

凸最適化に基づく劣決定 BSS に対する ADMM アルゴリズムの適用*

☆ 皆川朋樹 (農工大), 松本和樹 (早大), 山田宏樹, 矢田部浩平 (農工大)

1 はじめに

劣決定な音源分離は、わずかな観測信号から音源信号を推定しなければならない難しい問題である。そのため、高度な音源モデルを用いて定式化されることが多いが、混合系と音源を同時に推定する非凸で複雑な最適化問題となり、初期値依存性などの困難さが生じる。そこで、事前に定義した平面波を空間モデルとして固定し、音源のみを求める凸最適化問題として定式化した手法が提案され、音源方向と音源を同時に推定できる頑健な方法が実現されている [1]。しかし、空間モデルの性質から条件数の悪い行列によってモデルが表現されるので、従来法で用いていた forward-backward 型の主双対分離 (PDS) アルゴリズムでは多くの反復を要することや、そもそもステップサイズパラメータの選択がシビアであるという課題があった。本稿では、ステップサイズによらず無条件収束し、経験的に収束が速い交互方向乗数法 (ADMM) を新たに導出した。ADMM を用いたことで、高速かつ安定性の高いアルゴリズムを実現した。

2 凸最適化による音源分離

本稿では、マイクロホンの位置は既知とし、音源分離を凸最適化によって扱う。\$N\$ 個の音源 \$\mathbf{s}[f, t] = [s_1[f, t], \dots, s_N[f, t]]^T \in \mathbb{C}^N\$ が \$M\$ 個のマイクロホンで観測される時、時間周波数領域における観測信号 \$\mathbf{x}[f, t] = [x_1[f, t], \dots, x_M[f, t]]^T \in \mathbb{C}^M\$ は、

$$\mathbf{x}[f, t] \approx \mathbf{A}[f] \mathbf{s}[f, t] \quad (1)$$

と表せる。ただし、\$f\$ と \$t\$ は周波数と時間のインデックスであり、\$(\cdot)^T\$ は転置である。また、\$\mathbf{A}[f] \in \mathbb{C}^{M \times N}\$ は空間の辞書行列であり、ステアリングベクトル

$$\mathbf{a}_{\theta_k}[f] = [\exp(-j\frac{\omega_f}{c} \mathbf{u}_k^T \mathbf{r}_1), \dots, \exp(-j\frac{\omega_f}{c} \mathbf{u}_k^T \mathbf{r}_M)]^T$$

を用いて \$\mathbf{A}[f] = [\mathbf{a}_{\theta_1}[f], \dots, \mathbf{a}_{\theta_K}[f]]\$ で与える。ただし、\$\mathbf{r}_m\$ は \$m\$ 番目のマイクロホンの座標、\$\mathbf{u}_k\$ は方向 \$\theta_k\$ に対応する単位ベクトルであり、\$c\$ は音速、\$\omega_f = 2\pi f\$、\$j\$ は虚数単位である。式 (1) に基づき、音源分離問題を以下の凸最適化問題として定式化できる [1]。

$$\min_{\mathbf{y}} \|\mathbf{x} - \mathbf{A}\mathbf{y}\|_2^2 + \mu \|\mathbf{y}\|_{2,1} + \nu \|\mathbf{y}\|_{1,2}^2 \quad (2)$$

Algorithm 1 Proposed ADMM algorithm

Input: \$\chi^{[1]}, \mathbf{v}_1^{[1]}, \mathbf{v}_2^{[1]}, \mathbf{u}_0^{[1]}, \mathbf{u}_1^{[1]}, \mathbf{u}_2^{[1]}, \mu, \nu, \rho, \alpha\$

Output: \$\mathbf{y}^{[L+1]}\$

```

1: for $l = 1, \dots, L$ do
2:   $\mathbf{y}^{[l+1]} = (\mathbf{A}^H \mathbf{A} + 2\mathbf{I})^{-1}$
      $(\mathbf{A}^H(\chi^{[l]} - \mathbf{u}_0^{[l]}) + (\mathbf{v}_1^{[l]} - \mathbf{u}_1^{[l]}) + (\mathbf{v}_2^{[l]} - \mathbf{u}_2^{[l]}))$
3:   $\xi_0 = \alpha \mathbf{A}\mathbf{y}^{[l+1]} + (1 - \alpha)\chi^{[l]}$
4:   $\xi_1 = \alpha \mathbf{y}^{[l+1]} + (1 - \alpha)\mathbf{v}_1^{[l]}$
5:   $\xi_2 = \alpha \mathbf{y}^{[l+1]} + (1 - \alpha)\mathbf{v}_2^{[l]}$
6:   $\chi^{[l+1]} = \text{prox}_{\frac{1}{\rho} \|\cdot\|_2} [\xi_0 + \mathbf{u}_0^{[l]}]$
7:   $\mathbf{v}_1^{[l+1]} = \text{prox}_{\frac{\nu}{\rho} \|\cdot\|_{2,1}} [\xi_1 + \mathbf{u}_1^{[l]}]$
8:   $\mathbf{v}_2^{[l+1]} = \text{prox}_{\frac{\nu}{\rho} \|\cdot\|_{1,2}^2} [\xi_2 + \mathbf{u}_2^{[l]}]$
9:   $\mathbf{u}_0^{[l+1]} = \mathbf{u}_0^{[l]} + \xi_0 - \chi^{[l+1]}$
10:  $\mathbf{u}_1^{[l+1]} = \mathbf{u}_1^{[l]} + \xi_1 - \mathbf{v}_1^{[l+1]}$
11:  $\mathbf{u}_2^{[l+1]} = \mathbf{u}_2^{[l]} + \xi_2 - \mathbf{v}_2^{[l+1]}$
12: end for

```

ただし、分離音 \$\mathbf{y}\$ に対する正則化項はそれぞれ

$$\|\mathbf{y}\|_{2,1} = \sum_k \left(\sum_{f,t} |y_k[f, t]|^2 \right)^{\frac{1}{2}}$$

$$\|\mathbf{y}\|_{1,2}^2 = \sum_{f,t} \left(\sum_k |y_k[f, t]| \right)^2$$

と定義し、式 (1) を \$\mathbf{A}\mathbf{y}\$ によって略記した。この問題の解 \$\mathbf{y}^*\$ には \$\theta_1, \dots, \theta_K\$ の各方向に対する推定分離音が含まれているので、エネルギーの大きい方向を選ぶことで音源方向と音源信号を同時に推定する [1]。

3 提案アルゴリズム

式 (2) の定式化によって、劣決定な場合にも適用でき、初期値の選択に頑健な音源分離手法が得られたが、文献 [1] では PDS アルゴリズムを用いて最適化しており、多くの反復を要するという課題があった。実際、PDS アルゴリズムはモデル行列の条件数が悪いときに収束が遅いことが知られているが、式 (2) の \$\mathbf{A}\$ の各行列は空間モデルの性質から必然的に条件数の悪い行列になるので、PDS との相性が悪いと言える。そこで、そのような悪条件性に比較的強い ADMM アルゴリズム [2] を用いて式 (2) を解くことを考える。式 (2) の各項について変数分離した等価な問題

$$\begin{aligned} \min_{\mathbf{y}, \mathbf{v}_1, \mathbf{v}_2, \chi} \quad & \|\mathbf{x} - \chi\|_2^2 + \mu \|\mathbf{v}_1\|_{2,1} + \nu \|\mathbf{v}_2\|_{1,2}^2 \\ \text{s.t.} \quad & \chi = \mathbf{A}\mathbf{y}, \quad \mathbf{v}_1 = \mathbf{y}, \quad \mathbf{v}_2 = \mathbf{y} \end{aligned}$$

* Application of ADMM algorithm to convex-optimization-based underdetermined BSS. By Tomoki MINA-GAWA (Tokyo University of Agriculture and Technology), Kazuki MATSUMOTO (Waseda University), Koki YAMADA and Kohei YATABE (Tokyo University of Agriculture and Technology)

に対して ADMM を適用することで **Algorithm 1** が得られる。ただし、 $(\cdot)^H$ は複素共役転置、 \mathbf{I} は単位行列を表し、 $\alpha \in (0, 2)$ である。また、 $\text{prox}_{\lambda g}$ は関数 g の近接作用素であり、パラメータ $\lambda > 0$ に対して

$$\text{prox}_{\lambda g}[\mathbf{v}] = \arg \min_{\mathbf{x} \in \mathbb{C}^N} \left[g(\mathbf{x}) + \frac{1}{2\lambda} \|\mathbf{x} - \mathbf{v}\|_2^2 \right]$$

によって与えられる。提案アルゴリズムに現れる近接作用素は全て効率的に計算することができる。また、 $\mathbf{A}^H \mathbf{A}$ は $K \times K$ の行列を対角に連結したブロック対角行列なので、逆行列演算はブロックごとに計算できる。次の実験で見ると K は比較的小さな数なので、逆行列演算の計算量は問題にならない。

4 実験

SiSEC 2011 の dev1 liverec を用いて四つの音声の分離を行った。図-1 に模式的に示すが、音源方向は $-50, -10, 15, 45$ 度であり、5 cm 間隔の二つのマイクロホンで録音されたデータを用いた。残響時間は 130 ms である。空間辞書に用いる平面波は、角度の間隔が整数値 (9, 6, 5, 4, 3 度) となるように、個数 K を 21, 31, 37, 46, 61 に設定した。短時間 Fourier 変換には 4096 サンプルの Hann 窓を用い、窓のシフト量は 1024 サンプルとした。モデルパラメータは $\mu = 2$, $\nu = 100$ とし、アルゴリズムは $\rho = 10$, $\alpha = 1.5$ と設定した。比較対象として、PDS アルゴリズム [1] を用いた。反復回数は 500 回とし、評価指標には ΔSDR を用いた。

まず、最終反復における分離結果を図-2 に示す。ただし、横軸は平面波辞書の角度間隔を表し、音源ごとに異なる図として描いた。今回の設定では、全ての音源において ADMM は PDS よりも高い性能を示した。また、平面波辞書の角度を細かくしても、今回の設定では性能があまり変わらないことがわかった。

次に、反復ごとの分離性能を図-3 に示す。左の PDS は、平面波辞書の個数 K が増加するにつれ、より多くの反復が必要であることがわかる。これは、平面波辞書によって構成された行列は、特に低周波帯域において悪条件になり、PDS が行列の悪条件性の影響を受けやすいことが原因であると考えられる。一方 ADMM は、平面波辞書の個数 K によらず安定して分離できることがわかった。これは、 \mathbf{A} の小さな特異値の影響を ADMM アルゴリズムに現れる逆行列演算が補正していることが要因だと考えられる。

5 むすび

本稿では、平面波辞書を用いた凸最適化による劣決定音源分離手法に対し、ADMM による新たなアルゴ

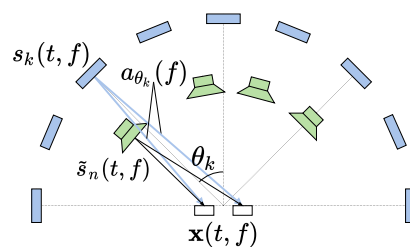


図-1 実験条件の模式図 (K 個の長方形は平面波を表す)

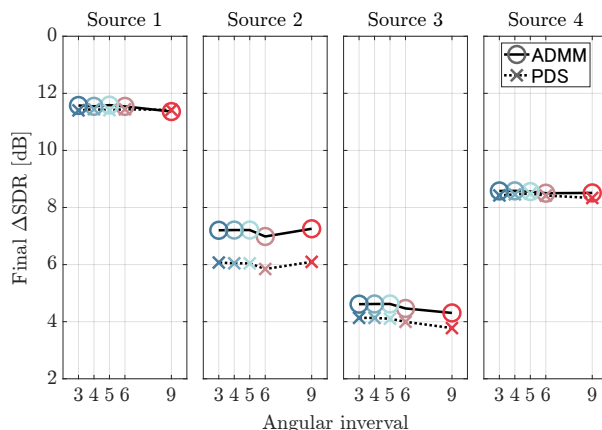


図-2 500 反復目における各音源の分離結果

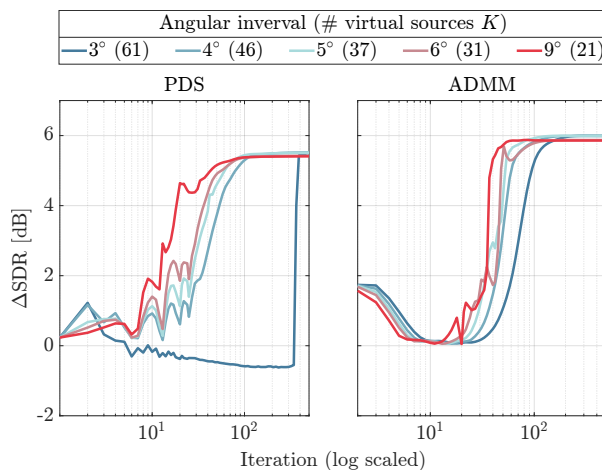


図-3 反復ごとの分離性能 (全音源の平均値)

リズムを提案した。実験の結果、従来法が用いていた PDS アルゴリズムよりも安定して最適化でき、高い性能が得られることがわかった。今後は、ADMM に対する加速法の導入や、スパース音源モデルの改善に取り組むつもりである。

参考文献

- [1] T. Tachikawa, K. Yatabe, and Y. Oikawa, “Underdetermined source separation with simultaneous DOA estimation without initial value dependency,” Int. Workshop Acoust. Signal Enhanc. (IWAENC), pp. 161–165, 2018.
- [2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Found. Trends Mach. Learn., **3**(1), 1–122 (2011).