Automated tight Lyapunov analysis for first-order methods

Manu Upadhyaya
manu.upadhyaya@control.lth.se

Lund University
Department of Automatic Control

Collaborators:

Sebastian Banert

Adrien Taylor

Pontus Giselsson

21st conference on advances in continuous optimization (EUROPT 2024)

This talk

- Based on:
  - U.M., Banert, S., Taylor, A.B., Giselsson, P. *Automated tight Lyapunov analysis for first-order methods*. Mathematical Programming (2024)
    - Code: `https://github.com/ManuUpadhyaya/TightLyapunovAnalysis`
  - Banert, S., U.M., Giselsson, P. *The Chambolle–Pock method converges weakly with* $\theta > 1/2$ *and* $\tau\sigma\|L\|^2 < 4/(1+2\theta)$ [arXiv:2309.03998] (2023)
- Content:
  - Methodology for proving algorithm convergence
  - Focus on first-order (splitting) methods for convex optimization problems

This talk

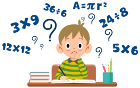
- Based on:
  - U.M., Banert, S., Taylor, A.B., Giselsson, P. *Automated tight Lyapunov analysis for first-order methods*. Mathematical Programming (2024)
    - Code: `https://github.com/ManuUpadhyaya/TightLyapunovAnalysis`
  - Banert, S., U.M., Giselsson, P. *The Chambolle–Pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$* [arXiv:2309.03998] (2023)
- Content:
  - Methodology for proving algorithm convergence
  - Focus on first-order (splitting) methods for convex optimization problems

This talk

- Based on:
  - U.M., Banert, S., Taylor, A.B., Giselsson, P. *Automated tight Lyapunov analysis for first-order methods*. Mathematical Programming (2024)
    - Code: `https://github.com/ManuUpadhyaya/TightLyapunovAnalysis`
  - Banert, S., U.M., Giselsson, P. *The Chambolle–Pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$* [arXiv:2309.03998] (2023)
- Content:
  - Methodology for proving algorithm convergence
  - Focus on first-order (splitting) methods for convex optimization problems

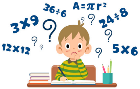
Proving convergence

- Pages of inequalities



Proving convergence

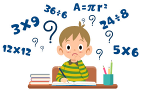
- Pages of inequalities



- However, proofs look very similar

Proving convergence

- Pages of inequalities

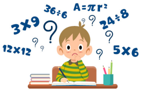


- However, proofs look very similar

- Automate!

Proving convergence

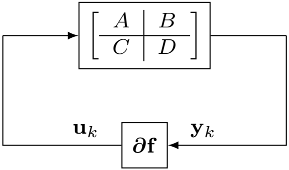
- Pages of inequalities



- However, proofs look very similar

- Automate!

- Our approach



and $\qquad V_{k+1} \leq \rho V_k - R_k$

One example of what we can show with our methodology

- Problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f(y) + g(y)$$

  where $f, g \in \mathcal{F}_{0,\infty}$, i.e. lower semicontinuous, proper and convex.

- Method (Chambolle and Pock, 2011, Algorithm 1):

$$x_{k+1} = \text{prox}_{\tau f}(x_k - \tau y_k),$$
$$y_{k+1} = \text{prox}_{\sigma g^*}(y_k + \sigma(x_{k+1} + \theta(x_{k+1} - x_k)))$$

  where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$, prox is the proximal operator and g^* is the convex conjugate of g (linear operator set to identity mapping)

- Parameter choices that give ($\mathcal{O}(1/k)$ ergodic primal-dual gap) convergence:

One example of what we can show with our methodology

- Problem:

$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f(y)+g(y)$$

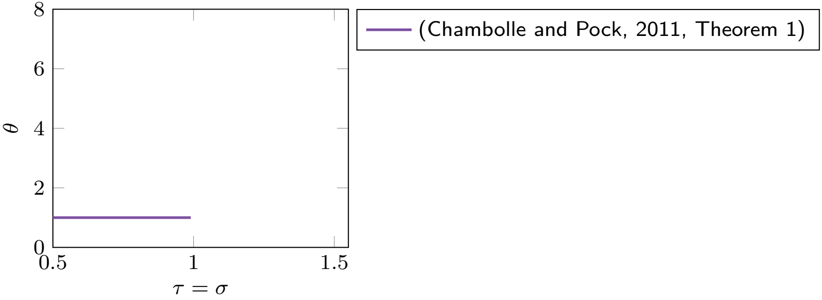
  where $f, g \in \mathcal{F}_{0,\infty}$, i.e. lower semicontinuous, proper and convex.
- Method (Chambolle and Pock, 2011, Algorithm 1):

$$x_{k+1} = \operatorname{prox}_{\tau f}(x_k - \tau y_k),$$
$$y_{k+1} = \operatorname{prox}_{\sigma g^*}(y_k + \sigma(x_{k+1} + \theta(x_{k+1} - x_k)))$$

  where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$, prox is the proximal operator and g^* is the convex conjugate of g (linear operator set to identity mapping)
- Parameter choices that give ($\mathcal{O}(1/k)$ ergodic primal-dual gap) convergence:

One example of what we can show with our methodology

- Problem:
$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f(y)+g(y)$$
where $f, g \in \mathcal{F}_{0,\infty}$, i.e. lower semicontinuous, proper and convex.
- Method (Chambolle and Pock, 2011, Algorithm 1):
$$x_{k+1} = \text{prox}_{\tau f}(x_k - \tau y_k),$$
$$y_{k+1} = \text{prox}_{\sigma g^*}(y_k + \sigma\,(x_{k+1} + \theta(x_{k+1} - x_k)))$$
where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$, prox is the proximal operator and g^* is the convex conjugate of g (linear operator set to identity mapping)
- Parameter choices that give ($\mathcal{O}(1/k)$ ergodic primal-dual gap) convergence:



One example of what we can show with our methodology

- Problem:

$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f(y)+g(y)$$

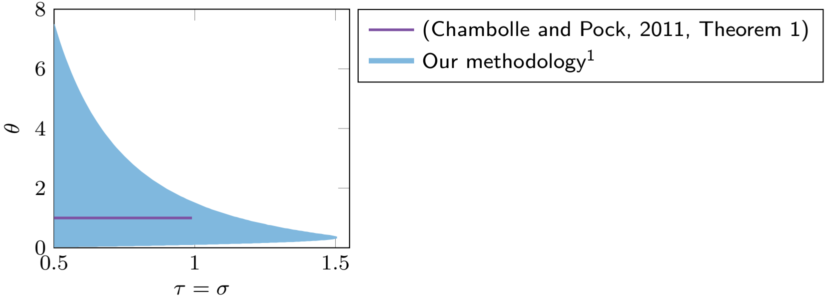
  where $f, g \in \mathcal{F}_{0,\infty}$, i.e. lower semicontinuous, proper and convex.

- Method (Chambolle and Pock, 2011, Algorithm 1):

$$x_{k+1} = \text{prox}_{\tau f}(x_k - \tau y_k),$$
$$y_{k+1} = \text{prox}_{\sigma g^*}(y_k + \sigma(x_{k+1} + \theta(x_{k+1} - x_k)))$$

  where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$, prox is the proximal operator and g^* is the convex conjugate of g (linear operator set to identity mapping)

- Parameter choices that give ($\mathcal{O}(1/k)$ ergodic primal-dual gap) convergence:



[1]Parameters evaluated on a square grid of size 0.01×0.01 with the restriction that $\tau = \sigma \geq 0.5$

An extended convergence analysis of the Chambolle–Pock method

- The numerical result above was the inspiration for the analytical result in (Banert et al., 2023)
- Problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f(y) + g(Ly)$$

where $f \in \mathcal{F}_{0,\infty}(\mathcal{H})$, $g \in \mathcal{F}_{0,\infty}(\mathcal{G})$ and L is a continuous linear operator from $\mathcal{H}$ to $\mathcal{G}$

- Method (Chambolle and Pock, 2011, Algorithm 1):

$$x_{k+1} = \text{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \text{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k)))$$

where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$

An extended convergence analysis of the Chambolle–Pock method

- The numerical result above was the inspiration for the analytical result in (Banert et al., 2023)
- Problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f(y) + g(Ly)$$

  where $f \in \mathcal{F}_{0,\infty}(\mathcal{H})$, $g \in \mathcal{F}_{0,\infty}(\mathcal{G})$ and L is a continuous linear operator from $\mathcal{H}$ to $\mathcal{G}$
- Method (Chambolle and Pock, 2011, Algorithm 1):

$$\begin{aligned} x_{k+1} &= \operatorname{prox}_{\tau f}(x_k - \tau L^* y_k), \\ y_{k+1} &= \operatorname{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k))) \end{aligned}$$

  where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$

An extended convergence analysis of the Chambolle–Pock method

- The numerical result above was the inspiration for the analytical result in (Banert et al., 2023)
- Problem:
$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f(y) + g(Ly)$$
where $f \in \mathcal{F}_{0,\infty}(\mathcal{H})$, $g \in \mathcal{F}_{0,\infty}(\mathcal{G})$ and L is a continuous linear operator from $\mathcal{H}$ to $\mathcal{G}$
- Method (Chambolle and Pock, 2011, Algorithm 1):
$$\begin{aligned} x_{k+1} &= \mathrm{prox}_{\tau f}(x_k - \tau L^* y_k), \\ y_{k+1} &= \mathrm{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k))) \end{aligned}$$
where $\tau, \sigma > 0$, $\theta \in \mathbb{R}$

An extended convergence analysis of the Chambolle–Pock method

- The problem has the equivalent primal–dual formulation

$$\underset{x\in\mathcal{H}}{\text{minimize}}\ \underset{y\in\mathcal{G}}{\text{maximize}}\ \ f(x) + \langle Lx, y\rangle - g^*(y)$$

- We assume the problem has at least one solution $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ that satisfies the Karush–Kuhn–Tucker (KKT) condition

$$\begin{aligned} -L^* y_\star &\in \partial f(x_\star), \\ Lx_\star &\in \partial g^*(y_\star) \end{aligned}$$

- We define the *primal–dual gap function* (our convergence measure)

$$\mathcal{D}_{x_\star,y_\star}(x, y) = \mathcal{L}(x, y_\star) - \mathcal{L}(x_\star, y) \geq 0$$

where

$$\mathcal{L}(x, y) = f(x) + \langle Lx, y\rangle - g^*(y)$$

is the *Lagrangian function*

An extended convergence analysis of the Chambolle–Pock method

- The problem has the equivalent primal–dual formulation

$$\underset{x \in \mathcal{H}}{\text{minimize}} \, \underset{y \in \mathcal{G}}{\text{maximize}} \;\; f(x) + \langle Lx, y \rangle - g^*(y)$$

- We assume the problem has at least one solution $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ that satisfies the Karush–Kuhn–Tucker (KKT) condition

$$\begin{aligned} -L^* y_\star &\in \partial f(x_\star), \\ L x_\star &\in \partial g^*(y_\star) \end{aligned}$$

- We define the *primal–dual gap function* (our convergence measure)

$$\mathcal{D}_{x_\star, y_\star}(x, y) = \mathcal{L}(x, y_\star) - \mathcal{L}(x_\star, y) \geq 0$$

where

$$\mathcal{L}(x, y) = f(x) + \langle Lx, y \rangle - g^*(y)$$

is the *Lagrangian function*

An extended convergence analysis of the Chambolle–Pock method

- The problem has the equivalent primal–dual formulation

$$\underset{x \in \mathcal{H}}{\text{minimize}} \, \underset{y \in \mathcal{G}}{\text{maximize}} \;\; f(x) + \langle Lx, y \rangle - g^*(y)$$

- We assume the problem has at least one solution $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ that satisfies the Karush–Kuhn–Tucker (KKT) condition

$$\begin{aligned} -L^* y_\star &\in \partial f(x_\star), \\ L x_\star &\in \partial g^*(y_\star) \end{aligned}$$

- We define the *primal–dual gap function* (our convergence measure)

$$\mathcal{D}_{x_\star, y_\star}(x, y) = \mathcal{L}(x, y_\star) - \mathcal{L}(x_\star, y) \geq 0$$

  where

$$\mathcal{L}(x, y) = f(x) + \langle Lx, y \rangle - g^*(y)$$

  is the *Lagrangian function*

An extended convergence analysis of the Chambolle–Pock method

Chambolle–Pock method

$$x_{k+1} = \mathrm{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \mathrm{prox}_{\sigma g^*}(y_k + \sigma L\,(x_{k+1} + \theta(x_{k+1} - x_k)))$$

- **Result 1 [Ergodic convergence rate]:** Suppose that

$$\theta \geq 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \leq 4/(1+2\theta)$$

where at least one is strict. Then

$$\mathcal{D}_{x_\star, y_\star}\left(\frac{1}{K}\sum_{k=1}^{K} x_k, \frac{1}{K}\sum_{k=1}^{K} y_k\right) = \mathcal{O}\left(\frac{1}{K}\right)$$

- **Result 2 [Weak convergence]:** Suppose that

$$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 < 4/(1+2\theta).$$

Then

$$(x_k, y_k) \rightharpoonup (x_\star, y_\star)$$

- **Result 3 [Tightness]:** Suppose that

$$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \geq 4/(1+2\theta).$$

Then there exists an example such that the Chambolle–Pock method does not converge

An extended convergence analysis of the Chambolle–Pock method

Chambolle–Pock method

$$x_{k+1} = \operatorname{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \operatorname{prox}_{\sigma g^*}(y_k + \sigma L\,(x_{k+1} + \theta(x_{k+1} - x_k)))$$

- **Result 1 [Ergodic convergence rate]:** Suppose that

$$\theta \geq 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \leq 4/(1+2\theta)$$

where at least one is strict. Then

$$\mathcal{D}_{x_\star, y_\star}\left(\frac{1}{K}\sum_{k=1}^{K} x_k, \frac{1}{K}\sum_{k=1}^{K} y_k\right) = \mathcal{O}\left(\frac{1}{K}\right)$$

- **Result 2 [Weak convergence]:** Suppose that

$$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 < 4/(1+2\theta).$$

Then

$$(x_k, y_k) \rightharpoonup (x_\star, y_\star)$$

- **Result 3 [Tightness]:** Suppose that

$$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \geq 4/(1+2\theta).$$

Then there exists an example such that the Chambolle–Pock method does not converge

An extended convergence analysis of the Chambolle–Pock method

Chambolle–Pock method

$$
\begin{aligned}
x_{k+1} &= \mathrm{prox}_{\tau f}(x_k - \tau L^* y_k),\\
y_{k+1} &= \mathrm{prox}_{\sigma g^*}(y_k + \sigma L\,(x_{k+1} + \theta(x_{k+1} - x_k)))
\end{aligned}
$$

- **Result 1 [Ergodic convergence rate]:** Suppose that

  $$\theta \geq 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \leq 4/(1+2\theta)$$

  where at least one is strict. Then

  $$\mathcal{D}_{x_\star, y_\star}\left(\frac{1}{K}\sum_{k=1}^{K} x_k, \frac{1}{K}\sum_{k=1}^{K} y_k\right) = \mathcal{O}\Big(\frac{1}{K}\Big)$$

- **Result 2 [Weak convergence]:** Suppose that

  $$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 < 4/(1+2\theta).$$

  Then

  $$(x_k, y_k) \rightharpoonup (x_\star, y_\star)$$

- **Result 3 [Tightness]:** Suppose that

  $$\theta > 1/2 \quad \text{and} \quad \tau\sigma\|L\|^2 \geq 4/(1+2\theta).$$

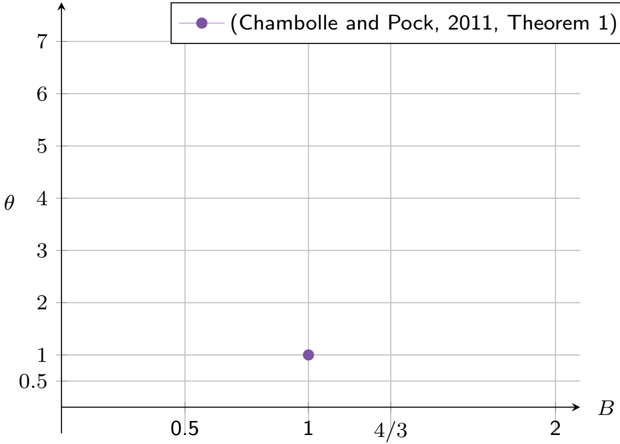
  Then there exists an example such that the Chambolle–Pock method does not converge

Comparison to results in the literature

Chambolle–Pock method

$$x_{k+1} = \mathrm{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \mathrm{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k)))$$

Convergence if $\tau\sigma \|L\|^2 < B$ where[2]



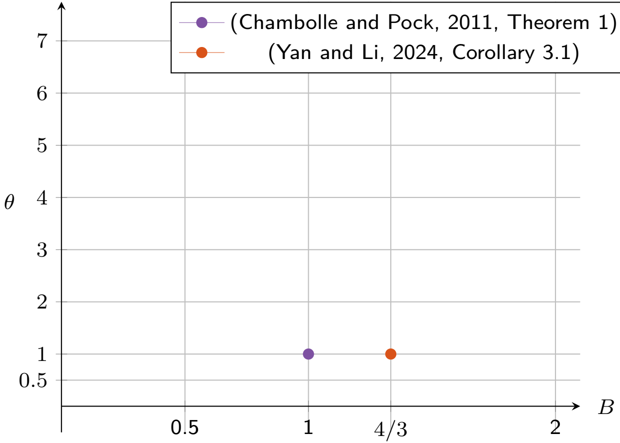
[2] larger B is better

Comparison to results in the literature

Chambolle–Pock method

$$x_{k+1} = \operatorname{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \operatorname{prox}_{\sigma g^*}(y_k + \sigma L\,(x_{k+1} + \theta(x_{k+1} - x_k)))$$

Convergence if $\tau\sigma\,\|L\|^2 < B$ where[2]



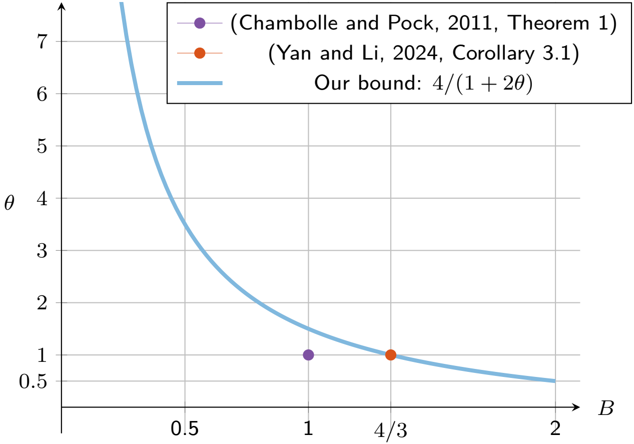
[2] larger B is better

Comparison to results in the literature

Chambolle–Pock method

$$x_{k+1} = \operatorname{prox}_{\tau f}(x_k - \tau L^* y_k),$$
$$y_{k+1} = \operatorname{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k)))$$

Convergence if $\tau\sigma \|L\|^2 < B$ where[2]



[2] larger B is better

Outline

1. Problem class
2. Algorithm representation
3. Lyapunov inequalities
4. Main result - A necessary and sufficient condition
5. Numerical results
6. Outlook

Problem class - Preliminaries

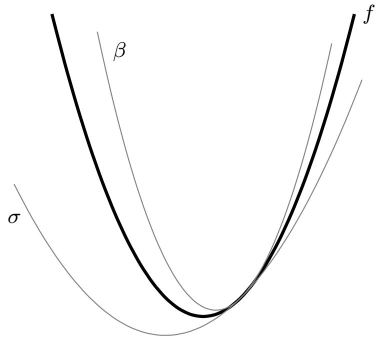
- $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ real Hilbert space. Associated norm $\|\cdot\|^2 = \langle\cdot,\cdot\rangle$
- Let $0 \leq \sigma < +\infty$ and $0 \leq \beta \leq +\infty$.

  $\mathcal{F}_{\sigma,\beta}$ class of all functions $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ that are proper, lower semicontinuous, σ-strongly convex and β-smooth (if $\beta < +\infty$)

Problem class - Preliminaries

- $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ real Hilbert space. Associated norm $\|\cdot\|^2 = \langle \cdot, \cdot \rangle$
- Let $0 \leq \sigma < +\infty$ and $0 \leq \beta \leq +\infty$.

  $\mathcal{F}_{\sigma,\beta}$ class of all functions $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ that are proper, lower semicontinuous, σ-strongly convex and β-smooth (if $\beta < +\infty$)



Problem class

- Convex optimization problem

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{m} f_i(y)$$

where $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ and $0 \leq \sigma_i < \beta_i \leq +\infty$, for each $i \in [\![1, m]\!]$ [3]

- Associated inclusion problem

$$\text{find } y \in \mathcal{H} \ \text{ such that } \ 0 \in \sum_{i=1}^{m} \partial f_i(y)$$

where ∂f_i are subdifferential operators

- Problem class $\mathcal{F}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ is all $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i,\beta_i}$ such that inclusion is solvable

[3] $[\![a, b]\!] = \{j \in \mathbb{Z} \mid a \leq j \leq b\}$

Problem class

- Convex optimization problem

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{m} f_i(y)$$

  where $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ and $0 \leq \sigma_i < \beta_i \leq +\infty$, for each $i \in [\![1, m]\!]$ [3]
- Associated inclusion problem

$$\text{find } y \in \mathcal{H} \ \text{ such that } \ 0 \in \sum_{i=1}^{m} \partial f_i(y)$$

  where ∂f_i are subdifferential operators
- Problem class $\mathcal{F}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ is all $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i,\beta_i}$ such that inclusion is solvable

[3] $[\![a, b]\!] = \{j \in \mathbb{Z} \mid a \leq j \leq b\}$

Problem class

- Convex optimization problem

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{m} f_i(y)$$

where $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ and $0 \leq \sigma_i < \beta_i \leq +\infty$, for each $i \in [\![1, m]\!]$ [3]

- Associated inclusion problem

$$\text{find } y \in \mathcal{H} \ \text{ such that } \ 0 \in \sum_{i=1}^{m} \partial f_i(y)$$

where ∂f_i are subdifferential operators

- Problem class $\boldsymbol{\mathcal{F}_{\sigma,\beta}}$ is all $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i,\beta_i}$ such that inclusion is solvable

[3] $[\![a, b]\!] = \{j \in \mathbb{Z} \mid a \leq j \leq b\}$

Outline

1 Problem class

2 **Algorithm representation**

3 Lyapunov inequalities

4 Main result - A necessary and sufficient condition

5 Numerical results

6 Outlook

Algorithm representation

Algorithms on state-space form[4] [5]:

$$\begin{aligned}\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\ \mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)\end{aligned}$$

where

$$A \in \mathbb{R}^{n \times n} \qquad B \in \mathbb{R}^{n \times m} \qquad C \in \mathbb{R}^{m \times n} \qquad D \in \mathbb{R}^{m \times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

$$\begin{aligned}&\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right) \\ &\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).\end{aligned}$$

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m \times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation

Algorithms on state-space form[4] [5]:

$$\begin{aligned}\mathbf{x}_{k+1} &= ({\color{red}A} \otimes \mathrm{Id})\mathbf{x}_k + ({\color{red}B} \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= ({\color{red}C} \otimes \mathrm{Id})\mathbf{x}_k + ({\color{red}D} \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\ \mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)\end{aligned}$$

where

$${\color{red}A} \in \mathbb{R}^{n\times n} \qquad {\color{red}B} \in \mathbb{R}^{n\times m} \qquad {\color{red}C} \in \mathbb{R}^{m\times n} \qquad {\color{red}D} \in \mathbb{R}^{m\times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

$$\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right)$$

$$\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).$$

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m\times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation

Algorithms on state-space form[4] [5]:

$$\begin{aligned}
\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\
\mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)
\end{aligned}$$

where

$$A \in \mathbb{R}^{n\times n} \qquad B \in \mathbb{R}^{n\times m} \qquad C \in \mathbb{R}^{m\times n} \qquad D \in \mathbb{R}^{m\times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

$$\begin{aligned}
&\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right) \\
&\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).
\end{aligned}$$

---

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m\times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation

Algorithms on state-space form[4] [5]:

$$\begin{aligned}
\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\
\mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)
\end{aligned}$$

where

$$A \in \mathbb{R}^{n\times n} \qquad B \in \mathbb{R}^{n\times m} \qquad C \in \mathbb{R}^{m\times n} \qquad D \in \mathbb{R}^{m\times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

$$\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right)$$

$$\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).$$

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m\times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation

Algorithms on state-space form[4] [5]:

$$\begin{aligned}
\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\
\mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)
\end{aligned}$$

where

$$A \in \mathbb{R}^{n\times n} \qquad B \in \mathbb{R}^{n\times m} \qquad C \in \mathbb{R}^{m\times n} \qquad D \in \mathbb{R}^{m\times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

$$\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right)$$

$$\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).$$

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m\times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation

Algorithms on state-space form[4] [5]:

$$
\begin{aligned}
\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\
\mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \\
\mathbf{F}_k &= \mathbf{f}(\mathbf{y}_k)
\end{aligned}
$$

where

$$A \in \mathbb{R}^{n\times n} \qquad B \in \mathbb{R}^{n\times m} \qquad C \in \mathbb{R}^{m\times n} \qquad D \in \mathbb{R}^{m\times m}$$

$$\mathbf{x}_k = \left(x_k^{(1)}, \ldots, x_k^{(n)}\right) \in \mathcal{H}^n \quad \mathbf{y}_k = \left(y_k^{(1)}, \ldots, y_k^{(m)}\right) \in \mathcal{H}^m \quad \mathbf{u}_k = \left(u_k^{(1)}, \ldots, u_k^{(m)}\right) \in \mathcal{H}^m$$

and

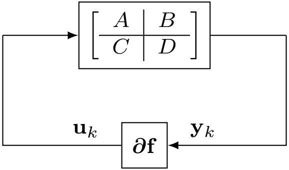
$$
\begin{aligned}
&\mathbf{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \left(f_1\left(y^{(1)}\right), \ldots, f_m\left(y^{(m)}\right)\right) \\
&\boldsymbol{\partial}\mathbf{f} : \mathcal{H}^m \rightrightarrows \mathcal{H}^m : \left(y^{(1)}, \ldots, y^{(m)}\right) \mapsto \prod_{i=1}^{m} \partial f_i\left(y^{(i)}\right).
\end{aligned}
$$

---

[4] Model used in control literature, (Lessard et al., 2016), and similar to the model in (Morin et al., 2024).

[5] Let $M \in \mathbb{R}^{m\times n}$ and $\mathbf{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. Then $(M \otimes \mathrm{Id})\mathbf{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)}\right)$.

Algorithm representation



Examples[6]

- gradient method
- proximal point method
- proximal gradient method
- Nesterov accelerated gradient method
- gradient method with heavy-ball momentum
- triple momentum method
- FISTA
- Davis–Yin three-operator splitting method
- Chambolle–Pock method
- etc.

[6] constant parameter versions

Algorithm representation - Chambolle–Pock method

- The problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y) + f_2(y)$$

- Method (Chambolle and Pock, 2011, Algorithm 1):

$$\begin{aligned} x_{k+1} &= \operatorname{prox}_{\tau_1 f_1}(x_k - \tau_1 y_k), \\ y_{k+1} &= \operatorname{prox}_{\tau_2 f_2^*}(y_k + \tau_2 \left(x_{k+1} + \theta(x_{k+1} - x_k)\right)) \end{aligned}$$

  where $\tau_1, \tau_2 > 0$, $\theta \in \mathbb{R}$ (linear operator set to identity mapping)

- On state-space form:

$$\begin{aligned} \mathbf{x}_{k+1} &= \left(\begin{bmatrix} 1 & -\tau_1 \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id} \right) \mathbf{x}_k + \left(\begin{bmatrix} -\tau_1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \mathrm{Id} \right) \mathbf{u}_k \\ \mathbf{y}_k &= \left(\begin{bmatrix} 1 & -\tau_1 \\ 1 & \frac{1}{\tau_2} - \tau_1(1+\theta) \end{bmatrix} \otimes \mathrm{Id} \right) \mathbf{x}_k + \left(\begin{bmatrix} -\tau_1 & 0 \\ -\tau_1(1+\theta) & -\frac{1}{\tau_2} \end{bmatrix} \otimes \mathrm{Id} \right) \mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial} \mathbf{f}(\mathbf{y}_k) \end{aligned}$$

Algorithm representation - Fixed points

- Algorithm *fixed points* $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$ satisfy

$$\begin{aligned}
\mathbf{x}_\star &= (A \otimes \mathrm{Id})\mathbf{x}_\star + (B \otimes \mathrm{Id})\mathbf{u}_\star \\
\mathbf{y}_\star &= (C \otimes \mathrm{Id})\mathbf{x}_\star + (D \otimes \mathrm{Id})\mathbf{u}_\star \\
\mathbf{u}_\star &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_\star) \\
\mathbf{F}_\star &= \mathbf{f}(\mathbf{y}_\star)
\end{aligned}$$

- Algorithm objective: find fixed point $\xi_\star$, extract solution from $\xi_\star$

Algorithm representation - Fixed points

- Algorithm *fixed points* $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$ satisfy

$$
\begin{aligned}
\mathbf{x}_\star &= (A \otimes \mathrm{Id})\mathbf{x}_\star + (B \otimes \mathrm{Id})\mathbf{u}_\star \\
\mathbf{y}_\star &= (C \otimes \mathrm{Id})\mathbf{x}_\star + (D \otimes \mathrm{Id})\mathbf{u}_\star \\
\mathbf{u}_\star &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_\star) \\
\mathbf{F}_\star &= \mathbf{f}(\mathbf{y}_\star)
\end{aligned}
$$

- Algorithm objective: find fixed point $\boldsymbol{\xi}_\star$, extract solution from $\boldsymbol{\xi}_\star$

Algorithm representation - Fixed-point encoding property

- We are only interested in algorithms such that

$$\text{"finding a fixed point} \quad \Longleftrightarrow \quad \text{solving inclusion problem"}$$

- More specifically[7]:
  - from each solution, it should be possible to construct a fixed point
  - from each fixed point, it should be possible to extract a solution
- Such algorithms have the *fixed-point encoding property* (FPEP)

[7] For the precise way to construct fixed points and extract solutions, see (Upadhyaya et al., 2024). This has been omitted from the presentation for clarity and simplicity

Algorithm representation - Fixed-point encoding property

- We are only interested in algorithms such that

$$\text{"finding a fixed point} \quad \Longleftrightarrow \quad \text{solving inclusion problem"}$$

- More specifically[7]:
  - from each solution, it should be possible to construct a fixed point
  - from each fixed point, it should be possible to extract a solution
- Such algorithms have the *fixed-point encoding property* (FPEP)

[7] For the precise way to construct fixed points and extract solutions, see (Upadhyaya et al., 2024). This has been omitted from the presentation for clarity and simplicity

Algorithm representation - Fixed-point encoding property

- We are only interested in algorithms such that

$$\text{"finding a fixed point} \quad \Longleftrightarrow \quad \text{solving inclusion problem"}$$

- More specifically[7]:
  - from each solution, it should be possible to construct a fixed point
  - from each fixed point, it should be possible to extract a solution
- Such algorithms have the *fixed-point encoding property* (FPEP)

[7] For the precise way to construct fixed points and extract solutions, see (Upadhyaya et al., 2024). This has been omitted from the presentation for clarity and simplicity

Algorithm representation - Fixed-point encoding property

- We are only interested in algorithms such that

$$\text{"finding a fixed point} \quad \iff \quad \text{solving inclusion problem"}$$

- More specifically[7]:
  - from each solution, it should be possible to construct a fixed point
  - from each fixed point, it should be possible to extract a solution
- Such algorithms have the *fixed-point encoding property* (FPEP)

[7] For the precise way to construct fixed points and extract solutions, see (Upadhyaya et al., 2024). This has been omitted from the presentation for clarity and simplicity

Algorithm representation - Fixed-point encoding property - Restrictions on (A, B, C, D)

- Let

$$N = \begin{bmatrix} I \\ -\mathbf{1}^\top \end{bmatrix} \in \mathbb{R}^{m \times (m-1)}$$

where $\mathbf{1}$ denotes the column vector of all ones of comfortable size

- Result:

The algorithm has the fixed-point encoding property

$$\iff$$

The matrices (A, B, C, D) satisfy

$$\operatorname{ran} \begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \subseteq \operatorname{ran} \begin{bmatrix} I - A \\ -C \end{bmatrix}$$

$$\operatorname{null} \begin{bmatrix} I - A & -B \end{bmatrix} \subseteq \operatorname{null} \begin{bmatrix} N^\top C & N^\top D \\ 0 & \mathbf{1}^\top \end{bmatrix}$$

(block row/column containing $N^\top/N$ removed when $m = 1$)

Algorithm representation - Fixed-point encoding property - Restrictions on (A, B, C, D)

- Let

$$N = \begin{bmatrix} I \\ -\mathbf{1}^\top \end{bmatrix} \in \mathbb{R}^{m\times(m-1)}$$

where $\mathbf{1}$ denotes the column vector of all ones of comfortable size

- Result:

The algorithm has the fixed-point encoding property

$$\Longleftrightarrow$$

The matrices (A, B, C, D) *satisfy*

$$\operatorname{ran}\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \subseteq \operatorname{ran}\begin{bmatrix} I - A \\ -C \end{bmatrix}$$

$$\operatorname{null}\begin{bmatrix} I - A & -B \end{bmatrix} \subseteq \operatorname{null}\begin{bmatrix} N^\top C & N^\top D \\ 0 & \mathbf{1}^\top \end{bmatrix}$$

(block row/column containing $N^\top/N$ *removed when* $m = 1$*)*

Algorithm representation - Well-posedness and uniqueness

- Recall: $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$ and

$$\begin{aligned} \mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \end{aligned}$$

- **Well-posedness**: Can we find at least one $\mathbf{x}_{k+1}$ for each $\mathbf{x}_k$?
- **Uniqueness**: If so, is $\mathbf{x}_{k+1}$ unique?

Algorithm representation - Well-posedness and uniqueness

- Recall: $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$ and

$$\begin{aligned} \mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \end{aligned}$$

- **Well-posedness**: Can we find at least one $\mathbf{x}_{k+1}$ for each $\mathbf{x}_k$?
- **Uniqueness**: If so, is $\mathbf{x}_{k+1}$ unique?

Algorithm representation - Well-posedness and uniqueness

- Recall: $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$ and

$$\begin{aligned}\mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k)\end{aligned}$$

- **Well-posedness**: Can we find at least one $\mathbf{x}_{k+1}$ for each $\mathbf{x}_k$?
- **Uniqueness**: If so, is $\mathbf{x}_{k+1}$ unique?

Algorithm representation - Well-posedness and uniqueness

- Recall: $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$ and

$$\begin{aligned} \mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \end{aligned}$$

- Sufficient condition for **well-posedness** and **uniqueness**:

D *lower triangular with nonpositive diagonal and*

$$\begin{aligned} I_{\text{differentiable}} &= \{i \in [\![1, m]\!] : \beta_i < +\infty\} \\ I_D &= \{i \in [\![1, m]\!] : [D]_{i,i} < 0]\} \end{aligned}$$

satisfy $I_{\text{differentiable}} \cup I_D = [\![1, m]\!]$

- Above is a running assumption

Algorithm representation - Well-posedness and uniqueness

- Recall: $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$ and

$$\begin{aligned} \mathbf{x}_{k+1} &= (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{y}_k &= (C \otimes \mathrm{Id})\mathbf{x}_k + (D \otimes \mathrm{Id})\mathbf{u}_k \\ \mathbf{u}_k &\in \boldsymbol{\partial}\mathbf{f}(\mathbf{y}_k) \end{aligned}$$

- Sufficient condition for **well-posedness** and **uniqueness**:

> D *lower triangular with nonpositive diagonal and*
>
> $$\begin{aligned} I_{\text{differentiable}} &= \{i \in [\![1, m]\!] : \beta_i < +\infty\} \\ I_D &= \{i \in [\![1, m]\!] : [D]_{i,i} < 0]\} \end{aligned}$$
>
> *satisfy* $I_{\text{differentiable}} \cup I_D = [\![1, m]\!]$

- Above is a running assumption

Algorithm representation - Explicit causal implementation

- Under the sufficient condition above, the algorithm can be implemented as

$$
\begin{array}{l}
\text{for } k = 0, 1, \ldots \\
\quad \text{for } i = 1, \ldots, m \\
\qquad v_k^{(i)} = \sum_{j=1}^{n} [C]_{i,j} x_k^{(j)} + \sum_{j=1}^{i-1} [D]_{i,j} u_k^{(j)}, \\
\qquad y_k^{(i)} = \begin{cases} \operatorname{prox}_{-[D]_{i,i} f_i}\left(v_k^{(i)}\right) & \text{if } i \in I_D, \\ v_k^{(i)} & \text{if } i \notin I_D, \end{cases} \\
\qquad u_k^{(i)} = \begin{cases} (-[D]_{i,i})^{-1}\left(v_k^{(i)} - y_k^{(i)}\right) & \text{if } i \in I_D, \\ \nabla f_i\left(y_k^{(i)}\right) & \text{if } i \notin I_D, \end{cases} \\
\quad \mathbf{x}_{k+1} = \left(x_{k+1}^{(1)}, \ldots, x_{k+1}^{(n)}\right) = (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k
\end{array}
$$

- Covers *all* frugal first-order methods with constant algorithm parameters where the subdifferentials are sampled either via gradient or proximal evaluation

Algorithm representation - Explicit causal implementation

- Under the sufficient condition above, the algorithm can be implemented as

$$
\begin{array}{l}
\text{for } k = 0, 1, \ldots \\
\quad \text{for } i = 1, \ldots, m \\
\quad\quad v_k^{(i)} = \sum_{j=1}^{n} [C]_{i,j} x_k^{(j)} + \sum_{j=1}^{i-1} [D]_{i,j} u_k^{(j)}, \\
\quad\quad y_k^{(i)} = \begin{cases} \operatorname{prox}_{-[D]_{i,i} f_i}\left(v_k^{(i)}\right) & \text{if } i \in I_D, \\ v_k^{(i)} & \text{if } i \notin I_D, \end{cases} \\
\quad\quad u_k^{(i)} = \begin{cases} (-[D]_{i,i})^{-1}\left(v_k^{(i)} - y_k^{(i)}\right) & \text{if } i \in I_D, \\ \nabla f_i\left(y_k^{(i)}\right) & \text{if } i \notin I_D, \end{cases} \\
\quad \mathbf{x}_{k+1} = \left(x_{k+1}^{(1)}, \ldots, x_{k+1}^{(n)}\right) = (A \otimes \mathrm{Id})\mathbf{x}_k + (B \otimes \mathrm{Id})\mathbf{u}_k
\end{array}
$$

- Covers *all* frugal first-order methods with constant algorithm parameters where the subdifferentials are sampled either via gradient or proximal evaluation

Outline

1 Problem class

2 Algorithm representation

3 **Lyapunov inequalities**

4 Main result - A necessary and sufficient condition

5 Numerical results

6 Outlook

Lyapunov inequalities

- Let $\boldsymbol{\xi}_k = (\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k, \mathbf{F}_k)$ and $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$
- Many first-order methods analyzed using *Lyapunov inequalities*

$$V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$$

  where $\rho \in [0, 1]$,
  - $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *Lyapunov function*
  - $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *residual function*

  and $\mathcal{S} = \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$
- Traditional way to find Lyapunov inequalities:
  - Use inequalities for the function classes involved (e.g. $\mathcal{F}_{\sigma_i, \beta_i}$)
  - Combine with algorithm updates
  - Manipulate to arrive at a Lyapunov inequality
- We want to automatically find such Lyapunov inequalities!

Lyapunov inequalities

- Let $\boldsymbol{\xi}_k = (\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k, \mathbf{F}_k)$ and $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$
- Many first-order methods analyzed using *Lyapunov inequalities*

$$V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$$

  where $\rho \in [0, 1]$,
  - $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *Lyapunov function*
  - $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *residual function*

  and $\mathcal{S} = \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$
- Traditional way to find Lyapunov inequalities:
  - Use inequalities for the function classes involved (e.g. $\mathcal{F}_{\sigma_i, \beta_i}$)
  - Combine with algorithm updates
  - Manipulate to arrive at a Lyapunov inequality
- We want to automatically find such Lyapunov inequalities!

Lyapunov inequalities

- Let $\boldsymbol{\xi}_k = (\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k, \mathbf{F}_k)$ and $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$
- Many first-order methods analyzed using *Lyapunov inequalities*

$$V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$$

  where $\rho \in [0, 1]$,
    - $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *Lyapunov function*
    - $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *residual function*

  and $\mathcal{S} = \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$
- Traditional way to find Lyapunov inequalities:
    - Use inequalities for the function classes involved (e.g. $\mathcal{F}_{\sigma_i, \beta_i}$)
    - Combine with algorithm updates
    - Manipulate to arrive at a Lyapunov inequality
- We want to automatically find such Lyapunov inequalities!

Lyapunov inequalities

- Let $\boldsymbol{\xi}_k = (\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k, \mathbf{F}_k)$ and $\boldsymbol{\xi}_\star = (\mathbf{x}_\star, \mathbf{u}_\star, \mathbf{y}_\star, \mathbf{F}_\star)$
- Many first-order methods analyzed using *Lyapunov inequalities*

$$V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$$

  where $\rho \in [0, 1]$,
  - $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *Lyapunov function*
  - $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a *residual function*

  and $\mathcal{S} = \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$
- Traditional way to find Lyapunov inequalities:
  - Use inequalities for the function classes involved (e.g. $\mathcal{F}_{\sigma_i, \beta_i}$)
  - Combine with algorithm updates
  - Manipulate to arrive at a Lyapunov inequality
- We want to automatically find such Lyapunov inequalities!

Lyapunov inequalities - Quadratic ansatzes

- We consider quadratic ansatzes of V and R:

$$V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \langle (\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star), (Q \otimes \mathrm{Id})(\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star) \rangle + q^\top (\mathbf{F} - \mathbf{F}_\star)$$
$$R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \langle (\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star), (S \otimes \mathrm{Id})(\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star) \rangle + s^\top (\mathbf{F} - \mathbf{F}_\star)$$

  where $Q, S \in \mathbb{S}^{n+2m}$, $q, s \in \mathbb{R}^m$ parameterize the functions[8]
- Our methodology searches for/provides (Q, q, S, s) that gives a valid Lyapunov inequality

[8] Inner-product $\langle \cdot, \cdot \rangle$ on $\mathcal{H}^d$ is given by

$$\langle \mathbf{z}_1, \mathbf{z}_2 \rangle = \sum_{i=1}^{d} \left\langle z_1^{(i)}, z_2^{(i)} \right\rangle$$

for each $\mathbf{z}_i = (z_i^{(1)}, \ldots, z_i^{(d)}) \in \mathcal{H}^d$ and $i \in [\![1, 2]\!]$

Lyapunov inequalities - Quadratic ansatzes

- We consider quadratic ansatzes of V and R:

$$V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \langle (\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star), (Q \otimes \mathrm{Id})(\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star) \rangle + q^\top (\mathbf{F} - \mathbf{F}_\star)$$
$$R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \langle (\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star), (S \otimes \mathrm{Id})(\mathbf{x} - \mathbf{x}_\star, \mathbf{u}, \mathbf{u}_\star) \rangle + s^\top (\mathbf{F} - \mathbf{F}_\star)$$

  where $Q, S \in \mathbb{S}^{n+2m}$, $q, s \in \mathbb{R}^m$ parameterize the functions[8]
- Our methodology searches for/provides (Q, q, S, s) that gives a valid Lyapunov inequality

[8] Inner-product $\langle \cdot, \cdot \rangle$ on $\mathcal{H}^d$ is given by

$$\langle \mathbf{z}_1, \mathbf{z}_2 \rangle = \sum_{i=1}^{d} \left\langle z_1^{(i)}, z_2^{(i)} \right\rangle$$

for each $\mathbf{z}_i = (z_i^{(1)}, \ldots, z_i^{(d)}) \in \mathcal{H}^d$ and $i \in [\![1, 2]\!]$

Lyapunov inequalities - Lower bounds

- However, we do not know (Q, q, S, s) that parameterize V and R in advance $\implies$ can not control convergence conclusions
- Solution: enforce nonnegative quadratic lower bounds on V and R

$$V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$$
$$R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$$

where $P, T \in \mathbb{S}^{n+2m}$ and $p, t \in \mathbb{R}^m$ are fixed

Lyapunov inequalities - Lower bounds

- However, we do not know (Q, q, S, s) that parameterize V and R in advance $\implies$ can not control convergence conclusions
- Solution: enforce nonnegative quadratic lower bounds on V and R

$$V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$$
$$R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$$

where $P, T \in \mathbb{S}^{n+2m}$ and $p, t \in \mathbb{R}^m$ are fixed

Lyapunov inequalities - Lower bounds - Convergence conclusions - Linear case

- Recall:
  C1 $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$
  C2 $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
  C3 $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
- For $\rho \in [0, 1[$

  $$0 \leq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \leq \rho^k V(\boldsymbol{\xi}_0, \boldsymbol{\xi}_\star) \to 0$$

  i.e., lower bound

  $$\left\{ \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \right\}_{k \in \mathbb{N}_0}$$

  converges ρ-linearly to 0

Lyapunov inequalities - Lower bounds - Convergence conclusions - Linear case

- Recall:
  C1 $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \le \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$
  C2 $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \ge \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \ge 0$
  C3 $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \ge \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \ge 0$
- For $\rho \in [0, 1[$

$$0 \le \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \le \rho^k V(\boldsymbol{\xi}_0, \boldsymbol{\xi}_\star) \to 0$$

  i.e., lower bound

$$\left\{ \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \right\}_{k \in \mathbb{N}_0}$$

  converges ρ-linearly to 0

Lyapunov inequalities - Lower bounds - Convergence conclusions - Sublinear case

- Recall:

  C1 $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$

  C2 $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$

  C3 $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$

- For $\rho = 1$, a telescoping summation gives

$$0 \leq \sum_{k=0}^{\infty} \Big(\langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \Big) \leq V(\boldsymbol{\xi}_0, \boldsymbol{\xi}_\star)$$

  i.e., lower bound

$$\Big\{ \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \Big\}_{k \in \mathbb{N}_0}$$

  is summable (and converges to zero)

- If the lower bound is convex in its arguments, Jensen's inequality gives that

$$\langle (\overline{\mathbf{x}}_k - \mathbf{x}_\star, \overline{\mathbf{u}}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\overline{\mathbf{x}}_k - \mathbf{x}_\star, \overline{\mathbf{u}}_k, \mathbf{u}_\star) \rangle + t^\top \Big(\overline{\mathbf{F}}_k - \mathbf{F}_\star \Big) = \mathcal{O}\left(\frac{1}{K} \right)$$

  where

$$\overline{\mathbf{x}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{x}_k \qquad \overline{\mathbf{u}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{u}_k \qquad \overline{\mathbf{F}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{F}_k$$

Lyapunov inequalities - Lower bounds - Convergence conclusions - Sublinear case

- Recall:
  - **C1** $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$
  - **C2** $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
  - **C3** $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
- For $\rho = 1$, a telescoping summation gives

$$0 \leq \sum_{k=0}^{\infty} \Big(\langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \Big) \leq V(\boldsymbol{\xi}_0, \boldsymbol{\xi}_\star)$$

  i.e., lower bound

$$\Big\{ \langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \Big\}_{k \in \mathbb{N}_0}$$

  is summable (and converges to zero)
- If the lower bound is convex in its arguments, Jensen's inequality gives that

$$\langle (\overline{\mathbf{x}}_k - \mathbf{x}_\star, \overline{\mathbf{u}}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\overline{\mathbf{x}}_k - \mathbf{x}_\star, \overline{\mathbf{u}}_k, \mathbf{u}_\star)\rangle + t^\top \Big(\overline{\mathbf{F}}_k - \mathbf{F}_\star \Big) = \mathcal{O}\left(\frac{1}{K}\right)$$

  where

$$\overline{\mathbf{x}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{x}_k \qquad \overline{\mathbf{u}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{u}_k \qquad \overline{\mathbf{F}}_k = \frac{1}{K} \sum_{k=1}^{K} \mathbf{F}_k$$

Lyapunov inequalities - Full definition

- (P, p, T, t, ρ)-*quadratic Lyapunov inequality* for algorithm and $\boldsymbol{\mathcal{F}_{\sigma,\beta}}$:
  C1 $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$
  C2 $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + p^\top(\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
  C3 $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + t^\top(\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
- Technical difficulty: Given (A, B, C, D), we only want this to hold for algorithm-consistent points ξ_k, fixed points $\xi_\star$, and $\mathrm{f} \in \mathcal{F}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$

Lyapunov inequalities - Full definition

- (P, p, T, t, ρ)-*quadratic Lyapunov inequality* for algorithm and $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$:
  C1 $V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star)$
  C2 $V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + p^\top(\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
  C3 $R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) \geq \langle(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star)\rangle + t^\top(\mathbf{F}_k - \mathbf{F}_\star) \geq 0$
- Technical difficulty: Given (A, B, C, D), we only want this to hold for algorithm-consistent points $\boldsymbol{\xi}_k$, fixed points $\boldsymbol{\xi}_\star$, and $\mathbf{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$

Outline

1. Problem class
2. Algorithm representation
3. Lyapunov inequalities
4. **Main result - A necessary and sufficient condition**
5. Numerical results
6. Outlook

Main result - Necessary and sufficient condition

There exists a (P, p, T, t, ρ)-quadratic Lyapunov inequality

if and only if[9]

a particular SDP involving (Q, q, S, s) is feasible

[9] Assuming dimension independence and Slater condition

Main result - Necessary and sufficient condition

There exists a (P, p, T, t, ρ)-quadratic Lyapunov inequality

if and only if[9]

a particular SDP involving (Q, q, S, s) is feasible

$$\text{C1}\begin{cases}\lambda^{\text{C1}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \{\varnothing, +, \star\},\\ \Sigma_\varnothing^\top(\rho Q - S)\Sigma_\varnothing - \Sigma_+^\top Q \Sigma_+ + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,+,\star\}\\ i\neq j}} \lambda^{\text{C1}}_{(l,i,j)} \mathbf{M}_{(l,i,j)} \succeq 0,\\ \begin{bmatrix}\rho q - s\\ -q\end{bmatrix} + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,+,\star\}\\ i\neq j}} \lambda^{\text{C1}}_{(l,i,j)} \mathbf{a}_{(l,i,j)} = 0,\end{cases}$$

$$\text{C2}\begin{cases}\lambda^{\text{C2}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \{\varnothing, \star\},\\ \Sigma_\varnothing^\top(Q - P)\Sigma_\varnothing + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\}\\ i\neq j}} \lambda^{\text{C2}}_{(l,i,j)} \mathbf{M}_{(l,i,j)} \succeq 0,\\ \begin{bmatrix}q - p\\ 0\end{bmatrix} + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\}\\ i\neq j}} \lambda^{\text{C2}}_{(l,i,j)} \mathbf{a}_{(l,i,j)} = 0,\end{cases}$$

$$\text{C3}\begin{cases}\lambda^{\text{C3}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \{\varnothing, \star\},\\ \Sigma_\varnothing^\top(S - T)\Sigma_\varnothing + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\}\\ i\neq j}} \lambda^{\text{C3}}_{(l,i,j)} \mathbf{M}_{(l,i,j)} \succeq 0,\\ \begin{bmatrix}s - t\\ 0\end{bmatrix} + \sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\}\\ i\neq j}} \lambda^{\text{C3}}_{(l,i,j)} \mathbf{a}_{(l,i,j)} = 0,\end{cases}$$

[9] Assuming dimension independence and Slater condition

Main result - How did we find this condition?

- Let us look at **C1**:[10] $V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star)$
- **C1** equivalent to that optimal value of

$$\begin{array}{ll} \text{maximize} & V(\xi_+, \xi_\star) - \rho V(\xi, \xi_\star) + R(\xi, \xi_\star) \\[1em] \text{subject to} & \xi \text{ is algorithm consistent for } \mathbf{f}, \\ & \xi_+ \text{ is a successor of } \xi \text{ for } \mathbf{f}, \\ & \xi_\star \text{ is a fixed point for } \mathbf{f}, \\ & \mathbf{f} \in \mathcal{F}_{\boldsymbol{\sigma}, \boldsymbol{\beta}}, \end{array} \qquad \text{(PEP)}$$

  is nonpositive!
- Arrived at the condition using:
  - Convex interpolation conditions (Taylor et al., 2017*b*)
  - Performance estimation problem (PEP) reformulations (Drori and Teboulle, 2014)

[10] We use the same trick for **C2** and **C3**

Main result - How did we find this condition?

- Let us look at **C1**:[10] $V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star)$
- **C1** equivalent to that optimal value of

$$\begin{aligned} \text{maximize} \quad & V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) - \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) + R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \\ \text{subject to} \quad & \boldsymbol{\xi} \text{ is algorithm consistent for } \mathbf{f}, \\ & \boldsymbol{\xi}_+ \text{ is a successor of } \boldsymbol{\xi} \text{ for } \mathbf{f}, \\ & \boldsymbol{\xi}_\star \text{ is a fixed point for } \mathbf{f}, \\ & \mathbf{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma}, \boldsymbol{\beta}}, \end{aligned} \tag{PEP}$$

  is nonpositive!
- Arrived at the condition using:
  - Convex interpolation conditions (Taylor et al., 2017*b*)
  - Performance estimation problem (PEP) reformulations (Drori and Teboulle, 2014)

[10] We use the same trick for **C2** and **C3**

Main result - How did we find this condition?

- Let us look at **C1**:[10] $V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star)$
- **C1** equivalent to that optimal value of

$$
\begin{array}{ll}
\text{maximize} & V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) - \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) + R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \\
\text{subject to} & \boldsymbol{\xi} \text{ is algorithm consistent for } \mathbf{f}, \\
& \boldsymbol{\xi}_+ \text{ is a successor of } \boldsymbol{\xi} \text{ for } \mathbf{f}, \\
& \boldsymbol{\xi}_\star \text{ is a fixed point for } \mathbf{f}, \\
& \mathbf{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}},
\end{array}
\tag{PEP}
$$

  is nonpositive!
- Arrived at the condition using:
  - Convex interpolation conditions (Taylor et al., 2017*b*)
  - Performance estimation problem (PEP) reformulations (Drori and Teboulle, 2014)

[10] We use the same trick for **C2** and **C3**

Main result - How did we find this condition?

- Let us look at **C1**:[10] $V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star)$
- **C1** equivalent to that optimal value of

$$\begin{aligned} \text{maximize} \quad & V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) - \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) + R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \\ \text{subject to} \quad & \boldsymbol{\xi} \text{ is algorithm consistent for } \mathbf{f}, \\ & \boldsymbol{\xi}_+ \text{ is a successor of } \boldsymbol{\xi} \text{ for } \mathbf{f}, \\ & \boldsymbol{\xi}_\star \text{ is a fixed point for } \mathbf{f}, \\ & \mathbf{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma}, \boldsymbol{\beta}}, \end{aligned} \tag{PEP}$$

  is nonpositive!
- Arrived at the condition using:
  - Convex interpolation conditions (Taylor et al., 2017*b*)
  - Performance estimation problem (PEP) reformulations (Drori and Teboulle, 2014)

[10] We use the same trick for **C2** and **C3**

Outline

1. Problem class
2. Algorithm representation
3. Lyapunov inequalities
4. Main result - A necessary and sufficient condition
5. **Numerical results**
6. Outlook

Numerical results - Douglas–Rachford method

- The problem:

$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f_1(y) + f_2(y)$$

where $f_1 \in \mathcal{F}_{1,2}$ and $f_2 \in \mathcal{F}_{0,\infty}$

- Douglas–Rachford method:

$$\begin{aligned} y_k^{(1)} &= \text{prox}_{\gamma f_1}(x_k) \\ y_k^{(2)} &= \text{prox}_{\gamma f_2}\left(2y_k^{(1)} - x_k\right) \\ x_{k+1} &= x_k + \lambda\left(y_k^{(2)} - y_k^{(1)}\right) \end{aligned}$$

where $\gamma \in \mathbb{R}_{++}$ and $\lambda \in \mathbb{R} \setminus \{0\}$ ($\lambda = 1$ in the plot below)

- $(P, p, T, t, \rho) \implies$ squared distance to the solution convergence ρ-linearly to zero[11]

[11] Smallest ρ via bisection search

Numerical results - Douglas–Rachford method

- The problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y) + f_2(y)$$

  where $f_1 \in \mathcal{F}_{1,2}$ and $f_2 \in \mathcal{F}_{0,\infty}$
- Douglas–Rachford method:

$$\begin{aligned} y_k^{(1)} &= \text{prox}_{\gamma f_1}(x_k) \\ y_k^{(2)} &= \text{prox}_{\gamma f_2}\left(2y_k^{(1)} - x_k\right) \\ x_{k+1} &= x_k + \lambda\left(y_k^{(2)} - y_k^{(1)}\right) \end{aligned}$$

  where $\gamma \in \mathbb{R}_{++}$ and $\lambda \in \mathbb{R} \setminus \{0\}$ ($\lambda = 1$ in the plot below)
- $(P, p, T, t, \rho) \implies$ squared distance to the solution convergence ρ-linearly to zero[11]

[11] Smallest ρ via bisection search

Numerical results - Douglas–Rachford method

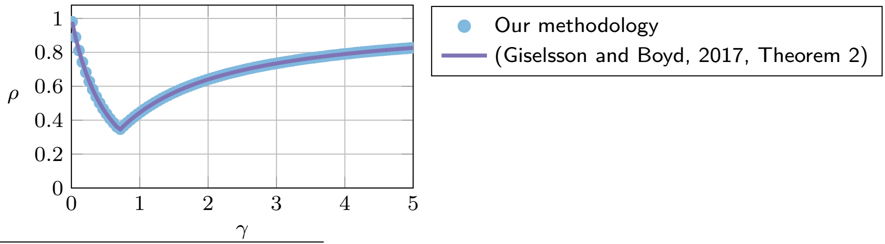
- The problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y) + f_2(y)$$

  where $f_1 \in \mathcal{F}_{1,2}$ and $f_2 \in \mathcal{F}_{0,\infty}$
- Douglas–Rachford method:

$$\begin{aligned} y_k^{(1)} &= \operatorname{prox}_{\gamma f_1}(x_k) \\ y_k^{(2)} &= \operatorname{prox}_{\gamma f_2}\left(2y_k^{(1)} - x_k\right) \\ x_{k+1} &= x_k + \lambda\left(y_k^{(2)} - y_k^{(1)}\right) \end{aligned}$$

  where $\gamma \in \mathbb{R}_{++}$ and $\lambda \in \mathbb{R} \setminus \{0\}$ ($\lambda = 1$ in the plot below)
- $(P, p, T, t, \rho) \implies$ squared distance to the solution convergence ρ-linearly to zero[11]



[11] Smallest ρ via bisection search

Numerical results - Gradient method with heavy-ball momentum

- The problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y)$$

  where $f_1 \in \mathcal{F}_{0,1}$
- Gradient method with heavy-ball momentum:

$$x_{k+1} = x_k - \gamma \nabla f_1(x_k) + \delta(x_k - x_{k-1})$$

- $(P, p, T, t, \rho) \implies \lim_{k\to\infty}(f_1(x_k) - f_1(x_\star)) = 0$ and $f_1\left(\frac{1}{K}\sum_{k=1}^{K} x_k\right) - f_1(x_\star) = \mathcal{O}\left(\frac{1}{K}\right)$

Numerical results - Gradient method with heavy-ball momentum

- The problem:

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y)$$

  where $f_1 \in \mathcal{F}_{0,1}$
- Gradient method with heavy-ball momentum:

$$x_{k+1} = x_k - \gamma \nabla f_1(x_k) + \delta(x_k - x_{k-1})$$

- $(P, p, T, t, \rho) \implies \lim_{k\to\infty}(f_1(x_k) - f_1(x_\star)) = 0$ and $f_1\left(\frac{1}{K}\sum_{k=1}^{K} x_k\right) - f_1(x_\star) = \mathcal{O}\left(\frac{1}{K}\right)$

Numerical results - Gradient method with heavy-ball momentum

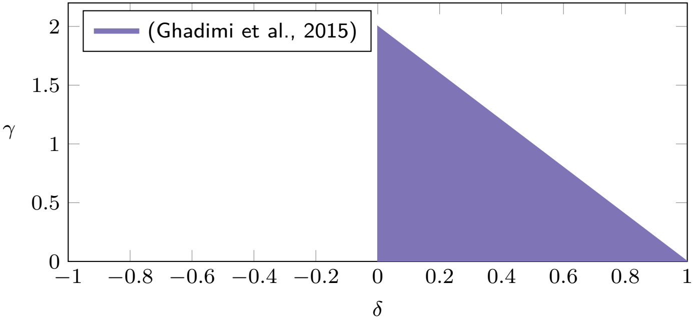
- The problem:

$$\underset{y\in\mathcal{H}}{\text{minimize}} \quad f_1(y)$$

  where $f_1 \in \mathcal{F}_{0,1}$
- Gradient method with heavy-ball momentum:

$$x_{k+1} = x_k - \gamma\nabla f_1(x_k) + \delta(x_k - x_{k-1})$$

- $(P, p, T, t, \rho) \implies \lim_{k\to\infty}(f_1(x_k) - f_1(x_\star)) = 0$ and $f_1\left(\frac{1}{K}\sum_{k=1}^{K} x_k\right) - f_1(x_\star) = \mathcal{O}\left(\frac{1}{K}\right)$



Numerical results - Gradient method with heavy-ball momentum

- The problem:

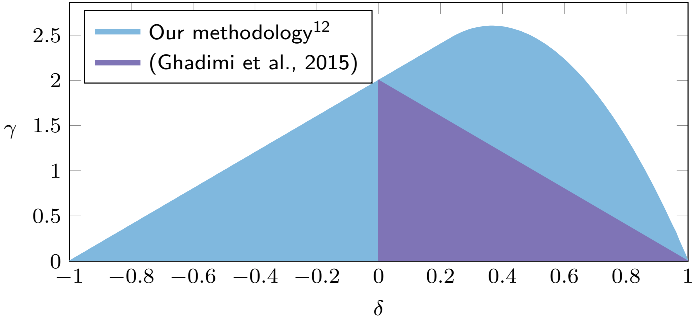
$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad f_1(y)$$

  where $f_1 \in \mathcal{F}_{0,1}$

- Gradient method with heavy-ball momentum:

$$x_{k+1} = x_k - \gamma \nabla f_1(x_k) + \delta(x_k - x_{k-1})$$

- $(P, p, T, t, \rho) \implies \lim_{k\to\infty}(f_1(x_k) - f_1(x_\star)) = 0$ and $f_1\left(\frac{1}{K}\sum_{k=1}^{K} x_k\right) - f_1(x_\star) = \mathcal{O}\left(\frac{1}{K}\right)$



[9] Parameters evaluated on a square grid of size 0.01×0.01

Outline

1. Problem class
2. Algorithm representation
3. Lyapunov inequalities
4. Main result - A necessary and sufficient condition
5. Numerical results
6. **Outlook**

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i, \beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i, \beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

Summary and outlook

- **Summary:**
  - A framework for automated convergence proofs for first-order methods used to solve convex optimization problems
  - Introduced a state-space representation based on matrices A, B, C, D
  - Introduced a necessary and sufficient condition for the existence of quadratic Lyapunov inequalities
  - Numerical and analytical examples extending previous results
- **Outlook:**
  - Iteration dependent algorithms (A_k, B_k, C_k, D_k)
  - Change $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ to any function and/or operator class that has quadratic interpolation constraints:
    - class of smooth functions (Taylor et al., 2017*a*)
    - class of convex and quadratically upper bounded functions (Goujaud et al., 2022)
    - class of convex and Lipschitz continuous functions (Taylor et al., 2017*a*)
    - class of smooth hypoconvex (weakly convex) functions (Rotaru et al., 2022)
    - class of smooth functions satisfying the Polyak-Łojasiewicz inequality (Abbaszadehpeivasti et al., 2023)
    - class of monotone operators (also with stronger assumptions) (Ryu et al., 2020)
  - Allow multiple evaluations of the same subdifferential ∂f_i during the same iteration (non-frugal methods)
    - enabling analysis of, e.g., the forward–backward–forward splitting method of Tseng (Tseng, 2000)
  - Extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k-h}$ for some integer $h \geq 0$

References I

Abbaszadehpeivasti, H., De Klerk, E. and Zamani, M. (2023), 'Conditions for linear convergence of the gradient method for non-convex optimization.', *Optimization Letters* pp. 1105 – 1125.

Banert, S., Upadhyaya, M. and Giselsson, P. (2023), 'The chambolle-pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|l\|^2 < 4/(1+2\theta)$', *arXiv:2309.03998* .

Chambolle, A. and Pock, T. (2011), 'A first-order primal-dual algorithm for convex problems with applications to imaging', *Journal of Mathematical Imaging and Vision* **40**(1), 120–145.

Drori, Y. and Teboulle, M. (2014), 'Performance of first-order methods for smooth convex minimization: a novel approach', *Mathematical Programming* **145**(1/2), 451–482.

Ghadimi, E., Feyzmahdavian, H. R. and Johansson, M. (2015), Global convergence of the heavy-ball method for convex optimization, *in* '2015 European Control Conference (ECC)', pp. 310–315.

Giselsson, P. and Boyd, S. (2017), 'Linear convergence and metric selection for Douglas-Rachford splitting and ADMM', *IEEE Transactions on Automatic Control* **62**(2), 532–544.

Goujaud, B., Taylor, A. and Dieuleveut, A. (2022), 'Optimal first-order methods for convex functions with a quadratic upper bound'.

Lessard, L., Recht, B. and Packard, A. (2016), 'Analysis and design of optimization algorithms via integral quadratic constraints', *SIAM Journal on Optimization* **26**(1), 57–95.

Morin, M., Banert, S. and Giselsson, P. (2024), 'Frugal splitting operators: Representation, minimal lifting, and convergence', *SIAM Journal on Optimization* **34**(2), 1595–1621.

Rotaru, T., Glineur, F. and Patrinos, P. (2022), 'Tight convergence rates of the gradient method on smooth hypoconvex functions'.

References II

Ryu, E. K., Taylor, A. B., Bergeling, C. and Giselsson, P. (2020), 'Operator splitting performance estimation: tight contraction factors and optimal parameter selection', *SIAM Journal on Optimization* **30**(3), 2251–2271.

Taylor, A. B., Hendrickx, J. M. and Glineur, F. (2017*a*), 'Exact worst-case performance of first-order methods for composite convex optimization', *SIAM Journal on Optimization* **27**(3), 1283–1313.

Taylor, A. B., Hendrickx, J. M. and Glineur, F. (2017*b*), 'Smooth strongly convex interpolation and exact worst-case performance of first-order methods.', *Mathematical Programming* **161**(1/2), 307–345.

Tseng, P. (2000), 'A modified forward-backward splitting method for maximal monotone mappings', *SIAM Journal on Control and Optimization* **38**(2), 431–446.

Upadhyaya, M., Banert, S., Taylor, A. B. and Giselsson, P. (2024), 'Automated tight lyapunov analysis for first-order methods', *Mathematical Programming* .

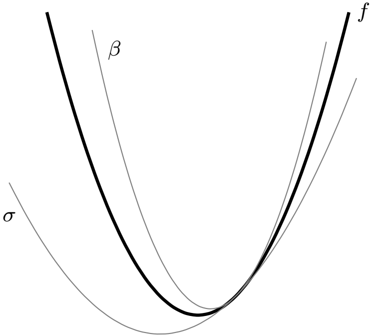
Yan, M. and Li, Y. (2024), 'On the improved conditions for some primal-dual algorithms.', *Journal of Scientific Computing* **99**(3).

Appendix - Preliminaries

- $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ real Hilbert space. Associated norm $\|\cdot\|^2 = \langle \cdot, \cdot \rangle$
- Let $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$. Then:
  (i) *effective domain* of f is the set $\operatorname{dom} f = \{x \in \mathcal{H} \mid f(x) < +\infty\}$
  (ii) f *proper* if $\operatorname{dom} f \neq \emptyset$
  (iii) *subdifferential* of a proper function f is the set-valued operator $\partial f : \mathcal{H} \to 2^{\mathcal{H}}$ such that $x \mapsto \{u \in \mathcal{H} \mid \forall y \in \mathcal{H},\ f(y) \geq f(x) + \langle u, y - x \rangle\}$
- Let $f \colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ and $\sigma, \beta \in \mathbb{R}_+$. The function f is:
  (i) *convex* if $f((1-\lambda)x + \lambda y) \leq (1-\lambda) f(x) + \lambda f(y)$ for each $x, y \in \mathcal{H}$ and $0 \leq \lambda \leq 1$
  (ii) σ*-strongly convex* if f is proper and $f - (\sigma/2) \|\cdot\|^2$ is convex
  (iii) β*-smooth* if f is differentiable and $\|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\|$ for each $x, y \in \mathcal{H}$

Appendix - More preliminaries

- Let $0 \leq \sigma < +\infty$ and $0 \leq \beta \leq +\infty$. $\mathcal{F}_{\sigma,\beta}$ class of all functions $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ that are proper, lower semicontinuous, σ-strongly convex and β-smooth (if $\beta < +\infty$)



- Let $f \in \mathcal{F}_{0,\infty}$ and $\gamma > 0$. Then the *proximal operator* $\mathrm{prox}_{\gamma f} : \mathcal{H} \to \mathcal{H}$ is defined as the single-valued operator given by

$$\mathrm{prox}_{\gamma f}(x) = \underset{z \in \mathcal{H}}{\mathrm{argmin}} \left(f(z) + \frac{1}{2\gamma} \|x - z\|^2 \right)$$

  for each $x \in \mathcal{H}$

- The *convex conjugate* of f, denoted $f^* : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is the proper, lower semicontinuous and convex function given by $f^*(u) = \sup_{x \in \mathcal{H}} (\langle u, x \rangle - f(x))$ for each $u \in \mathcal{H}$

Appendix - Some choices of (P, p, T, t, ρ)

Suppose $\rho \in [0, 1[$, let e_i be ith standard basis vector and

$$(P, p, T, t) = \left(\begin{bmatrix} C & D & -D \end{bmatrix}^\top e_i e_i^\top \begin{bmatrix} C & D & -D \end{bmatrix}, 0, 0, 0 \right).$$

Then

$$\langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (P \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + p^\top (\mathbf{F}_k - \mathbf{F}_\star) = \left\| y_k^{(i)} - y_\star \right\|^2 \geq 0$$

and the distance to the solution squared converges ρ-linear to zero.

Appendix - Some choices of (P, p, T, t, ρ)

Suppose $\rho = 1$, $m = 1$ and let

$$(P, p, T, t) = (0, 0, 0, 1).$$

Then

$$\langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) = f_1\left(y_k^{(1)}\right) - f_1(y_\star) \geq 0$$

which gives

- function value suboptimality converges to zero
- $\mathcal{O}(1/k)$ ergodic function value suboptimality convergence (via Jensen's inequality)

Appendix - Some choices of (P, p, T, t, ρ)

Suppose $\rho = 1$ and let

$$(P, p, T, t) = \left(0, 0, \begin{bmatrix} C & D & -D \\ 0 & 0 & I \end{bmatrix}^\top \begin{bmatrix} 0 & -\frac{1}{2}I \\ -\frac{1}{2}I & 0 \end{bmatrix} \begin{bmatrix} C & D & -D \\ 0 & 0 & I \end{bmatrix}, \mathbf{1}\right).$$

Then

$$\begin{aligned} &\langle (\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star), (T \otimes \mathrm{Id})(\mathbf{x}_k - \mathbf{x}_\star, \mathbf{u}_k, \mathbf{u}_\star) \rangle + t^\top (\mathbf{F}_k - \mathbf{F}_\star) \\ &= \sum_{i=1}^{m} \left(f_i\left(y_k^{(i)}\right) - f_i\left(y_\star^{(i)}\right) - \left\langle u_\star^{(i)}, y_k^{(i)} - y_\star^{(i)} \right\rangle \right) \\ &= \mathcal{L}(\mathbf{y}_k, \mathbf{u}_\star) - \mathcal{L}(\mathbf{y}_\star, \mathbf{u}_k) \geq 0 \end{aligned}$$

where $\mathcal{L} : \mathcal{H}^m \times \mathcal{H}^m \to \mathbb{R}$ is a *Lagrangian function* giving

- primal-dual gap converges to zero,
- $\mathcal{O}(1/k)$ ergodic primal-dual gap convergence (via Jensen's inequality).

Reduces to function value suboptimality when $m = 1$.

Appendix - Chambolle–Pock - Lyapunov inequality

- Lyapunov function:

$$
\begin{aligned}
V_k = \theta(f(x_{k+1}) - f(x_\star) + \langle L^* y_\star, x_{k+1} - x_\star \rangle) + \frac{1}{2\tau} \|x_{k+1} - x_\star\|^2 \\
+ \frac{1}{2\sigma} \|y_k - y_\star + \sigma\theta(Lx_{k+1} - Lx_k)\|^2 + \frac{\theta}{2\tau} \|x_{k+1} - x_k\|^2 \\
- \frac{\sigma\left(4\theta^2 + 1\right)}{16} \|Lx_{k+1} - Lx_k\|^2 \geq 0
\end{aligned}
$$

- Lyapunov inequality:

$$
\begin{aligned}
& V_{k+1} - V_k + \mathcal{D}_{x_\star, y_\star}(x_{k+1}, y_{k+1}) \leq \\
& - \frac{1}{2\sigma} \left\| y_{k+1} - y_k - \sigma\left(\frac{1}{2}(Lx_{k+1} - Lx_{k+2}) - \theta(Lx_k - Lx_{k+1}) \right) \right\|^2 \\
& - \frac{8\theta - \tau\sigma \|L\|^2 \left(4\theta^2 + 1\right)}{16\tau} \left\| x_{k+2} - x_{k+1} - \frac{4\left(1 - \tau_1 \sigma\theta \|L\|^2\right)}{8\theta - \tau\sigma \|L\|^2 (4\theta^2 + 1)} (x_{k+1} - x_k) \right\|^2 \\
& - \frac{\left(4\theta^2 - 1\right)\left(4 - \tau\sigma \|L\|^2 (2\theta + 1)\right)\left(4 - \tau\sigma \|L\|^2 (2\theta - 1)\right)}{16\tau\left(8\theta - \tau\sigma \|L\|^2 (4\theta^2 + 1)\right)} \|x_{k+1} - x_k\|^2 \leq 0
\end{aligned}
$$

Appendix - Chambolle–Pock - Counterexample

- For the problem instance ($\mathcal{H} = \mathcal{G} = \mathbb{R}$, $f = g^* = 0$, and $L = 1$)

$$\underset{x\in\mathbb{R}}{\text{minimize}}\,\underset{y\in\mathbb{R}}{\text{maximize}}\ xy$$

  the Chambolle–Pock method becomes

$$\begin{aligned} x_{k+1} &= x_k - \tau y_k, \\ y_{k+1} &= y_k + \sigma(x_{k+1} + \theta(x_{k+1} - x_k)), \end{aligned}$$

  or equivalently

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & -\tau \\ \sigma & 1 - \tau\sigma(1+\theta) \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix},$$

  where we assume that $\tau, \sigma, \theta > 0$
- If

$$\tau\sigma \geq 4/(1+2\theta),$$

  the matrix above has an eigenvalue with a magnitude greater or equal to 1
- The Chambolle–Pock method no longer converges for all initial points