An Augmented Lagrangian Primal-Dual Semismooth Newton method for multi-block composite optimization

Zaiwen Wen

Beijing International Center For Mathematical Research
Peking University

Joint work with Zhanwang Deng, Kangkang Deng, Jiang Hu

Introduction

Linearly constrained multi-block convex composite optimization:

$$\min_{\mathbf{x}_1,\cdots,\mathbf{x}_n} \sum_{i\in\mathcal{I}_1} h_i(\mathbf{x}_i) + \sum_{i\in\mathcal{I}_2} h_i(\mathbf{x}_i), \text{ s.t. } \sum_{i=1}^{n-1} \mathcal{A}_i \mathbf{x}_i + \mathbf{x}_n = \mathbf{b}, \qquad \text{(P)}$$

- $h_i : \mathcal{X}_i \to (-\infty, +\infty)$ is a convex and twice differentiable function with Lipschitz continuous gradient for any $i \in \mathcal{I}_1$.
- $h_i : \mathcal{X}_i \to (-\infty, +\infty]$ is a proper closed convex function and may not be differentiable for any $i \in \mathcal{I}_2$.
- $\mathcal{A}_i : \mathcal{X}_i \to \mathcal{Y}$ is a linear map for any $i = 1, \cdots, n-1$, $\mathcal{A}_n = \mathcal{I}$ is the identity mapping, $\mathbf{b} \in \mathcal{Y}$.
- Covers a wide range of nonsmooth problems and applications.

Goal: a unified second-order algorithmic framework

- General conic programming: **LP, SOCP, SDP**, etc

$$\min_{\boldsymbol{x}} \langle \boldsymbol{c}, \boldsymbol{x} \rangle, \mathcal{A}(\boldsymbol{x}) = \boldsymbol{b}, \boldsymbol{x} \in K, \boldsymbol{x} \in \mathcal{P}.$$

  The dual is:

$$\begin{aligned} \min_{\boldsymbol{y},\boldsymbol{s},\boldsymbol{z}} \quad & -\langle \boldsymbol{b}, \boldsymbol{y} \rangle + \delta_{\mathcal{P}}^*(-\boldsymbol{z}) \\ \text{s.t.} \quad & \mathcal{A}^*(\boldsymbol{y}) + \boldsymbol{s} + \boldsymbol{z} = \boldsymbol{c}, \boldsymbol{s} \in K^*. \end{aligned}$$

- Image processing including **denoising and deblurring**

$$\min_{u \in \mathbb{R}^n} \quad \frac{1}{2}\|Au - b\|^2 + \delta_{\mathcal{Q}}(u) + \lambda\|Du\|_1,$$

  The dual is

$$\begin{aligned} \min_{y,\mu,\nu} \quad & \frac{1}{2}\|y\|^2 + \langle y, b \rangle + \delta_{\mathcal{Q}}^*(s) + \delta_{\|\cdot\|_\infty < \lambda}(\mu), \\ \text{s.t.} \quad & A^\top y + D^\top s + \mu = 0. \end{aligned}$$

Goal: a unified second-order algorithmic framework

- The corrected **tensor nuclear norm** (CTNN) model:

$$\min_{\mathcal{X}} \frac{1}{2m}\|\boldsymbol{z}\|^2 + \mu\left(\|\mathcal{X}\|_{TNN} - \langle F(\mathcal{X}_m), \mathcal{X}\rangle\right) + \delta_{\mathfrak{U}}(\mathcal{X})$$
$$\text{s.t. } \boldsymbol{z} = \boldsymbol{y} - \mathcal{D}_\Omega(\mathcal{X}).$$

The dual is:

$$\min_{\boldsymbol{v},\mathcal{W},\mathcal{Z}} \frac{m}{2}\|\boldsymbol{v}\|^2 - \langle \boldsymbol{v}, \boldsymbol{y}\rangle + \delta^*_{\mathfrak{U}}(-\mathcal{W}) + \delta_{\mathfrak{X}}(\mathcal{Z})$$
$$\text{s.t. } \mathcal{D}^*_\Omega(\boldsymbol{v}) - \mathcal{Z} + \mathcal{W} = -\mu F(\mathcal{X}_m),$$

- Unified model: **three block composite problem**:

$$\min_{x\in\mathbb{R}^n} f(\mathcal{A}x) + g(\mathcal{B}x) + h(x),$$

where the dual is:

$$\min_{\lambda,\mu,w} f^*(\lambda) + g^*(\mu) + h^*(w), \text{ s.t. } \mathcal{A}^*\lambda + \mathcal{B}^*\mu + w = 0.$$

Why semismooth Newton (SSN) methods?

Consider

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}) + h(\boldsymbol{x})$$

- First-order methods: proximal gradient, ADMM, primal-dual, etc
  - **Advantages:** easy to implement; converge fast to a solution with moderate accuracy.
  - **Disadvantages:** slow tail convergence, may fail on slightly more **challenging problems.**
- Second-order methods: interior point or Newton methods:
  - **Advantages:** converge fast to a solution with high accuracy
  - **Disadvantages:** not easy to implement; usually smooth problem
- Second-order methods: semismooth Newton method
  - **Advantages:** nonsmooth problem; converge fast to a solution with high accuracy
  - **Disadvantages:** not easy to implement; most state-of-the-art SSN methods essentially only cover two blocks

SSN based on ALM

- The AL function: $L_t(\boldsymbol{x}, \boldsymbol{z}) = f(\boldsymbol{x}) + e_{th}(\boldsymbol{x} - t\boldsymbol{z}) - \frac{t}{2}\|\boldsymbol{z}\|^2$
- Moreau envelope: $e_{th}(\boldsymbol{x}) := \min_{\boldsymbol{y}\in\mathbb{R}^n} \; h(\boldsymbol{y}) + \frac{1}{2t}\|\boldsymbol{y} - \boldsymbol{x}\|^2$
  Proximal operator: $\mathbf{prox}_{th}(\boldsymbol{x}) := \arg\min_{\boldsymbol{y}\in\mathbb{R}^n} \; h(\boldsymbol{y}) + \frac{1}{2t}\|\boldsymbol{y} - \boldsymbol{x}\|^2$
- The ALM framework is:

$$\begin{cases} \boldsymbol{x}_{k+1} = \underset{\boldsymbol{x}\in\mathbb{R}^n}{\arg\min} \;\; \varphi(\boldsymbol{x}; t_k, \boldsymbol{z}_k) := L_{t_k}(\boldsymbol{x}, \boldsymbol{z}_k), \\ \boldsymbol{z}_{k+1} = \boldsymbol{z}_k + (\mathbf{prox}_{t_k h}(\boldsymbol{x}_{k+1} - t_k \boldsymbol{z}_k) - \boldsymbol{x}_{k+1})/t_k. \end{cases}$$

- SSN is applied to minimize each AL function:

$$F_{\mathrm{ALM}}(x; z_k) := \nabla f(x) + \frac{1}{t_k}(x - tz_k - \mathbf{prox}_{th}(x - t_k z_k)) = 0.$$

- Typical ALM based methods: SDPNAL, SDPNAL+, etc.
- ALM: Dual ascent gradient method in terms of the multipliers.

SSN based on residual mappings

- Natural residual from the proximal gradient method:

$$T_{\mathrm{PGM}}(\boldsymbol{u}) = \mathbf{prox}_{th}(\boldsymbol{u} - t\nabla f(\boldsymbol{u}))$$

- The Douglas-Rachford splitting (DRS) residual:

$$T_{\mathrm{DRS}}(\boldsymbol{u}) = \boldsymbol{u} + \mathbf{prox}_{tf}(2\mathbf{prox}_{th}(\boldsymbol{u}) - \boldsymbol{u}) - \mathbf{prox}_{th}(\boldsymbol{u})$$

- The original problem is equivalent to the system

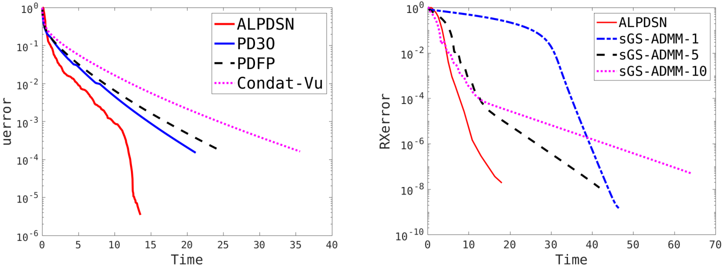
$$\boxed{F(\boldsymbol{u}) := (I - T)(\boldsymbol{u}) = 0.}$$

- Apply the SSN method to $F(\boldsymbol{u}) = 0$.
- The construction of $F(\boldsymbol{u}) = 0$ is case by case, not general.
- Only cover two block problems in most cases.
- Globalization of the SSN steps usually depends on the uderlying first-order method

Contributions

A versatile SSN method for **multi-block convex composite problems.**

- **An easy-to-follow paradigm of semismooth system.**
- **A tractable pure semismooth Newton strategy.**
- **A rigorous global and local convergence analysis.**
- **A promising performance on complicated applications.**



(a) CT image restoration (b) CTNN

Outline

1. A nonlinear system from the AL saddle point
2. A semismooth Newton method
3. Convergence analysis
   - Global convergence
   - Local convergence
4. Applications

Step 1: a routine for writing the AL function

Recall our problem:

$$\min_{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n} \sum_{i\in\mathcal{I}_1} h_i(\boldsymbol{x}_i) + \sum_{i\in\mathcal{I}_2} h_i(\boldsymbol{x}_i), \text{ s.t. } \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i + \boldsymbol{x}_n = \boldsymbol{b}, \tag{P}$$

The AL function:

$$\begin{aligned}\Phi(\boldsymbol{x}, \boldsymbol{z}) := \sum_{i\in\mathcal{I}_1} h_i(\boldsymbol{x}_i) + \sum_{i\in\mathcal{I}_2\backslash n} e_\sigma h_i(\boldsymbol{x}_i - \frac{\boldsymbol{z}_i}{\sigma}) \\ + e_\sigma h_n\left(\boldsymbol{b} - \frac{\boldsymbol{z}_n}{\sigma} - \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i\right) - \frac{1}{2\sigma} \sum_{i\in\mathcal{I}_2} \|\boldsymbol{z}_i\|^2,\end{aligned}$$

where $e_\sigma h_i(\boldsymbol{x}) := \min_{\boldsymbol{y}} h_i(\boldsymbol{y}) + \frac{\sigma}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2$ is the Moreau envelope.

Derivation of the AL function

- Introduce the auxiliary variables, original problem is recast as

$$\min \sum_{i\in\mathcal{I}_1} h_i(\boldsymbol{x}_i) + \sum_{i\in\mathcal{I}_2} h_i(\boldsymbol{y}_i) + h_n(\boldsymbol{x}_n),$$

$$\text{s.t.} \sum_{i=1}^{n-1} \mathcal{A}_i\boldsymbol{x}_i + \boldsymbol{x}_n = \boldsymbol{b},\ \ \boldsymbol{x}_i = \boldsymbol{y}_i, i \in \mathcal{I}_2\backslash n.$$

- The traditional AL function is given as follows:

$$\mathcal{L}_\sigma(\boldsymbol{x}, \boldsymbol{x}_n, \boldsymbol{y}; \boldsymbol{z}) := \sum_{i\in\mathcal{I}_1} h_i(\boldsymbol{x}_i) + \sum_{i\in\mathcal{I}_2\backslash n} h_i(\boldsymbol{y}_i) + h_n(\boldsymbol{x}_n) - \frac{1}{2\sigma}\sum_{i\in\mathcal{I}_2} \|\boldsymbol{z}_i\|^2$$

$$+ \frac{\sigma}{2}\|\sum_{i=1}^{n-1} \mathcal{A}_i\boldsymbol{x}_i + \boldsymbol{x}_n - \boldsymbol{b} + \boldsymbol{z}_n/\sigma\|^2 + \frac{\sigma}{2}\sum_{i\in\mathcal{I}_2\backslash n} \|\boldsymbol{x}_i - \boldsymbol{y}_i - \boldsymbol{z}_i/\sigma\|^2.$$

- Then Φ can be obtained by $\boxed{\Phi(\boldsymbol{x}, \boldsymbol{z}) := \min_{\boldsymbol{y},\boldsymbol{x}_n} \mathcal{L}_\sigma(\boldsymbol{x}, \boldsymbol{x}_n, \boldsymbol{y}; \boldsymbol{z})}$.

Explicit expression of the AL function

Proximal operator: $\textbf{prox}_{th}(\boldsymbol{x}) := \arg\min_{\boldsymbol{y}\in\mathbb{R}^n} \; h(\boldsymbol{y}) + \frac{1}{2t}\|\boldsymbol{y}-\boldsymbol{x}\|^2$.

$$\Phi(\boldsymbol{x};\boldsymbol{z}) = h_n(\textbf{prox}_{h_n/\sigma}(\boldsymbol{b} - \boldsymbol{z}_n/\sigma - \sum_{i=1}^{n-1}\mathcal{A}_i\boldsymbol{x}_i)) - \underbrace{\frac{1}{2\sigma}\sum_{i\in\mathcal{I}_2}\|\boldsymbol{z}_i\|^2}_{\text{multipliers}}$$

$$+\sum_{i\in\mathcal{I}_2\backslash n}\underbrace{\left(h_i(\textbf{prox}_{h_i/\sigma}(\boldsymbol{x}_i - \boldsymbol{z}_i/\sigma)) + \frac{\sigma}{2}\|\boldsymbol{x}_i - \boldsymbol{z}_i/\sigma - \textbf{prox}_{h_i/\sigma}(\boldsymbol{x}_i - \boldsymbol{z}_i/\sigma)\|^2\right)}_{\text{Moreau envelope of } h_i}$$

$$+\sum_{i\in\mathcal{I}_1} h(\boldsymbol{x}_i) + \underbrace{\frac{\sigma}{2}\left\|\sum_{i=1}^{n-1}\mathcal{A}_i\boldsymbol{x}_i + \textbf{prox}_{h_n/\sigma}(\boldsymbol{b} - \boldsymbol{z}_n/\sigma - \sum_{i=1}^{n-1}\mathcal{A}_i\boldsymbol{x}_i) - \boldsymbol{b} + \boldsymbol{z}_n/\sigma\right\|^2}_{\sum_{i=1}^{n-1}\mathcal{A}_i\boldsymbol{x}_i + \boldsymbol{x}_n = \boldsymbol{b}}$$

Theoretical foundation: Augmented Lagrangian duality

Assumption

Problem (P) has an optimal solution $\boldsymbol{x}^*$*. Furthermore, the Slater's condition holds, i.e., there exists* $\boldsymbol{x}_i \in \mathrm{ri}(\mathrm{dom}(h_i))(i = 1, \cdots, n)$ *such that* $\sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i + \boldsymbol{x}_n = \boldsymbol{b}$*.*

Lemma (AL duality)

Suppose the assumption holds, given $\sigma > 0$*, the strong duality holds for* $\Phi(\boldsymbol{x}, \boldsymbol{z})$*, namely,*

$$\min_{\boldsymbol{x}} \max_{\boldsymbol{z}} \Phi(\boldsymbol{x}, \boldsymbol{z}) = \max_{\boldsymbol{z}} \min_{\boldsymbol{x}} \Phi(\boldsymbol{x}, \boldsymbol{z}),$$

where both sides are equivalent to (P).

AL duality is different from the Lagrangian duality.

Step 2: construction of the semismooth system

- Problem (P) is equivalent to solving a nonlinear system:

$$F(\boldsymbol{w}) = \begin{pmatrix} \nabla_{\boldsymbol{x}} \Phi(\boldsymbol{w}) \\ -\nabla_{\boldsymbol{z}} \Phi(\boldsymbol{w}) \end{pmatrix} = 0, \qquad \boldsymbol{w} = (\boldsymbol{x}, \boldsymbol{z})$$

- The gradient of the function Φ is:

$$\nabla_{\boldsymbol{x}_i} \Phi(\boldsymbol{w}) = \nabla h_i(\boldsymbol{x}_i) - \mathcal{A}_i^* \mathbf{prox}_{\sigma h_n^*}\left(\sigma \boldsymbol{b} + \boldsymbol{z}_n - \sigma \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i\right), \quad i \in \mathcal{I}_1,$$

$$\nabla_{\boldsymbol{x}_i} \Phi(\boldsymbol{w}) = \mathbf{prox}_{\sigma h_i^*}(\sigma \boldsymbol{x}_i - \boldsymbol{z}_i) - \mathcal{A}_i^* \mathbf{prox}_{\sigma h_n^*}\left(\sigma \boldsymbol{b} + \boldsymbol{z}_n - \sigma \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i\right), \quad i \in \mathcal{I}_2 \backslash n,$$

$$\nabla_{\boldsymbol{z}_i} \Phi(\boldsymbol{w}) = -\boldsymbol{z}_i/\sigma - \mathbf{prox}_{\sigma h_i^*}(\sigma \boldsymbol{x}_i - \boldsymbol{z}_i)/\sigma, \quad i \in \mathcal{I}_2 \backslash n,$$

$$\nabla_{\boldsymbol{z}_n} \Phi(\boldsymbol{w}) = -\boldsymbol{z}_n/\sigma + \mathbf{prox}_{\sigma h_n^*}(\sigma \boldsymbol{b} + \boldsymbol{z}_n - \sigma \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i)/\sigma.$$

Definition of generalized Jacobian and semismooth

- Let F be a locally Lipschitz continuous mapping. Denote D_F by the set of differentiable points of F. The **B-subdifferential** of F at $\boldsymbol{x}$ is

$$\partial_B F(\boldsymbol{w}) := \left\{ \lim_{k\to\infty} J(\boldsymbol{w}^k) \,|\, \boldsymbol{w}^k \in D_F, \boldsymbol{w}^k \to \boldsymbol{w} \right\},$$

  where $J(\boldsymbol{w})$ is the generalized Jacobian of F at $\boldsymbol{w} \in D_F$. The set $\partial F(\boldsymbol{w}) = co(\partial_B F(\boldsymbol{w}))$ is called **Clarke's generalized Jacobian**.
- F is **semismooth** (or **strongly semismooth**) at $\boldsymbol{w}$ if F is directionally differentiable at $\boldsymbol{w}$ and for any $\boldsymbol{d}$ and $J \in \partial F(\boldsymbol{w} + \boldsymbol{d})$, it holds that

$$\|F(\boldsymbol{w} + \boldsymbol{d}) - F(\boldsymbol{w}) - J\boldsymbol{d}\| = o(\|\boldsymbol{d}\|) \text{ (or } O(\|\boldsymbol{d}\|^2)), \quad \boldsymbol{d} \to 0.$$

  We say F is semismooth (or strongly semismooth) if F is semismooth (or strongly semismooth) for any $\boldsymbol{w}$.

Properties of the nonlinear operator $F(\boldsymbol{w})$

Assume that the functions $\{h_i\}_{i\in\mathcal{I}_1}$ are twice continuously differentiable.

- The nonlinear operator $F(\boldsymbol{w})$ is monotone.
- $F(\boldsymbol{w})$ is Lipschitz continuous.
- Each element of $\partial F(\boldsymbol{w})$ is positive semidefinite for any $\boldsymbol{w}$.
- If the proximal operators prox_{h_i} are (strongly) semismooth, then the map F is (strongly) semismooth.
- $\boldsymbol{w}^*$ is the stationary point if and only if $\boldsymbol{w}^*$ satisfies $F(\boldsymbol{w}^*) = 0$.
- For $\alpha \in (0,1]$, the mapping $T = I - 2\alpha F/L$ is α-averaged.

Outline

1. A nonlinear system from the AL saddle point
2. A semismooth Newton method
3. Convergence analysis
   - Global convergence
   - Local convergence
4. Applications

A primal-dual semismooth Newton step

- Take a Clarke's generalized Jacobian: $J^k \in \hat{\partial} F(\boldsymbol{w}^k)$
- High level idea: obtain $\boldsymbol{d}^k$ by solving a linear system

$$(J^k + \tau_k \mathcal{I})\boldsymbol{d}^k = -F(\boldsymbol{w}^k).$$

- τ_k: crucial for both algorithm efficiency and global convergence. **Need to adjust τ_k adaptively.**
- Comparisons between different semismooth Newton steps:
  - Our algorithm: semismooth Newton steps for both primal variables $\boldsymbol{x}$ and dual variables $\boldsymbol{z}$
  - ALM SSN method: semismooth Newton steps only for the primal variables $\boldsymbol{x}$, then update the multiplier $\boldsymbol{z}$ in a standard way
  - Residual mapping method: semismooth Newton steps on an intermediate variable $\boldsymbol{u}$, and $\boldsymbol{x}$ and $\boldsymbol{z}$ are recovered from $\boldsymbol{u}$.

A globalized semismooth Newton method

- Set $\tau_{k,i} := \kappa\gamma^i \|F(\boldsymbol{w}^k)\|$ for $\kappa > 0$ and $\gamma > 1$
- Compute $\boldsymbol{d}^{k,i}$ by solving linear system

$$(J^k + \tau_{k,i}\mathcal{I})\boldsymbol{d}^{k,i} = -F(\boldsymbol{w}^k)$$

- Compute a trial SSN step: $\bar{\boldsymbol{w}}^{k,i} = \boldsymbol{w}^k + \boldsymbol{d}^{k,i}$
- **Globalization**: for an integer $\zeta \geq 1$, $\nu \in (0,1)$, and $\beta \in (1/2, 1]$, increase the integer i starting from 0 gradually and check

$$\|F(\bar{\boldsymbol{w}}^{k,i})\| \leq \nu \max_{\max(1,k-\zeta+1)\leq j\leq k} \|F(\boldsymbol{w}^j)\|, \quad (1)$$

$$\tau_{k,i} \geq k^\beta. \quad (2)$$

- When (1) or (2) holds, we set $\boldsymbol{w}^{k+1} = \bar{\boldsymbol{w}}^{k,i}$.

Comparisons between different globalization strategies

- Early iteration: τ_k can be large, ensure positive definiteness to speed up. Later iteration: τ_k should be small to guarantee accuracy.
- $\|F(\boldsymbol{w}^k)\|$ **may be increased, but** $\|\boldsymbol{w}^k - \boldsymbol{w}^*\|$ **may be decreased.**
- **No need to switch to first order method**: $\tau_{k,i} \geq k^\beta$ and Lipschitz continuity guarantee decrease of $\|F(\boldsymbol{w})\|^k$.
- Other globalization strategies:
  - Switch between first order and Newton step:

$$\boldsymbol{u}^{k+1} = \begin{cases} \boldsymbol{u}^k + \boldsymbol{d}^k & \text{(Newton step)} \\ \boldsymbol{u}^k + F(\boldsymbol{u}^k) & \text{(First order step)} \end{cases}$$

  - Extra gradient strategy:

$$\begin{cases} \boldsymbol{s}^k & = \boldsymbol{x}^{\boldsymbol{k}} + \beta \boldsymbol{d}^k \\ \boldsymbol{x}_+^k & = \mathbf{prox}_{\lambda_+ h}(\boldsymbol{x} + \alpha \boldsymbol{d} - \lambda_+ \nabla f(\boldsymbol{z}^k)) \end{cases}$$

Derivation of the Jacobian of F

- For any $\boldsymbol{w}$, we define the generalized Jacobian of $F(\boldsymbol{w})$ as

$$\hat{\partial} F(\boldsymbol{w}) := \left\{ \begin{pmatrix} \mathcal{H}_{\boldsymbol{xx}} & \mathcal{H}_{\boldsymbol{xz}} \\ -\mathcal{H}_{\boldsymbol{xz}}^\top & \mathcal{H}_{\boldsymbol{zz}} \end{pmatrix} : \hat{\mathcal{D}}_{h_i} \in \mathcal{D}_{h_i} \text{ for all } i \in \mathcal{I}_2 \right\}.$$

- The Clarke's generalized Jacobian of $\textbf{prox}_{\sigma h_i^*}(\sigma \boldsymbol{x}_i + \boldsymbol{z}_i)$:

$$\begin{aligned} \hat{\mathcal{D}}_{h_i} &:= \nabla^2 h_i(\boldsymbol{x}_i), i \in \mathcal{I}_1, \\ \mathcal{D}_{h_i} &:= \partial \textbf{prox}_{\sigma h_i^*}(\sigma \boldsymbol{x}_i - \boldsymbol{z}_i), i \in \mathcal{I}_2 \backslash n, \\ \mathcal{D}_{h_n} &:= \partial \textbf{prox}_{\sigma h_n^*}(\boldsymbol{b} + \boldsymbol{z}_n/\sigma - \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{x}_i) \end{aligned}$$

- The Fenchel conjugate function of a proper convex function h is $h^*(\boldsymbol{z}) = \sup_{\boldsymbol{x}} \{\langle \boldsymbol{x}, \boldsymbol{z} \rangle - h(\boldsymbol{x})\}$ and the subdifferential is $\partial h(\boldsymbol{x}) := \{\boldsymbol{z} : \ h(\boldsymbol{y}) - h(\boldsymbol{x}) \geq \langle \boldsymbol{z}, \boldsymbol{y} - \boldsymbol{x} \rangle, \ \forall \boldsymbol{y}\}$.

Derivation of the Jacobian of F (con's)

- The matrix operators:

$$\mathcal{H}_{\boldsymbol{xx}} := \begin{pmatrix} \sigma\hat{\mathcal{D}}_{h_1} + \sigma\mathcal{A}_1^*\hat{\mathcal{D}}_{h_n}\mathcal{A}_1 & \cdots & \sigma\mathcal{A}_1^*\hat{\mathcal{D}}_{h_n}\mathcal{A}_{n-1} \\ \vdots & \ddots & \vdots \\ \sigma\mathcal{A}_{n-1}^*\hat{\mathcal{D}}_{h_n}\mathcal{A}_1 & \cdots & \sigma\hat{\mathcal{D}}_{h_{n-1}} + \sigma\mathcal{A}_{n-1}^*\hat{\mathcal{D}}_{h_n}\mathcal{A}_{n-1} \end{pmatrix},$$

$$\mathcal{H}_{\boldsymbol{xz}} := -\left[\mathrm{blkdiag}\left(\left\{\hat{\mathcal{D}}_{h_i}\right\}_{i=1}^{n-1}\right), ((\mathcal{A}_1^*; \cdots, \mathcal{A}_{n-1}^*)\hat{\mathcal{D}}_{h_n})^\top\right],$$

and

$$\mathcal{H}_{\boldsymbol{zz}} := \mathrm{blkdiag}\left(\left\{\frac{1}{\sigma}I - \frac{1}{\sigma}\hat{\mathcal{D}}_{h_i}\right\}_{i=1}^{n}\right),$$

where $\hat{\mathcal{D}}_{h_i} \in \mathcal{D}_{h_i}$ for $i \in \mathcal{I}_2$.

Key: solving the semismooth Newton system

The semismooth Newton system is:

$$\begin{pmatrix} \mathcal{H}_{\boldsymbol{xx}} + \tau\mathcal{I} & \mathcal{H}_{\boldsymbol{xz}} \\ -\mathcal{H}_{\boldsymbol{xz}}^T & \mathcal{H}_{\boldsymbol{zz}} + \tau\mathcal{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{d}_{\boldsymbol{x}} \\ \boldsymbol{d}_{\boldsymbol{z}} \end{pmatrix} = \begin{pmatrix} -F_{\boldsymbol{x}} \\ -F_{\boldsymbol{z}} \end{pmatrix}.$$

Apply the **Gaussian elimination**:

- First, take:

$$\boldsymbol{d}_{\boldsymbol{z}} = (\mathcal{H}_{\boldsymbol{zz}} + \tau\mathcal{I})^{-1}(\mathcal{H}_{\boldsymbol{xz}}^\top \boldsymbol{d}_{\boldsymbol{x}} - F_{\boldsymbol{z}}).$$

- Then the linear system is only with respect to $\boldsymbol{d}_{\boldsymbol{x}}$:

$$\boxed{\mathcal{H}_{\boldsymbol{r}} \boldsymbol{d}_{\boldsymbol{x}} = F_{\boldsymbol{r}},}$$

  where

$$\mathcal{H}_{\boldsymbol{r}} := (\mathcal{H}_{\boldsymbol{xx}} + \mathcal{H}_{\boldsymbol{xz}}(\mathcal{H}_{\boldsymbol{zz}} + \tau\mathcal{I})^{-1}\mathcal{H}_{\boldsymbol{xz}}^\top + \tau\mathcal{I}),$$

$$F_{\boldsymbol{r}} := \mathcal{H}_{\boldsymbol{xz}}(\mathcal{H}_{\boldsymbol{zz}} + \tau\mathcal{I})^{-1}F_{\boldsymbol{z}} - F_{\boldsymbol{x}}.$$

- We essentially only need to handle $n-1$ blocks of variables.

Efficiency: solving the semismooth Newton system

- The definition of $\mathcal{H}_{\boldsymbol{xz}}$ yields

$$\mathcal{H}_{\boldsymbol{r}} = \begin{pmatrix} \overline{\mathcal{D}}_{h_1} + \mathcal{A}_1^* \overline{\mathcal{D}}_{h_n} \mathcal{A}_1 & \cdots & \mathcal{A}_1^* \overline{\mathcal{D}}_{h_n} \mathcal{A}_{n-1} \\ \vdots & \ddots & \vdots \\ \mathcal{A}_{n-1}^* \overline{\mathcal{D}}_{h_n} \mathcal{A}_1 & \cdots & \overline{\mathcal{D}}_{h_{n-1}} + \mathcal{A}_{n-1}^* \overline{\mathcal{D}}_{h_n} \mathcal{A}_{n-1} \end{pmatrix},$$

  where $\overline{\mathcal{D}}_{h_i} = \sigma \hat{\mathcal{D}}_{h_i} + \tilde{\mathcal{D}}_{h_i}, \tilde{\mathcal{D}}_{h_i} = \hat{\mathcal{D}}_{h_i}(\frac{1}{\sigma}\mathcal{I} - \frac{1}{\sigma}\hat{\mathcal{D}}_{h_i} + \tau\mathcal{I})^{-1}\hat{\mathcal{D}}_{h_i}$.
- Direct method: Sparse Cholesky factorization, SMW formula, etc.
- Iterative method: PCG, QMR, etc.

$$\boldsymbol{h}_{\boldsymbol{r}_i} = (\mathcal{H}_{\boldsymbol{r}} \boldsymbol{d}_{\boldsymbol{x}})_i = \overline{\mathcal{D}}_{h_i} \boldsymbol{d}_{\boldsymbol{x}_i} + \mathcal{A}_i^* \overline{\mathcal{D}}_{h_n} \sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{d}_{\boldsymbol{x}_i},$$

  where $\sum_{i=1}^{n-1} \mathcal{A}_i \boldsymbol{d}_{\boldsymbol{x}_i}$ only need to be computed once.
- $\overline{\mathcal{D}}_{h_i}$ can further be used to improve the computational efficiency if the solution is sparse or low-rank!

Outline

1. A nonlinear system from the AL saddle point
2. A semismooth Newton method
3. Convergence analysis
   - Global convergence
   - Local convergence
4. Applications

Global convergence

Theorem (Global convergence)

Suppose that Assumption holds and let $\{\boldsymbol{w}^k\}$ be a sequence generated. Then it holds

$$\lim_{k\to\infty} F(\boldsymbol{w}^k) = 0.$$

First prove $\|F(\boldsymbol{w}^k)\|$ is bounded:

1. The steps using (1) are finite. Easy to prove.
2. The steps using (2) are finite. Easy to prove.
3. let $k_1, k_2, \dots$ be the indices corresponding to the Newton steps with $\tau_{k_j,i} < k_j^\beta$. We have for any $k_n + 1 < j \leq k_{n+1}$,

$$\|F(\boldsymbol{w}^j)\|^2 \leq \|F(\boldsymbol{w}^{j-1})\|^2 + \frac{L^2 M}{(j-1)^{2\beta}} \leq \|F(\boldsymbol{w}^{k_n+1})\|^2 + \sum_{k=k_n+1}^{j-1} \frac{L^2 M}{k^{2\beta}}.$$

We then prove $\lim_{j\to\infty} \|F(\boldsymbol{w}^{k_j+1})\| \to 0$.

Local convergence: local smoothness

Main difficulty: The Jacobian is only positive semidefinite. It is **singular** at the optimal solution. BD regularity does not hold.

- **[Manifold identification]** For a proper, closed, and prox-regular function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$,
  - $\mathcal{M}$: identifiable manifold at $\bar{x}$ for y.
  - f is C^p-partial smooth $(p \geq 2)$ at $\bar{x}$ for $y \in \mathrm{ri}\big(\widehat{\partial} f(\bar{x})\big)$ relative to a C^p-manifold $\mathcal{M}$.
- **[Local smoothness]** For all sufficiently small $t > 0$, the proximal mapping prox_{tf} is C^{p-1}-smooth near $\bar{x} + ty$.

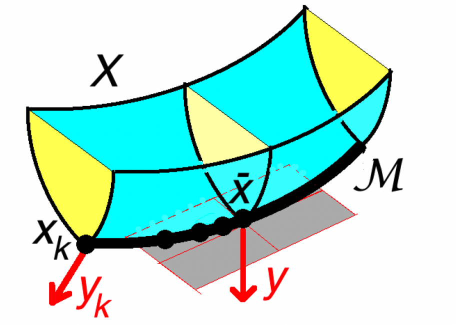


Figure: Manifold identification.

Partial smoothness: Reduced to the regularized Newton method: $(J^k + \mu^k I)d^k = -F(z^k)$

Error bound condition

Definition (LEB)

Let $\boldsymbol{W}^*$ is the solution set of $F(\boldsymbol{w}) = 0$. The local error bound (LEB) condition holds for F if there exist $\epsilon > 0$ and $\gamma > 0$ such that for any $\boldsymbol{w}$ with $\mathrm{dist}(\boldsymbol{w}, \boldsymbol{W}^*) \leq \epsilon$,

$$\|F(\boldsymbol{w})\| \geq \gamma \mathrm{dist}(\boldsymbol{w}, \boldsymbol{W}^*),$$

where $\mathrm{dist}(\boldsymbol{w}, \boldsymbol{W}^*) := \arg\min_{\boldsymbol{u}} \|\boldsymbol{w} - \boldsymbol{u}\|$.

Lemma

If the functions $h_i, i = 1, \ldots, n$ are proper, closed, convex, and piecewise linear-quadratic, then the residual mapping F satisfies the local error bound condition.

Strict Complementarity

- We can reformulate (P) as

$$\min_{\mathbf{x}_1,\cdots,\mathbf{x}_n} \phi(\mathbf{x}) := \sum_{i\in\mathcal{I}_1} h_i(\mathbf{x}_i) + \sum_{i\in\mathcal{I}_2} h_i(\mathbf{x}_i) + \delta_{\mathcal{C}}(\mathbf{x}),$$

where $\mathcal{C} := \{\mathbf{x} : \sum_{i=1}^{n-1} \mathcal{A}_i \mathbf{x}_i + \mathbf{x}_n = b\}$.

Definition (Strict Complementarity)

We say SC holds at $\tilde{\mathbf{x}}$ if $0 \in \mathrm{ri}\,(\partial\phi(\tilde{\mathbf{x}}))$.

- The SC holds at $(\tilde{\mathbf{x}}, \tilde{\mathbf{z}})$ if and only if they satisfy KKT optimality and

$$\tilde{\mathbf{z}}_i \in \mathrm{ri}(\partial h_i(\tilde{\mathbf{x}}_i)),\ i \in \mathcal{I}_2 \backslash n, \quad \tilde{\mathbf{z}}_n \in -\mathrm{ri}(\partial h_n(\tilde{\mathbf{x}}_n)).$$

Local convergence

Assumption (partly smooth)

The functions $h_i, i = 1, \ldots, n$ are C^p-partly smooth.

Lemma

Suppose that Assumptions hold. For any optimal solution $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{z}})$, if the SC is satisfied, F is locally C^{p-1}-smooth in a neighborhood of $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{z}})$.

Theorem (Local superlinear convergence)

Suppose that Assumptions with $p \geq 3$ hold and F satisfies the LEB condition. If $\boldsymbol{w}_\ell$ is close enough to $\tilde{\boldsymbol{w}} \in \boldsymbol{W}^*$ where the SC is satisfied, $\boldsymbol{w}^k$ converges to $\tilde{\boldsymbol{w}}$ Q-superlinearly.

Outline

1. A nonlinear system from the AL saddle point
2. A semismooth Newton method
3. Convergence analysis
   - Global convergence
   - Local convergence
4. Applications

Image restoration with two regularizations.

- The dual problem of image processing is

$$\begin{aligned}\min_{\boldsymbol{y},\boldsymbol{\mu},\boldsymbol{\nu}} \quad & \frac{1}{2}\|\boldsymbol{y}\|^2 + \langle \boldsymbol{y}, \boldsymbol{b}\rangle + \delta^*_{\mathcal{Q}}(\boldsymbol{\nu}) + \delta_{\|\cdot\|_\infty<\lambda}(\boldsymbol{\mu}), \\ \text{s.t.} \quad & \boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{D}^\top \boldsymbol{s} + \boldsymbol{\nu} = 0, \quad \boldsymbol{s} = \boldsymbol{\mu},\end{aligned}$$

- The corresponding AL function is:

$$\begin{aligned}\Phi(\boldsymbol{y}, \boldsymbol{s}; \boldsymbol{q}, \boldsymbol{u}) = {} & \frac{1}{2}\|\boldsymbol{y}\|^2 + \langle \boldsymbol{y}, \boldsymbol{b}\rangle + \frac{1}{2\sigma}\|\mathbb{P}_{\mathcal{Q}}\left(\boldsymbol{u} - \sigma(\boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{D}^\top \boldsymbol{s})\right)\|^2 \\ & + \frac{\sigma}{2}\|\psi_\lambda\left(\boldsymbol{q}/\sigma - \boldsymbol{s}\right)\|^2 - \frac{1}{2\sigma}\left(\|\boldsymbol{u}\|^2 + \|\boldsymbol{q}\|^2\right),\end{aligned}$$

- $\boldsymbol{y}, \boldsymbol{s}$ are the primal variables and $\boldsymbol{q}$, $\boldsymbol{u}$ are the Lagrangian multipliers.
- $\psi_\lambda(\boldsymbol{u}) = \operatorname{sign}(\boldsymbol{u}) \max\{|\boldsymbol{u}| - \lambda, 0\}$, $\mathbb{P}_{\mathcal{Q}}$ is the projection operator.

Image restoration with two regularizations.

- Then the nonlinear system has four terms as follows:

$$F(\boldsymbol{w}) = (F_{\boldsymbol{y}}, F_{\boldsymbol{s}}, F_{\boldsymbol{q}}, F_{\boldsymbol{u}}) = (\nabla_{\boldsymbol{y}}\Phi(\boldsymbol{w})^\top, \nabla_{\boldsymbol{s}}\Phi(\boldsymbol{w})^\top, -\nabla_{\boldsymbol{q}}\Phi(\boldsymbol{w})^\top, -\nabla_{\boldsymbol{u}}\Phi(\boldsymbol{w})^\top)^\top.$$

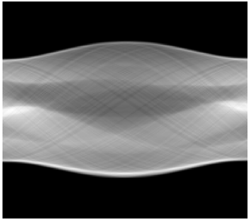
- The simplified semismooth Newton system is a two-block system

$$\boxed{\mathcal{H}_{\boldsymbol{r}}\boldsymbol{d}_{\boldsymbol{x}} = F_{\boldsymbol{r}},}$$

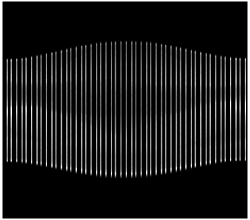
$$(\mathcal{H}_{\boldsymbol{r}})_{\boldsymbol{yy}} = \boldsymbol{A}\overline{D}_1\boldsymbol{A}^\top + (\tau_1 + 1)\mathcal{I},\ (\mathcal{H}_{\boldsymbol{r}})_{\boldsymbol{ys}} = \boldsymbol{A}\overline{D}_1\boldsymbol{D}^\top,\ (\mathcal{H}_{\boldsymbol{r}})_{\boldsymbol{ss}} = \boldsymbol{D}\overline{D}_1\boldsymbol{D}^\top + \overline{D}_2 + \tau_2\mathcal{I},$$
$$(F_{\boldsymbol{r}})_{\boldsymbol{y}} = -\boldsymbol{A}D_1(D_1^{\tau_3})^{-1}F_{\boldsymbol{u}} - F_{\boldsymbol{y}}, (F_{\boldsymbol{r}})_{\boldsymbol{s}} = -\boldsymbol{D}D_1(D_1^{\tau_4})^{-1}F_{\boldsymbol{u}} - D_2(D_2^{\tau_4})^{-1}F_{\boldsymbol{q}} - F_{\boldsymbol{z}},$$

- $D_1 \in \partial\mathbb{P}_{\mathcal{Q}}(\boldsymbol{u} - \sigma(\boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{D}^\top\boldsymbol{s}))$ and $D_2 \in \partial\psi_\lambda(\boldsymbol{q}/\sigma - \boldsymbol{s})$.
- $\tau_1, \tau_2, \tau_3, \tau_4$ correspond to the regularization parameter of $\boldsymbol{y}, \boldsymbol{s}, \boldsymbol{q}, \boldsymbol{u}$.

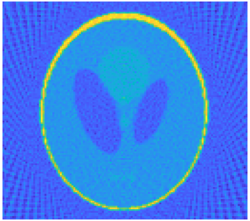
CT image restoration



(a) Radon transform of Phantom

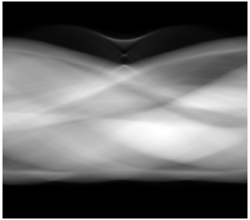


(b) Sampled Radon transform

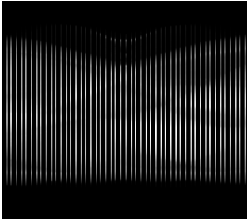


(c) Restored image by FBP

(d) Restored image by ALPDSN

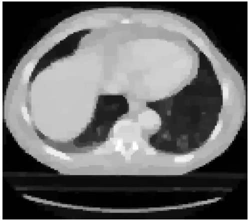


(e) Radon transform of Carcinoma



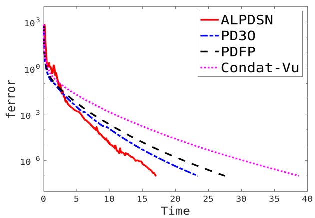
(f) Sampled Radon transform

(g) Restored image by FBP

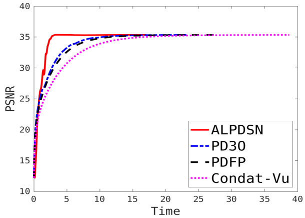


(h) Restored image by ALPDSN

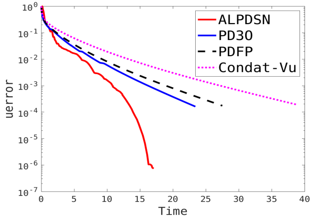
CT image restoration: first row: $\lambda = 0.1$, second row: $\lambda = 0.05$



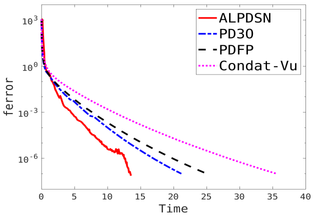
(i) ferror vs time



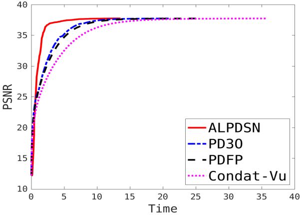
(j) PSNR vs time



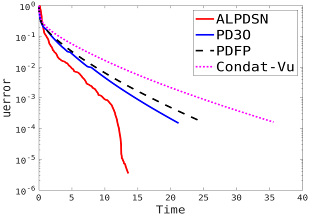
(k) uerror vs time



(l) ferror vs time



(m) PSNR vs time



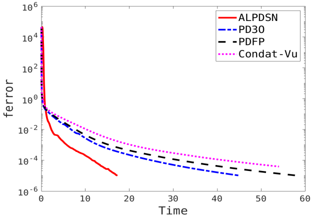
(n) uerror vs time

Image deblurring

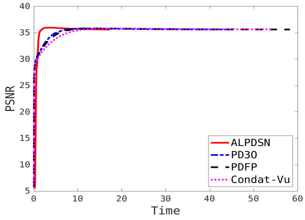
(o) Peppers

(p) Cameraman

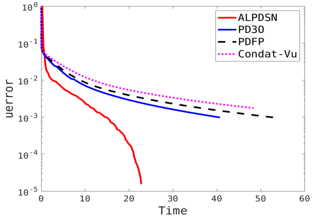
Image deblurring: first row: Pepper, second row: Animals



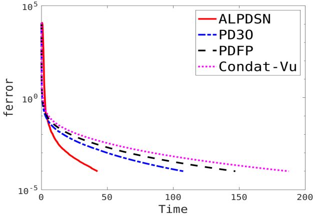
(a) ferror vs time



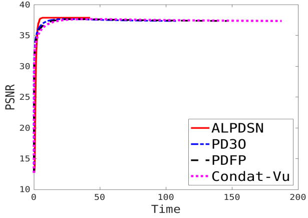
(b) PSNR vs time



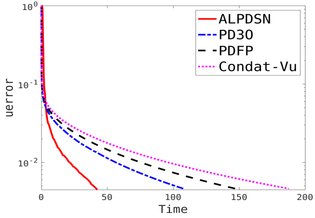
(c) uerror vs time



(d) ferror vs time



(e) PSNR vs time



(f) uerror vs time

Tensor notations

We define a block circular matrix from the frontal slices $X^{(i)}$ of $\mathcal{X}$:

$$\mathrm{bcirc}(\mathcal{X}) := \begin{bmatrix} X^{(1)} & X^{(n_3)} & \dots & X^{(2)} \\ X^{(2)} & X^{(1)} & \dots & X^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ X^{(n_3)} & X^{(n_3-1)} & \dots & X^{(1)} \end{bmatrix}.$$

Definition

The t-product of $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ and $\mathcal{Y} \in \mathbb{C}^{n_2 \times n_4 \times n_3}$ is a tensor $\mathcal{Z} \in \mathbb{C}^{n_1 \times n_4 \times n_3}$ given by $\mathcal{Z} = \mathcal{X} * \mathcal{Y} := \mathrm{fold}(\mathrm{bcirc}(\mathcal{X}) \cdot \mathrm{unfold}(\mathcal{Y}))$, where $\mathrm{unfold}(\mathcal{X})$ takes $\mathcal{X}$ into a block $n_1 n_3 \times n_2$ matrix:

$$\mathrm{unfold}(\mathcal{X}) := \left[(X^{(1)})^\top, (X^{(2)})^\top, \cdots, (X^{(n_3)})^\top\right]^\top,$$

and fold is the inverse of unfold : $\mathrm{fold}(\mathrm{unfold}(\mathcal{X})) = \mathcal{X}$. Moreover, $\mathcal{X} * \mathcal{Y} = \mathcal{Z}$ is equivalent to $\overline{\mathbf{X}}\,\overline{\mathbf{Y}} = \overline{\mathbf{Z}}$.

CTNN model

Definition

The tensor nuclear norm of a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is defined as $\|\mathcal{X}\|_{\mathrm{TNN}} = \frac{1}{n_3}\|\overline{\mathcal{X}}\|_*$, where $\|\overline{\mathcal{X}}\|_*$ denotes the nuclear norm of $\overline{\mathcal{X}}$.

The dual of CTNN model is given by:

$$\begin{aligned} \min_{\boldsymbol{v},\mathcal{W},\mathcal{Z}} \quad & \frac{m}{2}\|\boldsymbol{v}\|^2 - \langle \boldsymbol{v}, \boldsymbol{y} \rangle + \delta^*_{\mathfrak{U}}(-\mathcal{W}) + \delta_{\mathfrak{X}}(\mathcal{Z}) \\ \text{s.t.} \quad & \mathcal{D}^*_{\Omega}(\boldsymbol{v}) - \mathcal{Z} + \mathcal{W} = -\mu F(\mathcal{X}_m), \end{aligned}$$

where $\mathcal{D}_\Omega : \mathbb{R}^{n_1 \times n_2 \times n_3} \to \mathbb{R}^{|\Omega|}$ is the sampling operator, Ω is the given sample region, $\boldsymbol{y}$ and $F(\mathcal{X}_m)$ is the given observed term, $\mathfrak{U} := \{\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} | \|\mathcal{X}\|_\infty \leq c\}$.
$\mathfrak{X} := \{\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} | \|\mathcal{X}\|_{\mathrm{op}} \leq \mu\}$, where $\|\cdot\|_{\mathrm{op}}$ is the dual norm of $\|\cdot\|_{\mathrm{TNN}}$.

CTNN model

- When introducing a slack variable $\mathcal{Z}=\mathcal{Y}$, the AL function is :

$$\Phi(\boldsymbol{v},\mathcal{Y};\mathcal{P},\mathcal{X}) := \frac{m}{2}\|\boldsymbol{v}\|^2 - \langle \boldsymbol{v},\boldsymbol{y}\rangle + \frac{\sigma}{2}\|\mathcal{U}*\hat{\mathcal{S}}_t*\mathcal{V}^\top\|_F^2$$
$$+\frac{1}{2\sigma}\|\mathbf{prox}_{\sigma\delta_{\mathfrak{U}}}\left(\mathcal{X}+\sigma(F(\mathcal{X}_m)+\mathcal{D}_\Omega^*(\boldsymbol{v})-\mathcal{Y})\right)\|_F^2 - \frac{1}{2\sigma}\left(\|\mathcal{X}\|_F^2+\|\mathcal{P}\|_F^2\right),$$

- The nonlinear system has four parts as follows:

$$F(\boldsymbol{w}) = (F_{\boldsymbol{v}},F_{\mathcal{Y}},F_{\mathcal{P}},F_{\mathcal{X}}) = (\nabla_{\boldsymbol{v}}\Phi(\boldsymbol{w})^\top,\nabla_{\mathcal{Y}}\Phi(\boldsymbol{w})^\top,-\nabla_{\mathcal{P}}\Phi(\boldsymbol{w})^\top,-\nabla_{\mathcal{X}}\Phi(\boldsymbol{w})^\top)^\top.$$

- The simplified semismooth Newton system is a two-block system:

$$\boxed{\mathcal{H}_{\boldsymbol{r}}\boldsymbol{d}_{\boldsymbol{x}} = F_{\boldsymbol{r}},}$$

$$(\mathcal{H}_{\boldsymbol{r}})_{\boldsymbol{vv}} = \mathcal{D}_\Omega\overline{D}_1\mathcal{D}_\Omega^* + (\tau_1+1)\mathcal{I}, (\mathcal{H}_{\boldsymbol{r}})_{\boldsymbol{v}\mathcal{Y}} = -\mathcal{D}_\Omega\overline{D}_1,\quad (\mathcal{H}_{\boldsymbol{r}})_{\mathcal{Y}\mathcal{Y}} = \overline{D}_1+\overline{D}_2+\tau_2\mathcal{I},$$
$$(F_{\boldsymbol{r}})_{\boldsymbol{v}} = \mathcal{D}_\Omega D_1(D_1^{\tau_3})^{-1}F_{\mathcal{X}} - F_{\boldsymbol{v}}, (F_{\boldsymbol{r}})_{\mathcal{Y}} = -D_1(D_1^{\tau_4})^{-1}F_{\mathcal{X}} - D_2(D_2^{\tau_4})^{-1}F_{\mathcal{P}} - F_{\mathcal{Y}},$$

- $D_1\in\partial\mathbf{prox}_{\sigma\delta_{\mathfrak{U}}}\left(\mathcal{X}+\sigma(F(\mathcal{X}_m)+\mathcal{D}_\Omega^*(\boldsymbol{u})-\mathcal{Y})\right), D_2\in\partial\left(\mathcal{U}*\hat{\mathcal{S}}_t*\mathcal{V}^\top\right),$

Description of the operation of D_2

- Denote $\mathcal{V} = [\mathcal{V}_1, \mathcal{V}_2]$, $\mathcal{V}_1 \in \mathbb{R}^{n_2 \times n_1 \times n_3}$, $\mathcal{V}_2 \in \mathbb{R}^{n_2 \times (n_2 - n_1) \times n_3}$. Then applying D_2 to a tensor $\mathcal{G} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ yields

$$D_2(\mathcal{G}) = \mathcal{U} * \left[\frac{\Omega^{\mu}_{\sigma,\sigma} + \Omega^{\mu}_{\sigma,-\sigma}}{2} \odot \mathcal{G}_1 + \frac{\Omega^{\mu}_{\sigma,\sigma} - \Omega^{\mu}_{\sigma,-\sigma}}{2} \odot \mathcal{G}_1^{\top}, (\Omega^{\mu}_{\sigma,0} \odot (\mathcal{G}_2)) \right] * \mathcal{V}^{\top},$$

  where $\sigma = [\sigma^{(1)}, \cdots, \sigma^{(n_3)}]$ is the tensor singular value of $\mathcal{X}$

- $\mathcal{G}_1 = \mathcal{U}^{\top} * \mathcal{G} * \mathcal{V}_1 \in \mathbb{R}^{n_1 \times n_1 \times n_3}$, $\mathcal{G}_2 = \mathcal{U}^{\top} * \mathcal{G} * \mathcal{V}_2 \in \mathbb{R}^{n_1 \times (n_2 - n_1) \times n_3}$ and for the k-th frontal slices, $(\Omega^{\mu}_{\sigma,\sigma})^{(k)}$ is defined by:

$$(\Omega^{\mu}_{\sigma,\sigma})^{(k)}_{ij} := \begin{cases} \partial_B \mathbf{prox}_{\mu\|\cdot\|_1}(\sigma_i^{(k)}), & \text{if } \sigma_i^{(k)} = \sigma_j^{(k)}, \\ \left\{ \frac{\mathbf{prox}_{\mu\|\cdot\|_1}(\sigma_i^{(k)}) - \mathbf{prox}_{\mu\|\cdot\|_1}(\sigma_j^{(k)})}{\sigma_i^{(k)} - \sigma_j^{(k)}} \right\}, & \text{otherwise.} \end{cases}$$

  $\Omega_{\sigma,-\sigma}$ and $\Omega_{\sigma,0}$ are defined analogously.

CTNN model

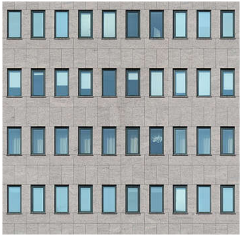
Define $\pi_r := \max\{\pi_d, \pi_u, \pi_w, \pi_z\}$, where

$$\pi_d = \frac{\|\mathcal{Y} - \mu F(\mathcal{X}_m) - \mathcal{D}_\Omega^*(\boldsymbol{v}) - \mathcal{W}\|_F}{1 + \|\mu F(\mathcal{X}_m)\|_F}, \quad \pi_u = \frac{\|m\boldsymbol{v} - \boldsymbol{y} + \mathcal{D}_\Omega(\mathcal{X})\|_F}{1 + \|\boldsymbol{y}\|},$$

$$\pi_w = \frac{\|\mathcal{X} - \mathbf{prox}_{\delta_{\mathfrak{U}}}(\mathcal{X} - \mathcal{W})\|_F}{1 + \|\mathcal{X}\|_F + \|\mathcal{W}\|_F}, \quad \pi_y = \frac{\|\mathcal{Y} - \mathbf{prox}_{\delta_{\mathfrak{X}}}(\mathcal{Y} + \mathcal{X})\|_F}{1 + \|\mathcal{Y}\|_F + \|\mathcal{X}\|_F},$$

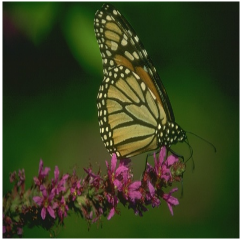
and $\mathcal{W} = \mathcal{R} + \frac{1}{\sigma}\mathbf{prox}_{\delta_{\mathfrak{U}}}(-\sigma\mathcal{R})$ where $\mathcal{R} = -\frac{1}{\sigma}\mathcal{X} - \mu F(\mathcal{X}_m) - \mathcal{D}_\Omega^*(\boldsymbol{u}) + \mathcal{Y}$. We stop the algorithms when $\pi_r < 10^{-7}$.

(a) Texture1



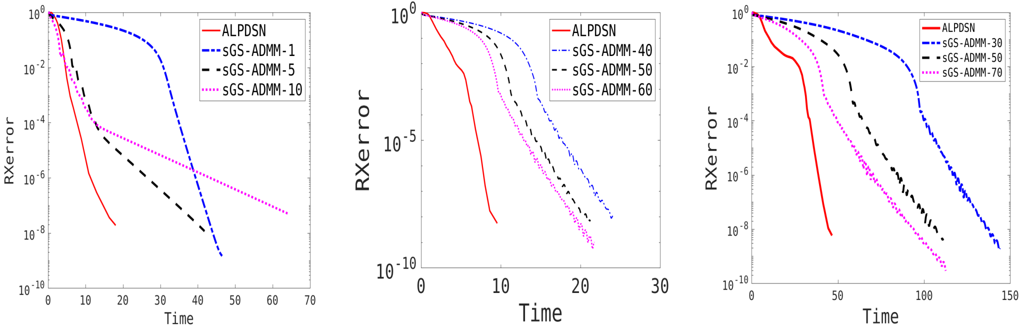
(b) Texture2

(c) Zebra



(d) Butterfly

CTNN model



(a) Synthetic data (b) Texture1 (c) Zebra

Figure: Comparison of ALPDSN and sGs-ADMM on the CTNN model.

SDP

- Conside the dual SDP:

$$\begin{aligned} \min_{\boldsymbol{y},\boldsymbol{s}} \quad & -\langle \boldsymbol{b}, \boldsymbol{y} \rangle \,, \\ \text{s.t.} \quad & \boldsymbol{c} - \mathcal{A}^*(\boldsymbol{y}) = \boldsymbol{s}, \\ & \boldsymbol{s} \in \mathcal{S}_+^n, \boldsymbol{y} \in \mathbb{R}^m. \end{aligned}$$

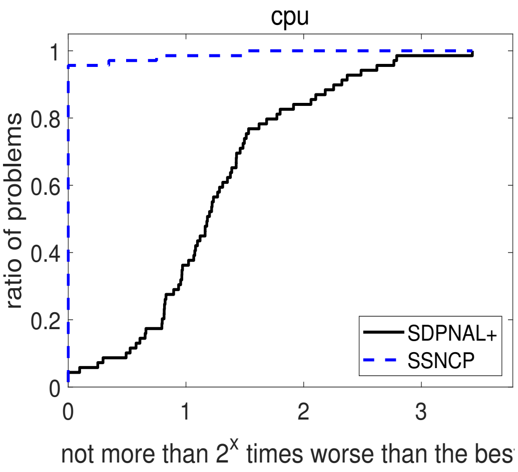
- The augmented Lagrangian function is defined as:

$$\mathbb{L}_\sigma(\boldsymbol{y}; \boldsymbol{x}) = -\langle \boldsymbol{b}, \boldsymbol{y} \rangle + \delta_{\mathcal{S}_+^n}(\boldsymbol{s}) - \frac{1}{2\sigma}\|\boldsymbol{x}\|^2 + \frac{\sigma}{2}\|\Pi_{\mathcal{S}_+^n}(\mathcal{A}^*(\boldsymbol{y}) - \boldsymbol{c} + \frac{1}{\sigma}\boldsymbol{x})\|^2.$$

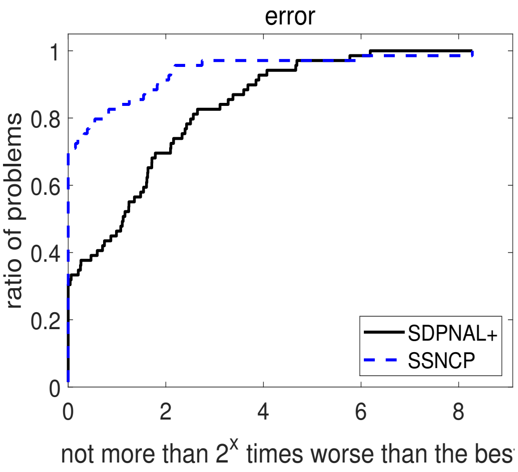
- Therefore, the augmented Lagrangian dual problem can be formulated as:

$$\max_{\boldsymbol{x}} \min_{\boldsymbol{y}} \mathbb{L}_\sigma(\boldsymbol{y}; \boldsymbol{x}).$$

Performance profiles of computational time and error



(a) SSNCP v.s. SDPNAL+



(b) SSNCP v.s. SDPNAL+

Performance: SDP

id	SSNCP				SDPNAL+			
	η_p	η_d	iter	time	η_p	η_d	iter	time
1dc.2048	1.3-7	2.0-7	65	**897.5**	9.8-7	5.6-7	113	959.8
1et.1024	7.1-7	1.1-7	91	**193.1**	5.3-7	9.1-7	226	558.3
1et.2048	1.1-7	3.3-7	157	**1621.8**	8.2-7	6.3-7	692	2851.0
1tc.1024	2.5-7	7.6-7	160	**273.1**	1.0-6	6.7-7	1117	554.5
1tc.2048	5.0-7	8.2-7	162	**2409.6**	2.4-12	1.0-6	1087	4284.4
1zc.1024	2.1-7	3.3-7	32	**16.6**	2.9-7	5.0-7	107	44.6
1zc.2048	3.3-8	5.8-8	50	**181.7**	1.6-7	7.0-7	105	352.4
1zc.4096	2.9-8	2.9-7	52	**1624.7**	3.2-7	9.4-7	107	3171.2
2dc.1024	3.5-7	7.2-7	110	**482.0**	9.8-7	9.8-7	1137	1262.3
2dc.2048	3.3-7	8.9-7	114	**2573.7**	9.8-7	9.7-7	1136	7203.1
G51.1000	9.9-8	9.0-7	149	**85.8**	8.3-11	1.6-7	188	368.3
G52.1000	7.0-7	1.6-7	258	**733.0**	9.5-7	9.6-7	3771	1152.7
G53.1000	9.3-7	2.5-7	449	**1051.4**	9.3-7	7.7-7	6493	1290.3

Many Thanks For Your Attention!

- 教材：刘浩洋, 户将, 李勇锋，文再文，最优化：建模、算法与理论; `http://bicmr.pku.edu.cn/~wenzw/optbook.html`
- 截止2023年6月累计印刷5次2.2万余本