ブラインド・デコンボリューションに対する非平滑正則化付き DC 最適化アプローチ

05001365　総合研究大学院大学　*高橋翔大　TAKAHASHI Shota
02303360　統計数理研究所　田中未来　TANAKA Mirai
統計数理研究所　池田思朗　IKEDA Shiro

1.　ブラインド・デコンボリューション

信号 $\boldsymbol{f}, \boldsymbol{g} \in \mathbb{R}^m$ の畳み込み

$$\tilde{\boldsymbol{y}} = \boldsymbol{f} * \boldsymbol{g} \tag{1}$$

が観測される状況を考える. $\tilde{\boldsymbol{y}}$ が与えられたときに信号 $\boldsymbol{f}, \boldsymbol{g}$ を求めることをブラインド・デコンボリューションという. ブラインド・デコンボリューションは信号処理, 画像処理に現れる問題である. 例えば, ぼけた画像 $\tilde{\boldsymbol{y}}$ からブレ $\boldsymbol{f}$ を除去し, 原画像 $\boldsymbol{g}$ を求める応用 (図 1) があり, 天文学, 医用画像処理でも広く利用されている.

実際にブラインド・デコンボリューションを行う際は, 信号 $\boldsymbol{f}, \boldsymbol{g}$ が既知の部分空間にあると仮定する. つまり, 線形演算子 $\tilde{\boldsymbol{B}}, \tilde{\boldsymbol{A}}$ に対し, $\boldsymbol{h} \in \mathbb{R}^{d_1}, \boldsymbol{x} \in \mathbb{R}^{d_2}$ が存在して

$$\boldsymbol{f} = \tilde{\boldsymbol{B}}\boldsymbol{h}, \quad \boldsymbol{g} = \tilde{\boldsymbol{A}}\boldsymbol{x}$$

が成り立つと仮定する. (1) の両辺に $m \times m$ 離散フーリエ変換 $\sqrt{m}\boldsymbol{F}$ を施すことで, 畳み込みの離散フーリエ変換は離散フーリエ変換の積で表せる. よって, $\boldsymbol{y} = \frac{1}{\sqrt{m}}\boldsymbol{F}\tilde{\boldsymbol{y}}$, $\boldsymbol{B} = \boldsymbol{F}\tilde{\boldsymbol{B}}$, $\overline{\boldsymbol{A}} = \boldsymbol{F}\tilde{\boldsymbol{A}}$ とおくことで (1) は

$$\boldsymbol{y} = \boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} \tag{2}$$

となる. ここで, $\odot$ はアダマール積, $\bar{\cdot}$ は複素数の共役である.

式 (2) を近似的に満たす $[\boldsymbol{h}, \boldsymbol{x}]^\top$ を最適化問題

$$\min_{[\boldsymbol{h},\boldsymbol{x}]^\top \in \mathrm{cl}\ C} \quad \Psi(\boldsymbol{h}, \boldsymbol{x}) := F(\boldsymbol{h}, \boldsymbol{x}) + G(\boldsymbol{h}, \boldsymbol{x}) \tag{3}$$

を解くことで求めることを考える. ただし, $F(\boldsymbol{h}, \boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} - \boldsymbol{y}\|_2^2$, $C \subset \mathbb{R}^{d_1+d_2}$ は制約に対応する非空な開凸集合, $G : \mathbb{R}^{d_1+d_2} \to (-\infty, +\infty]$ は $\boldsymbol{h}, \boldsymbol{x}$ に課される正則化に対応する凸関数とする. (3) は非凸最適化問題になっており, 局所最適解が多数存在する.

最適化問題 (3) において, $F(\boldsymbol{h}, \boldsymbol{x})$ の勾配は Lipschitz 連続でない. そのため, 勾配法や近接勾配法といった従来アルゴリズムを利用するためには, ステップ幅を工夫して調整しなければならない.

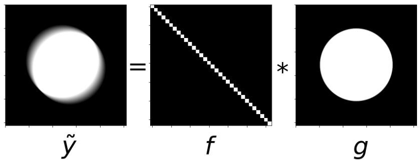

図 1: 画像処理におけるモデル (1) の例.

2.　DC 最適化による解法

2.1.　DC 最適化問題

目的関数が凸関数の差で表せる非凸最適化問題は DC 最適化問題と呼ばれる. DC 最適化問題に対しては効率の良いアルゴリズムが知られている. $F(\boldsymbol{h}, \boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{B}\boldsymbol{h} \odot \overline{\boldsymbol{A}\boldsymbol{x}} - \boldsymbol{y}\|_2^2$ は凸関数でないが, 凸関数 F_1, F_2 によって次のように表せる:

$$\begin{aligned}
F(\boldsymbol{h}, \boldsymbol{x}) &= F_1(\boldsymbol{h}, \boldsymbol{x}) - F_2(\boldsymbol{h}, \boldsymbol{x}),\\
F_1(\boldsymbol{h}, \boldsymbol{x}) &= \frac{1}{4}\|\boldsymbol{B}\boldsymbol{h}\|_4^4 + \frac{1}{4}\|\boldsymbol{A}\boldsymbol{x}\|_4^4\\
&\quad + \frac{1}{2}(\|\boldsymbol{B}\boldsymbol{h} \odot \boldsymbol{A}\boldsymbol{x}\|_2^2 + \|\boldsymbol{y} \odot \boldsymbol{B}\boldsymbol{h}\|_2^2 + \|\boldsymbol{A}\boldsymbol{x}\|_2^2 + \|\boldsymbol{y}\|_2^2),\\
F_2(\boldsymbol{h}, \boldsymbol{x}) &= \frac{1}{4}\|\boldsymbol{B}\boldsymbol{h}\|_4^4 + \frac{1}{4}\|\boldsymbol{A}\boldsymbol{x}\|_4^4 + \frac{1}{2}\|\bar{\boldsymbol{y}} \odot \boldsymbol{B}\boldsymbol{h} + \overline{\boldsymbol{A}\boldsymbol{x}}\|_2^2.
\end{aligned}$$

2.2.　DC アルゴリズム

前述した (3) の DC 表現では, F_1 は L-smooth でない. さらに, スパース正則化に対応する G は微分可能でない. そのため, 既存のアルゴリズムの収束解析の仮定を満たさない. そのような問題に対しても有効なアルゴリズムとして, Bregman Proximal DC Algorithm with Extrapolation (BPDCAe) [3] がある. (3) に対し, BPDCAe を適用することを考える.

$\mathcal{C}^1$ 級凸関数 $H : \mathbb{R}^{d_1+d_2} \to (-\infty, +\infty]$ に対し, Bregman 距離 $D_H : \mathrm{dom}\ H \times \mathrm{int}\ \mathrm{dom}\ H \to \mathbb{R}_+$ を次のように定義する:

$$D_H(\boldsymbol{z}, \boldsymbol{w}) = H(\boldsymbol{z}) - H(\boldsymbol{w}) - \langle \nabla H(\boldsymbol{w}), \boldsymbol{z} - \boldsymbol{w} \rangle.$$

組 (F_1, H) に対し, $LH - F_1, LH + F_1$ が凸関数となる定数 $L > 0$ が存在するとき, (F_1, H) は L-smooth adaptable (L-smad) [1] であるという．BPDCAe の収束解析ではこれを仮定する．

最適化問題 (3) に対して, H を次のように与える:

$$H(\boldsymbol{h},\boldsymbol{x}) = \frac{1}{4}\left(\|\boldsymbol{h}\|_2^2 + \|\boldsymbol{x}\|_2^2\right)^2 + \frac{1}{2}\left(\|\boldsymbol{h}\|_2^2 + \|\boldsymbol{x}\|_2^2\right).$$

$\boldsymbol{b}_j, \boldsymbol{a}_j$ をそれぞれ $\boldsymbol{B}, \boldsymbol{A}$ の第 j 行ベクトルとする．この H に対し,

$$L \geq \sum_{j=1}^{m}\{3\|\boldsymbol{b}_j\|_2^4 + 3\|\boldsymbol{a}_j\|_2^4 + \|\boldsymbol{b}_j\|_2^2\|\boldsymbol{a}_j\|_2^2 + |y_j|^2\|\boldsymbol{b}_j\|_2^2 + \|\boldsymbol{a}_j\|_2^2\}$$

を満たす L で, (F_1, H) は L-smad となる．この H を用いた BPDCAe を Algorithm 1 に示す．しかし, この L は大きいため, 数値実験ではこの L を調整した値を使用した．

Algorithm 1 (3) に対する BPDCAe [3]

入力: $\boldsymbol{z}^0 = \boldsymbol{z}^{-1} \in \mathrm{cl}\, C, \{\beta_k\}_{k=0}^{\infty}, \lambda \in (0, \frac{1}{L})$.

for $k = 0, 1, 2, \cdots,$ **do**

$\boldsymbol{\xi}^k \in \partial F_2(\boldsymbol{z}^k)$,

$\boldsymbol{w}^k = \boldsymbol{z}^k + \beta_k(\boldsymbol{z}^k - \boldsymbol{z}^{k-1})$,

$$\boldsymbol{z}^{k+1} = \mathop{\arg\min}_{\boldsymbol{z}\in \mathrm{cl}\, C}\{G(\boldsymbol{z}) + \langle \nabla F_1(\boldsymbol{w}^k) - \boldsymbol{\xi}^k, \boldsymbol{z}\rangle + \frac{1}{\lambda}D_H(\boldsymbol{z}, \boldsymbol{w}^k)\}$$

3.　数値実験: 画像復元への応用

以下のように問題設定を行った：

- $\boldsymbol{h} \in \mathbb{R}^{d_1}$ はぼかしカーネルで $\sqrt{d_1} \times \sqrt{d_1}$ ピクセル内に成分をもつ．
- $\boldsymbol{x} \in \mathbb{R}^{d_2}$ は離散ウェーブレット係数．
- $\tilde{\boldsymbol{B}}$ は $\boldsymbol{h}$ のピクセル数を揃える操作, $\tilde{\boldsymbol{A}}$ は離散ウェーブレット逆変換．
- $G(\boldsymbol{h},\boldsymbol{x}) = \theta\|\boldsymbol{h}\|_1$ ($\boldsymbol{x}$ は正則化を行わない)．
- $C = \{\boldsymbol{h} \in \mathbb{R}^{d_1}, \boldsymbol{x} \in \mathbb{R}^{d_2} \mid \boldsymbol{h} \geq \boldsymbol{0}, \boldsymbol{x} \geq \boldsymbol{0}\}$.

[2] では微分可能な ℓ_2 ノルムを含んだ正則化を行っている．一方, BPDCAe は微分不可能な正則化に対しても有効で, 子問題は閉形式で解くことが可能である．従来アルゴリズムとして, BPDCA [3], FISTA, 交互最小化と比較実験を行った．ただし, $F(\boldsymbol{h},\boldsymbol{x})$ の勾配は Lipschitz 連続でないので, FISTA のステップ幅はバックトラッキングによって得た．BPDCA, BPDCAe, FISTA の最大反復は 30,000, 交互最小化のみ 3,000 とした．交互最小化は各反復で子問題を FISTA によって近似的に解いた．

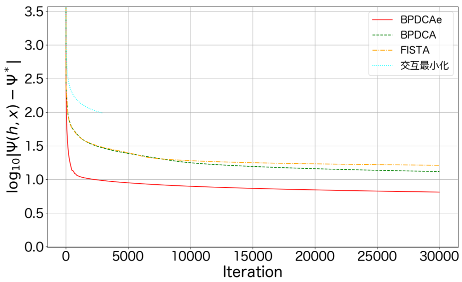

図 2: 数値実験結果 ($\theta = 0.01$). $\boldsymbol{h}^*, \boldsymbol{x}^*$ を正解データとし, $\Psi^* = \Psi(\boldsymbol{h}^*, \boldsymbol{x}^*)$.

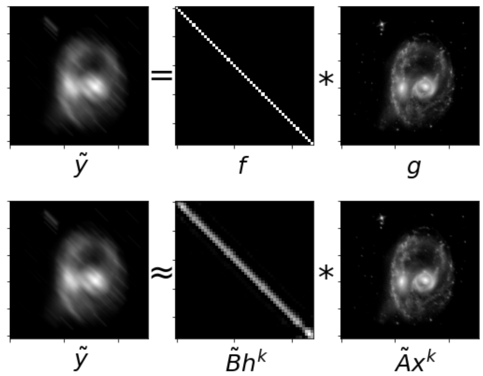

図 3: 上段: ぼけ画像 ($\tilde{\boldsymbol{y}}$), 正解画像 ($\boldsymbol{f}, \boldsymbol{g}$)．下段: BPDCAe による復元画像 ($\tilde{\boldsymbol{B}}\boldsymbol{h}^k, \tilde{\boldsymbol{A}}\boldsymbol{x}^k$).

図 2 は目的関数値の減少を比較しているが, BPDCAe が最も収束が速いことがわかる．また, 図 3 は BPDCAe の反復終了後の復元結果であり, ぼけをある程度除去し, 復元していることがわかる．具体的な画像に対する数値実験結果については当日発表する．

参考文献

[1] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd: First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems, *SIAM Journal on Optimization*, 28(3):2131–2151, 2018.

[2] X. Li, S. Ling, T. Strohmer, and K. Wei: Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *Applied and Computational Harmonic Analysis*, 47(3):893–934, 2019.

[3] S. Takahashi, M. Fukuda, and M. Tanaka: New Bregman proximal type algorithms for solving DC optimization problems, *arXiv preprint*, arXiv:2105.04873, 2021.