PERFORMANCE ESTIMATION OF FIRST-ORDER METHODS INVOLVING LINEAR MAPPINGS

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

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 1:

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** $\mathcal{M}$ on

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

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

Example 1:

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

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

Example 2:

Worst-case performance of methods involving linear mappings?

e.g. Chambolle-Pock method, Condat-Vũ method, PDFP, PD3O, PAPC, ADMM, etc.

Performance Estimation Problem (PEP)

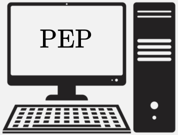
Theoretical and practical framework to analyze **performance** of optimization methods on problem classes.

- *Performance of first-order methods for smooth convex minimization: a novel approach* [Drori and Teboulle, 2014]
- *Convex interpolation and performance estimation of first-order methods for convex optimization* [Taylor, 2017]

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

- *Performance of first-order methods for smooth convex minimization: a novel approach* [Drori and Teboulle, 2014]
- *Convex interpolation and performance estimation of first-order methods for convex optimization* [Taylor, 2017]

Method
+
Class of functions



Exact worst-case performance

Performance Estimation Problem (PEP)

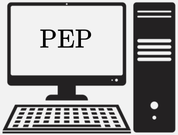
Theoretical and practical framework to analyze **performance** of optimization methods on problem classes.

- *Performance of first-order methods for smooth convex minimization: a novel approach* [Drori and Teboulle, 2014]
- *Convex interpolation and performance estimation of first-order methods for convex optimization* [Taylor, 2017]

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]*
- 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]*
- **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]*
- 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

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

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

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

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

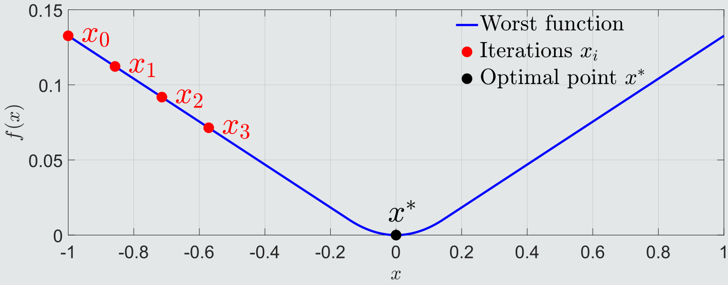
Input = optimization method + problem class
Output = worst instance in problem class

Example:

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

PEP solved outputs

- Worst performance: $\boldsymbol{f(x_N) - f^*} \leq \frac{L}{2}\frac{1}{2N+1}$ (for all N and L);
- Worst function: (for $N = 3$ and $L = 1$)



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

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

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

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

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

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

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

PEP

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

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

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

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

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

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

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

PEP

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

Interpolation condition to reformulate.

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

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

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

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

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

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

Interpolation condition to reformulate.

Can be reformulated as convex **semidefinite problem**, efficiently solvable!

Interpolation conditions for L-smooth convex functions

Given x_i, g_i, f_i $\quad \forall i = 0, \ldots, N$,

$\exists$ L-smooth convex f such that $\begin{cases} f(x_i) & = f_i \;\; \forall i = 0, \ldots, N, \\ \nabla f(x_i) & = g_i \;\; \forall i = 0, \ldots, N, \end{cases}$ if, and only if,

$$f_i \geq f_j + g_j^T (x_i - x_j) + \frac{1}{2L} ||g_i - g_j||^2 \quad \forall i, j = 0, \ldots, N.$$

Interpolation conditions for L-smooth convex functions

Given $x_i, g_i, f_i \quad \forall i = 0, \dots, N$,

$\exists$ L-smooth convex f such that $\begin{cases} f(x_i) & = f_i \ \ \forall i = 0, \dots, N, \\ \nabla f(x_i) & = g_i \ \ \forall i = 0, \dots, N, \end{cases}$ if, and only if,

$$\underbrace{f_i \geq f_j + g_j^T(x_i - x_j) + \frac{1}{2L}||g_i - g_j||^2}_{\text{Linear in } f_i,\ x_i^T g_j,\ x_i^T x_j,\ g_i^T g_j} \quad \forall i, j = 0, \dots, N.$$

Interpolation conditions must be convex in $f_i, \quad x_i^T g_j, \quad x_i^T x_j, \quad g_i^T g_j$.

Remark: Interpolation conditions and PEP formulation available for numerous function classes: **non-smooth, L-smooth, convex, μ-strongly convex, etc.**

Performance Estimation Problem for linear mappings

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

$F(x)$	Possible method	Iterations
$g(Mx)$	Gradient descent	$x_{i+1} = x_i - \frac{h}{L} M^T \nabla g(Mx_i)$
$\frac{1}{2} x^T Q x \ (= \frac{1}{2} \lVert Q^{\frac{1}{2}} x \rVert^2)$	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 mappings.

$F(x)$	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 mappings.

$F(x)$	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?

Interpolation conditions for linear mappings!

Decomposing the iteration

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

Decomposing the iteration

- 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

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

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

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

Also requires New interpolation conditions.

Definition (L-matrix-interpolability)

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

$$\exists M \text{ with } \sigma_{\max}(M) \leq L \ : \ \begin{cases} y_i = M x_i, & \forall i = 1, \dots, N_1, \\ v_j = M^T u_j, & \forall j = 1, \dots, N_2. \end{cases}$$

Interpolation conditions as semidefinite constraints

Definition (L-matrix-interpolability)

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

$$\exists M \text{ with } \sigma_{\max}(M) \leq L \ : \ \begin{cases} y_i = Mx_i, & \forall i = 1, \dots, N_1, \\ v_j = M^T u_j, & \forall j = 1, \dots, N_2. \end{cases}$$

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,*

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

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

Remark: [Colla and Hendrickx, 2022] and [Abbaszadehpeivasti et al., 2022] used these conditions knowing their necessity.

Theorem (L-matrix-interpolation conditions)

*$\{(x_i, y_i)\}_{i=1,\ldots,N_1}$ and $\{(u_j, v_j)\}_{j=1,\ldots,N_2}$ are **L-matrix-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 $X = (x_1 \ \cdots \ x_{N_1})$, $Y = (y_1 \ \cdots \ y_{N_1})$, $U = (u_1 \ \cdots \ u_{N_2})$ and $V = (v_1 \ \cdots \ v_{N_2})$.

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

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,*

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

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

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

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,*

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

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

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

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

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,*

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

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

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

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

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.

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

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

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

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,*

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

where $X = (x_1 \; \cdots \; x_{N_1})$, $Y = (y_1 \; \cdots \; y_{N_1})$, $U = (u_1 \; \cdots \; u_{N_2})$ 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.

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

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

$$\Rightarrow \underbrace{X^T M^T U}_{X^T V} = \underbrace{X^T M^T U}_{Y^T U}, \qquad \underbrace{X^T M^T M X}_{Y^T Y} \preceq L^2 X^T X, \qquad \underbrace{U^T M M^T U}_{V^T V} \preceq L^2 U^T U$$

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,*

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

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

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

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

$$\Rightarrow \underbrace{X^T M^T U}_{X^T V} = \underbrace{X^T M^T U}_{Y^T U}, \qquad \underbrace{X^T M^T M X}_{Y^T Y} \preceq L^2 X^T X, \qquad \underbrace{U^T M M^T U}_{V^T V} \preceq L^2 U^T U$$

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,*

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

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

$$\begin{aligned} &\Leftrightarrow \exists M \text{ with } \sigma_{\max}(M) \le L \ : \quad y_i = Mx_i \text{ and } v_j = M^T u_j \ \ \forall i, j \\ &\Leftrightarrow \exists M \text{ with } M^T M \preceq L^2 I \text{ and } MM^T \preceq L^2 I \ : \ Y = MX \text{ and } V = M^T U \\ &\Rightarrow \underbrace{X^T M^T U}_{X^T V} = \underbrace{X^T M^T U}_{Y^T U}, \quad \underbrace{X^T M^T M X}_{Y^T Y} \preceq L^2 X^T X, \quad \underbrace{U^T M M^T U}_{V^T V} \preceq L^2 U^T U \end{aligned}$$

Sufficiency: see [Bousselmi, Hendrickx, Glineur, 2023]. □

Definition ($(0, L)$-symmetric-matrix-interpolability)

$\{(x_i, y_i)\}_{i=1,\dots,N}$ is **$(0, L)$-symmetric-matrix-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } 0 \preceq Q \preceq LI \ : \ y_i = Qx_i \quad \forall i = 1, \dots, N.$$

Theorem ($(0, L)$-symmetric-matrix-interpolation conditions)

*$\{(x_i, y_i)\}_{i=1,\dots,N}$ is **$(0, L)$-symmetric-matrix-interpolable** if, and only if,*

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

where $X = (x_1 \ \cdots \ x_{N_1})$ and $Y = (y_1 \ \cdots \ y_{N_1})$.

Remark: Similar result for skew-symmetric matrices.

Definition ((μ, L)-symmetric-matrix-interpolability)

$\{(x_i, y_i)\}_{i=1,\ldots,N}$ is **(μ, L)-symmetric-matrix-interpolable** if, and only if,

$$\exists Q \text{ symmetric with } \mu I \preceq Q \preceq LI \;:\; y_i = Qx_i \quad \forall i = 1, \ldots, N.$$

Theorem ((μ, L)-symmetric-matrix-interpolation conditions)

*$\{(x_i, y_i)\}_{i=1,\ldots,N}$ is **(μ, L)-symmetric-matrix-interpolable** if, and only if,*

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

where $X = (x_1 \;\cdots\; x_{N_1})$ and $Y = (y_1 \;\cdots\; y_{N_1})$.

Remark: Similar result for skew-symmetric matrices.

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

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

$\{(x_i, g_i, f_i)\}_{i=1,\ldots,N}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

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

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

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

$\{(x_i, g_i, f_i)\}_{i=1,\ldots,N}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

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

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

*$\{(x_i, g_i, f_i)\}_{i=1,\ldots,N}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,*

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

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

Homogeneous quadratic functions: $\mathcal{Q}_{\mu,L} = \{f(x) = \frac{1}{2}x^T Q x, Q = Q^T, \mu I \preceq Q \preceq LI\}$

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

$\{(x_i, g_i, f_i)\}_{i=1,\dots,N}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,

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

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

*$\{(x_i, g_i, f_i)\}_{i=1,\dots,N}$ is **$\mathcal{Q}_{\mu,L}$-interpolable** if, and only if,*

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

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 matrix-interpolability

1 Gradient method on $g(Mx)$

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

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

Simplest problem $\min_x F(x)$ involving linear mapping

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

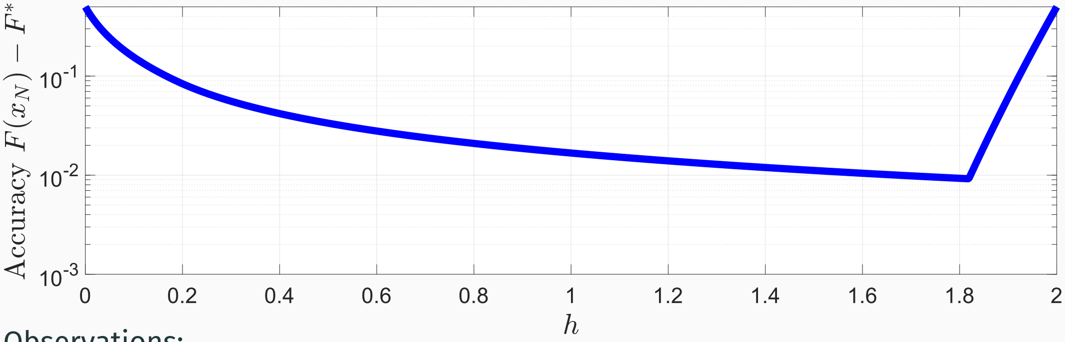
Gradient method (GM): $x_{i+1} = x_i - \frac{h}{L_g L_M^2} \nabla F(x_i)$

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 $h \in [0, 2]$



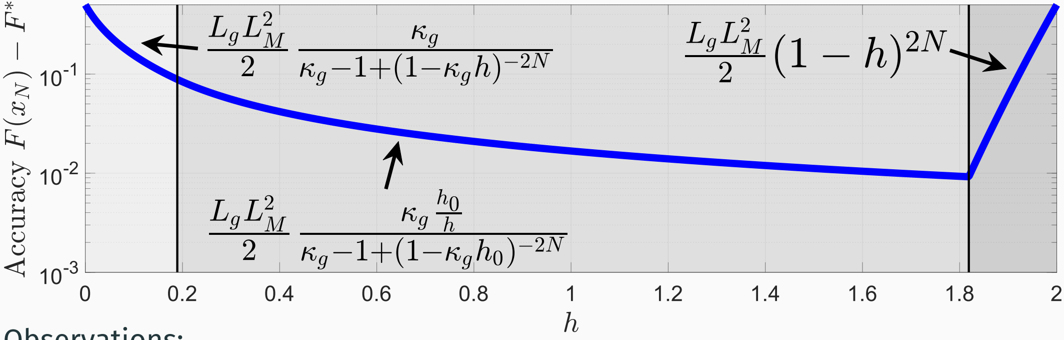
Observations:

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

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

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

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



Observations:

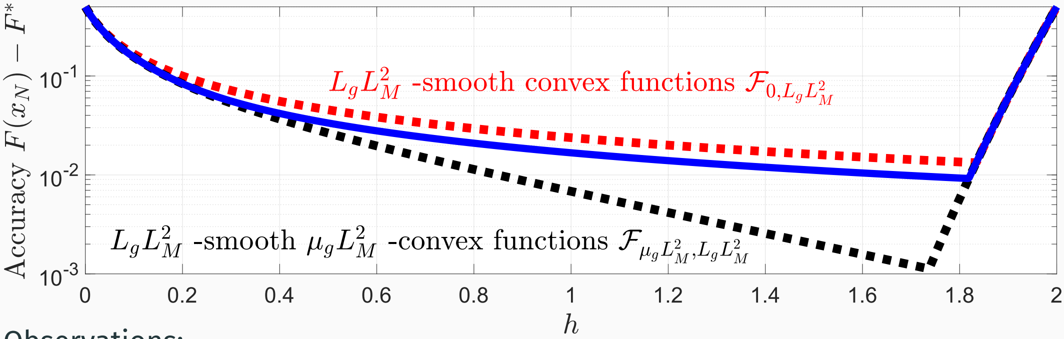
- 3 (identified) regimes;

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:

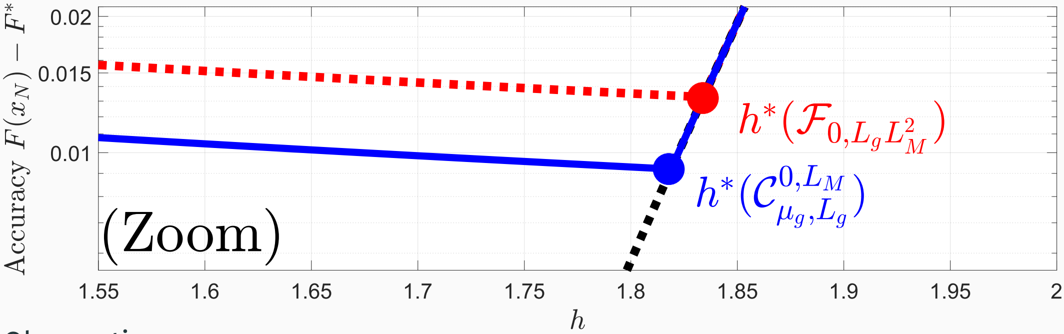
- 3 (identified) regimes;
- $\mathrm{Acc}(\mathcal{F}_{\mu_g L_M^2, L_g L_M^2}) \leq \mathrm{Acc}(\mathcal{C}^{0,L_M}_{\mu_g,L_g}) \leq \mathrm{Acc}(\mathcal{F}_{0,L_g L_M^2})$;

Function class: $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx), g$ is L_g-smooth μ_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;

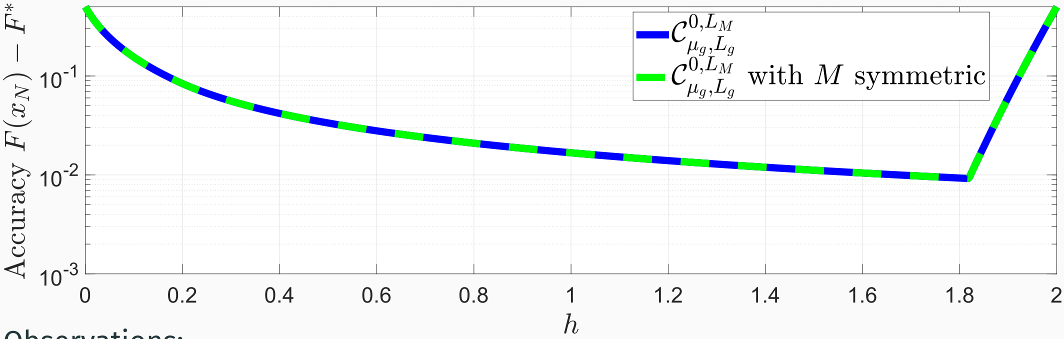
SIMPLEST PROBLEM $\min_x F(x)$ INVOLVING LINEAR MAPPING

Function class: $\mathcal{C}^{0,L_M}_{\mu_g,L_g} = \{F(x) = g(Mx)$, g is L_g-smooth μ_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.

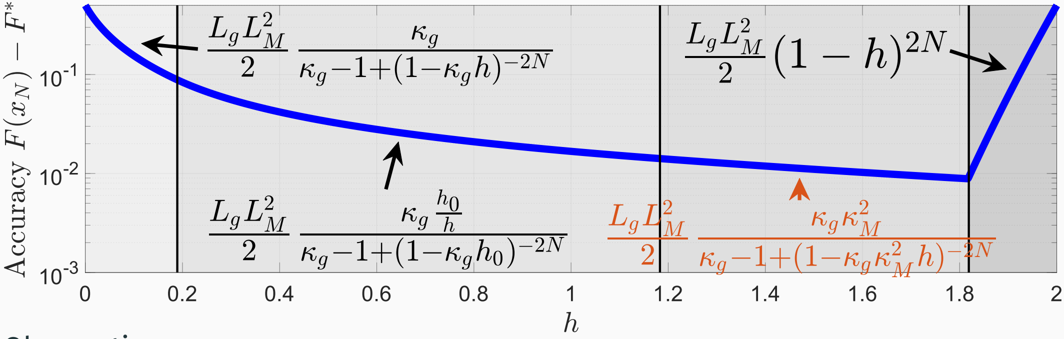
SIMPLEST PROBLEM $\min_x F(x)$ INVOLVING LINEAR MAPPING ($\mu_M > 0$)

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

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

Parameters: $L_g = L_M = 1$, $\mu_g = 0.1$, $\mu_M = 0.4$, $\kappa_g = \frac{\mu_g}{L_g}$, $\kappa_M = \frac{\mu_M}{L_M}$, $||x_0 - x^*||^2 \leq 1$, $h_0(\kappa_g, N)$

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



Observations:

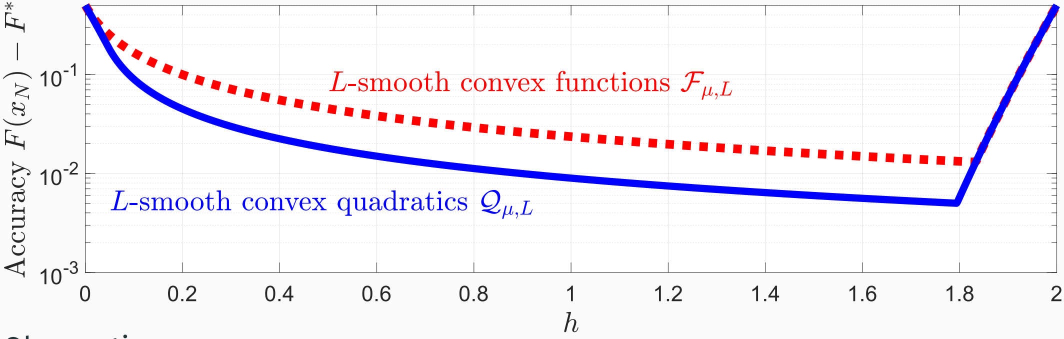
- 4 (identified) regimes;
- $\mathrm{Acc}(\mathcal{F}_{\mu_g L_M^2, L_g L_M^2}) \leq \mathrm{Acc}(\mathcal{S}^{\mu_M, L_M}_{\mu_g, L_g}) \leq \mathrm{Acc}(\mathcal{F}_{\mu_g \mu_M^2, L_g L_M^2})$;
- Information on structure improves optimal step.

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

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.

Analyzing a sophisticated algorithm

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

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

Analyzing a sophisticated algorithm

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

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

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

Convergence results available but implicit and technical assumptions

Examples of assumptions:

- Existence of sets B_1 and B_2 sufficiently large to contain all iterations x_i and u_i [Chambolle and Pock, 2011];
- Bound depending on the actual 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 B_1 and B_2 sufficiently large to contain all iterations x_i and u_i [Chambolle and Pock, 2011];
- Bound depending on the actual 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.

Explicit assumption and performance criterion of choice

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

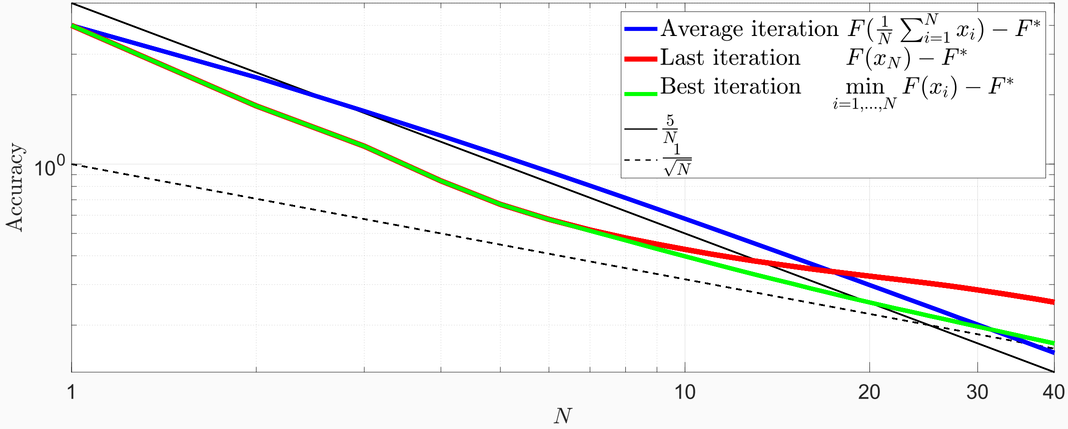
Chambolle-Pock (CP) method: $\begin{cases} x_{i+1} &= \text{prox}_{\tau f(.)}\left(x_i - \tau M^T u_i\right) \\ u_{i+1} &= \text{prox}_{\sigma g^*(.)}\left(u_i + \sigma M(2x_{i+1} - x_i)\right) \end{cases}$

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

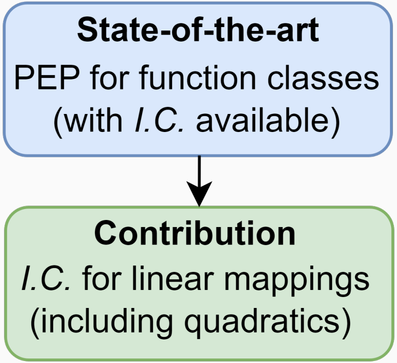
Worst-case performance of (CP) for varying number of iterations N



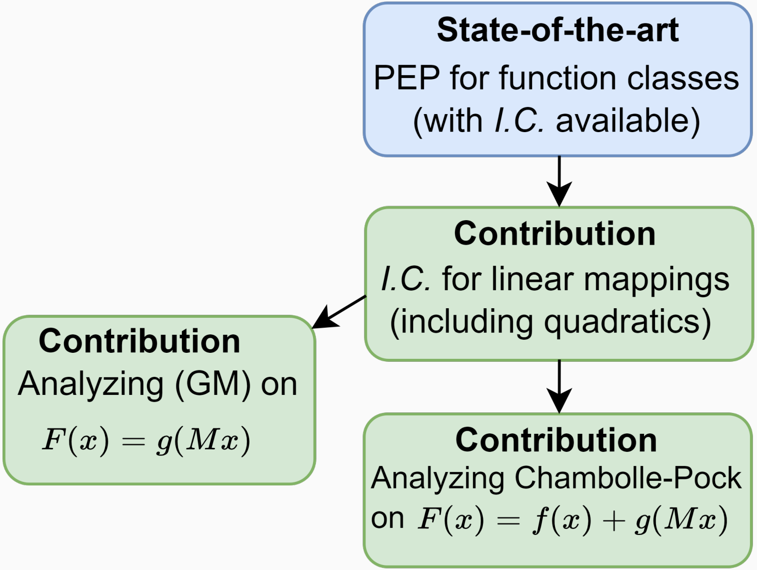
State-of-the-art

PEP for function classes
(with *I.C.* available)

I.C.: interpolation conditions

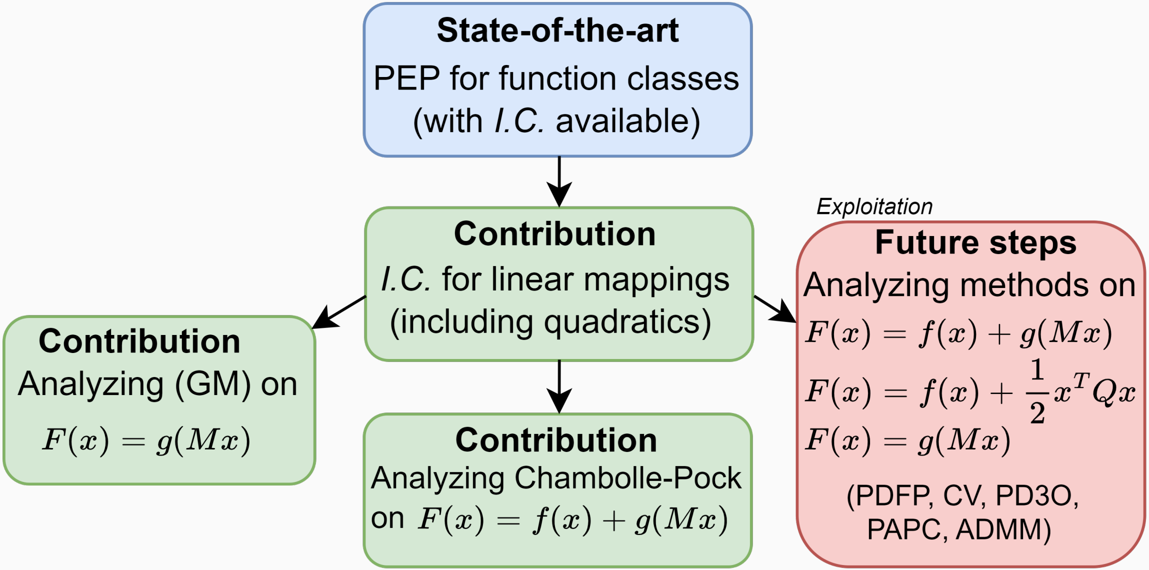


I.C.: interpolation conditions

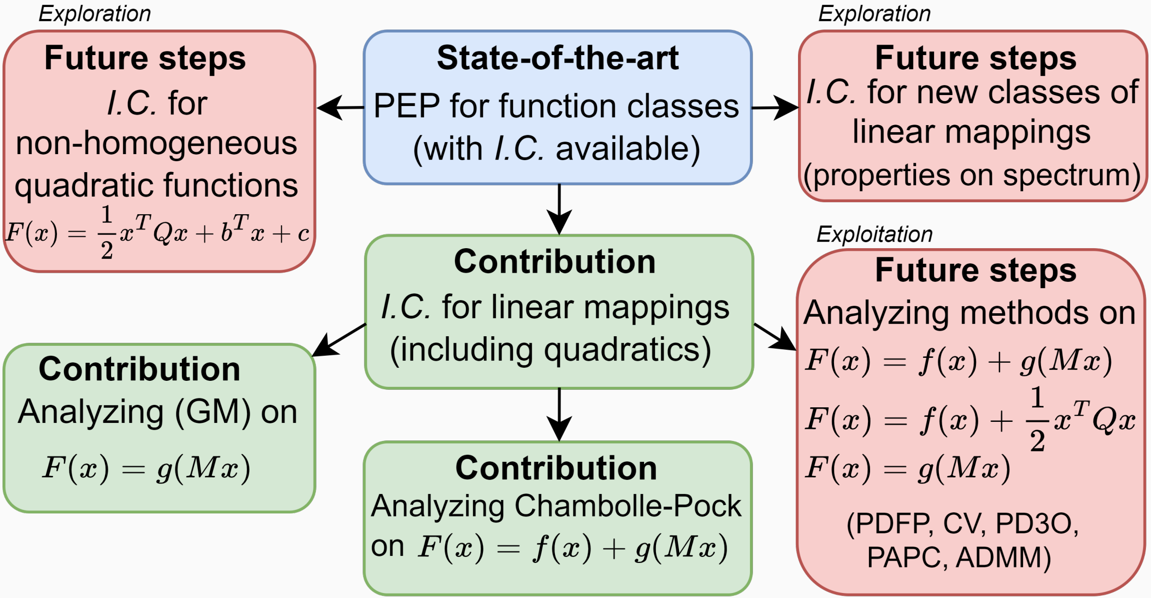


I.C.: interpolation conditions

PEP to analyze functions involving linear mapping



I.C.: interpolation conditions



I.C.: interpolation conditions

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 $M = \mathrm{Proj}_{[\kappa_M,1]}\left(\sqrt{\frac{h_0}{h}}\right)$.

Practical problems involving linear mappings

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

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

- ℓ_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

$$\min_x \; ||M_1x - b||_2^2 + ||M_2x||_1,$$

- Resource allocation

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

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

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

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

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

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

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

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

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

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

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]

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

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

Proof of sufficiency in short

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

- **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 $\sigma_{\mathbf{max}}(M_R) \leq L$: $\begin{cases} Y_R = M_R X_R, \\ V_R = M_R^T U_R. \end{cases}$

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

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