Cahier du GERAD G-2019-00

Adaptive Regularization with Inexact Gradients

Tiphaine Bonniot de Ruisselet[2] and Dominique Orban[1,2⋆]

[1] Department of Mathematics and Industrial Engineering, École Polytechnique, Montréal, QC, Canada.
[2] GERAD, Montréal, QC, Canada. dominique.orban@gerad.ca

1 Introduction

{sec:introduction}

We consider the problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x), \tag{1}$$

{{eq:unc}}

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable.

We consider the situation where it is possible to evaluate approximations of $\nabla f(x)$. Typically the cost of such approximations increases with the quality of the approximation. More specifically, we assume that it is possible to obtain $g(x, \omega_g) \approx \nabla f(x)$ using a user-specified relative error threshold $\omega_g > 0$, i.e.,

$$\|g(x, \omega_g) - g(x, 0)\| \le \omega_g \|g(x, \omega_g)\|, \quad \text{with } g(x, 0) = \nabla f(x). \tag{2}$$

{{eq:g-error}}

We use Householder notation throughout: capital Latin letters such as A, B, and H, represent matrices, lowercase Latin letters such as s, x, and y represent vectors in $\mathbb{R}^n$, and lowercase greek letter such as α, β and γ represent scalars.

2 Background and Assumptions

{sec:background}

2.1 Assumptions

{ass:f-bounded}

Assumption 1. *The function f is bounded below on $\mathbb{R}^n$, i.e., there exists κ_{low} such that $f(x) \ge \kappa_{low}$ for all $x \in \mathbb{R}^n$.*

{ass:f-C1}

Assumption 2. *The function f is continuously differentiable over $\mathbb{R}^n$.*

{ass:g-lipschitz}

Assumption 3. *The gradient of f is Lipschitz continuous, i.e., there exists $L > 0$ such that for all x, $y \in \mathbb{R}^n$, $\|\nabla f(x) - \nabla f(y)\| \le L \|x - y\|$.*

⋆ Research partially supported by NSERC Discovery Grant 299010-04.

2.2 Background on adaptive regularization

Consider the problem (1). Let $T(x,s)$ be the Taylor series of the function $f(x+s)$ at x truncated at the first order, i.e.

$$T(x,s) = f(x) + \nabla f(x)^T s.$$

From Birgin, Gardenghi, Martínez, Santos, and Toint (2017), we recall the following results implied by Taylor's theorem.

For all $x, s \in \mathbb{R}^n$,

$$f(x+s) - T(x,s) \le \tfrac{1}{2} L \|s\|^2, \tag{3}$$ {{eq:taylor-f-error}}

is it really 1/2 ?

$$\|\nabla f(x+s) - \nabla_s T(x,s)\| \le L\|s\|, \tag{4}$$ {{eq:taylor-g-error}}

where L is the Lipschitz constant presented in (Assumption 3).

This leads to considering at each iteration k the approximate Taylor series using the inexact gradient defined in 2.

$$\bar{T}_k(s) = f(x_k) + g(x_k, \omega_g^k)^T s.$$

The inequality (3), the Cauchy-Schwarz inequality and the tolerance on the inexact gradient (2) imply that, at each iteration k and for all $s \in \mathbb{R}^n$,

$$\begin{aligned} |f(x_k+s) - \bar{T}_k(s)| &\le |f(x_k+s) - T(x_k,s)| + |T(x_k,s) - \bar{T}_k(s)| \\ &\le |f(x_k+s) - T(x_k,s)| + |\nabla f(x_k)^T s - g(x_k,\omega_g^k)^T s| \\ &\le \tfrac{1}{2} L\|s\|^2 + \|\nabla f(x_k) - g(x_k,\omega_g^k)\| \ \|s\| \\ &\le \tfrac{1}{2} L\|s\|^2 + \omega_g^k \|g(x_k,\omega_g^k)\| \ \|s\|. \end{aligned} \tag{5}$$ {{eq:inexact-taylor-f-error}}

Similarly, using the inequality (4) and the tolerance on the inexact gradient (2), we have

$$\begin{aligned} \|\nabla f(x_k+s) - \nabla_s \bar{T}_k(s)\| \le& \|\nabla f(x_k+s) - \nabla_s T(x_k,s)\| + \|\nabla_s T(x_k,s) - \nabla_s \bar{T}_k(s)\| \\ \le& \|\nabla f(x_k+s) - \nabla_s T(x_k,s)\| + \|\nabla f(x_k) - g(x_k,\omega_g^k)\| \\ \le& L\|s\| + \omega_g^k \|g(x_k,\omega_g^k)\|. \end{aligned} \tag{6}$$ {{eq:inexact-taylor-g-error}}

In order to describe our algorithm, we also define the approximate regularized Taylor series

$$m_k(s) = \bar{T}_k(s) + \tfrac{1}{2}\sigma_k \|s\|^2, \tag{7}$$ {{eq:model}}

whose gradient is

$$\nabla_s m_k(s) = \nabla_s \bar{T}_k(s) + \sigma_k s = g(x_k,\omega_g^k) + \sigma_k s,$$

where σ_k is the regularization factor updated at each iteration according to the algorithm's mechanisms described in section 3.

3 Complete Algorithm

{sec:algorithm}

We summarize the complete process as Algorithm 3.1.

Algorithm 3.1 Adaptive Regularization with inexact gradients {alg:regularization-inexact}

Require: $x_0 \in \mathbb{R}^n$

1: Choose the accuracy level $\epsilon > 0$, the initial regularization parameter $\sigma_0 > 0$, and the constants $\eta_1, \eta_2, \gamma_1, \gamma_2, \gamma_3$ and $\sigma_{\min}$ such that

$$\sigma_{\min} \in \;]0, \sigma_0],\; 0 < \eta_1 < \eta_2 < 1 \text{ and } 0 < \gamma_1 < 1 < \gamma_2 < \gamma_3. \tag{8}$$ {{ar:parameters}}

Set $k = 0$.

2: Choose ω_g^k such that $0 < \omega_g^k \leq \frac{1}{\sigma_k}$ and compute $g(x_k, \omega_g^k)$ such that (2) holds. If $\|g(x_k, \omega_g^k)\| \leq \frac{\epsilon}{1+\omega_g^k}$, terminate with the approximate solution $x_\epsilon = x_k$.

3: Compute the step $s_k = -\frac{1}{\sigma_k} g(x_k, \omega_g^k)$.

4: Evaluate $f(x_k + s_k)$ and define

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{\bar{T}_k(0) - \bar{T}_k(s_k)} = \frac{f(x_k) - f(x_k + s_k)}{\frac{1}{\sigma_k}\|g(x_k, \omega_g^k)\|^2}. \tag{9}$$ {{ar:ratio}}

If $\rho_k \geq \eta_1$, then define $x_{k+1} = x_k + s_k$. Otherwise, define $x_{k+1} = x_k$.

5: Set

$$\sigma_{k+1} \in \begin{cases} [\max(\sigma_{\min}, \gamma_1 \sigma_k), \sigma_k] & \text{if } \rho_k \geq \eta_2, \\ [\sigma_k, \gamma_2 \sigma_k] & \text{if } \rho_k \in [\eta_1, \eta_2[, \\ [\gamma_2 \sigma_k, \gamma_3 \sigma_k] & \text{if } \rho_k < \eta_1. \end{cases} \tag{10}$$ {{ar:update}}

Increment k by one and go to step 1 if $\rho_k \geq \eta_1$ or to step 2 otherwise.

Note that the tolerance (2) imposed on the relative error on the gradient insures that at each iteration k

$$\|\nabla f(x_k)\| \leq \|\nabla f(x_k) - g(x_k, \omega_g^k)\| + \|g(x_k, \omega_g^k)\| \leq (1 + \omega_g^k)\|g(x_k, \omega_g^k)\|.$$

Thus when the termination occurs, $\|g(x_k, \omega_g^k)\| \leq \frac{1}{1+\omega_g^k}\epsilon$, hence $\|\nabla f(x_\epsilon)\| \leq \epsilon$ and the first order critical point x_ϵ satisfies the desired accuracy.

4 Convergence and Complexity Analysis

{sec:convergence}

The following is the adaptation of the general properties presented by Birgin et al. (2017) to a second order model with inexact gradients.

We deduce from (Birgin et al., 2017, Lemma 2.2) an upper bound on the regularization parameter σ_k.

{lem:sigma-bounded}

Lemma 1. *For all $k \geq 0$,*

$$\sigma_k \leq \sigma_{\max} = \max\left[\sigma_0, \frac{\gamma_3(\frac{1}{2}L + 1)}{1 - \eta_2}\right].$$

Proof. Using the definition of ρ_k (9), and the fact that the error on the inexact Taylor series is bounded (5), we may deduce that

$$|\rho - 1| = \frac{|f(x_k + s_k) - \bar{T}_k(s_k)|}{|\bar{T}_k(0) - \bar{T}_k(s_k)|} \leq \frac{\frac{1}{2}L + \omega_g^k \sigma_k}{\sigma_k}.$$

Since we require in step 2 that the tolerance on the inexact gradient ω_g^k be less or equal to $\frac{1}{\sigma_k}$, it comes

$$|\rho - 1| \leq \frac{\frac{1}{2}L + 1}{\sigma_k}.$$

Now assume that

$$\sigma_k \geq \frac{\frac{1}{2}L + 1}{1 - \eta_2}.$$

We obtain from the two previous inequalities that

$$|\rho_k - 1| \leq 1 - \eta_2 \text{ and thus } \rho_k \geq \eta_2.$$

Then the iteration k is very successful in that $\rho_k \geq \eta_2$ and $\sigma_{k+1} \leq \sigma_k$. As a consequence, the mechanism of the algorithm ensures that Lemma 1 holds. □

We then recall the result presented in (Birgin et al., 2017, Lemma 2.4) that bounds the number of unsuccessful iterations as a function of the number of successful ones.

{lem:k-bounded}

Lemma 2. *For all* $k \geq 0$,

$$k \leq |S_k| \left(1 + \frac{|\log \gamma_1|}{\log \gamma_2}\right) + \frac{1}{\log \gamma_2} \log\left(\frac{\sigma_{\max}}{\sigma_0}\right). \tag{11}$$

where $S_k = \{0 \leq j \leq k \mid \rho_j \geq \eta_1\}$ *denotes the set of "successful" iterations between* 0 *and* k.

Proof. (See (Birgin et al., 2017, Lemma 2.4).) We also denote by U_k its complement in $\{1, ..., k\}$, which corresponds to the index set of "unsuccessful" iterations between 0 and k. The regularization parameter update (10) gives that, for each $k \geq 0$,

$$\gamma_1 \sigma_j \leq \max[\gamma_1 \sigma_j, \sigma_{\min}] \leq \sigma_{j+1}, \quad j \in S_k, \quad \text{and} \quad \gamma_2 \sigma_j \leq \sigma_{j+1}, \quad j \in U_k.$$

Thus we deduce inductively that

$$\sigma_0 \gamma_1^{|S_k|} \gamma_2^{|U_k|} \leq \sigma_k.$$

Therefore, using Lemma 1, we obtain

$$|S_k| \log \gamma_1 + |U_k| \log \gamma_2 \leq \log\left(\frac{\sigma_{\max}}{\sigma_0}\right),$$

which then implies that

$$|U_k| \leq -|S_k| \frac{\log \gamma_1}{\log \gamma_2} + \frac{1}{\log \gamma_2} \log\left(\frac{\sigma_{\max}}{\sigma_0}\right),$$

since $\gamma_2 > 1$. The desired result then follows from the equality $k = |S_k| + |U_k|$ and the inequality $\gamma_1 < 1$ given by (8). $\square$

Using all the above results, we are now in position to state our main evaluation complexity result.

Theorem 1. *Let Assumption 1, Assumption 2 and Assumption 3 be satisfied. Assume $\omega_g^k \leq 1/\sigma_k$ for all $k \geq 0$. Then, given $\epsilon > 0$, Algorithm 3.1 needs at most*

$$\left\lfloor \kappa_s \frac{f(x_0) - f_{\text{low}}}{\epsilon^2} \right\rfloor$$

successful iterations (each involving one evaluation of f and its approximate derivative) and at most

$$\left\lfloor \kappa_s \frac{f(x_0) - f_{\text{low}}}{\epsilon^2} \right\rfloor \left(1 + \frac{|\log \gamma_1|}{\log \gamma_2}\right) + \frac{1}{\log \gamma_2} \log\left(\frac{\sigma_{\max}}{\sigma_0}\right)$$

iterations in total to produce an iterate x_ϵ such that $\|\nabla f(x_\epsilon)\| \leq \epsilon$, where $\sigma_{\max}$ is given by Lemma 1 and where

$$\kappa_s = \frac{(1 + \sigma_{\max})^2}{\eta_1 \sigma_{\min}}.$$

Proof. At each successful iteration, we have

$$\begin{aligned} f(x_k) - f(x_k + s_k) &\geq \eta_1 (\bar{T}_k(0) - \bar{T}_k(s_k)) \\ &\geq \frac{\eta_1}{\sigma_k} \|g(x_k, \omega_g^k)\|^2 \\ &\geq \frac{\eta_1 \sigma_{\min}}{(1 + \sigma_{\max})^2} \epsilon^2 \end{aligned}$$

where we used (9) and the fact that before termination

$$\|g(x_k, \omega_g^k)\| \geq \frac{1}{1 + \omega_g^k} \epsilon \geq \frac{1}{1 + \frac{1}{\sigma_k}} \epsilon \geq \frac{\sigma_k}{1 + \sigma_k} \epsilon \geq \frac{\sigma_{\min}}{1 + \sigma_{\max}} \epsilon.$$

Thus we deduce that as long as termination does not occur,

$$f(x_0) - f(x_{k+1}) = \sum_{j \in S_k} [f(x_j) - f(x_j + s_j)] \geq \frac{|S_k|}{\kappa_s} \epsilon^2, \tag{12}$$

from which the desired bound on the number of successful iterations follows. Lemma 2 is then invoked to compute the upper bound on the total of iterations.

□

5 Implementation and Numerical Results

{sec:implementation}

The Algorithm 3.1 was tested using the collection of unconstrained optimization problems available in the package `OptimizationProblems.jl`. The objective function and its exact derivative were evaluated using the `NLPModels.jl` package. Yet, the aim of the algorithm is to compute an approximate solution of the optimization problem based on a partial knowledge of the gradient. To that end, we chose to add some noise to the derivative furnished by `NLPModels.jl` as follows.

At each iteration k, we choose u_k is a unit random vector and $\lambda_k > 0$ a scalar such that

$$\lambda_k = \frac{\omega_g^k}{1+\omega_g^k}\|\nabla f(x_k)\|. \tag{13}$$

{{eq:lambda}}

From which we compute the inexact gradient

$$g(x_k, \omega_g^k) = \nabla f(x_k) + \lambda_k u_k. \tag{14}$$

Since $(1+\omega_g^k)\lambda_k = \omega_g^k\|\nabla f(x_k)\|$, we have

$$\lambda_k = \omega_g^k(\|\nabla f(x_k)\| - \lambda_k) \le \omega_g^k\|\nabla f(x_k) + \lambda_k u_k\| = \omega_g^k\|g(x_k, \omega_g^k)\|.$$

Therefore choosing λ_k as in (13) insures that the relative error on the gradient does not exceed the imposed tolerance (2).

Bibliography

E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163(1):359–368, May 2017. ISSN 1436-4646. DOI: 10.1007/s10107-016-1065-8.

Table of Contents

1 Introduction 1
2 Background and Assumptions 1
2.1 Assumptions 1
2.2 Background on adaptive regularization 2
3 Complete Algorithm 3
4 Convergence and Complexity Analysis 3
5 Implementation and Numerical Results 6

Todo list

is it really 1/2 ? 2