Quadratic Regularization Methods with Finite-Difference Gradient Approximations

Geovani Nunes Grapiglia
UCLouvain, Belgium

One World Optimization Seminar
January 31, 2022

Outline

- Introduction and Motivation
- New Methods
- Preliminary Numerical Results

Problem Definition

We are interested in the unconstrained optimization problem

$$\text{Minimize } f(x), \quad x \in \mathbb{R}^n, \tag{1}$$

where

- $f : \mathbb{R}^n \to \mathbb{R}$ is nonconvex and has a lower bound $f_{low} \in \mathbb{R}$.
- $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is L-Lipschitz continuous.

Problem Definition

We are interested in the unconstrained optimization problem

$$\text{Minimize } f(x), \quad x \in \mathbb{R}^n, \tag{1}$$

where

- $f : \mathbb{R}^n \to \mathbb{R}$ is nonconvex and has a lower bound $f_{low} \in \mathbb{R}$.
- $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is L-Lipschitz continuous.

Derivative-Free Optimization: We want to (approximately) solve (1) relying only on evaluations of $f(\cdot)$.

Worst-Case Evaluation Complexity Bounds

Practical Goal: Given $\epsilon > 0$, generate $\bar{x}$ such that

$$\|\nabla f(\bar{x})\| \leq \epsilon. \qquad (2)$$

Worst-Case Evaluation Complexity Bounds

Practical Goal: Given $\epsilon > 0$, generate $\bar{x}$ such that

$$\|\nabla f(\bar{x})\| \leq \epsilon. \quad (2)$$

Question: In the worst-case, how many function evaluations an specific method performs to generate $\bar{x}$ satisfying (2)?

Worst-Case Evaluation Complexity Bounds

Deterministic DFO Methods: $\mathcal{O}\left(n^2\epsilon^{-2}\right)$ function evaluations

- Direct Search Methods: Vicente (2013); Konecny & Richtárik (2014).
- Derivative-free trust-region methods: Garmanjani, Júdice & Vicente (2016)

Worst-Case Evaluation Complexity Bounds

Deterministic DFO Methods: $\mathcal{O}\left(n^2\epsilon^{-2}\right)$ function evaluations

- Direct Search Methods: Vicente (2013); Konecny & Richtárik (2014).
- Derivative-free trust-region methods: Garmanjani, Júdice & Vicente (2016)

Randomized Methods: Random steps, directions, subspaces...

- Nesterov & Spokoiny (2011): $\mathcal{O}\left(n\epsilon^{-2}\right)$.
- Gratton, Royer, Vicente & Zhang (2015): $\mathcal{O}\left(mn\epsilon^{-2}\right)$.
- Bergou, Gorbunov & Richtárik (2020): $\mathcal{O}\left(n\epsilon^{-2}\right)$.
- Kimiaei & Neumaier (2021): $\mathcal{O}\left(mn\epsilon^{-2}\right)$.
- Cartis & Roberts (2021): $\mathcal{O}\left(r\epsilon^{-2}\right)$.

Worst-Case Evaluation Complexity Bounds

Deterministic DFO Methods: $\mathcal{O}\left(n^2\epsilon^{-2}\right)$ function evaluations

- Direct Search Methods: Vicente (2013); Konecny & Richtárik (2014).
- Derivative-free trust-region methods: Garmanjani, Júdice & Vicente (2016)

Randomized Methods: Random steps, directions, subspaces...

- Nesterov & Spokoiny (2011): $\mathcal{O}\left(n\epsilon^{-2}\right)$.
- Gratton, Royer, Vicente & Zhang (2015): $\mathcal{O}\left(mn\epsilon^{-2}\right)$.
- Bergou, Gorbunov & Richtárik (2020): $\mathcal{O}\left(n\epsilon^{-2}\right)$.
- Kimiaei & Neumaier (2021): $\mathcal{O}\left(mn\epsilon^{-2}\right)$.
- Cartis & Roberts (2021): $\mathcal{O}\left(r\epsilon^{-2}\right)$.

Remark: Complexity bounds to generate x_k such that

$$E\left[\|\nabla f(x_k)\|\right] \leq \epsilon \quad \text{or} \quad P\left(\|\nabla f(x_k)\| \leq \epsilon\right) \geq 1 - e^{-c\epsilon^{-2}}.$$

This Work

Question: Is it possible to design deterministic DFO methods with worst-case evaluation complexity of $\mathcal{O}\left(n\epsilon^{-2}\right)$?

This Work

Question: Is it possible to design deterministic DFO methods with worst-case evaluation complexity of $\mathcal{O}\left(n\epsilon^{-2}\right)$? Yes!

Outline

- Introduction and Motivation
- New Methods
- Preliminary Numerical Results

Generic First-Order Method with Line Search (Template)

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Compute $\nabla f(x_k)$.

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) \nabla f(x_k).$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.2.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Generic First-Order Method with Line Search (Template)

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Compute $\nabla f(x_k)$.

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) \nabla f(x_k).$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.2.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Generic First-Order Method with Line Search (Template)

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Compute $\nabla f(x_k)$.

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) \nabla f(x_k).$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.2.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Generic First-Order Method with Line Search (Template)

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Compute $\nabla f(x_k)$.

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) \nabla f(x_k).$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.2.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Generic First-Order Method with Line Search (Template)

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Compute $\nabla f(x_k)$.

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) \nabla f(x_k).$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.2.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^{+}$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^+$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^+$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^+$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_1 \in \mathbb{R}^n$ and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 Choose $h_i > 0$ and compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^+$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \tag{3}$$

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \tag{3}$$

Inspired by condition (3), we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{4}$$

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \tag{3}$$

Inspired by condition (3), we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{4}$$

By the Lipschitz continuity of $\nabla f(\cdot)$ wee know that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \tag{3}$$

Inspired by condition (3), we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{4}$$

By the Lipschitz continuity of $\nabla f(\cdot)$ wee know that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Thus, to obtain (4) it is enough to take

$$0 < h_i \leq \frac{2\kappa_g}{\sqrt{n}L} \|x_k - x_{k-1}\|.$$

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \tag{3}$$

Inspired by condition (3), we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{4}$$

By the Lipschitz continuity of $\nabla f(\cdot)$ wee know that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Thus, to obtain (4) it is enough to take

$$0 < h_i \leq \frac{2\kappa_g}{\sqrt{n}L} \|x_k - x_{k-1}\|.$$

Our choice:

$$h_i = \frac{2\kappa_g}{\sqrt{n}\,(2^i \sigma_k)} \|x_k - x_{k-1}\|.$$

Choice of h_i

Recently, Kohler and Lucchi (2017) and Wang *et al.* (2019) considered adaptations of the Cubic Regularization of the Newton's Method where $\nabla^2 f(x_k)$ is replaced by a matriz B_k such that

$$\|\nabla^2 f(x_k) - B_k\| \leq \kappa_B \|x_k - x_{k-1}\|. \quad (3)$$

Inspired by condition (3), we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \quad (4)$$

By the Lipschitz continuity of $\nabla f(\cdot)$ wee know that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Thus, to obtain (4) it is enough to take

$$0 < h_i \leq \frac{2\kappa_g}{\sqrt{n}L} \|x_k - x_{k-1}\|.$$

Our choice:

$$h_i = \frac{2\kappa_g}{\sqrt{n}\,(2^i \sigma_k)} \|x_k - x_{k-1}\|.$$

If $2^i \sigma_k \geq L$, then (4) holds.

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, $\kappa_g > 0$, and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If $x_{k,i}^+$ satisfies a certain Line search condition, set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Lemma: Let

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

with

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|.$$

Derivative-Free Method

Lemma: Let

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

with

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|.$$

If

$$2^i \sigma_k \geq 2\,(L + \kappa_g)$$

then

$$f(x_k) - f(x_{k,i}^{+}) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^{+} - x_k\|^2 - \frac{\kappa_g}{2} \|x_k - x_{k-1}\|^2.$$

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, $\kappa_g > 0$, and $\sigma_1 > 0$, set $k := 1$

Step 1 Set $i := 0$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \dots, n.$$

Step 1.2 Set

$$x^+_{k,i} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If

$$f(x_k) - f(x^+_{k,i}) \geq \frac{2^i \sigma_k}{4} \|x^+_{k,i} - x_k\|^2 - \frac{\kappa_g}{2} \|x_k - x_{k-1}\|^2,$$

set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x^+_{k,i}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, $\kappa_g > 0$, and $\sigma_1 > 0$, set $k := 1$

Step 1 Set $i := 0$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If

$$f(x_k) - f(x_{k,i}^{+}) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^{+} - x_k\|^2 - \frac{\kappa_g}{2} \|x_k - x_{k-1}\|^2,$$

set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, $\kappa_g > 0$, and $\sigma_1 > 0$, set $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If

$$f(x_k) - f(x_{k,i}^{+}) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^{+} - x_k\|^2 - \frac{\kappa_g}{2} \|x_k - x_{k-1}\|^2,$$

set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^{+}$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k} \sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, and $\sigma_1 > 0$, set $\kappa_g = \sigma_1/2$, and $k := 1$.

Step 1 Set $i := 0$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If

$$f(x_k) - f(x_{k,i}^+) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^+ - x_k\|^2 - \frac{\sigma_1}{4} \|x_k - x_{k-1}\|^2,$$

set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k}\sigma_k\right)$, $k := k + 1$, and go to Step 1.

Derivative-Free Method

Step 0 Given $x_0, x_1 \in \mathbb{R}^n$ $(x_0 \neq x_1)$, and $\sigma_1 > 0$, set $\kappa_g = \sigma_1/2$, and $k := 1$.

Step 1 Find the smallest integer $i \geq 0$ such that $2^i \sigma_k \geq 2\sigma_1$.

Step 1.1 For

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$$

compute $g_{k,i} \in \mathbb{R}^n$ with

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad i = 1, \ldots, n.$$

Step 1.2 Set

$$x_{k,i}^+ = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i}.$$

Step 1.3 If

$$f(x_k) - f(x_{k,i}^+) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^+ - x_k\|^2 - \frac{\sigma_1}{4} \|x_k - x_{k-1}\|^2,$$

set $i_k := i$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2 Set $x_{k+1} = x_{k,i}^+$, $\sigma_{k+1} = \frac{1}{2}\left(2^{i_k} \sigma_k\right)$, $k := k + 1$, and go to Step 1.

Complexity Analysis

Lemma: The sequence $\{\sigma_k\}$ satisfies

$$\sigma_1 \leq \sigma_k \leq 2\left(L + \frac{\sigma_1}{2}\right) \equiv \sigma_{\max},$$

for all $k \geq 1$.

Complexity Analysis

Lemma: The sequence $\{\sigma_k\}$ satisfies

$$\sigma_1 \leq \sigma_k \leq 2\left(L + \frac{\sigma_1}{2}\right) \equiv \sigma_{\max},$$

for all $k \geq 1$.

Moreover, the number FE_T of function evaluations performed up to the Tth iteration is bounded as follows

$$FE_T \leq 1 + (n+1)\left[2T + \log_2(\sigma_{\max}) - \log_2(\sigma_1)\right] = \mathcal{O}(nT).$$

Complexity Analysis

Theorem: Given $\epsilon > 0$, let T be the first iteration index such that

$$\|\nabla f(x_T)\| \leq \epsilon.$$

Then,

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2} = \mathcal{O}(\epsilon^{-2}).$$

Complexity Analysis

Theorem: Given $\epsilon > 0$, let T be the first iteration index such that

$$\|\nabla f(x_T)\| \leq \epsilon.$$

Then,

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2} = \mathcal{O}(\epsilon^{-2}).$$

Corollary: $FE_T \leq \mathcal{O}(nT)$

Complexity Analysis

Theorem: Given $\epsilon > 0$, let T be the first iteration index such that

$$\|\nabla f(x_T)\| \leq \epsilon.$$

Then,

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2} = \mathcal{O}(\epsilon^{-2}).$$

Corollary: $FE_T \leq \mathcal{O}(nT) \leq \mathcal{O}(n\epsilon^{-2})$

Complexity Analysis

Theorem: Given $\epsilon > 0$, let T be the first iteration index such that

$$\|\nabla f(x_T)\| \leq \epsilon.$$

Then,

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2} = \mathcal{O}(\epsilon^{-2}).$$

Corollary: $FE_T \leq \mathcal{O}(nT) \leq \mathcal{O}(n\epsilon^{-2})$, i.e., the propose method requires at most $\mathcal{O}\left(n\epsilon^{-2}\right)$ function evaluations to generate x_k such that $\|\nabla f(x_k)\| \leq \epsilon$.

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}(2^i \sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{(2^i \sigma_k)}.$$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{(2^i \sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|$ whenever $2^i \sigma_k \geq \sqrt{n}L$.

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{(2^i \sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|$ whenever $2^i \sigma_k \geq \sqrt{n}L$.

In this case, we would get $\sigma_k \leq \sigma_{\max} = \mathcal{O}(\sqrt{n})$,

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g\|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i\sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2}h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g\|x_k - x_{k-1}\|}{(2^i\sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g\|x_k - x_{k-1}\|$ whenever $2^i\sigma_k \geq \sqrt{n}L$.

In this case, we would get $\sigma_k \leq \sigma_{\max} = \mathcal{O}(\sqrt{n})$, which would imply an iteration complexity bound

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2}$$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g\|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i\sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2}h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g\|x_k - x_{k-1}\|}{(2^i\sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g\|x_k - x_{k-1}\|$ whenever $2^i\sigma_k \geq \sqrt{n}L$.

In this case, we would get $\sigma_k \leq \sigma_{\max} = \mathcal{O}(\sqrt{n})$, which would imply an iteration complexity bound

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2} = \mathcal{O}(n\epsilon^{-2}).$$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i \sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2} h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g \|x_k - x_{k-1}\|}{(2^i \sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|$ whenever $2^i \sigma_k \geq \sqrt{n}L$.

In this case, we would get $\sigma_k \leq \sigma_{\max} = \mathcal{O}(\sqrt{n})$, which would imply an iteration complexity bound

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2} = \mathcal{O}(n\epsilon^{-2}).$$

In this case we would have $FE_T \leq \mathcal{O}(nT)$

What is the main trick?

The trick is to use $\sqrt{n}$ in $h_i = \dfrac{2\kappa_g\|x_k - x_{k-1}\|}{\sqrt{n}\,(2^i\sigma_k)}$

Remember that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L}{2}h_i.$$

Suppose that

$$h_i = \frac{2\kappa_g\|x_k - x_{k-1}\|}{(2^i\sigma_k)}.$$

Then, $\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g\|x_k - x_{k-1}\|$ whenever $2^i\sigma_k \geq \sqrt{n}L$.

In this case, we would get $\sigma_k \leq \sigma_{\max} = \mathcal{O}(\sqrt{n})$, which would imply an iteration complexity bound

$$T \leq 3 + \left(\frac{5L}{4} + \sigma_{\max}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2} = \mathcal{O}(n\epsilon^{-2}).$$

In this case we would have $FE_T \leq \mathcal{O}(nT) \leq \mathcal{O}(n^2\epsilon^{-2})$.

Quadratic Regularization Method: $\mathcal{O}(n\epsilon^{-2})$

Instead of

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i},$$

we can compute $x_{k,i}^{+}$ as an approximate minimizer of the quadratic model

$$M_{x_k, 2^i \sigma_k}(y) = f(x_k) + \langle g_{k,i}, y - x_k \rangle + \frac{1}{2}\langle B_k(y - x_k), y - x_k \rangle + \frac{2^i \sigma_k}{2}\|y - x_k\|^2,$$

where $B_k \in \mathbb{R}^{n \times n}$ is symmetric and positive semidefinite.

Quadratic Regularization Method: $\mathcal{O}(n\epsilon^{-2})$

Instead of

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i},$$

we can compute $x_{k,i}^{+}$ as an approximate minimizer of the quadratic model

$$M_{x_k, 2^i \sigma_k}(y) = f(x_k) + \langle g_{k,i}, y - x_k \rangle + \frac{1}{2}\langle B_k(y - x_k), y - x_k \rangle + \frac{2^i \sigma_k}{2}\|y - x_k\|^2,$$

where $B_k \in \mathbb{R}^{n \times n}$ is symmetric and positive semidefinite.

Example: B_k may be computed using Quasi-Newton formulas.

Quadratic Regularization Method: $\mathcal{O}(n\epsilon^{-2})$

Instead of

$$x_{k,i}^{+} = x_k - \left(\frac{1}{2^i \sigma_k}\right) g_{k,i},$$

we can compute $x_{k,i}^{+}$ as an approximate minimizer of the quadratic model

$$M_{x_k, 2^i\sigma_k}(y) = f(x_k) + \langle g_{k,i}, y - x_k \rangle + \frac{1}{2}\langle B_k(y - x_k), y - x_k \rangle + \frac{2^i \sigma_k}{2}\|y - x_k\|^2,$$

where $B_k \in \mathbb{R}^{n \times n}$ is symmetric and positive semidefinite.

Example: B_k may be computed using Quasi-Newton formulas.

Additional Assumption: $\|B_k\| \leq M$ for all $k \geq 1$.

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n.$$

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \dots, n.$$

Remember that we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{5}$$

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n.$$

Remember that we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{5}$$

If we assume that $\nabla^2 f(\cdot)$ is L_2-Lipschitz continuous, we can show that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L_2}{6} h_i^2.$$

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n.$$

Remember that we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{5}$$

If we assume that $\nabla^2 f(\cdot)$ is L_2-Lipschitz continuous, we can show that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L_2}{6} h_i^2.$$

Thus, to obtain (5) it is enough to take

$$0 < h_i \leq \left[\frac{6\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}L_2}\right]^{\frac{1}{2}}.$$

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n.$$

Remember that we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{5}$$

If we assume that $\nabla^2 f(\cdot)$ is L_2-Lipschitz continuous, we can show that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L_2}{6} h_i^2.$$

Thus, to obtain (5) it is enough to take

$$0 < h_i \leq \left[\frac{6\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}L_2}\right]^{\frac{1}{2}}.$$

Our choice:

$$h_i = \left[\frac{6\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}(2^i \sigma_k)}\right]^{\frac{1}{2}}.$$

QRM with Central Finite-Differences: $\mathcal{O}(n\epsilon^{-2})$

We can also use central finite-difference gradients:

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n.$$

Remember that we want

$$\|\nabla f(x_k) - g_{k,i}\| \leq \kappa_g \|x_k - x_{k-1}\|. \tag{5}$$

If we assume that $\nabla^2 f(\cdot)$ is L_2-Lipschitz continuous, we can show that

$$\|\nabla f(x_k) - g_{k,i}\| \leq \frac{\sqrt{n}L_2}{6} h_i^2.$$

Thus, to obtain (5) it is enough to take

$$0 < h_i \leq \left[\frac{6\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}L_2}\right]^{\frac{1}{2}}.$$

Our choice:

$$h_i = \left[\frac{6\kappa_g \|x_k - x_{k-1}\|}{\sqrt{n}(2^i \sigma_k)}\right]^{\frac{1}{2}}.$$

If $2^i \sigma_k \geq L_2$, then (5) holds.

Outline

- Introduction and Motivation
- New Methods
- Preliminary Numerical Results

Preliminary Numerical Results

Numerical experiments were performed on the set of 15 nonlinear least-squares problems from the Moré-Garbow-Hillstrom collection in which the dimension n can be chosen.

Preliminary Numerical Results

Numerical experiments were performed on the set of 15 nonlinear least-squares problems from the Moré-Garbow-Hillstrom collection in which the dimension n can be chosen.

For each problem, two choices of starting points were considered, namely

$$x_0 = 5^s \bar{x},$$

with $s \in \{0, 1\}$, where $\bar{x}$ is the standard starting point given in the MGH collection.

Preliminary Numerical Results

Numerical experiments were performed on the set of 15 nonlinear least-squares problems from the Moré-Garbow-Hillstrom collection in which the dimension n can be chosen.

For each problem, two choices of starting points were considered, namely

$$x_0 = 5^s \bar{x},$$

with $s \in \{0, 1\}$, where $\bar{x}$ is the standard starting point given in the MGH collection.

Four dimensions were considered: $n = 8, 12, 16, 20$.

Total of 120 test problems.

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

It was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points (15 problems).

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

It was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points (15 problems).

To investigate the ability of FDGM to generate approximate stationary points, the code was endowed with the stopping criterion

$$\|\nabla f(x_k)\| \leq \epsilon. \tag{6}$$

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

It was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points (15 problems).

To investigate the ability of FDGM to generate approximate stationary points, the code was endowed with the stopping criterion

$$\|\nabla f(x_k)\| \leq \epsilon. \tag{6}$$

Notation:

- $T(\epsilon)$: number of iterations required by the solver to generate x_k for which (6) holds.

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

It was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points (15 problems).

To investigate the ability of FDGM to generate approximate stationary points, the code was endowed with the stopping criterion

$$\|\nabla f(x_k)\| \leq \epsilon. \tag{6}$$

Notation:

- $T(\epsilon)$: number of iterations required by the solver to generate x_k for which (6) holds.
- $FE(\epsilon)$: Corresponding number of function evaluations.

Preliminary Numerical Results: Experiment 1

In the first experiment, the following code was tested:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.

It was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points (15 problems).

To investigate the ability of FDGM to generate approximate stationary points, the code was endowed with the stopping criterion

$$\|\nabla f(x_k)\| \leq \epsilon. \tag{6}$$

Notation:

- $T(\epsilon)$: number of iterations required by the solver to generate x_k for which (6) holds.
- $FE(\epsilon)$: Corresponding number of function evaluations.
- $A(\epsilon)$: It is defined as

$$A(\epsilon) = \frac{FE(\epsilon)}{T(\epsilon)(n+1)}.$$

Complexity Analysis

Lemma: The number FE_T of function evaluations performed up to the Tth iteration is bounded as follows

$$FE_T \leq 1 + (n+1)\left[2T + \log_2(\sigma_{\max}) - \log_2(\sigma_1)\right].$$

Complexity Analysis

Lemma: The number FE_T of function evaluations performed up to the Tth iteration is bounded as follows

$$A_T = \frac{FE_T}{T(n+1)} \leq 2 + \frac{1 + (n+1)\left[\log_2(\sigma_{\max}) - \log_2(\sigma_1)\right]}{T(n+1)}$$

Complexity Analysis

Lemma: The number FE_T of function evaluations performed up to the Tth iteration is bounded as follows

$$A_T = \frac{FE_T}{T(n+1)} \leq 2 + \frac{1 + (n+1)\left[\log_2(\sigma_{\max}) - \log_2(\sigma_1)\right]}{T(n+1)}$$

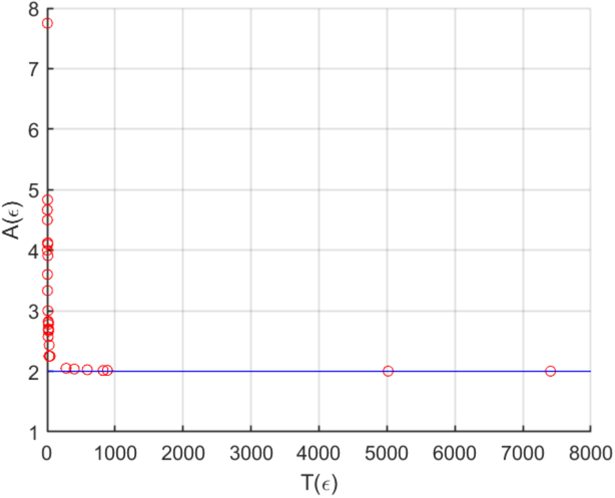
So, when $T \to +\infty$, the upper bound on A_T approaches to 2.

Preliminary Numerical Results: Experiment 1

	$\epsilon = 10^{-1}$			$\epsilon = 10^{-2}$		
PROBLEM	$T(\epsilon)$	$FE(\epsilon)$	$A(\epsilon)$	$T(\epsilon)$	$FE(\epsilon)$	$A(\epsilon)$
1. Extend. Rosenbrock	5017	90450	2.0032	7406	133452	2.0022
2. Extend. Powell Sing.	279	5148	2.0502	886	16074	2.0158
3. Penalty I	14	325	2.5714	14	324	2.5714
4. Penalty II	16	387	2.6875	44	891	2.2500
5. Variably Dim.	399	7317	2.0376	590	10755	2.0254
6. Trigonometric	4	162	4.5000	28	567	2.2500
7. Discrete BV	11	297	3.0000	824	14931	2.0133
8. Discrete IE	3	126	4.6667	5	162	3.6000
9. Broyden Tridiagonal	21	504	2.6667	30	657	2.4333
10. Broyden Banded	16	405	2.8125	20	486	2.7000
11. Brown AL	17	432	2.8235	18	450	2.7778
12. Linear	4	144	4.0000	6	180	3.3333
13. Linear-1	4	279	7.7500	4	279	7.7500
14. Linear-0	10	369	4.1000	11	387	3.9091
15. Chebyquad	6	261	4.8333	8	297	4.1250

Preliminary Numerical Results: Experiment 1

Figure below presents all the pairs $(T(\epsilon), A(\epsilon))$.



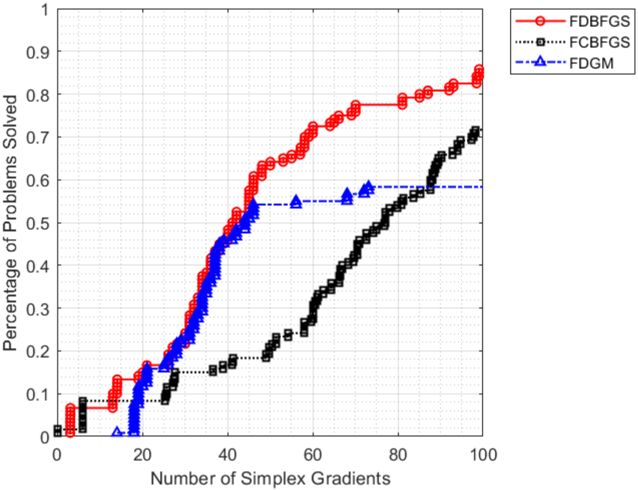
Preliminary Numerical Results: Experiment 2

The following codes were compared in the full set of 120 test problems:

- FDGM: New method with forward finite-difference gradients and $B_k = 0$.
- FDBFGS: New method with forward finite-difference gradients and B_k obtained by the BFGS formula.
- **FCBFGS**: New method with **central finite-difference** gradients and B_k obtained by the **BFGS formula**.

Preliminary Numerical Results

Data Profiles with $\tau = 10^{-7}$ (Moré & Wild, 2009)

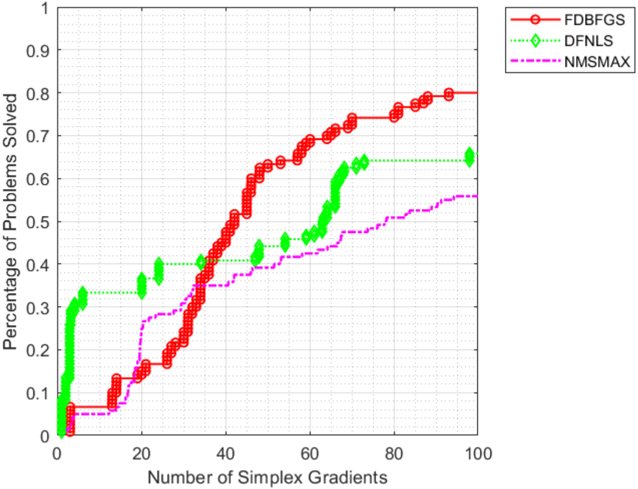


Preliminary Numerical Results

- FDBFGS: New method with forward finite-difference gradients and B_k obtained by the BFGS formula.
- DFNLS: derivative-free trust-region method proposed by G., Yuan & Yuan (2016).
- NMSMAX: Nelder-Mead Method.

Preliminary Numerical Results

Data Profiles with $\tau = 10^{-7}$ (Moré & Wild, 2009)



Conclusion

1. Deterministic Quadratic Regularization Derivative-Free Methods

Conclusion

1. Deterministic Quadratic Regularization Derivative-Free Methods

2. Worst-case evaluation complexity bounds of $\mathcal{O}\left(n\epsilon^{-2}\right)$.

Conclusion

1. Deterministic Quadratic Regularization Derivative-Free Methods
2. Worst-case evaluation complexity bounds of $\mathcal{O}\left(n\epsilon^{-2}\right)$.
3. Preliminary numerical seem promissing.

Topics for Future Research

1. Generalization to composite nonsmooth optimization problems ($f(x) = \psi(c(x))$, with ψ nonsmooth).

2. Adaptation to noisy problems: Berahas, Byrd & Nocedal (2019), Berahas, Cao, Choromanski & Scheinberg (2021), Berahas, Sohab & Vicente (2021), Shi, Xuan, Oztoprak & Nocedal (2021)...

Reference

G.N.G.: *Quadratic Regularization Methods with Finite-Difference Gradient Approximations*. Optimization Online (November, 2021)

Reference

G.N.G.: *Quadratic Regularization Methods with Finite-Difference Gradient Approximations*. Optimization Online (November, 2021)

Happy Lunar New Year!

geovani.grapiglia@uclouvain.be