Journal of the Operations Research
Society of Japan
Vol. 35, No. 3, September 1992

A PARAMETRIC SUCCESSIVE UNDERESTIMATION METHOD FOR CONVEX PROGRAMMING PROBLEMS WITH AN ADDITIONAL CONVEX MULTIPLICATIVE CONSTRAINT

Takahito Kuno
University of Tsukuba

Hiroshi Konno
Tokyo Institute of Technology

Yoshitsugu Yamamoto
University of Tsukuba

(Received July 4, 1991)

Abstract This paper addresses itself to an algorithm for a convex minimization problem with an additional convex multiplicative constraint. A convex multiplicative constraint is such that a product of two convex functions is less than or equal to some constant. It is shown that this nonconvex problem can be solved by solving a sequence of convex programming problems. The basic idea of this algorithm is to embed the original problem into a problem in a higher dimensional space and to apply a parametric programming technique. A branch-and-bound algorithm is proposed for obtaining an ϵ-optimal solution in finitely many iterations. Computational results indicate that this algorithm is efficient for linear programs with an additional linear multiplicative constraint.

1 Introduction

The purpose of this paper is to propose an efficient method for solving a special class of nonconvex minimization problem:

$$(1.1) \quad \left| \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X, \\ & f_1(x) \cdot f_2(x) \leq b, \end{array} \right.$$

where $f_i : R^n \rightarrow R^1$, $i = 0, 1, 2$ are convex functions, b is a positive constant and X is a compact convex set of R^n. We assume that both f_1 and f_2 are positive valued on X.

In their recent article, Thach and Burkard [21] studied a special case of (1.1) in which both f_1 and f_2 are linear. They converted the problem into a two-dimensional concave minimization problem and proposed an outer approximation method. There are several other researches on convex minimization problems with an additional nonconvex constraint [7,8,19,22].

In this paper, we propose a branch-and-bound algorithm for obtaining an ϵ-optimal solution of (1.1). Contrary to the method of Thach and Burkard, we reduce (1.1) to an $(n+1)$-dimensional master problem, which can be solved by solving a sequence of convex (rather than concave) minimization problems. This algorithm is closely related to the successive underestimation method developed in [15] for convex multiplicative programming problems:

$$(1.2) \quad \left| \begin{array}{ll} \text{minimize} & f_1(x) \cdot f_2(x) \\ \text{subject to} & x \in X. \end{array} \right.$$

The product of two convex functions appears in many applications such as microeconomics [6], VLSI chip design [16], bond portfolio optimization [11] and so forth (see [17]).

This type of nonconvex problems is also dealt with in [1,2,12,13,14,20]. Readers are referred to the recent books [10,18] for the state-of-the-art of nonconvex optimization as well.

In Section 2 we embed the original problem (1.1) into its master problem by introducing an additional variable. This reformulation enables us to apply a parametric programming approach. Section 3 is devoted to the construction of an algorithm for obtaining an ϵ-optimal solution of (1.1). We define an auxiliary problem, which gives a lower bound of the optimal value of the master problem. By exploiting this property, we apply a branch-and-bound algorithm to the master problem. Results of computational experiments of this algorithm are presented in Section 4.

2 Problem with a Convex Multiplicative Constraint

Let us consider a convex minimization problem with an additional convex multiplicative constraint:

$$\text{P}\left|\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X, \\ & f_1(x)\cdot f_2(x) \leq 1, \end{array}\right.$$

where $f_i : R^n \to R^1$, $i = 0, 1, 2$ are convex functions and X is a nonempty compact convex set of R^n. Denote

$$(2.1) \qquad Y = \{x \in R^n \mid f_1(x)\cdot f_2(x) \leq 1\}.$$

The product of two convex functions need not be convex [12,15], whence multiple local minima may exist in the feasible region $X \cap Y$. The values of both f_1 and f_2 are assumed to be positive on X, i.e.,

$$(2.2) \qquad f_1(x) > 0, \quad f_2(x) > 0, \quad \forall x \in X.$$

By removing the multiplicative constraint $f_1(x)\cdot f_2(x) \leq 1$ from P, we obtain a usual convex minimization problem:

$$\overline{\text{P}}\left|\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X. \end{array}\right.$$

This problem $\overline{\text{P}}$ has an optimal solution $\overline{x}$, since X is compact. If $\overline{x} \in Y$, then $\overline{x}$ is obviously optimal for the original problem P. Therefore we assume without loss of generality that

$$(2.3) \qquad \overline{x} \notin Y.$$

2.1 Master problem

Let us introduce an additional variable ξ and define the master problem of P:

$$\text{MP}\left|\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X, \\ & f_1(x) \leq 1/\xi, \quad f_2(x) \leq \xi, \\ & \xi > 0. \end{array}\right.$$

For p, $q \in R^1$ we define a convex set:

(2.4) $Z(p,q) = \{x \in R^n \mid f_1(x) \le p, \quad f_2(x) \le q\}.$

Lemma 2.1

(2.5) $X \cap Y = X \cap [\cup_{\xi>0} Z(1/\xi, \xi)].$

Proof: Let $x \in X \cap [\cup_{\xi>0} Z(1/\xi,\xi)]$. Then there exists $\xi_1 > 0$ such that $x \in X \cap Z(1/\xi_1, \xi_1)$. Thus

$$f_1(x) \cdot f_2(x) \le (1/\xi_1) \cdot \xi_1 = 1,$$

which implies $x \in X \cap Y$. Conversely, for $x \in X \cap Y$ let $\xi_2 = f_2(x)$. Then

$$f_1(x) \le 1/f_2(x) = 1/\xi_2,$$

because we assumed that $f_2(x) > 0$ for any $x \in X$. Hence $x \in X \cap [\cup_{\xi>0} Z(1/\xi,\xi)]$. □

Theorem 2.2 *Let (x^*, ξ^*) be an optimal solution of the master problem* MP. *Then x^* is optimal for* P.

Proof: Obvious from Lemma 2.1. □

Let us consider the subproblem of MP:

$$\text{MP}(\xi) \left| \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X \cap Z(1/\xi, \xi), \end{array} \right.$$

in which ξ has some fixed positive value. Since MP(ξ) is a convex minimization problem for all $\xi > 0$, we can obtain its optimal solution $x^*(\xi)$ by any standard method if $X \cap Z(1/\xi, \xi)$ is nonempty. Let us define

(2.6) $$h(\xi) = \begin{cases} f_0(x^*(\xi)), & \text{if} \quad X \cap Z(1/\xi,\xi) \neq \emptyset, \\ +\infty, & \text{otherwise.} \end{cases}$$

Then to solve the original problem P we need to locate a global minimum point ξ^* of $h(\xi)$ over $\xi > 0$. The solution $x^*(\xi^*)$ is guaranteed to be globally optimal for P.

2.2 ϵ-optimal solution

Let us consider the following relaxation of P:

$$\text{P}(\epsilon) \left| \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X, \\ & f_1(x) \cdot f_2(x) \le 1 + \epsilon, \end{array} \right.$$

where ϵ is a positive tolerance. Denote

(2.7) $Y(\epsilon) = \{x \in R^n \mid f_1(x) \cdot f_2(x) \le 1 + \epsilon\}.$

We say that x is an *ϵ-feasible solution* of P if $x \in X \cap Y(\epsilon)$. Let us denote by $\overline{x}(\epsilon)$ an optimal solution of P(ϵ) if it exists, and define

(2.8) $$g(\epsilon) = \begin{cases} f_0(\overline{x}(\epsilon)), & \text{if} \quad X \cap Y(\epsilon) \neq \emptyset, \\ +\infty, & \text{otherwise.} \end{cases}$$

Also, let

(2.9) $\quad S(\epsilon) = \{x \in R^n \mid x \in X \cap Y(\epsilon),\ f_0(x) \leq g(\epsilon)\}.$

Lemma 2.3

(i) *$g(\epsilon)$ is monotonically nonincreasing.*

(ii) $\lim_{\epsilon \to 0+} g(\epsilon) = g(0)$.

(iii) *Let $x(\epsilon)$ be a point of $S(\epsilon)$ and let x^* be a cluster point of $\{x(\epsilon) \mid \epsilon > 0\}$. Then x^* is a globally optimal solution of* P.

Proof: (i) It follows from that $Y(\epsilon_1) \subset Y(\epsilon_2)$ if $\epsilon_1 \leq \epsilon_2$.
(ii) Since the point-to-set mapping $X \cap Y(\epsilon)$ is uniformly compact and upper semicontinuous, $g(\epsilon)$ is a lower semicontinuous function (see [5,9]), i.e., $\liminf_{\epsilon \to \bar{\epsilon}} g(\epsilon) \geq g(\bar{\epsilon})$ for any $\bar{\epsilon}$. However, clearly $g(\epsilon) \leq g(0)$ for any $\epsilon \geq 0$. Thus

$$\limsup_{\epsilon \to 0+} g(\epsilon) \leq g(0) \leq \liminf_{\epsilon \to 0+} g(\epsilon).$$

(iii) It is obvious that x^* is a feasible solution of P because $f_1 \cdot f_2$ is continuous and X is closed. In addition, by choosing a subsequence $\{x(\epsilon')\}$ of $\{x(\epsilon) \mid \epsilon > 0\}$, we see from (ii) that

$$f_0(x^*) = \lim_{\epsilon' \to 0+} f_0(x(\epsilon')) = \lim_{\epsilon' \to 0+} g(\epsilon') = g(0).$$

Hence $x^* \in S(0)$. □

In the sense stated in Lemma 2.3, $\bar{x}(\epsilon)$ is an approximate solution of P. We refer to x as an *ϵ-optimal solution* of P if $x \in X \cap Y(\epsilon)$ and $f_0(x) \leq g(0)$.

3 Successive Underestimation Method

3.1 Underestimator

Let

(3.1) $\quad x^j = \operatorname{argmin}\{f_j(x) \mid x \in X\}, \quad j = 1, 2,$

and let

(3.2) $\quad \xi_{\min} = f_2(x^2), \quad \xi_{\max} = 1/f_1(x^1).$

Proposition 3.1

(i) *If $\xi_{\min} > \xi_{\max}$, then $X \cap Y = \emptyset$.*

(ii) $h(\xi) = +\infty, \quad \forall \xi \notin [\xi_{\min}, \xi_{\max}]$.

Proof: (i) If $\xi_{\min} > \xi_{\max}$, then $f_1(x) \cdot f_2(x) > 1$ for any $x \in X$, which implies that $X \cap Y = \emptyset$.
(ii) Since $f_2(x) \geq \xi_{\min}$ for any $x \in X$, $X \cap Z(1/\xi, \xi)$ is empty for any $\xi < \xi_{\min}$. Similarly, $X \cap Z(1/\xi, \xi)$ is empty for any $\xi > \xi_{\max}$. □

Thus there must be a global minimum point ξ^* of $h(\xi)$ in the interval $[\xi_{\min}, \xi_{\max}]$ if $X \cap Y \neq \emptyset$. Suppose $[\xi_s, \xi_t]$ be any nonempty interval included in $[\xi_{\min}, \xi_{\max}]$. For ξ_s and ξ_t we define the following auxiliary problem:

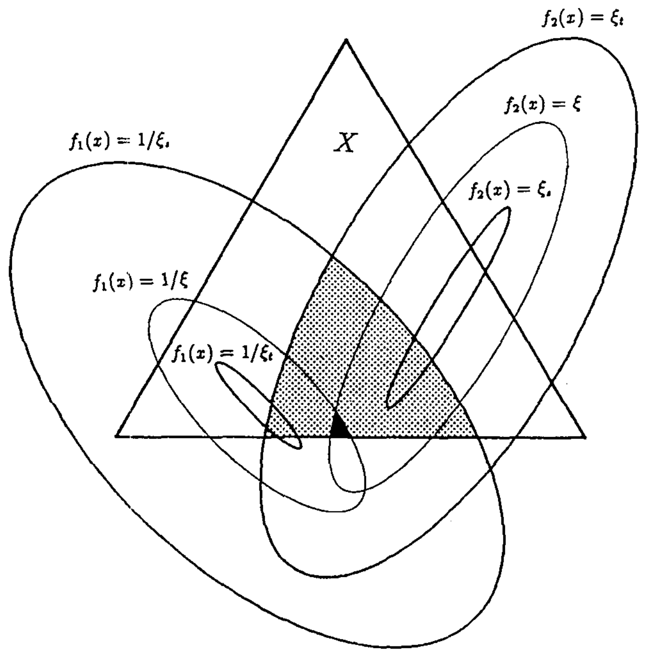

Figure 3.1: Feasible Regions of $Q(\xi_s, \xi_t)$ and $MP(\xi)$

$$Q(\xi_s, \xi_t) \left| \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in X \cap Z(1/\xi_s, \xi_t). \end{array} \right.$$

Let us denote by $\tilde{x}(\xi_s, \xi_t)$ an optimal solution of this convex minimization problem if exists, and define

$$(3.3) \qquad H(\xi_s, \xi_t) = \begin{cases} f_0(\tilde{x}(\xi_s, \xi_t)), & \text{if} \quad X \cap Z(1/\xi_s, \xi_t) \neq \emptyset, \\ +\infty, & \text{otherwise.} \end{cases}$$

Lemma 3.2

(i) *If* $\xi_t/\xi_s \leq 1 + \epsilon$ *for* $\epsilon > 0$, *any feasible solution of* $Q(\xi_s, \xi_t)$ *is* ϵ*-feasible for* P.

(ii) $H(\xi_s, \xi_t) \leq h(\xi), \quad \forall \xi \in [\xi_s, \xi_t]$.

Proof: (i) It is obvious because $Z(1/\xi_s, \xi_t) \subset Y(\epsilon)$.
(ii) It follows from $\xi_s \leq \xi \leq \xi_t$ that

$$Z(1/\xi_s, \xi_t) \supset Z(1/\xi, \xi),$$

because $f_1(x) \leq 1/\xi \leq 1/\xi_s$ and $f_2(x) \leq \xi \leq \xi_t$ for $x \in Z(1/\xi, \xi)$. □

Figure 3.1 shows the relation between the feasible region of $Q(\xi_s, \xi_t)$ and that of $MP(\xi)$ when the problem P has two variables.

For any $\epsilon > 0$ we can take $\xi_0, \xi_1, \ldots, \xi_m$ such that $\xi_{\min} = \xi_0 < \xi_1 < \cdots < \xi_m = \xi_{\max}$ and $\xi_k/\xi_{k-1} \leq 1 + \epsilon$, $k = 1, \ldots, m$. Let

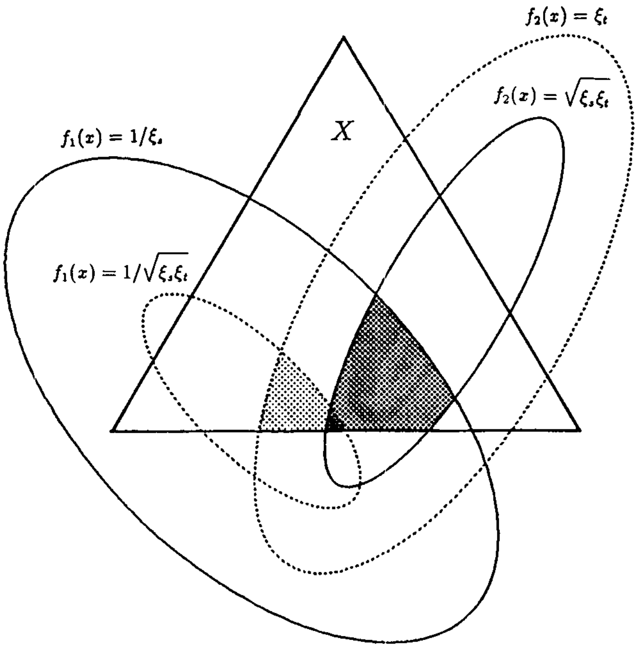

Figure 3.2: Division of the interval $[\xi_s, \xi_t]$

$$(3.4) \qquad H(\xi_{q-1}, \xi_q) = \min\{H(\xi_{k-1}, \xi_k) \mid k = 1, \ldots, m\}.$$

Then we have

$$(3.5) \qquad \tilde{x}(\xi_{q-1}, \xi_q) \in X \cap Y(\epsilon), \quad H(\xi_{q-1}, \xi_q) \leq h(\xi^*) = g(0).$$

By definition, $\tilde{x}(\xi_{q-1}, \xi_q)$ is an ϵ-optimal solution of P. Thus we can obtain an ϵ-optimal solution of P by solving a finite number of convex minimization problems $\mathrm{Q}(\xi_{k-1}, \xi_k)$'s.

3.2 Branch-and-bound algorithm

We are now ready to construct a branch-and-bound algorithm for obtaining an ϵ-optimal solution of P. First we define a procedure $\mathrm{A}(\epsilon, w, z, \mathrm{Q}(\xi_s, \xi_t))$ which solves an auxiliary problem $\mathrm{Q}(\xi_s, \xi_t)$ and shows whether the problem is fathomed or should be branched. Here w and z represent the incumbent and its objective function value, respectively, and $\mathcal{Q}$ is the set of auxiliary problems which are not fathomed. To make the set $Z(\xi_s, \xi_t)$ quickly included in $Y(\epsilon)$, we take $\sqrt{\xi_s\xi_t}$ as the new point dividing the interval $[\xi_s, \xi_t]$ into two subintervals (see Figure 3.2).

Procedure $\mathbf{A}(\epsilon, w, z, \mathbf{Q}(\xi_s, \xi_t))$

1° Compute $H(\xi_s, \xi_t)$ by solving $\mathrm{Q}(\xi_s, \xi_t)$.

2° If $H(\xi_s, \xi_t) \geq z$, then return.

3° If $\xi_t/\xi_s \leq 1 + \epsilon$, then let $w = \tilde{x}(\xi_s, \xi_t)$, $z = H(\xi_s, \xi_t)$ and return.

4° Let $\xi = \sqrt{\xi_s \xi_t}$ and add problems $Q(\xi_s, \xi)$ and $Q(\xi, \xi_t)$ to $\mathcal{Q}$. □

Choosing an appropriate $\epsilon > 0$, we obtain an ϵ-optimal solution of P by the following branch-and-bound algorithm:

Algorithm B

Stage 1 Let $x^1 = \text{argmin}\{f_1(x) \mid x \in X\}$, $x^2 = \text{argmin}\{f_2(x) \mid x \in X\}$. If $f_1(x^1) \cdot f_2(x^2) > 1$, then stop. Otherwise, let $\xi_{min} = f_2(x^2)$, $\xi_{max} = 1/f_1(x^1)$.

Stage 2 Let $z = +\infty$ and $\mathcal{Q} = \{Q(\xi_{min}, \xi_{max})\}$. Unless $\mathcal{Q}$ is vacant, take out a problem $Q(\xi_s, \xi_t)$ from $\mathcal{Q}$ and carry out Procedure $A(\epsilon, w, z, Q(\xi_s, \xi_t))$. □

Theorem 3.3 *Algorithm* B *terminates after finitely many iterations.*

Proof: Since we divide the interval $[\xi_s, \xi_t]$ by $\xi = \sqrt{\xi_s \xi_t}$ in 4° of Procedure A, we have

$$\xi/\xi_s = \xi_t/\xi = \sqrt{\xi_t/\xi_s}.$$

Therefore for the problems of depth d of the branching tree

$$\xi_t/\xi_t = (\xi_{\max}/\xi_{\min})^{1/2^d},$$

and $\xi_t/\xi_s \leq 1 + \epsilon$ hold when $d \geq (\ln\ln(\xi_{\max}/\xi_{\min}) - \ln\ln(1+\epsilon))/\ln 2$. Hence the tree is not branched out below some constant depth. □

4 Computational Experiments

We will report the results of the computational experiments of the algorithm presented in the previous section. We solved the simplest subclass of P, i.e., the linear program with an additional linear multiplicative constraint:

$$\text{PL}\left|\begin{array}{ll} \text{minimize} & c^t x \\ \text{subject to} & Ax \geq b, \quad x \geq 0, \\ & (d_1^t x) \cdot (d_2^t x) \leq 1, \end{array}\right.$$

where $c, d_1, d_2 \in R^n$, $b \in R^m$, $A \in R^{m \times n}$. All elements of A, b, c and d_k, $k = 1, 2$ were randomly generated, whose ranges are $[0.0, 1.0]$.

Note that every problem generated in Algorithm B is a linear program. In Stage 1, we determined the value of $\xi_{\min}$ and $\xi_{\max}$ by using the revised simplex method. In Stage 2, we applied the dual simplex method to every problem by taking the solution of the previous one as its dual feasible point. We employed the depth first rule in choosing a problem $Q(\xi_s, \xi_t)$ from $\mathcal{Q}$. In addition, among two problems $Q(\xi_s, \xi)$ and $Q(\xi, \xi_t)$ generated by Procedure $A(\epsilon, w, z, Q(\xi_s, \xi_t))$ we took out the one with the smaller value of H from $\mathcal{Q}$ before the other.

The program was coded in C language and run on a SUN4/280S computer. The size of problems ranges from $(m, n) = (30, 50)$ to $(220, 200)$. Ten examples were solved for each size.

Table 4.1 shows the results of Stage 1 of the Algorithm B. The average CPU time (Av.) and its standard deviation (S.d.) for solving the associated two linear programs, i.e.,

Table 4.1: CPU Time (in seconds) of Stage 1

m	30	70	70	130	130	180	180	220
n	50	50	100	100	150	150	200	200
Av.	1.112	5.580	9.498	35.632	57.157	139.912	171.158	227.217
S.d.	0.411	1.256	1.304	8.671	10.472	35.757	42.885	49.730

Table 4.2: Computational Results of Stage 2 ($\epsilon = 10^{-3}$)

m	30	70	70	130	130	180	180	220
n	50	50	100	100	150	150	200	200
CPU time (in seconds)								
Av.	4.132	11.255	22.742	54.157	120.048	174.170	234.097	322.903
S.d.	3.133	6.510	11.507	34.637	72.595	79.985	126.452	145.500
# of auxiliary problems								
Av.	89.8	93.8	75.8	170.0	189.6	187.6	163.0	150.4
Min.	25	25	27	55	57	97	33	57
Max.	277	139	157	657	439	305	355	251

Table 4.3: Computational Results of Stage 2 ($\epsilon = 10^{-5}$)

m	30	70	70	130	130	180	180	220
n	50	50	100	100	150	150	200	200
CPU time (in seconds)								
Av.	7.520	15.835	27.822	81.935	172.637	233.543	333.705	432.498
S.d.	5.690	9.691	15.742	61.833	94.834	103.426	193.124	225.171
# of auxiliary problems								
Av.	358.8	272.2	184.0	573.0	766.4	659.6	809.8	703.9
Min.	39	39	39	101	249	181	53	123
Max.	1173	775	501	2135	1489	1561	2825	2469

Table 4.4: Computational Results of Stage 2 ($(m, n) = (70, 100)$)

ϵ	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}
CPU time (in seconds)							
Av.	22.742	25.220	27.822	30.835	33.293	35.887	39.330
S.d.	11.507	12.939	15.742	18.171	19.599	21.448	24.645
# of auxiliary problems							
Av.	75.8	126.2	184.0	260.4	319.6	384.0	459.8
Min.	27	33	39	47	53	59	67
Max.	157	303	501	765	885	1093	1385

(4.1) $\quad \text{minimize}\{d_k^t x \mid Ax \geq b,\ x \geq 0\}, \quad k = 1, 2$

are listed in this table.

Tables 4.2 and 4.3 show the results of Stage 2 for $\epsilon = 10^{-3}$ and $\epsilon = 10^{-5}$, respectively. They contain the average number of problems generated in the process of computation as well as the average CPU time. Both its minimum (Min.) and maximum (Max.) numbers are also listed. Table 4.4 shows the results of Stage 2 when the size of problems was fixed at $(m, n) = (70, 100)$ and the tolerance ϵ ranges from 10^{-3} to 10^{-9}.

We see from these results that our algorithm can generate an ϵ-optimal solution of PL in less than twice more computational time as needed for solving the associated linear programs (4.1). The number of problems generated is more dependent on ϵ than the size of problems. However, the computational time is mildly dependent on the value of ϵ.

It appears that our algorithm is fairly efficient for the randomly generated class of problems PL. Also, our algorithm can be applied to a more general class of problems than the method proposed in [21], namely, problems P with nonlinear convex functions f_1 and f_2. Computational experiments for a more general class of problems are now under way, whose results will be reported subsequently.

Acknowledgment

This research was partly supported by Grant-in-Aid for Scientific Research of the Ministry of Education, Science and Culture, Grant No. (C)03832009, (C)03832018 and (C)04832010.

References

[1] Aneja, Y.P., V.Aggarwal and K.P.K.Nair, "On a class of quadratic programming," *European Journal of Operational Research 18* (1984) 62 – 70.

[2] Bector, C.R. and M.Dahl, "Simplex type finite iteration technique and reality for a special type of pseudo-concave quadratic functions," *Cahiers du Centre d'Etudes de Recherche Operetionnelle 16* (1974) 207 – 222.

[3] Chvătal, V., *Linear Programming* (W.H.Freeman and Company, N.Y., 1983).

[4] Dantzig, G.B., *Linear Programming and Extensions* (Princeton University Press, N.J., 1963).

[5] Fukushima, M., *Nonlinear Optimization Theory* (in Japanese, Sangyo-Tosho, Tokyo, 1980).

[6] Henderson, J.M. and R.E.Quandt, *Microeconomic Theory* (McGraw-Hill, N.Y., 1971).

[7] Hillestad, R.J. and S.E.Jacobsen, "Reverse convex programming," *Applied Mathematics and Optimization 6* (1980) 63 – 78.

[8] Hillestad, R.J. and S.E.Jacobsen, "Linear programs with an additional reverse convex constraint," *Applied Mathematics and Optimization 6* (1980) 257 – 269.

[9] Hogan, W.W., "Point-to-set maps in mathematical programming," *SIAM Review 15* (1973) 591 – 603.

[10] Horst, R. and H.Tuy, *Global Optimization: Deterministic Approaches* (Springer-Verlag, Berlin, 1990).

[11] Konno, H. and M.Inori, "Bond portfolio optimization by bilinear fractional programming," *Journal of the Operations Research Society of Japan 32* (1988) 143 – 158.

[12] Konno, H. and T.Kuno, "Linear multiplicative programming," *Mathematical Programming, A 56* (1992) 51 – 64.

[13] Konno, H. and T.Kuno, "Generalized linear multiplicative and fractional programming," *Annals of Operations Research 25* (1990) 147 – 162.

[14] Konno, H., Y.Yajima and T.Matsui, "Parametric simplex algorithm for solving a special class of nonconvex minimization problems," *Journal of Global Optimization 1* (1991) 65 – 81.

[15] Kuno, T. and H.Konno, "Parametric successive underestimation method for convex multiplicative programming problems," *Journal of Global Optimization 1* (1991) 267 – 285.

[16] Maling, K., S.H.Mueller and W.R.Heller, "On finding most optimal rectangular package plans," *Proceedings of the 19th Design Automation Conference* (1982) 663 – 670.

[17] Pardalos, P.M., "Polynomial time algorithms for some classes of constrained nonconvex quadratic problems, " *Optimization 21* (1990) 843 – 853.

[18] Pardalos, P.M. and J.B.Rosen, *Global Optimization: Algorithms and Applications, Lecture Notes in Computer Science No. 268* (Springer-Verlag, Berlin, 1987).

[19] Singer, I., "Minimization of continuous convex functions on complements of convex subsets of locally convex spaces," *Optimization 11* (1980) 221 – 234.

[20] Swarup, K., "Programming with indefinite quadratic function with linear constraints," *Cahiers du Centre d'Etudes de Recherche Operetionnelle 8* (1966) 133 – 136.

[21] Thach, P.T. and R.E.Burkard, "Reverse convex programs dealing with the product of two linear functions," Institute of Mathematics, Graz University of Technology (Graz, 1990).

[22] Tuy, H., "Convex programs with an additional reverse convex constraint," *Journal of Optimization Theory and Applications 52* (1987) 463 – 486.

Takahito KUNO:
Institute of Information Sciences and Electronics
University of Tsukuba
Tsukuba, Ibaraki 305, Japan