Automated Performance Estimation for Decentralized Optimization via Network Size Independent Problems

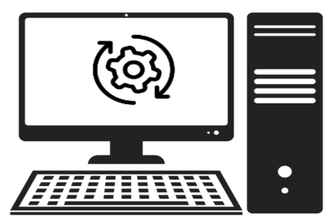
Sébastien Colla, Julien Hendrickx

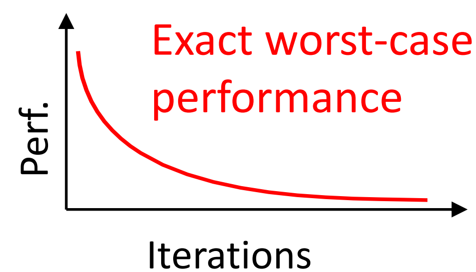
Mathematical Engineering Department, UCLouvain

PEP-talks 2023

```
function MyDecentralizedAlgo()
    N = 10; % number of agents
    x0 = init(N); % Initial point
    x = x0;

    for i=1:niter
        % any local computations
        % and local communications
        x = update(x,N);
    end
end
```





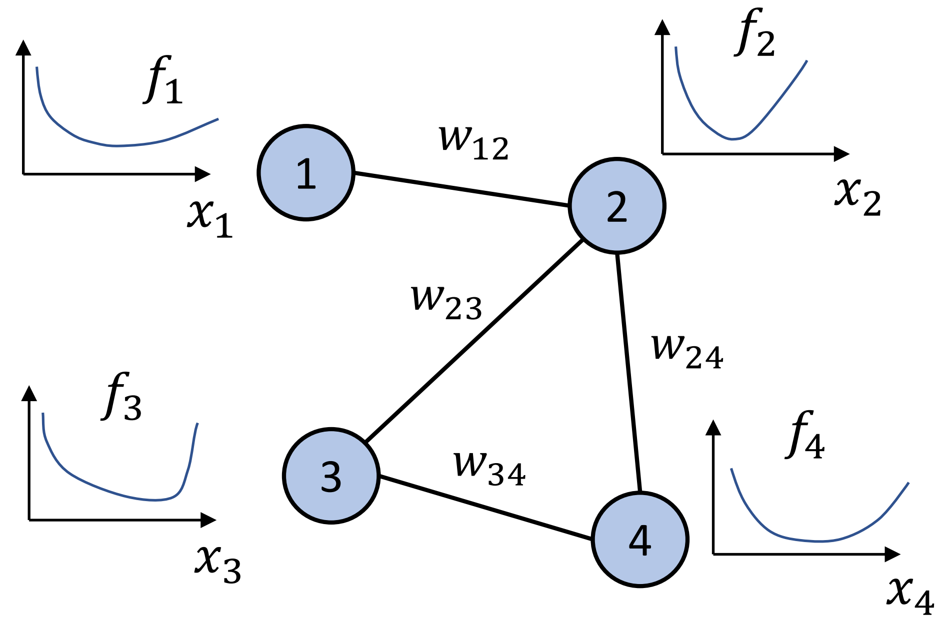
Decentralized Optimization

$$\min_x f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x)$$

⇕

$$\min_{x_1,\dots,x_N} F_s(x_1, \dots, x_N) = \frac{1}{N}\sum_{i=1}^{N} f_i(x_i)$$

$$\text{s.t. } x_i = x_j \quad \forall\, (i,j) \text{ neighbors}$$



Decentralization

- Local function: f_i
- Local copy of x: x_i

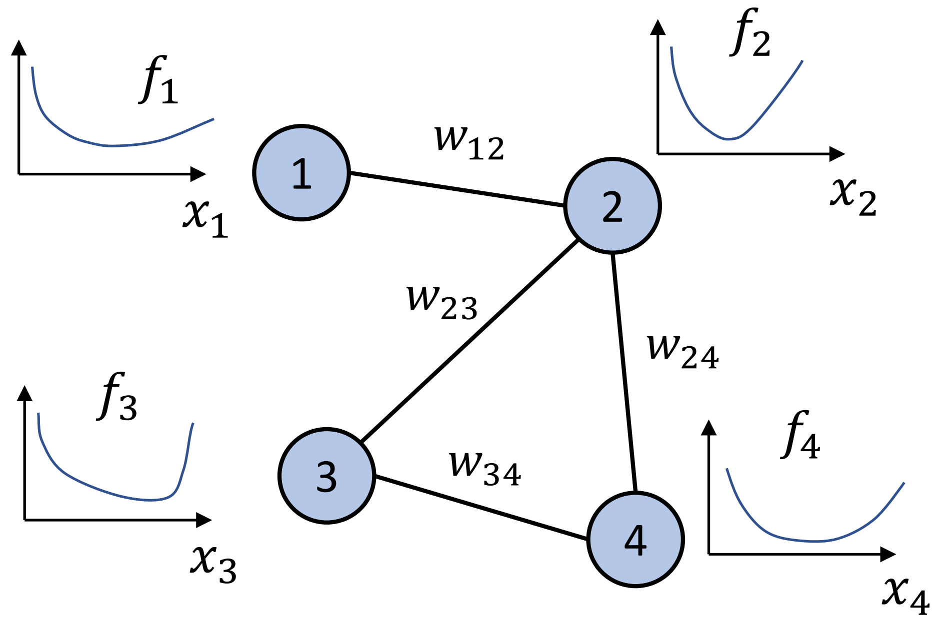
Iterative algorithm

- Local computations
- Local communications (W) so that $x_i = x_j$ (eventually)

Decentralized Gradient Descent (DGD)

$$\min_{x_1,\dots,x_N} F_s(x_1, \dots, x_N) = \frac{1}{N}\sum_{i=1}^{N} f_i(x_i)$$

$$\text{s.t. } x_i = x_j \quad \forall\ (i,j) \text{ neighbors}$$



Decentralization

- Local function: f_i
- Local copy of x: x_i

Decentralized Gradient Descent (DGD)

For each iteration k

$$y_i^k = \sum_j w_{ij}\, x_j^k \qquad \textit{Consensus step}$$

$$x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \qquad \textit{Local gradient step}$$

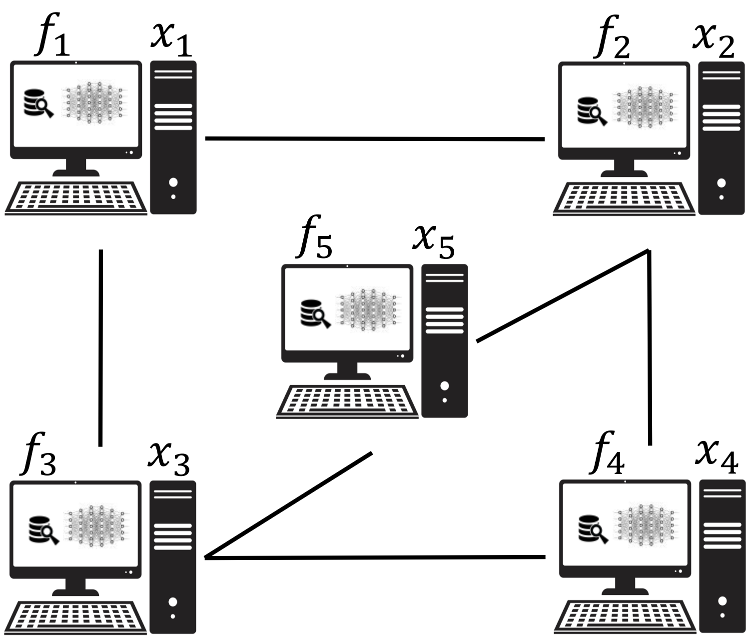
Motivations: Decentralized Machine Learning

Notations

- Model parameters x
- Data set $\{d \in \mathcal{D}\}$

Model training

$$\min_x \sum_{d \in D} \text{Error}(x, d) + \text{regul}\ (x)$$



Decentralization

- Part of the data $\mathcal{D}_i$
- Local function

$$f_i(x) = \sum_{d \in \mathcal{D}_i} \text{Error}(x, d)$$

- Local copy of x

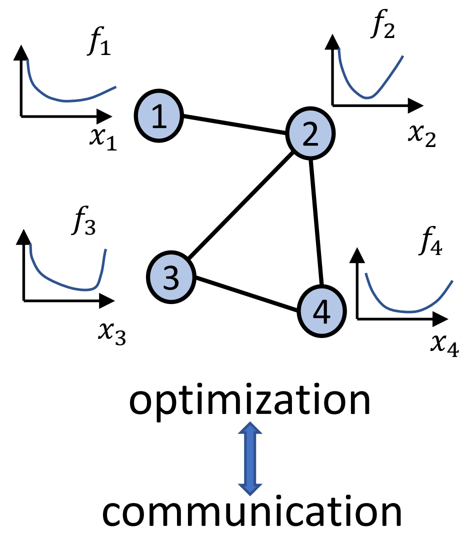
Motivations Big data – Privacy – Speed Up

Decentralized Optimization

Many challenges for better methods

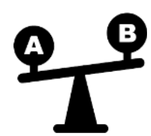
BUT

Analysis highly complex

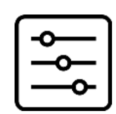


- Performance bounds: complex and **conservative**

- Difficult algorithms comparisons



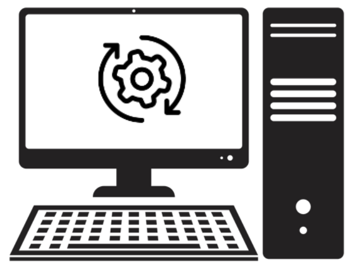
- Difficult parameters tuning

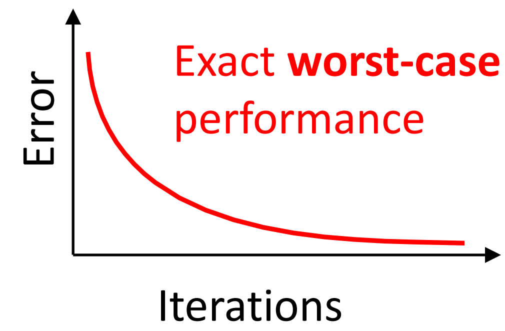


Objective

```
function MyDecentralizedAlgo()
    N = 10;         % number of agents
    x0 = init(N);   % initial points
    x = x0;

    for k=1:niter
        % any local computations
        % any local communications
        x = update(x,N);
    end
end
```





Impact for decentralized optimization

- Access to **accurate performance** of methods
- Easier **comparison and tuning** of algorithms
- **Rapid exploration** of new algorithms.

Performance Estimation Problem (PEP)

idea

Worst-cases are solutions to optimization problems

Find the worst f_i and x_i^k for K iterations of method M

→ f_i and x_i^k maximizing the error criterion after K iterations

PEP for Distributed Optimization

[Colla 2022]

$$\max_{f_i, x_i^k, y_i^k, x^*} \quad \text{perf}(f_i, x_i^0, \dots, x_i^K)$$

With

$f_\text{i} \in$		class of functions
x_i^0		initial condition
$\nabla f(x^*) = 0$		optimality condition
$x_i^0, \dots, x_i^K$ $y_i^0, \dots, y_i^{K-1}$		algorithm description

Toolbox
PESTO
PEPit

Can be solved with proper discretization
and **SDP reformulation**

PEP for Distributed Optimization

[Colla 2022]

Example of setting for DGD

$$\max_{f_i,\, x_i^k,\, y_i^k,\, x^*} \quad \mathrm{perf}(f_i, x_i^0, \dots, x_i^K) = f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

With f_{i} convex with bounded subgradients

$\frac{1}{N}\sum_{i=1}^{N} \|x_i^0 - x^*\|^2 \leq R^2$ initial condition

$\nabla f(x^*) = 0$ optimality condition

Toolbox PESTO PEPit

DGD
$$\begin{cases} y_i^k = \sum_{j=1}^{N} W_{ij}\, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases}$$

for $i = 1 \dots N$
for $\mathrm{k} = 0 \dots K-1$

Can be solved with proper discretization and **SDP reformulation**

PEP for Distributed Optimization

[Colla 2022]

Example of setting for DGD

$$\max_{f_i, x_i^k, y_i^k, x^*} \quad \text{perf}(f_i, x_i^0, \dots, x_i^K) = f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

With

W given	**Spectral class of W**
Exact Specific	W symmetric doubly stochastic $\lambda(W) \in [\lambda^-, \lambda^+]$ **relaxation**

DGD
$$\begin{cases} y_i^k = \sum_{j=1}^{N} \boxed{W_{ij}} \, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases}$$

for $i = 1 \dots N$

for k $= 0 \dots K-1$

Toolbox
PESTO
PEPit

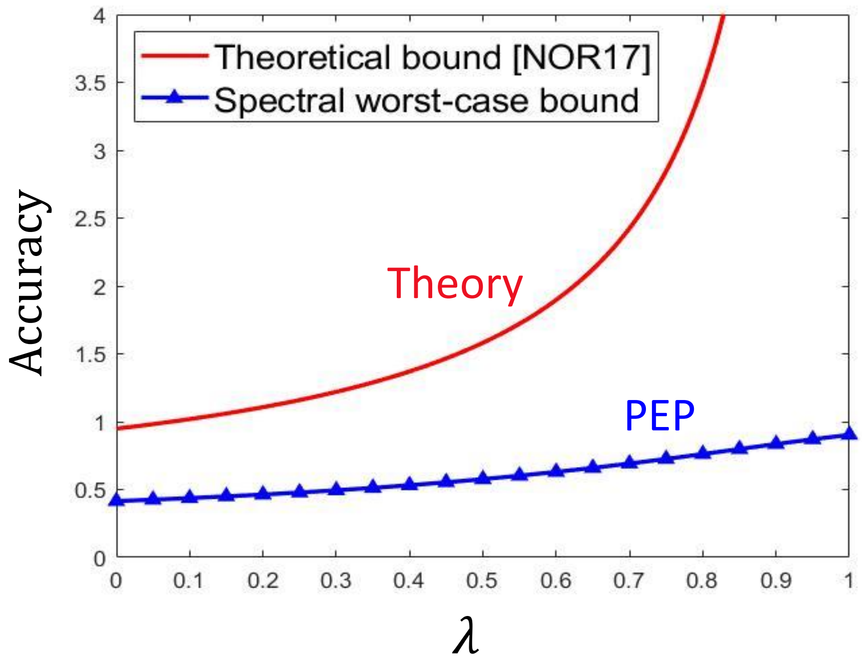
Can be solved with proper discretization and **SDP reformulation**

PEP for Distributed Optimization

[Colla 2022]

- Tightness observed in all experiments (DGD, DIGing, etc)
- Improved performance guarantees and tuning
- Interesting insights (e.g. worst network matrix)

Example for DGD:



PEP for Distributed Optimization

[Colla 2022]

- Tightness observed in all experiments (DGD, DIGing, etc)
- Improved performance guarantees and tuning
- Interesting insights (e.g. worst network matrix)

Size of the SDP PEP formulation

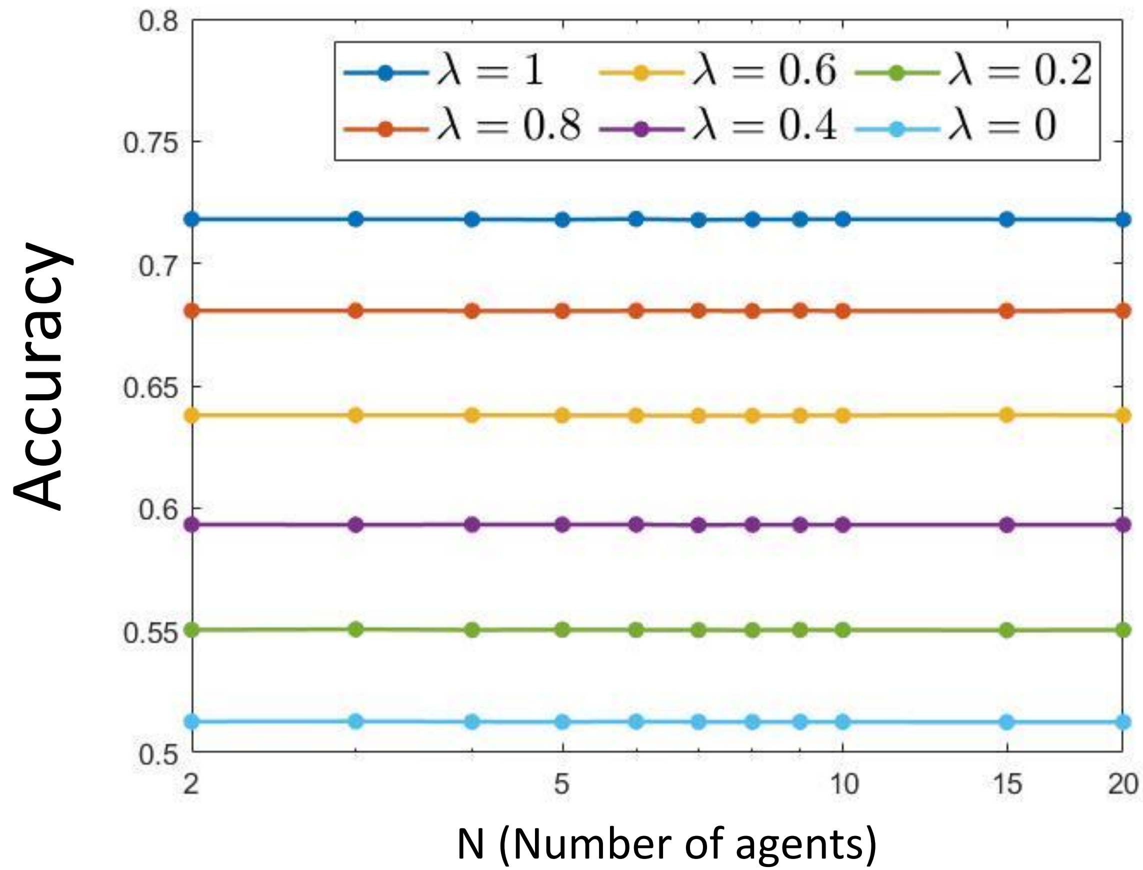
- Depends on the number of variables (increasing with N and K)
- Does not depend on their dimension d (*d* is unknown)

  because SDP reformulation involves scalar products

$$\max \quad \text{perf}(f_i, x_i^0, \dots, x_i^K)$$

$$f_i \quad x_i^k, y_i^k \in \mathbb{R}^d \qquad \text{for } i = 1 \dots N$$
$$\text{for } k = 0 \dots K$$

DGD – Worst-case evolution with N



For $K = 5$ iterations and $\lambda(W) \in [-\lambda, \lambda]$

New Contributions

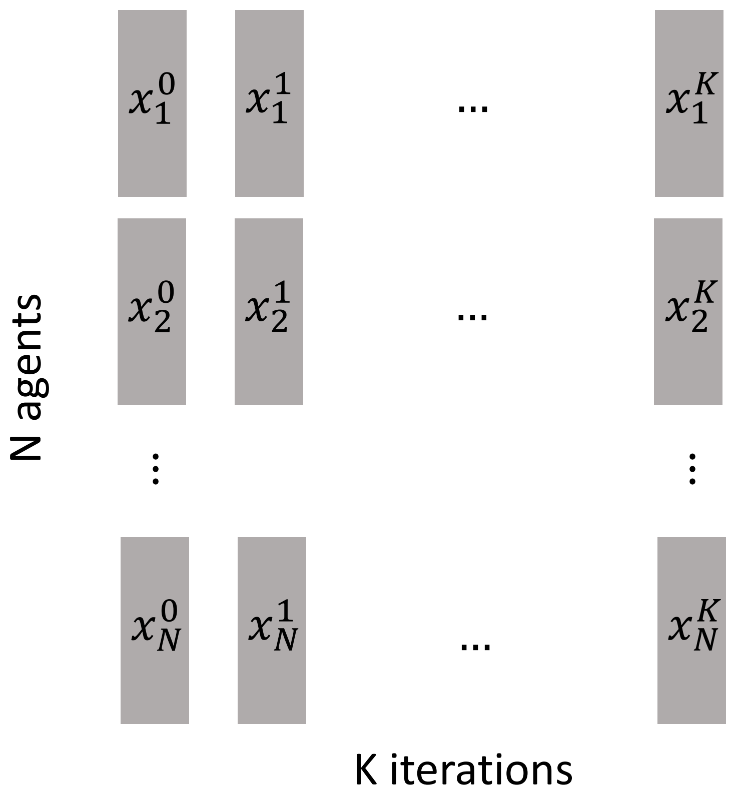
PEP for distributed optimization whose size (and results) are
independent of the number of agents N

Advantages

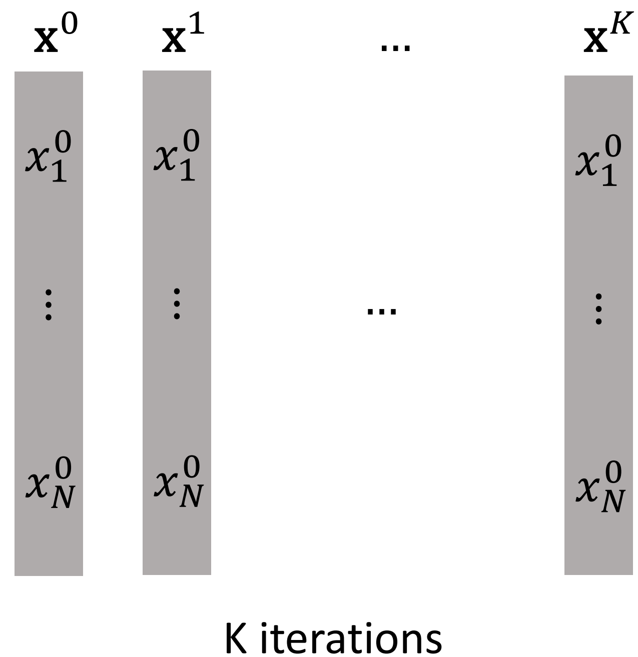
- Performance guarantees for any value of N
- Small size PEP are easier to exploit

Global representation

$$\max_{f_i,\, x_i^k,\, y_i^k} \quad \text{perf}(f_i, x_i^0, \ldots, x_i^K)$$



$$\max_{\substack{F_s(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^N f_i(x_i) \\ \mathbf{x}^k,\, \mathbf{y}^k}} \quad \text{perf}(F_s, \mathbf{x}^0, \ldots, \mathbf{x}^K)$$



Size of the SDP PEP formulation

- Depends on the number of variables
- Does not depend on their dimension

PEP - local representation | PEP - global representation

(example for DGD)

PEP - local representation

$$\max_{f_i,\, x_i^k,\, y_i^k,\, x^*} \quad f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

f_{i} convex with bounded subgradients

$$\frac{1}{N}\sum_{i=1}^{N}\left\|x_i^0 - x^*\right\|^2 \le R^2$$

$\nabla f(x^*) = 0$ optimality condition

PEP - global representation

$$\max_{F_S,\, \mathbf{x}^k,\, \mathbf{y}^k,\, \mathbf{x}^*} \quad F_S(\bar{x}^K \mathbf{1}) - F_S(\mathbf{x}^*)$$

F_S convex with bounded subgradients

$$\frac{1}{N}\|\mathbf{x}^0 - \mathbf{x}^*\|^2 \le R^2$$

$\mathbf{x}^*$ optimality condition

PEP - local representation

PEP - global representation

(example for DGD)

Local representation:

$$\max_{f_i,\, x_i^k,\, y_i^k,\, x^*} \quad f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

f_i convex with bounded subgradients

$$\frac{1}{N}\sum_{i=1}^{N}\left\|x_i^0 - x^*\right\|^2 \le R^2$$

$\nabla f(x^*) = 0$ optimality condition

DGD
$$\begin{cases} y_i^k = \sum_{j=1}^{N} W_{ij}\, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases} \quad \mathrm{i} = 1 \ldots N$$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

Global representation:

$$\max_{F_s,\, \mathbf{x}^k,\, \mathbf{y}^k,\, \mathbf{x}^*} \quad F_s(\bar{x}^K \mathbf{1}) - F_s(\mathbf{x}^*)$$

F_s convex with bounded subgradients

$$\frac{1}{N}\|\mathbf{x}^0 - \mathbf{x}^*\|^2 \le R^2$$

$\mathbf{x}^*$ optimality condition

DGD
$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^{\boldsymbol{k}} \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \boldsymbol{\nabla} F_s(\mathbf{x}^k) \end{cases}$$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

PEP - local representation

PEP - global representation

(example for DGD)

Local:

$$\max_{f_i,\, x_i^k,\, y_i^k,\, x^*} \quad f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

Global:

$$\max_{F_s,\, \mathbf{x}^k,\, \mathbf{y}^k,\, \mathbf{x}^*} \quad F_s(\bar{x}^K \mathbf{1}) - F_s(\mathbf{x}^*)$$

For any decentralized method that combines

- Gradient evaluations
- Consensus steps
- Linear combinations

Local:

DGD

$$\begin{cases} y_i^k = \sum_{j=1}^{N} W_{ij}\, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases}$$

for W: symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

Global:

DGD

$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^{\boldsymbol{k}} \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \boldsymbol{\nabla} F_s(\mathbf{x}^k) \end{cases}$$

for W: symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

PEP - local representation

PEP - global representation

(example for DGD)

$$\max_{f_i,\, x_i^k, y_i^k, x^*} \quad f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

$$\max_{F_s,\, \mathbf{x}^k, \mathbf{y}^k, \mathbf{x}^*} \quad F_s(\bar{x}^K \mathbf{1}) - F_s(\mathbf{x}^*)$$

?

?

$$F_s(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^N f_i(x_i)$$

For any decentralized method that combines

- Gradient evaluations
- Consensus steps
- Linear combinations

DGD
$$\begin{cases} y_i^k = \sum_{j=1}^N W_{ij}\, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases}$$

DGD
$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^k \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \nabla F_s(\mathbf{x}^k) \end{cases}$$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

PEP - local representation

PEP - global representation

(example for DGD)

$$\max_{f_i,\, x_i^k,\, y_i^k,\, x^*} \quad f\left(\frac{1}{N}\sum_i x_i^K\right) - f(x^*)$$

$$\max_{F_s,\, \mathbf{x}^k,\, \mathbf{y}^k,\, \mathbf{x}^*} \quad F_s(\bar{x}^K \mathbf{1}) - F_s(\mathbf{x}^*)$$

? ?

relax separability

$$F_s(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^N f_i(x_i)$$

For any decentralized method that combines

- Gradient evaluations
- Consensus steps
- Linear combinations

DGD

$$\begin{cases} y_i^k = \sum_{j=1}^N W_{ij}\, x_j^k \\ x_i^{k+1} = y_i^k - \alpha \nabla f_i(x_i^k) \end{cases}$$

DGD

$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^k \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \nabla F_s(\mathbf{x}^k) \end{cases}$$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

for W symmetric, doubly stochastic, $\lambda(W) \in [\lambda^-, \lambda^+]$

Generalized Decentralized Problem

relax separability of F_s

$$\min_{\mathbf{x}} \quad F_s(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} f_i(x_i)$$
$$\text{s.t. } x_1 = \cdots = x_N$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^{Nd}$$

$$\min_{\mathbf{x}} \quad F(\mathbf{x})$$
$$\text{s.t. } \mathbf{x} \in \mathrm{C}$$

Generalized Decentralized Problem

Consensus subspace $C = \{\mathbf{x} \in \mathbb{R}^{Nd} \mid x_1 = \cdots = x_N \in \mathbb{R}^d\}$

? How can we represent $\mathbf{x}^* \in \mathrm{C}$ in PEP?

Change of variables

to decouple consensus subspace and its orthogonal complement

Consensus subspace $C = \{\mathbf{x} \in \mathbb{R}^{Nd} \mid x_1 = \cdots = x_N \in \mathbb{R}^d\}$ size d

$C^\perp$ orthogonal complement of C size $(N-1)d$

Change of variables

$$_b\mathbf{x} = \begin{bmatrix} \bar{x} \\ x_\perp \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \quad \in \mathbb{R}^{Nd}$$

Change of variables

to decouple consensus subspace and its orthogonal complement

Consensus subspace $C = \{\mathbf{x} \in \mathbb{R}^{Nd} \mid x_1 = \cdots = x_N \in \mathbb{R}^d\}$ size d

$C^\perp$ orthogonal complement of C size $(N-1)d$

Change of variables

$$ {}_b\mathbf{x} = \begin{bmatrix} \bar{x} \\ x_\perp \end{bmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} Q_\parallel^T \\ Q_\perp^T \end{bmatrix} \mathbf{x} $$

with $Q_\parallel^T$ such that $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$

⇕ $Q = [Q_\parallel \; Q_\perp]$ is orthogonal

$$ \mathbf{x} = \sqrt{N}\,\mathrm{Q}\;{}_b\mathbf{x} = \underbrace{\sqrt{N}\,\mathrm{Q}_\parallel \bar{x}}_{\in C} + \underbrace{\sqrt{N}\,\mathrm{Q}_\perp x_\perp}_{\in C^\perp} $$

$$ \mathbf{x} \in C \quad \Leftrightarrow \quad x_\perp = 0 $$

Change of variables in the decentralized problem

Change of function $\tilde{\mathrm{F}}\colon \mathbb{R}^{Nd} \to \mathbb{R}$, $\qquad \tilde{\mathrm{F}}({}_b\mathbf{x}) = \mathrm{F}\left(\sqrt{N}Q\,{}_b\mathbf{x}\right) = \mathrm{F}(\mathbf{x})$

$$\nabla\tilde{\mathrm{F}}({}_b\mathbf{x}) = \begin{bmatrix} \nabla_{\parallel}\tilde{F}({}_b\mathbf{x}) \\ \nabla_{\perp}\tilde{F}({}_b\mathbf{x}) \end{bmatrix}$$

$$(GDP) \quad \min_{\mathbf{x}} \; F(\mathbf{x}) \quad \text{s.t.} \; \mathbf{x} \in \mathrm{C} \qquad \Longleftrightarrow \qquad \min_{{}_b\mathbf{x}} \; \tilde{F}({}_b\mathbf{x}) \quad \text{s.t.} \; x_{\perp} = 0 \quad ({}_bGDP)$$

Optimality conditions

${}_b\mathbf{x}^*$ is optimal solution of $({}_bGDP)$ iff

$x_{\perp}^* = 0$ and $\nabla_{\parallel}\tilde{F}({}_b\mathbf{x}^*) = 0$

Change of variables in the decentralized algorithm

Decouple the updates

✓ Gradient evaluations $\nabla \tilde{\mathrm{F}}({}_b\mathbf{x}) = \begin{bmatrix} \nabla_{\parallel} \tilde{F}({}_b\mathbf{x}) \\ \nabla_{\perp} \tilde{F}({}_b\mathbf{x}) \end{bmatrix}$

✓ Linear combinations $\alpha\mathbf{x} + \beta\mathbf{y} + \gamma\mathbf{g} = \alpha\ {}_b\mathbf{x} + \beta\ {}_b\mathbf{y} + \gamma\ {}_b\mathbf{g}$

$$\alpha \begin{bmatrix} \bar{x} \\ x_{\perp} \end{bmatrix} + \beta \begin{bmatrix} \bar{y} \\ y_{\perp} \end{bmatrix} + \gamma \begin{bmatrix} \bar{g} \\ g_{\perp} \end{bmatrix}$$

✓ Consensus step $\mathbf{y}^k = (W \otimes I_d)\mathbf{x}^{\boldsymbol{k}}$ with W symmetric doubly stochastic $\lambda(W) \in [\lambda^-, \lambda^+]$

doubly stochastic ➡ $\bar{y}^k = \bar{x}^k$

spectrum $\lambda(W)$ ➡ $y_{\perp}^k = \widetilde{W} x_{\perp}^k$ with $\widetilde{W}$ symmetric $\lambda(\widetilde{W}) \in [\lambda^-, \lambda^+]$

Agent-Independent PEP formulation

(example for DGD)

$$\max_{F,\mathbf{x}^k,\mathbf{y}^k,\mathbf{x}^*} \quad F(\bar{x}^K \mathbf{1}) - F(\mathbf{x}^*)$$

F convex with bounded subgradients

$$\frac{1}{N}\|\mathbf{x}^0 - \mathbf{x}^*\|^2 \leq R$$

$\mathbf{x}^*$ optimality condition

DGD
$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^{\boldsymbol{k}} \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \nabla F(\mathbf{x}^k) \end{cases}$$

for W
- symmetric
- generalized doubly stochastic
- $\lambda(W) \in [\lambda^-, \lambda^+]$

$$\max_{\substack{\tilde{F},\ \bar{x}^k, \bar{y}^k \\ x_\perp^k, y_\perp^k}} \quad \tilde{F}\left(\begin{bmatrix} \bar{x}^K \\ 0 \end{bmatrix}\right) - \tilde{F}\left(\begin{bmatrix} \bar{x}^* \\ x_\perp^* \end{bmatrix}\right)$$

Agent-Independent PEP formulation

(example for DGD)

$$\max_{F,\mathbf{x}^k,\mathbf{y}^k,\mathbf{x}^*} \quad F(\bar{x}^K\mathbf{1}) - F(\mathbf{x}^*)$$

F convex with bounded subgradients

$$\frac{1}{N}\|\mathbf{x}^0 - \mathbf{x}^*\|^2 \le R$$

$\mathbf{x}^*$ optimality condition

DGD

$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^{\boldsymbol{k}} \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N\, \boldsymbol{\nabla} F(\mathbf{x}^k) \end{cases}$$

for W symmetric
generalized doubly stochastic
$\lambda(W) \in [\lambda^-, \lambda^+]$

$$\max_{\substack{\tilde{F},\ \bar{x}^k, \bar{y}^k \\ x_\perp^k, y_\perp^k}} \quad \tilde{F}\left(\begin{bmatrix} \bar{x}^K \\ 0 \end{bmatrix}\right) - \tilde{F}\left(\begin{bmatrix} \bar{x}^* \\ x_\perp^* \end{bmatrix}\right)$$

$\tilde{F}$ convex with bounded subgradients

$$\|\bar{x}^0 - \bar{x}^*\|^2 + \left\|x_\perp^0 - x_\perp^*\right\|^2 \le R^2$$

$$x_\perp^* = 0 \quad \text{and} \quad \nabla_\parallel \tilde{F}\left(\begin{bmatrix} \bar{x}^* \\ x_\perp^* \end{bmatrix}\right) = 0$$

consensus step

$$\begin{cases} \bar{y}^k = \bar{x}^k \\ y_\perp^k = \widetilde{W} x_\perp^k \end{cases} \qquad \lambda(\widetilde{W}) \in [\lambda^-, \lambda^+]$$

gradient step

$$\begin{cases} \bar{x}^{k+1} = \bar{y}^k - \alpha \nabla_\parallel \tilde{F}\left(\begin{bmatrix} \bar{x}^k \\ x_\perp^k \end{bmatrix}\right) \\ x_\perp^{k+1} = y_\perp^k - \alpha \nabla_\perp \tilde{F}\left(\begin{bmatrix} \bar{x}^k \\ x_\perp^k \end{bmatrix}\right) \end{cases}$$

Agent-Independent PEP formulation

(example for DGD)

$$\max_{F, \mathbf{x}^k, \mathbf{y}^k, \mathbf{x}^*} \quad F(\bar{x}^K \mathbf{1}) - F(\mathbf{x}^*)$$

F convex with bounded subgradients

$$\frac{1}{N} \|\mathbf{x}^0 - \mathbf{x}^*\|^2 \leq R$$

$\mathbf{x}^*$ optimality condition

DGD

$$\begin{cases} \mathbf{y}^k = (W \otimes I_d)\mathbf{x}^k \\ \mathbf{x}^{k+1} = \mathbf{y}^k - \alpha N \nabla F(\mathbf{x}^k) \end{cases}$$

$$\max_{\substack{\tilde{F},\ \bar{x}^k, \bar{y}^k \\ x_\perp^k, y_\perp^k}} \quad \tilde{F}\left(\begin{bmatrix} \bar{x}^K \\ 0 \end{bmatrix}\right) - \tilde{F}\left(\begin{bmatrix} \bar{x}^* \\ x_\perp^* \end{bmatrix}\right)$$

$\tilde{F}$ convex with bounded subgradients

$$\|\bar{x}^0 - \bar{x}^*\|^2 + \|x_\perp^0 - x_\perp^*\|^2 \leq R^2$$

$$x_\perp^* = 0 \quad \text{and} \quad \nabla_\parallel \tilde{F}\left(\begin{bmatrix} \bar{x}^* \\ x_\perp^* \end{bmatrix}\right) = 0$$

consensus step

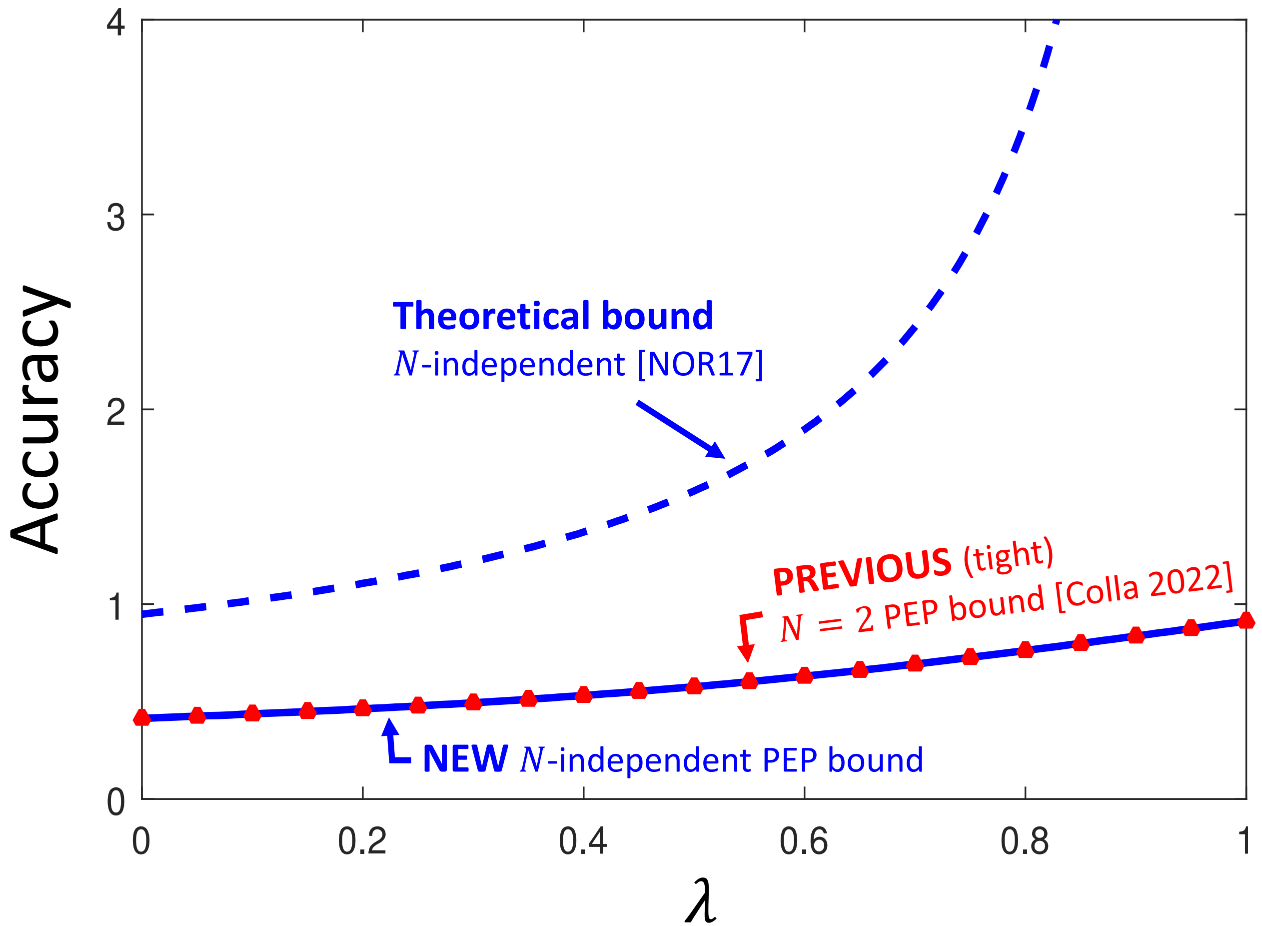
$$\begin{cases} \bar{y}^k = \bar{x}^k \\ y_\perp^k = \tilde{W} x_\perp^k \end{cases} \qquad \lambda(\tilde{W}) \in [\lambda^-, \lambda^+]$$

gradient step

$$\begin{cases} \bar{x}^{k+1} = \bar{y}^k - \alpha \nabla_\parallel \tilde{F}\left(\begin{bmatrix} \bar{x}^k \\ x_\perp^k \end{bmatrix}\right) \\ x_\perp^{k+1} = y_\perp^k - \alpha \nabla_\perp \tilde{F}\left(\begin{bmatrix} \bar{x}^k \\ x_\perp^k \end{bmatrix}\right) \end{cases}$$

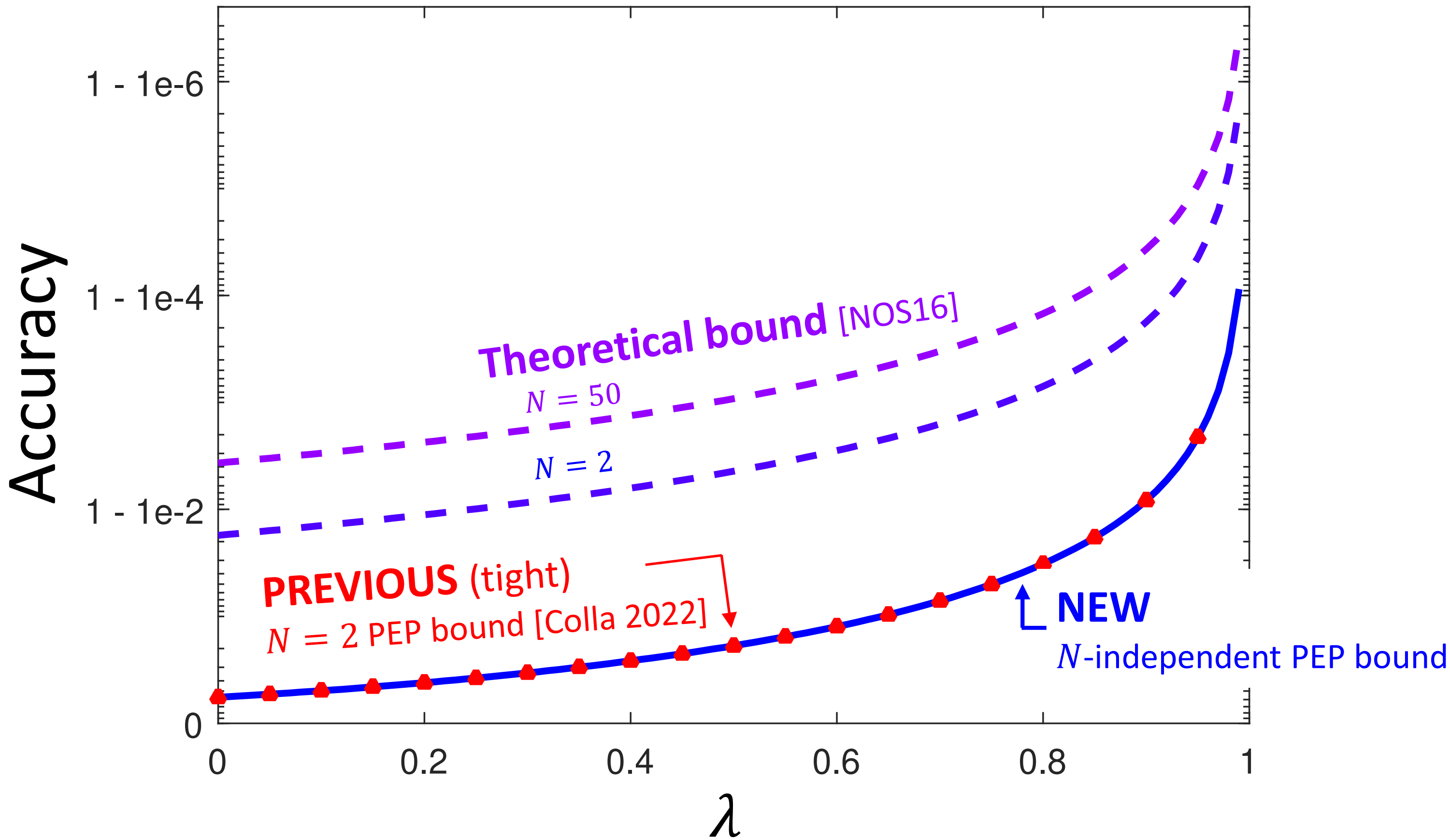
Can be solved with proper discretization and **SDP reformulation**

Tightness analysis - DGD



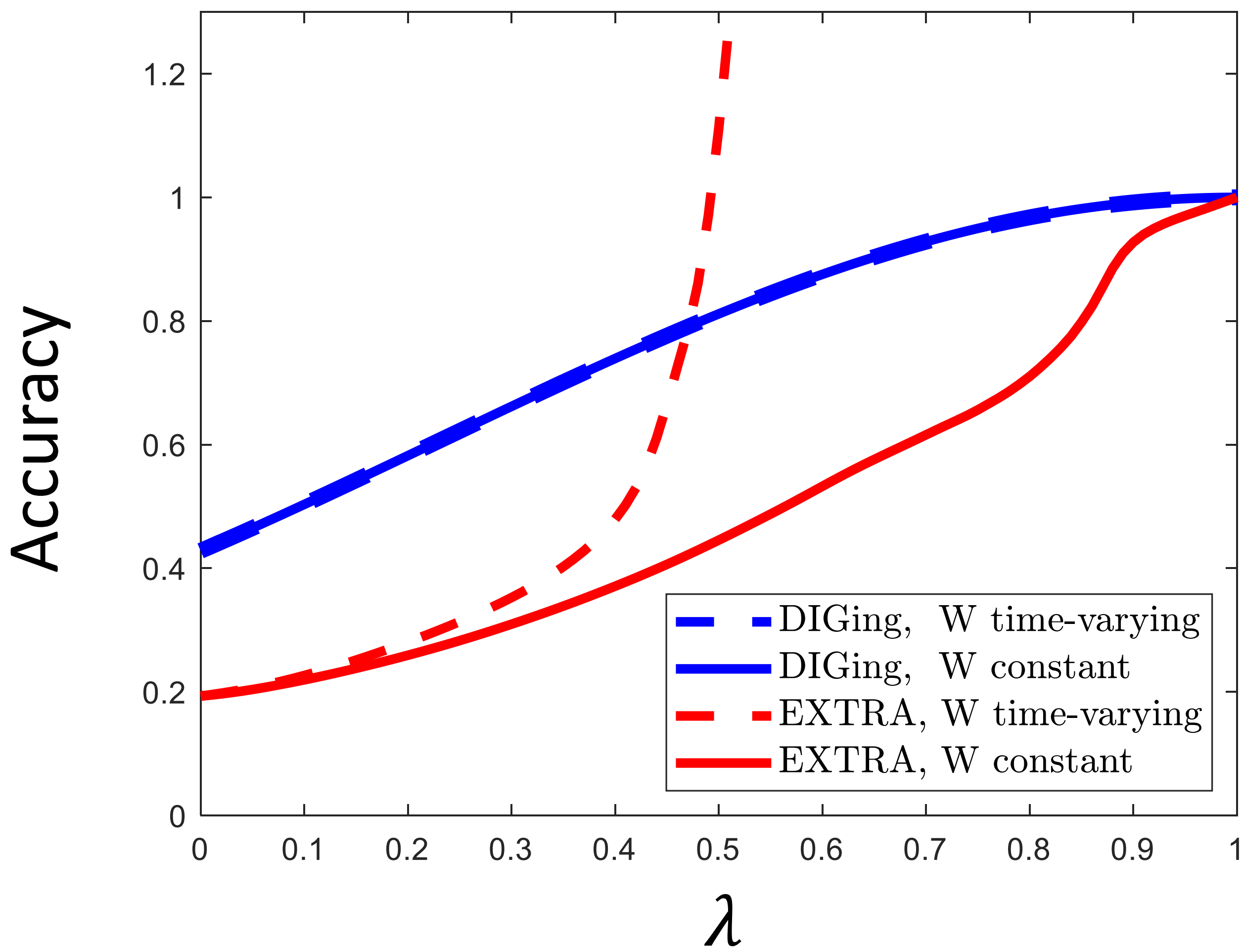
For $K = 10$ iterations, and $\lambda(W) \in [-\lambda, \lambda]$
With convex local functions with bounded subgradients ($R = 1$)

Tightness analysis - DIGing



For $K = 10$ iterations, $\mu = 0.1$, $L = 1$ and $\lambda(W) \in [-\lambda, \lambda]$.

Algorithms comparison



For $K = 10$ iterations, $\mu = 0.1$, $L = 1$ and $\lambda(W) \in [-\lambda, \lambda]$.

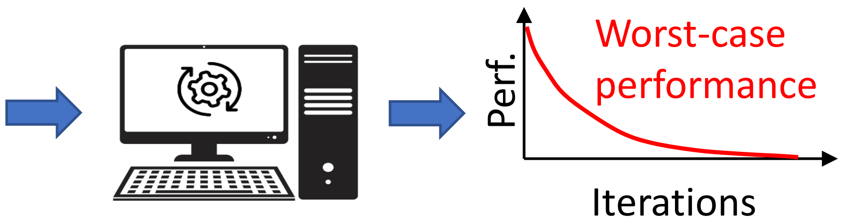
Conclusion

Sébastien Colla

Automatic tool for accurate **performance estimation** of decentralized optimization methods

```
function MyDecentralizedAlgo()
    N = 10; % number of agents
    x0 = init(N); % Initial point
    x = x0;

    for i=1:niter
        % any local computations
        % and local communications
        x = update(x,N);
    end
end
```



PEP idea: worst-cases are solutions of optimization problems

Agent-independent spectral PEP formulation

- ✓ Size problem independent of N
- ✓ Appears to be tight
- ✓ Improves on the literature bounds

We can answer a large diversity of (new) questions

Future works

- ❑ Understand tightness of this formulation
- ❑ Other classes of averaging matrices

References

[CH-CDC22] S. Colla, J. M. Hendrickx, “Automated Performance Estimation for Decentralized Optimization via Network Size Independent Problems”, CDC 2022.

[Colla 2022] S. Colla, J. M. Hendrickx, “Automatic Performance Estimation for Decentralized Optimization”, preprint 2022.

[NOR17] A. Nedic, A. Olshevsky, and M. G. Rabbat, “Network topology and communication computation tradeoffs in decentralized optimization”, 2017.

[NOS16] A. Nedic, A. Olshevsky, and W. Shi, “Achieving geometric convergence for distributed optimization over time-varying graphs,” SIAM Journal on Optimization, 2016.