Tight Analysis of Second-Order Optimization Methods via Interpolation of generalized Hessian Lipschitz univariate functions

EUROPT 2025

François Glineur

joint work with **Anne Rubbens**, **Nizar Bousselmi**, and **Julien Hendrickx**



icteam UCLouvain fnrs LA LIBERTÉ DE CHERCHER

Performance estimation of second-order optimization methods on classes of univariate functions

EUROPT 2025

François Glineur

joint work with **Anne Rubbens**, **Nizar Bousselmi**, and **Julien Hendrickx**

icteam UCLouvain fnrs LA LIBERTÉ DE CHERCHER

How to analyze worst-case of optimization methods?

- Optimization method $\mathcal{M}$ (e.g. gradient method, Newton's method,…)
- Function class $\mathcal{F}$ (e.g. convex, smooth, self-concordant,…)
- Problem : $\min_x f(x)$

Question : Worst-case performance of $\mathcal{M}$ on instance of $\mathcal{F}$?

Example: Worst-case performance of Gradient Method on L-smooth convex functions after N iterations?

$$f(x_N) - f^* \leq \frac{L}{2} \frac{||x_0 - x^*||^2}{2N+1}$$

How to construct a convergence rate proof?

Definition of function class $\mathcal{F}$ → **Step 1: Inequalities** from definition of class $\mathcal{F}$ → **Step 2: Optimal combination** of inequalities via **PEP** → Guarantee on the performance of $\mathcal{M}$

$\mathcal{F}$ = {smooth convex functions} → **Step 1: Inequalities**

$$f(y) \leq f(x) + \nabla f(x)^T(y-x) + \frac{L}{2}\|y-x\|^2, \; \forall x, y.$$

$$f(x) \geq f(y) + \langle \nabla f(y), x-y \rangle$$

→ **Step 2: combination** of inequalities and $x_{k+1} = x_k - \nabla f(x_k)$ → $f(x_N) - f^* \leq \frac{L}{2} \frac{||x_0 - x^*||^2}{2N+1}$

Two sources of (possible) **conservatism** on the guarantee:

- **combination** of inequalities may not be optimal *(PEP problem not solved optimally)*
- **inequalities** may not be sufficient conditions interpolation in the function class (only necessary)

Optimal combination of **exact** inequalities leads to exact/tight worst-case analysis

Take home messages

Performance estimation (PEP) provides worst-case convergence **rates**, **proofs** for those rates and corresponding explicit worst-case **functions**, and crucially rely on exact **interpolation inequalities**

In this work, we show how to derive exact **interpolation inequalities** for **higher order classes of functions**, such as Hessian Lipschitz functions and (generalized) self-concordant functions, in the **univariate** case

Using **PEP** with those **inequalities**, we study and find tight rates for many variants of second-order methods, such **Newton's** method and **cubic**/**adaptive** variants

Outline

1. **Performance Estimation Problem (PEP) Framework**
2. **Principled technique to characterize univariate class of functions**
3. **Interpolation conditions for second-order univariate function class**
4. **Convergence results for second-order methods**

Conceptual PEP: maximizing the worst-case performance

Idea: Finding the worst-case performance as an optimization problem

$$\max_{x_0, x^*, f} \quad \text{Perf}(x_N, f)$$

$$f \in \mathcal{F}$$

$$x_N = \mathcal{M}(x_0, f)$$

$$||\nabla f(x^*)||^2 = 0$$

$$||x_0 - x^*||^2 \leq 1$$

Maximize Perf of $\mathcal{M}$ among the set of functions $f \in F$

Perf(x_N, f) can be : $||x_N - x^*||$, $||\nabla f(x_N)||$, $f_N - f^*$, etc

Issue: untractable since optimization in function space

Solution: discretize function f and its gradient at iterates (equivalent for a black-box optimization method)

From conceptual PEP to tractable PEP (1)

Example: Worst-case performance of gradient method on L-smooth convex functions

$$\max_{\text{points } x_i, x^*, \text{function } f} \quad f(x_N) - f(x^*)$$

$$\text{s.t.} \quad f \; L\text{-smooth convex},$$

$$x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i),$$

$$||x^* - x_0||^2 \leq 1,$$

$$||\nabla f(x^*)||^2 = 0.$$

$\Longleftrightarrow$

$$\max_{\text{points } x_i, x^*, f_i, f^*, g_i, g^*} \quad f_N \quad - f^*$$

$$\text{s.t.} \quad \exists f \; L\text{-smooth convex}: \quad f(x_i) = f_i, \quad \nabla f(x_i) = g_i,$$

$$f(x^*) = f^*, \quad \nabla f(x^*) = g^*,$$

$$x_{i+1} = x_i - \frac{1}{L} g_i,$$

$$||x^* - x_0||^2 \leq 1,$$

$$||g^*||^2 \quad = 0.$$

Key concept: necessary and sufficient **interpolation conditions**

Interpolation conditions

Step 1: Inequalities from definition of $\mathcal{F}$

Theorem 1: f is L-smooth convex if and only if for all $x, y \in R^n$

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L}{2}\|y - x\|^2, \ \forall x, y.$$

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle$$

Theorem 2: f is L-smooth convex if and only if for all $x, y \in R^n$

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{1}{2L}\|\nabla f(y) - \nabla f(x)\|^2, \ \forall x, y.$$

Proof/PEP does **not use all $x, y \in R^n$, only $x_0, \ldots, x_N, x^*$**

Given $\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$,

$\exists$ L-smooth convex f such that $\begin{cases} f(x_i) & = f_i, \quad \forall i \\ \nabla f(x_i) & = g_i, \quad \forall i \end{cases}$ if, and only if,

$$f_i \geq f_k + g_k^T (x_i - x_k) + \frac{1}{2L}||g_i - g_k||^2 \quad \forall (i, k).$$

From conceptual PEP to tractable PEP (2)

Example: Worst-case performance of gradient method on L-smooth convex functions

$$\max_{\text{points } x_i, x^*, f_i, f^*, g_i, g^*} \quad f_N - f^*$$

$$\text{s.t.} \quad \exists f \ L\text{-smooth convex}: \quad f(x_i) = f_i, \ \nabla f(x_i) = g_i, \quad f(x^*) = f^*, \ \nabla f(x^*) = g^*,$$

$$x_{i+1} = x_i - \frac{1}{L} g_i, \quad ||x^* - x_0||^2 \leq 1, \quad ||g^*||^2 = 0.$$

$\Longleftrightarrow$

$$\max_{\text{points } x_i, x^*, f_i, f^*, g_i, g^*} \quad f_N - f^*$$

$$\text{s.t.} \quad f_i \geq f_k + g_k^T(x_i - x_k) + \frac{1}{2L}||g_i - g_k||^2,$$

$$x_{i+1} = x_i - \frac{1}{L} g_i, \quad ||x^* - x_0||^2 \leq 1, \quad ||g^*||^2 = 0.$$

- Non-convex Quadratically Constrained Quadratic Problem (QCQP)
- Linear on f_i and $x_i^T g_i, \ x_i^T x_j, \ g_i^T g_j$
- Can be sometimes be formulated as convex semidefinite program, hence efficiently solvable
- PEP gives the exact worst-case numerically (which helps to prove it analytically) [Drori, Teboulle 14] [Taylor, Hendrickx, G 17]
- It gives all the answers, but we should ask the relevant questions

Convex formulation of PEP

Convex formulation of PEP when:

Only First-Order methods

- Method analyzed is linear combination of (previous or future) gradients g_i and iterates x_i.
- Interpolation conditions are convex in f_i and $x_i^T g_i, \ \ x_i^T x_j, \ \ g_i^T g_j$

1. Gradient method : $x_{i+1} = x_i - \frac{h}{L}\nabla f(x_i)$

2. Fast gradient method :

$$y_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$$
$$\theta_{i+1} = \frac{1+\sqrt{4\theta_i^2+1}}{2}$$
$$x_{i+1} = y_{i+1} + \frac{\theta_i - 1}{\theta_{i+1}}(y_{i+1} - y_i)$$

OK

See more examples in « PEPit's documentation »

Key idea: lift products of variables (Gram)
$\mapsto$ efficiently solvable semidefinite optimization problem

3. Proximal method: $x_{i+1} = prox_{f(.)}(x_i) = \ x_i - \nabla f(x_{i+1})$

4. Chambolle-Pock method:

$$\begin{cases} x_{i+1} & = \mathrm{prox}_{\tau f}\left(x_i - \tau M^T u_i\right), \\ u_{i+1} & = \mathrm{prox}_{\sigma g^*}\left(u_i + \sigma M(2x_{i+1} - x_i)\right), \end{cases}$$

[Drori, Teboulle 14]
[Taylor, Hendrickx, G 17a]
[Taylor, Hendrickx, G 17b]
[Bousselmi, Hendrickx, G 23]

This work: PEP to analyze second-order methods (1)

Example: Analysis of **Newton's method**

$$\max_{x_k\in\mathbb{R}^d, g_k\in\mathbb{R}^d, h_k\in\mathbb{R}^{d\times d}, p_k\in\mathbb{R}^d} ||x_N - x^*||^2$$

$$\text{s.t.} \quad \exists f \in \mathcal{F} \text{ s.t. } f(x_k) = f_k, \nabla f(x_k) = g_k, \nabla^2 f(x_k) = h_k,$$

$$\text{(Newton step)} \quad x_{k+1} = x_k - p_k, \quad h_k p_k = g_k,$$

$$||x_0 - x^*||^2 \leq R^2,$$

$$||g^*||^2 = 0,$$

Or any other second order scheme:

[Nesterov, Polyak 2008]

- Cubic Newton method : $T_M(x) \in \operatorname{Arg\,min}_y \left[\langle f'(x), y-x\rangle + \frac{1}{2}\langle f''(x)(y-x), y-x\rangle + \frac{M}{6}\|y-x\|^3 \right], \quad (2.4)$
- Damped Newton method: $x_{k+1} = x_k - \frac{1}{1+M_f\lambda_f(x_k)}[\nabla^2 f(x_k)]^{-1}\nabla f(x_k)$
- Gradient Regularized Newton method: [Mishchenko 2022] $\lambda_k = \sqrt{H\|\nabla f(x^k)\|}$, $x^{k+1} = x^k - (\nabla^2 f(x^k) + \lambda_k \mathbf{I})^{-1}\nabla f(x^k)$

This work: PEP to analyze second-order methods (2)

Two main issues to extend PEP to second-order methods:

1) Interpolation conditions for second-order function class (e.g. Hessian Lipschitz, self-concordant functions)

Solution: Principled technique to provide interpolation conditions for **univariate generalized self-concordant functions**

2) PEP formulation is **non-convex**

Solution: Solve the non-convex problem with **global non-convex solver** (e.g. Gurobi)

- Lose nice properties of convex (lifted) PEP but allows more flexibility (e.g. non-convex/integer constraints, use of exponential functions)
- Non-convex PEPs have been solved previously in other settings (e.g. [Ryu, Taylor, Bergeling, Giselson 2020] and [Das Gupta, Van Parys, Ryu 2023])
- [de Klerk, G, Taylor 2020] used convex PEP to analyze a single iteration of a Newton step for self-concordant functions (with a trick to deal with Hessian norm)

Outline

1. **Performance Estimation Problem (PEP) Framework**
2. **Principled technique to characterize univariate class of functions**
3. **Interpolation conditions for second-order univariate function class**
4. **Convergence results for second-order methods**

Univariate generalized self-concordant functions

Focus on univariate functions (easier and still interesting)
(univariate) generalized self-concordant functions [Sun, Tran-Dinh 2019
includes Hessian Lipschitz, self-concordant, quasi-self-concordant, etc

$$|f'''(x)| \leq A f''(x)^{\alpha}$$
$$f''(x) \geq 0$$

Definition 1 (« 1-point def » univariate generalized self-concordant functions)

$$f \in \mathcal{F}_{M,\alpha} \quad \Leftrightarrow \quad \begin{cases} |f'(x)| \leq |\beta(\alpha)| M f(x)^{\alpha}, \\ f(x) \geq 0, \end{cases} \quad \forall x. \qquad \beta(\alpha) = \begin{cases} \frac{1}{1-\alpha} & \text{if } \alpha \neq 1 \\ 1 & \text{if } \alpha = 1. \end{cases}$$

Theorem 1 (« 2-points def » generalized Lipschitz functions)

$$f \in \mathcal{F}_{M,\alpha} \quad \Leftrightarrow \quad \begin{cases} |\tilde{f}(x) - \tilde{f}(y)| \leq M|x-y|, & \forall x, y, \\ f(x) \geq 0 \quad \forall x \end{cases} \quad \textit{where } \tilde{f}(x) = \begin{cases} f(x)^{1-\alpha}, & \textit{if } \alpha \neq 1, \\ \log(f(x)), & \textit{if } \alpha = 1. \end{cases}$$

Theorem 2 (Interpolation conditions <u>without gradient and function values</u>)

$S = \{(x_i, h_i)\}_{i \in [N]}$ *is* $\mathcal{F}_{M,\alpha}$*-interpolable if and only if*

$$\begin{cases} |\tilde{f}_i - \tilde{f}_j| \leq M|x_i - x_j|, & \forall i, j, \\ f_i \geq 0, & \forall i. \end{cases} \qquad \textit{where } \tilde{f}_i = \begin{cases} f_i^{1-\alpha}, & \textit{if } \alpha \neq 1, \\ \log(f_i), & \textit{if } \alpha = 1. \end{cases}$$

Principled technique to obtain interpolation conditions (1)

Summary: Given class $\mathcal{F}$ for which we have interpolation conditions (e.g., Lipschitz functions), gives interpolation conditions of the class whose derivative belongs to $\mathcal{F}$ (e.g., smooth functions) **called $\int \boldsymbol{F}$**.

(main reason for univariate restriction)

Theorem 3 (Interpolation conditions for Lipschitz functions)

$S = \{(x_i, f_i)\}_{i \in [N]}$ *is interpolable by a Lipschitz function if and only if*

$$|f_i - f_j| \leq M|x_i - x_j|, \quad \forall i, j$$

Principled technique ⬇

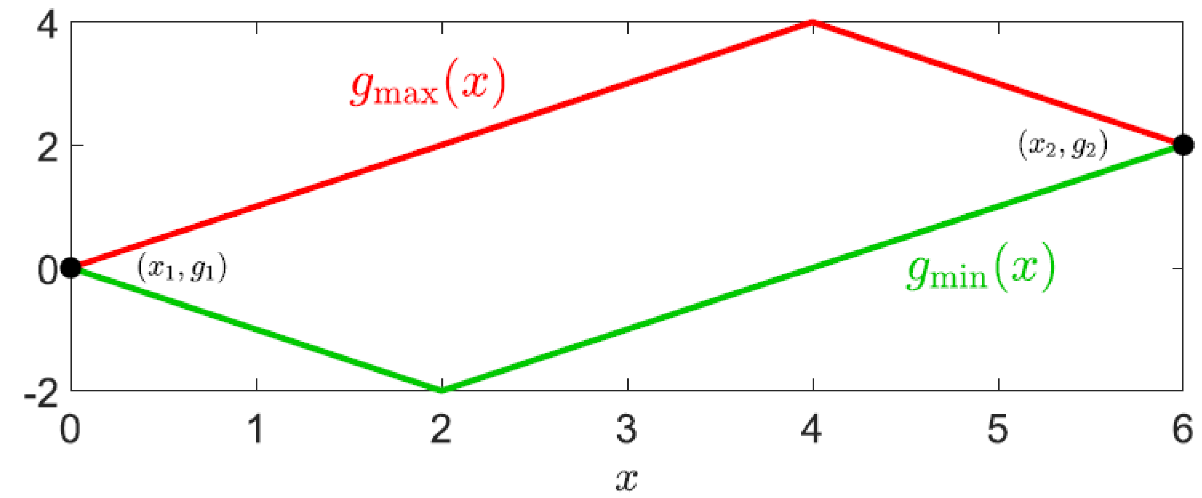
Recovers [Taylor Hendrickx G, 2017]

Theorem 4 (Interpolation conditions for gradient Lipschitz functions)

$S = \{(x_i, f_i, g_i)\}_{i \in [N]}$ *is interpolable by a gradient Lipschitz function if and only if*

$$f_j - f_i - g_i(x_j - x_i) \geq -\frac{M}{2}(x_j - x_i)^2 + \frac{1}{4M}\left(g_j - g_i + M(x_j - x_i)\right)^2, \quad \forall i, j$$

Principled technique to obtain interpolation conditions (2)

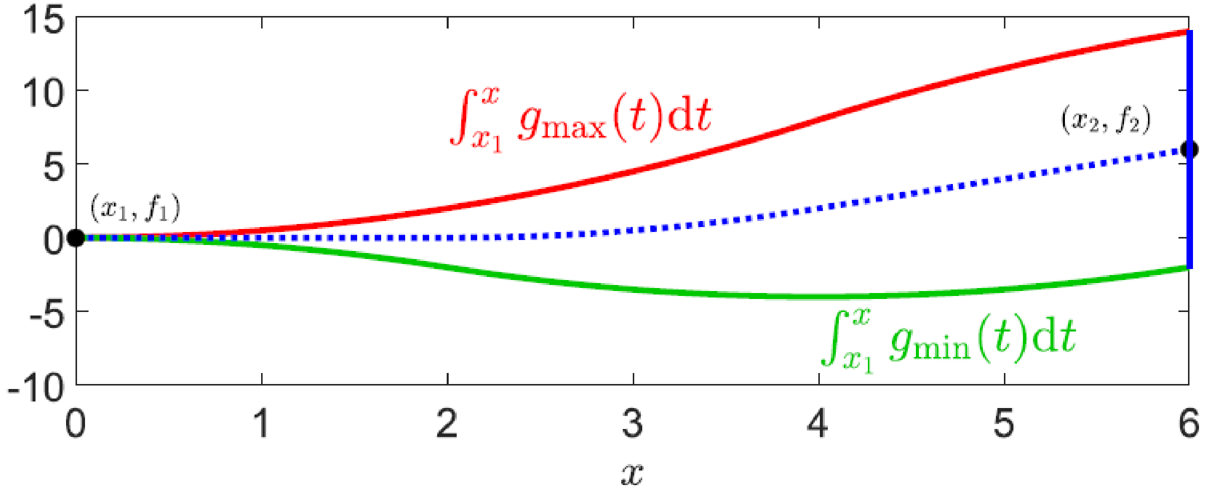


Theorem 1 *Let $\mathcal{F} \subseteq \bar{\mathcal{C}}^m$ be an extremally interpolable (Assumption 1) class of univariate functions, and let $\int \mathcal{F} \subseteq \bar{\mathcal{C}}^{m+1}$ (defined in (8)) be extremally completable (Assumption 2) and order $m+1$ connectable (Assumption 4).*

A set $S = \{(x_i, f_i^0, f_i^1, ..., f_i^m)\}_{i\in[N]}$, where $x_0 \leq x_1 \leq ... \leq x_N$ is $\int \mathcal{F}$-interpolable if and only if S is $\int \mathcal{F}$-interpolable without function values, and $\forall i \in [N]$,

$$\int_{x_i}^{x_{i+1}} g_{\min}(x)\mathrm{d}x \leq f_{i+1} - f_i \leq \int_{x_i}^{x_{i+1}} g_{\max}(x)\mathrm{d}x, \quad (14)$$

where $g_{\min}$ and $g_{\max}$ are defined as in (11).



Th. 2: **"initial"** interpolation conditions for univariate generalized self-concordant functions
Principled technique starts from Th. 2 and provides **"complete"** interpolation conditions

Theorem 2 (Interpolation conditions without gradient and function values)

$S = \{(x_i, h_i)\}_{i\in[N]}$ is $\mathcal{F}_{M,\alpha}$-interpolable if and only if

$$\begin{cases} |\tilde{f}_i - \tilde{f}_j| \leq M|x_i - x_j|, & \forall i, j, \\ h_i \geq 0, & \forall i. \end{cases}$$

Outline

1. **Performance Estimation Problem (PEP) Framework**

2. **Principled technique to characterize univariate class of functions**

3. **Interpolation conditions for second-order univariate function class**

4. **Convergence results for second-order methods**

Interpolation conditions for univariate Hessian Lipschitz functions

Step 1: Inequalities from definition of $\mathcal{F}$

$\mathcal{D}_M$: univariate functions with Lipschitz continuous Hessian. $(D_M = \int^{(2)} F_M)$

Definition. $f \in \mathcal{D}_M$ if, and only if

$$|f''(x) - f''(y)| \leq M|x-y| \quad \forall x, y. \tag{S}$$

Theorem. *If $f \in \mathcal{D}_M$ then,*

Not interpolation condition

$$|f(y) - f(x) - f'(x)(y-x) - \frac{1}{2}f''(x)(y-x)^2| \leq \frac{M}{6}|y-x|^3 \quad \forall x, y. \tag{S2}$$

Theorem. $f \in \mathcal{D}_M$ *if, and only if*

Interpolation condition

$$\begin{aligned} f(y) - f(x) - f'(x)(y-x) - \frac{1}{2}f(x)''(y-x)^2 \leq & \frac{M}{6}|y-x|^3 \\ & - \frac{(f'(y) - f'(x) - f''(x)(y-x) - \frac{M}{2}(y-x)|y-x|)^2}{2(M|y-x| - (f''(y) - f''(x)))} \\ & - \frac{(M|y-x| - (f''(y) - f''(x))^3}{96M^2} \quad \forall x, y. \end{aligned} \tag{S3}$$

Challenge: « (S2) implies (S)» is an open question as far as we know

Interpolation conditions for (quasi-)self-concordant functions

Self-concordant functions : $|f'''(x)| \leq 2Mf''(x)^{3/2}$

Corollary 3 *A set* $S = \{(x_i, g_i, h_i)\}_{i\in[N]}$ *is* $\mathcal{S}_{M,+}$*-interpolable if and only if,* $\forall i, j \in [N]$, $h_i = 0$ *and* $g_i = g_j$, *or* $\forall i, j \in [N]$,

$$|\tilde{h}_j - \tilde{h}_i| \leq M|x_j - x_i| \text{ and } h_i > 0 \tag{34}$$

$$\text{If } \tilde{h}_i + \tilde{h}_j > -M(x_j - x_i), \text{ then } g_j - g_i \geq \frac{1}{M\tilde{h}_i} + \frac{1}{M\tilde{h}_j} - \frac{4}{M(\tilde{h}_i + \tilde{h}_j + M(x_j - x_i))}, \tag{35}$$

where $\tilde{h}_i = h_i^{-1/2}$.

Quasi-self-concordant functions : $|f'''(x)| \leq Mf''(x)$

Lemma 6 *If* $f \in \mathcal{T}_{M,+}$, *then* $\forall x, y \in \mathbb{R}$,

$$f'(y) - f'(x) - f''(x)(y-x) \leq \frac{1}{M} f''(x)\left(e^{M|y-x|} - M|y-x| - 1\right) - \frac{1}{M}\left(\sqrt{f''(y)} - \sqrt{f''(x)e^{M(y-x)}}\right)^2. \tag{38}$$

Outline

1. **Performance Estimation Problem (PEP) Framework**
2. **Principled technique to characterize univariate class of functions**
3. **Interpolation conditions for second-order univariate function class**
4. **Convergence results for second-order methods**

Global convergence rate of Cubic Newton Method

$$x_{i+1} = \arg\min_x f(x) + f'(x_i)(x - x_i) + \frac{1}{2} f''(x_i)(x - x_i)^2 + \frac{M}{6}|x - x_i|^3. \qquad \text{(CNM)}$$

Theorem 6 ([38], Theorem 1) *The iterates of the Cubic Regularized Newton method (CNM) on Hessian M-Lipschitz univariate functions satisfy*

$$f(x_k) - f(x_{k+1}) \geq \frac{M}{12} \max\left\{\sqrt{\frac{|f'(x_{k+1})|}{M}}, -\frac{2}{3}\frac{f''(x_{k+1})}{M}\right\}^3. \qquad (47)$$

[Nesterov, Polyak 2008] (in multivariate case)

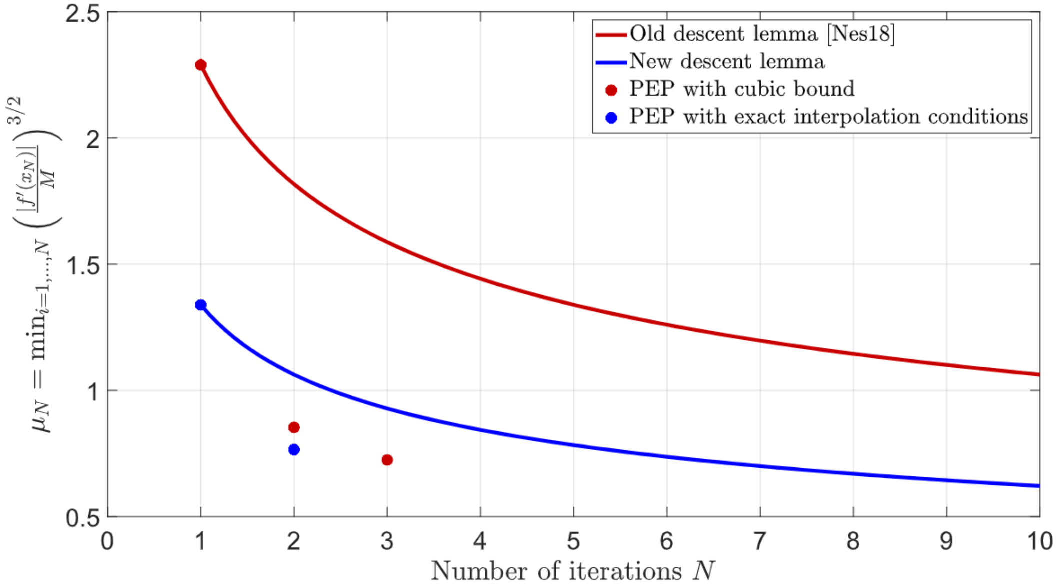
Theorem 7 (Improved descent lemma and gradient convergence rate) *The iterates of the Cubic Regularized Newton method (CNM) on Hessian M-Lipschitz univariate functions satisfy*

$$f(x_k) - f(x_{k+1}) \geq \frac{5M}{12} \sqrt{\frac{|f'(x_{k+1})|}{M}}^3. \qquad (49)$$

[Rubbens, Bousselmi, Hendrickx, G 2025]

Step 1: Inequalities from definition of $\mathcal{F}$

Step 2: combination of inequalities and Iteration of $\mathcal{M}$



Local quadratic convergence rate of Newton Method

Theorem. *If*

- f *has a* M*-Lipschitz continuous Hessian,*
- $\exists x^*$ *such that* $\nabla f(x^*) = 0,\ \nabla^2 f(x^*) = \mu I \succ 0,$
- $\frac{M}{\mu}||x_0 - x^*|| \leq \frac{2}{3},$

then all Newton iterations $x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k)$ *satisfy*

$$||x_{k+1} - x^*|| \leq \frac{\frac{M}{\mu}||x_k - x^*||^2}{2\left(1 - \frac{M}{\mu}||x_k - x^*||\right)}$$

[Nesterov 2018]

Observation: PEP numerical results exactly match the bound

Theorem. *Theorem above is tight and attained by the following univariate cubic by parts function.*

$$f_1(x) = \begin{cases} \frac{Mx^3}{6} + \mu\frac{x^2}{2} & \text{if } x \leq 0, \\ -\frac{Mx^3}{6} + \mu\frac{x^2}{2} & \text{if } x > 0. \end{cases}$$

[Rubbens, Bousselmi, Hendrickx, G 2024]

Univariate case is « sufficiently rich » to attain the worst-case performance

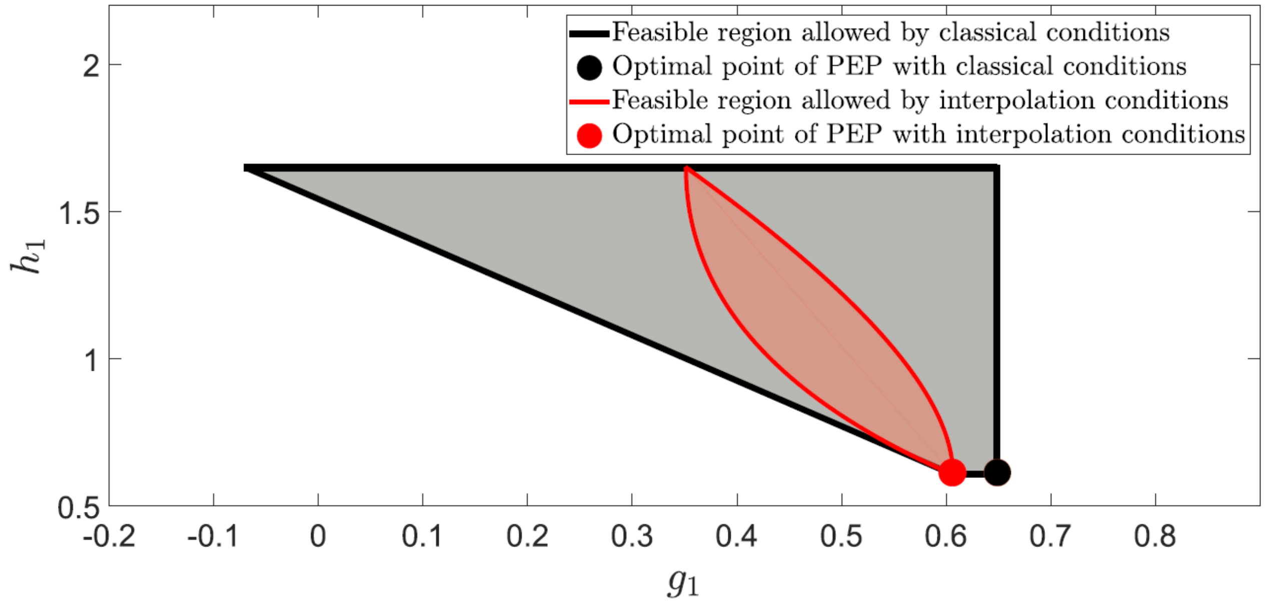
Local convergence of Gradient Regularized Newton method on quasi-self-concordant functions

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k) + M|f'(x_k)|}. \qquad \text{(GNM1)}$$

Lemma 9 *The iterations of (GNM1) on univariate M-quasi-self-concordant functions satisfy*

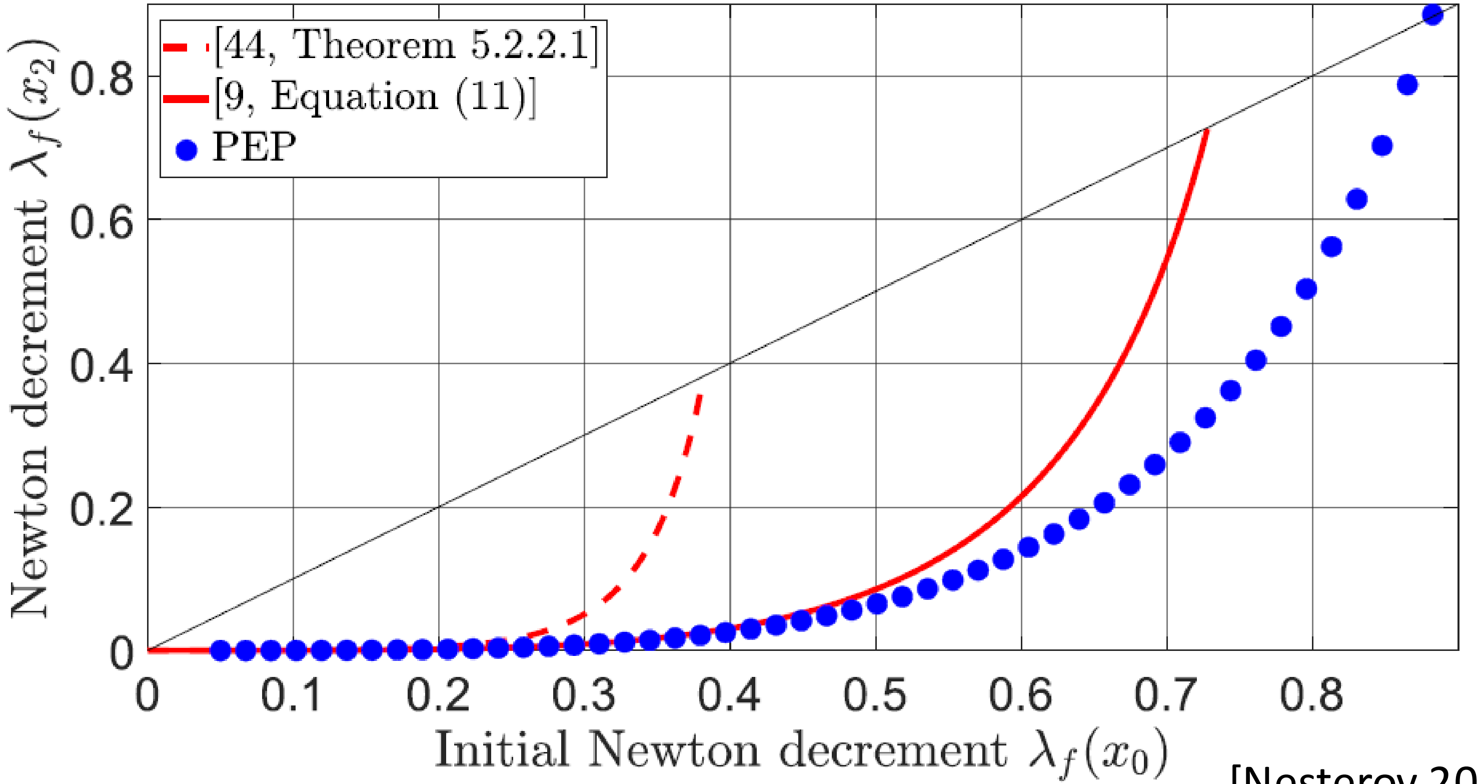
$$\eta(x_{k+1}) \leq e^{\frac{\eta(x_k)}{\eta(x_k)+1}}(\eta(x_k) - 1) + 1 \qquad (85)$$

where $\eta(x) = M\frac{|f'(x)|}{f''(x)}$.



In this simple case, we can reduce PEP to a two-dimensional problem and solve it 'by hand'

Newton method on self-concordant functions



[Nesterov 2018, Nesterov & Nemirovski 1998]
[Hildebrand 2021, tight for N=1]

Fig. 11: Worst-case performance $\lambda(x_2)$ for varying initial $\lambda(x_0)$ of two iterations of Newton method (black dots) compared to bounds [36, Theorem 5.2.2.1] (dashed lines), [28, Equation (11)] (solid lines), and the PEP results (dots).

Optimal step size of fixed damped Newton method

Fixed Damped Newton method : $x_{k+1} = x_k - \alpha \frac{f'(x_k)}{f''(x_k)}$

α that optimize the worst-case performance on convex Hessian Lipschitz functions

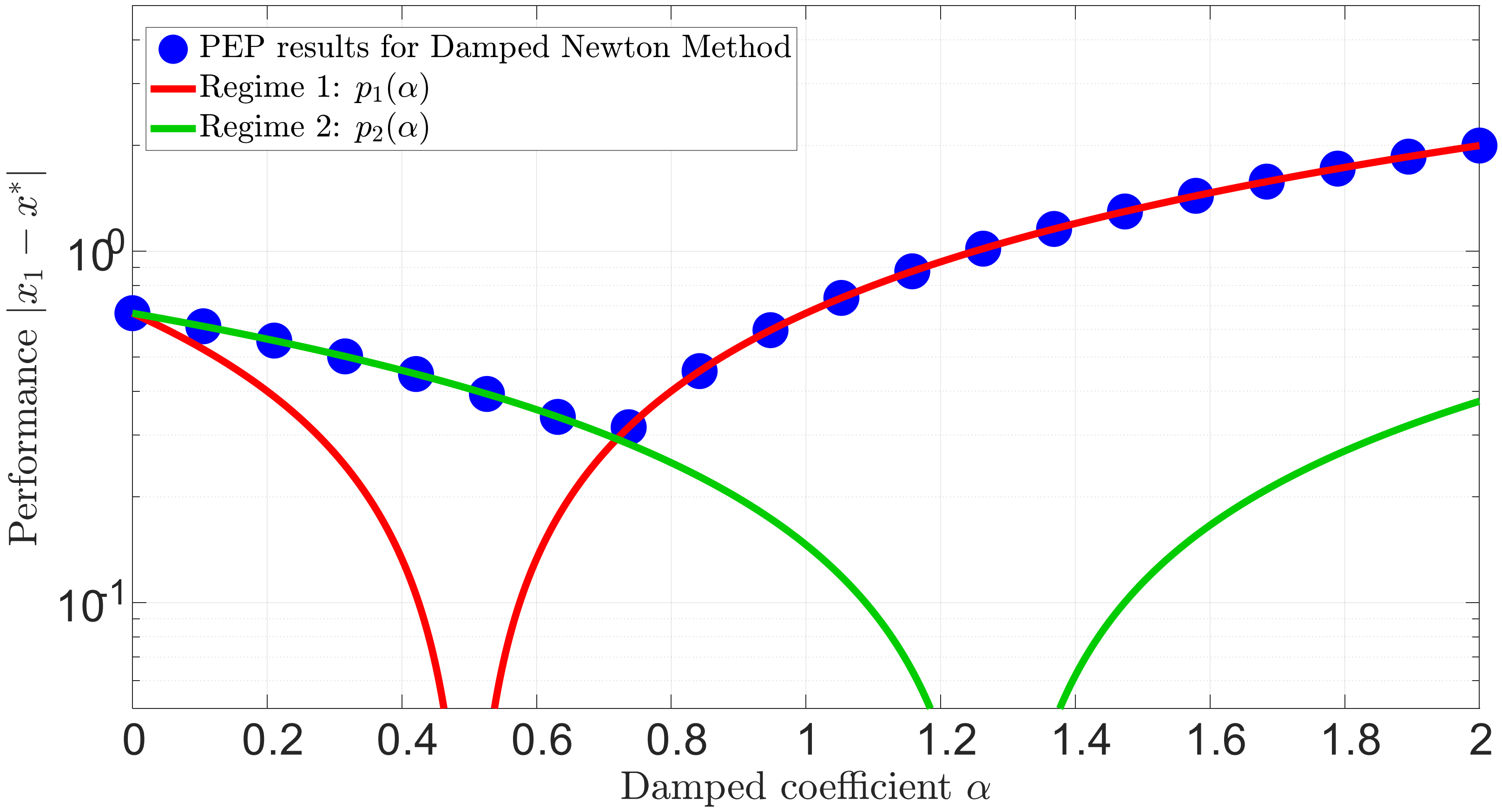


Fig. 1: $M = \mu = 1$ and $|x_0 - x^*| = \frac{2}{3}$

Take home messages

Performance estimation (PEP) provides
proofs for those rates and correspondin
and crucially rely on exact **interpolation**

In this work, we show how to derive exa
higher order classes of functions, such
(generalized) self-concordant functions,

Using **PEP** with those **inequalities**, we s
many variants of **second-order method**
cubic/**adaptive** variants

Performance Estimation of second-order optimization methods on classes of univariate functions

Nizar Bousselmi[1†] · **Anne Rubbens**[1†] · **Julien M. Hendrickx**[1] · **François Glineur**[1,2]

Received: date / Accepted: date

Abstract We develop a principled technique to obtain exact computer-aided worst-case guarantees on the performance of second-order optimization methods on classes of univariate functions. We first present a generic technique to derive interpolation conditions for a wide range of univariate functions, and rely on this technique to obtain such conditions for generalized self-concordant (including self- and quasi-self-concordant) functions and (strongly convex) functions with Lipschitz Hessian. We then exploit these conditions and the Performance Estimation framework to tightly analyze second-order methods, including (Cubic Regularized) Newton's method and variants thereof, on univariate functions. Thereby, we improve on existing convergence rates in the univariate case, exhibit lower bounds in the multivariate case, and compare different variants of Newton's method on a fair basis, i.e., with respect to the same setting.

Keywords Performance Estimation · Interpolation conditions · Second-Order Optimization · Newton's method · Worst-case analysis

Mathematics Subject Classification (2020) 68Q25 · 90C53 · 90C25 · 26A06

Git: https://github.com/NizarBousselmi/Second-Order-Univariate-PEP

Paper: https://arxiv.org/abs/2506.22764