An Algorithm for Indefinite Quadratic Programming with Convex Constraints

Lê D. Muu and Werner Oettli

Nr. 89 (1989)

Fakultät für Mathematik und Informatik, Universität Mannheim, D-6800 Mannheim 1, Germany

An Algorithm for Indefinite Quadratic Programming with Convex Constraints

Lê D. Muu[1,2] and **Werner Oettli**[1]

Abstract. We propose a branch–and–bound method for minimizing an indefinite quadratic function over a convex set. The bounding operation is based on a certain relaxation of the constraints.

Problem Statement. We propose a new branch-and-bound method for the following problem

$$\text{(P)} \qquad \min\{f(x,y) := p^T x + x^T M y + q^T y \mid x \in \mathbb{R}^n, y \in \mathbb{R}^m, (x,y) \in S\},$$

where $S \subset \mathbb{R}^n \times \mathbb{R}^m$ is a closed convex nonempty set, $p \in \mathbb{R}^n$ and $q \in \mathbb{R}^m$ are given vectors, and M is a given real $(n \times m)$-matrix.

Essentially the same problem has also been considered in [1]. The algorithm given there is quite different from ours. In [1] the bounding operation was based on using lower convex envelopes to the function $x^T M y$, whereas here it is based on relaxation of the constraints.

We suppose that problem (P) has an optimal solution, and we denote by f^* the optimal value of (P). We assume further that we can fix two compact convex polyhedra $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^m$ such that at least one optimal solution of (P) is contained in $X \times Y$.

Description of the Algorithm. For $B \subset Y$ we denote by $\mathrm{R}(B)$ the problem

$$\mathrm{R}(B) \qquad \min\{f(x,y) \mid x \in X, y \in B, u \in B, (x,u) \in S\},$$

and by $\beta(B)$ we denote the optimal value of $\mathrm{R}(B)$ (we let $\beta(B) := \infty$ if $\mathrm{R}(B)$ has no feasible points).If (x^B, y^B, u^B) is an optimal solution of $\mathrm{R}(B)$, then clearly

$$f(x^B, y^B) \leq \min\{f(x,u) \mid x \in X, u \in B, (x,u) \in S\} \leq f(x^B, u^B).$$

The algorithm can now be recursively described as follows:
At the beginning of iteration k (k=0,1,...) we have a collection Γ_k of polyhedral subsets $B \subset Y$ such that at least one optimal solution of (P) is contained in $X \times \cup\{B \mid B \in \Gamma_k\}$ (at the start set $\Gamma_0 := \{Y\}$). For each $B \in \Gamma_k$ we have determined $\beta(B)$ and, if $\beta(B) < \infty$, a solution (x^B, y^B, u^B) of $\mathrm{R}(B)$. Furthermore $\alpha_{k-1} \geq f^*$ is at hand (at the start set $\alpha_{-1} := \infty$). Let

$$\alpha_k := \min\{\alpha_{k-1}, \min\{f(x^B, u^B) \mid B \in \Gamma_k, \beta(B) < \infty\}\}.$$

[1] Universität Mannheim, Fakultät für Mathematik und Informatik, D-6800 Mannheim 1, Fed. Rep. Germany
[2] On leave from Institute of Mathematics, Hanoi, as fellow of Alexander von Humboldt Foundation

Let $\Delta_k := \{B \in \Gamma_k | \beta(B) \leq \alpha_k\}$. Select $B_k \in \Delta_k$ such that $\beta(B_k) = \min\{\beta(B) | B \in \Delta_k\}$.
Let (x^k, y^k, u^k) be a solution of $\mathrm{R}(B_k)$, and set $\beta_k := \beta(B_k) = f(x^k, y^k)$.
If $\beta_k \geq f(x^k, u^k)$, then terminate: (x^k, u^k) solves (P).
If $\beta_k < f(x^k, u^k)$, then let $c_k := (\beta_k + f(x^k, u^k))/2$ and bisect B_k into the two sets

$$B_k^- := \{y \in B_k | f(x^k, y) \leq c_k\}, \quad B_k^+ := \{y \in B_k | f(x^k, y) \geq c_k\}.$$

Solve $\mathrm{R}(B_k^-)$ and $\mathrm{R}(B_k^+)$, obtaining the optimal values and optimal solutions. Set $\Gamma_{k+1} := \Delta_k \backslash \{B_k\} \cup \{B_k^-, B_k^+\}$. Go to iteration $k+1$.

This completes the description of iteration k.

From $(x^B, u^B) \in S$ follows $f^* \leq f(x^B, u^B)$ and therefore $f^* \leq \alpha_k$. From $\min\{\beta(B) | B \in \Gamma_k\} \leq f^*$ follows then $\Delta_k \neq \emptyset$ and $\beta_k \leq f^*$. Moreover from $(x^k, u^k) \in S$ follows $\beta_k \leq f^* \leq f(x^k, u^k)$. Hence iteration k is well defined. If the algorithm terminates at iteration k, then $\beta_k = f^* = f(x^k, u^k)$, hence (x^k, u^k) solves (P). If no termination occurs in iteration k, then again $X \times \cup\{B | B \in \Gamma_{k+1}\}$ contains an optimal solution of (P), and it is clear that $\beta_k \leq \beta_{k+1} \leq f^*$. Of course, if $\beta_k \geq f(x^k, u^k) - \varepsilon$ for some $\varepsilon > 0$, then (x^k, u^k) is an ε–optimal solution of (P).

Convergence of the Algorithm. If the algorithm is not finite, then we have the following result.

THEOREM. If the algorithm does not terminate, then $\beta_k \nearrow f^*$, and any cluster point of $\{(x^k, u^k)\}$ solves (P).
Proof: From monotonicity, $\beta_k \nearrow \overline{\beta}$ for some $\overline{\beta} \leq f^*$. Let $(\overline{x}, \overline{u})$ be a cluster point of $\{(x^k, u^k)\}$. By extracting a subsequence if necessary, we may assume that $x^k \to \overline{x}, u^k \to \overline{u}, y^k \to \overline{y}$. Again by extracting a subsequence if necessary, we may assume that either $B_{k+1} \subset B_k^-$ for all k or $B_{k+1} \subset B_k^+$ for all k. In the first case we have $u^{k+1} \in B_k^-$ and therefore $f(x^k, u^{k+1}) \leq c_k$, hence

$$f(x^k, u^k) - \beta_k = 2(f(x^k, u^k) - c_k) \leq 2(f(x^k, u^k) - f(x^k, u^{k+1})) \to 0.$$

In the second case we have $y^{k+1} \in B_k^+$ and therefore $f(x^k, y^{k+1}) \geq c_k$, hence

$$f(x^k, u^k) - \beta_k = 2(c_k - f(x^k, y^k)) \leq 2(f(x^k, y^{k+1}) - f(x^k, y^k)) \to 0.$$

Thus in both cases we obtain in the limit that $f(\overline{x}, \overline{u}) \leq \overline{\beta} \leq f^*$. From $(x^k, u^k) \in S$ follows that $(\overline{x}, \overline{u})$ is feasible for (P). It remains $f(\overline{x}, \overline{u}) = \overline{\beta} = f^*$, and $(\overline{x}, \overline{u})$ solves (P). q.e.d.

Bounding Operation. A crucial operation in the algorithm is the solution of $\mathrm{R}(B)$. Due to the fact that $f(x, \cdot)$ is affine and B is a compact polyhedron, $\mathrm{R}(B)$ can be solved using only convex subprograms. Indeed, let $v^i (i = 1, 2, ..., q)$ be the vertices of B. Then since $\min_{y \in B} f(x, y) = \min_i f(x, v^i)$, we have

$$\begin{aligned}\beta(B) &= \min\{f(x, y) | x \in X, y \in B, u \in B, (x, u) \in S\} \\ &= \min\{\min_i f(x, v^i) | x \in X, u \in B, (x, u) \in S\} \\ &= \min_i(\min\{f(x, v^i) | x \in X, u \in B, (x, u) \in S\}),\end{aligned}$$

and the solution of R(B) is reduced to the solution of finitely many convex subprograms, one for each v^i. We observe that, since B is generated from some predecessor B' by adding one affine inequality, the vertices of B can be calculated from those of B' with reasonnable effort, see [2]. The starting polyhedron Y should be simple so that its vertices are easily obtained.

Outer Approximation. If S is a polyhedron, then our algorithm uses only linear subprograms. If we insist on obtaining linear subprograms even in the case of a general convex set S, then we must combine the above algorithm with polyhedral approximations to S. Assume for simplicity that $S := \{(x,y) \in \mathbb{R}^n \times \mathbb{R}^m | g(x,y) \leq 0\}$, where $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is convex, and subgradients of g are available. Then the above algorithm can be modified as follows.

At the start we set $S_0 := \mathbb{R}^n \times \mathbb{R}^m$. At iteration k we have a convex (possibly unbounded) polyhedron $S_k \subset \mathbb{R}^n \times \mathbb{R}^m$ such that $S \subset S_k$. Now we replace everywhere in iteration k the set S by S_k. Thus (x^B, y^B, u^B) is now determined as a solution of

$$\min\{f(x,y)|x \in X, y \in B, u \in B, (x,u) \in S_k\}.$$

$f(x^B, y^B) \leq f(x^B, u^B)$ continues to hold. Since (x^B, u^B) may not be feasible for (P), the rule for determining α_k must be modified as follows : With $(x^*, u^*) \in S$ fixed, for all $B \in \Gamma_k$ with $\beta(B) < \infty$ we let $\tau_B := \min\{f(\xi,\eta)|(\xi,\eta) \in [(x^*,u^*),(x^B,u^B)] \cap S\}$, and we set

$$\alpha_k := \min\{\alpha_{k-1}, \min\{\tau_B | B \in \Gamma_k, \beta(B) < \infty\}\}.$$

Again $f^* \leq \alpha_k$. Then Δ_k, B_k, β_k are determined as before. We have $\beta_k \leq f(x^k, u^k)$ and $\beta_k \leq f^*$.

If $\beta_k \geq f(x^k, u^k)$ and $(x^k, u^k) \in S$, then (x^k, u^k) solves (P), and the algorithm terminates. Otherwise we determine $c_k, B_k^-, B_k^+, \Gamma_{k+1}$ as before (however, if $f(x^k, \cdot) \equiv c_k$ on B_k, then we would have $B_k^- = B_k^+ = B_k$; to avoid this redundancy we can set $\Gamma_{k+1} := \Delta_k$ in this case). In addition : If $(x^k, u^k) \in S$, then $S_{k+1} := S_k$. If $(x^k, u^k) \notin S$, then

$$S_{k+1} := \{(x,y) \in S_k | g(x^k, u^k) + t_1^T(x - x^k) + t_2^T(y - u^k) \leq 0\},$$

where $(t_1, t_2) \in \partial g(x^k, u^k)$ - a subgradient of g at (x^k, u^k).

Since $S_k \supset S_{k+1} \supset S$ we have still $\beta_k \leq \beta_{k+1} \leq f^*$, and the convergence theorem remains valid. Indeed, if $(\overline{x}, \overline{u})$ is a cluster point of the sequence $\{(x^k, u^k)\}$, then the same proof as above shows that $f(\overline{x}, \overline{u}) \leq f^*$, and from $(x^k, u^k) \in S_k$ and the rule for constructing S_k follows by a standard argument that $g(\overline{x}, \overline{u}) \leq 0$. Hence $(\overline{x}, \overline{u})$ is feasible for (P), and thus optimal.

Indefinite Quadratic Programming. Problem (P) is equivalent to the (indefinite) quadratic programming problem with convex constraints, which we write in general form as

(Q) $$\min\{p^T x + x^T M x | x \in C\}.$$

Here $C \subset I\!R^n$ is a closed convex nonvoid set, $p \in I\!R^n$, and M is a real $(n \times n)$– matrix. We convert problem (Q) into the form (P) by introducing the function $f(x,y) := p^T x + x^T M y$ and writing (Q) as

$$(\tilde{\mathrm{P}}) \qquad \min\{f(x,y) | x \in I\!R^n, y \in I\!R^n, (x,y) \in S := (C \times C) \cap D\},$$

where $D := \{(x,y) \in I\!R^n \times I\!R^n | x = y\}$ is the diagonal in $I\!R^n \times I\!R^n$. The above algorithm can be applied to $(\tilde{\mathrm{P}})$ and thus solves (Q). At the start we need a compact convex polyhedron $X \subset I\!R^n$ containing an optimal solution of (Q). We set (fictitiously) $Y := X$. Then for $B \subset X$ problem $\mathrm{R}(B)$ with the above choice of S specializes to

$$\tilde{\mathrm{R}}(B) \qquad \min\{f(x,y) | x \in C \cap B, y \in B, u = x\}.$$

Clearly we may drop from $\tilde{\mathrm{R}}(B)$ the variable u altogether. We must then in the description of the algorithm replace u^B by x^B and u^k by x^k. Everything else remains unchanged. If the method does not terminate, then every cluster point of $\{x^k\}$ solves (Q).

If C is not a polyhedron, but is given by $C := \{x \in I\!R^n | g(x) \leq 0\}$, $g : I\!R^n \to I\!R$ convex, then we replace in iteration k the set S by $S_k := (C_k \times C_k) \cap D$, where C_k is a polyhedral approximation to C. At the start $C_0 := I\!R^n$, and

$$C_{k+1} := \begin{cases} C_k & \text{if } x^k \in C \\ \{x \in C_k | g(x^k) + t^T(x - x^k) \leq 0\} & \text{else,} \end{cases}$$

where $t \in \partial g(x^k)$. The τ_B needed in the modified rule for α_k should now satisfy

$$\tau_B := \min\{f(\xi,\xi) | \xi \in [x^*, x^B] \cap C\},$$

where $x^* \in C$ is fixed.

References

[1] Al-Khayyal F. A., Falk J. E.: Jointly constrained biconvex programming. Math. Oper. Res. **8** (1983), 273-286.

[2] Horst R., de Vries J., Thoai N. V.: On finding new vertices and redundant constraints in cutting plane algorithms for global optimization. Oper. Res. Lett. **7** (1988), 85-90.