On Fundamental Proof Structures in First-Order Optimization

Baptiste Goujaud

62nd IEEE Conference on Decision and Control (CDC 2023) – Singapore – December, 14th 2023

Collaborators

Pr. Aymeric Dieuleveut

Pr. Adrien Taylor

Goal and Roadmap

Goal of today: Improving our in-depth understanding of how proofs for first order optimization work.

Goal and Roadmap

Goal of today: Improving our in-depth understanding of how proofs for first order optimization work.

Roadmap:

- ◇ From explicit to implicit characterization of function classes.
- ◇ From explicit to implicit characterization of algorithms.
- ◇ Insights from the dual formulation of the problem.
- ◇ Natural performance metrics.
- ◇ Extensions of a certificate to larger set of class and algorithms.
- ◇ Lyapunov approaches.

References

B.G., D. Scieur, A. Dieuleveut, A. Taylor, F. Pedregosa (2022). Super-acceleration with cyclical step-sizes.

B.G., A. Taylor, A. Dieuleveut (2022). Quadratic minimization: from conjugate gradient to an adaptive Heavy-ball method with Polyak step-sizes.

Y. Drori, M. Teboulle (2014). Performance of first-order methods for smooth convex minimization: a novel approach.

A. Taylor, J. Hendrickx, F. Glineur (2017). Smooth strongly convex interpolation and exact worst-case performance of first-order methods.

L. Lessard, B. Recht, A .Packard (2014). Analysis and Design of Optimization Algorithms via Integral Quadratic Constraints.

Y. Drori and A. Taylor (2020). Efficient first-order methods for convex minimization: a constructive approach.

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

A. Taylor, B. Van Scoy, L. Lessard (2018). Lyapunov functions for first-order methods: Tight automated convergence guarantees

References

C. Park, E.K. Ryu (2021). Optimal First-Order Algorithms as a Function of Inequalities
N. Bousselmi, J.M. Hendrickx, F. Glineur (2023). Interpolation Conditions for Linear Operators and applications to Performance Estimation Problems
A. Taylor, J. Hendrickx, F. Glineur (2017). Performance Estimation Toolbox (PESTO): automated worst-case analysis of first-order optimization methods.
B.G., C. Moucer, F. Glineur, J. Hendrickx, A. Taylor, A. Dieuleveut (2022). "PEPit: computer-assisted worst-case analyses of first-order optimization methods in Python."

First order optimization

Optimization: We aim to solve

$$f_\star \triangleq \min_{x \in \mathbb{R}^d} f(x) \qquad \text{(OPT)}$$

First order optimization

Optimization: We aim to solve

$$f_\star \triangleq \min_{x \in \mathbb{R}^d} f(x) \qquad \text{(OPT)}$$

First order oracle:

- ◇ Allowed: request function values and gradients of f at some points.
  Examples:
  - Gradient descent: $x_{t+1} = x_t - \gamma_t \nabla f(x_t)$.
  - Heavy-ball: $x_{t+1} = x_t - \gamma_t \nabla f(x_t) + \beta_t (x_t - x_{t-1})$.
- ◇ Forbidden: request hessian and higher order derivatives.
  Examples:
  - Newton method: $x_{t+1} = x_t - \nabla^2 f(x_t)^{-1} \nabla f(x_t)$.

First order optimization

Optimization: We aim to solve

$$f_\star \triangleq \min_{x \in \mathbb{R}^d} f(x) \qquad \text{(OPT)}$$

First order oracle:

- ◇ Allowed: request function values and gradients of f at some points.
  Examples:
  - Gradient descent: $x_{t+1} = x_t - \gamma_t \nabla f(x_t)$.
  - Heavy-ball: $x_{t+1} = x_t - \gamma_t \nabla f(x_t) + \beta_t (x_t - x_{t-1})$.
- ◇ Forbidden: request hessian and higher order derivatives.
  Examples:
  - Newton method: $x_{t+1} = x_t - \nabla^2 f(x_t)^{-1} \nabla f(x_t)$.

Worst-case study:

- ◇ Fixed first order algorithm $\mathcal{A}$.
- ◇ Fixed class (set) of functions $\mathcal{F}$.
- ◇ Question: what *a priori* guarantees after N iterations?
  Examples:

$$\begin{aligned}
\forall f \in \mathcal{F}, \quad & \|x_N - x_\star\|^2 && \leq \rho_N \|x_0 - x_\star\|^2 \\
\forall f \in \mathcal{F}, \quad & f(x_N) - f_\star && \leq \rho_N \|x_0 - x_\star\|^2 \\
\forall f \in \mathcal{F}, \quad & \|\nabla f(x_N)\|^2 && \leq \rho_N (f(x_0) - f_\star)
\end{aligned}$$

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

First order methods: $x_N = x_{N-1} - \sum_{i=0}^{N-1} h_{N,i} \nabla f(x_i)$

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

First order methods: $x_N = x_{N-1} - \sum_{i=0}^{N-1} h_{N,i} \nabla f(x_i)$

$$\Updownarrow$$

Link with polynomials: $x_N - x_\star = P_N(H)(x_0 - x_\star)$.

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

First order methods: $x_N = x_{N-1} - \sum_{i=0}^{N-1} h_{N,i} \nabla f(x_i)$

$$\Updownarrow$$

Link with polynomials: $x_N - x_\star = P_N(H)(x_0 - x_\star)$.

$Sp(H) \subset [\mu, L]$
$\hookrightarrow$ Momentum.

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

First order methods: $x_N = x_{N-1} - \sum_{i=0}^{N-1} h_{N,i} \nabla f(x_i)$

$$\Updownarrow$$

Link with polynomials: $x_N - x_\star = P_N(H)(x_0 - x_\star)$.

$Sp(H) \subset [\mu, L]$
$\hookrightarrow$ Momentum.

$Sp(H) \subset [\mu_1, L_1] \cup [\mu_2, L_2]$
$\hookrightarrow$ Cycling step-size.

Class example: strongly convex quadratic functions

Quadratic functions: $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

First order methods: $x_N = x_{N-1} - \sum_{i=0}^{N-1} h_{N,i} \nabla f(x_i)$

$$\Updownarrow$$

Link with polynomials: $x_N - x_\star = P_N(H)(x_0 - x_\star)$.

$Sp(H) \subset [\mu, L]$
$\hookrightarrow$ Momentum.

$Sp(H) \subset [\mu_1, L_1] \cup [\mu_2, L_2]$
$\hookrightarrow$ Cycling step-size.

$Sp(H)$
$\hookrightarrow$ Conjugate gradient.

Implicit definition of classes

Explicit definition of a class: parametrization.

Example:

- Quadratic functions $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

Implicit definition of classes

Explicit definition of a class: parametrization.

Example:

- ◇ Quadratic functions $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

Implicit definition of a class: constraints.

Examples:

- ◇ Convex functions $\mathcal{F} = \{f \mid \forall x, y \ f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle\}$.
- ◇ Smooth functions $\mathcal{F} = \left\{f \mid \forall x, y \ f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2}\|x - y\|^2\right\}$.

How to study them? PEP!

Implicit definition of classes

Explicit definition of a class: parametrization.

Example:

- ◇ Quadratic functions $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

Implicit definition of a class: constraints.

Examples:

- ◇ Convex functions $\mathcal{F} = \{f \mid \forall x, y \ f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle\}$.
- ◇ Smooth functions $\mathcal{F} = \left\{f \mid \forall x, y \ f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2}\|x - y\|^2\right\}$.

How to study them? PEP!

$$\tau = \max_{f, x_0, x_1, x_\star} \frac{Perf(x_1)}{Init(x_0)}$$

$$\text{s.t. } f \in \mathcal{F} \qquad \text{Functional class}$$

Implicit definition of classes

Explicit definition of a class: parametrization.

Example:

- ◇ Quadratic functions $\mathcal{F} = \left\{ f \mid f(x) \triangleq \frac{1}{2}(x - x_\star)^T H(x - x_\star) + f_\star \mid \text{H is PSD.} \right\}$

Implicit definition of a class: constraints.

Examples:

- ◇ Convex functions $\mathcal{F} = \{f \mid \forall x, y \ f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle\}$.
- ◇ Smooth functions $\mathcal{F} = \left\{f \mid \forall x, y \ f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2}\|x - y\|^2\right\}$.

How to study them? PEP!

$$\tau = \max_{f, x_0, x_1, x_\star} \frac{Perf(x_1)}{Init(x_0)}$$

$$\text{s.t. } f \in \mathcal{F} \qquad \text{Functional class}$$

$$x_1 = \mathcal{A}(x_0, \nabla f(x_0)) \qquad \text{Algorithm}$$

$$\nabla f(x_\star) = 0 \qquad \text{Optimality of } x_\star$$

Derivation of a PEP for convergence rate of a gradient step

Derivation of a PEP for convergence rate of a gradient step

Toy example: What is the smallest τ such that:

$$\|x_1 - x_\star\|^2 \leqslant \tau \|x_0 - x_\star\|^2,$$

for all

- $\diamond$ L-smooth and μ-strongly convex function f (notation $f \in \mathcal{F}_{\mu,\mathrm{L}}$),
- $\diamond$ x_0, and x_1 generated by gradient step $x_1 = x_0 - \gamma \nabla f(x_0)$,
- $\diamond$ $x_\star = \underset{x}{\operatorname{argmin}}\ f(x)$?

Derivation of a PEP for convergence rate of a gradient step

Toy example: What is the smallest τ such that:

$$\|x_1 - x_\star\|^2 \leqslant \tau \|x_0 - x_\star\|^2,$$

for all

- L-smooth and μ-strongly convex function f (notation $f \in \mathcal{F}_{\mu,L}$),
- x_0, and x_1 generated by gradient step $x_1 = x_0 - \gamma \nabla f(x_0)$,
- $x_\star = \underset{x}{\operatorname{argmin}}\ f(x)$?

Final goal: optimize τ (as a function of the step-size, γ)

Derivation of a PEP for convergence rate of a gradient step

Toy example: What is the smallest τ such that:

$$\|x_1 - x_\star\|^2 \leqslant \tau \|x_0 - x_\star\|^2,$$

for all

- $\diamond$ L-smooth and μ-strongly convex function f (notation $f \in \mathcal{F}_{\mu,\mathrm{L}}$),
- $\diamond$ x_0, and x_1 generated by gradient step $x_1 = x_0 - \gamma \nabla f(x_0)$,
- $\diamond$ $x_\star = \underset{x}{\operatorname{argmin}}\ f(x)$?

Final goal: optimize τ (as a function of the step-size, γ)

First: let's compute τ!

Derivation of a PEP for convergence rate of a gradient step

Toy example: What is the smallest τ such that:

$$\|x_1 - x_\star\|^2 \leqslant \tau \|x_0 - x_\star\|^2,$$

for all

- $\diamond$ L-smooth and μ-strongly convex function f (notation $f \in \mathcal{F}_{\mu,\mathrm{L}}$),
- $\diamond$ x_0, and x_1 generated by gradient step $x_1 = x_0 - \gamma \nabla f(x_0)$,
- $\diamond$ $x_\star = \underset{x}{\operatorname{argmin}}\ f(x)$?

Final goal: optimize τ (as a function of the step-size, γ)

First: let's compute τ!

$$\tau = \max_{f, x_0, x_1, x_\star} \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2}$$

$$\begin{aligned} \text{s.t. } & f \in \mathcal{F}_{\mu,L} && \text{Functional class} \\ & x_1 = x_0 - \gamma \nabla f(x_0) && \text{Algorithm} \\ & \nabla f(x_\star) = 0 && \text{Optimality of } x_\star \end{aligned}$$

From infinite to finite dimensional problem

From infinite to finite dimensional problem

Performance estimation problem:

$$
\begin{aligned}
\max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\
\text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\
& x_1 = x_0 - \gamma \nabla f(x_0) \\
& \nabla f(x_\star) = 0.
\end{aligned}
$$

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.
Sampled version: f is only used at x_0 and $x_\star$.

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.

Sampled version: f is only used at x_0 and $x_\star$.

$$\begin{aligned} \max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad & \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \in \mathcal{F}_{\mu,L} \text{ such that } \begin{cases} f_i = f(x_i) & i = 0, \star \\ g_i = \nabla f(x_i) & i = 0, \star \end{cases} \\ & g_\star = 0. \end{aligned}$$

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.

Sampled version: f is only used at x_0 and $x_\star$.

$$\begin{aligned} \max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad & \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \in \mathcal{F}_{\mu,L} \text{ such that } \begin{cases} f_i = f(x_i) & i = 0, \star \\ g_i = \nabla f(x_i) & i = 0, \star \end{cases} \\ & g_\star = 0. \end{aligned}$$

$\hookrightarrow$ Variables: x_0, $x_\star$, g_0, $g_\star$, f_0, $f_\star$.

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.

Sampled version: f is only used at x_0 and $x_\star$.

$$\begin{aligned} \max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad & \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & \exists f \in \mathcal{F}_{\mu,L} \text{ such that } \begin{cases} f_i = f(x_i) & i = 0, \star \\ g_i = \nabla f(x_i) & i = 0, \star \end{cases} \\ & g_\star = 0. \end{aligned}$$

$\hookrightarrow$ Variables: x_0, $x_\star$, g_0, $g_\star$, f_0, $f_\star$.

Interpolation conditions:
$\exists f \in \mathcal{F}_{\mu,L}$ such that $\forall i, f(x_i) = f_i$, and $\nabla f(x_i) = g_i$ iif $\forall i, j$,

$$\begin{aligned} f_i \geq f_j + \langle g_j, x_i - x_j \rangle + \tfrac{1}{2L} \|g_i - g_j\|^2 \\ + \tfrac{\mu}{2(1-\mu/L)} \|x_i - x_j - \tfrac{1}{L}(g_i - g_j)\|^2. \end{aligned}$$

From infinite to finite dimensional problem

Performance estimation problem:

$$\begin{aligned} \max_{f, x_0, x_1, x_\star} \quad & \frac{\|x_1 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, \\ & x_1 = x_0 - \gamma \nabla f(x_0) \\ & \nabla f(x_\star) = 0. \end{aligned}$$

$\hookrightarrow$ Variables: f, x_0, x_1, $x_\star$.

Sampled version: f is only used at x_0 and $x_\star$.

$$\begin{aligned} \max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad & \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & \exists f \in \mathcal{F}_{\mu,L} \text{ such that } \begin{cases} f_i = f(x_i) & i = 0, \star \\ g_i = \nabla f(x_i) & i = 0, \star \end{cases} \\ & g_\star = 0. \end{aligned}$$

$\hookrightarrow$ Variables: x_0, $x_\star$, g_0, $g_\star$, f_0, $f_\star$.

Interpolation conditions:
$\exists f \in \mathcal{F}_{\mu,L}$ such that $\forall i, f(x_i) = f_i$, and $\nabla f(x_i) = g_i$ iif $\forall i, j$,

$$\begin{aligned} f_i \geq f_j + \langle g_j, x_i - x_j \rangle + \tfrac{1}{2L} \|g_i - g_j\|^2 \\ + \tfrac{\mu}{2(1-\mu/L)} \|x_i - x_j - \tfrac{1}{L}(g_i - g_j)\|^2. \end{aligned}$$

Reformulation:

$$\begin{aligned} \max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad & \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2} \\ \text{subject to} \quad & f_\star \geqslant f_0 + \langle g_0, x_\star - x_0 \rangle + \tfrac{1}{2L} \|g_\star - g_0\|^2 \\ & \quad + \tfrac{\mu}{2(1-\mu/L)} \|x_\star - x_0 - \tfrac{1}{L}(g_\star - g_0)\|^2 \\ & f_0 \geqslant f_\star + \langle g_\star, x_0 - x_\star \rangle + \tfrac{1}{2L} \|g_0 - g_\star\|^2 \\ & \quad + \tfrac{\mu}{2(1-\mu/L)} \|x_0 - x_\star - \tfrac{1}{L}(g_0 - g_\star)\|^2 \\ & g_\star = 0. \end{aligned}$$

Normalization invariance

Non convex objective:

$$\max_{\substack{x_0, x_\star \\ g_0, g_\star \\ f_0, f_\star}} \quad \frac{\|x_0 - \gamma g_0 - x_\star\|^2}{\|x_0 - x_\star\|^2}$$

$$\begin{aligned}
\text{subject to} \quad & f_\star \geqslant f_0 + \langle g_0, x_\star - x_0 \rangle + \tfrac{1}{2L}\|g_\star - g_0\|^2 \\
& \quad + \tfrac{\mu}{2(1-\mu/L)}\left\|x_\star - x_0 - \tfrac{1}{L}(g_\star - g_0)\right\|^2 \\
& f_0 \geqslant f_\star + \langle g_\star, x_0 - x_\star \rangle + \tfrac{1}{2L}\|g_0 - g_\star\|^2 \\
& \quad + \tfrac{\mu}{2(1-\mu/L)}\left\|x_0 - x_\star - \tfrac{1}{L}(g_0 - g_\star)\right\|^2 \\
& g_\star = 0.
\end{aligned}$$

Normalization invariance

Non convex objective:

$$\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \frac{\|x_0-\gamma g_0-x_\star\|^2}{\|x_0-x_\star\|^2}$$

subject to
$$f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2$$
$$f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2$$
$$g_\star = 0.$$

Homogeneity

$$\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \|x_0-\gamma g_0-x_\star\|^2$$

subject to
$$f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2$$
$$f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2$$
$$g_\star = 0$$
$$\|x_0-x_\star\|^2 \leq 1.$$

Semidefinite lifting

Quadratic constraints

$$\begin{aligned}
\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \quad & \|x_0-\gamma g_0-x_\star\|^2\\
\text{subject to} \quad & f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2\\
& \quad +\tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2\\
& f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2\\
& \quad +\tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2\\
& g_\star=0\\
& \|x_0-x_\star\|^2\leq 1.
\end{aligned}$$

Semidefinite lifting

Quadratic constraints

$$
\begin{aligned}
\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \quad & \|x_0-\gamma g_0-x_\star\|^2 \\
\text{subject to} \quad & f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2 \\
& \quad +\tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2 \\
& f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2 \\
& \quad +\tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2 \\
& g_\star=0 \\
& \|x_0-x_\star\|^2\leq 1.
\end{aligned}
$$

Homogeneity and Gram matrix

- ◇ Dependency on x and g in the objective and constraints all are quadratics.
  ↪ Introduce the new variables $G \succcurlyeq 0$ and F

$$
G=\begin{bmatrix}\|x_0-x_\star\|^2 & \langle g_0,x_0-x_\star\rangle\\ \langle g_0,x_0-x_\star\rangle & \|g_0\|^2\end{bmatrix},\qquad F=f_0-f_\star,
$$

Semidefinite lifting

Quadratic constraints

$$\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \quad \|x_0-\gamma g_0-x_\star\|^2$$

$$\text{subject to} \quad f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2$$

$$f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2$$

$$g_\star=0$$

$$\|x_0-x_\star\|^2\leq 1.$$

Homogeneity and Gram matrix

- ◇ Dependency on x and g in the objective and constraints all are quadratics.
  ↪ Introduce the new variables $G\succcurlyeq 0$ and F

$$G=\begin{bmatrix}\|x_0-x_\star\|^2 & \langle g_0,x_0-x_\star\rangle\\ \langle g_0,x_0-x_\star\rangle & \|g_0\|^2\end{bmatrix},\quad F=f_0-f_\star,$$

Semi-definite program (SDP) formulation

$$\max_{G,F} \quad G_{1,1}+\gamma^2G_{2,2}-2\gamma G_{1,2}$$

$$\text{subject to} \quad F+\tfrac{L\mu}{2(L-\mu)}G_{1,1}+\tfrac{1}{2(L-\mu)}G_{2,2}-\tfrac{L}{L-\mu}G_{1,2}\leqslant 0$$

$$-F+\tfrac{L\mu}{2(L-\mu)}G_{1,1}+\tfrac{1}{2(L-\mu)}G_{2,2}-\tfrac{\mu}{L-\mu}G_{1,2}\leqslant 0$$

$$G_{1,1}\leqslant 1$$

$$G\succcurlyeq 0,$$

Semidefinite lifting

Quadratic constraints

$$\max_{\substack{x_0,x_\star\\ g_0,g_\star\\ f_0,f_\star}} \ \|x_0-\gamma g_0-x_\star\|^2$$

$$\text{subject to} \quad f_\star \geqslant f_0+\langle g_0,x_\star-x_0\rangle+\tfrac{1}{2L}\|g_\star-g_0\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_\star-x_0-\tfrac{1}{L}(g_\star-g_0)\right\|^2$$

$$f_0 \geqslant f_\star+\langle g_\star,x_0-x_\star\rangle+\tfrac{1}{2L}\|g_0-g_\star\|^2 + \tfrac{\mu}{2(1-\mu/L)}\left\|x_0-x_\star-\tfrac{1}{L}(g_0-g_\star)\right\|^2$$

$$g_\star=0$$

$$\|x_0-x_\star\|^2\leq 1.$$

Homogeneity and Gram matrix

- ◇ Dependency on x and g in the objective and constraints all are quadratics.
  ↪ Introduce the new variables $G \succcurlyeq 0$ and F

$$G=\begin{bmatrix}\|x_0-x_\star\|^2 & \langle g_0,x_0-x_\star\rangle\\ \langle g_0,x_0-x_\star\rangle & \|g_0\|^2\end{bmatrix},\quad F=f_0-f_\star,$$

Semi-definite program (SDP) formulation

$$\max_{G,F} \ G_{1,1}+\gamma^2 G_{2,2}-2\gamma G_{1,2}$$

$$\text{subject to} \quad F+\tfrac{L\mu}{2(L-\mu)}G_{1,1}+\tfrac{1}{2(L-\mu)}G_{2,2}-\tfrac{L}{L-\mu}G_{1,2}\leqslant 0$$

$$-F+\tfrac{L\mu}{2(L-\mu)}G_{1,1}+\tfrac{1}{2(L-\mu)}G_{2,2}-\tfrac{\mu}{L-\mu}G_{1,2}\leqslant 0$$

$$G_{1,1}\leqslant 1$$

$$G\succcurlyeq 0,$$

Convex problem!!!

Other usages from PEPit

- 1. Unconstrained convex minimization
  - 1.1. Gradient descent
  - 1.2. Subgradient method
  - 1.3. Subgradient method under restricted secant inequality and error bound
  - 1.4. Gradient descent with exact line search
  - 1.5. Conjugate gradient
  - 1.6. Heavy Ball momentum
  - 1.7. Accelerated gradient for convex objective
  - 1.8. Accelerated gradient for strongly convex objective
  - 1.9. Optimized gradient
  - 1.10. Optimized gradient for gradient
  - 1.11. Robust momentum
  - 1.12. Triple momentum
  - 1.13. Information theoretic exact method
  - 1.14. Proximal point
  - 1.15. Accelerated proximal point
  - 1.16. Inexact gradient descent
  - 1.17. Inexact gradient descent with exact line search
  - 1.18. Inexact accelerated gradient
  - 1.19. Epsilon-subgradient method
  - 1.20. Gradient descent for quadratically upper bounded convex objective
  - 1.21. Gradient descent with decreasing step sizes for quadratically upper bounded convex objective
  - 1.22. Conjugate gradient for quadratically upper bounded convex objective
  - 1.23. Heavy Ball momentum for quadratically upper bounded convex objective
- 2. Composite convex minimization
  - 2.1. Proximal gradient
  - 2.2. Accelerated proximal gradient
  - 2.3. Bregman proximal point
  - 2.4. Douglas Rachford splitting
  - 2.5. Douglas Rachford splitting contraction
  - 2.6. Accelerated Douglas Rachford splitting
  - 2.7. Frank Wolfe
  - 2.8. Improved interior method
  - 2.9. No Lips in function value
  - 2.10. No Lips in Bregman divergence
  - 2.11. Three operator splitting
- 3. Non-convex optimization
  - 3.1. Gradient Descent
  - 3.2. No Lips 1
  - 3.3. No Lips 2
- 4. Stochastic and randomized convex minimization
  - 4.1. Stochastic gradient descent
  - 4.2. Stochastic gradient descent in overparametrized setting
  - 4.3. SAGA
  - 4.4. Point SAGA
  - 4.5. Randomized coordinate descent for smooth strongly convex functions
  - 4.6. Randomized coordinate descent for smooth convex functions
- 5. Monotone inclusions and variational inequalities
  - 5.1. Proximal point
  - 5.2. Accelerated proximal point
  - 5.3. Optimal Strongly-monotone Proximal Point
  - 5.4. Douglas Rachford Splitting
  - 5.5. Three operator splitting
  - 5.6. Optimistic gradient
  - 5.7. Past extragradient
- 6. Fixed point
  - 6.1. Halpern iteration
  - 6.2. Optimal Contractive Halpern iteration
  - 6.3. Krasnoselskii-Mann with constant step-sizes
  - 6.4. Krasnoselskii-Mann with increasing step-sizes
- 7. Potential functions
  - 7.1. Gradient descent Lyapunov 1
  - 7.2. Gradient descent Lyapunov 2
  - 7.3. Accelerated gradient method
- 8. Inexact proximal methods
  - 8.1. Accelerated inexact forward backward
  - 8.2. Partially inexact Douglas Rachford splitting
  - 8.3. Relatively inexact proximal point
- 9. Adaptive methods
  - 9.1. Polyak steps in distance to optimum
  - 9.2. Polyak steps in function value
- 10. Low dimensional worst-cases scenarios
  - 10.1. Inexact gradient
  - 10.2. Non-convex gradient descent
  - 10.3. Optimized gradient
  - 10.4. Frank Wolfe
  - 10.5. Proximal point
  - 10.6. Halpern iteration
  - 10.7. Alternate projections
  - 10.8. Averaged projections
  - 10.9. Dykstra
- 11. Continuous-time models
  - 11.1. Gradient flow for strongly convex functions
  - 11.2. Gradient flow for convex functions
  - 11.3. Accelerated gradient flow for strongly convex functions
  - 11.4. Accelerated gradient flow for convex functions
- 12. Tutorials
  - 12.1. Contraction rate of gradient descent

Key ingredients

- ◇ Algorithm: first order update;
- ◇ Class of functions: interpolation constraints expressible linearly in F and G;
- ◇ Performance metrics: expressible linearly in F and G.

Avoiding semidefinite programming modeling steps?

Performance Estimation Toolbox (**PESTO**) available on

PerformanceEstimation/Performance-Estimation-Toolbox

Python version (**PEPit**) available on PyPI.

PerformanceEstimation/PEPit

- ◇ Contain more than 50 examples.
- ◇ Contain several classical classes of functions.

Learning to PEP.

PerformanceEstimation/Learning-Performance-Estimation

- ◇ Contains theoretical exercises as well as code exercises.

From explicit to implicit algorithms

- ◇ So far algorithms were defined through an explicit update rule.
- ◇ Some update could be characterized by an inequality constraint.
- ◇ From now, we consider the general form:

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{rll} & \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 \\ \forall k, & \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 \\ \forall I, & \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 \end{array}\right. \end{array}\right. \qquad \text{(PEP-primal)}$$

From primal to dual problem

- Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{lll} \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 & : \tau \\ \forall k,\ \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 & : \lambda_{\mathcal{F}}^{(k)} \\ \forall I,\ \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 & : \lambda_{\mathcal{A}}^{(I)} \end{array}\right. \end{array}\right.$$

From primal to dual problem

- Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{lll} \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 & : \tau \\ \forall k,\ \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 & : \lambda_{\mathcal{F}}^{(k)} \\ \forall I,\ \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 & : \lambda_{\mathcal{A}}^{(I)} \end{array}\right. \end{array}\right.$$

- Lagrangian is

$$\begin{aligned} \mathcal{L} \triangleq\ & \langle F, v_P\rangle + \langle G, M_P\rangle - \tau\left[\langle F, v_I\rangle + \langle G, M_I\rangle - R^2\right] \\ & - \sum_k \lambda_{\mathcal{F}}^{(k)}\left[\left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle\right] \\ & - \sum_I \lambda_{\mathcal{A}}^{(I)}\left[\left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle\right] \end{aligned}$$

From primal to dual problem

◇ Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{lll} \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 & : \tau \\ \forall k,\ \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 & : \lambda_{\mathcal{F}}^{(k)} \\ \forall I,\ \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 & : \lambda_{\mathcal{A}}^{(I)} \end{array}\right. \end{array}\right.$$

◇ Lagrangian is

$$\begin{aligned} \mathcal{L} \triangleq\ & \langle F, v_P\rangle + \langle G, M_P\rangle - \tau\left[\langle F, v_I\rangle + \langle G, M_I\rangle - R^2\right] \\ & - \sum_k \lambda_{\mathcal{F}}^{(k)}\left[\left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle\right] \\ & - \sum_I \lambda_{\mathcal{A}}^{(I)}\left[\left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle\right] \\ =& \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} v_{\mathcal{A}}^{(I)}\right\rangle \\ & + \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} M_{\mathcal{A}}^{(I)}\right\rangle \end{aligned}$$

From primal to dual problem

- Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{rll} \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 & : \tau \\ \forall k,\ \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 & : \lambda_{\mathcal{F}}^{(k)} \\ \forall I,\ \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 & : \lambda_{\mathcal{A}}^{(I)} \end{array}\right. \end{array}\right.$$

- Lagrangian is

$$\begin{aligned} \mathcal{L} =& \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} v_{\mathcal{A}}^{(I)} \right\rangle \\ &+ \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} M_{\mathcal{A}}^{(I)} \right\rangle \end{aligned}$$

From primal to dual problem

◇ Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{rll} & \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 \quad : \tau \\ \forall k, & \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 \quad : \lambda_{\mathcal{F}}^{(k)} \\ \forall I, & \left\langle F, v_{\mathcal{A}}^{(I)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(I)}\right\rangle & \leq 0 \quad : \lambda_{\mathcal{A}}^{(I)} \end{array}\right. \end{array}\right.$$

◇ Lagrangian is

$$\begin{aligned} \mathcal{L} =& \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} v_{\mathcal{A}}^{(I)} \right\rangle \\ &+ \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} M_{\mathcal{A}}^{(I)} \right\rangle \end{aligned}$$

◇ Dual problem is

$$\left|\begin{array}{ll} \underset{\tau,\lambda_{\mathcal{F}}^{(k)},\lambda_{\mathcal{A}}^{(I)}\geq 0}{\text{minimize}} & \tau R^2 \\ \text{s.t.} & \left\{\begin{array}{ll} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} v_{\mathcal{A}}^{(I)} & = 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_I \lambda_{\mathcal{A}}^{(I)} M_{\mathcal{A}}^{(I)} & \preceq 0 \end{array}\right. \end{array}\right. \qquad \text{(PEP-dual)}$$

From primal to dual problem

◇ Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{rll} & \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 \ : \tau \\ \forall k, & \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 \ : \lambda_{\mathcal{F}}^{(k)} \\ \forall l, & \left\langle F, v_{\mathcal{A}}^{(l)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(l)}\right\rangle & \leq 0 \ : \lambda_{\mathcal{A}}^{(l)} \end{array}\right. \end{array}\right.$$

◇ Lagrangian is

$$\begin{aligned} \mathcal{L} =& \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle \\ &+ \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle \end{aligned}$$

◇ Dual problem is

$$\left|\begin{array}{ll} \underset{\tau,\lambda_{\mathcal{F}}^{(k)},\lambda_{\mathcal{A}}^{(l)}\geq 0}{\text{minimize}} & \tau R^2 \\ \text{s.t.} & \left\{\begin{array}{lll} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} & = & 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} & \preceq & 0 \end{array}\right. \end{array}\right. \qquad \text{(PEP-dual)}$$

◇ Any *primal* feasible point leads to a *lower* bound on the worst-case convergence rate. The *primal* variables provide adversarial objective.

From primal to dual problem

◇ Primal problem is

$$\left|\begin{array}{ll} \underset{F,G\succeq 0}{\text{maximize}} & \langle F, v_P\rangle + \langle G, M_P\rangle \\ \text{subject to} & \left\{\begin{array}{rll} & \langle F, v_I\rangle + \langle G, M_I\rangle & \leq R^2 \quad : \tau \\ \forall k, & \left\langle F, v_{\mathcal{F}}^{(k)}\right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)}\right\rangle & \leq 0 \quad : \lambda_{\mathcal{F}}^{(k)} \\ \forall l, & \left\langle F, v_{\mathcal{A}}^{(l)}\right\rangle + \left\langle G, M_{\mathcal{A}}^{(l)}\right\rangle & \leq 0 \quad : \lambda_{\mathcal{A}}^{(l)} \end{array}\right. \end{array}\right.$$

◇ Lagrangian is

$$\begin{aligned} \mathcal{L} =& \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle \\ &+ \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle \end{aligned}$$

◇ Dual problem is

$$\left|\begin{array}{ll} \underset{\tau,\lambda_{\mathcal{F}}^{(k)},\lambda_{\mathcal{A}}^{(l)}\geq 0}{\text{minimize}} & \tau R^2 \\ \text{s.t.} & \left\{\begin{array}{rl} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} & = 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} & \preceq 0 \end{array}\right. \end{array}\right. \qquad \text{(PEP-dual)}$$

◇ Any *primal* feasible point leads to a *lower* bound on the worst-case convergence rate. The *primal* variables provide adversarial objective.

◇ Any *dual* feasible point leads to an *upper* bound on the worst-case convergence rate. The *dual* variables provide the proof of convergence.

Deriving a proof of convergence

- Lagrangian is

$$\mathcal{L} = \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle$$

Deriving a proof of convergence

- Lagrangian is

$$\mathcal{L} = \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle$$

- Dual problem is

$$\left| \begin{array}{l} \text{minimize} \quad \tau R^2 \\ \tau, \lambda_{\mathcal{F}}^{(k)}, \lambda_{\mathcal{A}}^{(l)} \geq 0 \\ \text{s.t.} \begin{cases} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} = 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \preceq 0 \end{cases} \end{array} \right. \qquad \text{(PEP-dual)}$$

- Lagrangian $\leq$ Dual:

Deriving a proof of convergence

- Lagrangian is

$$\mathcal{L} = \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle$$

- Dual problem is

$$\left| \begin{array}{l} \underset{\tau, \lambda_{\mathcal{F}}^{(k)}, \lambda_{\mathcal{A}}^{(l)} \geq 0}{\text{minimize}} \quad \tau R^2 \\ \text{s.t.} \begin{cases} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} = 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \preceq 0 \end{cases} \end{array} \right. \qquad \text{(PEP-dual)}$$

- Lagrangian $\leq$ Dual: for any feasible primal F, G and feasible dual $\tau, (\lambda_{\mathcal{F}}^{(k)})_k, (\lambda_{\mathcal{A}}^{(l)})_l$,

$$\underbrace{\langle F, v_P \rangle + \langle G, M_P \rangle}_{\text{Performance metric}} - \tau \underbrace{[\langle F, v_I \rangle + \langle G, M_I \rangle]}_{\text{Initialization}} \leq \sum_k \lambda_{\mathcal{F}}^{(k)} \underbrace{\left[\left\langle F, v_{\mathcal{F}}^{(k)} \right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)} \right\rangle \right]}_{\text{Class constraint}} + \sum_l \lambda_{\mathcal{A}}^{(l)} \underbrace{\left[\left\langle F, v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_{\mathcal{A}}^{(l)} \right\rangle \right]}_{\text{Algorithm constraint}} \leq 0.$$

Deriving a proof of convergence

- Lagrangian is

$$\mathcal{L} = \tau R^2 + \left\langle F, v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \right\rangle$$

- Dual problem is

$$\left| \begin{array}{l} \underset{\tau, \lambda_{\mathcal{F}}^{(k)}, \lambda_{\mathcal{A}}^{(l)} \geq 0}{\text{minimize}} \quad \tau R^2 \\ \text{s.t.} \begin{cases} v_P - \tau v_I - \sum_k \lambda_{\mathcal{F}}^{(k)} v_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} v_{\mathcal{A}}^{(l)} = 0 \\ M_P - \tau M_I - \sum_k \lambda_{\mathcal{F}}^{(k)} M_{\mathcal{F}}^{(k)} - \sum_l \lambda_{\mathcal{A}}^{(l)} M_{\mathcal{A}}^{(l)} \preceq 0 \end{cases} \end{array} \right. \qquad \text{(PEP-dual)}$$

- Lagrangian $\leq$ Dual: for any feasible primal F, G and feasible dual $\tau, (\lambda_{\mathcal{F}}^{(k)})_k, (\lambda_{\mathcal{A}}^{(l)})_l$,

$$\underbrace{\langle F, v_P \rangle + \langle G, M_P \rangle}_{\text{Performance metric}} - \tau \underbrace{[\langle F, v_I \rangle + \langle G, M_I \rangle]}_{\text{Initialization}} \leq \sum_k \lambda_{\mathcal{F}}^{(k)} \underbrace{\left[\left\langle F, v_{\mathcal{F}}^{(k)} \right\rangle + \left\langle G, M_{\mathcal{F}}^{(k)} \right\rangle \right]}_{\text{Class constraint}} + \sum_l \lambda_{\mathcal{A}}^{(l)} \underbrace{\left[\left\langle F, v_{\mathcal{A}}^{(l)} \right\rangle + \left\langle G, M_{\mathcal{A}}^{(l)} \right\rangle \right]}_{\text{Algorithm constraint}} \leq 0.$$

- Performance metric - τ initialization $\leq \sum_i \lambda_i \text{Constraint}_i \leq 0$.

Some insights

- The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.

Some insights

- ◇ The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.
- ◇ Interpolation constraints are the right constraints to be considered when studying a class.

Some insights

- ◇ The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.
- ◇ Interpolation constraints are the right constraints to be considered when studying a class.
- ◇ Dual zeros:

Some insights

- ◇ The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.
- ◇ Interpolation constraints are the right constraints to be considered when studying a class.
- ◇ Dual zeros:
  - Each $\lambda = 0$ witnesses the non use of the corresponding inequality.

Some insights

- ◇ The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.
- ◇ Interpolation constraints are the right constraints to be considered when studying a class.
- ◇ Dual zeros:
  - Each $\lambda = 0$ witnesses the non use of the corresponding inequality.
  - By strong duality, the proof is tight for a function. We therefore recover the slackness condition: the corresponding inequality must be an equality.

Some insights

- ◇ The proof of worst-case guarantee simply consists in a positively linear combination of the constraints.
- ◇ Interpolation constraints are the right constraints to be considered when studying a class.
- ◇ Dual zeros:
  - Each $\lambda = 0$ witnesses the non use of the corresponding inequality.
  - By strong duality, the proof is tight for a function. We therefore recover the slackness condition: the corresponding inequality must be an equality.
- ◇ Obtaining a guarantee over the minimum of performance metrics is equivalent to obtaining one on a certain average of those.

Natural performance metric

- $\frac{1}{\tau}$Performance metric - initialization $\leq \sum_i \lambda_i \text{Constraint}_i \leq 0$ might be loose. There sometimes remains a residual.

Natural performance metric

- $\frac{1}{\tau}$Performance metric - initialization $\leq \sum_i \lambda_i \text{Constraint}_i \leq 0$ might be loose. There sometimes remains a residual.
- We can tighten the proof by considering

$$V_t = \underbrace{[\langle F, v_I \rangle + \langle G, M_I \rangle]}_{\text{Initialization}} + \sum_{\substack{j \mid \text{only involves} \\ \text{values observed} \\ \text{before step } t}} \lambda^{(j)} \underbrace{\left[\left\langle F, v^{(j)} \right\rangle + \left\langle G, M^{(j)} \right\rangle\right]}_{\text{Constraint}}. \quad (1)$$

Natural performance metric

- $\frac{1}{\tau}$Performance metric - initialization $\leq \sum_i \lambda_i$Constraint$_i \leq 0$ might be loose. There sometimes remains a residual.
- We can tighten the proof by considering

$$V_t = \underbrace{[\langle F, v_I \rangle + \langle G, M_I \rangle]}_{\text{Initialization}} + \sum_{\substack{j \mid \text{only involves} \\ \text{values observed} \\ \text{before step } t}} \lambda^{(j)} \underbrace{\left[\left\langle F, v^{(j)} \right\rangle + \left\langle G, M^{(j)} \right\rangle\right]}_{\text{Constraint}}. \qquad (1)$$

- Example on NAG:

$$\lambda_{t+1} = \frac{1}{2} + \sqrt{\frac{1}{4} + \lambda_t^2}, \qquad y_t = x_t + \frac{\lambda_t - 1}{\lambda_{t+1}}(x_t - x_{t-1}), \qquad x_{t+1} = y_t - \frac{1}{L}\nabla f(y_t).$$

$$V_t = \lambda_t^2(f_t - f_\star) + \frac{L}{2}\|\lambda_t(x_t - x_\star) + (1 - \lambda_t)(x_{t-1} - x_\star)\|^2$$
$$+ \frac{1}{2L}\sum_{s=1}^{t-1}[\lambda_{s+1}^2\|\nabla f(x_{s+1})\|^2 + \lambda_{s+1}\|\nabla f(y_s)\|\lambda_s^2\|\nabla f(y_s) - \nabla f(x_s)\|^2]$$

Extensions of a certificate

- Fewer class constraints allows extension to larger class,

Extensions of a certificate

- ◇ Fewer class constraints allows extension to larger class,
- ◇ Fewer algorithm constraints allows extension to new algorithms,

Extensions of a certificate

- ◇ Fewer class constraints allows extension to larger class,
- ◇ Fewer algorithm constraints allows extension to new algorithms,
- ◇ Fewer class constraints allows extension to new algorithms.

Extensions of a certificate

- ◇ Fewer class constraints allows extension to larger class,
- ◇ Fewer algorithm constraints allows extension to new algorithms,
- ◇ Fewer class constraints allows extension to new algorithms.
  Example of Backtracking line-search [C. Park et al., 2021]:

Extensions of a certificate

- ◇ Fewer class constraints allows extension to larger class,
- ◇ Fewer algorithm constraints allows extension to new algorithms,
- ◇ Fewer class constraints allows extension to new algorithms.
  Example of Backtracking line-search [C. Park et al., 2021]:
  - $f_{t+1} \leq f_t - \frac{1}{2L}\|g_t\|^2$ is verifiable,

Extensions of a certificate

- ◇ Fewer class constraints allows extension to larger class,
- ◇ Fewer algorithm constraints allows extension to new algorithms,
- ◇ Fewer class constraints allows extension to new algorithms.
  Example of Backtracking line-search [C. Park et al., 2021]:
  - $f_{t+1} \leq f_t - \frac{1}{2L}\|g_t\|^2$ is verifiable,
  - $f_{t+1} \leq f_t + < g_{t+1}, x_{t+1} - x_t > - \frac{1}{2L}\|g_{t+1} - g_t\|^2$ is not.

Greedy first order method (GFOM)

◇ Update:

$$x_{t+1} = \underset{x \in x_{\mathbf{0}} + \mathbf{span}(\nabla f(x_{\mathbf{0}}), \nabla f(x_{\mathbf{1}}), \cdots, \nabla f(x_t))}{\arg\min} f(x) \qquad \text{(GFOM)}$$

Greedy first order method (GFOM)

◇ Update:

$$x_{t+1} = \underset{x \in x_{\mathbf{0}} + \mathbf{span}(\nabla f(x_{\mathbf{0}}), \nabla f(x_{\mathbf{1}}), \cdots, \nabla f(x_t))}{\arg\min} f(x) \qquad \text{(GFOM)}$$

◇ Additional constraints implied by GFOM:

$$\langle g_i, g_j \rangle = 0 \qquad \text{for all } j < i$$
$$\langle g_i, x_j - x_0 \rangle = 0 \qquad \text{for all } j \leq i$$

Proof of worst case guarantee for GFOM

◇ Structure of the certificate proof:

$$\begin{aligned}\text{Performance metric} - \tau\text{initialization} \leq & \sum_i \lambda_i \underbrace{\text{Constraint}_i}_{\leq 0} \\ & + \sum_i \sum_{j<i} \beta_{i,j} \underbrace{\langle g_i, g_j \rangle}_{=0} \\ & + \sum_i \sum_{j \leq i} \gamma_{i,j} \underbrace{\langle g_i, x_j - x_0 \rangle}_{=0}\end{aligned}$$

Proof of worst case guarantee for GFOM

◇ Structure of the certificate proof:

$$\begin{aligned}\text{Performance metric} - \tau\text{initialization} \leq & \sum_i \lambda_i \underbrace{\text{Constraint}_i}_{\leq 0} \\ & + \sum_i \sum_{j<i} \beta_{i,j} \underbrace{\langle g_i, g_j \rangle}_{=0} \\ & + \sum_i \sum_{j\leq i} \gamma_{i,j} \underbrace{\langle g_i, x_j - x_0 \rangle}_{=0}\end{aligned}$$

◇ For all i,

$$\langle g_i, \sum_{j<i} \beta_{i,j} g_j + \sum_{j\leq i} \gamma_{i,j}(x_j - x_0) \rangle = 0$$

would be enough to obtain the same worst case guarantee.

Automating search for Lyapunov functions

- ◇ We look for proof under the form $\boxed{V_{t+1} - \tau V_t \leq \sum_i \lambda_i \text{Constraint}_i \leq 0}$ where V is a linear combination of a finite family of expressions of the iterates sequence. Fixing τ, this problem is an SDP.

Automating search for Lyapunov functions

- ◇ We look for proof under the form $\boxed{V_{t+1} - \tau V_t \leq \sum_i \lambda_i \text{Constraint}_i \leq 0}$ where V is a linear combination of a finite family of expressions of the iterates sequence. Fixing τ, this problem is an SDP.
- ◇ In [A. Taylor et al., 2018], authors show that V does not necessarily need to be a positive form of the iterates.

Automating search for Lyapunov functions

- ◇ We look for proof under the form $\boxed{V_{t+1} - \tau V_t \leq \sum_i \lambda_i \text{Constraint}_i \leq 0}$ where V is a linear combination of a finite family of expressions of the iterates sequence. Fixing τ, this problem is an SDP.
- ◇ In [A. Taylor et al., 2018], authors show that V does not necessarily need to be a positive form of the iterates.
- ◇ But it must be positive on the path of iterates, i.e.

$$\boxed{0 - V_{t+1} \leq \sum_i \widetilde{\lambda}_i \text{Constraint}_i \leq 0}$$

Automating search for Lyapunov functions

- ◇ We look for proof under the form $\boxed{V_{t+1} - \tau V_t \leq \sum_i \lambda_i \text{Constraint}_i \leq 0}$ where V is a linear combination of a finite family of expressions of the iterates sequence. Fixing τ, this problem is an SDP.
- ◇ In [A. Taylor et al., 2018], authors show that V does not necessarily need to be a positive form of the iterates.
- ◇ But it must be positive on the path of iterates, i.e.

  $\boxed{0 - V_{t+1} \leq \sum_i \widetilde{\lambda}_i \text{Constraint}_i \leq 0}$
- ◇ Bonus: this problem is defined as a feasibility problem, leaving free the objective to minimize!

Automating search for Lyapunov functions

- ◇ We look for proof under the form $\boxed{V_{t+1} - \tau V_t \leq \sum_i \lambda_i \text{Constraint}_i \leq 0}$ where V is a linear combination of a finite family of expressions of the iterates sequence. Fixing τ, this problem is an SDP.
- ◇ In [A. Taylor et al., 2018], authors show that V does not necessarily need to be a positive form of the iterates.
- ◇ But it must be positive on the path of iterates, i.e.

  $\boxed{0 - V_{t+1} \leq \sum_i \widetilde{\lambda}_i \text{Constraint}_i \leq 0}$
- ◇ Bonus: this problem is defined as a feasibility problem, leaving free the objective to minimize!
- ◇ Bonus 2: the search for Lyapunov can be combined with the SSEP design procedure.

Conclusion

- ◇ Finding the worst case guarantee of a First-order method can be solved through convex optimization.
  - Requires 3 ingredients: Interpolation, Homogenization and SDP lifting.
  - Can be applied for various function class, algorithms and performance metrics.
- ◇ Modelization is heavy $\Rightarrow$ PEPit package.
- ◇ Dual view enables to understand the systematic structure of convergence proofs.
  - Always linear sum of inequalities.
  - From a function class point of view, only interpolation conditions matter.
  - We can design algorithms.
  - Search for Lyapunov can be automatized as well and combined with the algorithm design.
- ◇ Given a class, running SSEP with Lyapunov search provides an algorithm with its rate, and a Lyapunov based proof of convergence. All the information are stored in the dual variables.
- ◇ Finally, the zeros tell us which inequalities must be cancel to prove tightness of the result.

In summary, many insights are hidden in the dual variables!

Application / Open problem

Using [N. Bousselmi et al., 2023]'s interpolation constraints for the classes of quadratic functions, determine a PEP-based proof of guarantee of Heavy-ball.

Application / Open problem

Using [N. Bousselmi et al., 2023]'s interpolation constraints for the classes of quadratic functions, determine a PEP-based proof of guarantee of Heavy-ball.

By removing useless class constraints, can we identify a larger class that quadratic functions on which HB accelerates?

Thanks!

Questions?