

Article

Efficient Solutions of Interval Programming Problems with Inexact Parameters and Second Order Cone Constraints

Ali Sadeghi ^{1,*}, Mansour Saraj ^{1,*} and Nezam Mahdavi Amiri ²¹ Department of Mathematics, Faculty of Mathematical Sciences and Computer, Shahid Chamran University of Ahvaz, 6135743136 Ahvaz, Iran² Faculty of Mathematical Sciences, Sharif University of Technology, 11365-11155 Tehran, Iran; nezamm@sharif.edu

* Correspondence: a-sadeghi@phdstu.scu.ac.ir (A.S.); msaraj@scu.ac.ir (M.S.)

Received: 1 October 2018; Accepted: 13 November 2018; Published: 20 November 2018



Abstract: In this article, a methodology is developed to solve an interval and a fractional interval programming problem by converting into a non-interval form for second order cone constraints, with the objective function and constraints being interval valued functions. We investigate the parametric and non-parametric forms of the interval valued functions along with their convexity properties. Two approaches are developed to obtain efficient and properly efficient solutions. Furthermore, the efficient solutions or Pareto optimal solutions of fractional and non-fractional programming problems over $R_+^n \cup \{0\}$ are also discussed. The main idea of the present article is to introduce a new concept for efficiency, called efficient space, caused by the lower and upper bounds of the respective intervals of the objective function which are shown in different figures. Finally, some numerical examples are worked through to illustrate the methodology and affirm the validity of the obtained results.

Keywords: second order cone (SOC); interval programming; inexact parameter; efficient solution

MSC: 90C46; 90C29

1. Introduction

We consider solving fractional interval programming problems with second order cone constraints with both the objective and constraints being interval valued functions. There are several approaches in the literature to solve such problems. Nonlinear interval optimization problems have been studied in several directions by many researchers during the past few decades [1–4]. Most considered models used quadratic programming problems with interval parameters. A methodology applied to interval valued convex quadratic programming problems by Bhurjee and Panda [1] which categorized how a solution of a general optimization problem can exist.

In the past few decades, fractional programming problems have also attracted the interest of many researchers. These problems have applications in the real physical world such as finance, production planning, electronic, etc. Fractional programming is being used for modelling real life problems involving one or more objective(s) such as actual cost/standard cost, inventory/sales and profit/cost. There are different algorithms to determine solutions of particular fractional programming problems. For example, Charnes and Cooper [5] converted a linear fractional program (LFP) to a linear program (LP) by a variable transformation technique. Tantawy [6] proposed an iterative method based on a conjugate gradient projection method. Dinkelbach [7] considered the same objective

over a convex feasible set. He also solved the same problem using a sequence of nonlinear convex programming problems.

On the other hand, we know that the convexity of the SOCP (Second Order Cone Constraints) problems are definite. The problems such as linear programs, convex quadratic programs and quadratically constrained convex quadratic programs can be easily converted to SOCP problems; for several other types of problems not falling into these three categories, see [8,9].

Lobo et al. [9] discussed several applications of SOCP in engineering. Nesterov and Nemirovski [10] and Lobo et al. [9,11] showed several kinds of problems formulated as SOCP problems, such as filter design, truss design, grasping force optimization in robotics, etc. In a pioneering paper, Nesterov and Nemirovski [10] applied the concept of self-concordant barrier to SOCP problems and found an iteration complexity of \sqrt{m} for problems with m second order cone inequalities.

Nestrov and Todd [12,13] were the first to investigate primal-dual interior methods for SOCP problems in which they investigate their outcome in the form of optimization over self-scaled cones having second order cones class as an especial case. Alizadeh & Goldfarb [8] considered and overviewed a large class of SOCP problems. They showed that many optimization problems such as linear programming (LP), quadratic programming (QP), quadratically constrained quadratic programming (QCQP) and other types of optimization problems could be rewritten as SOCP problems. They also demonstrated the method of converting different types of constraints into the form of SOC inequalities. Furthermore, they described an algebraic foundation of SOCs and showed how robust least squares and robust linear programming problems could be converted to SOCPs. The authors of [8] also discussed duality and complementary slackness for SOCP with notions of primal and dual non-degeneracy and strict complementarity along with logarithmic barrier function and primal-dual path following interior point methods (IPMs) for SOCPs.

Kim and Kojima [14] showed that semi-definite programming (SDP) and SOCP relaxation provide exact optimal solutions for a class of non-convex quadratic optimization problems. Moreover, SDP problems can in fact be formulated as SOCP problems and solved as such. There are a number of advantages for an SOCP problem. Adding a SOC constraint sometimes leads to negative decision variables, which usually does not occur with LP problems unless we let the variables be free in sign and usually get a much better solution, even though the dimension and convexity remain the same.

In our work here, we establish two results concerning efficient and properly efficient solutions of interval programming problems constrained with a second order cone constraint. The remainder of our work is organized as follows. In Section 2, the definitions and notations are provided. The interval valued functions in parametric and non-parametric forms along with their convexity properties are discussed in Section 3. In Section 4, we explain the existence of solutions for interval valued optimization problems and establish certain results concerning the efficient and properly efficient solutions of the interval problems involving SOC constraints. We also investigate the efficient solution for interval fractional and non-fractional programming problems in $R_+^n \cup \{0\}$. In Section 5, some numerical examples are worked through to verify the results on efficient and properly efficient solutions using MATLAB software environment. We conclude in Section 6.

2. Definitions and Notations

Let $I(R)$ be represented as the class of all closed intervals. A closed interval is shown by $M = [\underline{m}, \bar{m}]$, where \underline{m} and \bar{m} are respectively the lower and upper bounds of M . For closed intervals M , N , and $k \in R$, we have:

- (i) $M + N = \{m + n : m \in M, n \in N\} = [\underline{m} + \underline{n}, \bar{m} + \bar{n}]$,
- (ii) $-M = \{-m : m \in M\}$,
- (iii) $kM = \{km : m \in M\}$.

Definition 1. If $M = [\underline{m}, \bar{m}]$ and $N = [\underline{n}, \bar{n}]$ are bounded, real valued intervals, then the multiplication of M and N is defined to be

$$MN = \{\min\{\underline{m}\underline{n}, \underline{m}\bar{n}, \bar{m}\underline{n}, \bar{m}\bar{n}\}, \max\{\underline{m}\underline{n}, \underline{m}\bar{n}, \bar{m}\underline{n}, \bar{m}\bar{n}\}\}.$$

Let $F(x) = F(x_1, \dots, x_n)$ be a closed interval in R , for each $x \in R^n$. The interval-valued function F may be represented as $F(x) = [\underline{F}(x), \bar{F}(x)]$, where \underline{F} and \bar{F} are real-valued functions defined on R^n and satisfy $\underline{F}(x) \leq \bar{F}(x)$, for every $x \in R^n$. We say that the interval-valued function F is differentiable at $x_0 \in R^n$ if and only if the real-valued functions $\underline{F}(x)$ and $\bar{F}(x)$ are differentiable at x_0 . To know more, see [2].

Let $M = [\underline{m}, \bar{m}]$ and $N = [\underline{n}, \bar{n}]$ be two closed intervals in R and the relation " \preceq " be a partial ordering on $I(R)$. We write $M \preceq N$ if and only if $\underline{m} \leq \underline{n}$ and $\bar{m} \leq \bar{n}$. We also write $M \prec N$ if and only if $M \preceq N$ and $M \neq N$, meaning that M is inferior to N , or N is superior to M .

A second order cone is defined as follows:

$$Q_n = \{x = (x_1; \bar{x}) \in R^n : x_1 \geq \|\bar{x}\|\},$$

where $\|\cdot\|$ is the standard Euclidean norm, and n is the dimension of Q_n ; n is usually dropped from the subscript. We refer to inequality $x \succeq_Q 0$ as the second-order cone inequality.

For the cone Q , let

$$bdQ = \{x \in Q : x_1 = \|\bar{x}\| \text{ and } x \neq 0\}$$

denote the boundary of Q without the origin 0. In addition, let

$$intQ = \{x \in Q : x_1 > \|\bar{x}\|\}$$

denote the interior of Q .

We continue to present an overview of the SOCP problem. A standard form of an SOCP problem is given by

$$\begin{aligned} (P1) : \quad & \text{Min} \quad c_1^T x_1 + \dots + c_m^T x_m \\ & \text{s.t.} \quad A_1 x_1 + \dots + A_m x_m = b, \\ & \quad \quad x_i \in Q_{n_i} \quad i = 1, \dots, m, \end{aligned}$$

with its dual being

$$\begin{aligned} (D1) : \quad & \text{Max} \quad b^T y \\ & \text{s.t.} \quad A_i^T y + z_i = c_i, \quad i = 1, \dots, m, \\ & \quad \quad z_i \in Q_{n_i}, \quad i = 1, \dots, m, \end{aligned}$$

where $A_i \in R^{r \times n_i}$, $b \in R^r$, m is the number of blocks, $n = \sum_{i=1}^m n_i$ is the dimension of the problem, c_i , z_i , $x_i \in R^{n_i}$ and $y \in R^r$ [8].

We make the following assumptions regarding the primal-dual pair (P1) and (D1) [8].

Assumption 1. The matrix $A = (A_1, \dots, A_m)$ has r linearly independent rows.

Assumption 2. Due to the strict feasibility of both primal and dual, there exists a vector $x = (x_1; \dots; x_m)$ for every $x_i \succ_Q 0$, for $i = 1, \dots, m$, and dual-feasible y and $z = (z_1; \dots; z_m)$ such that $z_i \succ_Q 0$, for $i = 1, \dots, m$.

Remark 1. If problem (P1) has only one second order cone constraint, then the standard SOCP problem can be written as

$$\begin{aligned} (P2) : \quad & \text{Min} \quad c^T x \\ & \text{s.t.} \quad Ax = b, \\ & \quad \quad x \in Q_n, \end{aligned}$$

with its corresponding dual as

$$(D2) : \begin{array}{ll} \text{Max} & b^T y \\ \text{s.t.} & A^T y + z = c, \\ & z \in Q_n. \end{array}$$

Over time, we have seen a rapid development in improvement of software packages that can be applied to the problems such as SOCPs and mixed SOCP problems. SeDuMi [11] is a widely available package based on the Nesterov–Todd method.

3. Interval Valued Function

The definition of interval function in terms of functions of one and more intervals is given by see [2–4]. Walster and Hansen and Moore [2] defined an interval function as a function of one or more interval arguments onto an interval. Wu considered the interval valued function $F : R^n \rightarrow I(R)$ as $F(x) = [\underline{F}(x), \bar{F}(x)]$, where $\underline{F}, \bar{F} : R^n \rightarrow R$, $\underline{F}(x) \leq \bar{F}(x)$, $\forall x \in R^n$.

These functions may be defined on one or more interval arguments or maybe interval extension of real valued functions. The interval valued function in parametric form, introduced by [4], is as follows. For $m(t) \in M_v^k$, let $f_{m(t)} : R^n \rightarrow R$. Then, for a given interval vector M_v^k , an interval valued function $F_{M_v^k} : R^n \rightarrow I(R)$ is defined

$$F_{M_v^k}(x) = \{f_{m(t)}(x) | f_{m(t)} : R^n \rightarrow R, m(t) \in M_v^k\}.$$

For every fixed x , if $f_{m(t)}(x)$ is continuous in t , then $\min_{t \in [0,1]^k} f_{m(t)}(x)$ and $\max_{t \in [0,1]^k} f_{m(t)}(x)$ exist. Then,

$$F_{M_v^k}(x) = [\min_{t \in [0,1]^k} f_{m(t)}(x), \max_{t \in [0,1]^k} f_{m(t)}(x)].$$

If $f_{m(t)}(x)$ is linear in t , then $\min_{t \in [0,1]^k} f_{m(t)}(x)$ and $\max_{t \in [0,1]^k} f_{m(t)}(x)$ exist. If $f_{m(t)}(x)$ is monotonically increasing in t , then $F_{M_v^k} = [f_{m(0)}(x), f_{m(1)}(x)]$.

3.1. Interval Valued Convex Function

Interval valued convex function has the important property to guarantee the existence of solution of the interval optimization problem.

Definition 2. [4] An interval valued function $F_{M_v^k} : R^n \rightarrow I(R)$ is said to be convex regarding \preceq on a convex set $N \subseteq R^n$, if for every $x_1, x_2 \in N$ and $0 \leq \lambda \leq 1$, we have

$$F_{M_v^k}(\lambda x_1 + (1 - \lambda)x_2) \preceq \lambda F_{M_v^k}(x_1) \oplus (1 - \lambda)F_{M_v^k}(x_2).$$

Remark 2. From Definition 2, one may observe that $F_{M_v^k}$ is convex regarding \preceq , meaning that

$$f_{m(t)}(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f_{m(t)}(x_1) + (1 - \lambda)f_{m(t)}(x_2),$$

for all $t \in [0,1]^k$; we can realize that $F_{M_v^k}(x)$ is convex with respect to \preceq iff $f_{m(t)}(x)$ is a convex function on N , for every t .

3.2. Interval Valued Function in the Parametric Form

Let a binary operation on the set of real numbers be represented by $*$ $\in \{+, -, \cdot, /\}$. The binary operation \otimes between two intervals $M = [\underline{m}, \bar{m}]$ and $N = [\underline{n}, \bar{n}]$ in $I(R)$, denoted by $M \otimes N$, is the set $\{m * n | m \in M, n \in N\}$. In the case of division, M/N , it is to be noted that $0 \notin N$. An interval may be shown as a parameter form in several disciplines. Any point in M may be expressed as $m(t)$, where $m(t) = \underline{m} + t(\bar{m} - \underline{m})$. Throughout our work, we consider a specific parametric representation of an interval as $M = [\underline{m}, \bar{m}] = \{m(t) | t \in [0,1]\}$. The algebraic operations over classical intervals can

be represented as either lower or upper bounds of the intervals [1]. The parametric form of interval operations can be represented as follows:

$$M \otimes N = \{m(t_1) * n(t_2) | t_1, t_2 \in [0, 1]\}.$$

In addition, $M_v^k \in (I(R))^k$, $M_v^k = (M_1, M_2, \dots, M_k)^T$, can be expressed in terms of parameters as

$$M_v^k = \{m(t) | m(t) = (m_1(t_1), m_2(t_2), \dots, m_k(t_k))^T\},$$

where $m_j(t_j) \in M_j$, $m_j(0) = m_j^l$, $m_j(1) = m_j^u$, $t = (t_1, t_2, \dots, t_k)^T$, $0 \leq t_j \leq 1$.

4. Existence of Solutions

In this section, we consider the interval optimization problem as follows:

$$(P3): \quad \begin{aligned} & \text{minimize (min)} \quad F_{M_v^k}(x) \\ & \text{such that (s.t.)} \quad G_{jN_v^{m_j}}(x) \leq B_j, j = 1, 2, \dots, p, \end{aligned}$$

where $B_j \in I(R)$, and the interval valued functions $F_{M_v^k}, G_{jN_v^{m_j}} : R^n \rightarrow I(R)$ are the sets $F_{M_v^k}(x) = \{f_{m(t)}(x) | f_{m(t)} : R^n \rightarrow R, m(t) \in M_v^k\}$ and $G_{jN_v^{m_j}} = \{g_{jn(t'_j)}(x) | g_{jn(t'_j)}(x) : R^n \rightarrow R, n(t'_j) \in D_v^{m_j}\}$.

Discussion of the partial ordering is seen in Section 2, and the feasible space of (P3) is expressed as the following set:

$$\chi = \{x \in R^n | G_{jN_v^{m_j}}(x) \leq B_j, \quad j = 1, 2, \dots, p\}.$$

Definition 3. $x^* \in \chi$ is said to be an efficient solution of (P3), if there does not exist any $x \in \chi$ with $f_{m(t)}(x) \leq f_{m(t)}(x^*)$, $\forall t \in [0, 1]^k$, and $F_{M_v^k}(x) \neq F_{M_v^k}(x^*)$ [4].

Definition 4. $x^* \in \chi$ is said to be a properly efficient solution of (P3), if $x^* \in \chi$ is an efficient solution and there exists a real number $\mu > 0$ such that for some $t \in [0, 1]^k$ and all $x \in \chi$ with $f_{m(t)}(x) < f_{m(t)}(x^*)$, at least there is one $t' \in [0, 1]^k$, $t \neq t'$, exists with $f_{m(t')}(x) > f_{m(t')}(x^*)$ and

$$\frac{f_{m(t)}(x^*) - f_{m(t)}(x)}{f_{m(t')}(x) - f_{m(t')}(x^*)} \leq \mu.$$

Consider the following optimization problem with respect to a weight function $\omega : [0, 1]^k \rightarrow R_+ :$

$$(P4): \quad \min_{x \in \chi} \int_0^1 \int_0^1 \dots \int_0^1 \omega(t) f_{m(t)}(x) dt_1 dt_2 \dots dt_k,$$

where $\omega(t) = \omega(t_1, t_2, \dots, t_k)$. Here, t_1, t_2, \dots, t_k are mutually independent and each t_i varies from 0 to 1. Thus, $\int_0^1 \int_0^1 \dots \int_0^1 \omega(t) f_{m(t)}(x) dt_1 dt_2 \dots dt_k$ is a function of x only, say $h(x)$. Thus, (P4) as $\min_{x \in \chi} h(x)$ is a general nonlinear programming problem free from interval uncertainty. The problem can be solved by a nonlinear programming technique. The following theorem establishes the relationship between the solution of the transformed problem (P4) and the original problem (P3) [4].

Theorem 1. If $x^* \in \chi$ is an optimal solution of (P4), then x^* is a properly efficient solution of (P3).

Proof. See [4]. \square

4.1. Alternative Method for Solving an Interval Problem with an SOC Constraint

Here, we consider a problem by applying the order relation “ \preceq ” for the constraints as follows:

$$(P5) : \begin{aligned} \text{Min} \quad & F(x) = [\underline{F}(x), \bar{F}(x)] \\ \text{s.t.} \quad & G_i(x) \preceq [b_i^L, b_i^U], \quad i = 1, 2, \dots, m, \\ & x \in Q_n, \end{aligned}$$

where the $G_i(x) = [\underline{G}_i(x), \bar{G}_i(x)]$ are interval-valued constraints. A point $x = (x_1, \dots, x_n)$ is a feasible solution of problem (P5), if $G_i(x) \preceq [b_i^L, b_i^U]$, for $i = 1, \dots, m$, or equivalently, $\underline{G}_i(x) \leq b_i^L$ and $\bar{G}_i(x) \leq b_i^U$, for $i = 1, 2, \dots, m$. Then, the auxiliary interval-valued optimization problem (P5) can be rewritten as follows:

$$(P6) : \begin{aligned} \text{Min} \quad & F(x) = [\underline{F}(x), \bar{F}(x)] \\ \text{s.t.} \quad & \underline{G}_i(x) \leq b_i^L, \quad i = 1, 2, \dots, m, \\ & \bar{G}_i(x) \leq b_i^U, \quad i = 1, 2, \dots, m, \\ & x \in Q_n. \end{aligned}$$

It is obvious that the feasible regions of problems (P5) and (P6) are the same and, since their objective function is also the same, we have the same solution for both problems. The interval property of problem (P6) incurs a very important concept called efficient space which is a new concept from the optimization point of view.

Therefore, the interval-valued optimization problem (P6) is easily converted to a common form as below:

$$(P7) : \begin{aligned} \text{Min} \quad & F(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, 2, \dots, m, \\ & h_i(x) \leq 0, \quad i = 1, 2, \dots, m, \\ & x \in Q_n, \end{aligned}$$

where $F : R^n \rightarrow I(R)$ is an interval-valued function, and $g_i : R^n \rightarrow R$ and $h_i : R^n \rightarrow R, i = 1, \dots, m$, are real-valued functions. Let $M = [\underline{m}, \bar{m}]$ and $N = [\underline{n}, \bar{n}]$ be two closed intervals in R . We write $M \preceq N$ if and only if $\underline{m} \leq \underline{n}$ and $\bar{m} \leq \bar{n}$, and we write $M \prec N$ iff $N \preceq M$ and $M \neq N$. Equivalently, $M \prec N$ iff

$$\left\{ \begin{array}{l} \underline{m} < \underline{n} \\ \bar{m} \leq \bar{n} \end{array} \right\} \quad \text{or} \quad \left\{ \begin{array}{l} \underline{m} \leq \underline{n} \\ \bar{m} < \bar{n} \end{array} \right\} \quad \text{or} \quad \left\{ \begin{array}{l} \underline{m} < \underline{n} \\ \bar{m} < \bar{n} \end{array} \right\} \quad (1)$$

We need to interpret the meaning of minimization for (P7). Since \preceq is a partial ordering, not a total ordering on $I(R)$, we may follow the similar solution concept (efficient solution) used in multi-objective programming problem to interpret the meaning of minimization in the primal problem (P7). For the minimization problem (P7), we say that the feasible solution \bar{x} is better than (dominates) the feasible solution x^* , if $F(\bar{x}) < F(x^*)$. Therefore, we propose the following definition.

Definition 5. Let x^* be a feasible solution of the primal problem (P7). We say that x^* is an efficient solution of (P7) if there exists no $\bar{x} \in X$ such that $F(\bar{x}) \prec F(x^*)$. In this case, $F(x^*)$ is called the efficient objective value of F .

We denote the set of all efficient objective values of problem (P7) by $\text{Min}(F, X)$. More precisely, we write

$$\text{Min}(F, X) = \{F(x^*)\},$$

where x^* is an efficient solution of (P7). Let m be a real number. Then, it can be represented as an interval $[m, m]$. Let $M = [\underline{m}, \bar{m}]$ be a closed interval. By $M + m$, we mean $M + [m, m] = [\underline{m} + m, \bar{m} + m]$.

Now, consider the following optimization problem:

$$(P8): \begin{aligned} \text{Min} \quad & f(x) = \underline{F}(x) + \bar{F}(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, 2, \dots, m, \\ & h_i(x) \leq 0, \quad i = 1, 2, \dots, m, \\ & x \in Q_n. \end{aligned}$$

Obviously, if x^* is an optimal solution of problem (P8), then x^* is a nondominated solution of problem (P7); see [4]. We may now focus here on the two results given as Theorems 2 and 3 below, by which the optimal solutions of the problems (P9) and (P10) are indeed the efficient solutions of the problem (P7):

$$(P9): \begin{aligned} \text{Min} \quad & f(x) = \underline{F}(x)\bar{F}(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & h_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & \underline{F}(x) > 0, \\ & x \in Q_n, \end{aligned}$$

$$(P10): \begin{aligned} \text{Min} \quad & f(x) = \alpha \underline{F}(x) + \beta \bar{F}(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & h_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & x \in Q_n, \end{aligned}$$

where α and β are positive scalars.

Theorem 2. *If x^* is an optimal solution of problem (P9), then x^* is an efficient solution of problem (P7).*

Proof. We see that problems (P9) and (P7) have the same feasible region. Suppose that x^* is not an efficient solution. Then, there exists a feasible solution x such that $F(x) \prec F(x^*)$. From inequation (1), it means that

$$\begin{cases} \underline{F}(x) < \underline{F}(x^*) \\ \bar{F}(x) \leq \bar{F}(x^*) \end{cases}, \quad \text{or} \quad \begin{cases} \underline{F}(x) \leq \underline{F}(x^*) \\ \bar{F}(x) < \bar{F}(x^*) \end{cases}, \quad \text{or} \quad \begin{cases} \underline{F}(x) < \underline{F}(x^*) \\ \bar{F}(x) < \bar{F}(x^*) \end{cases}.$$

This also shows that $f(x) < f(x^*)$, which contradicts the fact that x^* is an optimal solution of problem (P7). This completes the proof. \square

Theorem 3. *If x^* is an optimal solution of problem (P10), then x^* is an efficient solution of problem (P7).*

Proof. We see that problems (P10) and (P7) have the same feasible region. Suppose that x^* is not an efficient solution. Then, there exists a feasible solution x such that $F(x) \prec F(x^*)$. From (1), it means that

$$\begin{cases} \underline{F}(x) < \underline{F}(x^*) \\ \bar{F}(x) \leq \bar{F}(x^*) \end{cases} \quad \text{or} \quad \begin{cases} \underline{F}(x) \leq \underline{F}(x^*) \\ \bar{F}(x) < \bar{F}(x^*) \end{cases} \quad \text{or} \quad \begin{cases} \underline{F}(x) < \underline{F}(x^*) \\ \bar{F}(x) < \bar{F}(x^*) \end{cases}.$$

Then, we have

$$\begin{cases} \alpha \underline{F}(x) < \alpha \underline{F}(x^*) \\ \beta \bar{F}(x) \leq \beta \bar{F}(x^*) \end{cases}, \quad \text{or} \quad \begin{cases} \alpha \underline{F}(x) \leq \alpha \underline{F}(x^*) \\ \beta \bar{F}(x) < \beta \bar{F}(x^*) \end{cases}, \quad \text{or} \quad \begin{cases} \alpha \underline{F}(x) < \alpha \underline{F}(x^*) \\ \beta \bar{F}(x) < \beta \bar{F}(x^*) \end{cases}.$$

This also shows that $f(x) < f(x^*)$, which contradicts the fact that x^* is an optimal solution of problem (P7). This completes the proof. \square

4.2. Interval Valued Convex Linear Programming Problem with SOC Constraint

An interval valued optimization problem (P3) is said to be an interval valued convex programming problem, if $F_{M_v^k}$ and $G_{jN_v^{m_j}}$ are convex functions with respect to \preceq .

If (P3) is an interval valued convex programming problem, then P4 is a convex programming problem. A general interval linear programming problem (P3) has the following form:

$$(P11) : \begin{aligned} \text{Min} \quad & M_v^k \diamond x \\ \text{s.t.} \quad & A_m x \preceq B_v^p, \\ & x \in Q_n, \end{aligned}$$

where $M_v^k \in (I(R))^n$, $B_v^p \in (I(R))^n$ and $A_m = (A_{ij})_{p \times n}$ is an interval valued matrix with $A_{ij} = [a_{ij}^l, a_{ij}^u]$ and $M_v^k \diamond x = \sum_{j=1}^k M_j x_j$, the product of a real vector $x \in R^k$ and an interval vector $M_v^k \in (I(R))^k$.

5. Numerical Results

In this section, we consider three examples having various dimensions to illustrate the obtained results. In order to solve problems using both theorems, we use the **fmincon** command of Matlab. Notations are given in Table 1 and the results are summarized in Tables 2–4 and the corresponding diagrams. We generate problems with different dimensions and report the required CPU times. All computations are performed on MATLAB R2015a (8.5) using a laptop with Intel(R) Core i3 CPU 2.53 GHz and 5.00 GB of RAM.

We present computational results on Examples 1 and 2 to compare the results due to Theorems 2 and 3. To compare the obtained results for the numerical examples, we use different diagrams and tables to show the advantages of the given Theorems 2 and 3 by showing that any solution of the problem (P9) or (P10) is an efficient solution of the problem (P7). In addition, the efficient space for different pairs of (α, β) is also shown, with the generated nonzero elements α taken randomly in the interval $(0, 1)$ and elements of vector β given in $(0, n)$ with step length 1.

Example 1. Consider the interval programming problem with SOC constraint as follows:

$$(P12) : \begin{aligned} \text{Min} \quad & \left[\frac{7x_1 + x_2}{3x_1 + 4x_2 + 36}, \frac{7x_1 + x_2}{3x_1 + 4x_2 + 12} \right] \\ \text{s.t.} \quad & x_1 + x_2 \leq 7, \\ & 4x_1 - 9x_2 \leq 3, \\ & x_1 + 2x_2 \geq 1.5, \\ & x \in Q_2. \end{aligned}$$

Example 2. Consider the interval programming problem with SOC constraint as follows:

$$(P13) : \begin{aligned} \text{Min} \quad & \left[\frac{x_1 + 3x_2 + 1.5x_3 + 3.5}{x_1 + x_2 + 2x_3 + 1}, \frac{2x_1 + 7x_2 + 2.5x_3 + 4}{.5x_1 + .75x_2 + 7/8x_3 + .5} \right] \\ \text{s.t.} \quad & x_1 + 2x_2 - x_3 \leq 6, \\ & -2x_1 + 3x_2 + x_3 \leq 8, \\ & x_1 + x_2 + x_3 \leq 13, \\ & x \in Q_3. \end{aligned}$$

Example 3. Consider the interval programming problem with SOC constraint as follows:

$$(P14) : \begin{aligned} \text{Min} \quad & [-10, -6]x_1 + [2, 3]x_2 \\ \text{s.t.} \quad & [1, 2]x_1 + 3x_2 \preceq [1, 10], \\ & [-2, 8]x_1 + [4, 6]x_2 \preceq [4, 6], \\ & x \in Q_2. \end{aligned}$$

Table 1. Notations.

n	Number of pairs corresponding to α and β related to Theorem 3
Exwsoc1	Example 1 without SOC constraint
Exsoc1	Example 1 with SOC constraint
Exwsoc2	Example 2 without SOC constraint
Exsoc2	Example 2 with SOC constraint
CPU	CPU time in seconds
CPU.ratio	The ratio of CPU times consumed by problem due to SOC constraint over without SOC constraint

Table 2. Objective function values using Theorems 2 and 3.

	Applying Theorem 2	Applying Theorem 3
Example 1	−12	−0.003973
Example 2	4.0454	171.31

Table 3. CPU times corresponding to Example 1.

n	CPU Time for Exwsoc1	CPU Time for Exsoc1	cpu.ratio
10	0.975	0.764	1.2762
50	4.013	2.760	1.454
100	7.353	5.320	1.3821
150	11.079	8.137	1.36155
200	14.674	10.543	1.392
250	18.435	12.890	1.43018
300	21.828	15.439	1.413822
350	25.269	18.388	1.37421
400	28.601	20.714	1.38076
450	32.044	22.860	1.40174
500	36.385	25.750	1.413
600	42.514	30.776	1.38140
700	49.899	35.852	1.39180
800	56.713	40.976	1.38405
900	64.219	45.873	1.39993
1000	71.645	51.256	1.398

Table 4. CPU times corresponding to Example 2.

n	CPU Time for Exwsoc2	CPU Time for Exsoc2	cpu.ratio
10	1.621	1.225	1.32327
50	5.076	4.814	1.05442
100	9.608	9.154	1.0496
150	14.424	13.018	1.10800
200	18.695	17.185	1.02804
250	22.747	21.714	1.04752
300	26.842	25.866	1.03773
350	32.512	31.570	1.02983
400	36.399	33.455	1.0880
450	40.249	38.412	1.04782
500	45.471	42.775	1.06303
600	54.623	52.015	1.05014
700	62.414	59.653	1.04628
800	71.676	68.400	1.04790
900	79.980	79.956	1.00030
1000	90.243	82.616	1.09231

By our described methodology, we get a properly efficient solution. Here, $t = (t_1, t_2)^T, t_j \in [0, 1]$.
 $f_{m(t)}(x) = (-10 + 4t_1)x_1 + (2 + t_2)x_2$ and $\chi = \{(x_1, x_2) | 2x_1 + 3x_2 \leq 1, 8x_1 + 6x_2 \leq 4\}$.

For some $\omega(t) : R^2 \rightarrow R$, the corresponding problem (P4) becomes:

$$(P15) : \text{Min}_{\chi} \int_0^1 \int_0^1 \omega(t) [(-10 + 4t_1)x_1 + (2 + t_2)] x_2 dt_1 dt_2.$$

This problem is an SOCP problem and can be solved by an interior point method.

If $\omega(t) = 2t_1$, then the properly efficient solution is $x^* = [0.2, 0.2]$ and the optimal interval is $[-1.6, -0.6]$ and efficient solution obtained by Theorem 2 is $x^* = [0.2, 0.2]$.

Table 2 shows the objective function values obtained using Theorems 2 and 3.

We see the results for various values of n in Table 3. The results for different values of n are summarized in Tables 3 and 4. We observe that the CPU times for problems with SOC constraints is lower than the ones for problems without SOC constraint.

Efficient spaces for Example 3 with SOC constraint for different n 's is given in the Figures 1–3 and without SOC constraint for different n 's is illustrated in the Figures 4–6, where efficient space is a new concept in efficiency literature.

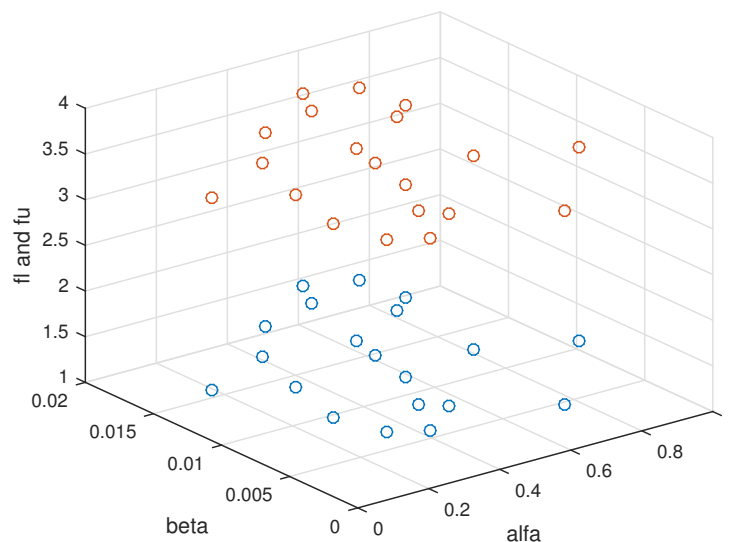


Figure 1. Efficient space for Example 3 with SOC constraint using $n = 20$.

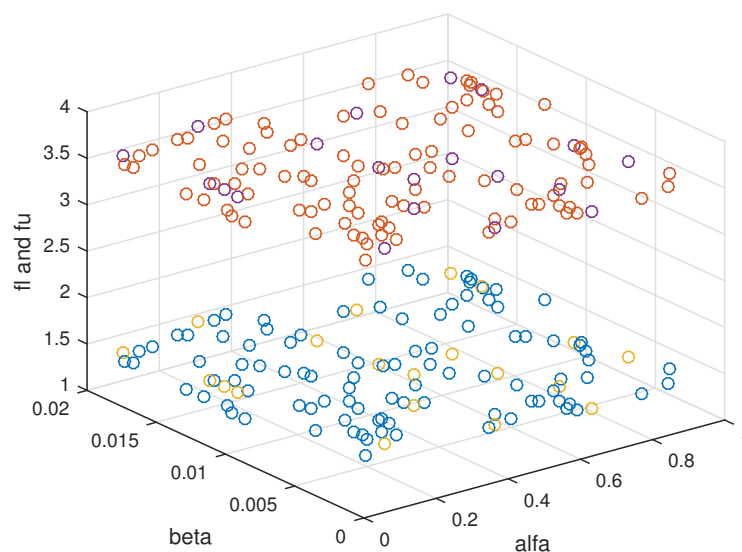


Figure 2. Efficient space for Example 3 with SOC constraint using $n = 100$.

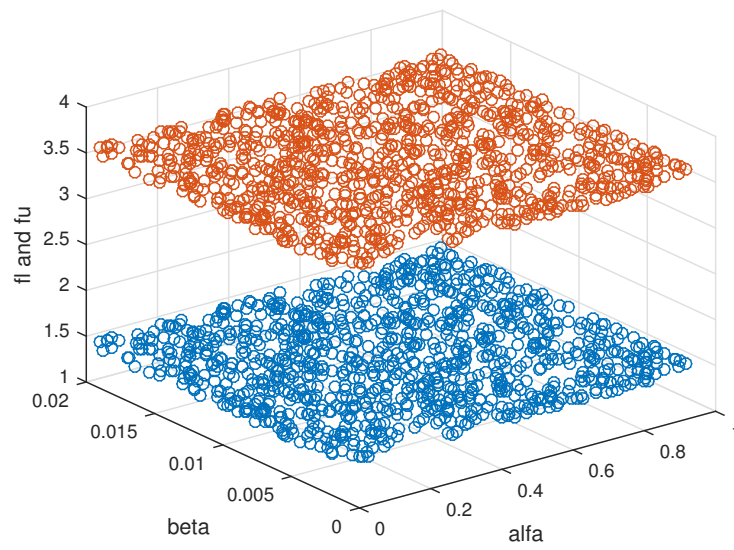


Figure 3. Efficient space for Example 3 with SOC constraint using $n = 1000$.

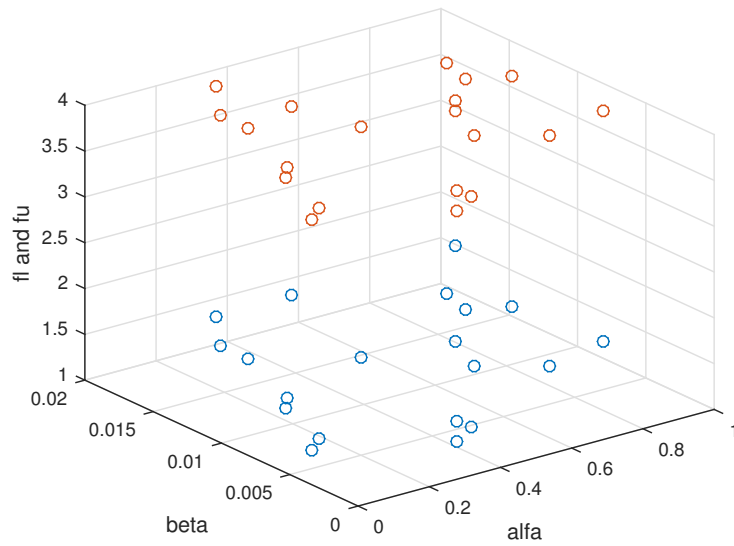


Figure 4. Efficient space for Example 3 without SOC constraint using $n = 20$.

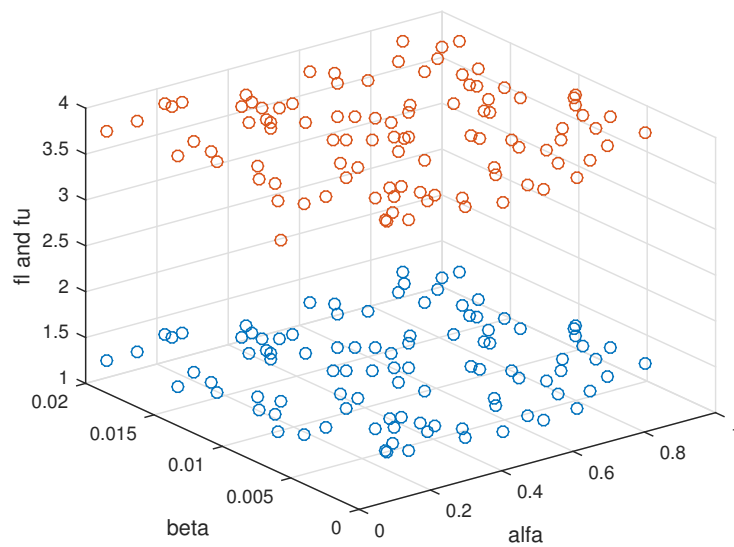


Figure 5. Efficient space for Example 3 without SOC constraint using $n = 100$.

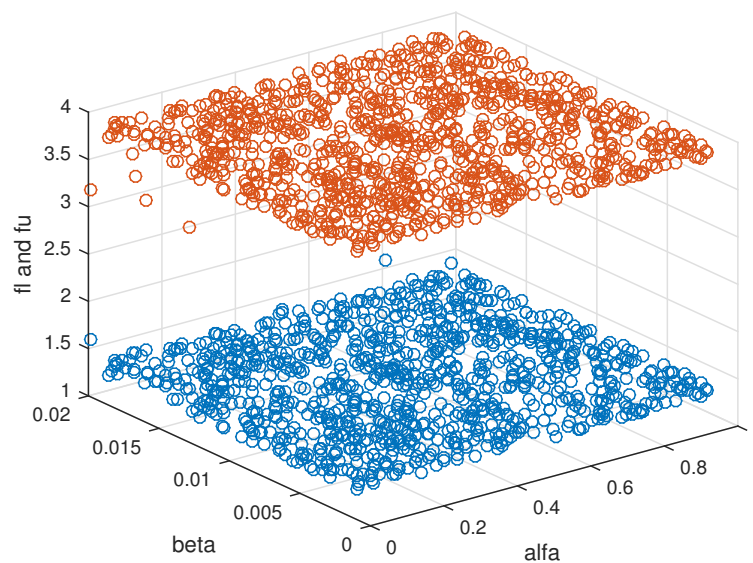


Figure 6. Efficient space for Example 3 without SOC constraint using $n = 1000$.

6. Conclusions

A very important concept of SOCP is investigated here. We have paid our attention to consider the interval fractional programming problem with second order cone constraints. To solve such problems, we established two important results concerning the efficient and properly efficient solutions of the second-order cone constrained interval programming problems. In addition and furthermore, a new notion of efficiency called efficient space was proposed due to interval form of the objective function and the corresponding obtained results were summarized in Tables 3 and 4 and simultaneously in Figures 1–6 with efficient spaces related to upper and lower bound properties of the interval problem. To illustrate the performance of our methodology, a few numerical examples were worked through to represent the importance of the study. The numerical results showed that the CPU times needed for solving problems with second order cone constraints are less than the ones for the problem without second-order cone constraints, which is a very important issue.

Author Contributions: A.S., M.S. and N.M.A. contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

Funding: This research received no external funding

Acknowledgments: The authors are thankful to the Shahid Chamran University of Ahvaz and Sharif University of Technology for making this joint work possible and Shahid Chamran University of Ahvaz for the financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bhurjee, A.K.; Panda, G. Efficient solution of interval optimization problem. *Math. Meth. Oper. Res.* **2012**, *76*, 273–288. [\[CrossRef\]](#)
2. Moore, R.E. *Interval Analysis*; Prentice-Hall: Upper Saddle River, NJ, USA, 1966.
3. Walster, G.W.; Hansen, E. *Global Optimization Using Interval Analysis*; Marcel Dekker, Inc.: New York, NY, USA, 2004.
4. Wu, H. On interval-valued nonlinear programming problems. *J. Math. Anal. Appl.* **2008**, *338*, 299–316. [\[CrossRef\]](#)
5. Charnes, A.; Cooper, W.W. Programs with linear fractional functions. *Nav. Res. Logist. Q.* **1962**, *9*, 181–196. [\[CrossRef\]](#)
6. Tantawy, S.F. A new procedure for solving linear fractional programming problems. *J. Math. Comput. Model.* **2008**, *48*, 969–973. [\[CrossRef\]](#)
7. Dinkelbach, W. On nonlinear fractional programming. *Manag. Sci.* **1967**, *13*, 492–498. [\[CrossRef\]](#)

8. Alizadeh, F.; Goldfarb, D. Second-order cone programming. *Math. Program. Ser. B* **2003**, *95*, 35–51. [[CrossRef](#)]
9. Lobo, M.S.; Vandenberghe, L.; Boyd, L.; Lebret, H. Applications of second order cone programming. *Linear Algebra Appl.* **1998**, *284*, 193–228. [[CrossRef](#)]
10. Nesterov, Y.; Nemirovski, A. *Interior Point Polynomial Methods in Convex Programming: Theory and Applications*; Society for Industrial and Applied Mathematics (SIAM): Philadelphia, PA, USA, 1994.
11. Lobo, M.S.; Vandenberghe, L.; Boyd, S. *SOCP: Software for Second-Order Cone Programming*; Information Systems Laboratory, Stanford University: Standford, CA, USA, 1997.
12. Nesterov, Y.E.; Todd, M.J. Primal-dual interior-point methods for self-scaled cones. *SIAM J. Optim.* **1998**, *8*, 324–364. [[CrossRef](#)]
13. Nesterov, Y.E.; Todd, M.J. Self-scaled barriers and interior-point methods for convex programming. *Math. Oper. Res.* **1997**, *22*, 1–42. [[CrossRef](#)]
14. Kim, S.; Kojima, M. Exact solutions of some nonconvex quadratic optimization problems via SDP and SOCP relaxations. *Comput. Optim. Appl.* **2003**, *26*, 143–154. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).