Journal of Computational Mathematics, Vol.15, No.3, 1997, 327–334.

FINDING THE STRICTLY LOCAL AND ϵ-GLOBAL MINIMIZERS OF CONCAVE MINIMIZATION WITH LINEAR CONSTRAINTS*[1)]

Patrice Marcotte
(*Centre de Recherche sur Les Transports, Université de Montréal, Quebéc, Canada*)
Shi-quan Wu
(*Probability Laboratory, Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, China*)

Abstract

This paper considers the concave minimization problem with linear constraints, proposes a technique which may avoid the unsuitable Karush-Kuhn-Tucker points, then combines this technique with Frank-Wolfe method and simplex method to form a pivoting method which can determine a strictly local minimizer of the problem in a finite number of iterations. Basing on strictly local minimizers, a new cutting plane method is proposed. Under some mild conditions, the new cutting plane method is proved to be finitely terminated at an ϵ-global minimizer of the problem.

1. Introduction

This paper considers the following nonlinear programming problem

$$(NLP) \qquad \min\{f(x) \mid x \in C\},$$

where $f(x)$ is a strictly concave function and $C \subset R^n$ is a convex polytope which will be specified later. It's well known that if (NLP) has a solution, then the minimum value can be attained at a vertex of the constraint. Generally speaking, this problem is NP-hard [1]. The ordinary descent methods usually generate a sequence of points which converges to a Karush-Kuhn-Tucker point of (NLP) under some conditions. Unfortunately, this Karush-Kuhn-Tucker point can not be guaranteed to be a local minimizer even if it satisfies the second order necessary conditions.

The purpose of this paper is to propose a technique for eliminating the unsuitable Karush–Kuhn–Tucker points. By combining this technique with Frank-Wolfe method and simplex method we form a descent method for (NLP). Under some mild conditions it is proved that, in a finite number of iterations, the method stops at a strictly local minimizer of (NLP). This kind of result was first obtained in [2] for a special class of problems they called concave knapsack problems. In their paper, they also gave out a tight complexity lower bound for their method. Although the global minimizer can not be guaranteed, the strictly local minimizer can provide good approximation to the global solution of (NLP) and they are very useful in the branch–and–bound

* Received April 17, 1995.
1)Sponsored by NSERC grant A5789 and Natural Science Foundation of China.

algorithms for the global optimization. Basing on the strictly local minimizer, we will further present a new cutting plane method which can be viewed as a revised version of Tuy's cutting plane method [3].

The convergence of Tuy's cutting plane method is still an open problem except we add some extra conditions on the method itself [4], [5], [6]. The new cutting plane method uses an ϵ procedure and an alternative implicit vertex enumerating procedure and is therefore finitely convergent without any extra assumptions.

The paper will be organized as follows. In section 2 we will introduce some assumptions and notations; describe the finitely convergent algorithm for the strictly local minimizers and the corresponding convergence analysis. In section 3 we will present a new cutting plane method for the ϵ-global minimizer and its theoretical analysis. Section 4 will be the conclusion section.

2. Finding The Strictly Local Minimizer

This section considers the following concave minimization problem

$$(P) \qquad \min\{f(x) \mid x \in R\},$$

where $f(x)$ is a strictly concave function, $R = \{x | Ax = b, x \geq 0\}$, $A \in R^{m\times n}$, $b \in R^m$.

Throughout of this section, we will make and use the following assumptions and notations.

Assumption 1 *$f(x)$ is strictly concave and continuously differentiable.*

Assumption 2 *R is nonempty, bounded and rank$(A) = m$.*

Notations: $N = \{1, 2, \cdots, n\}$, $M = \{1, 2, \cdots, m\}$, $A = (a_{ij} | i \in M, j \in N)$. If $J \subseteq N$, $L \subseteq M$, then $A_L^J = (a_{ij} | i \in L, j \in J)$, when $J = N$ or $L = M$, we also simply set $A_L = A_L^N$ or $A^J = A_L^M$. For a given subset $I \subset N$ with $|I| = m$, $| * |$ designates the cardinality of $*$, if A^I is invertible, then set $T(I) = (A^I)^{-1}A$ and $t(I) = (A^I)^{-1}b$. If $t(I) \geq 0$, then I is called a basis. Let $\bar{I} = N \backslash I$, $T^{\bar{I}}(I) = (A^I)^{-1}A^{\bar{I}}$ and $T_r^{\bar{I}}$ is the rth row of $T^{\bar{I}}(I)$. For a given basis I and $x \in R$, let $x = (x_I, x_{\bar{I}})$, $\bigtriangledown f(x) = \Big(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_n}\Big)$, $\bigtriangledown_{\bar{I}} f(x) = \Big(\frac{\partial f}{\partial x_i} | i \in \bar{I}\Big)$, $\bigtriangledown_I f(x) = \Big(\frac{\partial f}{\partial x_i} | i \in I\Big)$. It's clear that $x_I = t(I) - T^{\bar{I}}(I)x_{\bar{I}}$. If we define $\bar{f}(x_{\bar{I}}) = f(t(I) - T^{\bar{I}}(I)x_{\bar{I}}, x_{\bar{I}})$, then we have

$$\bigtriangledown \bar{f}(x_{\bar{I}}) = \bigtriangledown_{\bar{I}} f(x) - \bigtriangledown_I f(x) T^{\bar{I}}(I). \qquad (1)$$

This formula just designates what is usually called the reduced gradient of $f(x)$. conv$(*)$ and vol$(*)$ will represent the convex hull of $*$ and the volume of $*$ respectively. $\emptyset$ denotes the empty set.

It can be seen that the above notations inherit that of the simplex method for linear programming except the cost vector now is $\bigtriangledown f(x)$. The following algorithm is designed for finding the strictly local minimizer of the problem (P).

Algorithm I

- Initilization

  Given a vertex x^0 of R, let I be its corresponding basis, set $k = 0$.

Step 1. Calculate $\bigtriangledown \bar{f}(x_{\bar{I}}^k)$ and $T^{\bar{I}}(I)$.

Step 2. Set $\Omega = \left\{ j \middle| \frac{\partial \bar{f}(x_{\bar{I}}^k)}{\partial x_j^k} < 0, j \in \bar{I} \right\}$.

Step 3. If $\Omega = \emptyset$, then set $y_I = t(I), y_{\bar{I}} = 0$ and go to Step 4. If $\Omega \neq \emptyset$, then pivot according to Bland rules or any other anticycling rules and we get a new basis I, return to Step 1 with (x^k, I).

Step 4. If $y \neq x^k$, then set $x^{k+1} = y$, $k = k+1$ and return to Step 1 with (x^{k+1}, I). If $y = x^k$, then set $\Omega^0 = \left\{ j | j \in \bar{I}, \frac{\partial \bar{f}(x_{\bar{I}}^k)}{\partial x_j} = 0 \right\}$.

Step 5. If $\Omega^0 = \emptyset$, stop. If $\Omega^0 \neq \emptyset$, set $S_k = \{ i \mid x_i^k > 0, i = 1, 2, \cdots, n \}$. If $|S_k| = m$, then select any $j \in \Omega^0$ as the variable index which will be brought into basis, pivot and we get a new point y and a new basis I. Set $x^{k+1} = y$ and return to Step 1 with (x^{k+1}, I).

Step 6. For each $i = 1, 2, \cdots, n$, solve

$$(PI) \qquad \min\{x_i | x \in R, \nabla f(x^k)x = \nabla f(x^k)$$

and

$$(PI') \qquad \max\{x_i | x \in R, \nabla f(x^k)x = \nabla f(x^k)x^k\}$$

by using simplex method until we find another $y \neq x^k$. If for $i = 1, 2, \cdots, n$, all the solutions of (PI) and (PI') are equal to x^k, then stop. Otherwise, set $x^{k+1} = y$ and return to Step 1 with (x^{k+1}, I).

Remark In the algorithm, step 1 to step 3 is exactly the simplex method applying to

$$\min\{\nabla f(x^k)(x - x^k) \mid x \in R \}. \tag{2}$$

The function of other steps is to detect the strictly local minimizers.

Lemma 2.1 *If $y \neq x^k$ in Step 4, then $f(y) < f(x^k)$.*

Proof. By the definition of y, it's clear that $\nabla f(x^k)(y - x^k) \leq 0$. The strictly concavity of $f(x)$ implies

$$f(y) < f(x^k) + \nabla f(x^k)(y - x^k) \leq f(x^k). \tag{3}$$

Lemma 2.2 *At the end of Step 6, we either find a vertex y of R which differs from x^k and satisfies that $\nabla f(x^k)y = \nabla f(x^k)x^k$ or we conclude that x^k is the unique solution of* (2)

Proof. By the definition of (PI) and (PI'), we need only to prove that any solution of (PI) or (PI') is a vertex of R. Without loss of generality, let y be a vertex solution of (PI). If $y = x^k$, the conclusion is obvious. Now we suppose that $y \neq x^k$ and y is not a vertex of R. Then we have $y^1, y^2 \in R$ and $y^1 \neq y^2$ such that $y = \alpha y^1 + (1-\alpha)y^2$, where $\alpha \in (0,1)$. Since x^k is a solution of (2), if $\max\{\nabla f(x^k)y^1, \nabla f(x^k)y^2\} > \nabla f(x^k)x^k$, then

$$\nabla f(x^k)y = \nabla f(x^k)[\alpha y^1 + (1-\alpha)y^2] > \nabla f(x^k)x^k = \nabla f(x^k)y,$$

which is a contradiction. So we have $\nabla f(x^k)y^1 = \nabla f(x^k)y^2 = \nabla f(x^k)x^k$, and hence y^1 and y^2 are feasible for (PI). But this is impossible since y is a vertex of $\{ x \mid x \in R, \nabla f(x^k)x = \nabla f(x^k)x^k \}$ and $y = \alpha y^1 + (1-\alpha)y^2$. Therefore y must be a vertex of R. The proof has been completed.

Lemma 2.3 *If the algorithm I stops at x^k, then x^k is a strictly local minimizer of* (P).

Proof. By the stopping rules in Step 5 and Step 6 we know that x^k is the unique solution of (2). Hence we have $\nabla f(x^k)(x - x^k) > 0$ for any $x \in R$ and $x \neq x^k$. Now let

$$\epsilon = \min_{x \in R, x \neq x^k} \left\{ \nabla f(x^k) \frac{x - x^k}{\|x - x^k\|} \right\}.$$

It's clear that $\epsilon > 0$ since R is a polyhedron. From the continuous differentiability of $f(x)$ we conclude that there exists $\delta > 0$ such that: $x \neq x^k$ and $\|x - x^k\| \leq \delta$ imply that

$$|f(x) - f(x^k) - \nabla f(x^k)(x - x^k)| \leq \frac{\epsilon}{2}\|x - x^k\|. \tag{4}$$

Then it follows

$$f(x) \geq f(x^k) + \nabla f(x^k)(x - x^k) - \frac{\epsilon}{2}\|x - x^k\| \geq f(x^k) + \frac{\epsilon}{2}\|x - x^k\| > f(x^k).$$

The proof has been completed.

Lemma 2.4 *If $\Omega^0 \neq \emptyset$ and $|S_k| = m$, then for any $j \in \Omega^0$, we can implement the pivot and get a new point y such that $y \neq x^k$ and $\nabla f(x^k)y = \nabla f(x^k)x^k$.*

*Proof.*It's well known in linear programming.

Theorem 2.1 *After a finite number of iterations, the algorithm stops at a strictly local minimizer of (P).*

Proof. By the algorithm and Lemma 2.1 to Lemma 2.4 we know that either the algorithm stops at a strictly local minimum vertex x^k, or we get another vertex x^{k+1} such that $f(x^{k+1}) < f(x^k)$. The conclusion follows from the finiteness of the vertices of R. The proof has been completed.

Theorem 2.2 *Let $f(x)$ is a concave function of R^n, then there exists a positive number τ_0 such that, for any $\tau \in (0, \tau_0]$, the vertex solution of*

$$(P_\tau) \qquad \min\{f(x) - \tau \parallel x \parallel^2 \ \mid \ x \in R\ \}$$

is also a solution of (P).

Proof. Let x_τ be a vertex solution of (P_τ) and x^* be a vertex solution of (P). Then we have $f(x^*) \leq f(x_\tau)$ and $f(x_\tau) - \tau \parallel x_\tau \parallel^2 \leq f(x^*) - \tau \parallel x^* \parallel^2$. Thus

$$0 \leq f(x_\tau) - f(x^*) \leq \tau(\parallel x_\tau \parallel^2 - \parallel x^* \parallel^2). \tag{5}$$

Let $\gamma = \min\{f(v) - f(x^*) | v \in V(R),\ f(v) > f(x^*)\}$, where $V(R)$ is the vertices set of R, then $\gamma > 0$ since R has only a finite number of vertices. By (5) we conclude that $f(x_\tau) = f(x^*)$ when τ is small enough. The proof has been completed.

Remark This theorem says the assumption of strict concavity of $f(x)$ can be theoretically guaranteed by a small perturbation of $f(x)$ if it is only concave.

3. Cutting Plane Method for The Global Minimizer

This section will discuss how to get a global minimizer of the following problem

$$(Q) \qquad \min\{f(x) | \quad x \in \bar{R}\},$$

where $f(x)$ is a concave function and $\bar{R} = \{x | Ax \leq b\}$. We use this kind of formulation is only for convenience. All the results obtained in this section can be extended to other formulations. Throughout of this section, we assume that $\bar{R}$ is nonempty and bounded and of full dimension, $-f(x)$ is a strongly convex function, i.e. there exist $\hat{x} \in R^n$ and $\sigma_1 > 0, \sigma_2 > 0$ such that

$$\sigma_1 \parallel x - \hat{x} \parallel^2 \leq -f(x) \leq \sigma_2 \parallel x - \hat{x} \parallel^2 . \tag{6}$$

Definition 3.1 *Given a hyperplane H, if $\bar{R} \bigcap H$ is a nonempty face of $\bar{R}$, then H is called an affine face of $\bar{R}$.*

Definition 3.2 *Let $v_1, v_2, \cdots, v_{t+1}$ be all the vertices of $\bar{R}$, $v_1, v_2, \cdots, v_n$ be adjacent to v_{t+1}. Passing through $v_1, v_2, \cdots, v_n$, we derive a hyperplane*

$$e^T(v_1 - v_{t+1}, v_2 - v_{t+1}, \cdots, v_n - v_{t+1})^{-1}(x - v_{t+1}) = 1, \tag{7}$$

where $e \in R^n$ is the all ones vector. If this hyperplane is an affine face of conv $(v_1, v_2, \cdots, v_t)$ and it separates v_{t+1} from conv $(v_1, v_2, \cdots, v_t)$, then this hyperplane is called a valid cutting plane for $(v_{t+1}, \bar{R})$.

Remark i) It is possible in the algorithm proposed below that (7) is a hyperplane that contains conv $(v_1, v_2, \cdots, v_t)$. In this case, the dimension of the problem can be decreased by one. So we will neglect this case in the forthcoming discussions. ii) If v_{t+1} is non degenerate, then v_{t+1} will have only n adjacent vertices $v_1, v_2, \cdots, v_n$ and we can get a cutting plane like (7) easily. This cutting plane will cut off conv $(v_{t+1}, v_1, v_2, \cdots, v_n)$ completely from $\bar{R}$ and generate no new vertices. So we will only consider the case that v_{t+1} is degenerate in the following.

Lemma 3.1 *Let $v_1, v_2, \cdots, v_q$ $(q \geq n)$ be all the adjacent vertices of v_{t+1}, then the hyperplane passing through $v_1, v_2, \cdots, v_n$ is a valid cutting plane for $(v_{t+1}, \bar{R})$ if and only if that*

$$e^T(v_1 - v_{t+1}, v_2 - v_{t+1}, \cdots, v_n - v_{t+1})^{-1}(v_i - v_{t+1}) \geq 1 \tag{8}$$

holds for all $i = n+1, n+2, \cdots, q$.

Proof. Suppose that (8) holds for each $i = n+1, n+2, \cdots, q$. Since $v_1, v_2, \cdots, v_q$ are all the adjacent vertices of v_{t+1}, $\bar{R} - v_{t+1}$ must be contained in the convex polyhedral cone forming by $v_1 - v_{t+1}, v_2 - v_{t+1}, \cdots, v_q - v_{t+1}$. Hence for any $i = q+1, q+2, \cdots, t$, there exist $\lambda_j \geq 0$ such that

$$v_i - v_{t+1} = \sum_{j=1}^{q} \lambda_j (v_j - v_{t+1}). \tag{9}$$

If $\sum_{j=1}^{q} \lambda_j < 1$, from (9) we have

$$v_i = \sum_{1=1}^{q} \lambda_j v_j + \Big(1 - \sum_{j=1}^{q} \lambda_j) v_{t+1} \tag{10}$$

which implies that v_i is not a vertex of $\bar{R}$ and thus contradicts with the assumption. So it must be $\sum_{j=1}^{q} \lambda_j \geq 1$.

Now let $d^T = e^T(v_1 - v_{t+1}, v_2 - v_{t+1}, \cdots, v_n - v_{t+1})^{-1}$. From (8) and (9) we get

$$d^T(v_i - v_{t+1}) = \sum_{j=1}^{q} \lambda_j d^T(v_j - v_{t+1}) \geq \sum_{j=1}^{q} \lambda_j \geq 1.$$

By the linearity we conclude that

$$d^T(x - v_{t+1}) \geq 1 \tag{11}$$

holds for any $x \in conv(v_1, v_2, \cdots, v_t)$. Notice that $d^T(x - v_{t+1}) = 1$ separates v_{t+1} from $conv(v_1, v_2, \cdots, v_t)$ and $v_1, v_2, \cdots, v_n$ are located on it. So this hyperplane is a valid cutting plane for $(v_{t+1}, \bar{R})$.

On the other hand, if (7) is a valid cutting plane for $(v_{t+1}, \bar{R})$, all the vertices of $\bar{R}$ except v_{t+1} must be on the same side of the hyperplane. Since $d^T(v_i - v_{t+1}) = 1$ holds for $i = 1, 2, \cdots, n$. We conclude that $d^T(v_i - v_{t+1}) \geq 1$ holds for $i = n+1, \cdots, q$. The proof has been completed.

Lemma 3.2 *Let $v_1, v_2, \cdots, v_q$ be all the adjacent vertices of v_{t+1}. For any $i = 1, 2, \cdots, q$, we find α_i such that*

$$f(v_{t+1} + \alpha_i(v_i - v_{t+1})) = f(v_{t+1}) - \epsilon, \tag{12}$$

where ϵ is a pre-specified positive number. Let

$$u_i = v_{t+1} + \alpha_i(v_i - v_{t+1}) \tag{13}$$

and π be any solution of the following linear inequality system

$$(u_i - v_{t+1})^T \pi \geq 1, \quad i = 1, 2, \cdots, q, \tag{14}$$

then for any $x \in \bar{R}$ satisfying $f(x) < f(v_{t+1}) - \epsilon$, we have $x \in \{x | \pi^T(x - v_{t+1}) \geq 1\}$.

Proof. Suppose $x \in \bar{R}$, by the definition of u_i's we know that there exist non-negative numbers λ_i $(i = 1, 2, \cdots, q)$ such that $x - v_{t+1} = \sum_{i=1}^q \lambda_i(u_i - v_{t+1})$. From the definition of π we may obtain that $\pi^T(x - v_{t+1}) = \sum_{i=1}^q \lambda_i \pi^T(u_i - v_{t+1}) \geq \sum_{i=1}^q \lambda_i$. Therefore the conclusion is true if we can prove $\sum_{i=1}^q \lambda_i > 1$. This has actually been implied by the condition $f(x) < f(v_{t+1}) - \epsilon$. In fact, if $\sum_{i=1}^q \lambda_i \leq 1$, then $x = \sum_{i=1}^q \lambda_i u_i + (1 - \sum_{i=1}^q \lambda_i) v_{t+1}$ which is a convex combination of v_{t+1} and u_i's. Making use of concavity of $f(x)$ we have

$$f(x) \geq \sum_{i=1}^q \lambda_i f(u_i) + \Big(1 - \sum_{i=1}^q \lambda_i\Big) f(v_{t+1}) > f(v_{t+1}) - \epsilon \tag{15}$$

which contradicts with the assumption. The proof is completed.

Lemma 3.3 *Let $v_1, v_2, \cdots, v_{t+1}$ be all the vertices of $\bar{R}$, $v_1, v_2, \cdots, v_q$ be all the adjacent vertices of v_{t+1}; $\pi_1^T(x - v_{t+1}) = 1$, $\pi_2^T(x - v_{t+1}) = 1, \cdots, \pi_p^T(x - v_{t+1}) = 1$, where $p \leq N(q, n)$ (the number of all the possibilities for forming a valid cutting plane for $(v_{t+1}, \bar{R})$), be all the valid cutting planes for $(v_{t+1}, \bar{R})$, then we have*

$$conv\ (v_1, v_2, \cdots, v_t) = \{x | x \in \bar{R}, \pi_i^T(x - v_{t+1}) \geq 1, \quad i = 1, 2, \cdots, p\}. \tag{16}$$

Proof. Denote the right-hand side and the left-hand side of the equality as RH and LH respectively. Then it's clear that $LH \subseteq RH$ from the definition of a valid cutting plane for $(v_{t+1}, \bar{R})$. If $x \notin LH$ and $x \notin \bar{R}$ or if $x = v_{t+1}$, then $x \notin RH$.

Now we suppose that $x \notin LH$, $x \in \bar{R}$ and $x \neq v_{t+1}$. Then it follows that x must belong to the convex polyhedral cone forming by $v_1, v_2, \cdots, v_q$ and v_{t+1}. But $x \notin LH$, so the line passing through x and v_{t+1} must have an intersection point, say y, with at least one affine face of $conv(v_1, v_2, \cdots, v_t)$, say $\pi_i^T(x-v_{t+1}) = 1$. Hence there exists $\alpha \in (0,1)$ such that $x = \alpha y + (1-\alpha)v_{t+1}$. It follows $\pi_i^T(x-v_{t+1}) = \alpha\pi_i^T(y-v_{t+1}) = \alpha < 1$. This concludes that $x \notin RH$ and therefore $RH \subseteq LH$. The proof has been completed.

Algorithm II

- Initialization
  Given $\epsilon \geq 0$, $L \geq 0$, and a vertex $x^0 \in \bar{R}$, let $\bar{R}_1 = \bar{R}$, $k = 1$.

- kth iteration

1. If $\bar{R}_k = \emptyset$, then stop. Otherwise, find a strictly local minimizer z of $f(x)$ on $\bar{R}_k$.
2. If $f(z) \leq f(x^{k-1})$, set $x^k = z$; otherwise, set $x^k = x^{k-1}$. Let $v_1^k, v_2^k, \cdots, v_{p_k}^k$ be all the adjacent vertices of z. For $i = 1, 2, \cdots, p_k$, find α_i^k such that

$$f(z + \alpha_i^k(v_i^k - z)) = f(x^k) - \epsilon. \tag{17}$$

Let

$$u_i^k = z + \alpha_i^k(v_i^k - z). \tag{18}$$

3. Solve the following linear inequality system

$$(LIS_k) \qquad (u_i^k - z)^T\pi \geq 1, \quad i = 1, 2, \cdots, p_k\}.$$

Suppose π^k be one of its vertex solutions. Let

$$S_k = \{i|(u_i^k - z)^T\pi^k = 1, \quad i = 1, 2, \cdots, p_k\}, \tag{19}$$

$$W_k = conv(z, v_i^k, i \in S_k). \tag{20}$$

4. If vol $(W_k) \geq \frac{\epsilon}{L}$, then set

$$\bar{R}_{k+1} = \{x|x \in \bar{R}_k, (\pi^k)^T(x-z) \geq 1\} \tag{21}$$

and goto step $k+1$.
If vol $(W_k) < \frac{\epsilon}{L}$, then find all the valid cutting planes for $(z, \bar{R}_k)$, say

$$\pi_1^T(x-z) = 1, \pi_2^T(x-z) = 1, \cdots, \pi_{m_k}^T(x-z) = 1.$$

Set

$$\bar{R}_{k+1} = \{x|x \in \bar{R}_k, \pi_i^T(x-z) \geq 1, \quad i = 1, 2, \cdots, m_k\} \tag{22}$$

and go to step $k+1$.

Remark The efficient vertex listing methods can be found in [7] [8] and [9]. The volume of W_k can be calculated easily since $W_k = conv(z, v_i^k, i \in S_k)$. We can also choose W_k as a simplex forming by z and some other n vertices in S_k.

Theorem 3.1 *After a finite number of iterations, the algorithm II terminates at an ϵ- global minimizer or a global minimizer of (Q).*

Proof. Since the volume of $\bar{R}$ is finite, the inequality vol $(W_k) \geq \frac{\epsilon}{L}$ can hold only for a finite number of k. If at some iterate k we have $\bar{R}_k = \emptyset$. Then we get an ϵ-global minimizer of (Q). For otherwise, there exists an integer k_0 such that $k \geq k_0$ implies that

$$\text{vol}\ (W_k) < \frac{\epsilon}{L}. \tag{23}$$

According to the algorithm II and Lemma 3.3 we know that each iteration after k_0 deletes one vertex of $\bar{R}_{k_0}$ without inventing any new vertex. Since the number of the vertices of $\bar{R}_{k_0}$ is finite. We can terminate the algorithm in a finite number of iterations and get a global minimizer of (Q). The proof has been completed.

4. Conclusions

It is noted that the convergence of Tuy's cutting plane method is still an open problem, see [5] [6] for example. There are several convergent conditions are proposed for Tuy's method. These conditions are assumed usually on the algorithm itself, for instance, to assume the distance between the cut vertex and the corresponding cutting plane is uniformly greater than a positive constant [4]. There are also many modifications for Tuy's method. One of them is the facial cut method which needs more information about the newly generated polyhedron. The new cutting plane method relaxes the limitation on the convergence condition and uses only the basic information in each iteration. It is also evident that the techniques here can be applied to other global or integer optimization problems where the cutting plane methods can be used.

Acknowledgements The authors would like to express their many thanks to Prof. Michael Florian, Dr. Y. Chen and Dr. J.H.Wu for their helps and many valuable discussions, and to the referees for supplying so many useful ideas.

References

[1] K.G. Murty and S.N. Kabadi, Some NP-complete problems in quadratic and nonlinear programming, *Mathematical Programming,* **39** (1987), 117–129.

[2] J.J. Moré and S.A. Vavasis, On the solution of concave knapsack problems, *Mathematical Programming,* **49** (1991), 397–411.

[3] H. Tuy, Concave programming under linear constraints, *Soviet Mathematics,* **5** (1962), 1437–1441.

[4] R. Horst and H. Tuy, Global Optimization, Springer-Verlag 1990.

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

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

[7] M. Balinski, An algorithm for finding all the vertices of convex polyhedral set, *J. SIAM.,* (1961), 72–88.

[8] T.H. Matheis and D.S. Rubin, A survey and comparison of methods for finding all vertices of convex polyhedral sets, *Mathematics of Operations Research,* **5** (1980), 167–184.

[10] J. Scott Provan, Efficient enumeration of the vertices of polyhedral associated with network LP's, Technical Report, Department of Operations Research, University of North Carolina, Chapel Hill NC 27599-3180.