Augmented Lagrangian for Least-Squares

Pierre Borie[*]

Notations

We consider constrained nonlinear least squares problems of the form

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \frac{1}{2}\|r(x)\|^2 \\ \text{s.t.} \quad & h(x) = 0 \\ & \langle c_i, x\rangle = b_i \quad i = 1,\dots,m \\ & l \leq x \leq u, \end{aligned} \tag{1}$$

where $r\colon \mathbb{R}^n \to \mathbb{R}^d$ and $h\colon \mathbb{R}^n \to \mathbb{R}^t$ are assumed to be nonlinear, potentially non convex, continuously differentiable functions, $\langle\cdot,\cdot\rangle$ is the canonical inner product and $\|\cdot\|$ its induced euclidean norm, c_i are m independent vectors of $\mathbb{R}^n$, $(m \leq n)$, $b = (b_1,\dots,b_m)^T \in \mathbb{R}^m$ and ℓ and u are vectors in $\mathbb{R}^n$. Without loss of generality, some components of the latter two vectors can be set to $\pm\infty$ for unbounded parameters. In the context of least squares problems, components r_i of the function r are often denoted as the residuals.

We will also refer to the linear constraints using the following set notation

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid Cx = b,\ l \leq x \leq u\}, \tag{2}$$

where C is the matrix whose columns are the vectors c_i. By linear independence of those vectors, C is a full rank matrix.

The method presented in this report is conceived around an Augmented Lagrangian (AL) function where only the violation of the nonlinear constraints is penalized. Its expression is given by

$$\Phi_A(x,\lambda,\mu) := \frac{1}{2}\|r(x)\|^2 + \langle\lambda, h(x)\rangle + \frac{\mu}{2}\|h(x)\|^2, \tag{3}$$

We keep the linear constraints as is and only penalize the violation of the nonlinear constraints. One has the following expression of the gradient:

$$\nabla_x\Phi_A(x,\lambda,\mu) = J(x)^T r(x) + A^T\pi(x,\lambda,\mu), \tag{4}$$

with $\pi(x,\lambda,\mu) := \lambda + \mu h(x)$ is the first-order estimates of the Lagrange multipliers.

The Hessian is given by

$$\nabla^2_{xx}\Phi_A(x,\lambda,\mu) = J(x)^T J(x) + \mu A(x)^T A(x) + \sum_{i=1}^{d} r_i(x)\nabla^2 r_i(x) + \sum_{i=1}^{d} \nabla^2 h_i(x)\pi(x,\lambda,\mu). \tag{5}$$

[*] University of Montreal, Department of Computer Science and Operations Research, Montreal, QC, Canada

For fixed λ and μ, replacing the objective function with the AL gives the linearly constrained problem

$$\begin{aligned} \min_x \quad & \Phi_A(x, \lambda, \mu) \\ \text{s.t.} \quad & x \in \mathcal{X}. \end{aligned} \tag{6}$$

A local minimum x^* of (6) then satisfies the first-order necessary condition

$$x^* - P_{\mathcal{X}}(x^* - \nabla_x \Phi_A(x^*, \lambda, \mu) = 0,$$

where $P_{\mathcal{X}}$ denotes the projection operator onto the set $\mathcal{X}$.

1 The method

Let's consider a primal-dual iterate (x_k, λ_k) such that $x_k \in \mathcal{X}$ and a penalty parameter μ_k. Each iteration consists into computing an approximating minimizer x_{k+1} of (6) starting from x_k and such that

$$\|x_{k+1} - P_{\mathcal{X}}(x_{k+1} - \nabla_x \Phi_A(x_{k+1}, \lambda_k, \mu_k)\| \leq \omega_k, \tag{7}$$

where ω_k is a small tolerance. Index k means that we will adapt its value at each iteration (the closer from the solution, the small it will be). The iterate is computed by a trust region method. To do so, we first construct a quadratic model of the AL around x_k:

$$\mathcal{Q}_k(p) = \frac{1}{2} \langle p, H_k p \rangle + \langle g_k, p \rangle \,, \tag{8}$$

with $H_k \approx \nabla^2_{xx} \Phi_A(x_k, \lambda_k, \mu_k)$ and $g_k := \nabla_x \Phi_A(x_k, \lambda_k, \mu_k)$. We compute the step p_k by approximately minimizing the quadratic program

$$\begin{aligned} \min_p \quad & \mathcal{Q}_k(p) \\ \text{s.t.} \quad & x_k + p \in \mathcal{X} \\ & \|p\|_\infty \leq \Delta_k. \end{aligned} \tag{9}$$

Since we assumed x_k is feasible, it is sufficient that the step satisfies:

- $Cp = 0$
- $\bar{l} = \max(x_k - l, \Delta_k) \leq p \leq \min(u - x_k, \Delta_k) = \bar{u}$,

(the max and min are applied component-wise).

If the candidate step $x_k + p_k$ does not satisfy (7), we decrease the trust region radius Δ_k and restart the minimization procedure. Once the conditions is verified, we update the iterate by $x_{k+1} = x_k + p_k$ and then check the violation of the nonlinear constraints.

If the violation is lower than a given tolerance η_k, (depends on the iteration), than the step is fully accepted. We thus update the Lagrange multipliers by the first-order estimates, slightly increase the penalty parameter μ_k and decrease the tolerances ω_k, η_k. If the constraints violation condition is not satisfied, this means that the next iteration should emphasis on feasibility. We thus keep the Lagrange multipliers and tolerances but increase more significantly the penalty parameter.

The procedure is outlined is algorithm 1

Algorithm 1 Outer iteration of basic AL algorithm with trust region

```
Step 0: Initialization Choose a feasible starting point $x_0$, vector of Lagrange multipliers estimates $\lambda_0$.
    Set the penalty parameter to $\mu_0 > 1$, the tolerances $\omega_0 > 0, \eta_0 > 0$. Start the iteration index $k$ at 0.
Step 1: Inner Iteration Starting from $x_k$, approximately solve
    $$\min_{x \in \mathcal{X}} \Phi_A(x, y_k, \mu_k)$$
    by computing $x_{k+1}$ verifying condition (7).
    If $\|h(x_k)\| \leq \eta_k$, execute Step 2.
    Otherwise, execute Step 3.
Step 2: Iterate Update Update $\lambda_{k+1} \leftarrow \lambda_k + \mu_k h(x_k)$ and set $x^s_{k+1} \leftarrow x_k$.
    Choose new tolerances $\omega_{k+1}, \eta_{k+1}$ and new penalty parameter $\mu_{k+1} > \mu_k$.
    Increment $k$ and go back to Step 1.
Step 3: Adjustment of the Penalty Parameter Choose $\mu_{k+1}$ significantly greater than $\mu_k$ and new tolerances $\omega_{k+1}, \eta_{k+1}$.
    Leave the multipliers and tolerances unchanged: $\lambda_{k+1} \leftarrow \lambda_k$.
    Increment $k$ Go back to Step 1.
```

1.1 The inner iteration

For a point x, we define $\mathcal{A}(x) := \big\{ i \mid x_i \in \{\bar{l}_i, \bar{u}_i\} \big\}$ the indices of bounds constraints active at x. This set mixes the bounds from the problem and the trust region.

To compute the step, we follow the approach of Lin and Moré [1] and successively apply the conjugate gradient method and adaptively modify the active set if we encounter a bound or the trust region.

In order to do so, we compute $m + 1$ minor iterates $x_{k,1}, \ldots, x_{k,m+1}$ with $x_{k,1} = x_k$ and $p_k = x_{k,m+1} - x_k$. Each minor iterate is updated by $x_{k,j+1} = xk, j + p_{k,j}$, where $p_{k,j}$is an approximate solution the QP

$$\begin{aligned} \min_{p} \quad & \mathcal{Q}_k(p) \\ \text{s.t.} \quad & Cp = 0 \\ & p_i = 0, \quad i \in \mathcal{A}(x_{k,j}), \end{aligned} \tag{10}$$

starting from $x_{k,j}$. Subproblem (10) is solved by the projected conjugate gradient method with early stopping if we encounter a bound or the trust region. The next active set $\mathcal{A}(x_{k,j+1})$ is formed after $\mathcal{A}(x_{k,j})$ and the bounds newly active.

1.2 Questions

- Would it be interesting to start the inner from the Cauchy point? If yes how to compute it, i.e. how to project the steepest direction on the feasible set?
- Which approximation of the Hessian $\nabla^2_{xx} \Phi_A(x_k, \lambda_k, \mu_k)$?
- Rules for the update of the penalty parameter, the tolerances, the trust region radius?

References

[1] C.-J. Lin and J.J Moré. Newton's method for large bound-constrained optimization problems. *SIAM Journal on Optimization*, 9(4):1100–1127, 1999. doi: 10.1137/S1052623498345075.