

制御理論的スプラインのスパース表現

永原正章（京都大学） 山本裕（京都大学）

Sparse Representation for Control Theoretic Splines

*M. Nagahara (Kyoto University) and Y. Yamamoto (Kyoto University)

Abstract— In this article, we consider sparse representation for control signals which is transmitted through a rate-limited communication channel. The problem is formulated as ℓ^1 -penalized control theoretic splines. Then sparse solution is obtained by ℓ^1 - ℓ^2 optimization which can be effectively solved by iterative shrinkage method. Design examples are shown to illustrate the effectiveness of the proposed method.

Key Words: Sparse representation, control theoretic splines, ℓ^1 - ℓ^2 optimization

1 はじめに

圧縮センシングまたは圧縮サンプリングは、元のデータがスパースであるという仮定を置くことによって、非常に少ない観測データから元のデータをかなり正確に復元できる手法として、近年、とくに信号処理や情報理論の分野で盛んに研究されている^{2, 6)}。この手法の特徴は、スパースネスの指標であるベクトルの非ゼロ要素の数（しばしば“ ℓ^0 ノルム”と呼ばれる）の最小化を、その凸緩和である ℓ^1 ノルムの最小化に置き換えて逆問題を解く点にある。これは、最小二乗法の正則化項に ℓ^1 ノルムを用いる LASSO (Least Absolute Shrinkage and Selection Operator) と呼ばれるものとも関係がある⁶⁾。この最適解は、内点法を用いて計算することもできるが、近年、繰り返し縮小法 (iterative shrinkage) と呼ばれる手法が提案され、より高速に最適解を見つけることが可能となった^{1, 8)}。

本論文では、この手法にもとづき、制御入力信号のスパース表現を試みる。リモートコントロールでは、システムへの制御入力信号が帯域制限のある通信路を通してシステムに加えられる。このとき、この制御入力信号のスパース表現（すなわち、送信信号の多くが 0 であるような表現）が可能であれば、例えばエントロピー符号化⁴⁾の手法を用いて効率的な情報圧縮が可能となる。この性質にもとづき、システムの出力を ℓ^2 ノルムで最適化する制御入力の設計問題に設計パラメータの ℓ^1 ペナルティを含んだ枠組みでの最適化問題を定式化し、設計パラメータのスパース表現を得ることを提案する。具体的には、S. Sun ら⁵⁾によって提案された制御理論的スプラインの設計において、パラメータに対する ℓ^2 ペナルティのかわりに ℓ^1 ペナルティを導入し、パラメータのスパース表現を行う。この問題は、 ℓ^1/ℓ^2 最適化として定式化され、上記で述べた繰り返し縮小法により効率的に最適解を得ることができることを示す。また、設計例により、本手法の有効性を示す。

記法

ベクトル $v = [v_1, \dots, v_n]^T \in \mathbb{R}^n$ に対して、その ℓ^1 ノルムおよび ℓ^2 ノルムを

$$\|v\|_1 := \sum_{i=1}^n |v_i|, \quad \|v\|_2 := \left(\sum_{i=1}^n v_i^2 \right)^{1/2}$$

で定義する。また行列 Φ に対して、そのノルムを

$$\|\Phi\| := \max_{\|v\|_2 \neq 0} \frac{\|\Phi v\|_2}{\|v\|_2} = \sigma_{\max}(\Phi)$$

で表す。ここで $\sigma_{\max}(\Phi)$ は行列 Φ の最大特異値である。実数 $x \in \mathbb{R}$ に対して

$$\text{sgn}(x) := \begin{cases} 1, & \text{if } x \geq 0, \\ -1, & \text{if } x < 0, \end{cases}$$
$$(x)_+ := \max\{x, 0\}$$

と定義する。区間 $[0, T]$ ($T > 0$) 上の実数値 Lebesgue 2 乗可積分関数の全体を $L^2[0, T]$ と書く。 $f, g \in L^2[0, T]$ に対して、内積を

$$\langle f, g \rangle := \int_0^T f(t)g(t)dt$$

で表す。

2 問題設定

次の線形システム P を考える：

$$P : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t), \end{cases} \quad t \in [0, \infty).$$

ここで $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times 1}$ and $C \in \mathbb{R}^{1 \times n}$ とし、 $x(0) = 0$ とする。また、 (A, B) は可制御、 (C, A) は可観測とする。このシステムの出力信号 y に対して、次のデータ点を与える：

$$\mathcal{D} := \{(t_1, \alpha_1), (t_2, \alpha_1), \dots, (t_N, \alpha_N)\}.$$

ここで、各 t_i は時刻を表すデータで

$$0 = t_1 < t_2 < \dots < t_N =: T$$

とし、その各時刻において、システムの出力 $y(t_i)$ が α_i に近くなるように、すなわち $y(t_i) \approx \alpha_i$ となるように制御入力 u を設計したい。

この設計問題に対して、次の 2 ノルム評価関数を考える：

$$J_2(u) := \lambda \int_0^T u(t)^2 dt + \sum_{i=1}^N w_i (y(t_i) - \alpha_i)^2. \quad (1)$$

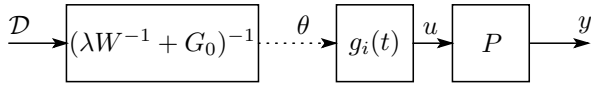


Fig. 1: Remote control system optimized with $J_2(u)$ in (1). The vector θ is transmitted through a communication channel.

ここで、 $w_i > 0$ は各データ点へのフィッティングに対する重みである．この評価関数 $J_2(u)$ を最小化する最適制御入力は、次式で表されることが知られており、制御理論的スプラインと呼ばれる^{5, 3)}:

$$u(t) = \sum_{i=1}^N \theta_i g_i(t). \quad (2)$$

関数 $g_i(t)$ は

$$g_i(t) := \begin{cases} C e^{A(t_i-t)} B, & \text{if } t_i > t, \\ 0, & \text{if } t_i \leq t \end{cases} \quad (3)$$

で定義され、係数 $\theta := [\theta_1, \dots, \theta_N]^\top$ は

$$\theta = (\lambda W^{-1} + G_0)^{-1} \alpha \quad (4)$$

で与えられる．ここで、 $\alpha := [\alpha_1, \dots, \alpha_N]^\top$ であり、また

$$G_0 := \begin{bmatrix} \langle g_{t_1}, g_{t_1} \rangle & \cdots & \langle g_{t_N}, g_{t_1} \rangle \\ \vdots & \ddots & \vdots \\ \langle g_{t_1}, g_{t_N} \rangle & \cdots & \langle g_{t_N}, g_{t_N} \rangle \end{bmatrix},$$

$$W := \text{diag}\{w_1, \dots, w_N\}$$

である．リモートコントロールの枠組みで考えると、この制御では、まずデータ \mathcal{D} が与えられたとき、(4) によって係数ベクトル θ が計算され、この θ が通信路に送信される．受信側では、(2) により制御入力信号 u が計算される．この状況では、通信路の容量に制約があるのが通常で、 θ のデータ量は少ない方が望ましい．したがって、圧縮センシングの考え方^{2, 6)} に従い、送信信号 θ のスパース表現を考える．まず、制御入力 u は、 $L^2[0, T]$ の部分空間

$$\mathcal{V}_M := \left\{ u \in L^2[0, T] : u = \sum_{i=1}^M \theta_i \psi_i, \theta_i \in \mathbb{R} \right\}. \quad (5)$$

に属する信号と仮定する ($M \geq N$ とする)．ここで、 $\{\psi_i\}$ は $L^2[0, T]$ の線形独立なベクトルとする．このとき、評価関数 $J_2(u)$ の第二項は次のように表現できる:

$$\begin{aligned} \sum_{i=1}^N w_i (y(t_i) - \alpha_i)^2 \\ = \theta^\top G^\top W G \theta - 2\theta^\top G^\top W \alpha + \alpha^\top W \alpha \\ = \|\Phi\theta - v\|_2^2. \end{aligned}$$

ただし、

$$G := \begin{bmatrix} \langle g_1, \psi_1 \rangle & \cdots & \langle g_N, \psi_1 \rangle \\ \vdots & \ddots & \vdots \\ \langle g_1, \psi_M \rangle & \cdots & \langle g_N, \psi_M \rangle \end{bmatrix}$$

とし、 $\Phi := W^{1/2} G$ 、 $v := W^{1/2} \alpha$ とする．これより、係数ベクトル θ に ℓ^1 のペナルティを与えた次の評価関数を導入する:

$$J_1(\theta) := \lambda \|\theta\|_1 + \|\Phi\theta - v\|_2^2. \quad (6)$$

上で述べたように、第一項の ℓ^1 ノルムはベクトル θ のスパースネスを表す指標である．この評価関数を最小化する最適解を θ^* とすると、最適制御入力は

$$u^*(t) = \sum_{i=1}^N \theta_i^* \psi_i(t), \quad t \in [0, T]$$

で与えられる．

3 ℓ^1 - ℓ^2 最適化によるスパース表現

(6) の評価関数は θ に関して凸であり、最適解は必ず存在する．しかし、その最適解は (4) のように解析的には表すことができず、最適解を求めるためには繰り返し計算が必要である．この繰り返し計算には、内点法などが従来用いられてきたが、近年、繰り返し縮小法 (iterative shrinkage)^{1, 8)} と呼ばれる手法が提案され、これにより高速な数値計算が可能となった．

繰り返し縮小法のアルゴリズムは以下で与えられる^{1, 8)}: 初期値 $\theta[0] \in \mathbb{R}^N$ を与え、 $\beta[1] = 1$ 、 $\eta[1] = \theta[0]$ とする．定数 c を

$$c > \|\Phi\| = \sigma_{\max}(\Phi)$$

となるように選ぶ．次を繰り返す¹⁾:

$$\begin{aligned} \theta[k] &= S_{\lambda/c} \left(\frac{1}{c} \Phi^\top (v - \Phi\eta[k]) + \eta[k] \right) \\ \beta[k+1] &= \frac{1 + \sqrt{1 + 4\beta[k]^2}}{2} \\ \eta[k+1] &= \theta[k] + \frac{\beta[k] - 1}{\beta[k+1]} (\theta[k] - \theta[k-1]) \\ k &= 1, 2, \dots \end{aligned}$$

関数 $S_{\lambda/c}$ は $\eta = [\eta_1, \dots, \eta_N]^\top$ に対して

$$S_{\lambda/c}(\eta) := \begin{bmatrix} \text{sgn}(\eta_1)(|\eta_1| - \lambda/c)_+ \\ \vdots \\ \text{sgn}(\eta_N)(|\eta_N| - \lambda/c)_+ \end{bmatrix}$$

で定義される非線形関数である．関数 $S_{\lambda/c}$ の各要素に対する非線形関数 $\text{sgn}(\eta)(|\eta| - \lambda/c)$ の入出力関係を Fig. 2 に示す．条件 $c > \|\Phi\|$ が成り立つような $c > 0$ を選べば、上のアルゴリズムは任意の Φ と v に対して収束し、 $\lim_{k \rightarrow \infty} \theta[k]$ は (6) を最小化する最適解に一致する¹⁾．さらに、最適解への収束の早さは最悪評価で $O(1/k^2)$ であることも知られている¹⁾．上記のアルゴリズムは、計算機に容易に実装が可能であり、収束も早いことから、リアルタイムでスパースなベクトル θ を得ることが可能である．

¹⁾繰り返し縮小法にも様々な方法が提案されているが⁸⁾、ここで用いた方法は FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) と呼ばれる方法である¹⁾．

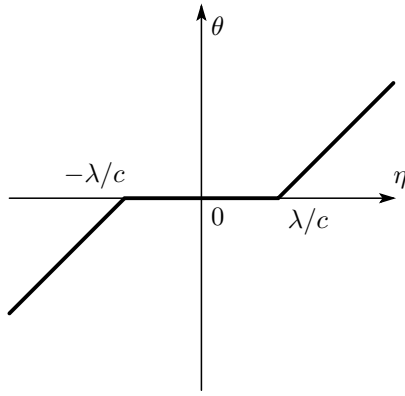


Fig. 2: Nonlinear function $\theta = \text{sgn}(\eta)(|\eta| - \lambda/c)_+$

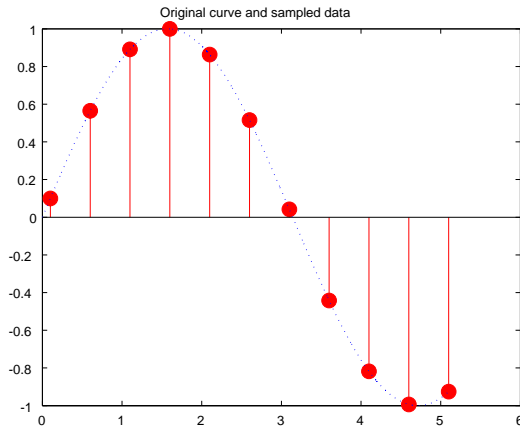


Fig. 3: Original curve $\sin t$ and sampled data D .

4 設計例

線形システム P の実現を

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad C = [1 \quad 0 \quad 0].$$

とする。すなわち、 $P(s) = 1/s^3$ である。データ D のサンプル時刻を $t_i = 0.1 + (i-1) \times 0.5, i = 1, 2, \dots, 11$ とし、データ α_i を正弦波 $\alpha_i = \sin t_i$ とする (図 3 を参照)。また、 $L^2[0, T]$ の線形独立なベクトル $\psi_i(t)$ は、上記の A, B, C および $\{t_1, \dots, t_{11}\}$ を使って (3) によって定義される $g_i(t)$ と同じものとする²。また、(1) に対する λ を 0.1, (6) に対する λ を 0.02 とする。これらの設定のもとで、(2) で計算される、(1) の ℓ^2 評価関数を最適化する θ を θ_2^* とおき、また (6) の ℓ^1 - ℓ^2 評価関数を最適化する θ を θ_1^* とおくと、それぞれ以

² (A, B) が可制御かつ (C, A) が可観測ならば、(3) で定義される $\{g_i\}$ は線形独立となることが知られている³⁾。

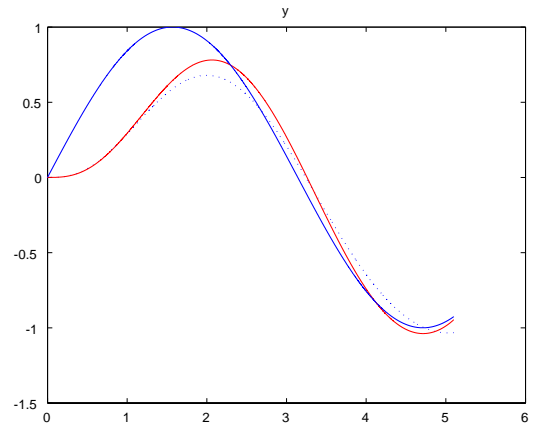


Fig. 4: Original curve $\sin t$: (blue line), Output $y(t)$ of ℓ^1 - ℓ^2 optimal input with $\lambda = 0.02$ (red line), ℓ^2 optimal input with $\lambda = 0.1$ (blue dots).

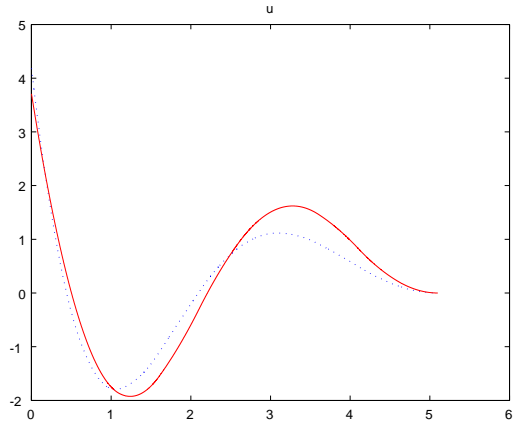


Fig. 5: ℓ^1 - ℓ^2 optimal input $u(t)$ with $\lambda = 0.02$ (red line), ℓ^2 optimal input $u(t)$ with $\lambda = 0.1$ (blue dots).

下で与えられる。

$$\theta_2^* = \begin{bmatrix} 0.9919 \\ 4.7303 \\ 5.4459 \\ 3.9878 \\ 1.8978 \\ 0.1898 \\ -0.8228 \\ -1.2059 \\ -1.0451 \\ -0.3015 \\ 1.0642 \end{bmatrix}, \quad \theta_1^* = \begin{bmatrix} 0 \\ 0 \\ 2.0665 \\ 3.8879 \\ 3.6590 \\ 1.1288 \\ 0 \\ -1.7358 \\ -2.6657 \\ -0.3231 \\ 1.7432 \end{bmatrix}$$

ℓ^1 - ℓ^2 最適化によりスパースな解 θ_1 が得られることがわかる。これらのパラメータに対するシステムの出力を Fig. 4 に、入力 $u(t)$ を Fig. 5 に示す。また復元誤差 $e(t) = |\sin t - y(t)|$ を Fig. 6 に示す。 ℓ^1 - ℓ^2 による最適化によるスパース表現にもかかわらず、 ℓ^2 最適化とほぼ同じ性能を示していることがわかる。

さらに、(6) の ℓ^1 - ℓ^2 評価関数の λ を 0.05 にした場

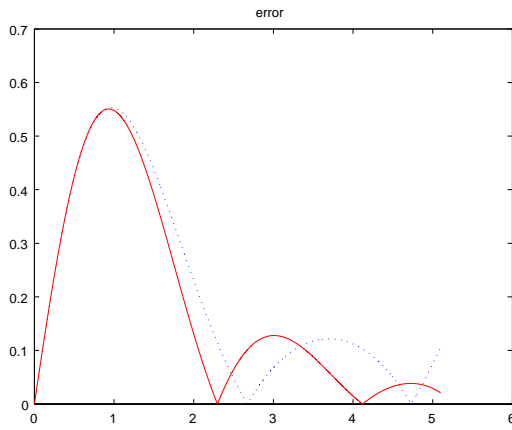


Fig. 6: Absolute error with θ_1^* (solid) and θ_2^* (dots).

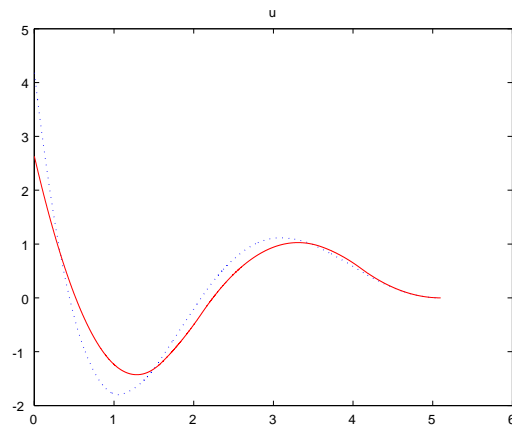


Fig. 7: ℓ^1 - ℓ^2 optimal input $u(t)$ with $\lambda = 0.05$ (red line), ℓ^2 optimal input $u(t)$ with $\lambda = 0.1$ (blue dots).

合の最適パラメータ θ_1^* は以下ようになる .

$$\theta_1^* = \begin{bmatrix} 0 \\ 0 \\ 0.3672 \\ 3.2113 \\ 3.2282 \\ 0 \\ 0 \\ -0.6546 \\ -2.2377 \\ 0 \\ 1.0940 \end{bmatrix} .$$

定数 λ の値を大きくしたことで、よりスパースな表現が得られていることがわかる . この θ_1^* を用いた入力 $u(t)$, および復元誤差をそれぞれ Fig. 7 , Fig. 6 に示す . 入力 $u(t)$ は $\lambda = 0.02$ の場合に比べて小さくなっていることがわかる . これは、 ℓ^1 ノルムのペナルティをより大きくとったことに対応している . さらに、このペナルティを大きくすることにより、復元誤差が上昇していることがわかる . この例題から、入力信号のパラメータ θ のスパースネスと復元誤差 (性能) との間にはトレードオフの関係があることがわかる .

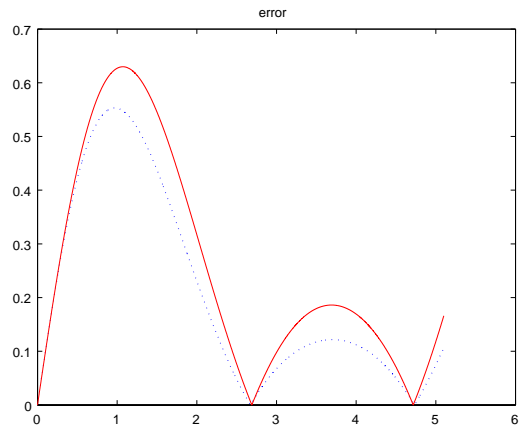


Fig. 8: Absolute error with θ_1^* (solid) and θ_2^* (dots).

5 おわりに

本稿では、リモートコントロールの枠組みで、送信信号 (最適制御入力を定義するパラメータ θ) のデータ量を抑えるために、 ℓ^1 - ℓ^2 最適化の枠組みで問題を定式化し、繰り返し縮小法により効率的に最適解を求めることができることを示した . また設計例を示し、性能をそれほど下げずにある程度のスパース表現が可能なこと、またパラメータのスパースネスと制御性能の間にトレードオフの関係があることを示した .

参考文献

- 1) A. Beck and M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sciences*, vol. 2, No. 1, 183/202 (2009)
- 2) E. J. Cnades and M. B. Wakin, An introduction to compressive sampling, *IEEE Signal Processing Magazine*, vo. 25, 21/30 (2008)
- 3) M. Egerstedt and C.F. Martin, *Control Theoretic Splines*, Princeton University Press (2010)
- 4) G. A. Jones and M. Jones, *Information and Coding Theory*, Springer (2000)
- 5) S. Sun, M. B. Egerstedt, and C. F. Martin, Control theoretic smoothing splines, *IEEE Trans. on Automatics Control*, Vol. 45, No. 12, 2271/2279 (2000)
- 6) 田中, 圧縮センシングの数理, *IEICE Fundamental Review*, Vol. 4, No. 1, 39/47 (2010)
- 7) K. L. Teo and X. Q. Yang, Computational discretization algorithms for functional inequality constrained optimization, *Annals of Operations Research*, Vol. 98, 215/234 (2000)
- 8) M. Zibulevsky and M. Elad, L1-L2 optimization in signal and image processing, *IEEE Signal Processing Magazine*, Vol. 27, 76/88 (2010)