Numerical frugality in optimization: approximated Newton's method

Giuseppe Carrino*, Nicolas Brisebarre+, Theo Mary†, Elisa Riccietti*

* ENS de Lyon, CNRS, Inria, UCBL, LIP, UMR 5668, 69342, Lyon cedex 07, France
† Sorbonne Université, CNRS, LIP6, 4 place Jussieu, 75005 Paris, France France
\+ CNRS, ENS de Lyon, Inria, UCBL, LIP, UMR 5668, 69342, Lyon cedex 07, France

TL; DR:
We need reliable and cheap optimizers,
we reduce cost of Newton's method harnessing inexactness
Application: Accurate model fitting with less expensive computation

Context: Newton's method

$$\min_{x\in\mathbb{R}^n} f(x) \xrightarrow[\text{iteratively solved by}]{} \text{Solve } H(x_i)d_i = -g(x_i), \quad x_{i+1} = x_i + d_i$$

*Usually, in **ML**:*

$$\min_\theta f_\theta(X), \text{ with } f_\theta(X) = \|F_\theta(X) - y\|^2$$

F : model function
$\{X, y\}$: dataset

H : Hessian matrix
g : gradient

Challenge: method's inexactness

Each operation within Newton's method can be **affected by inaccuracies**.

Willingly...

Inexact Newton $H(x_i)d_i \approx -g(x_i)$

Quasi - Newton $B(x_i) \approx H(x_i)$

...or not

Floating-point arithmetic $\mathrm{fl}(x+y) = (x+y)(1+\delta)$

Contribution: error analysis

We want to

- Develop a **convergence theory** for Newton with sources of inexactness
- Apply the theorem to **common** Newton's **approximations**
- Prove the analysis' soundness with **numerical experiments**

Error analysis framework

Adapting the work of [1] *to the optimization context*

Error model

$$\hat{x}_{i+1} = \hat{x}_i - (H(\hat{x}_i) + E_i^H)^{-1}(g(\hat{x}_i) + E_i^g) + E_i^+$$

$E_i^H \le \epsilon_i^H$, depending on H
$E_i^g \le \epsilon_i^g$ depending on g
$E_i^+ \le \epsilon_i$, depending on unit roundoff

Convergence theorem

Assuming...

$$\epsilon_i^H \kappa(H(\hat{x}_i)) \le 1/8$$

L_H : Lipschitz-constant of H

$$L_H\|H(x^*)^{-1}\|\|x_0 - x^*\| \le 1/8$$

...we can derive that...

$$\|x_{i+1} - x^*\| \le G_i\|x_i - x^*\| + lim\_acc_i$$

$$\|g(x_{i+1})\| \le P_i\|g(x_i)\| + lim\_g_i$$

Both terms G_i and P_i depend on ϵ_i^H

$$lim\_acc_i \approx \frac{\|H(x^*)^{-1}\|}{\|x^*\|}\epsilon_i^g + \epsilon_i$$

$$lim\_g_i \approx \epsilon_i^g + \epsilon_i\|H(\hat{x}_i)\|\|\hat{x}_i\|$$

Application: mixed-precision FP-Newton

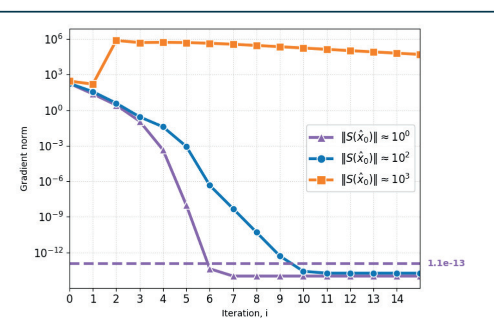
for $i = 0$ **to** $maxit$ **or until converged do**
  Compute $g_i = g(x_i)$ in precision π_g
  Solve $H(x_i)d_i = -g_i$ in precision π_ℓ
  Update $x_{i+1} = x_i + d_i$ in precision π_w
return x_i

Unequal inexactness → Mixed-precision!

Other Newton's approximations

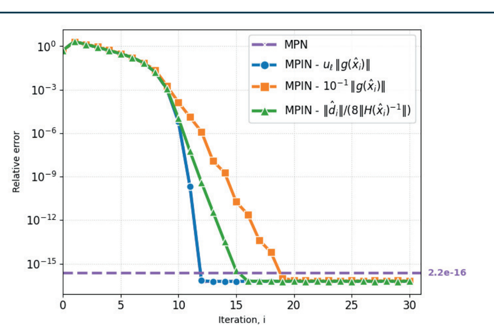
- **Quasi - Newton?**

(Potentially) Covered!
Example of Gauss-Newton



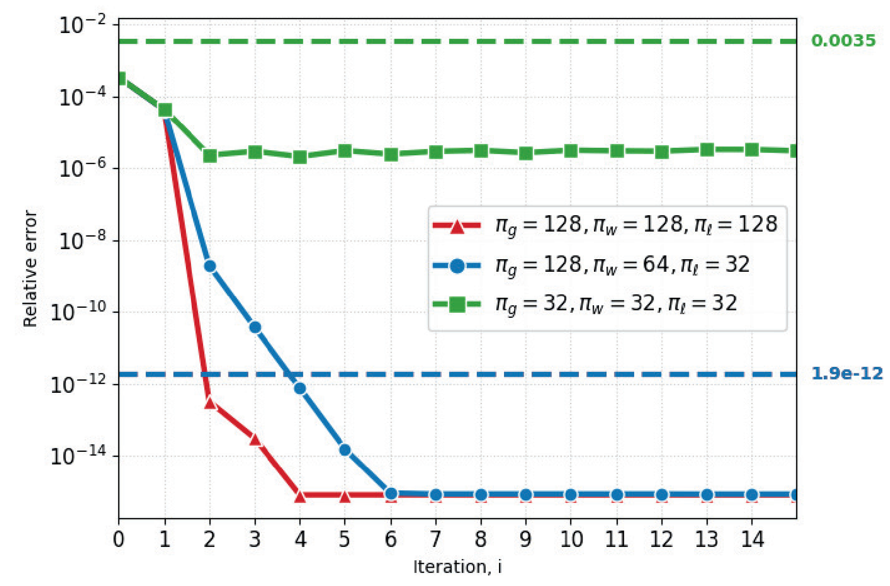
- **Inexact Newton?**

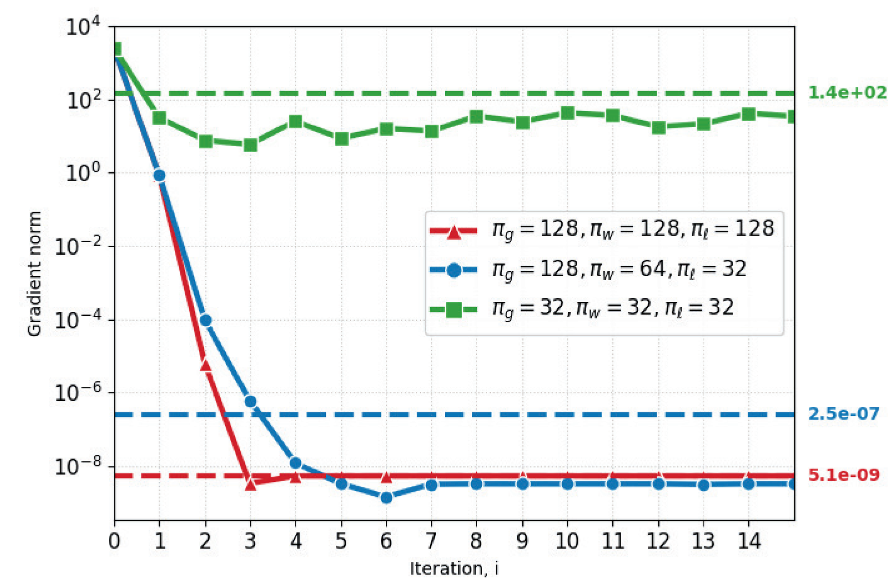
Covered!
and derived stopping conditions



Leveraging FP representation to reduce costs

Data fitting with Newton, High-precision, Mixed-precision, Low-precision





More sources of inexactness: finite-differences

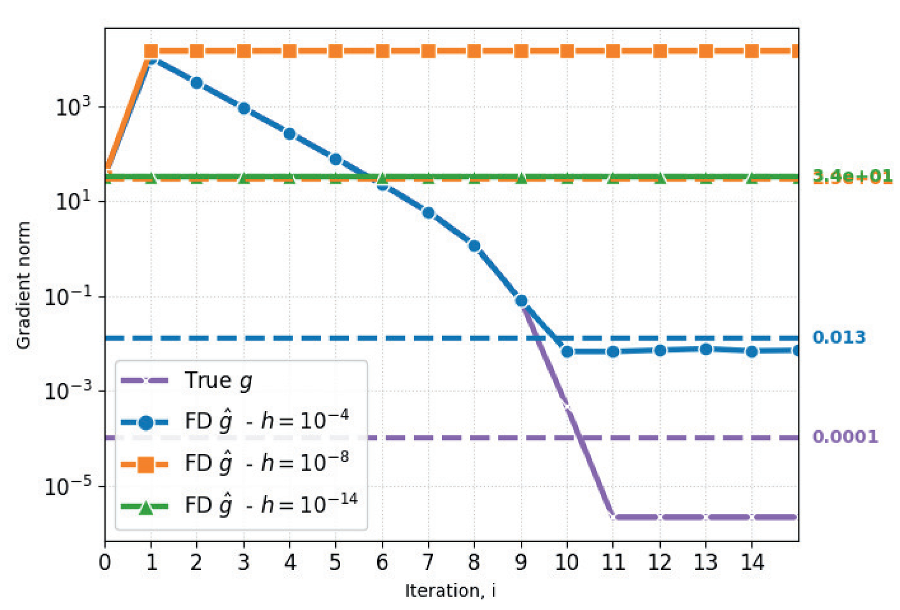
True g vs. finite-differences $\hat{g}$

$$g(x) = \nabla f(x)$$

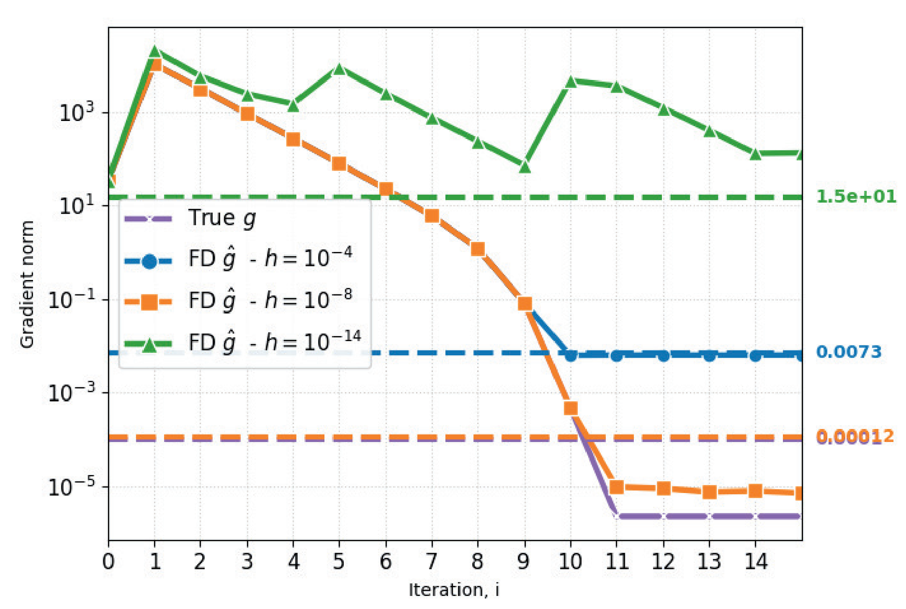
$$\hat{g}(x) = \left(\frac{f(x+he_i) - f(x)}{h}\right)_{i=1}^n$$

Highly **inexact** operation
↓
Improve its **precision**
↓
Better attainable **accuracy**

$\pi_g = \text{fp32}, \pi_w = \text{fp32}, \pi_\ell = \text{fp32}$



$\pi_g = \text{fp64}, \pi_w = \text{fp32}, \pi_\ell = \text{fp32}$



Key take-aways

- Newton's method can be **approximated** in **different ways**
- Operations are not **equally sensitive** to errors
- Our framework **adapts** to **different sources** of **inexactness**
- We can **save resources** sacrificing few accuracy following **theoretically-guided insights**

Question: ***larger-scale settings*?**

[1]: F. Tisseur. Newton's method in floating point arithmetic and iterative refinement of generalized eigenvalue problems. SIAM Journal on Matrix Analysis and Applications, 22(4):1038–1057, 2001.