# INTERPOLATION CONDITIONS FOR LINEAR OPERATORS AND APPLICATIONS TO PERFORMANCE ESTIMATION PROBLEMS


Nizar Bousselmi

François Glineur and Julien Hendrickx

Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM)
Université catholique de Louvain (UCLouvain)



Thesis supported by a FRIA grant


20th EUROpt Workshop, Budapest, 25, August 2023

Worst-case **performance** of an optimization **method** $\mathcal{M}$ on

$$\min_x f(x)$$

where $f \in \mathcal{F}$ has some properties (smoothness, convexity, ...)?

Worst-case **performance** of an optimization **method** $\mathcal{M}$ on

$$\min_x f(x)$$

where $f \in \mathcal{F}$ has some properties (smoothness, convexity, ...)?

**Example:** Worst-case performance of $\overbrace{\text{gradient method}}^{\mathcal{M}}$ on $\overbrace{L\text{-smooth convex functions}}^{\mathcal{F}}$ after $N$ iterations?

$$\overbrace{f(x_N) - f^*}^{\text{performance}} \leq \frac{L}{2} \frac{1}{2N+1}$$

Worst-case **performance** of an optimization **method** $\boldsymbol{\mathcal{M}}$ on

$$\min_x f(x) + g(Mx)$$

where $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}$ has some properties (smoothness, convexity, ...)?

**Example:** Worst-case performance of $\overbrace{\textbf{gradient method}}^{\mathcal{M}}$ on $\overbrace{\textbf{\textit{L}-smooth convex functions}}^{\mathcal{F}}$ after $N$ iterations?

$$\overbrace{f(x_N) - f^*}^{\text{performance}} \leq \frac{L}{2}\frac{1}{2N+1}$$

**In this work**

**Worst-case performance of methods involving linear operators?**

(Chambolle-Pock method, Condat-Vũ method, PDFP, PD30, PAPC, ADMM, etc.)

# Performance Estimation Problem (PEP)

Theoretical and practical framework to analyze exact performance of optimization methods on problem classes.

- [Drori and Teboulle, 2014] «Performance of first-order methods for smooth...»
- [Taylor, 2017] «Convex interpolation and performance estimation of first-order...»

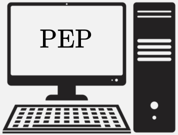
# Performance Estimation Problem (PEP)

Theoretical and practical framework to **analyze exact performance of optimization methods on problem classes**.

- [Drori and Teboulle, 2014] «Performance of first-order methods for smooth...»
- [Taylor, 2017] «Convex interpolation and performance estimation of first-order...»

Method
+
Class of functions

→




→

Exact worst-case performance

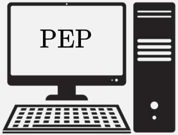
Theoretical and practical framework to **analyze exact performance of optimization methods on problem classes**.

- **[Drori and Teboulle, 2014]** «Performance of first-order methods for smooth...»
- **[Taylor, 2017]** «Convex interpolation and performance estimation of first-order...»

Gradient method

+

Class of $L$-smooth convex functions




Exact worst-case performance

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

# PEP IS WELL DEVELOPED: AROUND 40 RESEARCHERS

## Analysis of methods and frameworks

- Decentralized optimization [Colla, Hendrickx, 2022]
- Halper iteration *[Lieder, 2021]*
- Maximal monotone inclusion *[Gu, Yang, 2020]*
- Proximal GM *[Taylor, Hendrickx, Glineur, 2018]*
- Splitting methods *[Ryu, Taylor, Bergeling, Giselsson 2020]*
- Non linear conjugate gradients [Gupta, Freund, Sun, Taylor, 2023]
- Monotone variational inequality *[Gu and Yang, 2022]*
- GM on smooth hypoconvex *[Rotaru, Glineur, Patrinos, 2022]*
- Stochastic optimization *[Taylor, Bach, 2021]*
- Difference of convex method *[Abbaszadehpeivasti, de Klerk, Zamani, 2021]*
- ADMM [Abbaszadehpeivasti, de Klerk, Zamani, 2022]
- GM on smooth strongly convex *[Taylor, Hendrickx, Glineur, 2017]*
- GM on smooth convex *[Drori and Teboulle, 2014]*
- GM exact line search *[de Klerk, Glineur, Taylor, 2017]*
- Extragradient method *[Gorbunov, Loizou, Gidel, 2020]*
- Coordinate descent method *[Shi, Liu, 2016]* *[Kamri, Hendrickx, Glineur, 2022]* *[Abbaszadehpeivasti, de Klerk, Zamani, 2022]*
- Continuous-time models *[Moucer, Taylor, Bach, 2022]*
- Gradient descent-ascent method [Zamani, de Klerk, Abbaszadehpeivasti, 2022]
- PESTO *[Taylor, Hendrickx, Glineur, 2017]*
- Inexact GM *[de Klerk, Glineur, Taylor, 2020]*
- PEPit *[Goujaud, Moucer, Glineur, Hendrickx, Taylor, Dieuleveut, 2022]*
- Methods on convex functions with quadratic upper bound *[Goujaud, Taylor, Dieuleveut, 2022]*
- Bregman methods *[Dragomir, Taylor, d'Aspremont, 2021]*
- GM on functions with lower restricted secant inequality and upper error bound *[Guille-Escuret, Ibrahim, Goujaud, Mitliagkas, 2022]*

## Optimization of methods

- Optimal GM for smooth strongly convex *[Taylor, Drori, 2022]*
- Optimal methods for non-smooth and smooth convex *[Drori, Taylor, 2020]*
- Fast proximal point method *[Kim, 2021]*
- Kelley's cutting-plane method *[Drori, Teboulle, 2016]*
- Optimized GM for smooth convex *[Kim, Fessler, 2016]*
- FISTA *[Kim, Fessler, 2018]*
- Optimal first-order methods for convex and nonconvex *[Gupta, Van Parys, Ryu, 2022]*

GM = Gradient method ADMM = Alternating Directions Method of Multipliers

# PEP IS WELL DEVELOPED: AROUND 40 RESEARCHERS

## Analysis of methods and frameworks

- Decentralized optimization [Colla, Hendrickx, 2022]
- Halper iteration *[Lieder, 2021]*
- Maximal monotone inclusion *[Gu, Yang, 2020]*
- Proximal GM *[Taylor, Hendrickx, Glineur, 2018]*
- Splitting methods *[Ryu, Taylor, Bergeling, Giselsson 2020]*
- Non linear conjugate gradients [Gupta, Freund, Sun, Taylor, 2023]
- Monotone variational inequality *[Gu and Yang, 2022]*
- GM on smooth hypoconvex *[Rotaru, Glineur, Patrinos, 2022]*
- Stochastic optimization *[Taylor, Bach, 2021]*
- GM on smooth strongly convex *[Taylor, Hendrickx, Glineur, 2017]*
- GM on smooth convex *[Drori and Teboulle, 2014]*
- GM exact line search *[de Klerk, Glineur, Taylor, 2017]*
- Difference of convex method *[Abbaszadehpeivasti, de Klerk, Zamani, 2021]*
- ADMM [Abbaszadehpeivasti, de Klerk, Zamani, 2022]
- Continuous-time models *[Moucer, Taylor, Bach, 2022]*
- Extragradient method *[Gorbunov, Loizou, Gidel, 2020]*
- PESTO *[Taylor, Hendrickx, Glineur, 2017]*
- Inexact GM *[de Klerk, Glineur, Taylor, 2020]*
- Gradient descent-ascent method [Zamani, de Klerk, Abbaszadehpeivasti, 2022]
- Coordinate descent method *[Shi, Liu, 2016]* *[Kamri, Hendrickx, Glineur, 2022]* *[Abbaszadehpeivasti, de Klerk, Zamani, 2022]*
- PEPit *[Goujaud, Moucer, Glineur, Hendrickx, Taylor, Dieuleveut, 2022]*
- **Functions involving linear mappings** *[Bousselmi, Hendrickx, Glineur, 2023]*
- Methods on convex functions with quadratic upper bound *[Goujaud, Taylor, Dieuleveut, 2022]*
- Bregman methods *[Dragomir, Taylor, d'Aspremont, 2021]*
- GM on functions with lower restricted secant inequality and upper error bound *[Guille-Escuret, Ibrahim, Goujaud, Mitliagkas, 2022]*

## Optimization of methods

- Optimal GM for smooth strongly convex *[Taylor, Drori, 2022]*
- Fast proximal point method *[Kim, 2021]*
- Optimal methods for non-smooth and smooth convex *[Drori, Taylor, 2020]*
- Kelley's cutting-plane method *[Drori, Teboulle, 2016]*
- Optimized GM for smooth convex *[Kim, Fessler, 2016]*
- FISTA *[Kim, Fessler, 2018]*
- Optimal first-order methods for convex and nonconvex *[Gupta, Van Parys, Ryu, 2022]*

GM = Gradient method    ADMM = Alternating Directions Method of Multipliers

# Performance Estimation Problem

# Input = optimization method + problem class
# Output = worst instance in problem class

Example:

- $N$ steps of gradient method $x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$
- $L$-smooth convex functions $f$

**PEP**

$$
\begin{aligned}
\max_{\text{points } x_i, x^*, \text{ function } f} \quad & f(x_N) - f(x^*) \\
\text{s.t.} \quad & f \; L\text{-smooth convex}, \\
& x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i), \\
& ||x^* - x_0||^2 \leq 1, \\
& ||\nabla f(x^*)||^2 = 0.
\end{aligned}
$$

## INPUT = OPTIMIZATION METHOD + PROBLEM CLASS
## OUTPUT = WORST INSTANCE IN PROBLEM CLASS

Example:

- $N$ steps of gradient method $x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$
- $L$-smooth convex functions $f$

**PEP**

$$
\begin{aligned}
\max_{\text{points } x_i, x^*, \text{ function } f} \quad & f(x_N) - f(x^*) \\
\text{s.t.} \quad & f \ L\text{-smooth convex}, \\
& x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i), \\
& ||x^* - x_0||^2 \leq 1, \\
& ||\nabla f(x^*)||^2 = 0.
\end{aligned}
$$

## INPUT = OPTIMIZATION METHOD + PROBLEM CLASS
## OUTPUT = WORST INSTANCE IN PROBLEM CLASS

Example:

- $N$ steps of gradient method $x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$
- $L$-smooth convex functions $f$

**PEP**

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

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

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

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

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

# Input = optimization method + problem class
# Output = worst instance in problem class

Example:

- $N$ steps of gradient method $x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$
- $L$-smooth convex functions $f$

**PEP**

$$
\begin{aligned}
\max_{\text{points } x_i, x^*, \text{ function } f} \quad & f(x_N) - f(x^*) \\
\text{s.t.} \quad & f \ L\text{-smooth convex}, \\
& x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i), \\
& ||x^* - x_0||^2 \leq 1, \\
& ||\nabla f(x^*)||^2 = 0.
\end{aligned}
$$

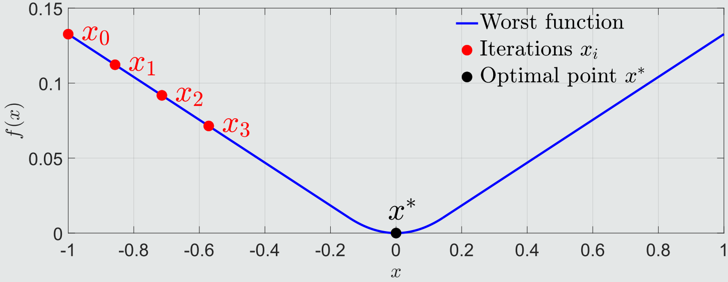
# Input = optimization method + problem class
# Output = worst instance in problem class

Example:

- $N$ steps of gradient method $x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$
- $L$-smooth convex functions $f$

**PEP solved outputs (for $N = 3$ and $L = 1$)**

- Worst performance: $\boldsymbol{f(x_N) - f^* = \frac{1}{14}}(= \frac{L}{2}\frac{1}{2N+1})$
- Worst function:



# PEP as finite-dimensional problem

$f$ **infinite-dimensional** but only access to $x_i$, $f(x_i)$, $\nabla f(x_i)$ ... **black-box** property!

**PEP**

$$
\begin{aligned}
\max_{\text{points } x_i, x^*, \text{function } f} \quad & f(x_N) - f(x^*) \\
\text{s.t.} \quad & f \; L\text{-smooth convex}, \\
& x_{i+1} = x_i - \frac{1}{L}\nabla f(x_i), \\
& ||x^* - x_0||^2 \leq 1, \\
& ||\nabla f(x^*)||^2 = 0.
\end{aligned}
$$

$f$ **infinite-dimensional** but only access to $x_i$, $f(x_i)$, $\nabla f(x_i)$ ... **black-box** property!

**PEP**

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

$$\begin{aligned}
\text{s.t.} \quad & \exists f \ L\text{-smooth convex}: \quad f(x_i) = f_i, \quad \nabla f(x_i) = g_i, \\
& \qquad\qquad\qquad\qquad f(x^*) = f^*, \quad \nabla f(x^*) = g^*, \\
& x_{i+1} = x_i - \frac{1}{L} g_i, \\
& ||x^* - x_0||^2 \leq 1, \\
& ||g^*||^2 = 0.
\end{aligned}$$

# PEP as finite-dimensional problem

$f$ **infinite-dimensional** but only access to $x_i$, $f(x_i)$, $\nabla f(x_i)$ ... **black-box** property!

**PEP**

$$
\begin{aligned}
\max_{\text{points } x_i, x^*, f_i, f^*, g_i, g^*} \quad & f_N - f^* \\
\text{s.t.} \quad & f_i \geq f_k + g_k^T(x_i - x_k) + \frac{1}{2L}||g_i - g_k||^2, \\
& x_{i+1} = x_i - \frac{1}{L} g_i, \\
& ||x^* - x_0||^2 \leq 1, \\
& ||g^*||^2 = 0.
\end{aligned}
$$

- **Reformulation 1:** Explicit interpolation conditions;

# PEP as finite-dimensional problem

$f$ **infinite-dimensional** but only access to $x_i$, $f(x_i)$, $\nabla f(x_i)$ ... **black-box** property!

**PEP**

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

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

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

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

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

- **Reformulation 1:** Explicit interpolation conditions;
- **Reformulation 2:** Lift to **convex semidefinite program** (efficiently solvable!)

## Interpolation conditions for $L$-smooth convex functions

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

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

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

We say that $\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is $L$-smooth-convex-interpolable.

**Interpolation conditions for $L$-smooth convex functions**

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

$\exists$ $L$-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,

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

We say that $\left\{(x_1, g_1, f_1), \dots, (x_N, g_N, f_N)\right\}$ is **$L$-smooth-convex-interpolable**.

**Interpolation conditions** must be convex in $f_i$, $x_i^T g_k$, $x_i^T x_k$, $g_i^T g_k$.
(Otherwise, we cannot solve the PEP for the class of interest).

# Performance Estimation Problem for linear operators

**Goal:** Extend and exploit PEP to analyze methods applied to $\min_x F(x)$ where $F$ involves linear operators.

| $F(x)$ | Possible method | Iterations |
|---|---|---|
| $g(Mx)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ |
| $\frac{1}{2} x^T Q x \; (= \frac{1}{2} \|\|Q^{\frac{1}{2}} x\|\|^2)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} Q x_i$ |
| $f(x) + g(Mx)$ | Chambolle-Pock [Chambolle and Pock, 2011] | $\begin{cases} x_{i+1} & = \mathrm{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} & = \mathrm{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$ |

What is missing in PEP to analyze such problems?

**Goal:** Extend and exploit PEP to analyze methods applied to $\min_x F(x)$ where $F$ involves linear operators.

| $F(x)$ | Possible method | Iterations |
|---|---|---|
| $g(Mx)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ |
| $\frac{1}{2}x^T Qx$ $(= \frac{1}{2}\|\|Q^{\frac{1}{2}}x\|\|^2)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} Qx_i$ |
| $f(x) + g(Mx)$ | Chambolle-Pock [Chambolle and Pock, 2011] | $\begin{cases} x_{i+1} &= \mathrm{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} &= \mathrm{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$ |

What is missing in PEP to analyze such problems?

**Goal:** Extend and exploit PEP to analyze methods applied to $\min_x F(x)$ where $F$ involves linear operators.

| $F(x)$ | Possible method | Iterations |
|---|---|---|
| $g(Mx)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ |
| $\frac{1}{2} x^T Q x$ $(= \frac{1}{2}\|\|Q^{\frac{1}{2}} x\|\|^2)$ | Gradient descent | $x_{i+1} = x_i - \frac{h}{L} Q x_i$ |
| $f(x) + g(Mx)$ | Chambolle-Pock [Chambolle and Pock, 2011] | $\begin{cases} x_{i+1} &= \mathrm{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} &= \mathrm{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$ |

What is missing in PEP to analyze such problems?

**Interpolation conditions** for linear operators!

# Decomposing the iteration

- Gradient method on $\min_x g(Mx)$: $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$

- Gradient method on $\min_x g(Mx)$: $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ or equivalently

$$\begin{cases} y_i & = Mx_i \\ u_i & = \nabla g(y_i) \\ v_i & = M^T u_i \\ x_{i+1} & = x_i - \frac{h}{L} v_i \end{cases}$$

- Gradient method on $\min_x g(Mx)$: $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ or equivalently

$$\begin{cases} y_i & = Mx_i & \text{New interpolation conditions} \\ u_i & = \nabla g(y_i) & \text{Standard interpolation conditions} \\ v_i & = M^T u_i & \text{New interpolation conditions} \\ x_{i+1} & = x_i - \frac{h}{L} v_i & \text{Standard} \end{cases}$$

- Gradient method on $\min_x g(Mx)$: $x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$ or equivalently

$$\begin{cases} y_i & = Mx_i \quad \text{New interpolation conditions} \\ u_i & = \nabla g(y_i) \quad \text{Standard interpolation conditions} \\ v_i & = M^T u_i \quad \text{New interpolation conditions} \\ x_{i+1} & = x_i - \frac{h}{L} v_i \quad \text{Standard} \end{cases}$$

- Chambolle-Pock method on $\min_x f(x) + g(Mx)$ :

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

Also requires New interpolation conditions.

**Definition ($\mathcal{L}_L$-interpolability)**

$\left\{(x_1, y_1), \ldots, (x_N, y_N)\right\}$ and $\left\{(u_1, v_1), \ldots, (u_N, v_N)\right\}$ are **$\mathcal{L}_L$-interpolable** if, and only if,

$$\exists M \text{ with } \sigma_{\max}(M) \leq L \ : \ \begin{cases} y_i = Mx_i, & \forall i, \\ v_i = M^T u_i, & \forall i. \end{cases}$$

**Definition ($\mathcal{L}_L$-interpolability)**

$\left\{(x_1, y_1), \dots, (x_N, y_N)\right\}$ and $\left\{(u_1, v_1), \dots, (u_N, v_N)\right\}$ are **$\mathcal{L}_L$-interpolable** if, and only if,

$$\exists M \text{ with } \sigma_{\max}(M) \leq L \ : \ \begin{cases} y_i = M x_i, & \forall i, \\ v_i = M^T u_i, & \forall i. \end{cases}$$

**Theorem ($\mathcal{L}_L$-interpolation conditions)**

*$\left\{(x_1, y_1), \dots, (x_N, y_N)\right\}$ and $\left\{(u_1, v_1), \dots, (u_N, v_N)\right\}$ are **$\mathcal{L}_L$-interpolable** if, and only if,*

$$\begin{cases} X^T V = Y^T U, \\ Y^T Y \preceq L^2 X^T X, \\ V^T V \preceq L^2 U^T U, \end{cases}$$

*where $X = (x_1 \ \cdots \ x_N)$, $Y = (y_1 \ \cdots \ y_N)$, $U = (u_1 \ \cdots \ u_N)$ and $V = (v_1 \ \cdots \ v_N)$.*

**Definition ($\mathcal{S}_{0,L}$-interpolability)**

$\left\{(x_1, y_1), \ldots, (x_N, y_N)\right\}$ is **$\mathcal{S}_{0,L}$-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } 0 \preceq Q \preceq LI \;:\; y_i = Qx_i \quad \forall i.$$

**Theorem ($\mathcal{S}_{0,L}$-interpolation conditions)**

*$\left\{(x_1, y_1), \ldots, (x_N, y_N)\right\}$ is **$\mathcal{S}_{0,L}$-interpolable** if, and only if,*

$$\begin{cases} X^T Y = Y^T X, \\ Y^T(LX - Y) \succeq 0, \end{cases}$$

*where $X = (x_1 \;\cdots\; x_N)$ and $Y = (y_1 \;\cdots\; y_N)$.*

**Remark:** Similar result for skew-symmetric matrices.

**Definition ($\mathcal{S}_{\mu,L}$-interpolability)**

$\left\{(x_1, y_1), \ldots, (x_N, y_N)\right\}$ is **$\mathcal{S}_{\mu,L}$-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } \mu I \preceq Q \preceq LI \;:\; y_i = Qx_i \quad \forall i.$$

**Theorem ($\mathcal{S}_{\mu,L}$-interpolation conditions)**

*$\left\{(x_1, y_1), \ldots, (x_N, y_N)\right\}$ is **$\mathcal{S}_{\mu,L}$-interpolable** if, and only if,*

$$\begin{cases} X^T Y = Y^T X, \\ (Y - \mu X)^T (LX - Y) \succeq 0, \end{cases}$$

*where $X = (x_1 \;\cdots\; x_N)$ and $Y = (y_1 \;\cdots\; y_N)$.*

**Remark:** Similar result for skew-symmetric matrices.

## Interpolation conditions for homogeneous quadratic functions $\mathcal{Q}_{\mu,L} = \left\{f(x) = \frac{1}{2}x^TQx, Q = Q^T, \mu I \preceq Q \preceq LI\right\}$

**Definition ($\mathcal{Q}_{\mu,L}$-interpolability)**

$\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } \mu I \preceq Q \preceq LI \ : \ \begin{cases} f_i = \frac{1}{2}x_i^TQx_i, & \forall i, \\ g_i = Qx_i, & \forall i. \end{cases}$$

# Interpolation conditions for homogeneous quadratic functions $\mathcal{Q}_{\mu,L} = \left\{f(x) = \frac{1}{2}x^TQx, Q = Q^T, \mu I \preceq Q \preceq LI\right\}$

**Definition ($\mathcal{Q}_{\mu,L}$-interpolability)**

$\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } \mu I \preceq Q \preceq LI \ : \ \begin{cases} f_i = \frac{1}{2}x_i^TQx_i, & \forall i, \\ g_i = Qx_i, & \forall i. \end{cases}$$

**Theorem ($\mathcal{Q}_{\mu,L}$-interpolation conditions)**

*$\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,*

$$\begin{cases} X^TG = G^TX, \\ (G - \mu X)^T(LX - G) \succeq 0, \\ f_i = \frac{1}{2}x_i^Tg_i \quad \forall i, \end{cases}$$

*where $X = (x_1 \ \cdots \ x_N)$ and $G = (g_1 \ \cdots \ g_N)$.*

# Interpolation conditions for homogeneous quadratic functions

$\mathcal{Q}_{\mu,L} = \left\{f(x) = \frac{1}{2}x^TQx, Q = Q^T, \mu I \preceq Q \preceq LI\right\}$

**Definition ($\mathcal{Q}_{\mu,L}$-interpolability)**

$\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } \mu I \preceq Q \preceq LI \ : \ \begin{cases} f_i = \frac{1}{2}x_i^TQx_i, & \forall i, \\ g_i = Qx_i, & \forall i. \end{cases}$$

**Theorem ($\mathcal{Q}_{\mu,L}$-interpolation conditions)**

*$\left\{(x_1, g_1, f_1), \ldots, (x_N, g_N, f_N)\right\}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,*

$$\begin{cases} X^TG = G^TX, \\ (G - \mu X)^T(LX - G) \succeq 0, \\ f_i = \frac{1}{2}x_i^Tg_i \quad \forall i, \end{cases}$$

*where $X = (x_1 \ \cdots \ x_N)$ and $G = (g_1 \ \cdots \ g_N)$.*

Allows to use PEP on the class of quadratic functions!

# Exploitation of linear operators

## interpolations

1 Gradient method on $g(Mx)$

2 Gradient method on $\frac{1}{2}x^T Qx$

3 Chambolle-Pock method on $f(x) + g(Mx)$

# Simplest problem $\min_x F(x)$ involving linear operator

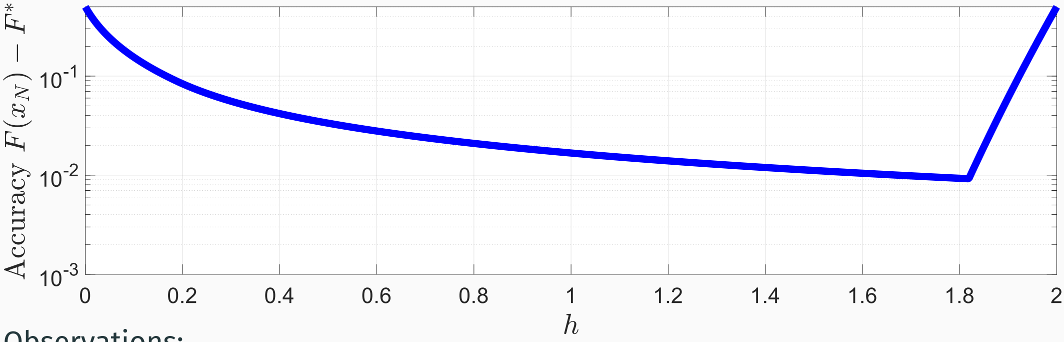
**Function class:** $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx), g \text{ is } L_g\text{-smooth } \mu_g\text{-strongly convex}, 0 \leq ||M|| \leq L_M\}$
**Gradient method (GM):** $x_{i+1} = x_i - \frac{h}{L_g L_M^2} \nabla F(x_i)$

**Function class:** $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx),\ g \text{ is } L_g\text{-smooth } \mu_g\text{-strongly convex},\ 0 \leq ||M|| \leq L_M\}$

**Gradient method (GM):** $x_{i+1} = x_i - \frac{h}{L_g L_M^2}\nabla F(x_i)$

**Parameters:** $L_g = L_M = 1$, $\mu_g = 0.1$, $\kappa_g = \frac{\mu_g}{L_g}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

**Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$**




**Observations:**

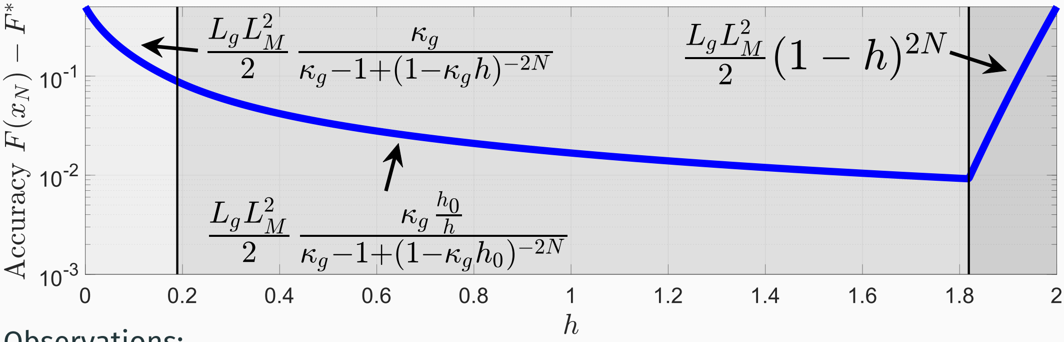
# Simplest problem $\min_x F(x)$ involving linear operator

**Function class:** $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx),\ g \text{ is } L_g\text{-smooth } \mu_g\text{-strongly convex},\ 0 \leq ||M|| \leq L_M\}$

**Gradient method (GM):** $x_{i+1} = x_i - \frac{h}{L_g L_M^2}\nabla F(x_i)$

**Parameters:** $L_g = L_M = 1$, $\mu_g = 0.1$, $\kappa_g = \frac{\mu_g}{L_g}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

**Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$**




**Observations:**

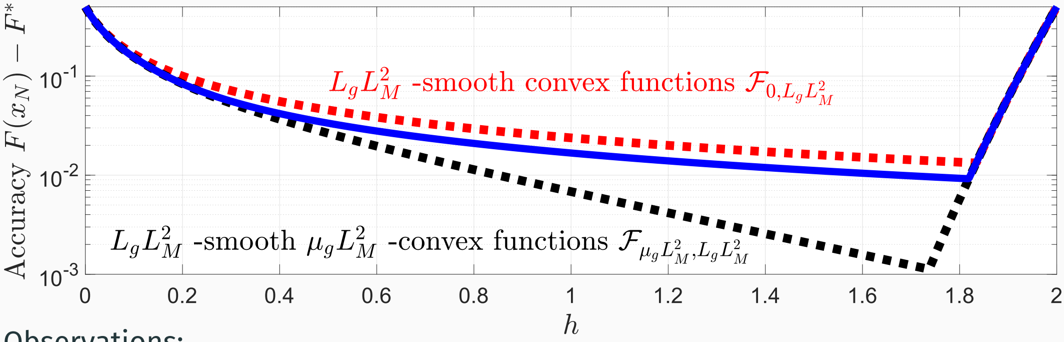
- 3 (identified) regimes;

Function class: $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx),\, g \text{ is } L_g\text{-smooth } \mu_g\text{-strongly convex},\, 0 \leq ||M|| \leq L_M\}$

Gradient method (GM): $x_{i+1} = x_i - \frac{h}{L_g L_M^2} \nabla F(x_i)$

Parameters: $L_g = L_M = 1$, $\mu_g = 0.1$, $\kappa_g = \frac{\mu_g}{L_g}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$




**Observations:**

- 3 (identified) regimes;
- $\mathrm{Acc}(\mathcal{F}_{\mu_g L_M^2, L_g L_M^2}) \leq \mathrm{Acc}(\mathcal{C}^{0,L_M}_{\mu_g,L_g}) \leq \mathrm{Acc}(\mathcal{F}_{0,L_g L_M^2})$;

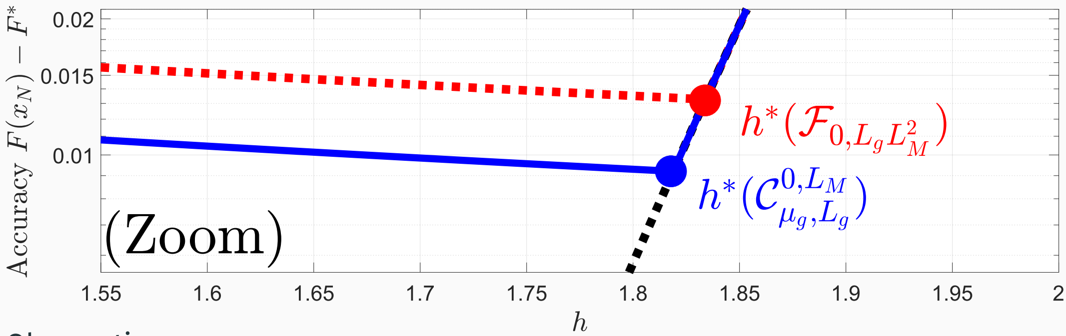
## Simplest problem $\min_x F(x)$ involving linear operator

Function class: $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx), g \text{ is } L_g\text{-smooth } \mu_g\text{-strongly convex}, 0 \leq ||M|| \leq L_M\}$

Gradient method (GM): $x_{i+1} = x_i - \frac{h}{L_g L_M^2} \nabla F(x_i)$

Parameters: $L_g = L_M = 1$, $\mu_g = 0.1$, $\kappa_g = \frac{\mu_g}{L_g}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

**Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$**




**Observations:**

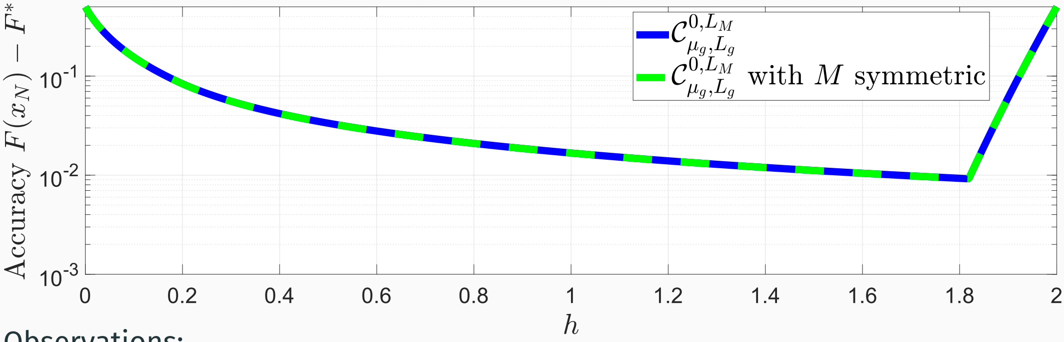
- 3 (identified) regimes;
- $\mathrm{Acc}(\mathcal{F}_{\mu_g L_M^2, L_g L_M^2}) \leq \mathrm{Acc}(\mathcal{C}^{0,L_M}_{\mu_g,L_g}) \leq \mathrm{Acc}(\mathcal{F}_{0,L_g L_M^2})$;
- Information on structure improves optimal step;

Function class: $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx),\ g$ is $L_g$-smooth $\mu_g$-strongly convex, $0 \leq ||M|| \leq L_M\}$

Gradient method (GM): $x_{i+1} = x_i - \frac{h}{L_g L_M^2}\nabla F(x_i)$

Parameters: $L_g = L_M = 1$, $\mu_g = 0.1$, $\kappa_g = \frac{\mu_g}{L_g}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$




Observations:

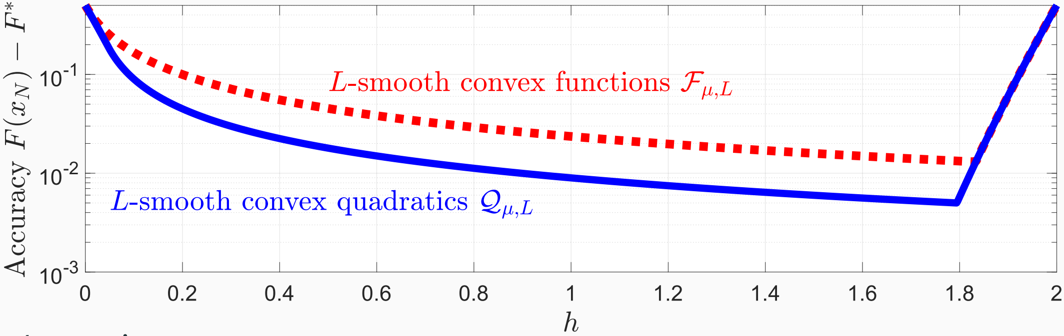
- 3 (identified) regimes;
- $\mathrm{Acc}(\mathcal{F}_{\mu_g L_M^2, L_g L_M^2}) \leq \mathrm{Acc}(\mathcal{C}^{0,L_M}_{\mu_g,L_g}) \leq \mathrm{Acc}(\mathcal{F}_{0, L_g L_M^2})$;
- Information on structure improves optimal step;
- Symmetry has no impact.

Function class: $\mathcal{Q}_{\mu,L} = \{F(x) = \frac{1}{2}x^TQx,\ \mu I \preceq Q \preceq LI,\ Q = Q^T\}(= \mathcal{D}_{1,1}^{\sqrt{\mu},\sqrt{L}})$

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

Notations/parameters: $L = 1$, $\mu = 0$, $||x_0 - x^*||^2 \leq 1$

Worst-case performance of $N = 10$ iterations (GM) for varying $h \in [0, 2]$




Observations:

- 3 (identified) regimes;
- $\mathrm{Acc}(\mathcal{Q}_{\mu,L}) \leq \mathrm{Acc}(\mathcal{F}_{\mu,L})$;
- Information on structure improves optimal step.

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$

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

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$

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

Convergence results available but implicit and technical assumptions

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$

**Chambolle-Pock (CP) method:** $\begin{cases} x_{i+1} & = \text{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} & = \text{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$

Convergence results available but implicit and technical assumptions

## Examples of assumptions:

- Existence of sets containing all iterations [Chambolle and Pock, 2011];
- Bound depending on the instance of the problem [Chambolle and Pock, 2016];
- Bound depending on the distance from initial to last point [Amir Beck, 2022].

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$

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

> Convergence results available but implicit and technical assumptions

**Examples of assumptions:**

- Existence of sets containing all iterations [Chambolle and Pock, 2011];
- Bound depending on the instance of the problem [Chambolle and Pock, 2016];
- Bound depending on the distance from initial to last point [Amir Beck, 2022].

> PEP uses classical explicit assumptions to unify performance guarantees.

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$, bounded subgradient

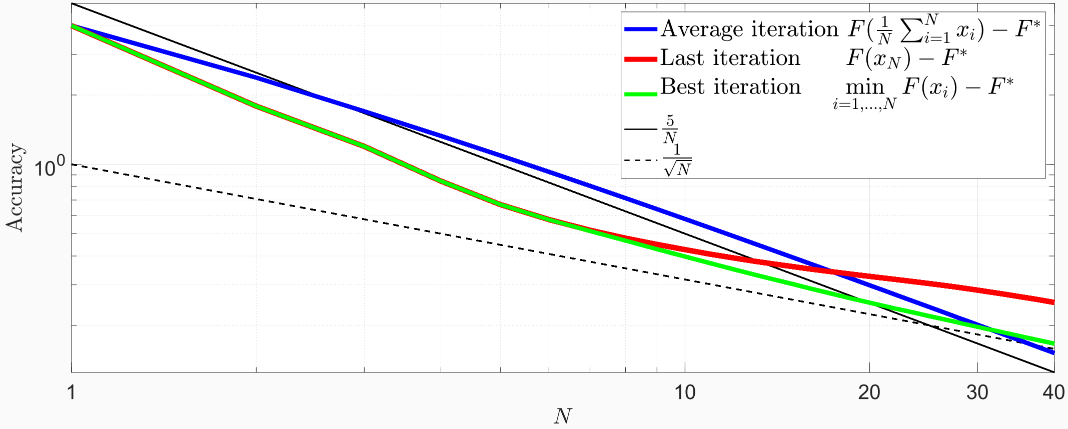
**Chambolle-Pock (CP) method:** $$\begin{cases} x_{i+1} &= \mathrm{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} &= \mathrm{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$$

**Problem:** $\min_x f(x) + g(Mx)$ where $f, g$ convex, proximable, $0 \leq ||M|| \leq L_M$, bounded subgradient

**Chambolle-Pock (CP) method:** $\begin{cases} x_{i+1} & = \text{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} & = \text{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$

**Parameters:** $L_M = \tau = \sigma = 1$, $\mu_M = 0$, $||x_0 - x^*|| \leq 1$, $||u_0 - u^*||^2 \leq 1$, $F(x) = f(x) + g(Mx)$

Worst-case performance of (CP) for varying number of iterations $N$



**State-of-the-art**

- PEP gives exact worst-case **performance of optimization methods**

**Contribution**

- Extend PEP to methods/functions involving **linear operators** (by obtaining **interpolation conditions**)
- Analyze **Gradient and Chambolle-Pock methods**

**Future steps**

- Analyze methods (e.g. PDFP, CV, PD30, ADMM) on

$$F_1(x) = f(x) + g(Mx), \qquad F_2(x) = f(x) + \frac{1}{2}x^T Q x, \qquad F_3(x) = g(Mx)$$

- Represent Hessian as linear operator to analyze **second-order methods**

Interpolation Conditions for Linear Operators and Applications to PEP

Preprint: *https://arxiv.org/abs/2302.08781*

Nizar Bousselmi (*nizar.bousselmi@uclouvain.be*)

# Worst-case performance of (GM) on $\mathcal{F}_{\mu,L}$ and $\mathcal{S}_{\mu_g,L_g}^{\mu_M,L_M}$

## Worst-case performance of (GM) on $\mathcal{F}_{\mu,L}$

$$F(x_N) - F^* \leq \frac{LR^2}{2} \max\left\{ \frac{\kappa}{\kappa - 1 + (1-\kappa h)^{-2N}}, (1-h)^{2N} \right\}$$

where $\kappa = \frac{\mu}{L}$.

## Worst-case performance of (GM) on $\mathcal{S}_{\mu_g,L_g}^{\mu_M,L_M}$

$$F(x_N) - F^* \leq \frac{L_g L_M^2 R^2}{2} \max\left\{ \frac{\kappa_g \alpha}{\kappa_g - 1 + (1-\kappa_g \alpha h)^{-2N}}, (1-h)^{2N} \right\}$$

where $\kappa_g = \frac{\mu_g}{L_g}$, $\kappa_M = \frac{\mu_M}{L_M}$ and $\alpha = \mathrm{proj}_{[\kappa_M^2,1]}\left(\frac{h_0}{h}\right)$ for $h_0$ solution of

$$\begin{cases} (1-\mu_g)(1-\mu_g h_0)^{2N+1} = 1 - (2N+1)\mu_g h_0 \\ 0 \leq h_0 \leq \frac{1}{\mu_g}. \end{cases}$$

# Worst-case functions of (GM) on $\mathcal{F}_{\mu,L}$ and $\mathcal{S}_{\mu_g,L_g}^{\mu_M,L_M}$

Let $q(x) = \frac{1}{2}x^2$ and $\ell_{\mu,h}(x) = \begin{cases} \frac{\mu}{2}x^2 + (1-\mu)\tau_{\mu,h}|x| - \left(\frac{1-\mu}{2}\right)\tau_{\mu,h}^2 & \text{if } |x| \geq \tau_{\mu,h}, \\ \frac{1}{2}x^2 & \text{else,} \end{cases}$ where $\tau_{\mu,h} = \frac{\mu}{\mu-1+(1-\mu h)^{-2N}}$.

## Worst-case functions of (GM) on $\mathcal{F}_{\mu,L}$ $(\kappa = \frac{\mu}{L})$ [Taylor, Hendrickx, Glineur, 2017]

$Lq(x) = \frac{L}{2}x^2$ and $L\ell_{\kappa,h}(x) = \begin{cases} \frac{\mu}{2}x^2 + (L-\mu)\tau_{\kappa,h}|x| - \left(\frac{L-\mu}{2}\right)\tau_{\kappa,h}^2 & \text{if } |x| \geq \tau_{\kappa,h}, \\ \frac{L}{2}x^2 & \text{else,} \end{cases}$

where $\tau_{\kappa,h} = \frac{\kappa}{\kappa-1+(1-\kappa h)^{-2N}}$.

## Worst-case functions of (GM) on $\mathcal{S}_{\mu_g,L_g}^{\mu_M,L_M}$ $(\kappa_g = \frac{\mu_g}{L_g}, \kappa_M = \frac{\mu_M}{L_M})$ [Bousselmi, Hendrickx, Glineur, 2023]

$L_g L_M q(x) = \frac{L_g}{2}(L_M x)^2$ and

$$L_g L_M M^2 \ell_{\kappa_g,M^2h}(x) = \begin{cases} \frac{\mu_g}{2}(L_M M x)^2 + (L_g - \mu_g)L_M M \tau_{\kappa_g,M^2h}|L_M M x| - \left(\frac{L_g-\mu_g}{2}\right) L_M M \tau_{\kappa_g,M^2h}^2 & \text{if } |x| \geq \tau_{\kappa_g,M^2h}, \\ \frac{L_g}{2}(L_M M x)^2 & \text{else,} \end{cases}$$

where $\tau_{\kappa_g,M^2h} = \frac{\kappa_g}{\kappa_g-1+\left(1-\kappa_g M^2 h\right)^{-2N}}$ and $M = \mathrm{Proj}_{[\kappa_M,1]}\left(\sqrt{\frac{h_0}{h}}\right)$.

- $\ell_p$-regularized robust regression ($p = 1, 2$)

$$\min_x ||Mx - b||_1 + ||x||_p^p,$$

- $\ell_1$-constrained least squares

$$\min_x \ ||Mx - b||_2^2,$$
$$||x||_1 \leq c,$$

- Basis pursuit

$$\min_x \ ||x||_1,$$
$$Mx = c,$$

- Total variation deblurring

$$\min_x \ ||M_1 x - b||_2^2 + ||M_2 x||_1,$$

- Resource allocation

$$\min_x \ F(x),$$
$$Mx = c.$$

# Methods with linear operators ($\min_x f(x) + g(Mx) + h(x)$, $g, h$ proximable and $f$ smooth)

- **Primal-Dual Fixed Point (PDFP)** [Chen et al., 2016]

$$\begin{cases} \tilde{x} = \mathrm{prox}_{\tau h}\left(x_i - \tau \nabla f(x_i) - \tau M^T u_i\right), \\ u = \mathrm{prox}_{\sigma g^*}\left(u_i + \sigma M \tilde{x}\right), \\ x = \mathrm{prox}_{\tau h}\left(x_i - \tau \nabla f(x_i) - \tau M^T u\right), \\ x_{i+1} = x_i + \rho_i (x - x_i), \\ u_{i+1} = u_i + \rho_i (u - u_i). \end{cases}$$

- **Condat-Vu (CV)** [Condat, 2013, Vu, 2013]

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

- **Primal-Dual Three-Operator Splitting (PD3O)**

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

- **Alternating Direction Method of Multipliers (ADMM)** [Gabay and Mercier, 1976]:

$$\begin{cases} x_{i+1} \in \arg\min_x f(x) + \frac{\rho}{2} M_1 x + M_2 y_i - c + \frac{1}{\rho} z_i^2, \\ y_{i+1} \in \arg\min_y g(y) + \frac{\rho}{2} M_1 x_{i+1} + M_2 y - c + \frac{1}{\rho} z_i^2, \\ z_{i+1} = z_i + \rho (M_1 x_{i+1} + M_2 y_{i+1} - c), \end{cases}$$

solves $\min_x f(x) + g(y)$ s.t. $M_1 x + M_2 y = c$.

- **Proximal Alternating Predictor-Corrector (PAPC)** [Drori et al., 2015]

$$\begin{cases} p_{i+1} = x_i - \tau\left(M y_i + \nabla f(x_i)\right), \\ y_{i+1} = \mathrm{prox}_h\left(y_i + M^T p_{i+1}\right), \\ x_{i+1} = x_i - \tau\left(M y_{i+1} + \nabla f(x_i)\right), \end{cases}$$

solves $\min_x \max_y f(x) + x^T M y - h(y)$.

# Proof of necessity

**Theorem ($L$-matrix-interpolation conditions)**

*$\{(x_i, y_i)\}_{i=1,\dots,N_1}$ and $\{(u_j, v_j)\}_{j=1,\dots,N_2}$ are **L-matrix-interpolable** if, and only if,*

$$\begin{cases} X^TV = Y^TU, \\ Y^TY \preceq L^2X^TX, \\ V^TV \preceq L^2U^TU, \end{cases}$$

*where $X = (x_1 \ \cdots \ x_{N_1})$, $Y = (y_1 \ \cdots \ y_{N_1})$, $U = (u_1 \ \cdots \ u_{N_2})$ and $V = (v_1 \ \cdots \ v_{N_2})$.*

Proof.
**Necessity**: Let $\{(x_i, y_i)\}_{i=1,\dots,N_1}$ and $\{(u_j, v_j)\}_{j=1,\dots,N_2}$ $L$-matrix-interpolable.

$$\Leftrightarrow \exists M \text{ with } \sigma_{\max}(M) \leq L \ : \qquad y_i = Mx_i \text{ and } v_j = M^Tu_j \ \ \forall i, j$$

$$\Leftrightarrow \exists M \text{ with } M^TM \preceq L^2I \text{ and } MM^T \preceq L^2I \ : \ Y = MX \text{ and } V = M^TU$$

$$\Rightarrow \underbrace{X^TM^TU}_{X^TV} = \underbrace{X^TM^TU}_{Y^TU}, \qquad \underbrace{X^TM^TMX}_{Y^TY} \preceq L^2X^TX, \qquad \underbrace{U^TMM^TU}_{V^TV} \preceq L^2U^TU$$

□

## Proof of sufficiency in short

Let $(X, Y, U, V)$ satisfying $\begin{cases} X^T V &= Y^T U, \\ Y^T Y &\preceq L^2 X^T X, \\ V^T V &\preceq L^2 U^T U, \end{cases}$ then,

- **Step 1:** $\exists (X_R, Y_R, U_R, V_R)$ building the same Gram matrices, i.e.

$$(X\ V)^T (X\ V) = (X_R\ V_R)^T (X_R\ V_R),$$
$$(Y\ U)^T (Y\ U) = (Y_R\ U_R)^T (Y_R\ U_R),$$

and such that $\exists M_R$ with $\sigma_{\max}(M_R) \leq L$ : $\begin{cases} Y_R = M_R X_R, \\ V_R = M_R^T U_R. \end{cases}$

- **Step 2:** If $(X, Y, U, V)$ and $(X_R, Y_R, U_R, V_R)$ build the same Gram matrices, then,

$\exists R_1, R_2$ unitary : $\begin{cases} (X_R\ V_R) = R_1 (X\ V), \\ (Y_R\ U_R) = R_2 (Y\ U), \end{cases}$ therefore, $\begin{cases} Y = \overbrace{R_2^T M_R R_1}^{M} X, \\ V = \overbrace{R_1^T M_R^T R_2}^{M^T} U. \end{cases}$