



Numerical frugality in optimization: Newton's methods in mixed precision

SIAM OP26, Edinburgh

Problem statement

Optimization problem

Solving the following optimization problem:

$$\min_{x \in \mathbb{R}^n} f(x)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a smooth function.

A second-order optimizer

We want to solve this using **Newton's method**:

1. Compute $g_i = g(x_i)$

A second-order optimizer

We want to solve this using **Newton's method**:

1. Compute $g_i = g(x_i)$
2. Solve $H(x_i)d_i = -g_i$

A second-order optimizer

We want to solve this using **Newton's method**:

1. Compute $g_i = g(x_i)$
2. Solve $H(x_i)d_i = -g_i$
3. Update $x_{i+1} = x_i + d_i$

Newton's cost

Newton's method comes with two main sources of cost...

Solving the linear system

$$H(x_i) d_i = -g_i$$

Computing the Hessian

$$H(x_i)$$

Newton's cost

Newton's method comes with two main sources of cost...

Solving the linear system

$$H(x_i) d_i = -g_i$$



inexact Newton methods

$$\|H(x_i)d_i + g_i\| \leq \eta \|g_i\|$$

Computing the Hessian

$$H(x_i)$$



quasi-Newton methods

$$B(x) \approx H(x)$$

...but many possible approximations!

Quasi-Newton: an example

When solving least-squares problems,

Quasi-Newton: an example

When solving least-squares problems,

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|R(x)\|^2, \quad R : \mathbb{R}^n \rightarrow \mathbb{R}^m,$$

Quasi-Newton: an example

When solving least-squares problems,

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|R(x)\|^2, \quad R : \mathbb{R}^n \rightarrow \mathbb{R}^m,$$

the Hessian is given by

Quasi-Newton: an example

When solving least-squares problems,

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|R(x)\|^2, \quad R : \mathbb{R}^n \rightarrow \mathbb{R}^m,$$

the Hessian is given by

$$H(x) = J_R(x)^T J_R(x) + S(x), \quad S(x) = \sum_{i=0}^{m-1} R_i(x) \nabla^2 R_i(x)$$

Quasi-Newton: an example

When solving least-squares problems,

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|R(x)\|^2, \quad R: \mathbb{R}^n \rightarrow \mathbb{R}^m,$$

the Hessian is given by

$$H(x) = J_R(x)^T J_R(x) + S(x), \quad S(x) = \sum_{i=0}^{m-1} R_i(x) \nabla^2 R_i(x)$$


Gauss-Newton discards this!

Reducing the cost

- **Issue:** these variants can also be expensive, especially for large-scale problems.

Reducing the cost

- *Issue*: these variants can also be expensive, especially for large-scale problems.
- *Idea*: use **low precision floating-point arithmetic** to reduce the overall cost.

Floating-point precision

$$\mathbf{fl}(x \circ y) = (x \circ y)(1 + \delta), \quad |\delta| \leq u, \quad \circ \in \{+, -, *, /\}$$


Floating-point arithmetic model.

Floating-point precision

$$\text{fl}(x \circ y) = (x \circ y)(1 + \delta), \quad |\delta| \leq u, \quad \circ \in \{+, -, *, /\}$$

Floating-point arithmetic model.

Type	Size	Unit roundoff (u)	Approx. range
bfloat16	16 bits	4×10^{-3}	$10^{\pm 38}$
fp16	16 bits	5×10^{-4}	$10^{\pm 4}$
fp32	32 bits	6×10^{-8}	$10^{\pm 38}$
fp64	64 bits	1×10^{-16}	$10^{\pm 308}$



Floating point formats.

Reducing the cost

- **Issue**: also these variants can be expensive, especially for large-scale problems.
- **Idea**: use **low floating-point precision** to reduce the overall cost.
- **Challenge**: how to do it without sacrificing convergence?

Reducing the cost

- **Issue**: also these variants can be expensive, especially for large-scale problems.
- **Idea**: use **low floating-point precision** to reduce the overall cost.
- **Challenge**: how to do it without sacrificing convergence?

We can leverage **mixed precision**!

Mixed precision Newton's method

Error model

For each iteration i , the mixed precision Newton's step satisfies

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

Error model

For each iteration i , the mixed precision Newton's step satisfies

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + \boxed{E_i^H}\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

- E_i^H : Hessian approximations and/or inexact linear solvers.

Error model

For each iteration i , the mixed precision Newton's step satisfies

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + \boxed{E_i^H}\right)^{-1} \left(g(\hat{x}_i) + \boxed{e_i^g}\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

- E_i^H : Hessian approximations and/or inexact linear solvers.
- e_i^g : inexact gradient computations.

Error model

For each iteration i , the mixed precision Newton's step satisfies

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

- E_i^H : Hessian approximations and/or inexact linear solvers.
- e_i^g : inexact gradient computations.
- e_i^+ : errors in the update step.

Local convergence analysis

$$\|\hat{\mathbf{x}}_{i+1} - \mathbf{x}^*\| \leq \alpha_i \|\hat{\mathbf{x}}_i - \mathbf{x}^*\|^2 + \beta_i \|\hat{\mathbf{x}}_i - \mathbf{x}^*\| + \gamma_i,$$

Mixed precision Newton's method convergence rate.

Local convergence analysis

$$\|\hat{x}_{i+1} - x^*\| \leq \boxed{\alpha_i} \|\hat{x}_i - x^*\|^2 + \beta_i \|\hat{x}_i - x^*\| + \gamma_i,$$

Mixed precision Newton's method convergence rate.

- α_i : standard Newton's quadratic convergence;

Local convergence analysis

$$\|\hat{x}_{i+1} - x^*\| \leq \alpha_i \|\hat{x}_i - x^*\|^2 + \beta_i \|\hat{x}_i - x^*\| + \gamma_i,$$

Mixed precision Newton's method convergence rate.

- α_i : standard Newton's quadratic convergence;
- β_i : linear convergence influenced by Hessian errors;

Local convergence analysis

$$\|\hat{x}_{i+1} - x^*\| \leq \alpha_i \|\hat{x}_i - x^*\|^2 + \beta_i \|\hat{x}_i - x^*\| + \gamma_i$$

Mixed precision Newton's method convergence rate.

- α_i : standard Newton's quadratic convergence;
- β_i : linear convergence influenced by Hessian errors;
- γ_i : **limiting accuracy** mainly related to gradient errors.

Error model applications

Floating-point arithmetic Newton's method

We take into account a floating-point precision for each operation.

Floating-point arithmetic Newton's method

We take into account a floating-point precision for each operation.

1. Compute $g_i = g(x_i)$ in precision u_g

Floating-point arithmetic Newton's method

We take into account a floating-point precision for each operation.

1. Compute $g_i = g(x_i)$ in precision u_g
2. Solve $H(x_i)d_i = -g_i$ in precision u_H

Floating-point arithmetic Newton's method

We take into account a floating-point precision for each operation.

1. Compute $g_i = g(x_i)$ in precision u_g
2. Solve $H(x_i)d_i = -g_i$ in precision u_H
3. Update $x_{i+1} = x_i + d_i$ in precision u

Finite precision and error model

$$\hat{d}_i := -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right),$$
$$\hat{x}_{i+1} = \hat{x}_i + \hat{d}_i + e_i^+.$$

Mixed precision Newton's error model.

Finite precision and error model

$$\hat{d}_i := -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right),$$
$$\hat{x}_{i+1} = \hat{x}_i + \hat{d}_i + e_i^+$$

Mixed precision Newton's error model.

- $\|e_i^+\| \lesssim u$, by standard floating-point arithmetic model;

Finite precision and error model

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

Mixed precision Newton's error model.

- $\|e_i^+\| \lesssim u$, by standard floating-point arithmetic model;
- $\|E_i^H\| \approx O(u_H)$, if linear system is solved with a backward stable method;

Finite precision and error model

$$\begin{aligned}\hat{d}_i &:= -\left(H(\hat{x}_i) + E_i^H\right)^{-1} \left(g(\hat{x}_i) + e_i^g\right), \\ \hat{x}_{i+1} &= \hat{x}_i + \hat{d}_i + e_i^+.\end{aligned}$$

Mixed precision Newton's error model.

- $\|e_i^+\| \lesssim u$, by standard floating-point arithmetic model;
- $\|E_i^H\| \approx O(u_H)$, if linear system is solved with a backward stable method;
- $\|e_i^g\| \approx O(u_g)$.

Mixed precision setting

The setting of interest is

$$u_g \leq u \leq u_H$$

Mixed precision setting

The setting of interest is

$$u_g \leq \boxed{u} \leq u_H$$

- **Target precision:** the accuracy we aim to achieve.

Mixed precision setting

The setting of interest is

$$u_g \leq u \leq u_H$$

- **Target precision:** the accuracy we aim to achieve.
- **Accurate gradient** (u_g small): better limiting accuracy γ_i .

Mixed precision setting

The setting of interest is

$$u_g \leq u \leq u_H$$

- **Target precision**: the accuracy we aim to achieve.
- **Accurate gradient** (u_g small): better limiting accuracy γ_i .
- **Reduced precision linear solver** (u_H large): reduced cost per iteration.

Mixed precision setting

The setting of interest is

$$u_g \leq u \leq u_H$$

- **Target precision:** the accuracy we aim to achieve.
- **Accurate gradient** (u_g small): better limiting accuracy γ_i .
- **Reduced precision linear solver** (u_H large): reduced cost per iteration.
- **Trade-off:** potentially slower convergence.

Mixed precision setting

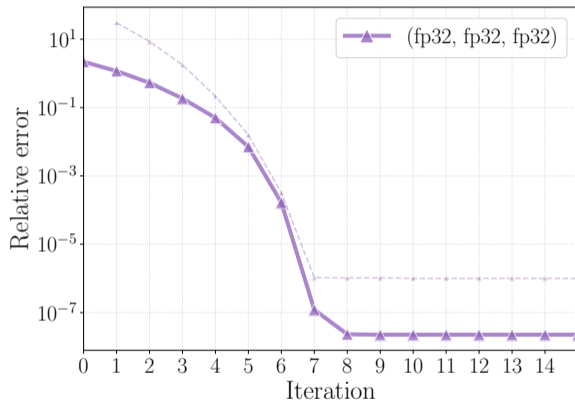
The setting of interest is

$$u_g \leq u \leq u_H$$

- **Target precision:** the accuracy we aim to achieve.
- **Accurate gradient** (u_g small): better limiting accuracy γ_i .
- **Reduced precision linear solver** (u_H large): reduced cost per iteration.
- **Trade-off:** potentially slower convergence.

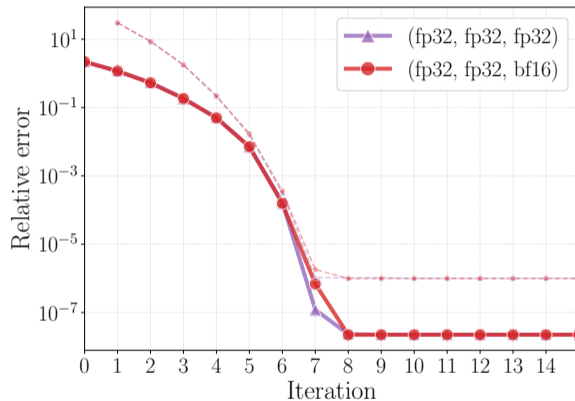
We will denote precision sets as (u_g, u, u_H)

Charts - Mixed precision Newton



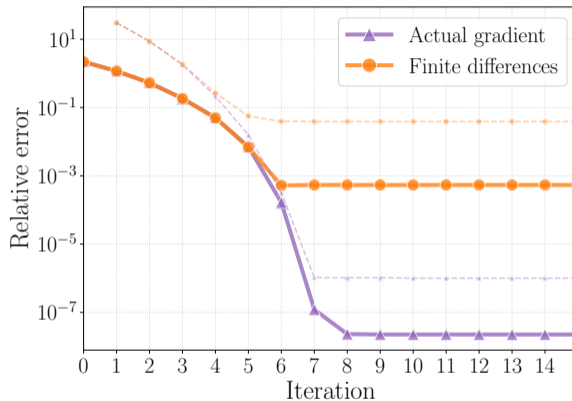
ENGVAL1. Mixed precision Newton relative error and its predicted rate.

Charts - Two mixed precision settings



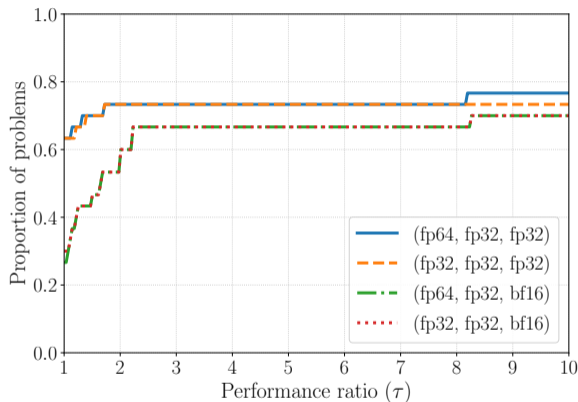
ENGVAl1. Different mixed precision settings relative error.

Charts - Finite differences



ENGVAL1. Finite differences vs true gradient relative error.

Charts - CUTEst performance profile



CUTEst problems collection. Performance profile of different mixed precision settings.

Error model extensions

Inexact Newton

$$\|H(x_i)d_i + g_i\| \leq \eta \|g_i\|$$

linear solver stopped early at tolerance η

Error model extensions

Inexact Newton

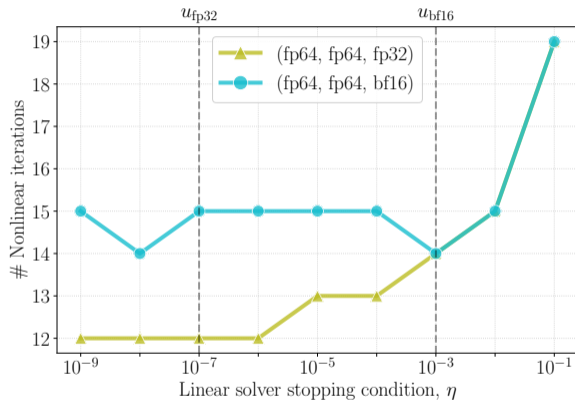
$$\|H(x_i)d_i + g_i\| \leq \eta \|g_i\|$$

linear solver stopped early at tolerance η



$$\|E_i^H\| \lesssim c_{\text{solver}} u_H + \frac{\|g(\hat{x}_i)\|}{\|H(\hat{x}_i)\| \|\hat{d}_i\|} \eta$$

Charts - Inexact Newton



ENGVAl1. Inexact Newton: iterations vs. η .

Error model extensions

Gauss-Newton

$$H(x) = J_R(x)^T J_R(x) \quad \text{discarding } S(x)$$

Hessian approximation discarding $S(\hat{x}_i)$

Error model extensions

Gauss-Newton

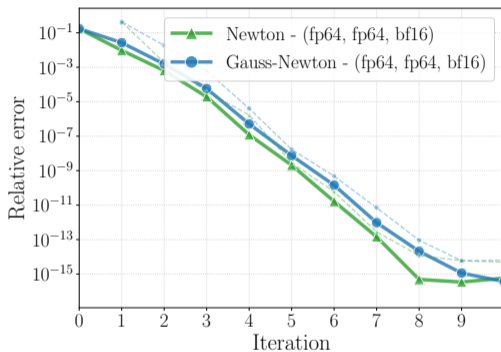
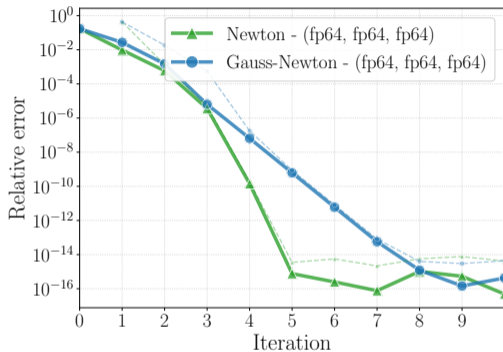
$$H(x) = J_R(x)^T J_R(x) \Rightarrow \cancel{S(x)}$$

Hessian approximation discarding $S(\hat{x}_i)$



$$\|E_i^H\| \leq c_{\text{solver}} u_H + \frac{\|S(\hat{x}_i)\|}{\|H(\hat{x}_i)\|}$$

Charts - Gauss-Newton



SINREG. Gauss-Newton vs. Newton's method.

Conclusions

Our contribution

- A comprehensive **error model** for mixed precision Newton's methods.

Our contribution

- A comprehensive **error model** for mixed precision Newton's methods.
- A rigorous **convergence analysis**, highlighting the interplay between *precision* and *convergence rates*.

Our contribution

- A comprehensive **error model** for mixed precision Newton's methods.
- A rigorous **convergence analysis**, highlighting the interplay between *precision* and *convergence rates*.
- **Numerical experiments** validated theoretical findings, demonstrating the practical benefits of mixed precision strategies.

Thank you!

Questions?

Gradient norm decrease

$$\|g(\hat{x}_{i+1})\| \leq \phi_i \|g(\hat{x}_i)\| + \psi_i.$$

Mixed precision Newton's method gradient norm decay.

- ϕ_i : "quadratic"/linear convergence influenced by Hessian errors;
- $\psi_i \approx \epsilon_i^g + \epsilon_i \|H(\hat{x}_i)\| \|\hat{x}_i\| \rightarrow$ **Potential stopping criterion!**

Convergence theorem assumptions

$$\|e_i^H\| \leq \epsilon_i^H \|H(\hat{x}_i)\|, \quad \|e_i^g\| \leq \epsilon_i \|g(\hat{x}_i)\| + \epsilon_i^g, \quad \|e_i^+\| \leq \epsilon_i (\|\hat{d}_i\| + \|\hat{x}_i\|).$$

At a given iteration

- $\epsilon_i^H \kappa(H(\hat{x}_i)) < 1$
- $L_H \|H(x^*)^{-1}\| \|\hat{x}_i - x^*\| < 1$

General local convergence

- $\epsilon_i^H \kappa(H(\hat{x}_i)) < 1 \quad \forall i$
- $L_H \|H(x^*)^{-1}\| \|\hat{x}_0 - x^*\| < 1$
- $\theta_i := \alpha_i \|\hat{x}_i - x^*\| + \beta_i < 1 \quad \forall i$
- until $\|\hat{x}_i - x^*\| \leq \frac{\gamma_i}{1-\theta_i}$.

Numerical experiments problems

$$f(x) = 3 + \sum_{k=0}^{n-2} (x_k^2 + x_{k+1}^2)^2 - 4x_k.$$

ENGVAL1 function (from CUTEst dataset).

$$R(x) = x_0 z + \sum_{k=1}^{\lceil \frac{n-1}{2} \rceil} x_{2k-1} z^{k+1} + \sum_{k=1}^{\lfloor \frac{n-1}{2} \rfloor} x_{2k} \sin(x_{2k} z),$$

SINREG least squares residual, with z fixed data.