# Comments on a Reverse Convex Programming Algorithm

S. BENSAAD[1] and S. E. JACOBSEN[2]
[1] *AT&T Bell Laboratories, Middletown, NJ 07748, U.S.A.* [2] *Electrical Engineering Department, UCLA , Los Angeles, CA 90024-1594, U.S.A.*

We have the following comments on the algorithms that have appeared in [1] and [2]. Consider the following reverse convex program, denoted by P,

$$\begin{aligned}&\min c^t x\\&0 \leq x_j \leq 1\,,\quad j = 1, 2, 3\\&g(x) = -x^t x + 2.5 \leq 0\end{aligned}$$

where $c^t = (1, 0, 0)$.

(i) This problem has as its unique optimal solution the vector $x^* = (0.7071, 1, 1)$; this vector is on the edge of the polytope that connects the vertices $(0, 1, 1)$ and $(1, 1, 1)$. The first linear program of the algorithm in [1] produces the solution $x^0 = (0, 1, 1)$. The resulting first Tuy cut, $a^t x \geq a_o$ is given by

$$a^t = (1.4142, -04495, -0.4495) \quad \text{and} \quad a_0 = 0.1010.$$

The next linear program, with the appended Tuy cut, of the algorithm in [1] produces the solution $x^1 = (0.0714, 0, 0)$. The algorithm then produces the three vectors, $u^1, u^2, u^3$. Briefly, these vectors are determined as follows. Compute the neighboring vertices of $x^1$. All of these vertices are on edges of the original polytope, and two of the vertices are also on the Tuy plane; the third vertex is on the same edge as $x^1$. Determine the direction of each such edge and move in that direction until the surface of the reverse constraint is met. The appropriate direction is the one that does not decrease the objective. The $u$'s are

$$\begin{aligned}u^1 &= (1.5811, 0, 0)\\u^2 &= (1.2247, 1, 0)\\u^3 &= (1.2247, 0, 1).\end{aligned}$$

The next step of the algorithm in [1] generates the hyperplane that passes through the affinely independent vectors $u^1, u^2, u^3$. The previous Tuy cut is deleted and the new cut, based upon the hyperplane that passes through $u^1, u^2, u^3$, is appended to the original linear program. This new constraint is given by $b^t x \geq b_0$, where

*Journal of Global Optimization* **5:** 95–96, 1994.

$b^t = (0.6624, 0.2361, 0.2361)$ and $b_0 = 1.0473$. However, this constraint cuts away the optimal solution $x^*$.

(ii) Now consider (P) with the vector $c^t = (1, 10^{-4}, 10^{-4})$. The purpose of the $10^{-4}$ components is to guarantee that the origin is the unique optimal solution of the initial linear program. The algorithm in [2] requires a vector $y$ that is interior to the polytope and on the surface of the reverse convex function. For this example, we take $y = (0.912871, 0.912871, 0.912871)$. The optimal solution for the first linear program is $x^0 = (0, 0, 0)$. Let $v^1, v^2, v^3$ denote the three neighboring vertices of $x^0$. For each of three cones, obtained by replacing a neighbor with $y$, the algorithm in [2] generates a sequence of vectors that converges to a reverse convex feasible point that is also on the boundary of the original polytope. At the first major iteration, the limits of those sequences are given by

$$\begin{aligned} z^1 &= (1, 0.866025, 0.866025) \\ z^2 &= (0.866025, 1, 0.866025) \\ z^3 &= (0.866025, 0.866025, 1). \end{aligned}$$

Each of these vectors is on a two-dimensional facet of the original polytope. The next step of the algorithm is to solve the three linear programs

$$\min\{c^t x | x \in F_A \cap H_i^+\}, \quad i = 1, 2, 3,$$

where $F_A$ is the original polytope and each $H_i^+$ is a half-space, not containing $x^0$, whose bounding hyperplane is obtained by, for example take $i = 1$, generating the plane that passes through $y, z^2, z^3$. The other half-spaces are similarly obtained. The optimal vectors for these three linear program are

$$\begin{aligned} x_*^1 &= (0.710227, 1, 1) \\ x_*^2 &= (0.750818, 1, 1) \\ x_*^3 &= (0.750818, 1, 1). \end{aligned}$$

Each of these three vectors is feasible and on the same edge as the optimal solution $x^*$. Hence, Step IV of the algorithm, as stated in the paper, will delete the optimal solution $x^*$.

## References

1. S. BenSaad (1992), A Cutting Plane Algorithm for a Class of Non-Linear/0-1 Integer Programs, in *Recent Advances in Global Optimization*, pp. 152–164, Princeton University Press.
2. S. BenSaad and S. E. Jacobsen (1990), A Level Set Algorithm for a Class of Reverse Convex Programs, *Annals of Operations Research* **25**, 19–42.