasi-Newton's method [51] is further extended for solving (UP) [52] where components of the Hessian matrix are computed using q-calculus.

The goal of this paper is to present the q-steepest descent method, a generalization of the steepest descent method for (UP) [10, 27]. A subproblem is formulated using the quadratic approximation of all functions at every iteration, and a feasible descent direction due to q-gradient is obtained as a solution of this subproblem [53]. The ordering of the objective functions is avoided. Instead of the classical gradient, the q-gradient of the objective function is used in the proposed algorithm to find critical points and (weak) efficient points and its theoretical results are validated by giving convergence proofs. The advantage of using q-gradient is that it allows the steepest descent direction to perform in a more diverse set of directions which makes it possible to escape local critical points. Several examples are presented where q-analog is implemented to shift the search process from global in the beginning to almost local search in the end. To evaluate the performance of the proposed method, we compare it with [28] and the weighted sum method. In general, the involvement of q-calculus effectively reduces the number of iterations to reach the critical point or (weak) efficient points. Thus, the proposed method is more promising to solve convex problems, non-convex problems, and multimodal multiobjective optimization problems.

The paper is arranged as follows: Section 2 recalls concepts related to q-calculus and multiobjective optimization. In Section 3, the subproblem for the descent direction using $q-$derivative is presented. The proposed algorithm and convergence analysis with numerical examples are provided in Section 4. Comparison with the existing method is done in Section 5. The conclusion is given in the last section.

2. Preliminaries

Denote $\mathbb{R}$ as the set of real numbers, $\mathbb{R}^n_+ := \{x \in \mathbb{R}^n | x_i \geq 0, \ \forall \ i = 1, \ldots, m\}$, and as $\mathbb{R}^n_{++} := \{x \in \mathbb{R}^n | x_i > 0, \ \forall \ i = 1, \ldots, m\}$. A continuous function on any interval not containing 0, is called the continuous q-differentiable function. Let the q-integer $[n]$ be defined as $[n] = \frac{1-q^n}{1-q}$, for $n \in N$, and the derivative of x^n with respect to x be given as $[n]x^{n-1}$, then the q-derivative [30] is given as:

$$D_q f(x) = \begin{cases} \frac{f(qx)-f(x)}{qx-x}, & x \neq 0, \ q \neq 1, \\ \frac{df(x)}{dx}, & \text{otherwise.} \end{cases} \tag{2.1}$$

The q-derivative reduces to an ordinary derivative as q equals 1. The first-order partial q-derivative [42] of function f with respect to the variable x_i, where $i = 1, \ldots, n$ is:

$$D_{q_i,x_i} f(x) = \begin{cases} \frac{f(x_1,\ldots,q_i x_i,\ldots,x_n) - f(x_1,\ldots,x_i,\ldots,x_n)}{q_i x_i - x_i}, & x_i \neq 0, q_i \neq 1, \\ \frac{\partial f(x)}{\partial x_i}, & \text{otherwise.} \end{cases} \tag{2.2}$$

The q-gradient is the vector of n first-order partial q-derivatives of f is:

$$\nabla_q f(x)^T = \begin{bmatrix} D_{q_1,x_1} f(x) & \ldots & D_{q_i,x_i} f(x) & \ldots & D_{q_n,x_n f(x)} \end{bmatrix}, \tag{2.3}$$

where the parameter q is a vector $q = (q_1, \ldots, q_i, \ldots, q_n) \in \mathbb{R}^n$. Consider the following unconstrained multiobjective optimization problem (UP):

$$\text{minimize} \quad (f_1(x), \ldots f_m(x)), \qquad x \in \mathbb{R}^n, \tag{2.4}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$, $\forall\ i = 1, \ldots, m$ are real-valued continuously q-differentiable function. For two vectors $x, y \in \mathbb{R}^n$, we denote $x = y$ if and only if $x_i = y_i\ \forall\ i = 1, \ldots, m$, $x \geqq y$ if and only if $x_i \geq y_i$ $\forall\ i = 1, \ldots, m$, $x \geq y$ if and only if $x_i \geq y_i$ and $x \neq y$, $\forall\ i = 1, \ldots, m$, $x > y$ if and only if $x_i > y_i$ $\forall\ i = 1, \ldots, m$. A feasible solution $x^* \in X \subseteq \mathbb{R}^n$ is called a locally (weakly) efficient solution [20] if there is a neighborhood $W \subseteq X$ of x^* is an (weakly) efficient solution, then $f_i(x^*)$ is called a locally (weakly) non-dominated point. Let $X \subset \mathbb{R}^n$ be a convex set, and $f_i : X \to \mathbb{R}^m$ is called $\mathbb{R}^m$-convex if f_i, $\forall\ i = 1, \ldots, m$ is componentwise-convex [17]. If the domain of the multiobjective optimization problem is a convex set and the objective functions are convex component-wise then critical point will be the weak efficient point, and if the objective functions are strictly convex component-wise then critical point will be efficient point. The relationship between critical point and efficient point is discussed in [20]. The classical gradient is obtained in the limit of $q_i \to 1$, for all $i = 1, \ldots, m$. The following example is from [49].

Example 1. *Consider a function $f : \mathbb{R}^2 \to \mathbb{R}$, where $f(x, y) = xy^2 + x^4$. Then, the q-gradient [52] is computed as:*

$$\nabla_q f(x)^T = \begin{bmatrix} y^2 + (1+q)(1+q^2)x^3 & (1+q)xy \end{bmatrix}.$$

3. The q-steepest descent for multiobjective optimization

The transpose of Jacobian matrix of q-partial derivative denoted as $J_q F(x)$ for the vector valued objective functions F is:

$$J_q F(x) = \begin{bmatrix} D_{q_1,x_1} f_1(x) & D_{q_2,x_2} f_1(x) & \ldots & D_{q_n,x_n} f_1(x) \\ D_{q_1,x_1} f_2(x) & D_{q_2,x_2} f_2(x) & \ldots & D_{q_n,x_n} f_2(x) \\ \ldots & \ldots & \ldots & \\ D_{q_1,x_1} f_m(x) & D_{q_2,x_2} f_m(x) & \ldots & D_{q_n,x_n} f_m(x) \end{bmatrix}. \tag{3.1}$$

Note that a variant $q \in (0, 1)$ has been considered in the above matrix. The range of the linear mapping given by the Jacobian matrix $J_q F(x)$ is range$(J_q F(x))$. A necessary condition for a (local) weak Pareto minimizer $x \in \mathbb{R}^n$ is given as:

$$\text{range}(J_q F(x)^T) \cap (-\mathbb{R}_{++})^m = \phi. \tag{3.2}$$

Using Gordan's theorem of alternative one can show that x is a critical point if there exists $\lambda \in \mathbb{R}^m_+$, $\lambda \neq 0$ such that

$$\sum_{i=0}^{m} \lambda_i \nabla_q f_i(x) = 0 \tag{3.3}$$

holds (see Lemma 3.1 of [55]). In the literature (3.2) or (3.3) is often considered as a necessary condition of weak efficiency. The points satisfying (3.2) are called Pareto critical points. If a point $x \in \mathbb{R}^n$ is not Pareto critical then there exists a descent direction $d_q \in \mathbb{R}^n$ for F satisfying

$$\nabla_q F(x) d_q \in (-\mathbb{R}_{++})^m. \tag{3.4}$$

If $d_q \in \mathbb{R}^n$ is not a descent direction, then

$$\nabla_q F(x) d_q \notin (-\mathbb{R}_{++})^m. \tag{3.5}$$

The descent gradient method for (UP) takes the schema $x^{k+1} = x^k + \alpha_k d_q^k$, where $\alpha_k \in (0, 1]$ is the step-length computed by Armijo-Wolfe line search with backtracking. For single-objective case, the negative q-gradient $-\nabla_q f(x)$ is the steepest descent direction, and it is the solution of quadratic programming whose objective function is an approximation to $f(x + d_q) - f(x)$, that is,

$$\min_{d_q \in \mathbb{R}^n} \nabla_q f_i(x)^T d_q + \frac{1}{(1+q)} \|d_q\|^2.$$

For (UP) case, the subproblem introduced in [27], is an approximation to the least reduction of all objective functions, which is given as:

$$\min_{d_q \in \mathbb{R}^n} \max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q + \frac{1}{1+q} \|d_q\|^2, \tag{3.6}$$

where $q \in (0, 1)$, and the search direction d_q^k is the solution of (3.6). The min-max subproblem (3.6) is depicted in the form of q-differentiable quadratic optimization problem (q-DQOP) with linear inequality constraints as:

$$\begin{aligned} \arg \min_{d_q \in \mathbb{R}^n, t \in \mathbb{R}} \quad & t + \frac{1}{(1+q)} \|d_q\|^2, \quad q \in (0, 1), \\ \text{subject to} \quad & \nabla_q f_i(x^k)^T d_q \leq t, \quad i = 1, \ldots, m. \end{aligned}$$

Note that the term $\frac{1}{(1+q)} \|d_q\|^2$ for $q \in (0, 1)$ in the objective function ensures that the problem is bounded, and (d_q^k, t^k) are the optimial solution of the above problem (q-DQOP). The above sub-problem is a convex programming problem and satisfies Slater's constraint qualification since for $t = 1$, and $d_q = 0$, the above inequalities become strict inequalities. The Karush-Kuhn-Tucker (KKT) conditions [54] for (q-DQOP) at x^k are:

$$d_q^k = -\sum_{i=1}^{m} \lambda_i^k \nabla_q f_i(x^k), \tag{3.7}$$

$$\nabla_q f_i(x^k)^T d_q^k \leq t^k, \quad i = 1, \ldots, m, \tag{3.8}$$

$$\sum_{i=1}^{m} \lambda_i^k = 1, \quad \lambda_i^k \geq 0, \tag{3.9}$$

$$\lambda_i^k (\nabla_q f_i(x^k)^T d_q - t^k) = 0, \quad i = 1, \ldots, m, \tag{3.10}$$

Note that $\lambda_i^k \geq 0$, where $i = 1, \ldots m$, are the Lagrange multipliers associated with linear inequality constraints of (3.7). Taking summation of (3.10) over i from 1 to m, and from (3.7), we get

$$t^k = \sum_{i=1}^{m} \lambda_i^k \nabla_q f_i(x^k)^T d_q^k = -\|d_q^k\|^2. \tag{3.11}$$

The solution of the (q-DQOP) is related to Pareto critical as stated in [27].

Lemma 1. *[29] Let (d_q^k, t^k) be the solution of (q-DQOP)*

1. *If x^k is Pareto critical, then $d_q^k = 0 \in \mathbb{R}^n$, and $t^k = 0$.*
2. *If x^k is not Pareto critical, then*

$$t^k \leq -\frac{1}{(1+q)} \|d_q^k\|^2 < 0; \quad q \in (0, 1), \tag{3.12}$$

$$\nabla_q f_i(x^k)^T d_q^k \leq t^k, \quad i = 1, \ldots, m. \tag{3.13}$$

Proof. If x^k is Pareto critical, then there is no d_q^k such that $\nabla_q f_i(x^k)^T d_q^k < 0$, $\forall\ i = 1, \ldots, m$, otherwise Part 2 would not be satisfied, and for any fix $\bar{i}$, we have $t^k \geq \nabla_q f_{\bar{i}}(x^k)^T d_q^k$. Part 1 of this lemma holds when $(t^k, d_q^k) = (0, 0)$, is a feasible point of (q-DQOP). The constraint of (q-DQOP) is equivalent to (3.13). Since $d_q = 0$, and $t = 0$, then we have $(d_q^k, t^t) \leq (0, 0)$, thus (3.12) holds. If x^k is not a Pareto critical, then there exists a descent search direction d_q^k such that $\nabla f_i(x^k)^T d^k < 0$, $\forall\ i = 1, \ldots, m$ to get $t^k < 0$. □

Suppose (t^k, d_q^k) is the solution of the sub-problem and $d_q^k \neq 0$, following the arguments in [50, 55] we select a suitable step length. Suppose θ_i^k is the angle between ∇f_i^k and d_q^k. If $\cos^2(\theta_i^k) \geq \delta$, where $\delta > 0$ holds for all i then we select α_k satisfying

$$f_i(x^{k+1}) - f_i(x^k) \leq \beta_1 \alpha_k \sum_{i=1}^{m} \lambda_i^k (\nabla_q f_i(x^k))^T d_q(x^k), \tag{3.14}$$

and

$$\beta_2 \sum_{i=1}^{m} \lambda_i^k (\nabla_q f_i(x^k))^T d_q(x^k) \leq \sum_{i=1}^{m} \lambda_i^k \nabla_q f_i(x^{k+1}) d_q(x^k), \tag{3.15}$$

where $\beta \in (0, 1)$, $\beta_1 < \beta_2 < 1$, and $\alpha \in (0, 1]$. If $cos^2(\theta_j^k) < \delta$ for any j then α_k satisfying (3.14) is selected. The last inequality is a criterion for accepting a step-length in the multiobjective descent direction. We start with $\alpha = 1$, and if the above condition is not satisfied, then set $\alpha := \frac{\alpha}{2}$, and process will be continued.

Proposition 1. *Let $\beta_1 \in (0,1)$, $x^k \in X \subseteq \mathbb{R}^n$, and $d_q^k \in X$. If $f_i : \mathbb{R}^n \to \mathbb{R}$ for $i = 1,\ldots,m$ are continuously q-differentiable functions, and $\nabla_q f_i(x^k) d_q^k < 0$, for $i = 1,\ldots,m$ then there exists $\varepsilon > 0$ such that $f_i(x^k + \alpha d^k) \leq f_i(x^k) + \beta_1 \alpha \nabla_q f_i(x^k) d_q^k$ for all $i = 1,\ldots,m$ and $\alpha \in (0,\varepsilon]$.*

Proof. Since f_i for all $i \in 1,\ldots,m$, are q-differentiable functions, then

$$f_i(x^k + \alpha d_q^k) = f_i(x^k) + \alpha(\nabla_q f_i(x^k) d_q^k + R(t)), \tag{3.16}$$

where $\lim_{t\to 0} R(t) = 0$. For $\beta_1 \in (0,1)$, we have

$$(\nabla_q f_i(x^k))^T d_q < \beta_1 (\nabla_q f_i(x^k))^T d_q, \tag{3.17}$$

where $i = 1,\ldots,m$. Combining (3.16) and (3.17), we get

$$f_i(x^k + \alpha d_q^k) \leq f_i(x^k) + \beta_1 \alpha (\nabla_q f_i(x^k) d_q^k, \tag{3.18}$$

for all $\alpha \in (0,\varepsilon]$, $i = 1,\ldots,m$. □

4. Convergence analysis and numerical examples

Let us now summarize the steepest descent method for multiobjective proposed in [19] based on q-derivative which is given in Algorithm 1 as:

Algorithm 1 q-Steepest Descent for Unconstrained Multiobjective Optimization (q-SDUP)

Result: Critical Point or (Weak) Efficient Points

Data: Choose $\beta_1 \in (0,1)$, $x^0 \in \mathbb{R}^n$, error of tolerance $\varepsilon_1, \varepsilon_2$, small positive number $\delta > 0, q \in (0,1)$

```
for k=1,2,... do
    Solve (q-DQOP) and compute (d_q^k, t^k, λ^k)
    for i=1,...,m do
        Choose step-length α_k ∈ (0,1] such that x^k + α_k d_q^k ∈ X
        if cos^2(θ_j^k) >= δ then
            choose α_k satisfying (3.14) and (3.15).
        else
            Choose α_k satisfying (3.14).
        end
    end
    Update x^{k+1} = x^k + α_k d_q^k
    if |t(x^{k+1})| ≤ ε_1 or ||d_q(x^{k+1})|| ≤ ε_2 then
        Stop
    end
end
```

Algorithm 1 ends up with a critical point or produces (weak) efficient points. In this case, if $\cos^2(\theta_i^k) > \delta$, where $i = 1,\ldots,m$ is assumed, then assumptions of Theorem 1 are satisfied and every accumulation point of the sequence $\{x^k\}$ will be a (weak) efficient point otherwise assumptions of Theorem 2 are considered and the accumulation point becomes a critical point of (UP).

Theorem 1. *Let f_i where $i = 1, \ldots, m$ be continuously q-differentiable on a set $X \subseteq \mathbb{R}^n$ for $\beta_1, q \in (0, 1)$ and $\{x^k\}$ be the sequence updated by $x^{k+1} = x^k + \alpha_k d_q(x^k)$, where α_k satisfies*

$$f_i(x^{k+1}) - f_i(x^k) \leq \beta_1 \alpha_k \sum_{i=1}^{m} \lambda_i^k (\nabla_q f_i(x^k))^T d_q(x^k), \tag{4.1}$$

for all $i = 1, \ldots, m$. Suppose that $L_0 = \{x \in X : f(x) < f(x^0)\}$ is bounded and convex, where $x^0 \in X$ is a starting point. The function $f_i(x)$ is bounded below for at least one $i \in \{1, \ldots, m\}$. Then, the accumulation point of $\{x^k\}$ is a critical point x^* of (UP).

Proof. We know that

$$f_i(x^{k+1}) - f_i(x^k) \leq \beta_1 \alpha_k \sum_{i=1}^{m} \lambda_j^k (\nabla_q f_i(x^k))^T d_q(x^k).$$

Since $\sum_{i=1}^{m} \lambda_i^k = 1$, where $\lambda_i^k \geq 0$, then

$$f_i(x^{k+1}) - f_i(x^k) \leq \beta_1 \alpha_k \max_{i=1,\ldots m} ((\nabla_q f_i(x^k))^T d_q(x^k)),$$

that is,

$$f_i(x^{k+1}) - f_i(x^k) < \beta_1 \alpha_k \max_{i=1,\ldots m} \left[(\nabla_q f_i(x^k))^T d_q(x^k) + \frac{1}{(1+q)} \|d_q\|^2 \right] = \beta_1 \alpha_k d_q^k.$$

We obtain

$$f_i(x^{k+1}) < f_i(x^0) + c \sum_{j=0}^{k} \alpha_i d_q(x^i) \text{ for all } i = 1, \ldots, m.$$

For at least one i_1 from $i = 1, \ldots, m$ for which $f_i(x)$ is bounded below such that $f_{i1}(x) > -\infty$ for all $x \in X$. The sequence $\{f_{i1}(x^k)\}$ is also monotonically decreasing sequence and bounded below so that $f_{i1}(x^k)$ converges to $f_{i1}(x^*)$ as $k \to \infty$ where $f_{i1}(x^*) > -\infty$. We obtain the inequality as

$$f_{i1}(x^0) - f_{i1}(x^{k+1}) > -\beta_1 \sum_{j=0}^{k} \alpha_j d_q(x^j).$$

As $k \to \infty$, we have

$$\beta_1 \sum_{j=0}^{\infty} \alpha_j (-d_q(x^j)) < f_{i1}(x^0) - f_{i1}(x^*) < \infty \tag{4.2}$$

Since $d_q(x^j) \leq 0$ for all j due to [10, 52], and $\beta_1 \sum_{j=0}^{\infty} \alpha_j (-d_q(x^j))$ is finite. Thus, we obtain $\beta_1 \alpha_k (-d_q(x^k)) \to 0$ as $k \to \infty$. Since the step length is bounded above so $\alpha_k \to \infty$ for some k implies L_0 unbounded which is contradiction to the assumption. If $\alpha_k \geq \beta_1$ for all k and for some $\beta_1 \in (0, 1)$, then we get $-d_q(x^k) \to 0$ as $k \to \infty$. Note that L_0 is bounded sequence, and has at least one accumulation point. Let $\{R_1^*, R_2^*, \ldots, R_r^*\}$ be the set of accumulation points $\{x^k\}$. Since R_s^* is an accumulation point for every $s \in \{1, 2, \ldots, r\}$, and d_q is a continuous function, then $d_q(R_s^*)$ is a critical point of f_s for every $s \in \{1, 2, \ldots, r\}$. □

Theorem 2. *Let f_i for all $i = 1, \ldots, m$ be a continuously q-differentiable on a set $X \subset \mathbb{R}^n$, and $\{x^k\}$ be the sequence by $x^{k+1} = x^k + \alpha_k d_q^k(x^k)$, and given that*

1.

$$\beta_2 \sum_{i=1}^{m} \lambda_i^k (\nabla_q f_i(x^k))^T d_q(x^k) \leq \sum_{i=1}^{m} \lambda_i^k \nabla_q f_i(x^{k+1}) d_q(x^k), \text{ where } \beta_1 < \beta_2 < 1, \tag{4.3}$$

2. $\|\nabla_q f_i(x^k) - \nabla_q f_i(x^{k+1})\| \leq L_i \|x^k - x^{k+1}\|, \ \forall \ x^k, x^{k+1} \in \mathbb{R}^n$, $i = 1, \ldots, m$, *and*

3. $\|\cos^2 \theta_i^k\| \geq \delta$ *for some* $\delta > 0$, *for all* $i = 1, \ldots, m$, *where* θ_i^k *is the angle between* $d_q(x^k)$ *and* $\nabla_q f_i(x^k)$.

Then, every accumulation point of $\{x^k\}$ *generated is a weak efficient solution of (UP).*

Proof. We have proved that every accumulation point of $\{x^k\}$ is a critical point of f_i, for all $i = 1, \ldots, m$. Let x^* be an accumulation point of $\{x^k\}$. Fix one i_0 from $i = 1, \ldots, m$ for which $\nabla_q f_{i_0}(x^*) = 0$, then x^* will be a (weak) efficient solution. Form Cauchy-Schwartz inequality for all $i = 1 \ldots, m$, we obtain

$$\begin{aligned}(\nabla_q f_i(x^{k+1}) - \nabla_q f_i(x^k))^T d_q(x^k) &\leq \|\nabla_q f_i(x^{k+1}) - \nabla_q f_i(x^k)\| \|d_q(x^k)\| \\ &\leq L_i \|x^{k+1} - x^k\| \|d_q(x^k)\| \\ &\leq L_i \alpha_k \|d_q(x^k)\|^2,\end{aligned}$$

Since $L = \max L_i$, where $i = 1, 2 \ldots, m$, then

$$(\nabla_q f_i(x^{k+1}) - \nabla_q f_i(x^k))^T d_q(x^k) \leq L\alpha_k \|d_q(x^k)\|^2.$$

We obtain

$$\begin{aligned}L\alpha_k \|d_q(x^k)\|^2 &\geq \max_{i=1,\ldots m} (\nabla_q f_i(x^{k+1}) - \nabla_q f_i(x^k))^T d_q(x^k) \\ &\geq \sum_{i=1}^{m} \lambda_i^k (\nabla_q f_i(x^{k+1}) - \nabla_q f_i(x^k))^T d_q(x^k).\end{aligned}$$

From part 1 of this theorem,

$$\begin{aligned}L\alpha_k \|d_q(x^k)\|^2 &\geq (\beta_2 - 1) \sum_{i=1}^{m} \lambda_i^k \nabla_q f_i(x^k)^T d_q(x^k) \\ &\geq (\beta_2 - 1) \max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q(x^k),\end{aligned}$$

this is,

$$\alpha_k \geq \frac{\beta_2 - 1}{L\|d_q(x^k)\|^2} \max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q(x^k).$$

Since $\max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q(x^k) < 0$, then

$$\alpha_k \Big(\max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q(x^k) \Big) \leq \frac{\beta_2 - 1}{L\|d_q(x^k)\|^2} \Big(\max_{i=1,\ldots,m} \nabla_q f_i(x^k)^T d_q(x^k) \Big)^2,$$

that is,

$$-\beta_1\alpha_k\Big(\max_{i=1,\dots,m}\nabla_q f_i(x^k)^T d_q(x^k)\Big) \geq \frac{\beta_1(\beta_2-1)}{L\|d_q(x^k)\|^2}\min_{i=1,\dots,m}\Big(\nabla_q f_i(x^k)^T d_q(x^k)\Big)^2.$$

Since $(\nabla_q f_i(x^k)^T d_q(x^k))^2 = (\nabla_q f_i(x^k)^T)^2(d_q(x^k))^2(\cos^2\theta_i^k)$, for all $i = 1,\dots,m$ then

$$-\beta_1\alpha_k\Big(\max_{i=1,\dots,m}\nabla_q f_i(x^k)^T d_q(x^k)\Big) \geq \frac{\beta_1(1-\beta_2)}{L}\min_{i=1,\dots,m}[\|\nabla_q f_i(x^k)\|^2\cos^2(\theta_i)^T],$$

where θ_i^k is the angle between $\nabla_q f_i(x^k)$ and $d_q(x^k)$. We have

$$\begin{aligned}\infty > f_{i1}(x^0) - f_{i1}(x^{k+1}) &\geq -\beta_1\sum_{j=0}^{k}\alpha_j\max_{y\in Y,i=1,\dots,m}(\nabla_q f_i(x^i))^T d_q(x^i)\\ &= \sum_{j=0}^{k}\alpha_j(-\beta_1\max_{y\in Y,i=1,\dots,m}(\nabla_q f_i(x^i))^T d_q(x^i)).\end{aligned}$$

Taking $k\to\infty$,

$$\infty > f_{i1}(x^0) - f_{i1}(x^*) \geq \sum_{j=0}^{\infty}\alpha_j(-\beta_1\max_{i=1,\dots,m}(\nabla_q f_i(x^i))^T d_q(x^i)).$$

Since $-\beta_1\max_{i=1,\dots,m}\nabla_q f_i(x^i)^T d_q(x^i) > 0$, then

$$\alpha_k(-\beta_1\max_{i=1,\dots,m}(\nabla_q f_i(x^k))^T d_q(x^k)) \to 0 \text{ as } k\to\infty.$$

We also have

$$\frac{\beta_1(1-\beta_2)}{L}\min_{i=1,\dots,m}[\|\nabla_q f_i(x^k)\|^2\cos^2\theta_i^k] \to 0 \quad \text{as} \quad k\to\infty.$$

Since $\cos^2\theta_i^k > \delta$ for $i = 1,\dots,m$, then $\min_{i=1,\dots,m}\|\nabla_q f_i(x^k)\|^2 \to 0$ as $k\to\infty$. Fix any i_0 from $i = 1,\dots,m$ such that $\|\nabla_q f_{i0}(x^k)\|^2 \to 0$ as $k\to\infty$. Since $\|\nabla_q f_{i0}(x^k)|$ is a continuous function, and $\|\nabla_q f_{i0}(x^k)\| \to 0$ as $k\to\infty$, then $\nabla_q f_{i0}(x^*) = 0$ for every accumulation points x^* of $\{x^k\}$. Thus, x^* is a local weak efficient solution. □

Iteration Complexity of Algorithm 1: Based on the ideas above, we consider iteration complexity. For a given iteration k, a point solution for a particular x^k, reaches to an optimal point under a desired tolerance ϵ. The steepest descent direction moves from d_q^k to some d_q^{k+1} where $d_q^{k+1} - d_q^k$ is small enough to be able to determine the associated solution on the Pareto front. For non-convex, Algorithm 1 has a convergence rate of the order of $\frac{1}{\sqrt{k}}$, and for the convex case, Algorithm 1 establishes the desired $\frac{1}{k}$ rate for a certain sequence of weights $\{\bar{\lambda}^k\}$ (see Theorem 4.1 of [29]) since for large value of k, q-gradient behaves as a classical gradient [43]. The global rates translate into worst-case complexity bounds of the order of $\frac{1}{\epsilon^2}$, $\frac{1}{\epsilon}$, and $\log\frac{1}{\epsilon}$ iterations, respectively, to reach an approximate optimality criterion of the form $\|d_q^k\| \leq \epsilon$ for some $\epsilon\in(0,1)$ where d_q^k is the steepest descent direction (3.6). Due to the existence

of a uniform lower bound on the step-length α_k in Algorithm 1 will always stop in a finite number of steps.

It is important to note that (weak) efficient solution of (UP) is not unique. Therefore, one can execute Algorithm 1 with any starting point to reach at one of these (weak) efficient points. User may consider a sufficiently large compact subset of $\mathbb{R}^n$ as a domain of (UP) to solve. We have implemented algorithm 1 with the Armijo-Wolfe line search with backtracking. We have considered both convex and nonconvex problems. To avoid unbounded solutions of the subproblem at any iterating point x^k, we consider the following subproblem:

$$\begin{aligned} \min_{t,d_q} \quad & t + \frac{1}{2} d_q^T d_q \\ \text{subject to} \quad & \nabla_q f_i(x^k)^T d_q \leq t \quad i = 1, 2, ..., m, \\ & lb \leq x^k + d_q \leq ub, \end{aligned} \tag{4.4}$$

where lb and ub are lower and upper bound of x^k, respectively. We now illustrate the methodology in numerical examples.

Example 2 (One Dimension [55])**.** *Consider the problem (UP): Minimize*$_{x\in\mathbb{R}}$ $(f_1(x), f_2(x))$*, where* $f_1(x) = x^2 - 4$ *and* $f_2(x) = (x-1)^2$.

This problem has two objective functions and one decision variable. Both functions are convex. The set of efficient solutions to this problem is [0, 1] which can be shown in Figure 1. Using Algorithm 1 with any starting point $x^0 > 1$, one can find the efficient point approximately equal to 1. At starting point $x^0 = 10$, the approximate solution is obtained at 29th step as $x^{29} = 1.00240356 \approx 1$ in [55]. Under the same starting point and stopping criterion, i.e, $\epsilon_1 = 10^{-7}, \epsilon_2 = 10^{-11}$, and $\delta = 10^{-4}$, our algorithm provides the efficient solution at 2nd step. The Pareto front using Algorithm 1 and [28] for $-2 \leq x \leq 2$ is found as a convex curve shown in Figure 2.

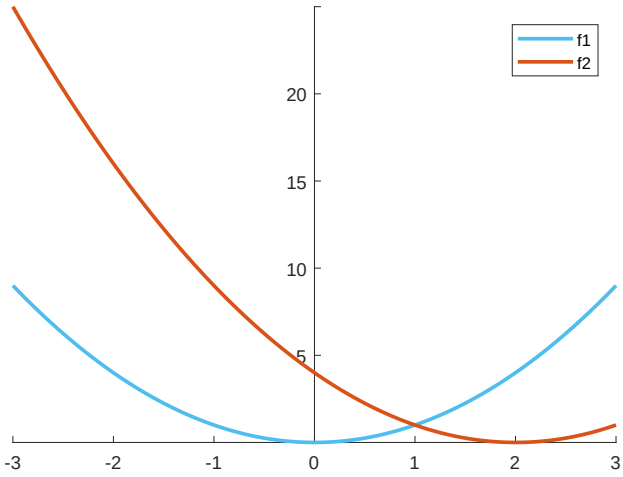
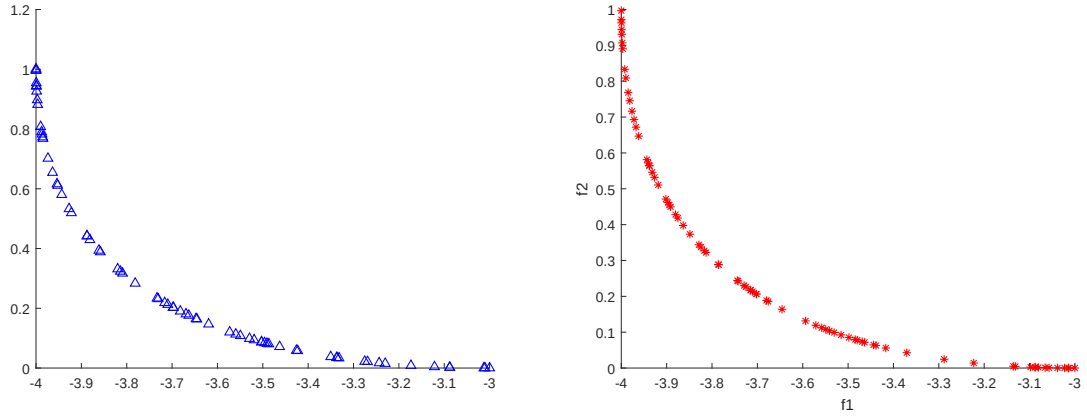


Figure 1. Graph of Example 2.



(a) (b)

Figure 2. (a) Approximate Pareto Front without q-derivative for Example 2. (b) Approximate Pareto Front with q-derivative for Example 2.

Example 3 (Two Dimension)**.** *Consider an unconstrained multiobjective optimization problem [55] minimize*$_{x\in\mathbb{R}^2}$ $(f_1(x), f_2(x))$*, where* $f_1(x_1, x_2) = \frac{1}{4}[(x_1-1)^4 + 2(x_2-2)^4]$, $f_2(x_1, x_2) = (x_2 - x_1^2)^2 + (1-x_1)^2$,

After running the Algorithm 1, and methodology used in [28], we obtain the point $x^* = (1, 2)^T$ and $\lambda_1^* = 0.9774$, $\lambda_2^* = 0.226$, in two iterations as final solution with a starting point $x^0 = (1, 3)^T$. One can verify that

$$\lambda_1^* \nabla_q f_1(x^*) + \lambda_2^* \nabla_q f_2(x^*) \approx \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

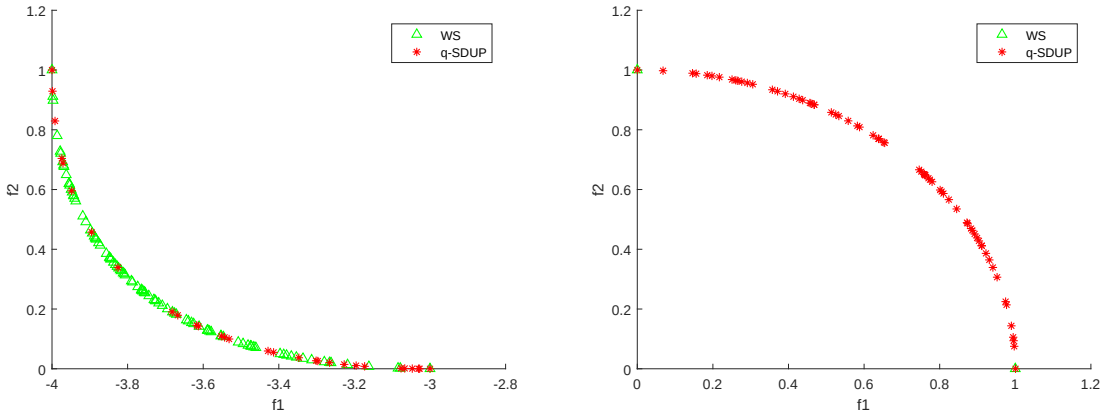
Note that function f_2 is not convex. A multiobjective optimization does not produce a unique solution but a set of efficient solutions. The algorithm does not depend upon the choice of the starting point. The sequence converges to one of the sets of critical or weak efficient solutions, starting at any starting point. For example, with starting point $x^0 = (0, 0)^T$, Algorithm 1 converges to $x^* = (0.9975, 0.9936)^T$ in 60 iterations, while methodology used in [28] converges to $x^* = (0.9981, 0.9950)$ in 64 iterations.

Example 4 (Two Dimension Non-Convex Problem)**.** *Consider the test problem (DTLZP) from [56]:* $\min\limits_{0\le x_1, x_2\le 1}$ $(f_1(x_1, x_2), f_2(x_1, x_2))$*, where* $f_1(x_1, x_2) = (1+gx)\cos(0.5\pi x_1)$, $f_2(x_1, x_2) = (1+gx)\sin(0.5\pi x_1)$, $gx = (x_2 - 0.5)^2$.

Consider a starting point $x^0 = (0.5060, 0.6991)^T$. Then, we have $f(x^0) = (0.7282, 0.7420)^T$. Note that both f_1 and f_2 are non-convex. At x^0, the q-gradients are $|\nabla_q f_1(x^0)| = -2.7161 < 0$ and $|\nabla_q f_2(x^0)| = -2.8055 < 0$, respectively. Solution of subproblem at x^0 is $t^0 = -0.0793$, and steepest descent direction is $d_q^0 = (0.0006, -0.2816)^T$. Because $\cos^2(\theta_1) = 0.0552 > 10^{-4}$ and $\cos^2(\theta_2) = 0.0571 > 10^{-4}$, thus step length $\alpha_0 = 1$ is selected to satisfying both (3.14) and (3.15). Next iterating point is $x^1 = x^0 + \alpha_0 d_q^0 = (0.5066, 0.4175)$. Clearly, we get $f(x^1) = (0.7044, 0.7193)^T < f(x^0)$. Final solution is obtained at 7^{th} step as $x^7 = (0.5068, 0.5010)^T$. One can verify that x^7 is an approximate critical point since

$$\|0.4957\nabla_q f_1(x^7) + 0.5053\nabla_q f_2(x^7)\| = 1.4 \times 10^{-3} \approx 0.$$

Comparison of approximate Pareto front with weighted sum method: To obtain an approximate Pareto front we have considered a multi-start technique. Here, 100 uniformly distributed random points are selected to execute Algorithm 1 and weighted sum (WS) algorithm individually. Approximate Pareto front obtained by Algorithm 1 is compared with the approximate Pareto front obtained by the weighted sum method. In (WS) method we have used weights $(1, 0)$, $(0, 1)$, and 98 random weights. The single objective optimization problem is solved by a single objective steepest descent method with some random initial approximation. Approximate Pareto fronts generated by Algorithm 1 for Example 2 and Example 4 are given in Figure 3(a), (b), respectively. One can observe that Algorithm 1 generates an approximate Pareto front for both problems but (WS) fails for Example 4 (non-convex problem).



(a) **(b)**

Figure 3. (a) Approximate Pareto Front for Example 2. (b) Approximate Pareto front for Example 4.

Table 1 lists the average of number of iterations *it*, that of function evaluations $evalf$, that of gradient evaluations *evalg* and that of subproblem solved *evalsp*, respectively to show the performance of q-steepest descent method for (UP). The reported data in Table 1 represent a typical execution of algorithm 1. We conclude that Algorithm 1 performs better in terms of the number of iterations than the method of [28]. It is noteworthy that the number of steepest descent direction calculations is equal to the number of iterations.

Table 1. Performance of q-Steepest Descent Method.

Problem	(q-SDUP)				[28]			
	it	$evalf$	*evalg*	*evalsp*	*it*	$evalf$	*evalg*	*evalsp*
Example 2	171	238	238	1623	176	250	176	1929
Example 4	684	777	775	4561	705	715	714	4903

Example 5 (Three Dimension)**.** *Consider multiobjective problem: Minimize* $(f_1(x), f_2(x), f_3(x))$*, where* $f_1(x_1, x_2, x_3) = x_1^2 + x_2^2 + x_3^2$, $f_2(x_1, x_2, x_3) = (x_1 - 2)^2 + x_2^2 + x_3^2$, $f_3(x_1, x_2, x_3) = x_1^2 + x_2^2 + (x_3 - 2)^2$.

We solve the above problem with the method described in [28] and compare it with the proposed method of this paper. Consider a starting point $x^0 = (2, 1, 3)^T$, and stopping criteria $\min\{\nabla_q f_1, \nabla_q f_2, \nabla_q f_3\} < 10^{-5}$. A program in MATLAB (2017a) is written in the lines of Algorithm 1. Of course, the search process moves from global at the beginning to particularly neighborhood at last due to the advantage of q-derivative shown in Table 2. The solution to this problem is found at 14th iterations, and this point is a local weak efficient solution of the above problem.

Table 2. Computational Details of Example 5.

k	x^k	α_k	d_q^k	x^{k+1}	$[\lambda_1^{k+1}, \lambda_2^{k+1}, \lambda_3^{k+1}]$	$\min\cos(\theta_k^i)$
0	$\begin{pmatrix}2\\1\\3\end{pmatrix}$	0.5	$\begin{pmatrix}-34.6980\\-4.3275\\-55.3018\end{pmatrix}$	$\begin{pmatrix}1.6510\\7.8363\\0.3491\end{pmatrix}$	[0.0000, 0.8250, 0.1750]	1.9734
1	$\begin{pmatrix}1.6510\\7.8363\\0.3491\end{pmatrix}$	0.5	$\begin{pmatrix}0.6983\\-0.9999\\-0.6971\end{pmatrix}$	$\begin{pmatrix}2.3493\\6.8364\\-0.3480\end{pmatrix}$	[0.0000, 1.0000, 0.0000]	1.9738
2	$\begin{pmatrix}2.3493\\6.8364\\-0.3480\end{pmatrix}$	0.5	$\begin{pmatrix}-0.6987\\-1.0000\\0.6969\end{pmatrix}$	$\begin{pmatrix}1.6506\\5.8364\\0.3489\end{pmatrix}$	[0.0000 1.0000 0.0000]	1.9751
3	$\begin{pmatrix}1.6506\\5.8364\\0.3489\end{pmatrix}$	0.5	$\begin{pmatrix}0.6990\\-1.0000\\-0.6975\end{pmatrix}$	$\begin{pmatrix}2.3496\\4.8364\\-0.3486\end{pmatrix}$	[0.0000 1.0000 0.0000]	1.9758
4	$\begin{pmatrix}2.3496\\4.8364\\-0.3486\end{pmatrix}$	0.5	$\begin{pmatrix}-0.6989\\-1.0000\\0.6981\end{pmatrix}$	$\begin{pmatrix}1.6507\\3.8364\\0.3495\end{pmatrix}$	[0.0000 1.0000 0.0000]	1.9769
5	$\begin{pmatrix}1.6507\\3.8364\\0.3495\end{pmatrix}$	0.5	$\begin{pmatrix}0.6990\\-0.9999\\-0.6990\end{pmatrix}$	$\begin{pmatrix}2.3496\\2.8365\\-0.3495\end{pmatrix}$	[0.0000 1.0000 0.0000]	1.9774
6	$\begin{pmatrix}2.3496\\2.8365\\-0.3495\end{pmatrix}$	0.5	$\begin{pmatrix}-0.6993\\-1.0000\\0.6989\end{pmatrix}$	$\begin{pmatrix}1.6503\\1.8365\\0.3493\end{pmatrix}$	[0.0000 1.0000 0.0000]	1.7603
7	$\begin{pmatrix}1.6503\\1.8365\\0.3493\end{pmatrix}$	0.5	$\begin{pmatrix}0.4009\\-1.1996\\-0.4007\end{pmatrix}$	$\begin{pmatrix}1.8507\\1.2367\\0.1490\end{pmatrix}$	[0 0.9254 0.0746]	1.1536
8	$\begin{pmatrix}1.8507\\1.2367\\0.1490\end{pmatrix}$	0.5	$\begin{pmatrix}0.1917\\-1.0393\\-0.1915\end{pmatrix}$	$\begin{pmatrix}1.9466\\0.7170\\0.0533\end{pmatrix}$	[0 0.9733 0.0267]	1.0172
9	$\begin{pmatrix}1.9466\\0.7170\\0.0533\end{pmatrix}$	0.5	$\begin{pmatrix}0.0546\\-1.0056\\-0.0539\end{pmatrix}$	$\begin{pmatrix}2.0012\\-0.2886\\-0.0007\end{pmatrix}$	[0 0.9869 0.0131]	0.8673
10	$\begin{pmatrix}2.0012\\-0.2886\\-0.0007\end{pmatrix}$	0.5	$\begin{pmatrix}-0.3413\\-0.8665\\0.0010\end{pmatrix}$	$\begin{pmatrix}1.8305\\-0.7219\\-0.0002\end{pmatrix}$	[0.0843 0.9153 0.0004]	0.579
11	$\begin{pmatrix}1.8305\\-0.7219\\-0.0002\end{pmatrix}$	0.5	$\begin{pmatrix}-0.3678\\-0.5775\\0\end{pmatrix}$	$\begin{pmatrix}1.6542\\-1.0106\\-0.0002\end{pmatrix}$	[0.0865 0.8271 0.0865]	0.1452
12	$\begin{pmatrix}1.6542\\-1.0106\\-0.0002\end{pmatrix}$	0.5	$\begin{pmatrix}-0.2297\\-0.3041\\0\end{pmatrix}$	$\begin{pmatrix}1.4245\\-1.3147\\-0.0002\end{pmatrix}$	[0 0.7697 0.2303]	0.0011
13	$\begin{pmatrix}1.4245\\-1.3147\\-0.0002\end{pmatrix}$	0.5	$\begin{pmatrix}0.0221\\0.0243\\0\end{pmatrix}$	$\begin{pmatrix}1.4466\\-1.2904\\-0.0002\end{pmatrix}$	[0.1411 0.7178 0.1411]	4.5278e-05
14	$\begin{pmatrix}1.4466\\-1.2904\\-0.0002\end{pmatrix}$	0.5	$\begin{pmatrix}-0.0045\\-0.0050\\0\end{pmatrix}$	$\begin{pmatrix}1.4421\\-1.2954\\-0.0002\end{pmatrix}$	[0.2779 0.7221 0]	1.5914e-06
15	$\begin{pmatrix}1.4421\\-1.2954\\-0.0002\end{pmatrix}$	0.5	$\begin{pmatrix}0.8427\times10^{-3}\\0.9387\times10^{-3}\\0\end{pmatrix}$	$\begin{pmatrix}1.4421\\-1.2954\\-0.0002\end{pmatrix}$	[0.2788 0.7212 0.0000]	0.0013

Example 6 (Two Dimension Problem)**.** *Consider multimodal multiobjective optimization problem [57]: minimize* $_{-5\leq x_1,x_2\leq 5}$ $(f_1(x_1,x_2), f_2(x_1,x_2))$*, where* $f_1(x_1,x_2) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2$, $f_2(x_1,x_2) = 100(x_1^2 - x_2)^2 + (x_1 - 2)^2$.

Functions f_1 and f_2 have global minimizers at $x^* = (1,1)^T$ and $\hat{x} = (2,4)^T$, respectively. It is noteworthy the $f_1(x^*) = 0 = f_2(\hat{x}) = 0$ and $f_1(\hat{x}) = f_2(x^*) = 1$. We execute the Algorithm 1 100 times using starting points from a uniform random distribution belonging to $]-5,5[\times]-5,5[$. In all instances, Algorithm 1 stopped at a point satisfying the convergence criterion. We see that Algorithm 1 could estimate the Pareto front as shown in Figure 4.

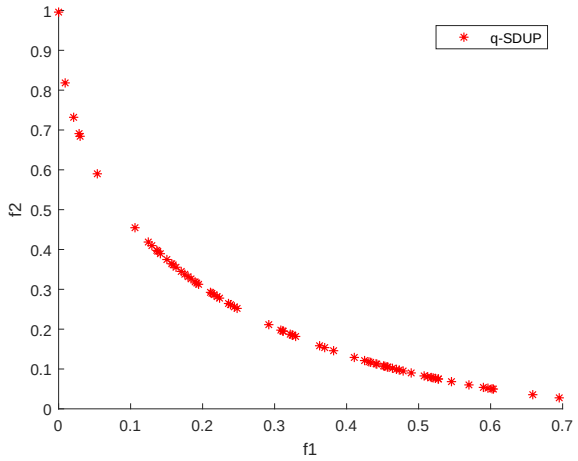


Figure 4. Approximate Pareto Front for Example 6.

5. Comparison with the existing method

We select the work of [28] to compare the proposed method. The basic difference between the method of [28] and our method is that q-gradient based steepest descent algorithm gives faster convergence for $q \in (0,1)$. The q-drivative evaluates tangent, computes the secant of the objective functions, and therefore takes larger steps towards the optimum solution. Table 3 summarizes the results after executing the method of [28] and the proposed algorithm of this paper, with several starting points for the problem of Example 5. Both methods are solved using the MATLAB (2017a) program. It is seen that number of iterations using Algorithm 1 is less than a number of iterations in the methodology by [28].

Table 3. Comparison Results with [28] for Example 5.

Sl. No.	Starting Point	Number of iterations in Algorithm 1	Final result in Algorithm 1	Number of iterations in [28]	Final result in [28]
1	$\begin{pmatrix}1\\5\\6\end{pmatrix}$	10	$\begin{pmatrix}1.4434\\-1.2907\\0.0035\end{pmatrix}$	15	$\begin{pmatrix}1.3652\\-1.0703\\-0.0000\end{pmatrix}$
2	$\begin{pmatrix}3\\4\\1\end{pmatrix}$	10	$\begin{pmatrix}1.4117\\-1.2015\\0.0003\end{pmatrix}$	14	$\begin{pmatrix}1.4472\\-1.3035\\-0.0000\end{pmatrix}$
3	$\begin{pmatrix}5\\7\\3\end{pmatrix}$	13	$\begin{pmatrix}1.4048\\-1.1773\\0.0001\end{pmatrix}$	18	$\begin{pmatrix}1.3975\\-1.1569\\-0.0000\end{pmatrix}$
4	$\begin{pmatrix}10\\8\\9\end{pmatrix}$	13	$\begin{pmatrix}1.4559\\-1.3422\\-0.0011\end{pmatrix}$	16	$\begin{pmatrix}1.2051\\-0.7542\\-0.0000\end{pmatrix}$
5	$\begin{pmatrix}7\\3\\8\end{pmatrix}$	9	$\begin{pmatrix}1.3421\\-1.0236\\0.3291\end{pmatrix}$	13	$\begin{pmatrix}1.2603\\-0.8490\\0.3862\end{pmatrix}$
6	$\begin{pmatrix}2\\10\\7\end{pmatrix}$	17	$\begin{pmatrix}1.3766\\-1.1076\\0.1774\end{pmatrix}$	20	$\begin{pmatrix}1.3556\\-1.0486\\0.1748\end{pmatrix}$
7	$\begin{pmatrix}5\\7\\6\end{pmatrix}$	15	$\begin{pmatrix}1.3805\\-1.1174\\0.1072\end{pmatrix}$	17	$\begin{pmatrix}1.3793\\-1.1073\\0.1094\end{pmatrix}$
8	$\begin{pmatrix}-9\\-5\\-1\end{pmatrix}$	8	$\begin{pmatrix}1.2285\\-0.7931\\0.0020\end{pmatrix}$	8	$\begin{pmatrix}1.2274\\-0.7917\\-0.0000\end{pmatrix}$
9	$\begin{pmatrix}3\\9\\5\end{pmatrix}$	17	$\begin{pmatrix}1.4242\\-1.2351\\0.0938\end{pmatrix}$	19	$\begin{pmatrix}1.3861\\-1.1252\\0.0935\end{pmatrix}$

6. Conclusions

We have applied the gradient descent approach to solve (UP), and gave necessary optimality condition based on q-derivative. We proved the convergence based on the proposed algorithm. The steepest descent method for multiobjective optimization with q-derivative converges independently of the starting point to a critical Pareto point. The solution strategy does not require any ordering information. The proposed algorithm has been compared with the existing method to measure the efficiency of algorithm.

Acknowledgments

The authors would like to sincerely thanks the anonymous referees for their valuable comments and suggestions, which helped us to improve the article. The research is supported by the Science and Engineering Research Board (Grant No. DST-SERB-MTR-2018/000121) and the University Grants Commission (IN) (Grant No. UGC-2015-UTT–59235).

Conflict of interest

The authors declare that they have no competing interests.

References

1. I. Marusciac, *On Fritz John type optimality criterion in multi-objective optimization*, Anal. Numer. Theor. Approx., **11** (1982), 109–114.
2. G. Palermo, C. Silvano, S. Valsecchi, et al. *A system-level methodology for fast multi-objective design space exploration*, ACM, New York, 2003.
3. M. Tavana, *A subjective assessment of alternative mission architectures for the human exploration of mars at NASA using multicriteria decision making*, Comput. Oper. Res., **31** (2004), 1147–1164.
4. H. Eschenauer, J. Koski, A. Osyczka, *Multicriteria design optimization: Procedures and applications*, Springer-Verlag, Berlin, 1990.
5. I. Das, J. E. Dennis, *Normal-Boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems*, SIAM J. Optim.,**8** (1998), 631–657.
6. I. Y. Kim, O. L. de Weck, *Adaptive weighted-sum method for bi-objective optimization: Pareto front generation*, Struct. Multidisc. Optim., **29** (2005), 149–158.
7. Y. Fu, U. M. Diwekar, *An efficient sampling approach to multiobjective optimization*, Ann. Oper. Res., **132** (2004), 109–134.
8. A. Jüschke, J. Jahn, A. Kirsch, *A bicriterial optimization problem of antenna design*, Comput. Optim. Appl., **7** (1997), 261–276.
9. T. M. Leschine, H. Wallenius, W. A. Verdini, *Interactive multiobjective analysis and assimilative capacity-based ocean disposal decisions*, Eur. J. Oper. Res., **56** (1992), 278–289.
10. J. Fliege, *OLAF-A general modeling system to evaluate and optimize the location of an air polluting facility*, OR Spektrum, **23** (2001), 117–136.
11. E. Carrizosa, J. B. G. Frenk, *Dominating sets for convex functions with some applications*, J. Optim. Theory Appl., **96** (1998), 281–295.
12. G. W. Evan, *An overview of techniques for solving multiobjective mathematical programs*, Manage. Sci., **30** (1984), 1268–1282.
13. D. Prabuddha, J. B. Ghosh, C. E. Wells, *On the minimization of completion time variance with a bicriteria extension*, Oper. Res., **40** (1992), 1148–1155.
14. D. J. White, *Epsilon-dominating solutions in mean-variance portfolio analysis*, Eur. J. Oper. Res., **105** (1998), 457–466.
15. M. G. C. Tapia, C. A. C. Coello, *Applications of multi-objective evolutionary algorithms in economics and finance: A survey*, IEEE Congr. Evol. Comput., (2007), 532–539.

16. E. Zitzler, K. Deb, L. Thiele, *Comparison of multiobjective evolutionary algorithms: Empirical results*, Evol. Comput., **8** (2000), 173–195.

17. M. Ehrgott, *Multicriteria optimization*, Springer, Berlin, 2005.

18. K. Miettinen, M. M. Mäkelä, *Interactive bundle-based method for nondifferentiable multiobjeective optimization: NIMBUS*, Optimization, **34** (1995), 231–246.

19. L. M. G. Drummond, B. F. Svaiter, *A steepest descent method for vector optimization*, J. Comput. Appl. Math., **175** (2005), 395–414.

20. J. Fliege, L. M. G. Drummond, B. F. Svaiter, *Newton's method for multiobjective optimization*, SIAM J. Optim., **20** (2009), 602–626.

21. A. Cauchy, *Méthode générale pour la résolution des systèms d'équations simultanées*, Comp. Rend. Sci. Paris, **25** (1847), 536–538.

22. H. Zhao, J. Zheng, W. Deng, et al. *Semi-supervised broad learning system based on manifold regularization and broad network*, IEEE Trans. Circuits Syst., **67** (2020), 983–994.

23. W. Deng, J. Xu, Y. Song, et al. *An effective improved co-evolution ant colony optimization algorithm with multi-strategies and its application*, Int. J. Bio-Inspir. Com., (2020) 1–10.

24. W. Deng, J. Xu, H. Zhao, *An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem*, IEEE access, **7** (2019), 20281–20292.

25. W. Deng, H. Liu, J. Xu, et al. *An improved quantum-inspired differential evolution algorithm for deep belief network*, IEEE Trans. Instrum. Meas., 2020.

26. S. K. Mishra, B. Ram, *Introduction to Unconstrained Optimization with R*, Springer Nature, Singapore, 2019.

27. J. Fliege, B. F. Svaiter, *Steepest descent methods for multicriteria optimization*, Math. Method Oper. Res., **51** (2000), 479–494.

28. T. D. Chuong, J. C. Yao, *Steepest descent methods for critical points in vector optimization problems*, Appl. Anal., **91** (2012), 1811–1829.

29. J. Fliege, A. I. F. Vaz, L. N. Vicente, *Complexity of gradient descent for multiobjective optimization*, Optim. Method Softw., **34** (2019), 949–959.

30. F. H. Jackson, *On q-functions and a certain difference operator*, Trans. Roy Soc. Edin., **46** (1908) 253–281.

31. F. H. Jackson, *On q-Definite Integrals*, Pure Appl. Math. Q., **41** (1910), 193–203.

32. A. Aral, V. Gupta, R. P. Agarwal, *Applications of q-calculus in operator theory*, Springer, New York, 2013.

33. P. M. Rajković, M. S. Stanković, S. D. Marinković, *Mean value theorems in q-calculus*, Mat. Vesn., **54** (2002), 171–178.

34. M. E. Ismail, D. Stanton, *Applications of q-Taylor theorems*, J. Comput. Appl. Math., **153** (2003), 259–272.

35. S. C. Jing, H. Y. Fan, *q-Taylor's formula with its q-remainder*, Commun. Theor. Phys., **23** (1995), 117.

36. P. M. Rajković, S. D. Marinković, M. S. Stanković, *Fractional integrals and derivatives in q-calculus*, Appl. Anal. Discrete Math., **1** (2007), 311–323.

37. H. Gauchman, *Integral inequalities in q-calculus*, Comput. Math. Appl., **47** (2004), 281–300.

38. G. Bangerezako, *Variational q-calculus*, J. Math. Anal. Appl., **289** (2004), 650–665.

39. L. Abreu, *A q-sampling theorem related to the q-Hankel transform*, Proc. Am. Math. Soc., **133** (2005), 1197–1203

40. T. H. Koornwinder, R. F. Swarttouw, *On q-analogues of the Fourier and Hankel transforms*, Trans. Am. Math. Soc., **333** (1992), 445–461.

41. J. Ablinger, A. K. Uncu, *qFunctions-A Mathematica package for q-series and partition theory applications*, arXiv preprint arXiv:1910.12410, 2019.

42. A. C. Soterroni, R. L. Galski, F. M. Ramos, *The q-gradient vector for unconstrained continuous optimization problems*, In: Operations research proceedings 2010. Springer-Verlag Berlin Heidelberg, (2011), 365–370.

43. A. C. Soterroni, R. L. Galski, F. M. Ramos, *The q-gradient method for global optimization*, arXiv:1209.2084, 2012.

44. A. C. Soterroni1, R. L. Galski, M. C. Scarabello1, et al. *The q-G method: A q-version of the steepest descent method for global optimization*, SpringerPlus, **4** (2015), 647.

45. A. C. Soterroni, R. L. Galski, F. M. Ramos, *Satellite constellation design using the q-G global optimization method*, Third World Conference on Complex Systems (WCCS), IEEE, (2015), 1–6.

46. U. M. Al-Saggaf, M. Moinuddin, M. Arif, et al. *The q-least mean squares algorithm*, Signal Process., **111** (2015), 50–60.

47. A. Sadiq, S. Khan, I. Naseem, et al. *Enhanced q-least mean square*, Circ. Syst. Signal Process., **38** (2019), 4817–4839.

48. P. M. Rajković, S. D. Marinković, M. S. Stankovic, *On q-Newton-Kantorovich method for solving systems of equations*, Appl. Math. Comput., **168** (2005), 1432–1448.

49. S. K. Chakraborty, G. Panda, *Newton like line search method using q-calculus*, In: D. Giri, R. N. Mohapatra, H. Begehr, M. Obaidat, Communications in Computer and Information Science, Singapore: Springer, 2017.

50. S. K. Mishra, G. Panda, M. A. T. Ansary, et al. *On q-Newton's method for unconstrained multiobjective optimization problems*, J. Appl. Math. Comput., (2020), 1–20.

51. Ž. Povalej, *Quasi-Newton's method for multiobjective optimization*, J. Comput. Appl. Math., **255** (2014), 765–777.

52. K. K. Lai, S. K. Mishra, B. Ram, *On q-Quasi-Newton's Method for Unconstrained Multiobjective Optimization Problems*, Mathematics, **8** (2020), 616.

53. 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.

54. W. E. Karush, *Minima of functions of several variables with inequalities as side conditions. PhD thesis*, Univ. Chicago, 1939.

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

56. K. Deb, L. Thiele, M. Laumanns, et al. *Scalable multi-objective optimization test problems*, IEEE, 1 (2002), 825–830.

57. O. P. Ferreira, M. S. Louzeiro, L. F. Prudente, *Iteration-complexity and asymptotic analysis of steepest descent method for multiobjective optimization on Riemannian manifolds*, J. Optim. Theory Appl., **184** (2020), 507–533.

AIMS Press