D^4L: Decentralized Dictionary Learning over Dynamic Digraphs

Amir Daneshmand[†], Ying Sun[†], Gesualdo Scutari[†]
Francisco Facchinei[‡]

[†]Purdue University,
[‡] University of Rome La Sapienza

The 42nd IEEE International Conference on Acoustics, Speech and Signal Processing

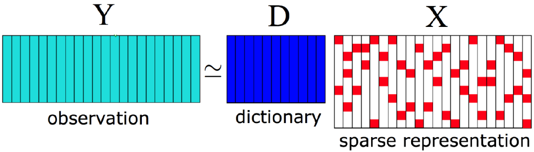
March 5-9, 2017

Outline

- Motivation and problem formulation
- Main challenges and literature overview
- Algorithmic framework (bottom-up approach)
- Numerical results
- Conclusions

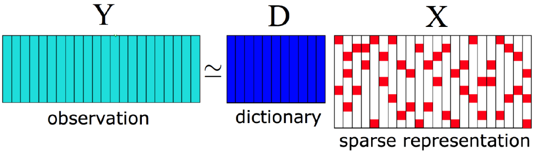
Problem of Study

Dictionary Learning



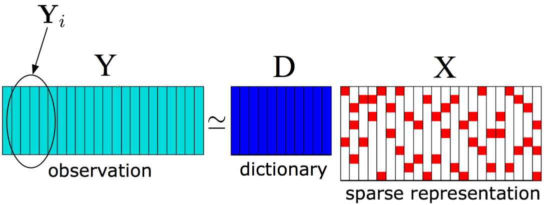
Problem of Study

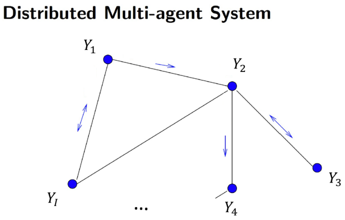
Dictionary Learning



Applications: estimation, image denoising/debluring/inpainting, superresolution, dimensionality reduction, bi-clustering, feature-extraction, classification, prediction, ...

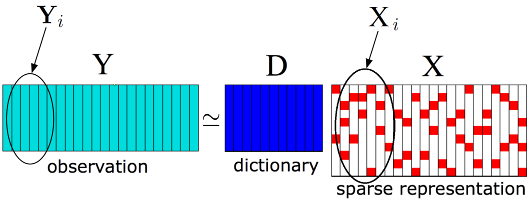
Problem of Study

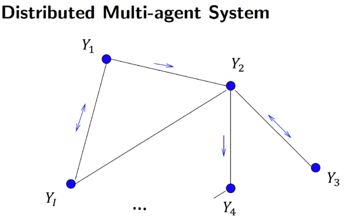




Goal: designing a distributed algorithm over a network

Problem of Study





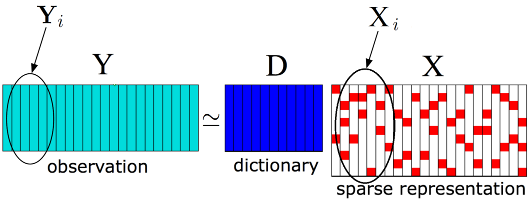
Goal: designing a distributed algorithm over a network

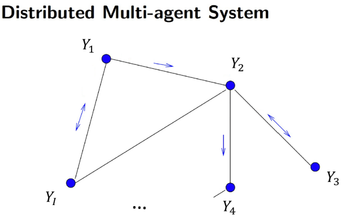
Problem of Study

Distributed Dictionary Learning

$$\underset{\mathbf{D}\in\mathcal{D},\, \mathbf{X}\triangleq(\mathbf{X}_i)_{i=1}^{I}}{\text{minimize}} \quad \sum_{i=1}^{I} \underbrace{\frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|_F^2}_{\triangleq f_i(\mathbf{D},\mathbf{X}_i)} + \underbrace{\lambda\|\mathbf{X}_i\|_1 + \frac{\mu}{2}\|\mathbf{X}_i\|_F^2}_{\triangleq g_i(\mathbf{X}_i)}$$

where $\mu, \lambda > 0$ and $\mathcal{D}$ is a compact convex set.



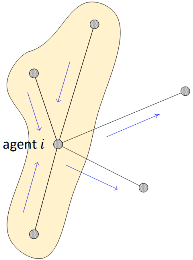


Goal: designing a distributed algorithm over a network

Problem of Study

Network Model

- Dynamic network topology: Agents are embedded in a possibly *time-varying directed* communication *graph* $\mathcal{G}[\nu]$
    - The vertices of $\mathcal{G}[\nu]$ correspond to the agents
    - The set of directed edges may change over the time
    - $\mathcal{N}_i[\nu]$: set of agents that can send information to agent i at time ν including node i



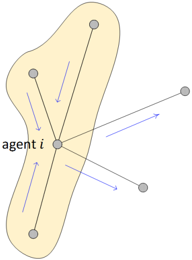
Assumptions on the network & agents' knowledge

- T-strongly connected digraphs: $\exists T \in \mathbb{N}_+$ such that the graph $\left([I], \bigcup_{t=0,\ldots,T-1} \mathcal{G}[t+\nu]\right)$ is connected for all $\nu \geq 0$.
- Local information: each agent i knows its f_i and g_i but not $\sum_{j \neq i} f_j$
- Local communications: agents can *only* receive information from their "neighbors"

Problem of Study

Network Model

- Dynamic network topology: Agents are embedded in a possibly *time-varying directed* communication *graph* $\mathcal{G}[\nu]$
  - The vertices of $\mathcal{G}[\nu]$ correspond to the agents
  - The set of directed edges may change over the time
  - $\mathcal{N}_i[\nu]$: set of agents that can send information to agent i at time ν including node i



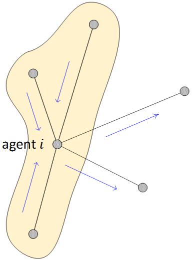
Assumptions on the network & agents' knowledge

- T-strongly connected digraphs: $\exists T \in \mathbb{N}_+$ such that the graph $\left([I], \bigcup_{t=0,\ldots,T-1} \mathcal{G}[t+\nu]\right)$ is connected for all $\nu \geq 0$.
- Local information: each agent i knows its f_i and g_i but not $\sum_{j \neq i} f_j$
- Local communications: agents can *only* receive information from their "neighbors"

Problem of Study

Network Model

- Dynamic network topology: Agents are embedded in a possibly *time-varying directed* communication *graph* $\mathcal{G}[\nu]$
  - The vertices of $\mathcal{G}[\nu]$ correspond to the agents
  - The set of directed edges may change over the time
  - $\mathcal{N}_i[\nu]$: set of agents that can send information to agent i at time ν including node i



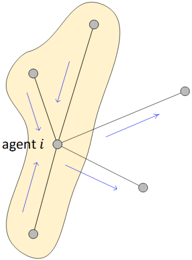
Assumptions on the network & agents' knowledge

- T-strongly connected digraphs: $\exists T \in \mathbb{N}_+$ such that the graph $\left([I], \bigcup_{t=0,\ldots,T-1} \mathcal{G}[t+\nu]\right)$ is connected for all $\nu \geq 0$.
- Local information: each agent i knows its f_i and g_i but not $\sum_{j \neq i} f_j$
- Local communications: agents can *only* receive information from their "neighbors"

Problem of Study

Network Model

- Dynamic network topology: Agents are embedded in a possibly *time-varying directed* communication *graph* $\mathcal{G}[\nu]$
  - The vertices of $\mathcal{G}[\nu]$ correspond to the agents
  - The set of directed edges may change over the time
  - $\mathcal{N}_i[\nu]$: set of agents that can send information to agent i at time ν including node i



Assumptions on the network & agents' knowledge

- T-strongly connected digraphs: $\exists T \in \mathbb{N}_+$ such that the graph $\left([I], \bigcup_{t=0,\ldots,T-1} \mathcal{G}[t+\nu]\right)$ is connected for all $\nu \geq 0$.
- Local information: each agent i knows its f_i and g_i but not $\sum_{j \neq i} f_j$
- Local communications: agents can *only* receive information from their "neighbors"

Literature Review and Challenges

Distributed Dictionary Learning

- Ad-hoc schemes for distributed DL problem:[Lia-Zha-Zen'14], [Che-Zai-Say'15], [Wai-Cha-Sca'15], [Cha-Ric'13], [Kop-Gar-War-Stum-Rib'15,'16]
  - Time invariant undirected graphs
  - No proof of convergence to stationary solutions of DL problem
- Distributed Nonconvex Multiagent Optimization: [DiLor-Scu'15]
  - Can not handle both $\mathbf{X}_i$'s (private variables) and $\mathbf{D}$ (shared variables)
  - Some technical conditions are not satisfied; e.g., ∇f_i is NOT bounded or Lipschitz continuous over the feasible set
- Our contribution: extending [Dan-Scu-Fac, Asilomar'16] to deal with time-varying digraphs
- Full picture: A. Daneshmand, Y. Sun, G. Scutari, F. Facchinei, B. M. Sadler, *"Decentralized Dictionary learning over Dynamic Digraphs"*, J. Mach. Learn. Res. (under review). Available online.

Literature Review and Challenges

Distributed Dictionary Learning

- Ad-hoc schemes for distributed DL problem:[Lia-Zha-Zen'14], [Che-Zai-Say'15], [Wai-Cha-Sca'15], [Cha-Ric'13], [Kop-Gar-War-Stum-Rib'15,'16]
  - Time invariant undirected graphs
  - No proof of convergence to stationary solutions of DL problem
- Distributed Nonconvex Multiagent Optimization: [DiLor-Scu'15]
  - Can not handle both $\mathbf{X}_i$'s (private variables) and $\mathbf{D}$ (shared variables)
  - Some technical conditions are not satisfied; e.g., ∇f_i is NOT bounded or Lipschitz continuous over the feasible set
- Our contribution: extending [Dan-Scu-Fac, Asilomar'16] to deal with time-varying digraphs
- Full picture: A. Daneshmand, Y. Sun, G. Scutari, F. Facchinei, B. M. Sadler, *"Decentralized Dictionary learning over Dynamic Digraphs"*, J. Mach. Learn. Res. (under review). Available online.

Literature Review and Challenges

Distributed Dictionary Learning

- Ad-hoc schemes for distributed DL problem:[Lia-Zha-Zen'14], [Che-Zai-Say'15], [Wai-Cha-Sca'15], [Cha-Ric'13], [Kop-Gar-War-Stum-Rib'15,'16]
  - Time invariant undirected graphs
  - No proof of convergence to stationary solutions of DL problem
- Distributed Nonconvex Multiagent Optimization: [DiLor-Scu'15]
  - Can not handle both $\mathbf{X}_i$'s (private variables) and $\mathbf{D}$ (shared variables)
  - Some technical conditions are not satisfied; e.g., ∇f_i is NOT bounded or Lipschitz continuous over the feasible set
- Our contribution: extending [Dan-Scu-Fac, Asilomar'16] to deal with time-varying digraphs
- Full picture: A. Daneshmand, Y. Sun, G. Scutari, F. Facchinei, B. M. Sadler, *"Decentralized Dictionary learning over Dynamic Digraphs"*, J. Mach. Learn. Res. (under review). Available online.

Literature Review and Challenges

Distributed Dictionary Learning

- Ad-hoc schemes for distributed DL problem:[Lia-Zha-Zen'14], [Che-Zai-Say'15], [Wai-Cha-Sca'15], [Cha-Ric'13], [Kop-Gar-War-Stum-Rib'15,'16]
  - Time invariant undirected graphs
  - No proof of convergence to stationary solutions of DL problem
- Distributed Nonconvex Multiagent Optimization: [DiLor-Scu'15]
  - Can not handle both $\mathbf{X}_i$'s (private variables) and $\mathbf{D}$ (shared variables)
  - Some technical conditions are not satisfied; e.g., ∇f_i is NOT bounded or Lipschitz continuous over the feasible set
- Our contribution: extending [Dan-Scu-Fac, Asilomar'16] to deal with time-varying digraphs
- Full picture: A. Daneshmand, Y. Sun, G. Scutari, F. Facchinei, B. M. Sadler, *"Decentralized Dictionary learning over Dynamic Digraphs"*, J. Mach. Learn. Res. (under review). Available online.

Algorithmic Design

Main Idea

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)} + g_i(\mathbf{X}_i)\right\},$$

Each agent i: maintains a *local copy* $\mathbf{D}_{(i)}$ of $\mathbf{D}$, and controls $\mathbf{X}_i$:

1. [local optimization]: optimizes $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ alternatingly by solving strongly convex problems
2. [consensus update]: exchanges the local copies $\mathbf{D}_{(i)}$ to force consensus

Algorithmic Design

Step 1: Local Optimization

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\Big\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\Big\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i)+\sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j)+\sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

[optimization of $\mathbf{D}_{(i)}$]: Given $(\mathbf{D}_{(i)}^{\nu},\mathbf{X}_i^{\nu})$, each agent i updates $\mathbf{D}_{(i)}$ setting $\mathbf{X}_i=\mathbf{X}_i^{\nu}$ and solving

$$\widetilde{\mathbf{D}}_{(i)}^{\nu} \triangleq \underset{\mathbf{D}_{(i)}\in\mathcal{D}}{\operatorname{argmin}}\left\{f_i\left(\mathbf{D}_{(i)},\mathbf{X}_i^{\nu}\right)+\frac{\tau_{D,i}^{\nu}}{2}\|\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\|_F^2+\left\langle\widetilde{\mathbf{\Pi}}_i^{\nu},\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\right\rangle\right\}$$

$$\mathbf{U}_{(i)}^{\nu}=\mathbf{D}_{(i)}^{\nu}+\gamma^{\nu}(\widetilde{\mathbf{D}}_{(i)}^{\nu}-\mathbf{D}_{(i)}^{\nu})$$

where $\tau_{D,i}^{\nu}>0$ and $\widetilde{\mathbf{\Pi}}_i^{\nu}$ aims to $\widetilde{\mathbf{\Pi}}_i^{\nu}\to\sum_{j\neq i}\nabla f_j(\mathbf{D}_{(i)}^{\nu},\mathbf{X}_j^{\nu})$.

[optimization of $\mathbf{X}_i$]: Given $(\mathbf{U}_{(i)}^{\nu},\mathbf{X}_i^{\nu})$, each agent i updates $\mathbf{X}_i$ setting $\mathbf{D}_i=\mathbf{U}_{(i)}^{\nu}$ and solving

$$\mathbf{X}_i^{\nu+1}\triangleq\underset{\mathbf{X}_i}{\operatorname{argmin}}\left\{f_i(\mathbf{U}_{(i)}^{\nu},\mathbf{X}_i)+\frac{\tau_{X,i}^{\nu}}{2}\|\mathbf{X}_i-\mathbf{X}_i^{\nu}\|_F^2+g_i(\mathbf{X}_i)\right\},$$

with $\tau_{X,i}^{\nu}>0$.

Algorithmic Design

Step 1: Local Optimization

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\right\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i) + \sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j) + \sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

[optimization of $\mathbf{D}_{(i)}$]: Given $(\mathbf{D}^{\nu}_{(i)},\mathbf{X}^{\nu}_i)$, each agent i updates $\mathbf{D}_{(i)}$ setting $\mathbf{X}_i=\mathbf{X}^{\nu}_i$ and solving

$$\widetilde{\mathbf{D}}^{\nu}_{(i)} \triangleq \underset{\mathbf{D}_{(i)}\in\mathcal{D}}{\text{argmin}}\left\{f_i\left(\mathbf{D}_{(i)},\mathbf{X}^{\nu}_i\right)+\frac{\tau^{\nu}_{D,i}}{2}\|\mathbf{D}_{(i)}-\mathbf{D}^{\nu}_{(i)}\|_F^2+\left\langle\widetilde{\mathbf{\Pi}}^{\nu}_i,\mathbf{D}_{(i)}-\mathbf{D}^{\nu}_{(i)}\right\rangle\right\}$$

$$\mathbf{U}^{\nu}_{(i)} = \mathbf{D}^{\nu}_{(i)} + \gamma^{\nu}(\widetilde{\mathbf{D}}^{\nu}_{(i)}-\mathbf{D}^{\nu}_{(i)})$$

where $\tau^{\nu}_{D,i}>0$ and $\widetilde{\mathbf{\Pi}}^{\nu}_i$ aims to $\widetilde{\mathbf{\Pi}}^{\nu}_i \to \sum_{j\neq i}\nabla f_j(\mathbf{D}^{\nu}_{(i)},\mathbf{X}^{\nu}_j)$.

[optimization of $\mathbf{X}_i$]: Given $(\mathbf{U}^{\nu}_{(i)},\mathbf{X}^{\nu}_i)$, each agent i updates $\mathbf{X}_i$ setting $\mathbf{D}_i=\mathbf{U}^{\nu}_{(i)}$ and solving

$$\mathbf{X}^{\nu+1}_i \triangleq \underset{\mathbf{X}_i}{\text{argmin}}\left\{f_i(\mathbf{U}^{\nu}_{(i)},\mathbf{X}_i)+\frac{\tau^{\nu}_{X,i}}{2}\|\mathbf{X}_i-\mathbf{X}^{\nu}_i\|_F^2+g_i(\mathbf{X}_i)\right\},$$

with $\tau^{\nu}_{X,i}>0$.

Algorithmic Design

Step 1: Local Optimization

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\right\}, \quad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i)+\sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j)+\sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

[optimization of $\mathbf{D}_{(i)}$]: Given $(\mathbf{D}_{(i)}^{\nu},\mathbf{X}_i^{\nu})$, each agent i updates $\mathbf{D}_{(i)}$ setting $\mathbf{X}_i=\mathbf{X}_i^{\nu}$ and solving

$$\widetilde{\mathbf{D}}_{(i)}^{\nu} \triangleq \operatorname*{argmin}_{\mathbf{D}_{(i)}\in\mathcal{D}} \left\{ f_i\left(\mathbf{D}_{(i)},\mathbf{X}_i^{\nu}\right)+\frac{\tau_{D,i}^{\nu}}{2}\|\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\|_F^2+\left\langle\widetilde{\mathbf{\Pi}}_i^{\nu},\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\right\rangle\right\}$$

$$\mathbf{U}_{(i)}^{\nu}=\mathbf{D}_{(i)}^{\nu}+\gamma^{\nu}(\widetilde{\mathbf{D}}_{(i)}^{\nu}-\mathbf{D}_{(i)}^{\nu})$$

where $\tau_{D,i}^{\nu}>0$ and $\widetilde{\mathbf{\Pi}}_i^{\nu}$ aims to $\widetilde{\mathbf{\Pi}}_i^{\nu}\to\sum_{j\neq i}\nabla f_j(\mathbf{D}_{(i)}^{\nu},\mathbf{X}_j^{\nu})$.

[optimization of $\mathbf{X}_i$]: Given $(\mathbf{U}_{(i)}^{\nu},\mathbf{X}_i^{\nu})$, each agent i updates $\mathbf{X}_i$ setting $\mathbf{D}_i=\mathbf{U}_{(i)}^{\nu}$ and solving

$$\mathbf{X}_i^{\nu+1}\triangleq\operatorname*{argmin}_{\mathbf{X}_i}\left\{f_i(\mathbf{U}_{(i)}^{\nu},\mathbf{X}_i)+\frac{\tau_{X,i}^{\nu}}{2}\|\mathbf{X}_i-\mathbf{X}_i^{\nu}\|_F^2+g_i(\mathbf{X}_i)\right\},$$

with $\tau_{X,i}^{\nu}>0$.

Algorithmic Design

Step 2: Broadcasting

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\Bigg\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\Bigg\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i)+\sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j)+\sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

Each agent i: maintains a *local copy* $\mathbf{D}_{(i)}$ of $\mathbf{D}$, and controls $\mathbf{X}_i$:

1. [local optimization]: optimizes $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ alternatingly by solving strongly convex problems
2. [consensus update]: Each agent i collects $\mathbf{U}_{(j)}$ from its neighbors and updates:

$$\mathbf{D}_{(i)}^{\nu+1} = \sum_{j\in\mathcal{N}_i[\nu]} w_{ij}^{\nu}\mathbf{U}_{(j)}^{\nu}$$

Question: How to distibutively determine the weights $(w_{ij}^{\nu})_{i,j}$ matching an *arbitrary (time-varying)* digraph that will guarantee eventual consensus?

Algorithmic Design

Step 2: Broadcasting

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I} \Big\{ \underbrace{\frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)} + g_i(\mathbf{X}_i) \Big\}, \quad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i) + \sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j) + \sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

Each agent i: maintains a *local copy* $\mathbf{D}_{(i)}$ of $\mathbf{D}$, and controls $\mathbf{X}_i$:

1. [local optimization]: optimizes $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ alternatingly by solving strongly convex problems
2. [consensus update]: Each agent i collects $\mathbf{U}_{(j)}$ from its neighbors and updates:

$$\mathbf{D}_{(i)}^{\nu+1} = \sum_{j\in\mathcal{N}_i[\nu]} w_{ij}^{\nu}\mathbf{U}_{(j)}^{\nu}$$

Question: How to distibutively determine the weights $(w_{ij}^{\nu})_{i,j}$ matching an *arbitrary (time-varying)* digraph that will guarantee eventual consensus?

Algorithmic Design

Step 2: Broadcasting

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|_F^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\right\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i) + \sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j) + \sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

Each agent i: maintains a *local copy* $\mathbf{D}_{(i)}$ of $\mathbf{D}$, and controls $\mathbf{X}_i$:

1. [local optimization]: optimizes $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ alternatingly by solving strongly convex problems
2. [consensus update]: Each agent i collects $\mathbf{U}_{(j)}$ from its neighbors and updates:

$$\mathbf{D}_{(i)}^{\nu+1} = \sum_{j\in\mathcal{N}_i[\nu]} w_{ij}^{\nu}\mathbf{U}_{(j)}^{\nu}$$

Question: How to distibutively determine the weights $(w_{ij}^{\nu})_{i,j}$ matching an *arbitrary (time-varying)* digraph that will guarantee eventual consensus?

Algorithmic Design

Consensus Weights $\mathbf{W}^\nu \triangleq \left(w_{ij}^\nu\right)_{i,j=1}^I$

- Doubly-stochasticity ($\mathbf{W}^\nu \mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T \mathbf{W}^\nu = \mathbf{1}^T$) on digraphs [Cat-Say'10]
  - not all digraphs admit a doubly-stochastic matrix
  - when exists, constructing one calls for additional (de-)centralized algorithms
- Our approach: Introducing a new consensus protocol requiring only column stochasticity

Algorithmic Design

Consensus Weights $\mathbf{W}^\nu \triangleq \left(w_{ij}^\nu\right)_{i,j=1}^I$

- Doubly-stochasticity ($\mathbf{W}^\nu \mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T \mathbf{W}^\nu = \mathbf{1}^T$) on digraphs [Cat-Say'10]
  - not all digraphs admit a doubly-stochastic matrix
  - when exists, constructing one calls for additional (de-)centralized algorithms
- Our approach: Introducing a new consensus protocol requiring only column stochasticity

column stochastic matrix
(equally mixing)

$$\widetilde{\mathbf{W}}^\nu \triangleq \left(\tilde{w}_{ij}^\nu\right)_{i,j=1}^I$$

Special case for $\widetilde{\mathbf{W}}$: push-sum weights

Algorithmic Design

Consensus Weights $\mathbf{W}^{\nu} \triangleq \left(w_{ij}^{\nu}\right)_{i,j=1}^{I}$

- Doubly-stochasticity ($\mathbf{W}^{\nu}\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^{T}\mathbf{W}^{\nu} = \mathbf{1}^{T}$) on digraphs [Cat-Say'10]
  - not all digraphs admit a doubly-stochastic matrix
  - when exists, constructing one calls for additional (de-)centralized algorithms
- Our approach: Introducing a new consensus protocol requiring only column stochasticity

extra variables $\{\phi_i^{\nu}\}_{i=1}^{I}$

column stochastic matrix
(equally mixing)

$\widetilde{\mathbf{W}}^{\nu} \triangleq \left(\tilde{w}_{ij}^{\nu}\right)_{i,j=1}^{I}$

Special case for $\widetilde{\mathbf{W}}$: push-sum weights

Algorithmic Design

Consensus Weights $\mathbf{W}^\nu \triangleq (w_{ij}^\nu)_{i,j=1}^I$

- Doubly-stochasticity ($\mathbf{W}^\nu \mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T \mathbf{W}^\nu = \mathbf{1}^T$) on digraphs [Cat-Say'10]
  - not all digraphs admit a doubly-stochastic matrix
  - when exists, constructing one calls for additional (de-)centralized algorithms
- Our approach: Introducing a new consensus protocol requiring only column stochasticity

extra variables $\{\phi_i^\nu\}_{i=1}^I$

column stochastic matrix
(equally mixing)

$$\widetilde{\mathbf{W}}^\nu \triangleq (\tilde{w}_{ij}^\nu)_{i,j=1}^I$$

reconstruct row stochastic matrix
(consensus + feasibility)

$$\phi_i^{\nu+1} = \sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \, \phi_j^\nu$$

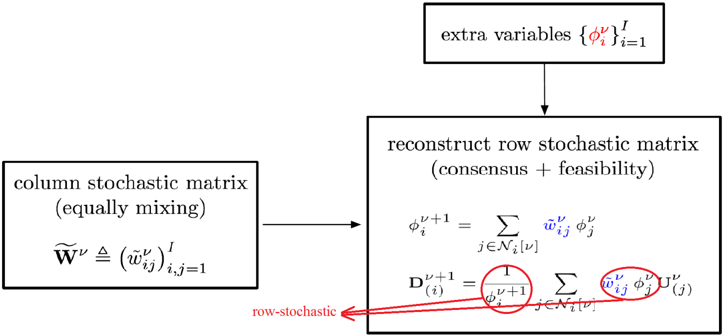
$$\mathbf{D}_{(i)}^{\nu+1} = \frac{1}{\phi_i^{\nu+1}} \sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \, \phi_j^\nu \mathbf{U}_{(j)}^\nu$$

Special case for $\widetilde{\mathbf{W}}$: push-sum weights

Algorithmic Design

Consensus Weights $\mathbf{W}^\nu \triangleq \left(w_{ij}^\nu\right)_{i,j=1}^I$

- Doubly-stochasticity ($\mathbf{W}^\nu \mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T \mathbf{W}^\nu = \mathbf{1}^T$) on digraphs [Cat-Say'10]
  - not all digraphs admit a doubly-stochastic matrix
  - when exists, constructing one calls for additional (de-)centralized algorithms
- Our approach: Introducing a new consensus protocol requiring only column stochasticity



Special case for $\widetilde{\mathbf{W}}$: push-sum weights

Algorithmic Design

Step 2: Broadcasting

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I} \Bigg\{ \underbrace{\frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|^2}_{f_i(\mathbf{D},\mathbf{X}_i)} + g_i(\mathbf{X}_i) \Bigg\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i) + \sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j) + \sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

Each agent i: maintains a *local copy* $\mathbf{D}_{(i)}$ of $\mathbf{D}$, and controls $\mathbf{X}_i$:

1. [local optimization]: optimizes $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ alternatingly by solving strongly convex problems
2. [consensus update]: collects $\mathbf{U}_{(j)}$ from its neighbors and updates:

$$\phi_i^{\nu+1} = \sum_{j\in\mathcal{N}_i[\nu]} \tilde{w}_{ij}^{\nu}\, \phi_j^{\nu}$$

$$\mathbf{D}_{(i)}^{\nu+1} = \frac{1}{\phi_i^{\nu+1}} \sum_{j\in\mathcal{N}_i[\nu]} \tilde{w}_{ij}^{\nu}\, \phi_j^{\nu}\, \mathbf{U}_{(j)}^{\nu}$$

Algorithmic Design

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|^2}_{f_i(\mathbf{D},\mathbf{X}_i)} + g_i(\mathbf{X}_i)\right\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i) + \sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j) + \sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

[optimization of $\mathbf{D}_{(i)}$]: Each agent i updates $\mathbf{D}_{(i)}$ setting $\mathbf{X}_i = \mathbf{X}_i^\nu$ and solving

$$\widetilde{\mathbf{D}}_{(i)}^\nu \triangleq \operatorname*{argmin}_{\mathbf{D}_{(i)}\in\mathcal{D}} \left\{ f_i\left(\mathbf{D}_{(i)},\mathbf{X}_i^\nu\right) + \frac{\tau_{D,i}^\nu}{2}\|\mathbf{D}_{(i)} - \mathbf{D}_{(i)}^\nu\|_F^2 + \left\langle \widetilde{\mathbf{\Pi}}_i^\nu, \mathbf{D}_{(i)} - \mathbf{D}_{(i)}^\nu \right\rangle \right\}$$

$$\mathbf{U}_{(i)}^\nu = \mathbf{D}_{(i)}^\nu + \gamma^\nu(\widetilde{\mathbf{D}}_{(i)}^\nu - \mathbf{D}_{(i)}^\nu)$$

where $\tau_{D,i}^\nu > 0$ and $\widetilde{\mathbf{\Pi}}_i^\nu$ aims to

$$\widetilde{\mathbf{\Pi}}_i^\nu \longrightarrow \sum_{j\neq i} \nabla f_j(\mathbf{D}_{(i)}^\nu, \mathbf{X}_j^\nu)$$

Question: How to choose $\widetilde{\mathbf{\Pi}}_i^\nu$ to convergence while using ONLY *local* information?

Algorithmic Design

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} U(\mathbf{D},\mathbf{X}) \triangleq \sum_{i=1}^{I}\left\{\underbrace{\frac{1}{2}\|\mathbf{Y}_i-\mathbf{D}\mathbf{X}_i\|^2}_{f_i(\mathbf{D},\mathbf{X}_i)}+g_i(\mathbf{X}_i)\right\}, \qquad U(\mathbf{D},\mathbf{X}) = f_i(\mathbf{D},\mathbf{X}_i)+\sum_{j\neq i} f_j(\mathbf{D},\mathbf{X}_j)+\sum_{i=1}^{I} g_i(\mathbf{X}_i)$$

[optimization of $\mathbf{D}_{(i)}$]: Each agent i updates $\mathbf{D}_{(i)}$ setting $\mathbf{X}_i=\mathbf{X}_i^\nu$ and solving

$$\widetilde{\mathbf{D}}_{(i)}^{\nu} \triangleq \underset{\mathbf{D}_{(i)}\in\mathcal{D}}{\text{argmin}}\left\{f_i\left(\mathbf{D}_{(i)},\mathbf{X}_i^\nu\right)+\frac{\tau_{D,i}^{\nu}}{2}\|\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\|_F^2+\left\langle\widetilde{\mathbf{\Pi}}_i^{\nu},\mathbf{D}_{(i)}-\mathbf{D}_{(i)}^{\nu}\right\rangle\right\}$$

$$\mathbf{U}_{(i)}^{\nu} = \mathbf{D}_{(i)}^{\nu}+\gamma^{\nu}(\widetilde{\mathbf{D}}_{(i)}^{\nu}-\mathbf{D}_{(i)}^{\nu})$$

where $\tau_{D,i}^{\nu}>0$ and $\widetilde{\mathbf{\Pi}}_i^{\nu}$ aims to

$$\widetilde{\mathbf{\Pi}}_i^{\nu} \longrightarrow \cancel{\sum_{j\neq i}\nabla f_j(\mathbf{D}_{(i)}^{\nu},\mathbf{X}_j^{\nu})} \longleftarrow \sum_{j\in\mathcal{N}_i[\nu]}\nabla f_j(\mathbf{D}_{(j)}^{\nu},\mathbf{X}_j^{\nu})$$

Question: How to choose $\widetilde{\mathbf{\Pi}}_i^{\nu}$ to convergence while using ONLY *local* information?

Algorithmic Design

Local update of $\widetilde{\mathbf{\Pi}}_i^\nu$

$$\widetilde{\mathbf{\Pi}}_i^\nu \longrightarrow \sum_{j \neq i} \nabla f_j(\mathbf{D}_{(i)}^\nu, \mathbf{X}_j^\nu)$$

- Distributed Tracking of Gradient Averages (similar to [DiLor-Scu'15]):

$$\begin{aligned}
\widetilde{\mathbf{\Theta}}_i^{\nu+1} &= \frac{1}{\phi_i^{\nu+1}} \left(\sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \phi_j^\nu \widetilde{\mathbf{\Theta}}_j^\nu + \left(\nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1}) - \nabla_D f_i(\mathbf{D}_{(i)}^\nu, \mathbf{X}_i^\nu) \right) \right) \\
\widetilde{\mathbf{\Pi}}_i^{\nu+1} &= I \cdot \widetilde{\mathbf{\Theta}}_i^{\nu+1} - \nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1})
\end{aligned}$$

with $\mathbf{\Theta}_i^0 = \nabla f_i(\mathbf{D}_{(i)}^0, \mathbf{X}_i^0)$.

Algorithmic Design

Local update of $\widetilde{\mathbf{\Pi}}_i^\nu$

$$\widetilde{\mathbf{\Pi}}_i^\nu \longrightarrow \sum_{j \neq i} \nabla f_j(\mathbf{D}_{(i)}^\nu, \mathbf{X}_j^\nu)$$

- Distributed Tracking of Gradient Averages (similar to [DiLor-Scu'15]):

$$\widetilde{\mathbf{\Theta}}_i^{\nu+1} = \frac{1}{\phi_i^{\nu+1}} \left(\sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \phi_j^\nu \widetilde{\mathbf{\Theta}}_j^\nu + \left(\nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1}) - \nabla_D f_i(\mathbf{D}_{(i)}^\nu, \mathbf{X}_i^\nu) \right) \right)$$

$$\widetilde{\mathbf{\Pi}}_i^{\nu+1} = I \cdot \widetilde{\mathbf{\Theta}}_i^{\nu+1} - \nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1})$$

with $\mathbf{\Theta}_i^0 = \nabla f_i(\mathbf{D}_{(i)}^0, \mathbf{X}_i^0)$.

D^4L alg.: Decentralized Dictionary Learning over Dynamic Digraphs

Data: $\{\gamma^\nu\}_\nu > 0$, $\phi_i^0 = 1$, $\mathbf{D}_{(i)}^0 \in \mathcal{D}$, $\mathbf{X}_i^0 = \mathbf{0}$, $\widetilde{\mathbf{\Theta}}_i^0 = \nabla_D f_i(\mathbf{D}_{(i)}^0, \mathbf{X}_i^0)$ for all i's; Set $\nu = 0$;

(S.1): If $(\mathbf{D}_{(i)}^\nu, \mathbf{X}_i^\nu)$ satisfies a suitable termination criterion, STOP;

(S.2): [Optimization step]: Each agent i updates $\mathbf{D}_{(i)}$ and $\mathbf{X}_i$ locally:

$$
\begin{aligned}
\widetilde{\mathbf{D}}_{(i)}^\nu &= \underset{\mathbf{D}_{(i)} \in \mathcal{D}}{\text{argmin}}\; f_i\left(\mathbf{D}_{(i)}, \mathbf{X}_i^\nu\right) + \left\langle \widetilde{\mathbf{\Pi}}_i^\nu, \mathbf{D}_{(i)} - \mathbf{D}_{(i)}^\nu \right\rangle + \frac{\tau_{D,i}^\nu}{2} \|\mathbf{D}_{(i)} - \mathbf{D}_{(i)}^\nu\|_F^2 \\
\mathbf{U}_{(i)}^\nu &= \mathbf{D}_{(i)}^\nu + \gamma^\nu (\widetilde{\mathbf{D}}_{(i)}^\nu - \mathbf{D}_{(i)}^\nu) \\
\mathbf{X}_i^{\nu+1} &= \underset{\mathbf{X}_i}{\text{argmin}}\; f_i(\mathbf{U}_{(i)}^\nu, \mathbf{X}_i) + g_i(\mathbf{X}_i) + \frac{\tau_{X,i}^\nu}{2} \|\mathbf{X}_i - \mathbf{X}_i^\nu\|_F^2
\end{aligned}
$$

(S.3): [Consensus step]: Each agent i collects from its neighbors and updates

$$
\begin{aligned}
\phi_i^{\nu+1} &= \sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \, \phi_j^\nu \\
\mathbf{D}_{(i)}^{\nu+1} &= \frac{1}{\phi_i^{\nu+1}} \sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \, \phi_j^\nu \mathbf{U}_{(j)}^\nu \\
\widetilde{\mathbf{\Theta}}_i^{\nu+1} &= \frac{1}{\phi_i^{\nu+1}} \left(\sum_{j \in \mathcal{N}_i[\nu]} \tilde{w}_{ij}^\nu \phi_j^\nu \widetilde{\mathbf{\Theta}}_j^\nu + \left(\nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1}) - \nabla_D f_i(\mathbf{D}_{(i)}^\nu, \mathbf{X}_i^\nu) \right) \right) \\
\widetilde{\mathbf{\Pi}}_i^{\nu+1} &= I \cdot \widetilde{\mathbf{\Theta}}_i^{\nu+1} - \nabla_D f_i(\mathbf{D}_{(i)}^{\nu+1}, \mathbf{X}_i^{\nu+1})
\end{aligned}
$$

(S.4): $\nu \leftarrow \nu + 1$ and go to (S.1).

Theorem: D^{4}L Convergence

Given the optimization problem (P) in the setting above, suppose that

- [Mixing Weights]: The weights $\widetilde{\mathbf{W}}^\nu \triangleq (\tilde{w}_{ij}^\nu)_{i,j=1}^I$ are chosen so that, for all ν, it holds
$$\tilde{w}_{ij}^\nu = \begin{cases} > \theta \in (0,1] & \text{if } j \in \mathcal{N}_i[\nu]; \\ = 0 & \text{otherwise.} \end{cases}, \quad \mathbf{1}^T \widetilde{\mathbf{W}}^\nu = \mathbf{1}^T$$
- [Step-size]: The step-size $\gamma^\nu \in [0,1]$ is chosen so that $\sum_\nu \gamma^\nu = +\infty$ and $\sum_\nu (\gamma^\nu)^2 < +\infty$.
- [Proximal weights]: The sequences $\{\tau_{D,i}^\nu\}$ and $\{\tau_{X,i}^\nu\}$ satisfy:
$$\tau_{X,i}^\nu = \max\left(\sigma_{\max}(\mathbf{U}_{(i)}^\nu)^2, \epsilon_1\right), \quad \tau_{D,i}^\nu = \epsilon_2,$$
with $\epsilon_1,\ \epsilon_2 > 0$.

Then, we have:

(a) *[convergence]*: $\{(\overline{\mathbf{D}}^\nu, \mathbf{X}^\nu)\}_\nu$ is bounded (where $\overline{\mathbf{D}}^\nu \triangleq \frac{1}{I}\sum_{i=1}^I \phi_i^\nu \mathbf{D}_{(i)}^\nu$) and all of its limit points are stationary solutions of Problem (P1);

(b) *[consensus]*: All $\{\mathbf{D}_{(i)}^\nu\}_\nu$ asymptotically reach consensus, i.e., $||\mathbf{D}_{(i)}^\nu - \overline{\mathbf{D}}^\nu|| \underset{n\to\infty}{\longrightarrow} 0$, for all $i = 1, 2, \ldots, I$

Numerical Results

1. Image restoration (denoising)
2. Biclustering of gene expressions

Distributed Image Restoration (denoising)

Setup:

- Corrupted 512×512 image
- 255,000 patches of size 8×8
- Network of 150 agents
- Dictionary $\mathbf{D}$ of size 64×64
- Total of ($\approx$) 16.4 million variables

Figure: original and noisy images

Distributed Image Restoration (denoising)

$$\min_{\substack{\mathbf{D}\in\mathcal{D}\\ \{\mathbf{X}_i\}_i}} \sum_{i=1}^{I} \left\{ \underbrace{\frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|^2}_{f_i(\mathbf{D},\mathbf{X}_i)} + \lambda\|\mathbf{X}_i\|_1 + \frac{\mu}{2}\|\mathbf{X}_i\|_F^2 \right\}$$

$\mathbf{Y}_i$ = patches of noisy image
$\mathbf{D}$ = dictionary
$\mathbf{X}_i$ = sparse representation

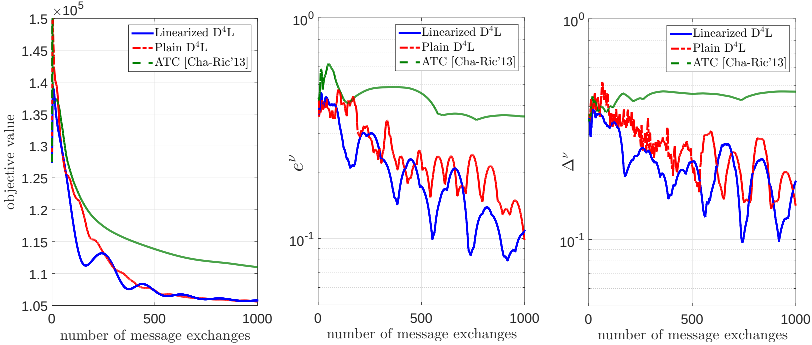
Two instances of our algorithm:

- Plain D^{4}L: using original function f_i in the convex subproblems;
  - $\mathbf{D}_{(i)}^{\nu}$ has closed form solution
  - $\mathbf{X}_i^{\nu+1}$ is solution of a LASSO
- Linearized D^{4}L: using first order approximation of function f_i in the convex subproblems
  - $\mathbf{D}_{(i)}^{\nu}$ has closed form solution
  - $\mathbf{X}_i^{\nu+1}$ has closed form solution

Distributed Image Restoration (denoising)

Merit functions

- Optimality merit: Δ^{ν} = distance from stationarity of $(\overline{\mathbf{D}}^{\nu}, \mathbf{X}^{\nu})$ [Fac-Scu-Sag'15]
- Consensus merit: e^{ν} = consensus disagreement



Distributed Image Restoration (denoising)

Quality of the reconstruction (~ 200 message exchanges)

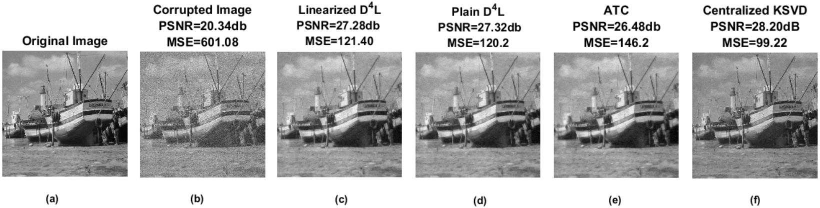


Figure: Comparison after 200 message exchanges

	Linearized D^4L	Plain D^4L	ATC
200 message exchanges	PSNR=27.28db MSE=121.4	PSNR=27.32db MSE=120.2	PSNR=26.48db MSE=146.2
1000 message exchanges	PSNR=27.53db MSE=114.6	PSNR=27.65db MSE=111.69	PSNR=27.29db MSE=121.23

Figure: Comparison of reconstructed images after 200 and 1000 message exchanges

Distributed Image Restoration (denoising)

Computational time per iteration

	Linearized D^{4}L	Plain D^{4}L	ATC
Averaged Comp. Time (sec)	2.862	11.328	9.838

Table: Computation time per message passing

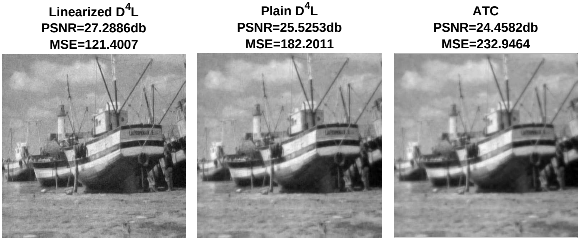


Figure: Reconstructed images after ∼300 seconds

Conclusions

- We proposed a novel *decentralized* algorithmic framework for a fairly general class of Dictionary Learning problems
  - parallel and distributed updates
  - arbitrary digraphs
  - shared $\mathbf{D}$ and private variables $\{\mathbf{X}_i\}_i$
- Preliminary numerical results show promising performance
- The framework is applicable to a variety of other learning problems (with general biconvex function)
  - supervised/discriminative learning
  - low-rank plus sparse decomposition
  - sparse SVD
  - ...