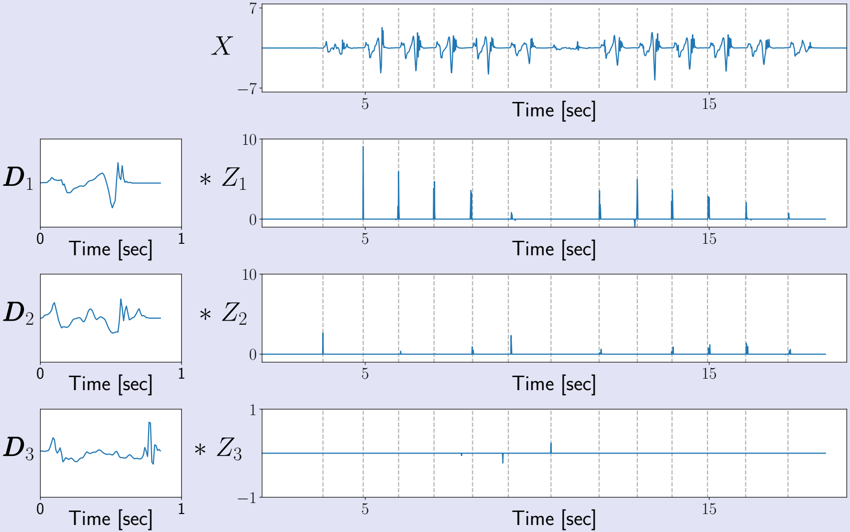
DICOD: Distributed Coordinate Descent for Convolutional Sparse Coding

Thomas Moreau, Laurent Oudre, Nicolas Vayatis
CMLA, ENS Paris-Saclay

Convolutional Dictionary Learning



Convolutional Sparse coding

$\rightarrow$ $\boldsymbol{D}$ fixed in $\mathbb{R}^{K \times W}$, update Z in $\mathbb{R}^{K \times L}$ with X of length $T = L + W - 1$

$$Z^* = \operatorname*{argmin}_Z \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda \|Z\|_1$$

Classical Algorithms:

- Coordinate Descent (CD)
  [Friedman et al., 2007; Kavukcuoglu et al., 2010]
- Fast Iterative Soft-Thresholding Algorithm (FISTA)
  [Beck and Teboulle, 2009; Chalasani et al., 2013]
- Alternated Direction Method of Multiplier (ADMM)
  [Gabay and Mercier, 1976; Bristow et al., 2013]

$\Rightarrow$ Do not scale well with long signals $T \gg 1$.

Coordinate Descent (CD)

$\rightarrow$ $\boldsymbol{D}$ fixed, update Z

$$Z^* = \underset{Z}{\operatorname{argmin}} \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda \|Z\|_1$$

Coordinate Descent:

Select a coordinate (k_0, t_0) to update.

- Cyclic/Random updates; $\mathcal{O}(1)$,
- Greedy updates; $\mathcal{O}(KL)$.

Update the value for $Z_{k_0}[t_0]$

- For convolutional setting; $\mathcal{O}(KW)$.
- Local operation: only impact a time segment of size $2W - 1$

[Friedman et al., 2007; Nesterov, 2010; Osher and Li, 2009]

Locally Greedy Coordinate Descent

Key idea: Matches CD selection complexity with update complexity.

$\Rightarrow$ Select the coordinate in a locally greedy fashion.

Locally Greedy Coordinate Descent

Key idea: Matches CD selection complexity with update complexity.

$\Rightarrow$ Select the coordinate in a locally greedy fashion.

Take the update greedily in a subsegment of the signal,

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\} .$$

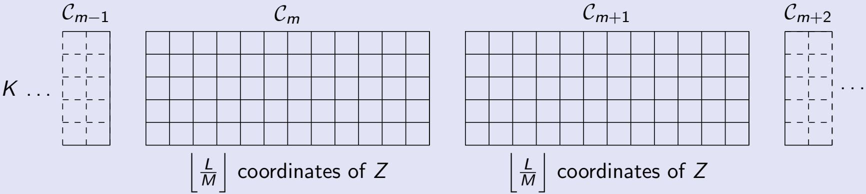
This is efficient when $M = \mathcal{O}\left(\frac{L}{M}\right)$ as both part of the CD algorithm have the same computational complexity $\mathcal{O}\left(KW\right)$.

Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

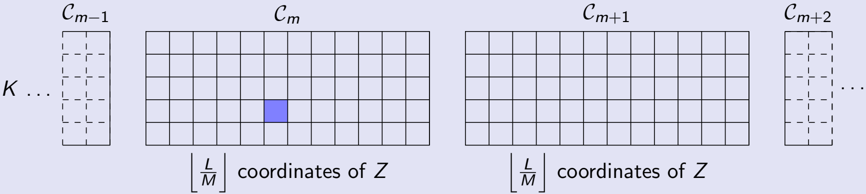


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

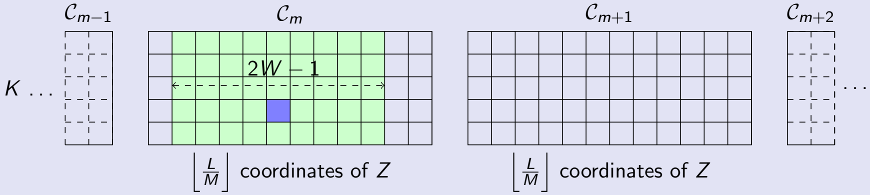


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

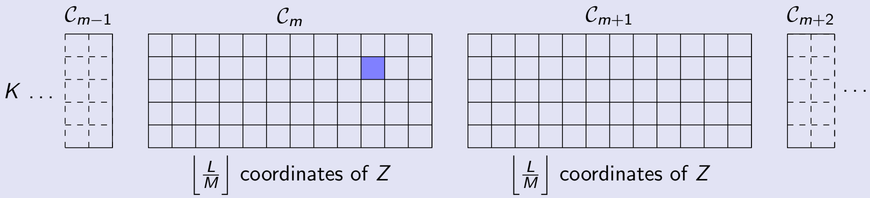


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\} .$$

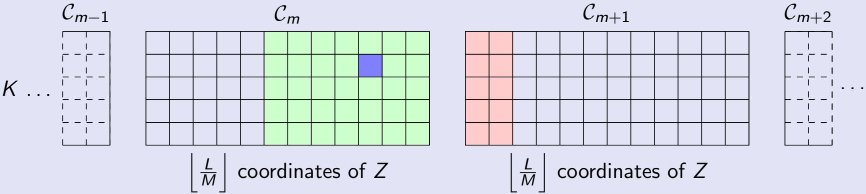


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

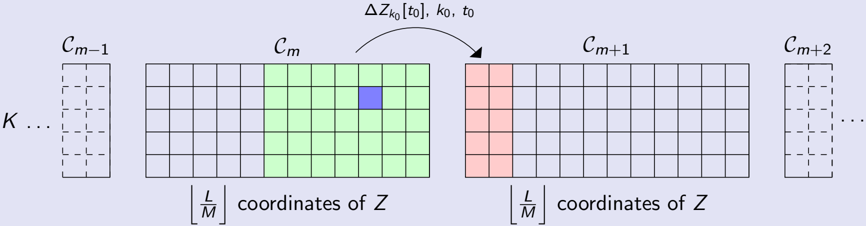


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m\left\lfloor\frac{L}{M}\right\rfloor, \dots (m+1)\left\lfloor\frac{L}{M}\right\rfloor - 1 \right\}.$$

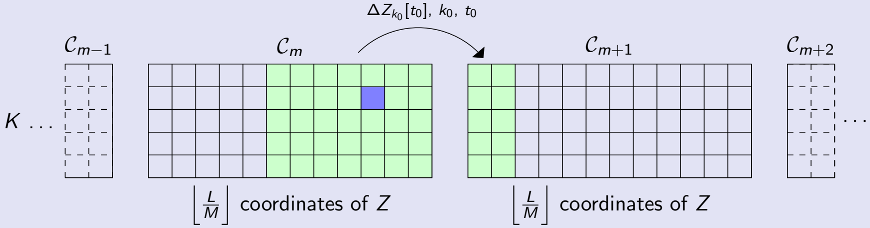


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

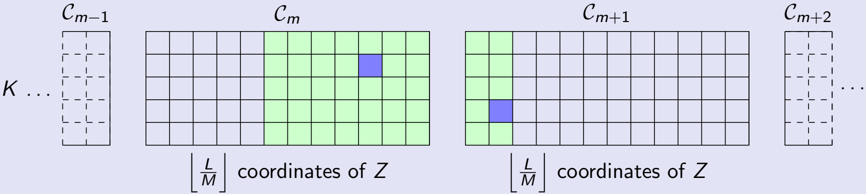


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

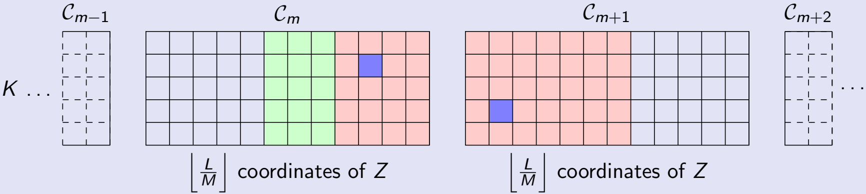


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

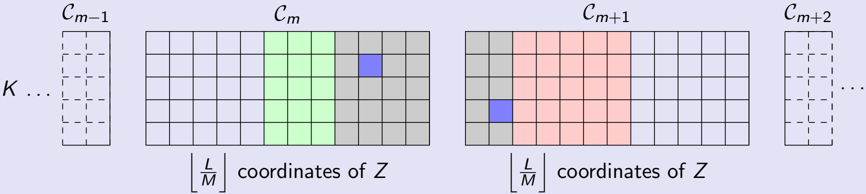


Distributed Convolutional Coordinate Descent (DICOD)

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a contiguous segment

$$\mathcal{C}_m = \left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}.$$

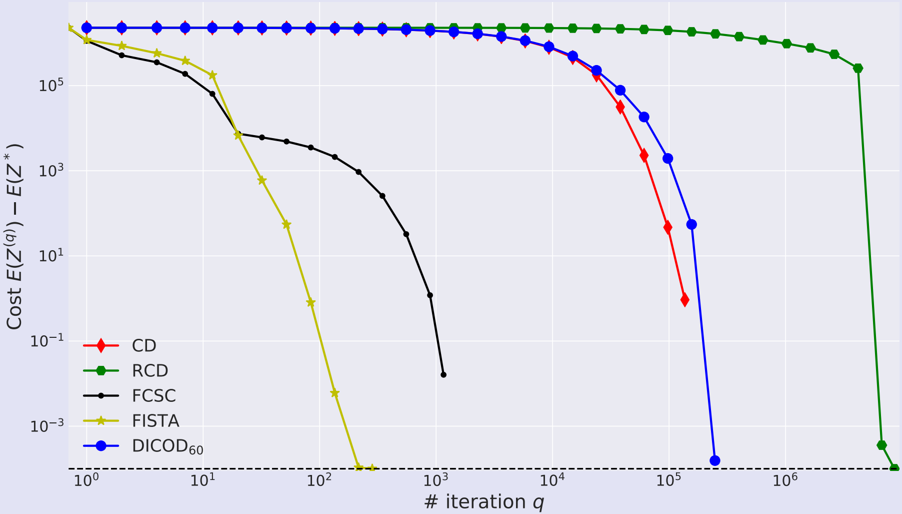


Distributed Convolutional Coordinate Descent (DICOD)

Theoretical results:

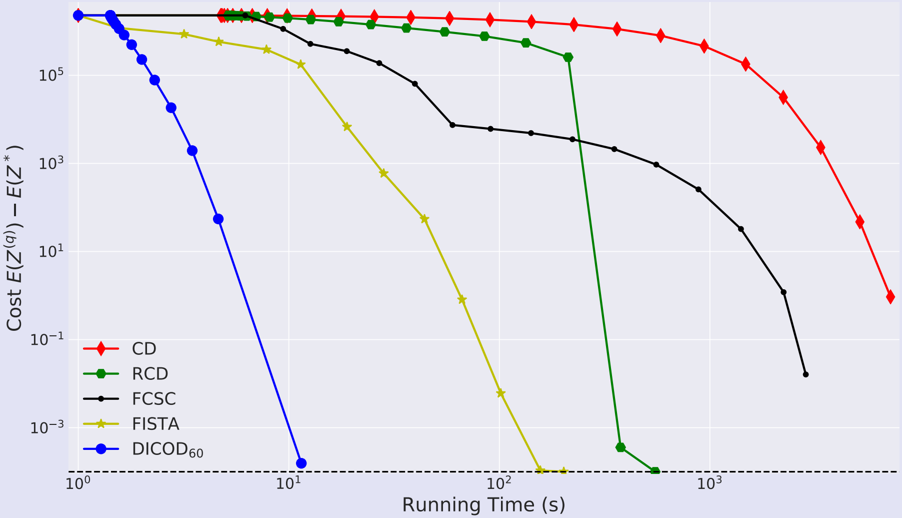
- Guaranteed convergence: We show that the interferences do not break the algorithm. even when run in a synchronous setting
- Superlinear speedup: More than twice as fast when doubling M.
  - Speed-up due to the parallelization on M cores.
  - Speed-up to the complexity reduction of each iteration.

Numerical convergence



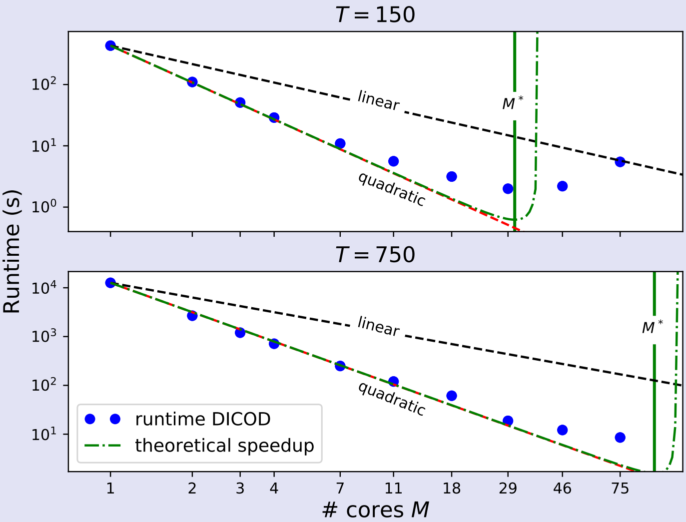
Cost as a function of the iterations

Numerical convergence



Cost as a function of the runtime

Super-linear scaling of DICOD



Runtime as a function of the number of cores M

DICOD: Distributed Coordinate Descent for Convolutional Sparse Coding

Use the **structure** of convolutional LASSO to derive a parallel algorithm based on **CD** with

- Asynchronous updates
- Efficient communications
- No exogenous parameters
- Maximal update rules

github.com/tommoral/dicod

@tomamoral (note the extra 'a')

DICOD: Distributed Coordinate Descent for Convolutional Sparse Coding

Use the **structure** of convolutional LASSO to derive a parallel algorithm based on **CD** with

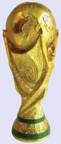
- Asynchronous updates
- Efficient communications
- No exogenous parameters
- Maximal update rules

github.com/tommoral/dicod

@tomamoral (note the extra 'a')

Thanks!

Poster#33 tonight!



It's coming home!
Allez les bleus!