Tight analyses of first-order methods using error feedback

EUROPT 2025

Daniel Berg Thomsen, Adrien Taylor, Aymeric Dieuleveut

Outline of talk

Part I: What we did

- Intro to distributed optimization
- The methods analyzed
- The results

$\rightarrow$ Aymeric

Outline of talk

Part I: What we did

- Intro to distributed optimization
- The methods analyzed
- The results

$\rightarrow$ Aymeric

Part II: How we did it

- Lyapunov functions: our definition
- Defining the problem
- A bag of tricks

$\rightarrow$ Daniel

Outline of talk

Part I: What we did

- Intro to distributed optimization
- The methods analyzed
- The results

$\to$ Aymeric

Part II: How we did it

- Lyapunov functions: our definition
- Defining the problem
- A bag of tricks

$\to$ Daniel

These two parts are *exactly* as interesting.

Part I: Distributed optimization

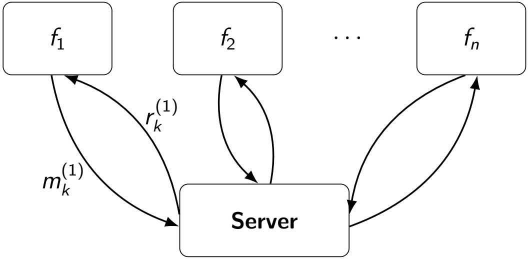
Finite sum minimization:

$$f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

Part I: Distributed optimization

Finite sum minimization:

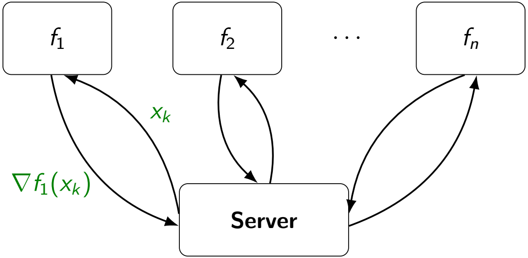
$$f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$



Part I: Distributed GD

Finite sum minimization:

$$f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

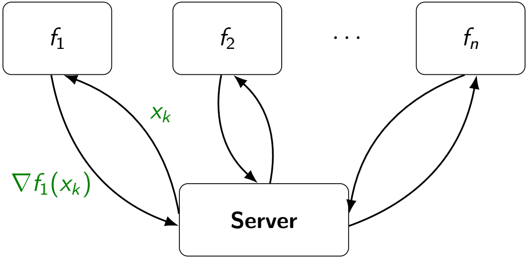


$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} \nabla f_i(x_k)$$

Part I: Distributed GD

Finite sum minimization:

$$f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$



$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} \nabla f_i(x_k)$$

$\rightarrow$ Communication bottleneck.

Part I: Compressed GD (CGD)

$\rightarrow$ **First solution:** compression

Part I: Compressed GD (CGD)

$\rightarrow$ **First solution:** compression

$\mathcal{C}$ compression operator, e.g.

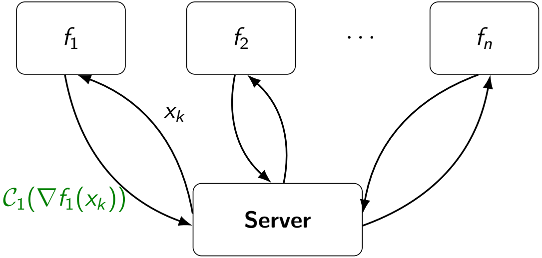
1. Top1 coordinate
2. Quantization operator

Part I: Compressed GD (CGD)

$\rightarrow$ **First solution:** compression

$\mathcal{C}$ compression operator, e.g.

1. Top1 coordinate
2. Quantization operator



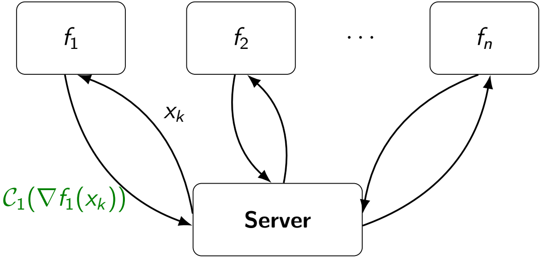
$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} \mathcal{C}_i(\nabla f_i(x_k))$$

Part I: Compressed GD (CGD)

$\rightarrow$ **First solution:** compression

$\mathcal{C}$ compression operator, e.g.

1. Top1 coordinate
2. Quantization operator



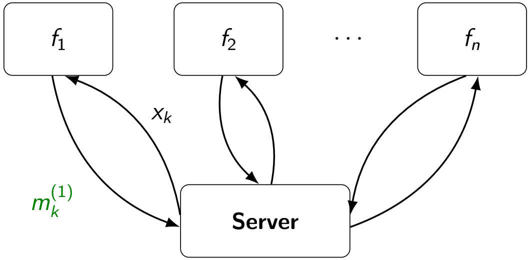
$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} \mathcal{C}_i(\nabla f_i(x_k))$$

$\rightsquigarrow$ Gauss Southwell - top Coordinate Gradient Descent.

Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".



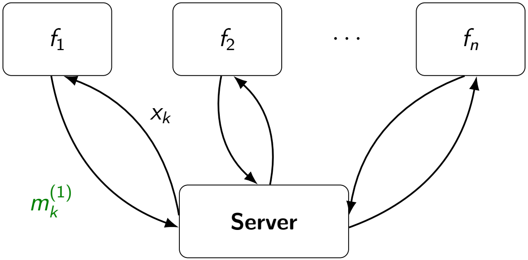
$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}$$

Part I: Classic error feedback (EF)

→ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".

$$m_0^{(i)} \quad = \quad \mathcal{C}_i(\eta \nabla f_i(x_0))$$



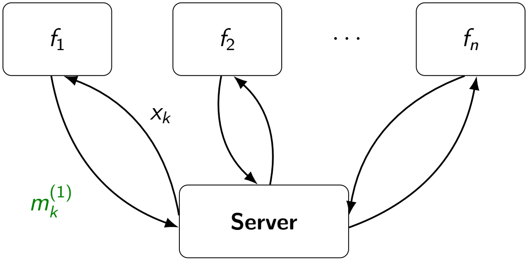
$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}$$

Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".

$$
\begin{aligned}
m_0^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) \\
e_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) - \eta \nabla f_i(x_0)
\end{aligned}
$$



$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}$$

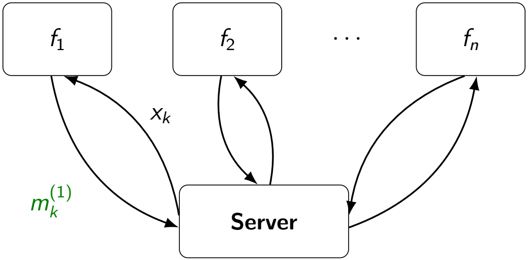
Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".

$$
\begin{aligned}
m_0^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) \\
e_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) - \eta \nabla f_i(x_0)
\end{aligned}
$$

$$
m_1^{(i)} = \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)})
$$



$$
x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}
$$

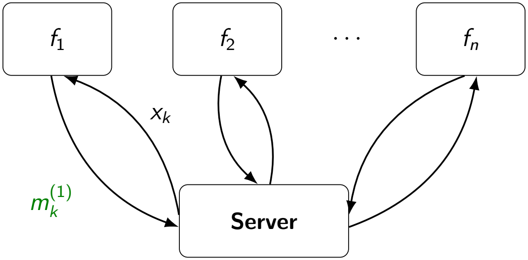
Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".

$$m_0^{(i)} = \mathcal{C}_i(\eta\nabla f_i(x_0))$$
$$e_1^{(i)} = \mathcal{C}_i(\eta\nabla f_i(x_0)) - \eta\nabla f_i(x_1)$$

$$m_1^{(i)} = \mathcal{C}_i(\eta\nabla f_i(x_1) + e_1^{(i)})$$
$$e_2^{(i)} =$$



$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}$$

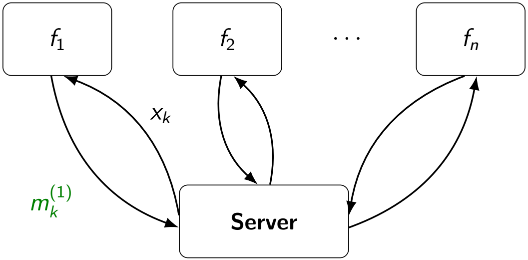
Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".

$$
\begin{aligned}
m_0^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) \\
e_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) - \eta \nabla f_i(x_k)
\end{aligned}
$$

$$
\begin{aligned}
m_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) \\
e_2^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) - \eta \nabla f_i(x_1) + e_1^{(i)}
\end{aligned}
$$

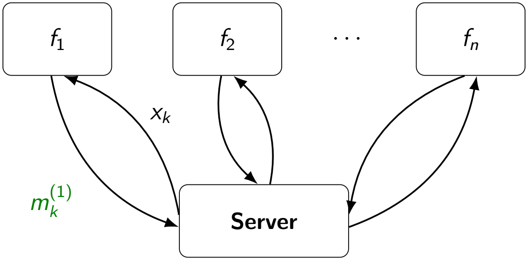


$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}$$

Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".



$$
\begin{aligned}
m_0^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) \\
e_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) - \eta \nabla f_i(x)
\end{aligned}
$$

$$
\begin{aligned}
m_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) \\
e_2^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) - \eta \nabla f_i(x_1) + e_1^{(i)}
\end{aligned}
$$

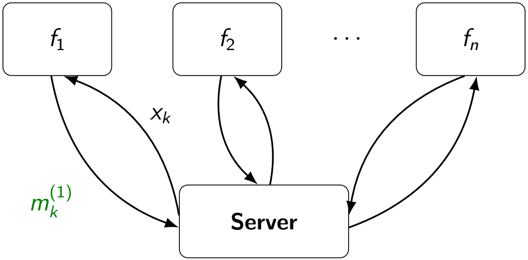
$$
\begin{aligned}
m_k^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_k) + e_k^{(i)}) \\
e_{k+1}^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_k) + e_k^{(i)}) - \eta \nabla f_i(x) + e_k^{(i)}
\end{aligned}
$$

$$
x_{k+1} = x_k - \eta \sum_{i=1}^{n} m_k^{(i)}
$$

Part I: Classic error feedback (EF)

$\rightarrow$ **2nd solut.:** Error feedback (EF)

Motivation: keep track of the mistakes made, and "retro-propagate them".



$$
\begin{aligned}
m_0^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) \\
e_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_0)) - \eta \nabla f_i(x)
\end{aligned}
$$

$$
\begin{aligned}
m_1^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) \\
e_2^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_1) + e_1^{(i)}) - \eta \nabla f_i(x_1) + e_1^{(i)}
\end{aligned}
$$

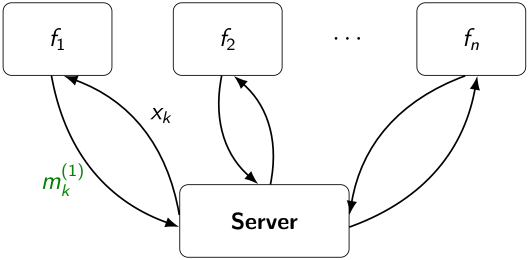
$$
\begin{aligned}
m_k^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_k) + e_k^{(i)}) \\
e_{k+1}^{(i)} &= \mathcal{C}_i(\eta \nabla f_i(x_k) + e_k^{(i)}) - \eta \nabla f_i(x) + e_k^{(i)}
\end{aligned}
$$

$$
\begin{aligned}
x_{k+1} &= x_k - \eta \sum_{i=1}^{n} m_k^{(i)} \\
&= x_k - \eta \sum_{i=1}^{n} \mathcal{C}_i(e_k^{(i)} + \nabla f_i(x_k))
\end{aligned}
$$

Part I: Modern error feedback (EF^{21})

$\rightarrow$ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression

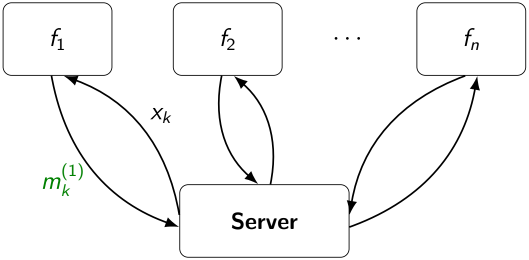


Part I: Modern error feedback (EF^{21})

$\rightarrow$ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression

$$m_k^{(i)} = \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$



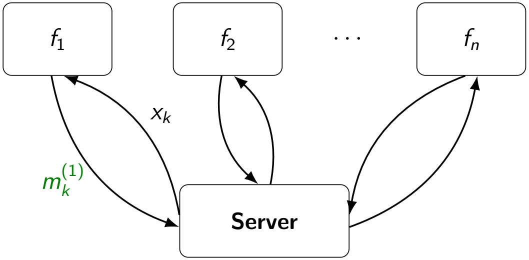
Part I: Modern error feedback (EF^{21})

$\rightarrow$ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression

$$m_k^{(i)} = \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} (m_k^{(i)} + d_{k-1}^{(i)})$$



Part I: Modern error feedback (EF^{21})

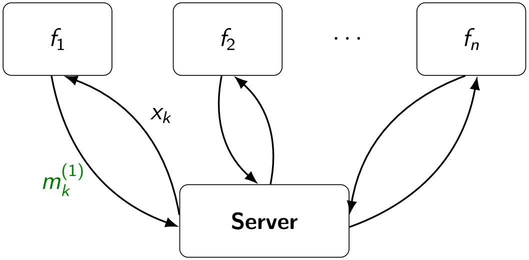
→ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression

$$m_k^{(i)} = \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} (m_k^{(i)} + d_{k-1}^{(i)})$$

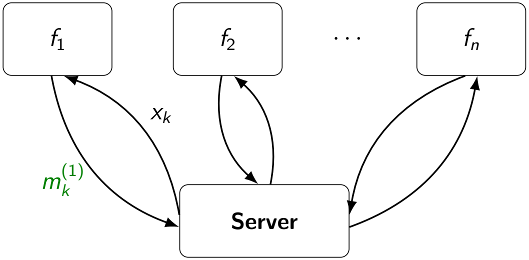
$$= x_k - \eta \sum_{i=1}^{n} d_k^{(i)}$$



Part I: Modern error feedback (EF^{21})

$\rightarrow$ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression



$$m_k^{(i)} = \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} (m_k^{(i)} + d_{k-1}^{(i)})$$

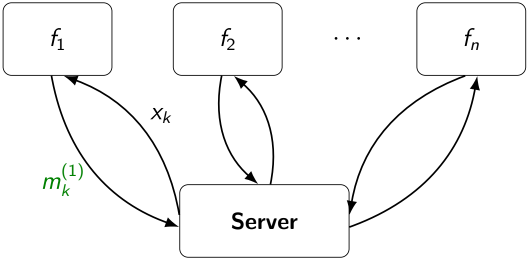
$$= x_k - \eta \sum_{i=1}^{n} d_k^{(i)}$$

$$d_k^{(i)} = d_{k-1}^{(i)} + m_k^{(i)} = d_{k-1}^{(i)} + \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

Part I: Modern error feedback (EF^{21})

$\rightarrow$ **3rd solut.:** Error feedback 21 (EF^{21})

Motivation: learn a *control variate* for each client $(d_k^i)_{k,i}$, subtracted before compression



$$m_k^{(i)} = \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

$$x_{k+1} = x_k - \eta \sum_{i=1}^{n} (m_k^{(i)} + d_{k-1}^{(i)})$$

$$= x_k - \eta \sum_{i=1}^{n} d_k^{(i)}$$

$$d_k^{(i)} = d_{k-1}^{(i)} + m_k^{(i)} = d_{k-1}^{(i)} + \mathcal{C}_i(\nabla f_i(x_k) - d_{k-1}^{(i)})$$

Remark: EF and EF^{21} are different in "spirit", but can be matched for linear operators.

Problem

Abundant literature on the topic: Seide et al. [2014], Stich et al. [2018], Wu et al. [2018], Karimireddy et al. [2019], Richtarik et al. [2021], Fatkhullin et al. [2021], Richtarik et al. [2022], Makarenko et al. [2022], Gruntkowska et al. [2023], Zhao et al. [2022], Wang et al. [2022], Dorfman et al. [2023].

Problem

Abundant literature on the topic: Seide et al. [2014], Stich et al. [2018], Wu et al. [2018], Karimireddy et al. [2019], Richtarik et al. [2021], Fatkhullin et al. [2021], Richtarik et al. [2022], Makarenko et al. [2022], Gruntkowska et al. [2023], Zhao et al. [2022], Wang et al. [2022], Dorfman et al. [2023].

1. Multiple different schemes (CGD, EF, EF^{21}, Error Control), etc.
2. Many bounds, not all comparable.

Problem

Abundant literature on the topic: Seide et al. [2014], Stich et al. [2018], Wu et al. [2018], Karimireddy et al. [2019], Richtarik et al. [2021], Fatkhullin et al. [2021], Richtarik et al. [2022], Makarenko et al. [2022], Gruntkowska et al. [2023], Zhao et al. [2022], Wang et al. [2022], Dorfman et al. [2023].

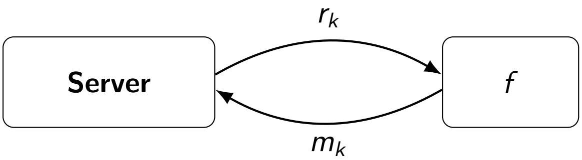
1. Multiple different schemes (CGD, EF, EF^{21}, Error Control), etc.
2. Many bounds, not all comparable.

Our goal:

1. Provide a tight analysis of EF, EF^{21}, and compare it to that of CGD.

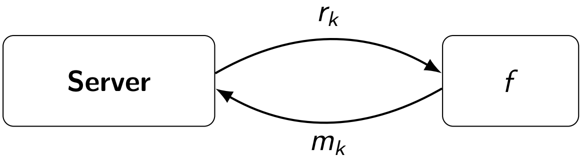
Part I: Analysis setup

1. Focus on single worker case



Part I: Analysis setup

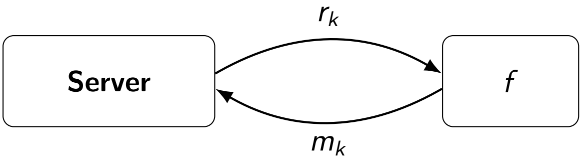
1. Focus on single worker case



2. Focus on μ-strongly convex and L-smooth, i.e., $f \in \mathcal{F}_{\mu,L}$.

Part I: Analysis setup

1. Focus on single worker case



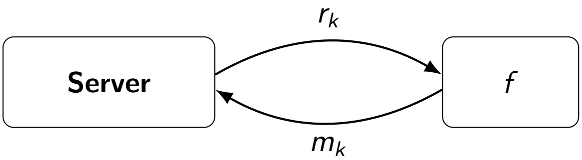
2. Focus on μ-strongly convex and L-smooth, i.e., $f \in \mathcal{F}_{\mu,L}$.
3. Consider deterministic and contractive compressors: $\mathcal{C}(\cdot)$ satisfies

$$\|x - \mathcal{C}(x)\|^2 \leq \epsilon \cdot \|x\|^2, \quad \forall x \in \mathbb{R}^d.$$

   where $0 \leq \epsilon \leq 1$ is typically close to 1.

Part I: Analysis setup

1. Focus on single worker case



2. Focus on μ-strongly convex and L-smooth, i.e., $f \in \mathcal{F}_{\mu,L}$.
3. Consider deterministic and contractive compressors: $\mathcal{C}(\cdot)$ satisfies

$$\|x - \mathcal{C}(x)\|^2 \leq \epsilon \cdot \|x\|^2, \quad \forall x \in \mathbb{R}^d.$$

   where $0 \leq \epsilon \leq 1$ is typically close to 1.
4. Looking for a *contraction*: $\mathcal{V}_{k+1} \leq \rho \mathcal{V}_k$

Betting time

Betting time

1. Who has read the paper ?

Betting time

1. Who has read the paper ?
2. For a given step size η, which is the best method? $(\mathrm{CGD}$, EF, $\mathrm{EF}^{21})$

Betting time

1. Who has read the paper ?
2. For a given step size η, which is the best method? $(\mathrm{CGD}$, EF, $\mathrm{EF}^{21})$
3. Which method has the largest admissible learning rate?

Betting time

1. Who has read the paper ?
2. For a given step size η, which is the best method? $(\mathrm{CGD}$, EF, $\mathrm{EF}^{21})$
3. Which method has the largest admissible learning rate?
4. Which method has the best possible linear contraction rate ?

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF

$$\eta_\star := \left(\frac{2}{L+\mu}\right) \cdot \left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF

$$\eta_\star := \left(\frac{2}{L+\mu}\right) \cdot \left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Convergence rate

$$\mathcal{V}_{k+1} \le \rho_\star \mathcal{V}_k,$$

$$\rho_\star = \sqrt{\epsilon} + \frac{1}{4}(1+\sqrt{\epsilon})(L-\mu)\lambda,$$

$$\lambda = \frac{\eta^\star}{L+\mu}\left[(1-\sqrt{\epsilon})(L-\mu) + (1+\sqrt{\epsilon})\sqrt{(L-\mu)^2 + 16L\mu\frac{\sqrt{\epsilon}}{(1+\sqrt{\epsilon})^2}}\right]$$

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF

$$\eta_\star := \left(\frac{2}{L+\mu}\right) \cdot \left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Convergence rate

$$\mathcal{V}_{k+1} \leq \rho_\star \mathcal{V}_k,$$

$$\rho_\star = \sqrt{\epsilon} + \frac{1}{4}(1+\sqrt{\epsilon})(L-\mu)\lambda,$$

$$\lambda = \tfrac{\eta^\star}{L+\mu}\left[(1-\sqrt{\epsilon})(L-\mu) + (1+\sqrt{\epsilon})\sqrt{(L-\mu)^2 + 16L\mu\tfrac{\sqrt{\epsilon}}{(1+\sqrt{\epsilon})^2}}\right]$$

Remark 1: *Tight* results, both in terms of **rate** and **optimal** Lyapunov function.

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF

$$\eta_\star := \left(\frac{2}{L+\mu}\right) \cdot \left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Convergence rate

$$\mathcal{V}_{k+1} \leq \rho_\star \mathcal{V}_k,$$

$$\rho_\star = \sqrt{\epsilon} + \frac{1}{4}(1+\sqrt{\epsilon})(L-\mu)\lambda,$$

$$\lambda = \frac{\eta^\star}{L+\mu}\left[(1-\sqrt{\epsilon})(L-\mu) + (1+\sqrt{\epsilon})\sqrt{(L-\mu)^2 + 16L\mu\frac{\sqrt{\epsilon}}{(1+\sqrt{\epsilon})^2}}\right]$$

Remark 2: Recovers GD for $\epsilon = 0$: $\eta_\star = \left(\frac{2}{L+\mu}\right)$; $\rho_\star = (\frac{L-\mu}{L+\mu})^2$

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF

$$\eta_\star := \left(\frac{2}{L+\mu}\right) \cdot \left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Convergence rate

$$\mathcal{V}_{k+1} \le \rho_\star \mathcal{V}_k,$$

$$\rho_\star = \sqrt{\epsilon} + \frac{1}{4}(1+\sqrt{\epsilon})(L-\mu)\lambda,$$

$$\lambda = \frac{\eta^\star}{L+\mu}\left[(1-\sqrt{\epsilon})(L-\mu) + (1+\sqrt{\epsilon})\sqrt{(L-\mu)^2 + 16L\mu\frac{\sqrt{\epsilon}}{(1+\sqrt{\epsilon})^2}}\right]$$

Remark 3: For all three methods, $\rho_\star(\epsilon) < 1 \Leftrightarrow \epsilon < 1$

Part I: Results

Results: *Optimal rates* for EF and EF^{21}: we can identify an optimal Lyapunov, the optimal step size, and the optimal rate

Step size setting

Optimal for EF and EF^{21}:

$$\eta_\star := \left(\frac{2}{L+\mu}\right)\cdot\left(\frac{1-\sqrt{\epsilon}}{1+\sqrt{\epsilon}}\right)$$

Convergence rate

Same for EF and EF^{21}:

$$\mathcal{V}_{k+1} \leq \rho_\star \mathcal{V}_k,$$

$$\rho_\star = \sqrt{\epsilon} + \frac{1}{4}(1+\sqrt{\epsilon})(L-\mu)\lambda,$$

$$\lambda = \frac{\eta^\star}{L+\mu}\left[(1-\sqrt{\epsilon})(L-\mu) + (1+\sqrt{\epsilon})\sqrt{(L-\mu)^2 + 16L\mu\frac{\sqrt{\epsilon}}{(1+\sqrt{\epsilon})^2}}\right]$$

Remark 4: Same rates for EF^{21}!! Same optimal step size, different Lyapunov function!

Comparing CGD / EF / EF^{21}: an apples-to-apples comparison

$\rightarrow$ We have obtained an analytical expression for the best possible Lyapunov / step size / contraction, for any contraction parameter $\epsilon, \kappa = \mu/L$.

Comparing CGD / EF / EF^{21}: an apples-to-apples comparison

→ We have obtained an analytical expression for the best possible Lyapunov / step size / contraction, for any contraction parameter $\epsilon, \kappa = \mu/L$.

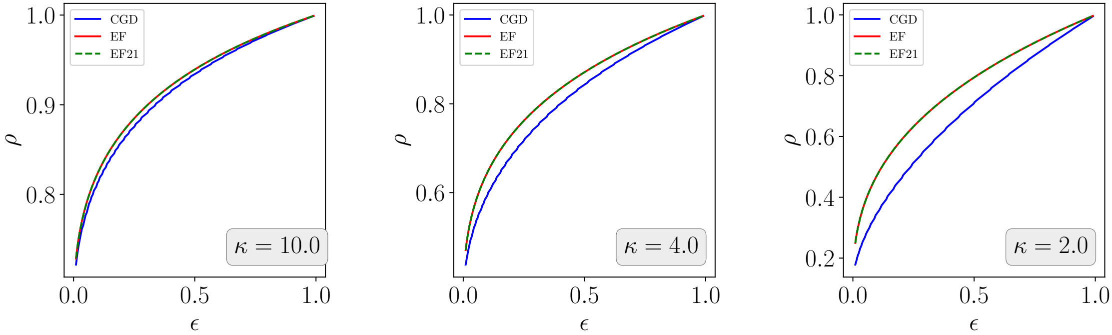


Figure: Optimal rate as a function of ϵ for three values fo κ : $10, 4, 2$

Comparing CGD / EF / EF^{21}: an apples-to-apples comparison

$\rightarrow$ We have obtained an analytical expression for the best possible Lyapunov / step size / contraction, for any contraction parameter $\epsilon, \kappa = \mu/L$.

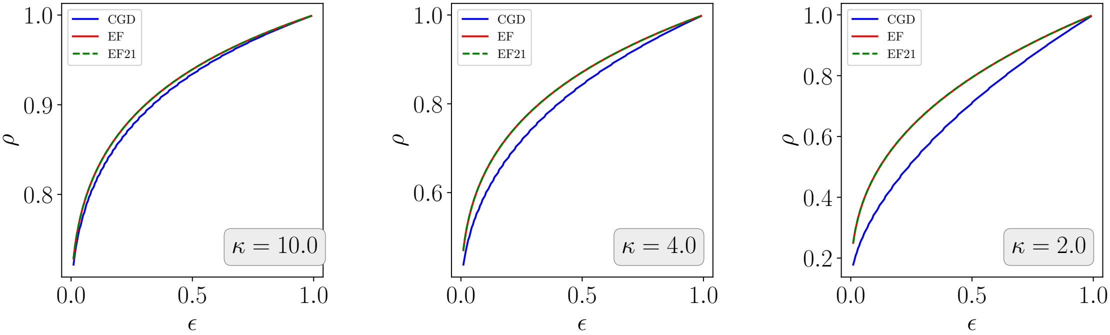


Figure: Optimal rate as a function of ϵ for three values fo κ : 10, 4, 2

Remark 1: The curves can be obtained numerically / analytically!

Comparing CGD / EF / EF^{21}: an apples-to-apples comparison

$\rightarrow$ We have obtained an analytical expression for the best possible Lyapunov / step size / contraction, for any contraction parameter $\epsilon, \kappa = \mu/L$.

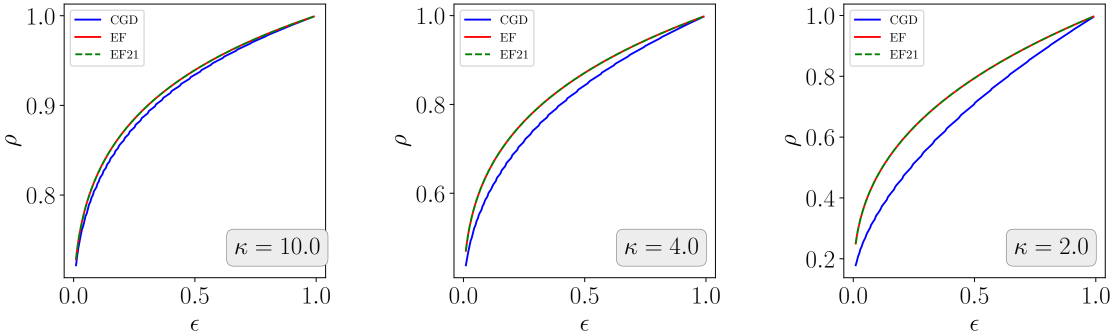
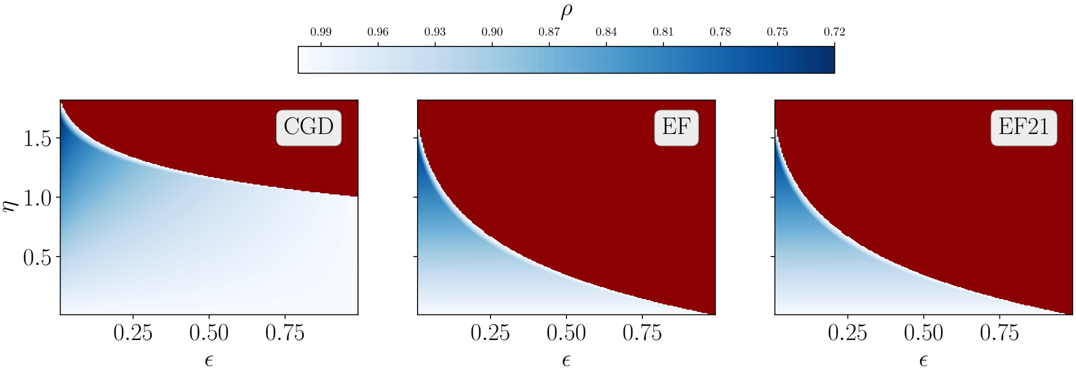


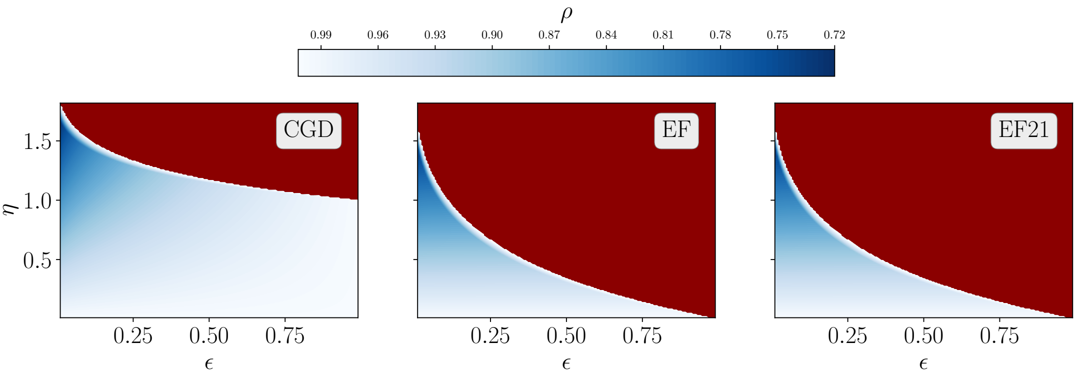
Figure: Optimal rate as a function of ϵ *for three values fo* $\kappa : 10, 4, 2$

Remark 2: CGD always dominates EF and EF^{21}!

Comparing CGD / EF / EF^{21}: analytical any step comparison

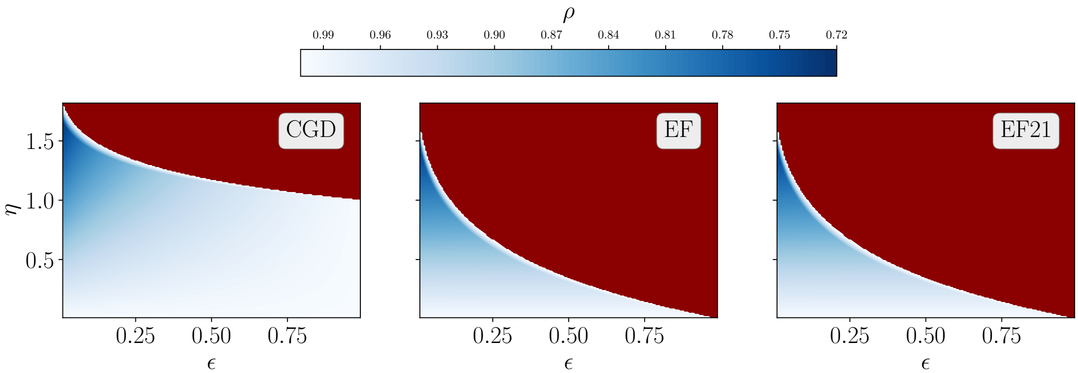


Comparing CGD / EF / EF^{21}: analytical any step comparison



1. No closed-form formula for the rate for the non-optimal step-size!

Comparing CGD / EF / EF^{21}: analytical any step comparison



1. No closed-form formula for the rate for the non-optimal step-size!
2. But numerical insights:
   - For a given (very small) step-size, EF and EF^{21} can dominate CGD.
   - The rate for EF and EF^{21} is the same

Part I: Conclusion

1. We give tight worst case analysis of EF and EF^{21}.

Part I: Conclusion

1. We give tight worst case analysis of EF and EF^{21}.

2. Showing that in the single agent, smooth and strongly convex, contractive and deterministic case, (surprisingly)
   - CGD dominates both EF and EF^{21}
   - EF and EF^{21} have the exact same best convergence rate.

Part I: Conclusion

1. We give tight worst case analysis of EF and EF^{21}.

2. Showing that in the single agent, smooth and strongly convex, contractive and deterministic case, (surprisingly)
   - CGD dominates both EF and EF^{21}
   - EF and EF^{21} have the exact same best convergence rate.

3. What's next? Some extensions!

Part I: Conclusion

1. We give tight worst case analysis of EF and EF^{21}.

2. Showing that in the single agent, smooth and strongly convex, contractive and deterministic case, (surprisingly)
   - CGD dominates both EF and EF^{21}
   - EF and EF^{21} have the exact same best convergence rate.

3. What's next? Some extensions!
4. The story is a bit different for multi-agent systems! (coming soon)

Outline of talk

Part I: What we did

- Intro to distributed optimization
- The methods analyzed
- The results

$\longrightarrow$ Aymeric

Part II: How we did it

- Lyapunov functions: our definition
- Defining the problem
- A bag of tricks

$\longrightarrow$ Daniel

Systematically identifying simple, yet optimal Lyapunov functions

Part II: How we did it

Part II: Lyapunov functions

Lyapunov functions help us prove convergence! We design them such that

$$\mathcal{V}(\xi_k) \xrightarrow{k \to \infty} 0 \implies x_k \xrightarrow{k \to \infty} x_\star$$

Part II: Lyapunov functions

Lyapunov functions help us prove convergence! We design them such that

$$\mathcal{V}(\xi_k) \xrightarrow{k\to\infty} 0 \implies x_k \xrightarrow{k\to\infty} x_\star$$

Examples:

1. $\|x_k - x_\star\|^2$
2. $f(x_k) - f_\star + \|g_k\|^2$
3. $f(x_k) - f_\star + \frac{1}{n}\sum_{i=1}^n \|d_k^{(i)} - \nabla f_i(x_k)\|^2$

Part II: Lyapunov functions

Candidate Lyapunov functions $\mathcal{V}(\xi, x; f)$ such that:

1. (non-negative) $\mathcal{V}(\xi, x; f) \geq 0$

Part II: Lyapunov functions

Candidate Lyapunov functions $\mathcal{V}(\xi, x; f)$ such that:

1. (non-negative) $\mathcal{V}(\xi, x; f) \geq 0$
2. (zero at fixed-point) $\mathcal{V}(\xi, x; f) = 0 \iff \xi = \xi_\star$ and $x = x_\star$

Part II: Lyapunov functions

Candidate Lyapunov functions $\mathcal{V}(\xi, x; f)$ such that:

1. (non-negative) $\mathcal{V}(\xi, x; f) \geq 0$
2. (zero at fixed-point) $\mathcal{V}(\xi, x; f) = 0 \iff \xi = \xi_\star$ and $x = x_\star$
3. (quadratic lower bound) $\mathcal{V}(\xi, x; f) \geq (\xi - \xi_\star)^\top (A \otimes I_d)(\xi - \xi_\star) + a(f(x) - f_\star)$, for some matrix A and scalar a.

Part II: Lyapunov functions

Candidate Lyapunov functions $\mathcal{V}(\xi, x; f)$ such that:

1. (non-negative) $\mathcal{V}(\xi, x; f) \geq 0$
2. (zero at fixed-point) $\mathcal{V}(\xi, x; f) = 0 \iff \xi = \xi_\star$ and $x = x_\star$
3. (quadratic lower bound) $\mathcal{V}(\xi, x; f) \geq (\xi - \xi_\star)^\top (A \otimes I_d)(\xi - \xi_\star) + a(f(x) - f_\star)$, for some matrix A and scalar a.

Additional desired property:

$$\mathcal{V}(\xi_1, x_1; f) \leq \rho \cdot \mathcal{V}(\xi_0, x_0; f), \tag{1}$$

where $0 \leq \rho < 1$ is a **contraction factor**.

Part II: The problem

Want to find good Lyapunov functions:

Part II: The problem

Want to find good Lyapunov functions:

$$\min_{\mathcal{V}} \left\{ \max_{f \in \mathcal{F}} \frac{\mathcal{V}(\xi_1, x_1; f)}{\mathcal{V}(\xi_0, x_0; f)} : (\xi_1, x_1) = \mathcal{M}(\xi_0, x_0; f) \right\}. \tag{2}$$

Part II: The problem

Want to find good Lyapunov functions:

$$\min_{\mathcal{V}} \left\{ \max_{f \in \mathcal{F}} \frac{\mathcal{V}(\xi_1, x_1; f)}{\mathcal{V}(\xi_0, x_0; f)} : (\xi_1, x_1) = \mathcal{M}(\xi_0, x_0; f) \right\}. \tag{2}$$

Taylor et al. [2018]: PEPs let us find worst-case contractions for **given** $\mathcal{V}$:

$$\max_{f \in \mathcal{F}_{\mu,L}} \left\{ \frac{\mathcal{V}(\xi_1, x_1; f)}{\mathcal{V}(\xi_0, x_0; f)} : (\xi_1, x_1) = \mathrm{GD}(\xi_0, x_0; f) \right\} \tag{GD-PEP}$$

Part II: The problem

Want to find good Lyapunov functions:

$$\min_{\mathcal{V}} \left\{ \max_{f \in \mathcal{F}} \frac{\mathcal{V}(\xi_1, x_1; f)}{\mathcal{V}(\xi_0, x_0; f)} : (\xi_1, x_1) = \mathcal{M}(\xi_0, x_0; f) \right\}. \tag{2}$$

Taylor et al. [2018]: PEPs let us find worst-case contractions for **given** $\mathcal{V}$:

$$\max_{f \in \mathcal{F}_{\mu,L}} \left\{ \frac{\mathcal{V}(\xi_1, x_1; f)}{\mathcal{V}(\xi_0, x_0; f)} : (\xi_1, x_1) = \mathrm{GD}(\xi_0, x_0; f) \right\} \tag{GD-PEP}$$

We can find a good Lyapunov function for a **given** ρ by solving:

$$\underset{\substack{P \succeq 0, \\ \lambda_{ij} \geq 0}}{\text{feasible}} \left\{ \begin{array}{l} A_1^\top P A_1 - \rho A_0^\top P A_0 - \sum_{i,j} \lambda_{ij} M_{ij} \succeq 0 \\ \sum_{i,j} \lambda_{ij} m_{ij} = 0 \\ \mathrm{tr}(P) = 1. \end{array} \right\} \tag{GD-SDP}$$

Part II: Our Lyapunov functions

$$\xi_k^{\mathrm{CGD}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ \mathcal{C}(\eta \nabla f(x_k)) \end{bmatrix}, \quad \xi_k^{\mathsf{EF}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ \mathcal{C}(e_k + \eta \nabla f(x_k)) \\ e_k \end{bmatrix}, \quad \xi_k^{\mathrm{EF}^{21}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ d_k \end{bmatrix},$$

Part II: Our Lyapunov functions

$$\xi_k^{\mathrm{CGD}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ \mathcal{C}(\eta \nabla f(x_k)) \end{bmatrix}, \quad \xi_k^{\mathsf{EF}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ \mathcal{C}(e_k + \eta \nabla f(x_k)) \\ e_k \end{bmatrix}, \quad \xi_k^{\mathrm{EF}^{21}} = \begin{bmatrix} x_k \\ \nabla f(x_k) \\ d_k \end{bmatrix},$$

EF^{21} optimal Lyapunov function ($\eta = 0.3$, $\epsilon = 0.1$):

$$\begin{bmatrix} x_k - x_\star \\ g_k \\ d_k \end{bmatrix}^\top \begin{bmatrix} 1.44935440e^{-04} & -4.87502814e^{-04} & 1.43273186e^{-04} \\ -4.87502814e^{-04} & 5.71381080e^{-01} & -4.28203308e^{-01} \\ 1.43273186e^{-04} & -4.28203308e^{-01} & 4.26190780e^{-01} \end{bmatrix} \begin{bmatrix} x_k - x_\star \\ g_k \\ d_k \end{bmatrix}$$
$$+ 0.00228321 \cdot (f(x_k) - f_\star)$$

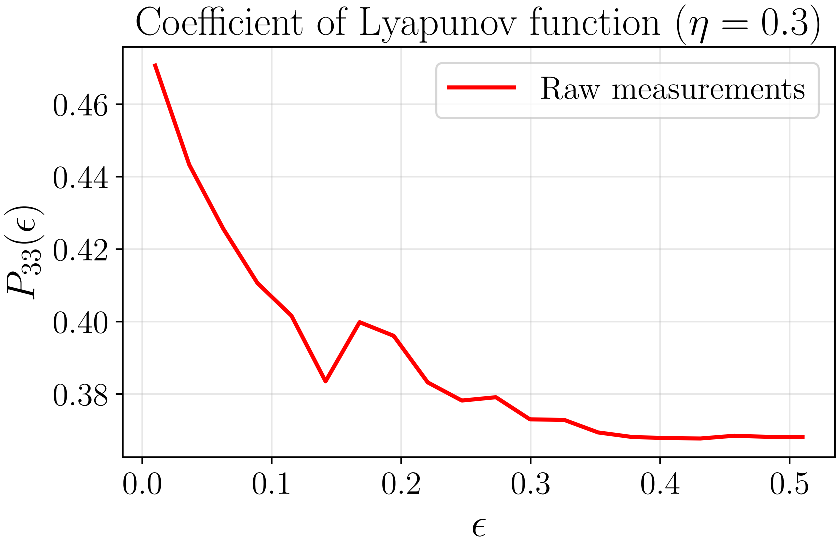
Part II: Forced sparsity

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix}, \quad \text{and} \quad p$$

Part II: Forced sparsity

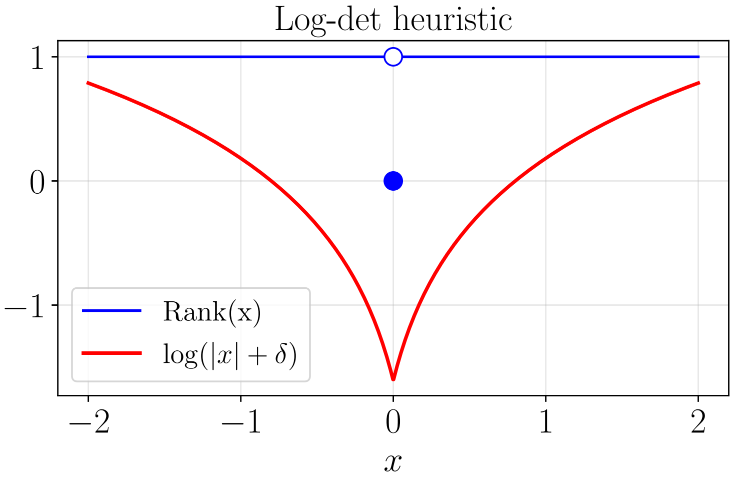
$$P = \begin{bmatrix} 1 - P_{33} & -P_{33} \\ & \\ -P_{33} & P_{33} \end{bmatrix}$$

Part II: Forced sparsity



Part II: log-det heuristic

Fazel et al. [2003]: Minimize rank of P using heuristics!



Part II: log-det heuristic

First-order Taylor expansion at iteration P_k:

$$\log\det(P + \delta I) \approx \log\det(P_k + \delta I) + \operatorname{Tr}\left[(P_k + \delta I)^{-1}(P - P_k)\right]$$

Part II: log-det heuristic

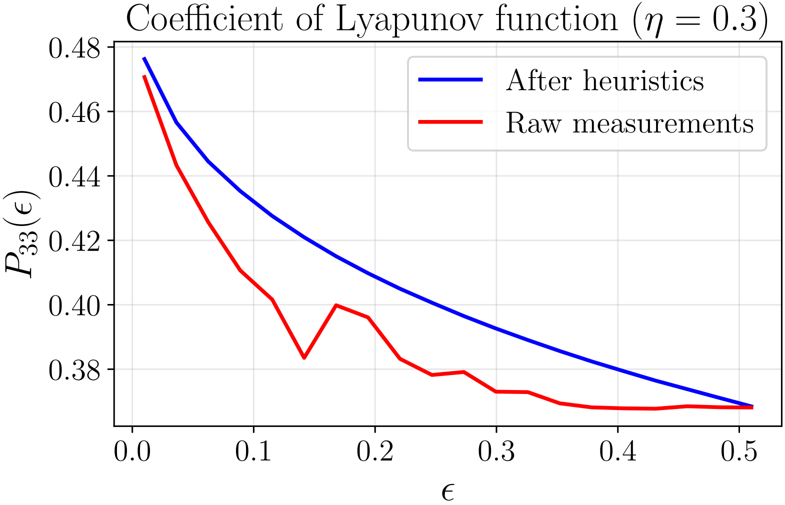
First-order Taylor expansion at iteration P_k:

$$\log\det(P + \delta I) \approx \log\det(P_k + \delta I) + \operatorname{Tr}\left[(P_k + \delta I)^{-1}(P - P_k)\right]$$

Iterative procedure:

$$P_{k+1} = \arg\min_P \operatorname{Tr}(P_k + \delta I)^{-1} P.$$

Part II: log-det heuristic



Part II: Symbolic regression

PySR & SymbolicRegression.jl

github.com/MilesCranmer/pysr_paper

Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl

Miles Cranmer[1,2]

[1] *Princeton University, Princeton, NJ, USA*
[2] *Flatiron Institute, New York, NY, USA*

May 2, 2023

Part II: Symbolic regression

PySR: High-Performance Symbolic Regression in Python and Julia

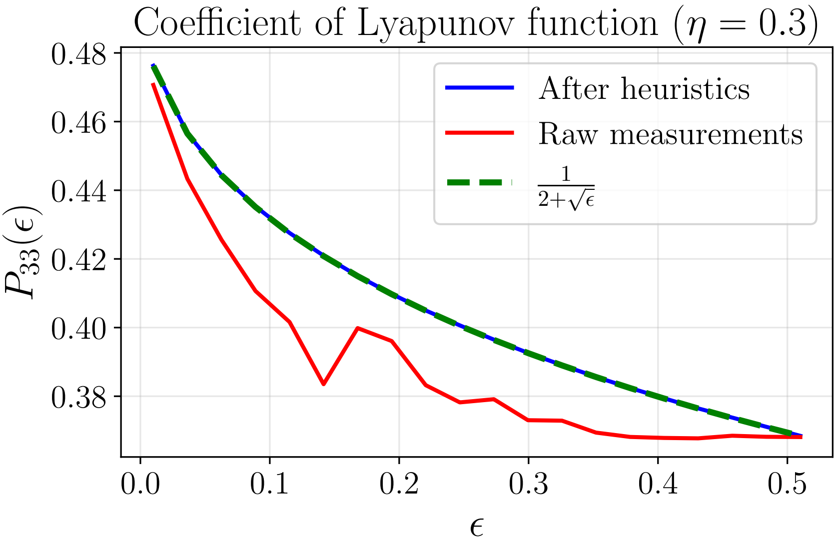
Docs	Forums	Paper	colab demo
docs passing	discussions github	arXiv 2305.01582	colab notebook

pip	conda	Stats
pypi package 1.5.8	conda-forge v1.5.8	pip: downloads 503k conda: downloads 616k total

If you find PySR useful, please cite the paper arXiv:2305.01582. If you've finished a project with PySR, please submit a PR to showcase your work on the research showcase page!

▸ PySR on GitHub

Part II: Symbolic regression



Closed-forms

(tight analysis for CGD found in De Klerk et al. [2020]):

$$\mathcal{V}^{\mathrm{CGD}} := f(x) - f_\star,$$

$$\mathcal{V}^{\mathrm{EF}} := \|x_k - x_\star - e_k\|^2 + \frac{1}{\sqrt{\epsilon}}\|e_k\|^2,$$

$$\mathcal{V}^{\mathrm{EF}^{21}} := \|g_k - d_k\|^2 + \sqrt{\epsilon} \cdot \|d_k\|^2.$$

Thank you!

Tight analyses of first-order methods with error feedback

Daniel Berg Thomsen[1,2*] **Adrien Taylor**[1] **Aymeric Dieuleveut**[2]
[1]INRIA, D.I. École Normale Supérieure, PSL Research University, 75005 Paris, France
[2]CMAP, CNRS, École Polytechnique, Institut Polytechnique de Paris, 91120 Palaiseau, France

Abstract

Communication between agents often constitutes a major computational bottleneck in distributed learning. One of the most common mitigation strategies is to compress the information exchanged, thereby reducing communication overhead. To counteract the degradation in convergence associated with compressed communication, error feedback schemes—most notably EF and EF^{21}—were introduced. In this work, we provide a *tight analysis* of both of these methods. Specifically, we find the Lyapunov function that yields the best possible convergence rate for each method—with matching lower bounds. This principled approach yields sharp performance guarantees and enables a rigorous, apples-to-apples comparison between EF, EF^{21}, and compressed gradient descent. Our analysis is carried out in a simplified yet representative setting, which allows for clean theoretical insights and fair comparison of the underlying mechanisms.

References I

Miles Cranmer. Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl. *arXiv:2305.01582 [physics]*, May 2023.

Etienne De Klerk, Francois Glineur, and Adrien B Taylor. Worst-Case Convergence Analysis of Inexact Gradient and Newton Methods Through Semidefinite Programming Performance Estimation. *SIAM Journal on Optimization*, 30(3):2053–2082, 2020.

Ron Dorfman, Shay Vargaftik, Yaniv Ben-Itzhak, and Kfir Yehuda Levy. DoCoFL: Downlink compression for cross-device federated learning. In *International Conference on Machine Learning (ICML)*, 2023.

Ilyas Fatkhullin, Igor Sokolov, Eduard Gorbunov, Zhize Li, and Peter Richtárik. EF21 with Bells & Whistles: Practical Algorithmic Extensions of Modern Error Feedback, October 2021. arXiv:2110.03294 [cs, math].

References II

Maryam Fazel, Haitham Hindi, and Stephen P. Boyd. Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In *American Control Conference (ACC)*, 2003.

Kaja Gruntkowska, Alexander Tyurin, and Peter Richtárik. EF21-P and Friends: Improved Theoretical Communication Complexity for Distributed Optimization with Bidirectional Compression. In *International Conference on Machine Learning (ICML)*, 2023.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error Feedback Fixes SignSGD and other Gradient Compression Schemes. In *International Conference on Machine Learning (ICML)*, 2019.

Dmitry Makarenko, Elnur Gasanov, Rustem Islamov, Abdurakhmon Sadiev, and Peter Richtárik. Adaptive Compression for Communication-Efficient Distributed Training. *arXiv:2211.00188 [cs]*, October 2022.

References III

Peter Richtarik, Igor Sokolov, and Ilyas Fatkhullin. EF21: A New, Simpler, Theoretically Better, and Practically Faster Error Feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Peter Richtarik, Igor Sokolov, Elnur Gasanov, Ilyas Fatkhullin, Zhize Li, and Eduard Gorbunov. 3PC: Three Point Compressors for Communication-Efficient Distributed Training and a Better Theory for Lazy Aggregation. In *International Conference on Machine Learning (ICML)*, 2022.

Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-Bit Stochastic Gradient Descent and its Application to Data-Parallel Distributed Training of Speech DNNs. In *Annual Conference of the International Speech Communication Association*, 2014.

Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified SGD with Memory. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2018.

References IV

Adrien Taylor, Bryan Van Scoy, and Laurent Lessard. Lyapunov Functions for First-Order Methods: Tight Automated Convergence Guarantees. In *International Conference on Machine Learning (ICML)*, 2018.

Yujun Wang, Lu Lin, and Jinghui Chen. Communication-Compressed Adaptive Gradient Method for Distributed Nonconvex Optimization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error Compensated Quantized SGD and its Applications to Large-scale Distributed Optimization. In *International Conference on Machine Learning (ICML)*, 2018.

Haoyu Zhao, Boyue Li, Zhize Li, Peter Richtárik, and Yuejie Chi. BEER: Fast $O(1/T)$ Rate for Decentralized Nonconvex Optimization with Communication Compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.