

# 予測にもとづくネットワーク化制御系の安定化

永原 正章<sup>†</sup>

<sup>†</sup> 京都大学情報学研究科 〒606-8501 京都市左京区吉田本町

あらまし 本稿では、ネットワーク化制御におけるパケットロスに対してロバストな制御手法であるパケット化予測制御を紹介し、その枠組みにもとづき、通信帯域の制約に対処するために、制御ベクトルのスパース表現を提案する。フィードバック系を漸近安定にする制御ベクトルを特徴付け、そのなかで最もスパースなベクトルを求める  $\ell^0$  最適化として設計問題を定式化する。この問題はそのままでは組み合わせ最適化であり、その近似解を求めるための高速解法として OMP (Orthogonal Matching Pursuit) 法、および  $\ell^1/\ell^2$  最適化による緩和とその高速解法である FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) を導入する。数値例により従来法である  $\ell^2$  最適化と比較した提案手法の有効性を示し、さらに OMP 法および  $\ell^1/\ell^2$  最適化の利点と欠点を明らかにする。

キーワード ネットワーク化制御、モデル予測制御、スパース表現

## Stabilization of networked control systems based on prediction

Masaaki NAGAHARA<sup>†</sup>

<sup>†</sup> Graduate School of Informatics, Kyoto University Sakyo-ku, Yoshida-Honmachi, Kyoto, 606-8501 Japan

**Abstract** In this article, we introduce packetized predictive control that is known to be robust against packet dropouts in networked control, and based on the control we propose sparse representation of control vectors to overcome rate-limitedness of networks. We first characterize the control vectors that asymptotically stabilize the feedback control system, and then formulate the control problem as an  $\ell^0$  optimization to obtain the sparsest vector among the stabilizing control vectors. This problem is combinatorial optimization that is difficult to solve, and hence we adopt the OMP (Orthogonal Matching Pursuit) method and relaxed  $\ell^1/\ell^2$  optimization with FISTA (Fast Iterative Shrinkage-Thresholding Algorithm). Examples show the effectiveness of the proposed methods compared with the conventional  $\ell^2$  optimization method, and make clear advantages and disadvantages of the two proposed methods.

**Key words** Networked control, model predictive control, sparse representation

### 1. はじめに

ネットワーク化制御系 [2], [4], [10] とは、無線通信などのネットワークを介して制御対象を制御するシステムのことであり、現実のネットワークにはパケットロスや帯域の制限などがあり、したがってネットワーク化制御系では、それらによる安定性や制御性能への影響を考慮する必要がある。本稿では、パケットロスに対してロバストな制御手法であるパケット化予測制御 [9] を紹介し、その枠組みにもとづき、通信帯域の制約に対処するための新しい安定化手法を提案する。

パケット化予測制御では、モデル予測制御 [5] において計算される未来の制御入力信号系列をパケット化して送信する。送信が成功すれば、それを受信側のバッファで保存する。保存された制御入力系列は、以降、パケットロスが生じたときに使用される。この方式により、パケットロスに対してロバストなネッ

トワーク制御系が得られる。

送信する制御入力系列は長ければ長いほど、より多くのパケットロスに対応できるが、ネットワークは帯域制限されており送信できるビットレートは限られるので、あまり長くはできない。そこで本稿では、圧縮センシング等の研究で用いられているスパース表現のテクニック [3] を導入し、送信パケットのスパース表現を試みる。この枠組みでのスパースな送信パケットは、OMP (Orthogonal Matching Pursuit) 法 [6] または  $\ell^1/\ell^2$  最適化 [11] により効率的に計算可能であることを示す。また数値例により、提案手法の有効性を示す。

### 記法

本稿では、以下の記法を用いる。 $\mathbb{N}_0$  で非負整数の集合を表す。すなわち、 $\mathbb{N}_0 := \{0, 1, 2, \dots\}$  である。 $\mathbb{R}^n$  は  $n$  次元実ベクトル空間であり、 $n = 1$  のときは  $\mathbb{R}$  と略記する。ベクトル  $x \in \mathbb{R}^n$  と正定値対称行列  $P > 0$  に対して、 $\|x\|_P := \sqrt{x^\top P x}$  と定義

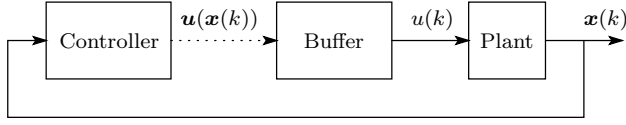


図1 Networked Control System with PPC. The dotted line indicates an erasure channel.

する．ただし， $\top$  は転置を表す．特に  $P = I$  (単位行列) のときは  $\|\mathbf{x}\|_I := \|\mathbf{x}\|_2$  と表す．対称行列  $A$  に対して，最小および最大固有値をそれぞれ， $\lambda_{\min}(A)$ ， $\lambda_{\max}(A)$  と表す．ベクトル  $\mathbf{x} = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$  に対して， $\text{supp}\{\mathbf{x}\}$  を  $\mathbf{x}$  の非ゼロ要素の添え字の集合とする．すなわち， $\text{supp}\{\mathbf{x}\} := \{i : x_i \neq 0\}$  と定義する．

## 2. パケット化予測制御

本稿では，制御対象として以下の線形時不変系を考える．

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + Bu(k), \quad k \in \mathbb{N}_0, \quad \mathbf{x}(0) = \mathbf{x}_0, \quad (1)$$

ここで  $\mathbf{x}(k) \in \mathbb{R}^n$ ， $u(k) \in \mathbb{R}$  とし， $(A, B)$  は可到達であると仮定する．この制御対象に対して，パケットロスが発生する信頼性の低いネットワークを介して制御を行うことを考える．

パケット化予測制御 [9] とは，時刻  $k \in \mathbb{N}_0$  での (1) の状態  $\mathbf{x}(k)$  の観測から， $N$  ステップ未来までの制御入力を計算する．通常モデル予測制御 [5] では，計算された制御入力のうち，一番はじめのもののみを送信するが，パケット化予測制御では，すべての制御入力を次式のようにパケット化し送信する．

$$\mathbf{u}(\mathbf{x}(k)) \triangleq [u_0(\mathbf{x}(k)) \quad \dots \quad u_{N-1}(\mathbf{x}(k))]^\top \in \mathbb{R}^N \quad (2)$$

送信されたパケットは受信側でバッファに記憶される(図1を見よ)．もし時刻  $k+1$  で送信パケットがロスしたとすると，入力  $u(k+1)$  として，バッファに記憶された  $u_1(\mathbf{x}(k))$  を用いる．同様に時刻  $k+2$  でパケットロスが起これば， $u(k+2) = u_2(\mathbf{x}(k))$  とする．パケット化予測制御のより詳細な定式化は [9] を参照せよ．上記の操作は  $N$  回連続でパケットロスが起こるまで可能である．

[仮定1] 本稿では， $N+1$  回以上連続でパケットロスは起こらないと仮定する．

## 3. パケットロスのもとでの安定化

本節では，仮定1を満たす任意のパケットロスに対して，フィードバック系が漸近安定となるような制御ベクトルを特徴付ける．

状態ベクトル  $\mathbf{x} \in \mathbb{R}^n$  が与えられたとき，制御入力を  $\mathbf{u} = (u_0, u_1, \dots, u_{N-1})$  としたときのモデル (1) にもとづく状態の予測  $\{\mathbf{x}_i\}_{i=0}^{N-1}$  は以下のように与えられる．

$$\mathbf{x}_0 = \mathbf{x}, \quad \mathbf{x}_{i+1} = A\mathbf{x}_i + Bu_i, \quad i = 0, 1, \dots, N-1. \quad (3)$$

ここで，与えられた正定値対称行列  $P, Q, W$  に対して，次のような集合を定義する．

$$\mathcal{F}(\mathbf{x}) \triangleq \left\{ \mathbf{u} \in \mathbb{R}^N : \|\mathbf{x}_N\|_P^2 + \sum_{i=0}^{N-1} \|\mathbf{x}_i\|_Q^2 \leq \mathbf{x}^\top W \mathbf{x} \right\}.$$

この集合は， $N$  ステップまでの未来の状態の予測値が次の不等式

$$\|\mathbf{x}_N\|_P^2 + \sum_{i=0}^{N-1} \|\mathbf{x}_i\|_Q^2 \leq \mathbf{x}^\top W \mathbf{x}$$

を満たすような制御入力  $\mathbf{u} = (u_0, u_1, \dots, u_{N-1}) \in \mathbb{R}^N$  の集合である．

次の各行列を定義する．

$$\Phi \triangleq \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix} = \begin{bmatrix} \Phi_0 \\ \Phi_1 \\ \vdots \\ \Phi_{N-1} \end{bmatrix},$$

$$\Phi_i \triangleq [A^i B, A^{i-1} B, \dots, B, 0, \dots, 0], \quad i = 0, 1, \dots, N-1,$$

$$\Upsilon \triangleq \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad \bar{Q} \triangleq \text{blockdiag}\{Q, \dots, Q, P\},$$

$$G \triangleq \bar{Q}^{1/2} \Phi, \quad H \triangleq -\bar{Q}^{1/2} \Upsilon.$$

このとき，集合  $\mathcal{F}(\mathbf{x})$  は次のように表すことができる．

$$\mathcal{F}(\mathbf{x}) = \left\{ \mathbf{u} \in \mathbb{R}^N : \|\mathbf{G}\mathbf{u} - \mathbf{H}\mathbf{x}\|_2^2 \leq \mathbf{x}^\top W \mathbf{x} \right\}.$$

以上の準備のもとで，次の定理が成り立つ．

[定理1] 行列  $P > 0, Q > 0, W > 0$  を次の手順で与えるとする．

(1)  $Q > 0$  を任意に与える．

(2)  $P > 0$  を次の Riccati 方程式の解とする：

$$P = A^\top P A - \frac{A^\top P B B^\top P A}{B^\top P B} + Q. \quad (4)$$

(3) 次の各定数を求める．

$$c_1 \triangleq \max_{i=0, \dots, N-1} \lambda_{\max}\{\Phi_i^\top P \Phi_i (G^\top G)^{-1}\},$$

$$\rho \triangleq 1 - \lambda_{\min}(QP^{-1}),$$

$$c \triangleq \frac{(1 - \rho^N)c_1}{1 - \rho},$$

(4) 次の不等式を満たす行列  $\mathcal{E}$  を与える．

$$0 < \mathcal{E} < \frac{1 - \rho}{c} P.$$

(5) 行列  $W$  を  $W = P - Q + \mathcal{E}$  とする．

このとき，任意の  $\mathbf{x} \in \mathbb{R}^n$  に対して，集合  $\mathcal{F}(\mathbf{x})$  は少なくとも一つの要素を持ち，かつ各時刻  $k \in \mathbb{N}_0$  において，送信パケットを

$$\mathbf{u}(\mathbf{x}(k)) \in \mathcal{F}(\mathbf{x}(k))$$

を満たすように選べば，仮定1を満たす任意のパケットロスに対して，図1のネットワーク化制御系は漸近安定，すなわち，

---

**Algorithm 1** OMP for sparse control vector  $\mathbf{u}(\mathbf{x})$ 

---

**Require:**  $\mathbf{x} \in \mathbb{R}^n$  {observed state vector}**Ensure:**  $\mathbf{u}(\mathbf{x})$  {sparse control packet} $k := 0.$  $\mathbf{u}[0] := \mathbf{0}.$  $\mathbf{r}[0] := H\mathbf{x} - G\mathbf{u}[0] = H\mathbf{x}.$  $S[0] := \text{supp}\{\mathbf{x}[0]\} = \emptyset.$ **while**  $\|\mathbf{r}[k]\|_2^2 > \mathbf{x}^\top W\mathbf{x}$  **do****for**  $j = 1$  **to**  $N$  **do**

$$z_j := \frac{\mathbf{g}_j^\top \mathbf{r}[k]}{\|\mathbf{g}_j\|_2} = \arg \min_{z \in \mathbb{R}} \|\mathbf{g}_j z - \mathbf{r}[k]\|_2^2.$$

$$e_j := \|\mathbf{g}_j z_j - \mathbf{r}[k]\|_2^2.$$

**end for**Find a minimizer  $j_0 \notin S[k]$  such that  $e_{j_0} \leq e_j$ , for all  $j \notin S[k]$ . $S[k+1] := S[k] \cup \{j_0\}$ 

$$\mathbf{u}[k+1] := \arg \min_{\text{supp}\{\mathbf{u}\}=S[k+1]} \|G\mathbf{u} - H\mathbf{x}\|_2^2.$$

$$\mathbf{r}[k+1] := H\mathbf{x} - G\mathbf{u}[k+1].$$

 $k := k + 1.$ **end while****return**  $\mathbf{u}(\mathbf{x}) = \mathbf{u}[k].$ 

---

$$\lim_{k \rightarrow \infty} \mathbf{x}(k) = \mathbf{0}$$

を満たす.

証明: 省略. □

#### 4. スパースな制御パケットの設計

図 1 のパケット化予測制御において, ネットワークを介して送信される制御パケットがスパースであれば, パケットを符号化するとき, 密なパケットよりもより効率的なデータ圧縮が可能である. そこで, 時刻  $k \in \mathbb{N}_0$  での状態の観測値  $\mathbf{x}(k)$  にもとじて決定される制御パケット (2) をスパースにする, すなわち, パケット  $\mathbf{u}(\mathbf{x}(k))$  の要素  $u_i(\mathbf{x}(k))$  がなるべく 0 となるような制御を考える. 定理 1 の条件を満たす正定値対称行列  $P, Q, W$  をあらかじめ決めておき, 時刻  $k \in \mathbb{N}_0$  における送信パケット  $\mathbf{u}(\mathbf{x}(k))$  を  $\mathcal{F}(\mathbf{x}(k))$  の中から選べば, フィードバック系の漸近安定性が保証される. これにもとづき, 送信パケット  $\mathbf{u}(\mathbf{x}(k))$  として, 集合  $\mathcal{F}(\mathbf{x}(k))$  のなかで最もスパースなものを選ぶ. すなわち, 次の最適化問題を考える.

$$\mathbf{u}(\mathbf{x}(k)) = \arg \min_{\mathbf{u} \in \mathcal{F}(\mathbf{x}(k))} \|\mathbf{u}\|_0, \quad k \in \mathbb{N}_0. \quad (5)$$

この最適化問題は, 組み合わせ最適化であり, ベクトル  $\mathbf{u}$  のサイズ  $N$  が大きくなると, 指数関数的に計算量は増大する [8]. しかし, (5) の最適化は各時刻  $k \in \mathbb{N}_0$  においてオンラインで計算する必要があり, その計算時間はなるべく短いほうが望ましい. そこで, (5) の近似解を高速に求めることが可能な OMP (Orthogonal Matching Pursuit) 法 [6] を導入する. 状態ベクトル  $\mathbf{x}$  が得られたときに, スパースな送信パケット  $\mathbf{u}(\mathbf{x})$  を得るための OMP アルゴリズムを Algorithm 1 に示す.

スパースな送信パケットを得るもう一つの方法として, 次の  $\ell^1/\ell^2$  最適化がある:

---

**Algorithm 2** FISTA for sparse control vector  $\mathbf{u}(\mathbf{x})$ 

---

**Require:**  $\mathbf{x} \in \mathbb{R}^n$  {observed state vector}**Ensure:**  $\mathbf{u}(\mathbf{x})$  {sparse control packet} $k := 1.$  $\mathbf{u}[0] := \mathbf{0}.$  $\tilde{\mathbf{u}}[1] := \mathbf{0}.$  $\beta[1] := 0.$ **repeat**

$$\mathbf{u}[k] := \text{shrink}_{\nu_1} \left( \frac{1}{\kappa} G^\top (H\mathbf{x} - G\tilde{\mathbf{u}}[k]) + \tilde{\mathbf{u}}[k] \right).$$

$$\beta[k+1] := \frac{1}{2} + \sqrt{\frac{1}{4} + \beta[k]^2}.$$

$$\tilde{\mathbf{u}}[k+1] := \mathbf{u}[k] + \frac{\beta[k]-1}{\beta[k+1]} (\mathbf{u}[k] - \mathbf{u}[k-1]).$$

 $k := k + 1.$ **until**  $\|\mathbf{u}[k-1] - \mathbf{u}[k-2]\|_2 \leq \text{EPS}$  or  $k \geq \text{MAXITER}.$ **return**  $\mathbf{u}(\mathbf{x}) = \mathbf{u}[k-1].$ 

---

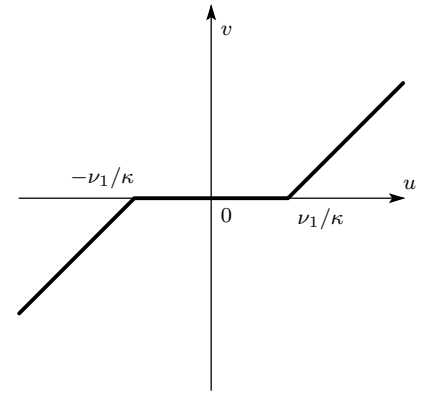


図 2 Single shrinkage function  $v = \text{sgn}(u)(|u| - \nu_1/\kappa)_+$  in  $\text{shrink}_{\nu_1}(\mathbf{u})$

$$\mathbf{u}(\mathbf{x}) = \arg \min_{\mathbf{u} \in \mathbb{R}^N} \nu(\mathbf{x}) \|\mathbf{u}\|_1 + \frac{1}{2} \|G\mathbf{u} - H\mathbf{x}\|_2^2.$$

ただし,  $\nu(\mathbf{x})$  は上記の最適解が集合  $\mathcal{F}(\mathbf{x})$  に入るように適切に決定する必要があるが, これを求めるのは困難であるので, 問題を緩和し, 定数値  $\nu(\mathbf{x}) = \nu_1$  を採用する. このとき, 定理 1 に示したような漸近安定性は失われるが<sup>(注1)</sup> この  $\ell^1/\ell^2$  最適化は FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) と呼ばれる高速アルゴリズムにより OMP 法よりもさらに効率的に解くことができる [1]. FISTA のアルゴリズムを Algorithm 2 に示す. ただし, Algorithm 2 中の  $\kappa$  は  $\kappa > \lambda_{\max}(G^\top G)$  を満たす定数であり, また関数  $\text{shrink}_{\nu_1}(\cdot)$  は, スカラー  $\nu_1 \in \mathbb{R}$  およびベクトル  $\mathbf{u} = [u_0, \dots, u_{N-1}]^\top$  に対して,

$$\text{shrink}_{\nu_1}(\mathbf{u}) \triangleq \begin{bmatrix} \text{sgn}(u_0)(|u_0| - \nu_1/\kappa)_+ \\ \vdots \\ \text{sgn}(u_{N-1})(|u_{N-1}| - \nu_1/\kappa)_+ \end{bmatrix},$$

で定義される. ここで,  $u \in \mathbb{R}$  に対して,  $\text{sgn}(u) := u/|u|$ ,  $(u)_+ := \max\{u, 0\}$  である (図 2 を参照せよ).

---

(注1): Practical 安定性, すなわち原点を含むある有界閉集合への収束は示すことができる [7].

## 5. 設 計 例

本節では提案法の有効性を確認するために、数値例題を示す。制御対象として、次の状態空間行列をもつ線形時不変システム (1) を考える<sup>(注2)</sup>：

$$A = \begin{bmatrix} -0.1242 & -0.1962 & -0.6071 & -1.1221 \\ -2.5415 & -0.3057 & -0.8284 & 0.0460 \\ 0.2772 & -1.1289 & 0.5358 & -1.2386 \\ -0.1960 & 0.1942 & 0.1095 & 0.6382 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.1242 \\ -2.5415 \\ 0.2772 \\ -0.1960 \end{bmatrix}.$$

状態予測の長さ、すなわち制御バケット (2) のサイズを  $N = 10$  とする。次に、正定値対称行列  $Q$  を単位行列、すなわち  $Q = I$  として、定理 1 の手順に従って、正定値対称行列  $Q, P, W$  を選ぶ。Riccati 方程式 (4) の解は

$$P = \begin{bmatrix} 1.0000 & 0 & 0 & 0 \\ 0 & 2.9856 & -0.6991 & 2.1910 \\ 0 & -0.6991 & 1.8548 & 0.1418 \\ 0 & 2.1910 & 0.1418 & 4.7892 \end{bmatrix},$$

で与えられ、また定数  $\rho$  と  $c$  は  $\rho = 0.8406$  および  $c = 5.1685$  となった。これらの定数を用いて、

$$\varepsilon = \frac{2}{3} \cdot \frac{1-\rho}{c} \cdot P < \frac{1-\rho}{c} \cdot P$$

とし、行列  $W = P - Q + \varepsilon$  を計算すると

$$W = \begin{bmatrix} 0.0206 & 0 & 0 & 0 \\ 0 & 2.0470 & -0.7135 & 2.2361 \\ 0 & -0.7135 & 0.8930 & 0.1447 \\ 0 & 2.2361 & 0.1447 & 3.8877 \end{bmatrix}$$

となる。

以上の準備のもと、Algorithm 1 で与えられる OMP 法、および  $\nu(x) = \nu_1 > 0$  (定数) として Algorithm 2 の FISTA で計算される  $\ell^1/\ell^2$  最適な制御バケットを用いてバケット化予測制御のシミュレーションを時刻  $k = 0$  から 99 まで行う。比較のため、従来法である次の標準的な  $\ell^2$  最適化による制御バケットの設計も考える：

$$\min_{u \in \mathbb{R}^N} \frac{\nu_2}{2} \|u\|_2^2 + \frac{1}{2} \|Gu - Hx\|_2^2.$$

ここで  $\nu_2 > 0$  は定数である。なお、この最適解  $u_2(x)$  は以下のように解析的に求めることができる：

$$u_2(x) = (\nu_2 I + G^\top G)^{-1} G^\top Hx. \quad (6)$$

(注2)：これらの行列の各要素は、平均 0、分散 1 の正規分布から独立にランダムサンプルされたものである。

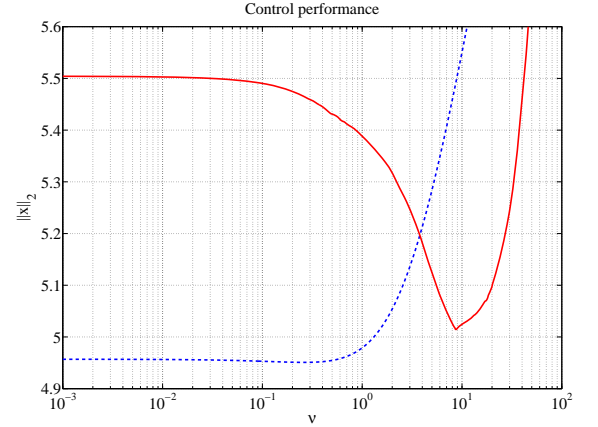


図3 Regularization parameter  $\nu$  versus control performance  $\|x\|_2$  for  $\ell^1/\ell^2$  optimization (solid) and  $\ell^2$  optimization (dash).

さらに、上の  $\ell^2$  最適化で  $\nu_2 = 0$  とおいた理想的な制御バケット<sup>(注3)</sup>

$$u^*(x) = (G^\top G)^{-1} G^\top Hx.$$

も比較の対象とする。

まず、 $\ell^1/\ell^2$  最適化および  $\ell^2$  最適化における正則化パラメータ  $\nu_1 > 0$  と  $\nu_2 > 0$  を決定する。これらのパラメータを  $10^{-3}$  から  $10^2$  まで変化させて、制御性能、すなわち、状態  $\{x(k)\}_{k=0}^{99}$  の  $\ell^2$  ノルムを計算したものを図 3 に示す。この図より、制御性能が最も良くなる  $\nu_1 = 8.5$  と  $\nu_2 = 0.25$  をそれぞれ  $\ell^1/\ell^2$  最適化および  $\ell^2$  最適化の正則化パラメータとして採用する。

シミュレーションは 500 回行った。それぞれのシミュレーションにおいて、バケットロスは仮定 1 の条件を満たすようにランダムに生成し、また制御対象の初期状態  $x(0) = x_0$  は、その要素を平均 0、分散 1 の正規分布から独立にランダムサンプルして選ぶ。

図 4 に時刻  $k$  における状態  $x(k)$  の  $\ell^2$  ノルムの平均値を示す。この図より、OMP 法および  $\ell^2$  最適化による制御バケットと理想的な制御バケットはほぼ指数安定性を示しているが、 $\ell^1/\ell^2$  最適な制御バケットによる制御はある値以下にはならない。これらの結果は、本稿の定理 1 および文献 [7] で示された  $\ell^1/\ell^2$  最適バケットによる安定性定理の結果と一致している。特に、提案法である OMP 法による制御と従来法である  $\ell^2$  最適化による制御で、制御性能はほぼ互角であることが図 4 よりわかる。

図 5 に OMP 法および  $\ell^1/\ell^2$  最適化による制御バケットのスパースさ、すなわち  $\ell^0$  ノルムの平均値を示す。なお、 $\ell^1/\ell^2$  最適化による制御バケットおよび理想的な制御バケットは、常に密なベクトル、すなわち  $\ell^0$  ノルムが  $N = 10$  となる。この図より、OMP 法で約 40% の、 $\ell^1/\ell^2$  最適化で約 75% の要素が 0 となることがわかる。 $\ell^1/\ell^2$  最適化では、制御ベクトルの計算のために漸近安定性を犠牲にした<sup>が</sup>、その分、漸近安定性を示す OMP 法による制御ベクトルよりもはるかにスパースなベク

(注3)：この制御バケット  $u^*(x)$  は誤差  $\|Gu - Hx\|_2^2$  を最小にするという意味で理想的である。

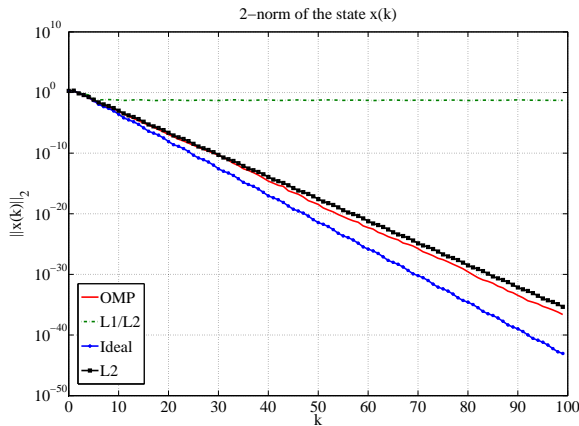


図 4 Average 2-norm of the state  $x(k)$ .

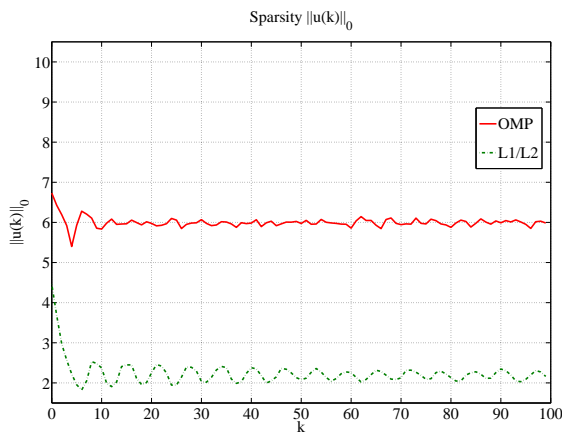


図 5 Average sparsity of the control vectors.

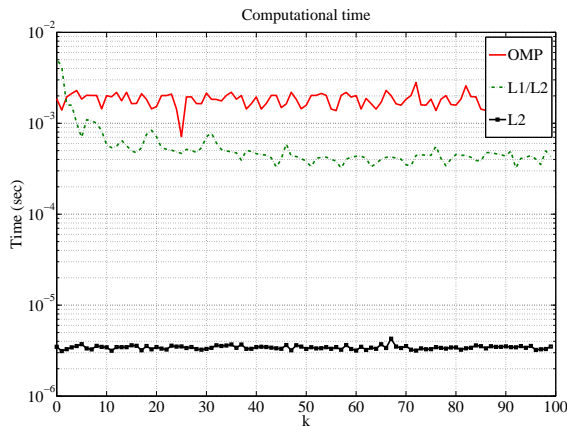


図 6 Average computational time.

トルが得られていることがわかる。

次にそれぞれの方法による最適制御ベクトルの計算時間の平均値を図 6 に示す。FISTA アルゴリズムを用いた  $\ell^1/\ell^2$  最適化のほうが OMP 法よりも約 5 倍ほど早く計算できることがわかる。なお、 $\ell^2$  最適化の計算は、(6) で与えられる公式により、行列  $(\nu_2 I + G^T G)^{-1} G^T H$  をオフラインであらかじめ計算しておいて、それを状態の観測ベクトル  $x(k)$  に掛けるだけであるので、非常に高速に計算できる。

以上のシミュレーション結果をまとめる。OMP 法によるパケット化予測制御では、フィードバック系を漸近安定性につつ、スパースな制御ベクトルを効率的に求めることができる。いっぽう、 $\ell^1/\ell^2$  最適化による制御では、漸近安定性は成り立たないが、OMP 法に比べ、はるかにスパースな制御ベクトルがより高速に求まる。従来法の  $\ell^2$  最適化も漸近安定を達成することができ、また制御ベクトルの計算は行列の掛け算だけであるので非常に高速であるが、制御ベクトルはスパースとはならず、効率的なデータ圧縮には向かないと考えられる。

## 6. おわりに

本稿では、パケットロスに対してロバストな制御手法であるパケット化予測制御を紹介し、その枠組みにもとづき、通信帯域の制約に対処するために、制御ベクトルのスパース表現を提案した。フィードバック系を漸近安定にする制御ベクトルを特徴付け、そのなかで最もスパースなベクトルを求める  $\ell^0$  最適化として設計問題を定式化した。この問題はそのままでは組み合わせ最適化であり、その近似解を求めるための高速解法として OMP 法、および  $\ell^1/\ell^2$  最適化による緩和とその高速解法である FISTA を導入した。数値例により従来法である  $\ell^2$  最適化と比較した提案手法の有効性を示し、さらに OMP 法および  $\ell^1/\ell^2$  最適化の利点と欠点を明らかにした。

## 文 献

- [1] A. Beck and M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.*, Vol. 2, No. 1, pp. 183-202, 2009.
- [2] A. Bemporad, M. Heemels, and M. Johansson, *Networked Control Systems*, Springer, Oct. 2010.
- [3] M. Elad, *Sparse and Redundant Representations*, Springer, 2010.
- [4] J. Hespanha, P. Naghshtabrizi, and Y. Xu, A survey of recent results in networked control systems, *Proc. IEEE*, vol. 95, no. 1, pp. 138-162, Jan. 2007.
- [5] J. M. Maciejowski, *Predictive Control with Constraints*, Pearson Education, 2002.
- [6] S. G. Mallat and Z. Zhang, Matching pursuits with time-frequency dictionaries, *IEEE Trans. Signal Process.*, Vol. 41, No. 12, pp. 3397-3415, 1993.
- [7] M. Nagahara and D. E. Quevedo, Sparse representations for packetized predictive networked control, *IFAC world congress*, pp. 84-89, 2011.
- [8] B. K. Natarajan, Sparse approximate solutions to linear systems, *SIAM J. Comput.*, vol. 24, no. 2, pp. 227-234, 1995.
- [9] D. E. Quevedo and D. Nešić, Input-to-state stability of packetized predictive control over unreliable networks affected by packet-dropouts, *IEEE Trans. Automat. Contr.*, Vol. 56, No. 2, pp. 370-375, 2011.
- [10] W. Zhang, M. Branicky, and S. Phillips, Stability of networked control systems, *IEEE Control Syst. Mag.*, vol. 21, no. 1, pp. 84-99, Feb. 2001.
- [11] M. Zibulevsky and M. Elad, L1-L2 optimization in signal and image processing, *IEEE Signal Processing Magazine*, Vol. 27, pp. 76-88, 2010.