U.P.B. Sci. Bull., Series C, Vol. 80, Iss. 1, 2018 ISSN 2286-3540

MODIFIED SIT ALGORITHM FOR MULTIOBJECTIVE QUADRATICALLY CONSTRAINED QUADRATIC PROGRAMMING

Hossein SALMEI[1] and Mohammad Ali YAGHOOBI[2*]

Most of existing methods for solving multiobjective quadratic programming problems consider convex objective functions and linear constraints. We suppose that objective functions are either convex or nonconvex. Moreover, constraints can be stated in both linear and quadratic forms. To solve these problems, a modified version of the SIT algorithm is proposed which converges to a finite subset of essential epsilon-efficient solutions. Further, we use polyblocks which their upper boundaries approximate the efficient frontier of the problem.

Keywords: multiobjective programming, the constraint method, minmax optimization, quadratic programming, essential ε-optimal solution, the SIT algorithm.

1. Introduction

We consider the following problem:

$$\begin{array}{lll} \min & f(x) = (f_1(x), \dots, f_p(x)) & \\ s.t. & f_k(x) \geqslant 0 & k = p+1, \dots, m, \\ & x \in [a,b], & \end{array} \tag{1}$$

where a and b are n dimensional vectors of nonnegative real numbers, $x \in [a,b]$ means $a_i \leqslant x_i \leqslant b_i$ for all $i = 1, \dots, n$, and each function $f_k : \mathbb{R}^n \to \mathbb{R}$ is of the form,

$$f_k(x) = \sum_{i=1}^{n} c_i^k x_i^2 + \sum_{i,j=1; i<j}^{n} c_{ij}^k x_i x_j + \sum_{i=1}^{n} d_i^k x_i + d_k, \; k = 1, \dots, m. \tag{2}$$

Problem (1) is a multiobjective quadratically constrained quadratic programming (MQCQP) problem. If $p = 1$, it involves a single objective and we use the term SQCQP instead of MQCQP. In either case, the feasible set of problem (1) is denoted by $\mathcal{F} := \{x \in \mathbb{R}^n \mid f_k(x) \geqslant 0, \; k = p+1, \dots, m, \; x \in [a,b]\}$ and the set $\mathcal{Y} := \{y \in \mathbb{R}^p \mid y = f(x), \; x \in \mathcal{F}\}$ called the image of $\mathcal{F}$ under f in the objective space.

Quadratically constrained quadratic programming is an important and well-known technique for formulating and dealing with different mathematical programming problems. Many programming problems such as mixed integer, fractional,

[1] Ph.D. Student, Dept. of Applied Mathematics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran, e-mail: salmei@vru.ac.ir

[2*] (Corresponding author) Assoc. Prof., Dept. of Applied Mathematics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran, e-mail: yaghoobi@uk.ac.ir

bilevel, and polynomial programming problems can be written as instances of SQCQP [6, 9]. Hladik [5] computed the optimal value range of convex SQCQP problems when the coefficients are subject to perturbations in given intervals. Tuy and Hoai-Phuong [10] developed the SIT algorithm in order to solve SQCQP problems in nonconvex case. Their algorithm is stable under small perturbations.

On the other hand, in many real world problems more than one objective function should be considered. Thus, multiobjective programming problems occur during the modelling of those problems [3]. In the literature, most of researches in the field of multiobjective quadratic programming deal with convex quadratic objective functions and linear constraints. Beato-Moreno et al. [1] proposed a technique for calculating the equations of the efficient points of an unconstrained multiobjective quadratic programming with strictly convex functions. In the case of convex objective functions, Beato-Moreno et al. [2] obtained some conditions for (weakly) efficient solutions. Goh and Yang [4] presented an analytical method for computing the exact efficient solution set of multiobjective convex quadratic programs with linear constraints. Oberdieck and Pistikopoulos [8] suggested an approximate algorithm for the explicit calculation of the efficient frontier of a multiobjective optimization problem with convex quadratic cost functions and linear constraints.

In this paper, we introduce a technique for solving an MQCQP problem generally in nonconvex case. In fact, we convert problem (1) to an SQCQP problem based on combination of the well-known minmax optimization and constraint method [3]. Then, we solve the SQCQP problem by developing the SIT algorithm [10] to multiobjective functions. The rest of the paper is organized as follows. Section 2 introduces preliminaries and explains the basic concepts of the SIT algorithm briefly. Main results are given in Section 3. The modified SIT algorithm for solving problem (1) is developed in Section 4. Finally, Section 5 is devoted to conclusions.

2. **Preliminaries**

At first we introduce some basic notations and definitions from [3, 9, 10]. Throughout the paper, $\mathbb{R}^n$ denotes the n dimensional Euclidean space, e^i is the i-th unit vector of $\mathbb{R}^n$, and $\mathbf{1} \in \mathbb{R}^n$ is a vector with all components equal to one. If $x, y \in \mathbb{R}^n$ then $x \leqq y$ $(x < y)$ if and only if $x_i \leqslant y_i$ $(x_i < y_i)$, $\forall\, i = 1, \ldots, n$. In addition, $x \leq y$ means that $x \leqq y$ and $x \neq y$. We will denote by $\mathbb{R}^n_{\geqq}$ the set $\{x \in \mathbb{R}^n \mid 0 \leqq x\}$.

Definition 2.1. ([3]) Consider an MQCQP problem. The feasible solution $\hat{x} \in \mathcal{F}$ is called efficient (weak efficient) if there is no another $x \in \mathcal{F}$ such that $f(x) \leq f(\hat{x})$ $(f(x) < f(\hat{x}))$. If $\hat{x} \in \mathcal{F}$ is efficient (weak efficient) then $\hat{y} = f(\hat{x})$ is called a nondominated (weak nondominated) point. The set of all efficient solutions and nondominated points are called the efficient set and efficient frontier, respectively.

Definition 2.2. ([9]) A function $h : \mathbb{R}^n \longrightarrow \mathbb{R}$ is said to be increasing on the orthant $\mathbb{R}^n_{\geqq}$ if $h(x) \leqq h(y)$ whenever $0 \leqq x \leqq y$. It is a dm (difference of monotonic) function on $\mathbb{R}^n_{\geqq}$ if $h(x) = h^+(x) - h^-(x)$, where h^+, h^- are increasing on $\mathbb{R}^n_{\geqq}$.

Clearly a polynomial of n variables with positive coefficients is increasing on $\mathbb{R}^n_{\geqq}$. Since a polynomial can be represented as a difference of two polynomials with positive coefficients, it is a dm function on $\mathbb{R}^n_{\geqq}$. Consequently, every quadratic function of form (2) is also a dm function on $\mathbb{R}^n_{\geqq}$.

Definition 2.3. ([9])

(i) A set $P \subseteq \mathbb{R}^n_{\geqq}$ is called *polyblock* if $P = \cup_{z\in V}[0,z]$ whenever the set V, called the *vertex set*, is a finite subset of P. A point $z \in V$ is called a *proper vertex* of P if there is no $z' \in V\backslash\{z\}$ such that $z' \leqq z$. Otherwise, $z \in V$ is called *improper*.

(ii) Let $P \subseteq [\bar{a},\bar{b}]$ be a polyblock. A point $y \in P$ is called an *upper boundary point* if there does not exist another point $x \in P$ such that $x = \bar{a} + \lambda(y-\bar{a})$ with $\lambda > 1$. We denote the set of all upper boundary points of P by $\partial^+ P$.

Proposition 2.1. *(*[9]*) Let $P \subseteq [\bar{a},\bar{b}]$ be a polyblock with proper vertex set V and let $x \in [\bar{a},\bar{b}]$ satisfy $V_x := \{z \in V \mid x < z\} \neq \emptyset$. Then:*

(i) *$P' := P\backslash(x,\bar{b}]$ is a polyblock with vertex set*
$$V' = (V\backslash V_x) \cup \{z^i = z + (x_i - z_i)e^i \mid z \in V_x, i = 1,...,n\}.$$

(ii) *The proper vertex set of P' is obtained from V' by removing improper elements according to the rule: "For every pair $z \in V_x, y \in V_x^+ := \{y \in V \mid x \leqq y\}$ compute $J(z,y) := \{j \mid z_j > y_j\}$. If $J(z,y) = \{i\}$ then z^i is an improper element of V'."*

In the reminder of this section, we consider problem (1) with $p = 1$, i.e. an SQCQP problem. Since $f_1, f_2, ..., f_m$ are dm functions on $\mathbb{R}^n_{\geqq}$, the SQCQP problem can be converted, without loss of generality, to the following problem [10]:

$$\min\{h(x)|g(x) \geqslant 0, x \in [a,b]\} \tag{3}$$

where $h : \mathbb{R}^n \longrightarrow \mathbb{R}$ is an increasing quadratic function of form (2) with positive coefficients and $g(x) = \min_{k=1,...,m}\{g_k(x) := u_k(x) - v_k(x)\}$ where u_k, v_k are increasing quadratic functions with the same form as (2).

2.1. Essential optimal solutions

It has been argued in [9, 10] that an isolated optimal solution of problem (3) is often difficult to be implemented in practice because of its instability under small perturbations of the constraints. This is the motivation of the following definition.

Definition 2.4. ([9, 10]) Consider Problem (3). Assume that $\{x \mid g(x) > 0,\ x \in [a,b]\} \neq \emptyset$ and let $\varepsilon > 0$ be given. A vector $x \in \mathbb{R}^n$ is an *ε-essential (nonisolated) feasible solution* if $x \in \mathcal{F}_\varepsilon := \{x \mid g(x) \geqslant \varepsilon,\ x \in [a,b]\}$. A solution $x^\star \in \mathcal{F}_\varepsilon$ is called an *essential ε-optimal solution* if:

$$h(x^\star) - \varepsilon \leqslant \min\{h(x) \mid x \in \mathcal{F}_\varepsilon\}. \tag{4}$$

We call $x^\star$ an *essential strictly ε-optimal solution* if inequality (4) is strict.

Theorem 2.1. *(*[9, 10]*) Suppose that $z^\star$ and $x^\star$ are the optimal value and the optimal solution of the following auxiliary problem, respectively:*

$$\max\{g(x)|\ h(x) \leqslant \gamma - \varepsilon,\ x \in [a,b]\}. \tag{5}$$

If $z^\star > 0$ then $x^\star$ is a nonisolated feasible solution of problem (3) with $h(x^\star) \leqslant \gamma - \varepsilon$. If $z^\star < \varepsilon$ and $\gamma = h(\bar{x})$ for some $\bar{x} \in \mathcal{F}_\varepsilon$, then $\bar{x}$ is an essential ε-optimal solution of problem (3). If $z^\star < \varepsilon$ and $\gamma = h(b) + \varepsilon$ then problem (3) is ε-essentially infeasible (i.e problem (3) has no ε-essential feasible solution).

To solve the auxiliary problem (5), an approach is developed in [10] which is based on three operations: Branching, Reducing and Bounding.

- *Branching* produces a sequence of nested partition sets that shrinks to a singleton. It can be done by the popular *standard bisection* technique.
- *Reduction* reduces a box $M = [p,q] \subseteq [a,b]$ to $[p',q']$ without losing any desirable feasible point. The box $[p',q']$ is called *valid reduction* and denoted by redM. We use Lemma 1 of [10] to calculate $[p',q']$.
- *Bounding* consists of estimating an upper bound $\beta(M)$ to problem (5) over a valid reduction $[p',q']$. We use LP(M) of [10] to estimate $\beta(M)$ as the optimal value of a linear relaxation of problem (5).

3. Main results

The well-known minmax approach converts problem (1) to a single optimization problem with the objective function "min $\max\{f_1(x),...,f_p(x)\}$" without changing the constraint set [3]. Another well-known approach to deal with problem (1) is the constraint method. It chooses one objective function (for instance, f_j) and considers upper bounds for the other objective functions. In fact, it solves [3]:

$$\begin{array}{lll} \min & f_j(x) & \\ s.t. & f_k(x) \leqslant f_k^u & k = 1,...,p,\ k \neq j, \\ & f_k(x) \geqslant 0 & k = p+1,...,m;\ x \in [a,b], \end{array} \tag{6}$$

where f_k^u is an arbitrary upper bound of the k-th objective function. A candidate for such an upper bound is $f_k^u = f_k^+(b) - f_k^-(a)$.

The minmax approach and the constraint method are known as scalarization techniques [3]. In what follows, we propose another scalarization technique. In fact, we combine the above two methods and convert problem (1) to:

$$\begin{array}{lll} \min & \max\{f_1(x),...,f_p(x)\} & \\ s.t. & f_k(x) \leqslant f_k^u & k = 1,...,p, \\ & f_k(x) \geqslant 0 & k = p+1,...,m;\ x \in [a,b]. \end{array} \tag{7}$$

If $x_{n+1} := \max\{f_1(x),...,f_p(x)\}$, model (7) can be rewritten equivalently as follows:

$$\begin{array}{lll} \min & x_{n+1} & \\ s.t. & x_{n+1} - f_k(x) \geqslant 0 & k = 1,...,p, \\ & f_k^u - f_k(x) \geqslant 0 & k = 1,...,p, \\ & f_k(x) \geqslant 0 & k = p+1,...,m, \\ & (x, x_{n+1}) \in [(a, a_{n+1}), (b, b_{n+1})], & \end{array} \tag{8}$$

where $a_{n+1} = \max\{f_k^+(a) - f_k^-(b)\}_{k=1}^p$, $b_{n+1} = \max\{f_k^+(b) - f_k^-(a)\}_{k=1}^p$.

The objective function of problem (8) is a strictly increasing quadratic function and the left hand side of its constraints are quadratic functions. Thus, problem (8) is an SQCQP problem of the form (3) and can be solved by the approach discussed in Section 2 for each fixed vector $(f_1^u,...,f_p^u)$. We develop that approach to approximate the efficient frontier of problem (1). To this end, initially, we use the definition of ε-efficient solutions [7] for extending the concept of essential ε-optimality to essential ε-efficiency in the case of multiobjective functions.

Definition 3.1. Consider problem (1) and let $\varepsilon > 0$ be given.

- A point $\hat{x} \in \mathcal{F}_\varepsilon$ is called an *essential ε-efficient* (ε-weak efficient) solution if there is no another $x \in \mathcal{F}_\varepsilon$ such that $f(x) + \varepsilon\mathbf{1} \leq (<) f(\hat{x})$.
- If $\hat{x} \in \mathcal{F}_\varepsilon$ is an essential ε-efficient (ε-weak efficient) solution then $\hat{y} = f(\hat{x})$ is called an *essential ε-nondominated (ε-weak nondominated)* point.

Theorem 3.1. *Suppose that $(x^\star, x_{n+1}^\star)$ is an essential ε-optimal solution of problem (8). Then $x^\star$ is an essential ε-weak efficient solution of problem (1). If $(x^\star, x_{n+1}^\star)$ is an essential strictly ε-optimal solution then $x^\star$ is an essential ε-efficient solution.*

Proof. Suppose that $x^\star$ is not an essential ε-weak efficient solution of problem (1). Then, there exists $\bar{x} \in \mathcal{F}_\varepsilon$ such that $f_k(\bar{x}) + \varepsilon < f_k(x^\star)$ for $k = 1,...,p$. Thus, there is $\varepsilon_0 > 0$ such that $f_k(\bar{x}) + \varepsilon + \varepsilon_0 < f_k(x^\star)$ for all $k = 1,...,p$. Since $(x^\star, x_{n+1}^\star)$ is an essential ε-optimal solution of problem (8), we have:

- $f_k^u - f_k(\bar{x}) > f_k^u - f_k(x^\star) + \varepsilon + \varepsilon_0 > f_k^u - f_k(x^\star) \geqslant \varepsilon,\ k = 1,...,p. \quad (9)$
- $x_{n+1}^\star - \varepsilon_0 - f_k(\bar{x}) > x_{n+1}^\star - f_k(x^\star) + \varepsilon > x_{n+1}^\star - f_k(x^\star) \geqslant \varepsilon,\ k = 1,...,p. \quad (10)$
- $x_{n+1}^\star - \varepsilon_0 \geqslant f_k(x^\star) + \varepsilon - \varepsilon_0 > f_k(\bar{x}) + 2\varepsilon > f_k(\bar{x}) \geqslant f_k^+(a) - f_k^-(b),\ k = 1,...,p. \quad (11)$
- $x_{n+1}^\star - \varepsilon_0 \leqslant b_{n+1} - \varepsilon_0 \leqslant b_{n+1}. \quad (12)$

Define $\bar{x}_{n+1} = x_{n+1}^\star - \varepsilon_0$. Inequalities (11) and (12) imply that $\bar{x}_{n+1} \in [a_{n+1}, b_{n+1}]$. Since $\bar{x} \in \mathcal{F}_\varepsilon$, we conclude, by (9) and (10), that $(\bar{x}, \bar{x}_{n+1})$ is an ε-essential feasible solution of problem (8). We also have $\bar{x}_{n+1} - \varepsilon = x_{n+1}^\star - \varepsilon_0 - \varepsilon < x_{n+1}^\star - \varepsilon$, which contradicts the essential ε-optimality of $(x^\star, x_{n+1}^\star)$ to problem (8). This completes the proof of the first statement. For the second part, we conclude similarly that $(\bar{x}, \bar{x}_{n+1} = x_{n+1}^\star)$ is an ε-essential feasible solution of problem (8). Now, $\bar{x}_{n+1} - \varepsilon = x_{n+1}^\star - \varepsilon$ contradicts the essential strictly ε-optimality of $(x^\star, x_{n+1}^\star)$. □

The next theorem shows that problem (8) is able to obtain any essential ε-efficient solution of an arbitrary MQCQP problem.

Theorem 3.2. *Consider problem (1) and let $\hat{x} \in \mathcal{F}_\varepsilon$ be an essential ε-efficient solution. Then, there is a vector $(f_1^u, ..., f_p^u)$ such that $(\hat{x}, \hat{x}_{n+1})$ is an essential ε-optimal solution of problem (8) where $\hat{x}_{n+1} = \max\{f_1(\hat{x}), ..., f_p(\hat{x})\} + \varepsilon$.*

Proof. Define $f_k^u = f_k(\hat{x})$ for $k = 1, ..., p$ and, on the contrary, suppose that $(\hat{x}, \hat{x}_{n+1})$ is not an essential ε-optimal solution of problem (8). Then, there exists an ε-essential feasible solution $(\bar{x}, \bar{x}_{n+1})$ of problem (8) such that $\bar{x}_{n+1} < \hat{x}_{n+1} - \varepsilon$. Thus, $f_k^u - f_k(\bar{x}) \geqslant \varepsilon,\ k = 1, ..., p$ which implies that $f_k(\bar{x}) + \varepsilon \leqslant f_k(\hat{x}),\ k = 1, ..., p$. Moreover, the constraints of problem (8) imply that $\bar{x}_{n+1} \geqslant \max\{f_1(\bar{x}), ..., f_p(\bar{x})\} + \varepsilon$. Let $\max\{f_1(\hat{x}), ..., f_p(\hat{x})\} = f_t(\hat{x})$. Then we have:

$$f_t(\bar{x}) + \varepsilon \leqslant \max\{f_1(\bar{x}), ..., f_p(\bar{x})\} + \varepsilon \leqslant \bar{x}_{n+1} < \hat{x}_{n+1} - \varepsilon = f_t(\hat{x}).$$

Therefore, $f_t(\bar{x}) + \varepsilon < f_t(\hat{x})$. Consequently, $f(\bar{x}) + \varepsilon\mathbf{1} \leq f(\hat{x})$, which contradicts the essential ε-efficiency of $\hat{x}$. It completes the proof. □

Theorem 3.3. *Let $(x^\star, x_{n+1}^\star)$ be an essential ε-optimal solution of problem (8). Then, the polyblock $P^\star \subseteq \mathcal{Y}$ with vertex set $V^\star = \{f(x^\star)\}$ involves at least one nondominated point of problem (1).*

Proof. Consider the problem:

$$\begin{aligned} \min \quad & f_1(x) + f_2(x) + ... + f_p(x) \\ s.t. \quad & x \in \Omega = \{x \in \mathcal{F} \mid f(x) \in P^\star\}. \end{aligned} \tag{13}$$

Since $f_1, f_2, ..., f_p$ are continuous functions and $\mathcal{F}, P^\star$ are compact sets then Ω is a compact set and the objective function of problem (13) is continuous. By Weierstrass Theorem there is $\hat{x} \in \mathcal{F}$ such that $f(\hat{x}) \in P^\star$ and $\sum_{k=1}^p f_k(\hat{x}) \leqslant \sum_{k=1}^p f_k(x)$, $\forall\, x \in \Omega$. Then, $\hat{y} = f(\hat{x})$ is a nondominated point. Since otherwise there is $\bar{x} \in \mathcal{F}$ such that $f(\bar{x}) \leq f(\hat{x})$. This implies that $f(\bar{x}) \in P^\star$ $(f(\bar{x}) \leq f(\hat{x}) \leqq f(x^\star))$ and $\sum_{k=1}^p f_k(\bar{x}) < \sum_{k=1}^p f_k(\hat{x})$, which is a contradiction. It completes the proof. □

Indeed, Theorem 3.3 shows that the polyblock made by an essential ε-optimal solution of problem (8) includes a part of efficient frontier. In the next section, we will show that the whole efficient frontier is a subset of a polyblock made by only a finite number of essential ε-optimal solutions of problem (8).

Proposition 3.1. *Consider $M = [p, q] \subseteq [a, b]$. Suppose that L is a lower bound of the function f on M. Then, M does not contain any essential ε-weak efficient (ε-efficient) solution if there is $\bar{x} \in [a, b]$ such that $f(\bar{x}) < L - \varepsilon\mathbf{1}$ ($f(\bar{x}) \leq L - \varepsilon\mathbf{1}$).*

Proof. Let $x \in M$. Then $f(x) \geqq L$. If $f(\bar{x}) < L - \varepsilon\mathbf{1}$ then $L > f(\bar{x}) + \varepsilon\mathbf{1}$ which implies that $f(x) > f(\bar{x}) + \varepsilon\mathbf{1}$. Therefore, x is not an essential ε-weak efficient solution. Similarly, If $f(\bar{x}) \leq L - \varepsilon\mathbf{1}$ then $f(x) \geq f(\bar{x}) + \varepsilon\mathbf{1}$ and thus x cannot be an essential ε-efficient solution. □

4. **Modified SIT Algorithm**

In this section, we present a modification of the SIT algorithm to approximate the efficient set and the efficient frontier of problem (1). In fact, we convert the MQCQP problem (1) to the SQCQP problem (8). Then, the new algorithm assigns different upper bound vectors $(f_1^u,...,f_p^u)$ to problem (8) and runs the steps of SIT algorithm [10] for each individual upper bound vector to obtain different essential ε-weak efficient solutions. The set of those solutions approximates the efficient set.

Algorithm 4.1.

Input *a, b,* ε, $f=(f_1=f_1^+-f_1^-,...,f_p=f_p^+-f_p^-)$.

Step 0 *(Initialization) Let* $f^{u_0}:=(f_1^+(b)-f_1^-(a),...,f_p^+(b)-f_p^-(a))$, $\mathcal{U}:=\{f^{u_0}\}$, $a_{n+1}:=max\{f_k^+(a)-f_k^-(b)\}_{k=1}^p$, $b_{n+1}:=max\{f_k^+(b)-f_k^-(a)\}_{k=1}^p$, *and* $\mathcal{X}_{out}:=\emptyset$.

Step 1 *Set* $\gamma:=b_{n+1}+\varepsilon$, $M_1:=[(a,a_{n+1}),(b,b_{n+1})]$, $\mathcal{P}_1:=\{M_1\}$, $\mathcal{R}_1:=\emptyset$, $k:=1$, *and select a vector* $f^u\in\mathcal{U}$.

Step 2 *For each box* $M\in\mathcal{P}_k$ *related to problem (8) with the selected* $f^u\in\mathcal{U}$:

- 2-1 *Compute valid reduction, redM. If* $redM=\emptyset$ *then* $\mathcal{P}_k:=\mathcal{P}_k\setminus\{M\}$. *Otherwise, substitute M by redM.*
- 2-2 *If* $redM=[p',q']$ *then obtain* $\beta(M)$ *by solving* $LP([p',q'])$. *If* $\beta(M)<0$ *then* $\mathcal{P}_k:=\mathcal{P}_k\setminus\{M\}$.

Step 3 *Set* $\mathcal{R}_k:=\mathcal{R}_k\cup\mathcal{P}_k$. *If* $\mathcal{R}_k=\emptyset$ *then go to Step 6; otherwise obtain:*

$$[p^k,q^k]:=M_k\in argmax\{\beta(M)\mid M\in\mathcal{R}_k\}.$$

Step 4 *If* $\beta(M_k)<\varepsilon$ *then go to step 6; otherwise compute*

$$\lambda_k:=\max\{\alpha|p_{n+1}^k+\alpha(q_{n+1}^k-p_{n+1}^k)\leqslant\gamma-\varepsilon\} \text{ and } x^k:=p^k+\lambda_k(q^k-p^k).$$

- 4-1 *If* $g(x^k)>0$ *then* x^k *is a new nonisolated feasible solution of problem (8) with* $x_{n+1}^k\leqslant\gamma-\varepsilon$. *If* $g(p^k)<0$ *then compute the point* $\bar{x}^k$ *where the line segment joining* p^k *to* x^k *meets the surface* $g(x)=0$, *and set* $\bar{x}:=\bar{x}^k$; *otherwise set* $\bar{x}:=p^k$. *Let* $\gamma:=\bar{x}_{n+1}$ *and go to Step 5.*
- 4-2 *If* $g(x^k)\leqslant 0$ *then go to Step 5.*

Step 5 *Divided* M_k *into two subboxes* M_k^1 *and* M_k^2, *by the standard bisection (or any bisection consistent with the bounding* $\beta(M)$*). Let* L^1 *and* L^2 *be the lower bounds of f on* M_k^1 *and* M_k^2, *respectively (without considering the* $n+1$*-th dimension).*

- 5-1 *Set* $\mathcal{P}_{k+1}:=\{M_k^1,M_k^2\}$ *and* $\mathcal{R}_{k+1}:=\mathcal{R}_k\setminus\{M_k\}$.
- 5-2 *If* $f(\bar{x}_1,...,\bar{x}_n)<L^i-\varepsilon\mathbf{1}$ *then* $\mathcal{P}_{k+1}:=\mathcal{P}_{k+1}\setminus\{M_k^i\}$, *for* $i=1,2$.
- 5-3 *Let* $k:=k+1$ *and go to Step 2.*

Step 6 *Set* $\mathcal{U}:=\mathcal{U}\setminus\{f^u\}$. *If* $\gamma=b_{n+1}+\varepsilon$ *then problem (8) with the selected* $f^u\in\mathcal{U}$ *is* ε*-essentially infeasible. Otherwise:*

$$\mathcal{U}:=\mathcal{U}\cup\{f^{u_j}|f^{u_j}=(f_1^u,...,f_{j-1}^u,f_j(\bar{x}_1,...,\bar{x}_n)-\varepsilon,f_{j+1}^u,...,f_p^u),\ j=1,...,p\},$$

$$\mathcal{X}_{out}:=\mathcal{X}_{out}\cup\{(\bar{x}_1,...,\bar{x}_n)\}.$$

If $\mathcal{U}=\emptyset$ *then stop; otherwise go to Step 1.*

Output *The set $\mathcal{X}_{out}$ and $\mathcal{Y}_{out} := \{y \in \mathbb{R}^p \mid y = f(x),\ x \in \mathcal{X}_{out}\}$ as discrete approximations to the efficient set and the efficient frontier of problem (1), respectively.*

Note that in Algorithm 4.1 all vectors $x^k, p^k, q^k, \bar{x}^k, \bar{x}$ belong to $\mathbb{R}^{n+1}$ and the function g defines as in problem (3) according to the constraints of problem (8).

Remark 4.1. *Steps 1-5 of Algorithm 4.1 are taken from Steps 0-7 of the SIT algorithm* [10] *and are rewritten in a relevant way to solve problem (8) for a selected $f^u \in \mathcal{U}$. Therefore, these steps, based on Proposition 2 of* [10]*, terminate after a finite number of iterations and converge to an essential ε-optimal solution or show that the underling problem is ε-essentially infeasible. It should be noted that, based on Proposition 3.1 and Theorem 3.1, we delete some boxes in Step 5 which do not contain any ε-optimal solutions.*

In what follows, we discuss some properties of Algorithm 4.1. Initially, we prove that it is convergent.

Theorem 4.1. *Algorithm 4.1 terminates after a finite number of iterations.*

Proof. On the contrary suppose that Algorithm 4.1 does not terminate in finitely many iterations. Based on Remark 4.1, steps 1-5 terminate after a finite number of iterations. Thus, Step 6 will be visited infinitely many times. This means that the set $\mathcal{U}$ should involve infinite number of elements. Moreover, the upper bound of at least one objective function should be changed infinitely. Without loss of generality, suppose that it is the function f_t $(1 \leqslant t \leqslant p)$ and the corresponding upper bounds are $\{f_t^{u_1}, f_t^{u_2}, ...\}$ with the corresponding essential ε-optimal solutions $\{\bar{x}^1, \bar{x}^2, ...\}$. By Step 6 of Algorithm 4.1, $f_t^{u_{i+1}} = f_t(\bar{x}_1^i, ..., \bar{x}_n^i) - \varepsilon,\ \forall\ i \geqslant 1$. Since $\bar{x}^1, \bar{x}^2, ...$ are essential ε-optimal solutions of problem (8), we have:

$$f_t(\bar{x}_1^i, ..., \bar{x}_n^i) \leqslant f_t^{u_i} - \varepsilon = f_t(\bar{x}_1^{i-1}, ..., \bar{x}_n^{i-1}) - 2\varepsilon \leqslant ... \leqslant f_t(\bar{x}_1^1, ..., \bar{x}_n^1) - i\varepsilon. \tag{14}$$

From (14), we conclude that $f_t(\bar{x}_1^i, ..., \bar{x}_n^i) \to -\infty$ as $i \to +\infty$, which is a contradiction, since $f_t^+(a) - f_t^-(b)$ is a lower bound of f_t. This completes the proof. $\square$

Let $\bar{a}_i := f_i^+(a) - f_i^-(b)$ and $\bar{b}_i := f_i^+(b) - f_i^-(a)$ for $i = 1, 2, ..., p$. Then, it is clear that $\mathcal{Y}_{out} \subseteq \mathcal{Y} \subseteq [\bar{a}, \bar{b}]$. Without loss of generality, we suppose that $[\bar{a}, \bar{b}]$ is a polyblock. Since otherwise the simple shift $\mathcal{Y} - \bar{a}$ makes the desired feature. Therefore, the polyblock $P_0 := [\bar{a}, \bar{b}]$ with the proper vertex set $\{\bar{b}\}$ involves the efficient frontier. Suppose that $\mathcal{Y}_{out} = \{y^1, y^2, ..., y^{\mathcal{N}}\}$. By using Proposition 2.1, we can generate new polyblocks by considering the points of $\mathcal{Y}_{out}$ one by one. In fact, we obtain $\mathcal{N}$ polyblocks $P_1, P_2, ..., P_{\mathcal{N}}$ such that $P_i = P_{i-1} \setminus (y^i, \bar{b}],\ i = 1, 2, ..., \mathcal{N}$. Next theorem shows that all these polyblocks contain the efficient frontier.

Theorem 4.2. *If the efficient frontier of problem (1) is nonempty then it is a subset of all polyblocks $P_i,\ i = 0, 1, ..., \mathcal{N}$.*

Proof. The proof is by induction on i. It is clear that the efficient frontier is a subset of $P_0 = [\bar{a}, \bar{b}]$. Suppose that the efficient frontier is a subset of P_{i-1} and, on the contrary, it is not a subset of P_i. Therefore, there is a nondominated point $\tilde{y} = f(\tilde{x})$ such that $\tilde{y} \notin P_i$. Thus, $\tilde{y} \in (y^i = f(x), \bar{b}]$ where $x \in \mathcal{X}_{out}$. It means that $y^i = f(x) < \tilde{y} = f(\tilde{x})$, which contradicts $\tilde{y}$ being a nondominated point. $\square$

Indeed, by Theorem 4.2, the upper boundaries of the polyblocks $P_0, P_1, ..., P_{\mathcal{N}}$ can be considered as approximations to the efficient frontier of problem (1). However, $\partial^+ P_{\mathcal{N}}$ approximates the efficient frontier more precisely since it involves the most points. The following proposition shows an interesting feature of the set $\mathcal{Y}_{out}$ in biobjective case. It states that the minimum Euclidean distance among all points of $\mathcal{Y}_{out}$ is ε, which approves that the approximation points are well dispersed.

Proposition 4.1. *Consider a biobjective instance of problem (1). If $\mathcal{Y}_{out}$ is nonempty then the Euclidean distance between any two points of it is at least ε.*

Proof. Let $y^r = f(x^r)$ and $y^s = f(x^s)$ be two points of $\mathcal{Y}_{out}$ where $1 \leqslant r, s \leqslant \mathcal{N}$ and $x^r, x^s \in \mathcal{X}_{out}$ with corresponding vectors $f^{u_r}, f^{u_s} \in \mathcal{U}$. Then, by Step 6 of Algorithm 4.1 and the constraints of problem (8), the following three cases are possible:

(i) there is $1 \leqslant j \leqslant 2$ such that $f_j(x^r) \leqslant f_j^{u_r} - \varepsilon \leqslant f_j(x^s) - 2\varepsilon$,

(ii) there is $1 \leqslant j \leqslant 2$ such that $f_j(x^s) \leqslant f_j^{u_s} - \varepsilon \leqslant f_j(x^r) - 2\varepsilon$,

(iii) there exist $x^q \in \mathcal{X}_{out}$ and $f^{u_q} \in \mathcal{U}$ such that:

(1) $f_j(x^r) \leqslant f_j^{u_q} - \varepsilon = f_j(x^q) - 2\varepsilon$, (2) $f_h(x^s) \leqslant f_h^{u_q} - \varepsilon = f_h(x^q) - 2\varepsilon$,

where either $j = 1, h = 2$ or $j = 2, h = 1$.

Case (i) happens when Algorithm 4.1 attains x^s first and $f_j(x^s) - \varepsilon$ induces an upper bound on $f_j(x^r)$ based on problem (8). Case (ii) is the opposite of case (i). If none of cases (i) and (ii) happen then there will be x^q such that $f_j(x^q) - \varepsilon$ induces an upper bound on both $f_j(x^r)$ and $f_j(x^s)$. Cases (i) and (ii) imply that $f_j(x^s) - f_j(x^r) \geqslant \varepsilon$ or $f_j(x^r) - f_j(x^s) \geqslant \varepsilon$. We show that case (iii) also leads to a similar result. Since (x^q, x^q_{n+1}) is an essential ε-optimal solution of problem (8), we have:

$$\max\{f_1(x^q), f_2(x^q)\} = x^q_{n+1} - \varepsilon \leqslant \min\ x_{n+1},\ \forall\ (x, x_{n+1}) \in \mathcal{F}_\varepsilon,$$

where $\mathcal{F}_\varepsilon$ is the set of all ε-essential feasible solutions of problem (8). Moreover:

$$\min\ x_{n+1} = \max\{f_1(x), f_2(x)\} + \varepsilon,\ \forall\ (x, x_{n+1}) \in \mathcal{F}_\varepsilon.$$

Therefore, since (x^r, x^r_{n+1}) and (x^s, x^s_{n+1}) are ε-essential feasible solutions of problem (8) with corresponding $f^{u_q} \in \mathcal{U}$, we obtain:

$$\max\{f_1(x^q), f_2(x^q)\} - \varepsilon \leqslant \max\{f_1(x^r), f_2(x^r)\}, \tag{15}$$

$$\max\{f_1(x^q), f_2(x^q)\} - \varepsilon \leqslant \max\{f_1(x^s), f_2(x^s)\}. \tag{16}$$

Now, suppose that $j = 1$ and $h = 2$. Then, $\max\{f_1(x^s), f_2(x^s)\} = f_1(x^s)$. Since otherwise, by (16), we obtain:

$$f_2(x^q) - \varepsilon \leqslant \max\{f_1(x^q), f_2(x^q)\} - \varepsilon \leqslant \max\{f_1(x^s), f_2(x^s)\} = f_2(x^s).$$

Thus, $f_2(x^q) - f_2(x^s) \leqslant \varepsilon$, which is a contradiction with (iii-2). From (iii-1) and (16), we have $\varepsilon \leqslant f_1(x^q) - \varepsilon - f_1(x^r)$, which implies that $\varepsilon \leqslant \max\{f_1(x^s), f_2(x^s)\} - f_1(x^r)$. Therefore, $\varepsilon \leqslant f_1(x^s) - f_1(x^r)$.

By similar arguments, $j = 2$ and $h = 1$ leads to $f_2(x^r) - f_2(x^s) \geqslant \varepsilon$. Consequently, in all cases (i)-(iii) we have either $|f_1(x^s) - f_1(x^r)| \geqslant \varepsilon$ or $|f_2(x^s) - f_2(x^r)| \geqslant \varepsilon$. Hence, $\sqrt{(f_1(x^s) - f_1(x^r))^2 + (f_2(x^s) - f_2(x^r))^2} \geqslant \varepsilon$. $\square$

4.1. Numerical examples

In this section two examples are solved by Algorithm 4.1. The first example considers a nonconvex MQCQP problem where the objective functions are convex but the feasible set is not convex. In the other example a convex problem is solved.

Example 4.1. Consider the following nonconvex biobjective quadratically constraint programming problem (taken from [3]):

$$\begin{array}{ll} \min & f(x) = (f_1(x) = x_1, f_2(x) = x_2) \\ s.t. & x_1^2 + x_2^2 \geqslant 1;\ x \in [(0,0),(2,2)]. \end{array} \tag{17}$$

It is easy to see that in problem (17) the efficient set coincides with the efficient frontier. In fact, the efficient set is $\{(x_1, x_2) \mid x_1^2 + x_2^2 = 1,\ x_1 \geqslant 0,\ x_2 \geqslant 0\}$. Figures 1 and 2 show the output of Algorithm 4.1 with $\varepsilon = 0.02$ (19 generated points) and $\varepsilon = 0.01$ (39 generated points), respectively. The minimum Euclidean distance among 19 and 39 points of $\mathcal{Y}_{out}$ are 0.042 and 0.018, respectively, which approve Proposition 4.1. Figures 3 and 4 show the approximated efficient frontiers by $\partial^+ P_{19}$ and $\partial^+ P_{39}$, respectively. Figures 3 and 4 illustrate that as ε decreases the upper boundary of polyblocks tend to the efficient frontier.

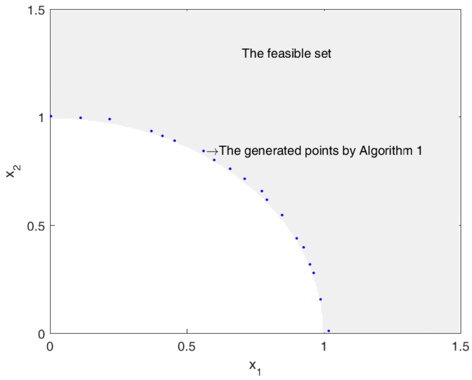

FIGURE 1. The set $\mathcal{X}_{out}$ for problem (17) with $\varepsilon = 0.02$.

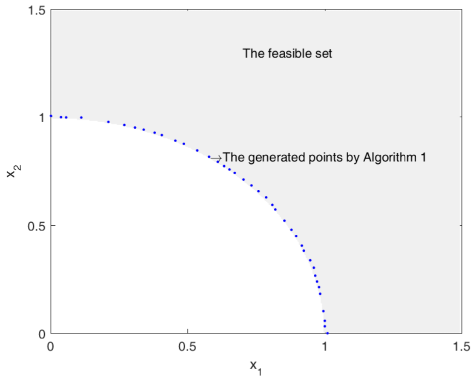

FIGURE 2. The set $\mathcal{X}_{out}$ for problem (17) with $\varepsilon = 0.01$.

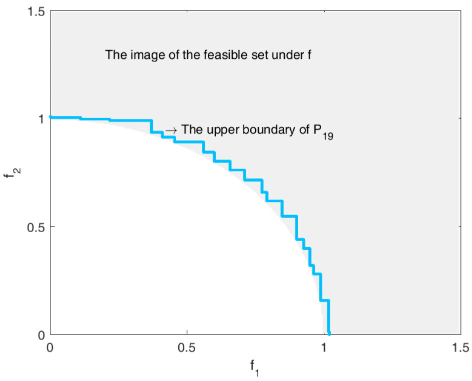

FIGURE 3. The approximated efficient frontier of problem (17) with $\varepsilon = 0.02$.

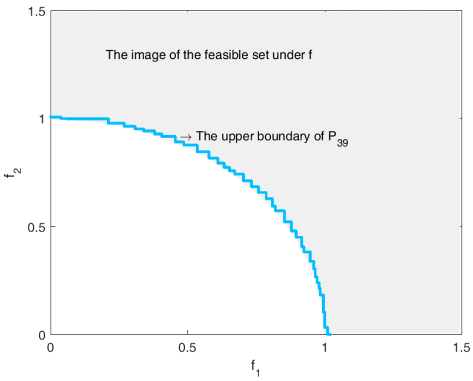

FIGURE 4. The approximated efficient frontier of problem (17) with $\varepsilon = 0.01$.

Example 4.2. Consider a biobjective quadratic programming problem as follows (taken from [4]):

$$\begin{array}{ll} \min & f(x) = (f_1(x) = 0.5(5x_1^2 + x_2^2), f_2(x) = 0.5(x_1^2 + 5x_2^2)) \\ s.t. & f_k(x) \geqslant 0,\ k = 3,4,5;\ x \in [(0.5,0.5),(2,2)], \end{array} \tag{18}$$

where $f_3(x) = 2x_1 + x_2 - 3,\ f_4(x) = x_1 + 2x_2 - 3,\ f_5(x) = 2x_1 - 3x_2 + 3.$

Since f_1, f_2 are convex functions and constraints are linear, problem (18) is a convex problem. In [4], the efficient set is obtained as two line segments between the points of $\{(0.75,1.5),(1,1)\}$ and $\{(1,1),(5/3,2/3)\}$. Figures 5 and 6 show the output of Algorithm 4.1 with $\varepsilon = 0.01$ in feasible and objective spaces, respectively. The minimum Euclidean distance among 28 points of $\mathcal{Y}_{out}$ is 0.082, which approves Proposition 4.1. Figure 7 depicts the approximated efficient frontier by $\partial^+ P_{28}$.

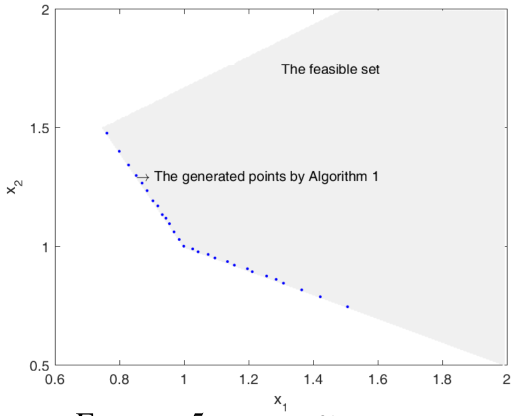

FIGURE 5. The set $\mathcal{X}_{out}$ for problem (18) with $\varepsilon = 0.01$.

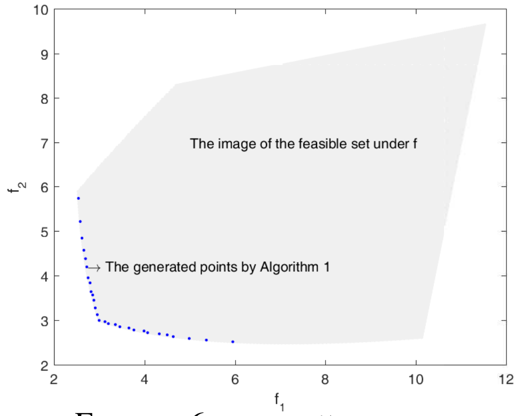

FIGURE 6. The set $\mathcal{Y}_{out}$ for problem (18) with $\varepsilon = 0.01$.

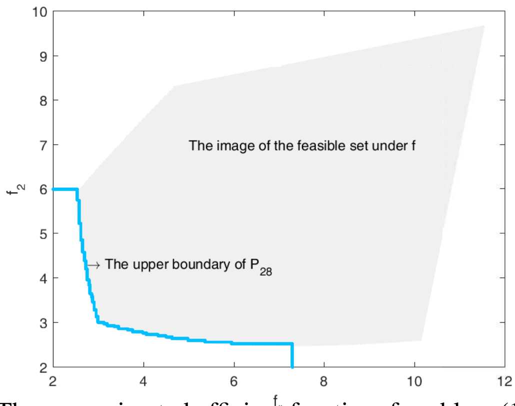

FIGURE 7. The approximated efficient frontier of problem (18) with $\varepsilon = 0.01$.

5. **Conclusions**

We introduced and studied the concept of essential epsilon-efficient (epsilon-weak efficient) solutions to multiobjective quadratic programming problems with quadratic constraints. It was done based on the concept of essential epsilon-optimal solutions in monotonic optimization. These solutions are stable under small perturbations of the objective functions and the constraints. To obtain essential epsilon-efficient solutions, we suggested a single objective quadratically constraint quadratic model which was obtained by combining the well-known constraint method and minmax approach of multiobjective programming. We proved that the suggested model is able to obtain all essential epsilon-efficient solutions. Then, we modified the SIT algorithm of [10] for solving the single objective programming model. The new algorithm terminates after a finite number of iterations and converges to a finite subset of essential epsilon-weak efficient or epsilon-efficient solutions. The output of the algorithm approximates the efficient set and the efficient frontier of the multiobjective problem in a reasonable way.

REFERENCES

[1] *A. Beato-Moreno, P. Luque-Calvo, R. Osuna-Gómez, A. Rufián-Lizana*, Finding efficient points in multiobjective quadratic programming with strictly convex quadratic functions, In: R. Caballero, F. Ruiz, R. E. Steuer (eds) Advances in Multiple Objective and Goal Programming, Springer, Berlin, 1997, pp. 365-373.

[2] *A. Beato-Moreno, P. Ruiz-Canales, P. L. Luque-Calvo, R. Blanquero-Bravo*, Multiobjective quadratic problem: Characterization of the efficient points, In: J. P. Crouzeix, J. E. Martinez-Legaz, M. Volle (eds) Generalized Convexity, Generalized Monotonicity, Kluwer Academic Publishers, London, 1998, pp. 425-438.

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

[4] *C. J. Goh and X. Q. Yang*, Analytic efficient solution set for multi-criteria quadratic programs, European J. Oper. Res., **92**(1996), No. 1, 166-181.

[5] *M. Hladik*, Interval convex quadratic programming problems in a general form, Cent. Eur. J. Oper. Res., **25**(2017), 725-737.

[6] *J. Linderoth*, A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs, Math. Prog., Ser. B, **103**(2005), No. 2, 251-282.

[7] *P. Loridan*, ε-solutions in vector minimization problems, J. Optim. Theory Appl., **43**(1984), No. 2, 265-276.

[8] *R. Oberdieck and E. N. Pistikopoulos*, Multi-objective optimization with convex quadratic cost functions: A multi-parametric programming approach, Comput. Chem. Eng., **85**(2016), 36-39.

[9] *H. Tuy*, Convex Analysis and Global Optimization, Springer, Berlin, 2016.

[10] *H. Tuy, N. T. Hoai-Phuong*, A robust algorithm for quadratic optimization under quadratic constraints, J. Glob. Optim., **37**(2007), No. 4, 557-569.