A New Implementation of Fletcher's Exact Merit Function for Nonlinear Optimization (12rit180)

Michael P. Friedlander (Vancouver, Canada),
Dominique Orban (Montréal, Canada)

May 27–June 3, 2012

1 Overview

Consider the general nonlinear optmization problem

$$\text{minimize} \quad f(x) \quad \text{subject to} \quad c(x) = 0, \tag{1}$$

where $f : \Re^n \to \Re$ and $c : \Re^n \to \Re^m$ are twice-continuously differentiable functions. A fundamental challenge in developing iterative algorithms for this solution of this problem is the inherent tradeoff between minimizing the objective function f, and satisfying the constraint $c(x) = 0$. Penalty functions encapsulate these competing demands by providing a measure of progress towards the solution, and provide for a way of transforming the constrained (and difficult) problem into an unconstrained (and easier problem). A penalty function may be either *exact*—i.e., its unconstrained minimizer coincides with a solution of (1)—or *inexact*—i.e., its unconstrained minimizer is only an approximate solution, and an infinite sequence of unconstrained problems must be solved. Exact-penalty functions, however, are generally nonsmooth, which entail a host of complicating factors.

The method that we consider is rooted in a smooth and exact penalty function first proposed by [2] for equality-constrained problems. There has been a long-held view that that Fletcher's penalty function is not practicable because it is costly to compute; see comments by Bertsekas (1976), [1], and [3]. Our aim in this project is to challenge that notion, and to demonstrate that the computational kernels are no more expensive than other well-accepted methods for nonlinear optimization, such as sequential quadratic programming.

The penalty function that we consider for (1)

$$\phi_\sigma(x) := f(x) - c(x)^T y_\sigma(x), \tag{2}$$

where $y_\sigma(x)$ are Lagrange multiplier estimates defined as the solution of the least-squares problem

$$\text{minimize}_y \ \frac{1}{2}\|A(x)y - g(x)|_2^2 + \sigma c(x)^T y, \tag{3}$$

where we used the notation

$$g(x) := \nabla f(x), \quad A(x) := \nabla c(x), \quad \text{and} \quad Y_\sigma(x) := \nabla y_\sigma(x). \tag{4}$$

Note that A and Y_σ are n-by-m matrices. In our initial exploration, we make the simplifying assumption that $A(x)$ is full rank for all x. Hence, the solution $y_\sigma(x)$ and its gradient $Y_\sigma(x)$ are uniquely defined.

1.1 Notation

Let $H(x) = \nabla^2 f(x)$ and $H_i(x) = \nabla^2 c_i(x)$. We also define

$$g_\sigma(x) := g(x) - A(x)y_\sigma(x), \tag{5a}$$

$$H_\sigma(x) := H(x) - \sum_{i=1}^{m} [y_\sigma(x)]_i H_i(x), \tag{5b}$$

which we recognize as the gradient and Hessian, respectively, of the usual Lagrangian function $L(x, y)$ evaluated at x and $y(x)$. Also, define the matrix operators

$$S(x, v) := \nabla_x [A(x)^T v] = \nabla_x \begin{bmatrix} \nabla c_1(x)^T v \\ \vdots \\ \nabla c_m(x)^T v \end{bmatrix} = \begin{bmatrix} v^T H_1(x) \\ \vdots \\ v^T H_m(x) \end{bmatrix};$$

$$T(x, w) := \nabla_x [A(x)w] = \nabla_x \left[\sum_{i=1}^{m} \nabla c_i(x) w_i \right] = \sum_{i=1}^{m} w_i H_i(x),$$

for all $v \in \Re^n$ and $w \in \Re^m$. In particular, note that for all $u \in \Re^m$, all $v \in \Re^n$ and all $w \in \Re^m$,

$$S(x, v)^T u = \left[\sum_{i=1}^{m} u_i H_i(x) \right] v = T(x, u)v\,,$$

$$S(x, v)w = \begin{bmatrix} v^T H_1(x) w \\ \vdots \\ v^T H_m(x) w \end{bmatrix},$$

and

$$T(x, w)^T v = \left[\sum_{i=1}^{m} w_i H_i(x) \right] v = T(x, w)v\,.$$

If A has full rank at some feasible x^*, the operators

$$P = A(x^*)\big(A(x^*)^T A(x^*)\big)^{-1} A(x^*)^T \quad \text{and} \quad \bar{P} := I - P \tag{6}$$

define orthogonal projectors onto range$(A(x^*))$ and its complement, respectively.

The gradient and Hessian of ϕ_σ may be written as

$$\nabla \phi_\sigma(x) = g_\sigma(x) - Y_\sigma(x)c(x), \tag{7a}$$

$$\nabla^2 \phi_\sigma(x) = H_\sigma(x) - A(x)Y_\sigma(x)^T - Y_\sigma(x)A(x)^T - \nabla \left[Y_\sigma(x)c \right], \tag{7b}$$

where $H(x) = \nabla^2 f(x)$ and $H_i(x) = \nabla^2 c_i(x)$ are the Hessians of the objective and each constraint function, respectively. The last term $\nabla_x [Y_\sigma(x)c]$ in the expression for $\nabla^2 \phi_\sigma$ purposefully drops the argument on c to emphasize that this gradient is made on the product $Y(x)c$, with $c := c(x)$ held fixed. This term involves third derivatives of f and c, and as Fletcher shows, it is both convenient and computationally efficient to ignore this term; we leave this term unexpanded.

2 Scientific Progress Made

During our workshop we established a better understanding of how an algorithm might dynamically update the penalty parameter. In this section we give explicit expressions for threshold values of the penalty parameter.

It follows directly from the gradient and Hessian expressions (7) of ϕ_σ and the definition (3) of y_σ that the following definitions are equivalent to the usual optimality conditions defined via the Lagrangian function; see, e.g., [3, Ch. 12].

First-order KKT point A point x^* is a first-order KKT point of (1) if for any $\sigma \geq 0$ the following hold:

$$c(x^*) = 0, \tag{8a}$$
$$\nabla \phi_\sigma(x^*) = 0. \tag{8b}$$

The elements of $y^* := y_\sigma(x^*)$ comprise the vector of Lagrange multipliers of (1) associated to x^*.

We can similarly derive second-order optimality conditions based on the Hessian of ϕ_σ.

Second-order KKT point The first-order KKT point x^* satisfies the second-order necessary KKT condition for (1) if for any $\sigma \geq 0$

$$p^T \nabla^2 \phi_\sigma(x^*) p \geq 0 \quad \text{for all } p \text{ such that } A(x^*)^T p = 0. \tag{9}$$

The condition is sufficient if

$$p^T \nabla^2 \phi_\sigma(x^*) p > 0 \quad \text{for all } p \neq 0 \text{ such that } A(x^*)^T p = 0. \tag{10}$$

The second-order KKT condition holds for all $\sigma \geq 0$, and only requires the correct curvature of ϕ_σ in directions in the tangent space of the constraints. However, we can explicitly derive a threshold value of σ that causes a stationary point of ϕ_σ to be feasible, or causes ϕ_σ to be locally convex at a second-order KKT point x^*. For a given first or second-order KKT point x^* for (1), we define

$$\sigma^* := \frac{1}{2} \|P H_\sigma(x^*) P\|. \tag{11}$$

Theorem 1. *If $\nabla \phi_\sigma(\bar{x}) = 0$ for some $\bar{x}$, then*

$$\sigma > \|A(\bar{x})^T Y_\sigma(\bar{x})\| \quad \Longrightarrow \quad g(\bar{x}) = A(\bar{x}) y_\sigma(\bar{x}), \quad c(\bar{x}) = 0. \tag{12a}$$

If x^* is a first-order KKT point for (1)*, then*

$$\sigma \geq \|A(x^*) Y_\sigma(x^*)^T\| \quad \Longrightarrow \quad \sigma \geq \sigma^*. \tag{12b}$$

If x^* is a second-order necessary KKT point for (1)*, then*

$$\nabla^2 \phi_\sigma(x^*) \succeq 0 \quad \Longleftrightarrow \quad \sigma \geq \sigma^*, \tag{12c}$$

If x^* is second-order sufficient, then the inequalities in (12c) *hold strictly.*

Proof. We prove, in order, (12a), (12c), and (12b). First note that for any x, the vector of Lagrange multiplier estimates $y_\sigma(x)$ satisfies the linear system

$$A(x)^T A(x) y_\sigma(x) = A(x)^T g(x) - \sigma c(x), \tag{13}$$

which define necessary and sufficient optimality conditions for (3).

Proof of (12a). The condition $\nabla \phi_\sigma(\bar{x}) = 0$ implies that

$$g(\bar{x}) = A(\bar{x}) y_\sigma(\bar{x}) + Y_\sigma(\bar{x}) c(\bar{x}).$$

Using this equation in (13) evaluated at $\bar{x}$, yields, after simplifying,

$$A(\bar{x})^T Y_\sigma(\bar{x}) c(\bar{x}) = \sigma c(\bar{x}).$$

Taking norms of both sides and using the triangle inequality gives $\sigma \|c(\bar{x})\| \leq \|A(\bar{x})^T Y_\sigma(\bar{x})\| \, \|c(\bar{x})\|$, which immediately implies that $c(\bar{x}) = 0$. The condition $\nabla \phi_\sigma(\bar{x}) = 0$ then becomes $g_\sigma(\bar{x}) = 0$, which completes the proof of (12a).

Proof of (12c). We first obtain an expression for $A(x) Y_\sigma(x)^T$ in terms of $H_\sigma(x)$ by differentiating both sides of (13), which yields

$$S(x, A(x) y_\sigma(x)) + A(x)^T \left[T(x, y_\sigma(x)) + A(x) Y_\sigma(x)^T \right] = S(x, g(x)) + A(x)^T [H(x) - \sigma I].$$

Isolating the term $A(x)^T A(x) Y_\sigma(x)$ on the left-hand side, and using the linearity of S in its second argument, we rearrange terms to arrive at

$$A(x)^T A(x) Y(x)^T = S(x, g(x) - A(x) y_\sigma(x)) + A(x)^T [H(x) - T(x, y_\sigma(x)) - \sigma I].$$

Using the definitions (5), this can be expressed as

$$A(x)^T A(x) Y(x)^T = A(x)^T [H_\sigma(x) - \sigma I] + S(x, g_\sigma(x)). \tag{14}$$

Because x^* satisifes the first-order conditions (8), $g_\sigma(x^*) = 0$, and it follows from the above equation and the definition of P that

$$A(x^*) Y_\sigma(x^*)^T = P(H_\sigma(x^*) - \sigma I). \tag{15}$$

We substitute this equation into (7b) and use the relation $P + \bar{P} = I$ to obtain the expression

$$\nabla^2 \phi_\sigma(x^*) = \bar{P} H_\sigma(x^*) \bar{P} - P H_\sigma(x^*) P + 2\sigma P. \tag{16}$$

Because $\|P\| \leq 1$, the relationship (12c) follows.

Proof of (12b). Again using properties of the projector P, it follows from (15) that

$$\begin{aligned}
\sigma &\geq \|A(x^*) Y_\sigma(x^*)^T\| \\
&= \|P(H_\sigma(x^*) - \sigma I)\| \\
&\geq \|P(H_\sigma(x^*) - \sigma I) P\| \\
&\geq \|P H_\sigma(x^*) P\| - \sigma \|P\| \\
&\geq 2\sigma^* - \sigma.
\end{aligned}$$

Thus, $\sigma \geq \sigma^*$, as required. □

3 Outcome of the Meeting

Theorem 1 gives us a concrete method for testing if a candidiate threshold parameter is sufficiently large. Other crucial items completed during this workshop included

- A method for computing the gradient and Hessians in (7) that has a cost of factorizing only a single projection matrix;
- Extensions of the penalty function to handle more general constraints, including affine and bound constraints.

References

[1] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. MPS-SIAM Series on Optimization. Society of Industrial and Applied Mathematics, Philadelphia, 2000.

[2] R. Fletcher. A class of methods for nonlinear programming with termination and convergence properties. In J. Abadie, editor, *Integer and nonlinear programming*, pages 157–175. North-Holland, Amsterdam, 1970.

[3] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.