最小燃料制御問題に対する主-双対分割法 *

秋田県立大学大学院 システム科学研究科　徳永 友貴
秋田県立大学 システム科学技術学部　松下 慎也
秋田県立大学 システム科学技術学部　徐 粒
Graduate School of Systems Science and Technology
Akita Prefectural University, Tomoki Tokunaga,
Faculty of Systems Science and Technology
Akita Prefectural University, Shin-ya Matsushita,
Faculty of Systems Science and Technology
Akita Prefectural University, Li Xu

1　背景

ばね-質量-ダンパ系や電気回路などの入出力の関係を表す工学の問題は状態方程式で表現できる:

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + \boldsymbol{b}u(t), \quad (t \geq 0) \tag{1}$$

制御工学においては，式（1）と所望する条件を同時に満たす制御入力 $u(t)$ の中からなるべく 0 の値を多くもつ解（最適解）を見つける問題（最小燃料制御問題）が古くから研究されている. この最小燃料制御問題の最適解を求める手法として交互方向乗数法（Alternating Direction Method of Multipliers:ADMM）[4] が知られており，先行研究ではこの手法が用いられている [2, 7]. しかし，ADMM には「適用できる形が決まっている」「繰り返しごとに逆行列の計算をする必要があるため処理に時間がかかる」といった課題が存在する. 一方で，主-双対分割法 [3] と呼ばれる凸最適化問題に対する ADMM とは異なる手法が盛んに研究されている．しかし、主-双対分割法を最小燃料問題に適用した例はこれまでになく、ADMM との比較は行われていなかった,

2　目的

本研究では，最小燃料制御問題に対して主-双対分割法を適用し，数値実験を行うことによってその有効性を検証する. 具体的には宇宙空間におけるロケットの飛行モデルに対して主-双対分割法，及び ADMM を適用し，両者の実行時間や解の精度を比較することで有効性の検証を行う.

* This work was supported by the Research Institute for Mathematical Sciences, an International Joint Usage/Research Center located in Kyoto University.

3　準備

以下の形で表される式を状態方程式と呼ぶ:

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + \boldsymbol{b}u(t), \quad t \geq 0 \tag{1}$$

上記の式において，$A \in \mathbb{R}^{d\times d}, \boldsymbol{b} \in \mathbb{R}^d$ はそれぞれ対象のモデルごとにあらかじめ与えられている定数行列, 及び定数ベクトルである．また，$\boldsymbol{x}(t) \in \mathbb{R}^d$, $u(t) \in \mathbb{R}$ はそれぞれ時刻 t での状態, 及び制御入力を表しており，$\boldsymbol{x}(0) = \boldsymbol{\xi} \in \mathbb{R}^d$ はモデルの初期状態を示している．

次に，一般的な凸最適化問題を以下に示す．

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) + g(\boldsymbol{x}) + \sum_{i=1}^{J} h_i(L_i\boldsymbol{x}) \tag{2}$$

$f : \mathbb{R}^n \to \mathbb{R}$ は微分可能かつ勾配 ∇f が β-リプシッツ連続[*1]であることを仮定し，$g : \mathbb{R}^n \to \mathbb{R}, h_i : \mathbb{R}^m \to \mathbb{R}$ は凸関数，$L_i \in \mathbb{R}^{m\times n}$ は行列とする．また，凸関数 h_i はモデルの制約や条件に合わせて任意の数だけ増やすことができる．

C を $\mathbb{R}^n$ の集合とする．本研究で取り扱う最小燃料制御問題を構成している l^1 ノルムと指示関数 I_C の定義を以下に示す．

$$\|\boldsymbol{x}\|_{l_1} := \sum_{i=1}^{n} |x_i|, \quad I_C(\boldsymbol{u}) := \begin{cases} 0, & (\boldsymbol{u} \in C) \\ \infty, & (\boldsymbol{u} \notin C) \end{cases} \tag{3}$$

続いて，一般的な凸最適化問題に対する主-双対分割法のアルゴリズムを以下に示す．

Algorithm　主-双対分割法 [3]

---

input : $\boldsymbol{x}^{(0)}$, $\boldsymbol{y}_1^{(0)}$, $\cdots$, $\boldsymbol{y}_J^{(0)}$　　(初期点)

τ , γ_1 , $\cdots$, γ_J >0　　(パラメータ)

$$\begin{cases} \boldsymbol{x}^{(k+1)} = \mathrm{prox}_{\tau g}(\boldsymbol{x}^{(k)} - \tau(\nabla f(\boldsymbol{x}^{(k)}) + L_1^\top \boldsymbol{y}_1^{(k)} + , \cdots , + L_J^\top \boldsymbol{y}_J^{(k)})) \\ \boldsymbol{y}_1^{(k+1)} = (I - \mathrm{prox}_{\gamma_1 h_1})(\boldsymbol{y}_1^{(k)} + \gamma_1 L_1(2\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})) \\ \qquad\qquad \vdots \\ \boldsymbol{y}_J^{(k+1)} = (I - \mathrm{prox}_{\gamma_J h_J})(\boldsymbol{y}_J^{(k)} + \gamma_J L_J(2\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})) \end{cases}$$

このとき，任意の $\gamma > 0$ に対して, 凸関数 $f : \mathbb{R}^n \to \mathbb{R}$ の近接写像 $\mathrm{prox}_{\gamma f}(\boldsymbol{x})$ は以下で定義される．

$$\mathrm{prox}_{\gamma f}(\boldsymbol{x}) := \mathrm{argmin}_{\boldsymbol{y}\in\mathbb{R}^N}\{f(\boldsymbol{y}) + \frac{1}{2\gamma}\|\boldsymbol{x} - \boldsymbol{y}\|^2\} \tag{4}$$

また，以下の条件式を満たすとき生成される点列 $\{\boldsymbol{x}^{(k)}\}$ は問題 (2) の解に収束することが知られている.[5]

$$\tau(\frac{\beta}{2} + \sum_{i=1}^{J} \gamma_i \|L_i^\top L_i\|) \quad < \quad 1 \tag{5}$$

ただし，$\|L_i^\top L_i\|$ は行列ノルムであり，行列 $L_i^\top L_i$ の最大固有値を表している．

[*1] 関数 f がリプシッツ連続であるとは，実定数 $\beta > 0$ が存在し $\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\|_2 \leq \beta\|\boldsymbol{x} - \boldsymbol{y}\|_2$ を満たす時を言う．

4　最小燃料制御問題

はじめに文献 [2] で紹介されている宇宙空間を飛行するロケットのモデルについて考える．宇宙空間上を想定しているため，摩擦や重力などの外力は考慮していない．また，ロケットに関する条件は次の表に示す．

表 1　ロケットに関する諸条件

質量	m
ロケットに働く力	$F(t)$
時刻 $t(\geq 0)$ の位置	$r(t)$
初期位置	$r(0) = \xi_1$
初期速度	$v(0) = \dot{r}(0) = \xi_2$

このとき，ニュートンの法則より，以下の式を得る．

$$m\ddot{r}(t) = F(t) \tag{6}$$

続いて，状態 $\boldsymbol{x}(t)$[*2]を次の式で定義する:

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} := \begin{bmatrix} r(t) \\ \dot{r}(t) \end{bmatrix} \tag{7}$$

式 (7) を両辺を t で微分すると，

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} \dot{r}(t) \\ \ddot{r}(t) \end{bmatrix}$$

となる．ここで，$\dot{r}(t) = x_2(t)$，$\ddot{r}(t) = m^{-1}F(t)$　$(\because$ 式 (6)$)$ より，

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} x_2(t) \\ m^{-1}F(t) \end{bmatrix}$$

となる．行列とベクトルを用いることで，宇宙空間を飛行するロケットは以下の状態方程式で表現できる．

$$\underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix}}_{\dot{\boldsymbol{x}}(t)} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{\boldsymbol{x}(t)} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\boldsymbol{b}} u(t) \tag{8}$$

状態方程式（8）は可制御であることが知られており，初期状態 $\boldsymbol{\xi} \in \mathbb{R}^d$ と終端時刻 $T > 0$ が与えられている．ここで，可制御であるとは，任意の初期状態 $\boldsymbol{x}(0) = \boldsymbol{\xi} \in \mathbb{R}^d$ に対して，ある終端時刻 $T > 0$ と制御入力 $u(t)$ $(0 \leq t \leq T)$ が存在して，この $u(t)$ によって状態 $\boldsymbol{x}(t)$ が $t = T$ で $\boldsymbol{x}(T) = \boldsymbol{0}$ となるときをいう．
このとき，$\boldsymbol{x}(0) = \boldsymbol{\xi}$, $\boldsymbol{x}(T) = \boldsymbol{0}$, $|u(t)| \leq 1$, $\forall t \in [0, T]$ を満たし，かつ今回の目的関数である L^1 ノルム

$$\|u\|_1 = \int_0^T |u(t)|dt \tag{9}$$

を最小化する $u(t)$ を見つける問題を最小燃料制御問題（L^1 最適制御問題） と呼ぶ．

[*2] $x_1(t)$：時刻 t における位置 , $x_2(t)$：時刻 t における速度, 初期状態 $x(0) = \boldsymbol{\xi} = (\xi_1, \xi_2)^\top$

前頁の最小燃料制御問題で見つかる制御入力 $u(t)$ は関数であるため一般に解を直接求めることが困難である．文献 [2] では，時間軸を離散化することでベクトルの解を見つける凸最適化問題として定式化している．

$$\min_{\boldsymbol{u}\in\mathbb{R}^n} \|\boldsymbol{u}\|_{l_1} \quad s.t. \quad \Phi\boldsymbol{u}=\boldsymbol{\zeta}, \|\boldsymbol{u}\|_\infty \leq 1 \tag{10}$$

上記の式で使用されている Φ, ξ, 及びそれに付随する $A_d, \boldsymbol{b}_d$ は文献 [2] を参考に設定した．

$$\Phi := (A_d^{n-1}\boldsymbol{b}_d, A_d^{n-2}\boldsymbol{b}_d, \cdots, \boldsymbol{b}_d), \qquad \boldsymbol{\zeta} := -A_d\boldsymbol{\xi}$$

$$A_d := e^{Ah}, \qquad \boldsymbol{b}_d := \int_0^h e^{Ah}\boldsymbol{b}dt$$

式 (10) は集合 C_1, C_2 を次のように設定し，指示関数 I_C(式 (3)) を用いることで次の問題へと定式化できる．

$$C_1 := \{\boldsymbol{u}\in\mathbb{R}^n : \|\boldsymbol{u}\|_\infty \leq 1\} \qquad C_2 := \{\boldsymbol{\zeta}\}$$

$$\min_{\boldsymbol{u}\in\mathbb{R}^n} \|\boldsymbol{u}\|_{l_1} + I_{C_1} + I_{C_2} \tag{11}$$

本研究では問題 (11) に対する主-双対分割法について検証する．

5 主-双対分割法

l^1 最適制御問題 (11) は問題 (2) の具体例となっている．

$$\min_{\boldsymbol{u}\in\mathbb{R}^n} \|\boldsymbol{u}\|_{l_1} + I_{C_1} + I_{C_2} \tag{11}$$

ここで，$f=0, \quad g=\|\cdot\|, \quad h_1=I_{C_1}, \quad h_2=I_{C_2}, \quad L_1=I, \quad L_2=\Phi$ である．これより，l^1 最適制御問題に対する主-双対分割法のアルゴリズムは以下のようになる．

Algorithm　主-双対分割法 [3]

---

input : $\boldsymbol{x}^{(0)}$, $\boldsymbol{y}_1^{(0)}$, $\boldsymbol{y}_2^{(0)}$　(初期点)

τ , γ_1 , γ_2 >0　(パラメータ)

$$\begin{cases} \boldsymbol{x}^{(k+1)} = S_\tau(\boldsymbol{x}^{(k)} - \tau(\boldsymbol{y}_1^{(k)} + \Phi^\top \boldsymbol{y}_2^{(k)})) \\ \boldsymbol{y}_1^{(k+1)} = (I - \Pi_{C_1})(\boldsymbol{y}_1^{(k)} + \gamma_1(2\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})) \\ \boldsymbol{y}_2^{(k+1)} = (I - \Pi_{C_2})(\boldsymbol{y}_2^{(k)} + \gamma_2\Phi(2\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)})) \end{cases}$$

ここで，問題 (11) で設定した関数の近接写像 $\mathrm{prox}_{\tau g}(\boldsymbol{x})$, $\mathrm{prox}_{\gamma h_1}(\boldsymbol{x})$, $\mathrm{prox}_{\gamma h_2}(\boldsymbol{x})$ は以下のようになることが知られている．

$$[\mathrm{prox}_{\tau g}(\boldsymbol{x})]_i := [S_\tau(\boldsymbol{x})]_i = \begin{cases} x_i - \tau, & (x_i \geq \tau) \\ 0, & (|x_i| < \tau) \\ x_i + \tau, & (x_i \leq -\tau) \end{cases}$$

$$\mathrm{prox}_{\gamma h_1}(\boldsymbol{x}) := \Pi_{C_1}(\boldsymbol{x}) = \begin{bmatrix} \mathrm{sgn}(x_1)\min\{|x_1|, 1\} \\ \mathrm{sgn}(x_2)\min\{|x_2|, 1\} \\ \vdots \\ \mathrm{sgn}(x_k)\min\{|x_k|, 1\} \end{bmatrix}$$

$$\mathrm{prox}_{\gamma h_2}(\boldsymbol{x}) := \Pi_{C_2}(\boldsymbol{x}) = \quad \boldsymbol{\zeta}$$

以下の条件式を満たすとき生成される点列 $\{\boldsymbol{x}^{(k)}\}$ は問題 (11) の解に収束することが知られている.

$$\tau(\gamma_1 + \gamma_2\|\Phi^\top\Phi\|) \quad < \quad 1 \tag{12}$$

6　数値実験

宇宙空間におけるロケットの状態方程式 (8) に対して l^1 最適制御問題 (11) を構成し，$T = 5, n = 1000$ として数値実験を行った．また，実験環境は以下の通りである．

表 2　実験環境

OS	Windows 10
CPU	Intel(R) Core(TM) i5-9400
メモリ	8.00 GB
プログラム言語	MATLAB(R2022a)

l^1 最適制御問題に対して主-双対分割法・ADMM をそれぞれ適用し，得られた解の精度・実行時間を比較した．このとき，アルゴリズムの初期点をランダムに 10 回与え，得られた結果の平均を表示している．また，表 3 中の誤差に関しては MATLAB の CVX を用いて導出した解とそれぞれのアルゴリズムで導出した解 (繰り返し回数が 10^3 時点) の差を表示している．

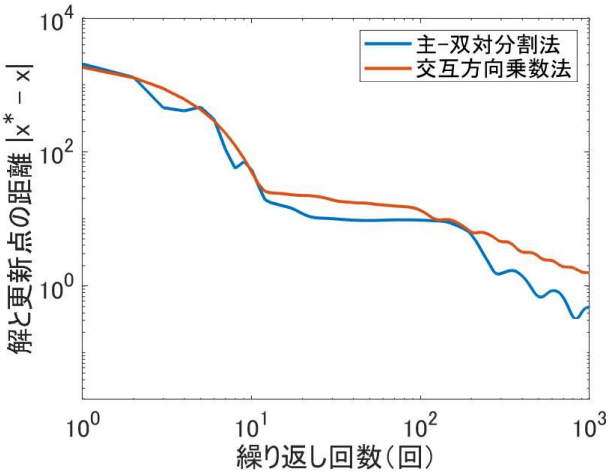

図 1　収束の様子

表 3　解の精度と実行時間の比較

	主-双対分割法	交互方向乗数法
実行時間 [s]	0.0209	2.5032
誤差	0.4958	1.5010

また，以下に求めた制御入力 (ベクトル) とそれを用いた時の状態 $\boldsymbol{x}$ の軌跡を示す.

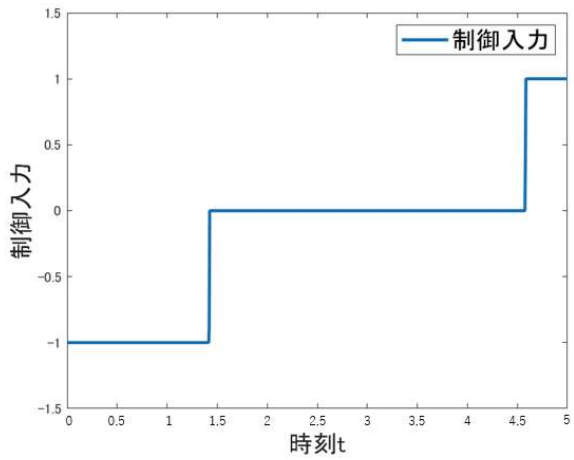

図 2　制御入力

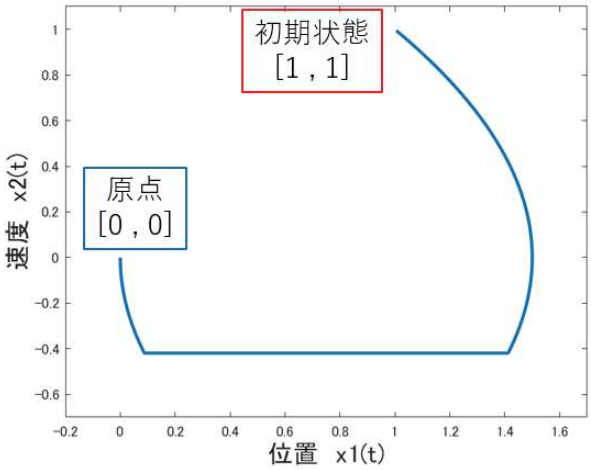

図 3　状態 $\boldsymbol{x}(t)$ の軌跡

図 (1) 表 (3) より，l^1 最適制御問題を対象とした場合，主双対分割法は ADMM と比較して実行時間は短くなり，精度は高くなった．これは ADMM は計算の過程で繰り返しごとに解を組み合わせて行列を構成しているのに対して主双対分割法はそのまま計算していることが影響しているのではないかと考えられる．また図 (2) より，要素の大半が 0 となっている制御入力を導出することができた．さらに図 (3) より，各アルゴリズムを用いて導出した制御入力を用いても終端時間 $T = 5$ 秒経過後に原点に遷移していることが確認できた.

7　結論

最小燃料制御問題に対して主-双対分割法を適用させることで ADMM を適用させた時より，解の精度・実行時間共に向上させることができた．一方，求めた制御入力は-1 から 0，0 から 1 へ瞬時に値を変化させる必要がありモータなどのアクチュエータを用いる場合は過度な負荷を与えてしまうことが懸念される．今後の課題としてはこの問題を解決するために解に連続性を持たせる L^2 ノルムを用いてスパース制を維持しつつ，制御入力の変化を緩やかに抑えることができる L^1/L^2 最適制御問題 [7] を検討する必要がある.

参考文献

[1] 小野峻佑，”近接分離アルゴリズムとその応用”，オペレーションズリサーチ，2019

[2] 永原正章，” スパースモデリング 基礎から動的システムへの応用”，コロナ社，2017

[3] A.Chambolle and T.Pock, “*A first-order primal-dual algorithm for convex problems with applications to imaging*”，Journal of Mathematical Imaging Vision，2010

[4] J. Eckstein and D. Bertsekas, “*On the Douglas-Rachford splitting method and proximal point algorithm for maximal monotone operators*”，Mathematical Programming，1992

[5] L.Condat，”A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms”，J.Optim.Theory Appl，2013

[6] M.Athans and P.L.Falb,” *Optimal Control*” ,Dover Publications, 2007

[7] M.Nagahara，”*Sparsity Methods for Systems and Control*”，Now Publishers，2020