The 7th Balkan Conference on Operational Research
"BACOR 05"
Constanta, May 2005, Romania

ON EXACT PENALTY FUNCTION FOR NONLINEAR PROGRAMMING PROBLEMS

CRISTINCA FULGA

Academy of Economic Sciences, Department of Mathematics, Bucharest, Romania

---

Abstract

In this paper an algorithm that uses sequential quadratic programming techniques in conjunction with a two-parameter penalty function is described. It is considered the nonlinear programming problem with interval constraints (NLP)
$\{\min f(x) | a \le c(x) \le b, x \in R^n\}$.
The approach taken is to replace the NLP by the more tractable problem of minimizing a non-differentiable penalty function chosen so that the solutions of the NLP are also solutions of the penalty function problem. The exact penalty function used in this paper is based on the infinity norm of the constraints violations.

1. PRELIMINARIES

The nonlinear programming problem considered is of the form:

$$(1)\quad \begin{cases} \min f(x) \\ a \le c(x) \le b \\ x \in R^n \end{cases}$$

where $f : R^n \to R, c : R^n \to R^m, c = (c_1, ..., c_m)$ are continuously differentiable functions and $a = (a_1, a_2, ..., a_m)^T \in R^m$, $b = (b_1, b_2, ..., b_m)^T \in R^m$.

Assumption. At each local minimizer of the nonlinear programming problem (1) an appropriate constraint qualification is assumed to hold, thereby ensuring that any optimal point x^* of the nonlinear programming problem (1) satisfies the following Karush-Kuhn-Tucker conditions: there exists a vector of Lagrange multipliers $\lambda^* = (\lambda_1^*, \lambda_2^*) \in R^m \times R^m$, where $\lambda_1^* = (\lambda_{11}^*, \lambda_{12}^*, ..., \lambda_{1m}^*) \in R^m$,

$\lambda_2^* = \left(\lambda_{21}^*, \lambda_{22}^*, \ldots, \lambda_{2m}^*\right) \in R^m$ such that

$$(2)\quad \begin{cases} c_i\left(x^*\right) - b_i \le 0\,; \quad \lambda_{1i}^* \ge 0\,; \quad \lambda_{1i}^* \cdot \left(c_i\left(x^*\right) - b_i\right) = 0\,; \quad i = \overline{1,m} \\ a_i - c_i\left(x^*\right) \le 0\,; \quad \lambda_{2i}^* \ge 0\,; \quad \lambda_{2i}^* \cdot \left(a_i - c_i\left(x^*\right)\right) = 0\,; \quad i = \overline{1,m} \\ \nabla f\left(x^*\right) + \sum_{i=1}^{m} \lambda_{1i}^* \cdot \nabla c_i\left(x^*\right) + \sum_{i=1}^{m} \lambda_{2i}^* \cdot \left(-\nabla c_i\left(x^*\right)\right) = 0. \end{cases}$$

2. THE PENALTY FUNCTION PROBLEM

The nonlinear programming problem is not solved directly; instead a non-differentiable exact penalty function Φ is minimized, where the exact penalty function is constructed so that local minimizers of the nonlinear programming problem are also local minimizers of the penalty function Φ. The penalty function is

(3) $\Phi(x) = f(x) + \mu \cdot \theta(x) + \frac{1}{2}\nu \cdot \theta^2(x),$ with $\mu > 0, \nu \ge 0$ where the degree of infeasibility $\theta(x)$ is defined as

(4) $\theta(x) = \max\limits_{1 \le i \le m}\left\{ \left[c_i(x) - b_i\right]_+ ; \left[a_i - c_i(x)\right]_+ \right\}$, and

$\left[c_i(x) - b_i\right]_+ = \max\left\{0; c_i(x) - b_i\right\}.$

The penalty function Φ may be viewed as a hybrid of a quadratic penalty function based on the infinity norm and the single parameter exact penalty function of [5], [6] and [9]. Clearly θ is continuous $\forall x \in R^n$, but it is usually not differentiable for some x. However, the directional derivative

$D_p\theta(x) = \lim\limits_{\substack{\alpha \to 0 \\ \alpha > 0}} \frac{\theta(x + \alpha \cdot p) - \theta(x)}{\alpha}$ exists for any $x, p \in R^n$. The definition (4) imply that $\forall x, p \in R^n$

$$D_p\theta(x) = \begin{cases} \max\left\{ \max\limits_{i \in I_1(x)} p^T \nabla c_i(x)\,;\ \max\limits_{i \in I_2(x)} p^T\left(-\nabla c_i(x)\right) \right\}, \\ \qquad\qquad \text{if } \theta(x) > 0,\ I(x) \ne \varnothing \\ \max\left\{ \max\limits_{i \in I_1(x)} \left(p^T \nabla c_i(x)\right)_+\,;\ \max\limits_{i \in I_2(x)} \left(p^T\left(-\nabla c_i(x)\right)\right)_+ \right\}, \\ \qquad\qquad \text{if } \theta(x) = 0,\ I(x) \ne \varnothing \\ 0 \quad \text{if } I(x) = \varnothing \end{cases}$$

where

$$\begin{cases} I(x) = I_1(x) \cup I_2(x), \\ I_1(x) = \left\{ i \,\middle|\, c_i(x) - b_i = \theta(x) \right\},\ I_2(x) = \left\{ i \,\middle|\, a_i - c_i(x) = \theta(x) \right\}. \end{cases}$$

Definition 2.1 For fixed values of $\mu > 0$ and $\nu \geq 0$, a point x^* is a critical point of Φ if and only if for all $p \in R^n$ the directional derivative $D_p\Phi(x^*)$ is non-negative.

Definition 2.2 The solution set of the penalty function problem with fixed values for $\mu > 0, \nu \geq 0$ is defined as the set of critical points of Φ.

Theorem 2.1 Let x^* be an optimal solution of the nonlinear programming problem (1) at which Karush-Kuhn-Tucker conditions hold and let $\lambda^* = (\lambda_1^*, \lambda_2^*) \in R^m \times R^m$ be a vector of Lagrange multipliers satisfying these conditions for which $\|(\lambda_1^*, \lambda_2^*)\|_1$ is minimal. If $\mu > \|(\lambda_1^*, \lambda_2^*)\|_1$ then x^* is a critical point of Φ. Conversely, if x^* is both feasible and a critical point of Φ for some $\mu > 0, \nu \geq 0$, then x^* is a Karush-Kuhn-Tucker point of the nonlinear programming problem (1).

3. DETERMINING DESCENT DIRECTIONS

In order to determine a suitable descent direction at the k-th iterate, a continuous piecewise quadratic approximation to Φ near the current point is defined:

$$\psi^k(p) = f(x^k) + p^T \cdot \nabla f(x^k) + \frac{1}{2} p^T \cdot H^k \cdot p + \mu^k \cdot \zeta(p) + \frac{1}{2}\nu^k \cdot \zeta^2(p),$$ where

$$\zeta(p) = \max_{1 \leq i \leq m}\left\{0\,; c_i(x^k) - b_i + p^T \nabla c_i(x^k); a_i - c_i(x^k) - p^T \nabla c_i(x^k)\right\}$$

and H^k is positive definite. Clearly ψ^k is strictly convex in *p*, and the level set $\{p \in R^n | \psi^k(p) \leq \psi^k(0)\}$ is bounded for all $\mu > 0, \nu \geq 0$. Thus, ψ^k has an unique global minimizer p^k which also solves the quadratic programming problem:

$$(P^k)\begin{cases} \min\limits_{p,\zeta} p^T \nabla f(x^k) + \frac{1}{2} p^T \cdot H^k \cdot p + \mu^k \cdot \zeta + \frac{1}{2}\nu^k \cdot \zeta^2 \\ c_i(x^k) - b_i + p^T \nabla c_i(x^k) \leq \zeta, \quad i = \overline{1,m} \\ a_i - c_i(x^k) + p^T(-\nabla c_i(x^k)) \leq \zeta, \quad i = \overline{1,m} \\ \zeta \geq 0. \end{cases}.$$

Theorem 3.1 Let (p^k, ζ^k) be the unique solution of the quadratic programming problem, with H^k positive definite. Let $(\lambda_1^k, \lambda_2^k)$ denote an optimal Lagrange multiplier vector, which need not be unique, for which $\|(\lambda_1^k, \lambda_2^k)\|_1$ is least. If

$$p^k \neq 0$$
$$\zeta^k \leq \theta(x^k)$$
$$\mu + \nu\theta(x^k) \geq \|(\lambda_1^k, \lambda_2^k)\|_1$$

then p^k is a descent direction for Φ at x^k.

The following algorithm is based on the results of the preceding sections.

4. EXACT PENALTY FUNCTION ALGORITHM.

For purposes of ensuring convergence, the following bound is imposed at each iteration:

$\|p^k\|_\infty \le S_{bound}$.

4.1 INITIALIZATION

$k=1$	$\mu^1=1$	$\nu^1=1$	$H^1=I$	
	$\varepsilon=10^{-5}$	$\rho=0.02$	$\delta=10^{-8}$	
	$S_{bound}=10^{10}$	$\theta_{cross}=1$	$\theta_{cap}=100$	
	$k_1=1.2$	$k_2=1.5$	$k_3=1.2$	$k_4=4$.

4.2 UPDATE *H* AND THE PENALTY PARAMETERS.

This step is omitted from the first iteration. *H* is updated using The Broyden-Fletcher-Goldfarb-Shanno update provided this maintains positive definiteness; otherwise *H* is not updated. The penalty parameters are updated as follows:

(i) If $\theta^k \le \theta_{cross}$ and $\mu^k < k_1\|\lambda^k\|_1$ then $\mu^{k+1}=k_2\|\lambda^k\|_1$, $\nu^{k+1}=\nu^k$.

(ii) If $\theta^k > \theta_{cross}$ and $\mu^k+\nu^k\theta^k < k_3\|\lambda^k\|_1$ then $\mu^{k+1}=\mu^k$, $\nu^{k+1}=\dfrac{k_4\|\lambda^k\|_1-\mu^k}{\theta^k}$.

4.3 SOLVE THE (P^K) PROBLEM

If $\theta^k > \theta_{cap}$ then the capping constraint $\zeta \le \theta^k$ is also imposed. Then this problem is solved. If the capping constraint is not active at the (P^k)'s solution, then the algorithm proceeds directly to Step 4. Otherwise, the penalty parameters are updated as described in Step 2, except that $\|\lambda^k\|_1$ is replaced by $\mu^k+\nu^k\theta^k+|\xi|$, where ξ is the Lagrange multiplier of the capping constraint. The (P^k) problem is then solved again.

4.4 ATTEMPT THE PROPOSED STEP

If (i) $\Phi(x^k)-\Phi(x^k+p^k)\ge\rho\left[\Psi^k(0)-\Psi^k(p^k)\right]$

(ii) either the penalty parameters were not altered in Step 3 or the inequality $\theta(x^k+p^k)\le\theta(x^k)$ is satisfied, then the proposed step is accepted and the algorithm proceeds to step 7. Otherwise, the execution continues at the next step.

4.5 CALCULATE THE MARATOS EFFECT CORRECTION VECTOR

Solve the following quadratic problem for the second order correction t^k :

$$\begin{cases} \min_{t \in R^n} \| t \|_2^2 \\ c_i\left(x^k + p^k\right) - b_i + t^T \nabla c_i\left(x^k\right) \geq 0 \\ a_i - c_i\left(x^k + p^k\right) - t^T \nabla c_i\left(x^k\right) \geq 0, \ \forall i \in T \end{cases}$$

where T is the set of indices of the constraints active at the $\left(P^k\right)$'s solution in Step 3. If $\| t^k \|_2 \geq \| p^k \|_2$ then set $t^k = 0$.

4.6 ARC SEARCH

Consider successive values of the sequence $1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots$ as trial values of α. If $t^k = 0$, then omit the first member of the sequence. Accept the first trial value which satisfies

(i) $\Phi\left(x^k\right) - \Phi\left(x^k + q^k(\alpha)\right) \geq \rho\alpha\left[\Psi^k(0) - \Psi^k\left(p^k\right)\right]$

where $q^k(\alpha) = \alpha p^k + \alpha^2 t^k$

(ii) If the penalty parameters were altered in Step 3, then the step $q^k(\alpha)$ is also required to satisfy the condition $\theta\left(x^k + q^k(\alpha)\right) \leq \theta^k$. After a satisfactory value of α has been found, set $x^{k+1} = x^k + q^k(\alpha)$.

4.7 CHECK THE STOPPING CONDITIONS

The algorithm halts if either the length of the previous step $\left\| x^k - x^{k-1} \right\|_2 \leq \delta$ or both of the following conditions hold:

(i) $\theta^k < \varepsilon$

(ii) $\left\| \nabla f\left(x^k\right) + \sum_{i \in A^k} \lambda_i^k \nabla c_i\left(x^k\right) - \sum_{j \in B^k} \lambda_j^k \nabla c_j\left(x^k\right) \right\|_2 < \varepsilon$

where $A^k = \left\{ i \middle| \ \left| c_i\left(x^k\right) - b_i \right| < 10^{-5} \right\}, B^k = \left\{ \mathrm{j} \middle| \ \left| a_j - c_j\left(x^k\right) \right| < \varepsilon \right\}$. Otherwise, k is incremented, and the algorithm proceeds to Step 2.

The convergence properties of the algorithm are summarized in the following:

Theorem 4.1 Assume that

- the sequence of iterates $\left(x^k\right)_k$ is bounded in norm;
- the sequence of matrices $\left(H^k\right)_k$ generated is bounded in norm;
- the penalty parameters μ, ν are altered only a finite number of times.

Then, every cluster point of the sequence of iterates $\left(x^k\right)_k$ generated by the algorithm is a critical point of $\Phi(x; \mu, \nu)$ where μ, ν are at their final values.

5. CONCLUDING REMARKS

The purpose of this paper is to show that there are some advantages to be gained from using a two-parameter exact penalty function based on the infinity norm of constraint violations. This function has an advantage over one-norm based exact penalty function in that only the gradients of the most violated constraints need be calculated in order to find a search direction: for one-norm exact penalty functions, the gradients of all active and violated constraints may be required.

The algorithm generates convergent sequences under mild conditions; it is effective in practice and the use of the second penalty parameter significantly reduces the effort required to solve constraint nonlinear programs.

BIBLIOGRAPHY

[1] Coope I. D., Price C. J., "A Two-Parameter Exact Penalty Function for Nonlinear Programming", Journal of Optimization Theory and Applications, Vol. 83, No. 1, pp. 49-61, 1994;

[2] Fulga C., "Penalty Concepts by Means of the Separation Functions in the Image Space", Analele Universitatii "Dunarea de Jos", Galati, Vol. 1, Fascicula II, Supp. Tome XVI (XXI), pp. 115-122, 1998;

[3] Han S. P., "A Globally Convergent Method for Nonlinear Programming", Journal of Optimization Theory and Applications, Vol. 22, pp. 297-309, 1977;

[4] Han S. P., Mangasarian O. L., "Exact Penalty Functions in Nonlinear Programming", Mathematical Programming 17, pp. 251-269, 1979;

[5] Mayne D. Q., Polak E., "A Superlinearly Convergent Algorithm for Constrained Optimization Problems", Mathematical Programming Study, Vol. 16, pp. 45-61, 1982;

[6] Pantoja J. F. E. De O., Mayne D. Q., "Exact Penalty Function Algorithm with Simple Updating of the Penalty Parameter", Journal of Optimization Theory and Applications, Vol. 69, pp. 441-467, 1991;

[7] Pietrzykowski T., "An Exact Potential Method for Constrained Maxima", SIAM Journal on Numerical Analysis 6, pp. 299-304, 1969;

[8] Preda V., "Optimality Conditions for a Class of Mathematical Programming Problems", Bull. Soc. Sci. Math. Roumanie, 34 (82), pp. 343-350, 1990;

[9] Sahba M., "A Globally Convergent Algorithm for Nonlinearly Constrained Optimization Problems", Journal of Optimization Theory and Applications, Vol. 52, pp. 291-307, 1987;

[10] Zangwill W.I., "Nonlinear Programming via Penalty Functions", Management Science 13, pp. 344-358, 1967.