Sparse command generator for remote control

by Masaaki Nagahara, Daniel E. Quevedo, Jan Østergaard, Takahiro Matsuda, & Kazunori Hayashi

Sparse Command Generator for Remote Control

Masaaki Nagahara, Daniel E. Quevedo, Jan Østergaard, Takahiro Matsuda, and Kazunori Hayashi

Abstract— In this article, we consider remote-controlled systems, where the command generator and the controlled object are connected with a bandwidth-limited communication link. In the remote-controlled systems, efficient representation of control commands is one of the crucial issues because of the bandwidth limitations of the link. We propose a new representation method for control commands based on compressed sensing. In the proposed method, compressed sensing reduces the number of bits in each control signal by representing it as a sparse vector. The compressed sensing problem is solved by an ℓ^1-ℓ^2 optimization, which can be effectively implemented with an *iterative shrinkage algorithm*. A design example also shows the effectiveness of the proposed method.

I. INTRODUCTION

Compressed sensing has recently been a focus of intensive researches in the signal processing community. It aims at reconstructing a signal by assuming that the original signal is sparse [2]. The core idea used in this area is to introduce a sparsity index in the optimization. The sparsity index of a vector $\boldsymbol{v}$ is defined by the amount of nonzero elements in $\boldsymbol{v}$ and is usually denoted by $\|\boldsymbol{v}\|_0$, called the "ℓ^0 norm." The compressed sensing decoding problem is then formulated by least squares with ℓ^0-norm regularization. The associated optimization problem is however hard to solve, since it is a combinatorial one. Thus, it is common to introduce a convex relaxation by replacing the ℓ^0 norm with the ℓ^1 norm [3]. Under some assumptions, the solution of this relaxed optimization is known to be exactly the same as that of the ℓ^0-norm regularization [8], [2]. That is, by minimizing the ℓ^1-regularized least squares, or by ℓ^1-ℓ^2 *optimization*, one can obtain a sparse solution. Moreover, recent studies have examined fast algorithms for ℓ^1-ℓ^2 optimization [5], [1], [15].

The purpose of this paper is to investigate the use of sparsity-inducing techniques for remote control [11], see [10] for an alternative approach. In remote-controlled systems, control information is transmitted through bandwidth-limited channels such as wireless channels [14] or the Internet [9].

M. Nagahara and K. Hayashi are with Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan, nagahara@ieee.org, kazunori@i.kyoto-u.ac.jp.

D. E. Quevedo is with School of Electrical Engineering & Computer Science, The University of Newcastle, Callaghan, NSW 2308, Australia, dquevedo@ieee.org.

J. Østergaard is with Department of Electronic Systems, Multimedia Information and Signal Processing (MISP), Aalborg University, Niels Jernes Vej 12, A6-305, DK-9220, Aalborg. jo@es.aau.dk

T. Matsuda is with Graduate School of Engineering, Osaka University, Osaka, 5650871, Japan, matsuda@comm.eng.osaka-u.ac.jp

There are two approaches to reduce the number of bits transmitted on a wireless link, *source coding* and *channel coding* approaches [4]. In the former, information compression techniques reduce the number of bits to be transmitted. In the latter, efficient forward error-correcting codes reduce redundant data (i.e., parity) in channel-coded information. In this paper, we study the former approach and propose a sparsity-inducing technique to produce sparse representation of control commands, which can reduce the number of bits in transmitted data.

Our optimization to obtain sparse representation of control commands is formulated as follows: we measure the tracking error in the output trajectory of a controlled system by its ℓ^2 norm, and add an ℓ^1 penalty to achieve sparsity of transmitted vector. This is an ℓ^1-regularized ℓ^2-optimization, or shortly ℓ^1-ℓ^2-optimization, which is effectively solved by the iterative shrinkage method mentioned above. The problem of command generator has been solved when the penalty is taken solely as an ℓ^2 norm, the solution of which is given by a linear combination of base functions, called control theoretic splines [13]. In this work, we also present a simple method for achieving sparse control vectors when the control commands are assumed to be in a subspace of these splines. An example illustrates the effectiveness of our method compared with the ℓ^2 optimization.

Notation

For a vector $\boldsymbol{v} = [v_1, \ldots, v_n]^\top \in \mathbb{R}^n$, the ℓ^1 and ℓ^2 norms are respectively defined by $\|\boldsymbol{v}\|_1 := \sum_{i=1}^n |v_i|$ and $\|\boldsymbol{v}\|_2 := \sqrt{\boldsymbol{v}^\top \boldsymbol{v}}$. For a real number $x \in \mathbb{R}$,

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

We denote the determinant of a square matrix M by $\det(M)$, and the maximum eigenvalue of a symmetric matrix M by $\lambda_{\max}(M)$. Let $L^2[0, T]$ be the set of Lebesgue square integrable functions on $[0, T]$. For $f, g \in L^2[0, T]$, the inner product is defined by

$$\langle f, g \rangle := \int_0^T f(t)g(t)\mathrm{d}t.$$

II. COMMAND GENERATION PROBLEM

Let us consider the following linear SISO (Single-Input Single-Output) plant:

$$P : \begin{cases} \dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + \boldsymbol{b}u(t), \\ y(t) = \boldsymbol{c}^\top \boldsymbol{x}(t), \quad t \in [0, \infty), \quad x(0) = 0, \end{cases} \quad (1)$$

where $A \in \mathbb{R}^{n \times n}$, $\boldsymbol{b} \in \mathbb{R}^n$ and $\boldsymbol{c} \in \mathbb{R}^n$. We assume that the system P is stable and the state space realization (1) is reachable and observable. The output reference signal is given by data points $\mathcal{D} := \{(t_1, Y_1), (t_2, Y_1), \ldots (t_N, Y_N)\}$, where t_i's are time instants such that $0 < t_1 < t_2 < \cdots < t_N =: T$. Our objective here is to design the control signal $u(t)$ such that the output trajectory $y(t)$ is close to the data points $Y_1, \ldots, Y_N$ at $t = t_1, \ldots, t_N$, that is, $y(t_i) \approx Y_i$, $i = 1, \ldots, N$. To measure the difference between $\{y(t_i)\}_{i=1}^N$ and $\{Y_i\}_{i=1}^N$, we adopt the square-error cost function

$$\mathbf{E}_2(u) = \sum_{i=1}^N (y(t_i) - Y_i)^2,$$

where we have made the dependence of $y(t_i)$ on $u = \{u(t)\}_{t \in [0,T]}$ through the system equation (1).

In principle, one can achieve perfect tracking, that is, $\mathbf{E}_2 = 0$, by some input signal[1]. However, the optimal input for perfect tracking has very large gain especially when the number N is very large, and may lead to oscillation between the sampling instants $t_1, \ldots, t_N$. This phenomenon is known as overfitting [12]. To avoid this, one can adopt a *regularization* or *smoothing* technique. This method is to add a regularization term $\Omega(u)$ to the cost function $\mathbf{E}_2(u)$. We formulate our problem as follows:

Problem 1: Given data $\mathcal{D}$, find a control signal u which minimizes the regularized cost function $\mathbf{J}(u) = \mathbf{E}_2(u) + \mu\Omega(u)$, where $\mu > 0$ is the regularization parameter which specifies the tradeoff between minimization $\mathbf{E}_2(u)$ and the smoothness by $\Omega(u)$.

A well-known regularization is to use L^2 function for $\Omega(u)$, called the *control theoretic smoothing spline* [13], [6]. We review this in the next section.

III. ℓ^2 COMMAND DESIGN BY CONTROL THEORETIC SMOOTHING SPLINES

For the problem given in section II, the following L^2-regularized cost function was considered in [13]:

$$\mathbf{J}_2(u) := \mathbf{E}_2(u) + \mu\Omega_2(u), \quad \Omega_2(u) := \int_0^T u(t)^2 \mathrm{d}t. \quad (2)$$

The optimal control u_2^* which minimizes $\mathbf{J}_2(u)$ is given by a linear combination of the following functions called control theoretic splines [13], [6]:

$$g_i(t) := \begin{cases} \boldsymbol{c}^\top \mathrm{e}^{A(t_i - t)} \boldsymbol{b}, & \text{if } t_i > t, \\ 0, & \text{if } t_i \leq t, \end{cases} \quad (3)$$

see Fig. 1. More precisely, the optimal control for (2) is given by

$$u_2^*(t) = \sum_{i=1}^N \theta_i g_i(t) = \boldsymbol{g}(t)^\top \boldsymbol{\theta}_2^*, \quad (4)$$

$$\boldsymbol{\theta}_2^* := (\mu I + G)^{-1} \boldsymbol{y}_{\mathrm{ref}}, \quad (5)$$

[1]The explicit form of this input is given by (4) and (5) in Section III, with $\mu = 0$.
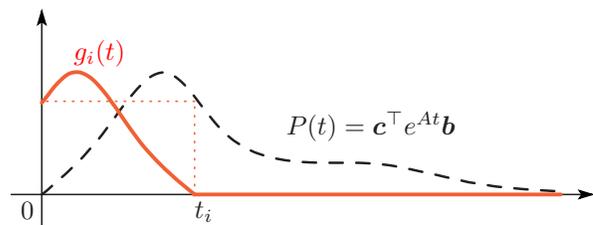

Fig. 1. Control theoretic spline $g_i(t)$ (solid) and the impulse response $P(t)$ of the plant P (dots).
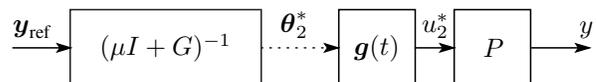

Fig. 2. Remote-controlled system optimized with $\mathbf{J}_2(u)$ in (2). The vector $\boldsymbol{\theta}_2^*$ is transmitted through a communication channel.

where $\boldsymbol{g}(t) := [g_1(t), \ldots, g_N(t)]^\top$, $\boldsymbol{y}_{\mathrm{ref}} := [Y_1, \ldots, Y_N]^\top$, and G is the Grammian matrix of $\{g_1, \ldots, g_N\}$, defined by $[G]_{ij} := \langle g_i, g_j \rangle$, $i, j = 1, \ldots, N$.

IV. ℓ^1-ℓ^2 COMMAND DESIGN FOR SPARSE REMOTE CONTROL

In remote-controlled systems, we transmit the control input $u = \{u(t)\}_{t \in [0,T]}$ to the system P through a communication channel. Since $\{u(t)\}_{t \in [0,T]}$ is a continuous-time signal, we should discretize it.

An easy way to communicate information on the input signal is to transmit the data $\boldsymbol{y}_{\mathrm{ref}}$ itself, and produce the input $u(t)$ by the formulae (4) and (5) at the receiver side. The vector $\boldsymbol{y}_{\mathrm{ref}}$ is just an N-dimensional one, and much easier to transmit than the infinite-dimensional vector $\{u(t)\}_{t \in [0,T]}$.

An alternative method consists in transmitting the coefficient vector $\boldsymbol{\theta}_2^*$ given in (5) instead of the continuous-time signal u. This procedure is shown in Fig. 2. In this procedure, we fix the sampling instants $t_1, \ldots, t_N$ and the vector $\boldsymbol{y}_{\mathrm{ref}}$ is given. We first compute the parameter vector $\boldsymbol{\theta}_2^*$ by (5), and transmit this through a communication channel. The transmitted vector is received at the receiver, and then the control signal $u_2^*(t)$ is computed by (4), and applied to the plant P. We assume that the time instants $t_1, \ldots, t_N$ are shared at the transmitter and the receiver.

A problem of the above-mentioned strategies is that the communication channel is band-limited and therefore the vector to be transmitted has to be first quantized and encoded. To solve this, we will seek a *sparse representation* of the transmitted vector $\boldsymbol{\theta}$ in accordance with the notion of compressed sensing [2], [7].

Define a subspace $\mathcal{V}$ of $L^2[0, T]$ by

$$\mathcal{V} := \left\{ u \in L^2[0, T] : u = \sum_{j=1}^M \theta_j \phi_j, \ \theta_i \in \mathbb{R} \right\}, \quad (6)$$

where $\phi_1, \ldots, \phi_M$ are linearly independent vectors in $L^2[0, T]$. Note that if $M = N$ and $\phi_i = g_i$, $i = 1, \ldots, N$ defined in (3), the optimal control $u_2^*(t)$ in (4) belongs to

this subspace[2]. We assume that the control u is in $\mathcal{V}$, that is, we find a control u in this subset. Under this assumption, the squared-error cost function $\mathbf{E}_2(u)$ is represented by

$$\mathbf{E}_2(u) = \sum_{i=1}^{N} (y(t_i) - Y_i)^2 = \|\Phi\boldsymbol{\theta} - \mathbf{y}_{\mathrm{ref}}\|_2^2, \qquad (7)$$

where $[\Phi]_{ij} = \langle g_i, \phi_j \rangle$, $i = 1, \ldots, N$, $j = 1, \ldots, M$. To induce sparsity in $\boldsymbol{\theta}$, we adopt ℓ^1 penalty on $\boldsymbol{\theta}$ and introduce the following mixed ℓ^1-ℓ^2 cost function:

$$\mathbf{J}_1(\boldsymbol{\theta}) := \frac{1}{2}\|\Phi\boldsymbol{\theta} - \mathbf{y}_{\mathrm{ref}}\|_2^2 + \kappa\|\boldsymbol{\theta}\|_1. \qquad (8)$$

Note that if $\|\phi_j\|_1 = 1$ for $j = 1, \ldots, M$, then the cost function (8) is an upper bound of the following L^1-L^2 cost function:

$$\mathbf{J}_1(u) = \frac{1}{2}\mathbf{E}_2(u) + \kappa\boldsymbol{\Omega}_1(u), \quad \boldsymbol{\Omega}_1(u) = \int_0^T |u(t)|\mathrm{d}t.$$

As mentioned in the introduction, the ℓ^1-regularized least-squares optimization is a good approximation to one regularized by the ℓ^0 norm which counts the nonzero elements in $\boldsymbol{\theta}$. Although the solution which minimizes $\mathbf{J}_1(\boldsymbol{\theta})$ cannot be represented analytically as in (4), we can compute an approximated solution by using a fast numerical algorithm. The algorithm is described in the next section. By using this solution, say $\boldsymbol{\theta}_{\mathrm{sparse}}^*$, the optimal control u_1^* can be obtained from

$$u_1^*(t) = \sum_{i=1}^{N} \theta_i^* \phi_i(t) = \phi(t)^\top \boldsymbol{\theta}_{\mathrm{sparse}}^*, \quad t \in [0, T].$$

V. Sparse Representation by ℓ^1-ℓ^2 Optimization

We here describe a fast algorithm for obtaining the optimal vector $\boldsymbol{\theta}_{\mathrm{sparse}}^*$. First, we consider a general case of optimization. Next, we simplify the design procedure in a special case.

A. General case

The cost function (8) is convex in θ and hence the optimal value $\boldsymbol{\theta}_{\mathrm{sparse}}^*$ uniquely exists. However, an analytical expression as in (5) for this optimal vector is unknown except when the matrix Φ is unitary. To obtain the optimal vector $\boldsymbol{\theta}_{\mathrm{sparse}}^*$, one can use an iteration method. Recently, a very fast algorithm for the optimal ℓ^1-ℓ^2 solution has been proposed, which is called *iterative shrinkage* [1], [15].

This algorithm is given by the following: Give an initial value $\boldsymbol{\theta}[0] \in \mathbb{R}^M$, and let $\beta[1] = 1$, $\boldsymbol{\theta}'[1] = \boldsymbol{\theta}[0]$. Fix a constant c such that $c > \|\Phi\|^2 := \lambda_{\max}(\Phi^\top\Phi)$. Execute the

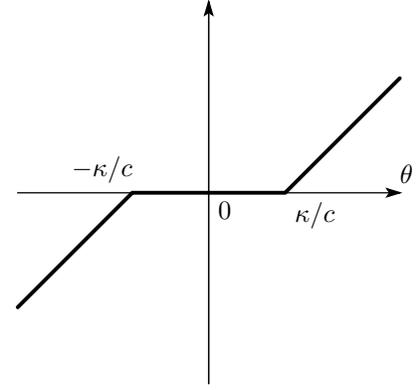[2]The functions $\{g_1, \ldots, g_N\}$ are linearly independent [13].

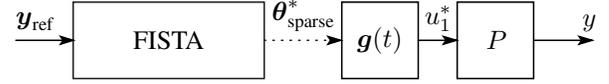Fig. 3. Nonlinear function $\mathrm{sgn}(\theta)(|\theta| - \kappa/c)_+$

Fig. 4. Remote-controlled system optimized with $\mathbf{J}_1(\boldsymbol{\theta})$ in (8). The vector $\boldsymbol{\theta}_{\mathrm{sparse}}^*$ minimizing (8) is computed by the FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) given in (9), and transmitted through a communication channel.

following iteration[3]:

$$\boldsymbol{\theta}[j] = \mathcal{S}_{\kappa/c}\left(\frac{1}{c}\Phi^\top(\mathbf{y}_{\mathrm{ref}} - \Phi\boldsymbol{\theta}'[j]) + \boldsymbol{\theta}'[j]\right),$$
$$\beta[j+1] = \frac{1 + \sqrt{1 + 4\beta[j]^2}}{2}, \qquad (9)$$
$$\boldsymbol{\theta}'[j+1] = \boldsymbol{\theta}[j] + \frac{\beta[j] - 1}{\beta[j+1]}(\boldsymbol{\theta}[j] - \boldsymbol{\theta}[j-1]),$$
$$j = 1, 2, \ldots,$$

where the function $\mathcal{S}_{\kappa/c}$ is defined for $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_M]^\top$ by

$$\mathcal{S}_{\kappa/c}(\boldsymbol{\theta}) := \begin{bmatrix} \mathrm{sgn}(\theta_1)(|\theta_1| - \kappa/c)_+ \\ \vdots \\ \mathrm{sgn}(\theta_M)(|\theta_M| - \kappa/c)_+ \end{bmatrix}.$$

The nonlinear function $\mathrm{sgn}(\theta)(|\theta| - \kappa/c)$ in $\mathcal{S}_{\kappa/c}$ is shown in Fig. 3. If $c > \|\Phi\|^2$, the above algorithm converges to the optimal solution minimizing the ℓ^1-ℓ^2 cost function (8) for any initial value $\boldsymbol{\theta}[0] \in \mathbb{R}^M$ with a worst-case convergence rate $O(1/j^2)$ [5], [1]. The above algorithm is very simple and fast; it can be effectively implemented in digital devices, which leads to a real-time computation of a sparse vector $\boldsymbol{\theta}_{\mathrm{sparse}}^*$.

B. The case $\Phi = G$

We here assume $M = N$ and $\phi_i = g_i$, $i = 1, 2, \ldots, N$, that is, $\Phi = G$. Since $g_1, \ldots, g_N$ are linearly independent

[3]Several methods have been proposed for the iterative shrinkage [15]. The algorithm given here is called FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) [1].
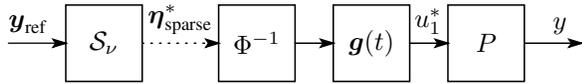
Fig. 5. Remote-controlled system optimized with $J(\boldsymbol{\eta})$ in (10). The vector $\boldsymbol{\eta}$ is transmitted through a communication channel.

vectors in $L^2[0,T]$, the Grammian matrix $\Phi = G$ is non-singular. Let the control input u be

$$u(t) = \sum_{i=1}^{N} \theta_i g_i(t) = \boldsymbol{g}(t)^\top \boldsymbol{\theta},$$

and let $\boldsymbol{\eta} := \Phi\boldsymbol{\theta}$. Then, by (7) we have

$$\sum_{i=1}^{N} (y(t_i) - Y_i)^2 = \|\boldsymbol{\eta} - \boldsymbol{y}_{\text{ref}}\|_2^2.$$

Consider the following ℓ^1-ℓ^2 cost function:

$$J(\boldsymbol{\eta}) = \nu\|\boldsymbol{\eta}\|_1 + \frac{1}{2}\|\boldsymbol{\eta} - \boldsymbol{y}_{\text{ref}}\|_2^2. \qquad (10)$$

The optimal solution $\boldsymbol{\eta}^*_{\text{sparse}}$ minimizing this cost function is given analytically by

$$\boldsymbol{\eta}^*_{\text{sparse}} = \mathcal{S}_\nu(\boldsymbol{y}_{\text{ref}}). \qquad (11)$$

Then we transmit this optimal vector $\boldsymbol{\eta}^*_{\text{sparse}}$, and at the receiver we reconstruct the optimal control by $u_1^*(t) = \boldsymbol{g}(t)^\top \Phi^{-1}\boldsymbol{\eta}^*_{\text{sparse}}$. Fig. 5 shows the remote-controlled system with the optimizer $\boldsymbol{\eta}^*_{\text{sparse}}$. In this case, we compute (11) only one time, while in the general case considered in Section V-A we should execute the iteration algorithm (9).

VI. EXAMPLE

We here show an example of the sparse command generator. The state-space matrices of the controlled plant P is assumed to be

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

Note that the transfer function of the plant P is $1/(s+1)^2$. The sampling instants are given by $t_i = i \times \pi/6$, $i = 1, 2, \ldots, 12$, and the data $Y_1, \ldots, Y_{12}$ is given by $Y_i = \sin t_i$, that is, we try to track the sine function $y(t) = \sin t$ in one period $[0, 2\pi]$. We assume the base functions ϕ_i in the subspace $\mathcal{V}$ in (6) are the same as g_i's, that is, we consider the case $\Phi = G$ discussed in Section V-B. We design three signals to be transmitted: the ℓ^2-optimized vector $\boldsymbol{\theta}_2^*$ in (5), the sparse vector $\boldsymbol{\theta}^*_{\text{sparse}}$ given in subsection V-A, and the sparse vector $\boldsymbol{\eta}^*_{\text{sparse}}$ in (11). We set the regularization parameters $\mu = 0.01$, $\kappa = 0.001$, and $\nu = 0.05$, see equations (2), (8) and (10).

The obtained vectors are shown in Table I. We can see that the vector $\boldsymbol{\theta}^*_{\text{sparse}}$ is the sparsest due to the sparsity-inducing approach. The second sparsest vector is $\boldsymbol{\eta}^*_{\text{sparse}}$ which converts small elements in $\boldsymbol{y}_{\text{ref}}$ to 0. The vector $\boldsymbol{\theta}_2^*$ is not sparse.

Fig. 6 shows the plant outputs obtained by the above vectors. The transient responses show relatively large errors

tbp

TABLE I
DESIGNED VECTORS

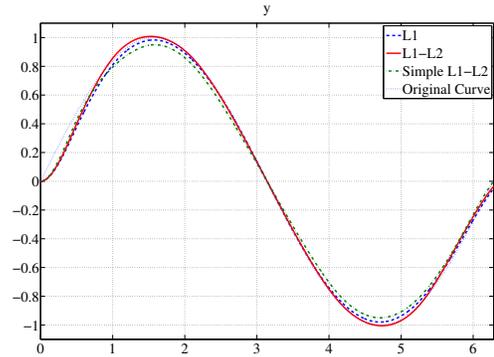| $\boldsymbol{\theta}_2^*$	$\boldsymbol{\theta}^*_{\text{sparse}}$	$\boldsymbol{\eta}^*_{\text{sparse}}$	$\boldsymbol{y}_{\text{ref}}$
9.7994	9.6727	0.4500	0.5000
2.7995	4.5626	0.8160	0.8660
1.6544	0	0.9500	1.0000
1.6695	2.9973	0.8160	0.8660
1.0358	0	0.4500	0.5000
0.0059	0	0	0.0000
-1.0231	0	-0.4500	-0.5000
-1.7456	-2.8678	-0.8160	-0.8660
-2.0234	-0.6316	-0.9500	-1.0000
-2.2424	-4.8575	-0.8160	-0.8660
-2.4153	0	-0.4500	-0.5000
5.1813	4.4185	0	-0.0000

Fig. 6. The original curve (dots) and outputs: by ℓ^2-optimal $\boldsymbol{\theta}_2^*$ (dash), ℓ^1-ℓ^2-optimal $\boldsymbol{\theta}^*_{\text{sparse}}$ (solid), and simple ℓ^1-ℓ^2-optimal $\boldsymbol{\eta}^*_{\text{sparse}}$ (dash-dots).
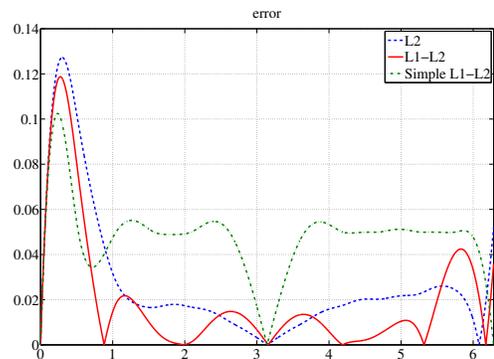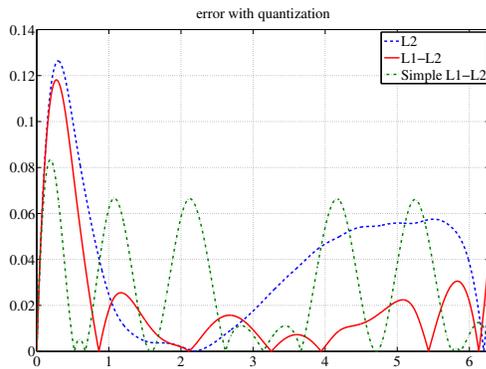

Fig. 7. The reconstruction errors: by ℓ^2-optimal $\boldsymbol{\theta}_2^*$ (dash), ℓ^1-ℓ^2-optimal $\boldsymbol{\theta}^*_{\text{sparse}}$ (solid), and simple ℓ^1-ℓ^2-optimal $\boldsymbol{\eta}^*_{\text{sparse}}$ (dash-dots).

advantages of sparse representation in view of information theory.

TABLE II

QUANTIZED VECTORS

$Q(\boldsymbol{\theta}_2^*)$	$Q(\boldsymbol{\theta}_{\text{sparse}}^*)$	$Q(\boldsymbol{\eta}_{\text{sparse}}^*)$	$Q(\boldsymbol{y}_{\text{ref}})$
9.8	9.7	0.5	0.5
2.8	4.6	0.8	0.9
1.7	0.0	1.0	1.0
1.7	3.0	0.8	0.9
1.0	0.0	0.5	0.5
0.0	0.0	0.0	0.0
-1.0	0.0	-0.5	-0.5
-1.7	-2.9	-0.8	-0.9
-2.0	-0.6	-1.0	-1.0
-2.2	-4.9	-0.8	-0.9
-2.4	0.0	-0.5	-0.5
5.2	4.4	0.0	0.0

Fig. 8. The reconstruction errors with quantization: by ℓ^2-optimal $\hat{\boldsymbol{\theta}}_2^*$ (dash), ℓ^1-ℓ^2-optimal $\hat{\boldsymbol{\theta}}_{\text{sparse}}^*$ (solid), and simple ℓ^1-ℓ^2-optimal $\hat{\boldsymbol{\eta}}_{\text{sparse}}^*$ (dash-dots).

REFERENCES

[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. Candes and M. B. Wakin, An introduction to compressive sampling, *IEEE Signal Processing Mag.*, vol. 25, 21/30 (2008)
[3] S. S. Chen, D. L. Donoho, and M. A. Saunders, Atomic decomposition by basis pursuit, *SIAM Journal on Sci. Comp.*, vol. 20, No. 1, pp. 33–61 (1998)
[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd edition, Wiley (2006)
[5] I. Daubechies, M. Defrise, and C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, *Communications on Pure and Applied Mathematics*, Vol. 57, No. 11, pp. 1413–1457 (2004)
[6] M. Egerstedt and C.F. Martin, *Control Theoretic Splines*, Princeton University Press (2010)
[7] M. Elad, *Sparse and Redundant Representations*, Springer (2010)
[8] M. Elad and A. Bruckstein, A generalized uncertainty principle and sparse representation in pairs of bases, *IEEE Trans. Inform. Theory*, Vol. 48, No. 9, pp. 2558–2567 (2002)
[9] R. C. Luo and T. M. Chen, Development of a multibehavior-based mobile robot for remote supervisory control through the Internet, *IEEE/ASME Trans. Mechatron.*, Vol. 5, No. 4 (2000)
[10] M. Nagahara and D. E. Quevedo, Sparse representation for packetized predictive networked control, *IFAC World Congress* (2011) (to be presented)
[11] C. Sayers, *Remote Control Robotics*, Springer (1999)
[12] B. Schölkopf and A. J. Smola, *Learning with Kernels*, The MIT Press (2002)
[13] S. Sun, M. B. Egerstedt, and C. F. Martin, Control theoretic smoothing splines, *IEEE Trans. Automat. Contr.* Vol. 45, No. 12, 2271/2279 (2000)
[14] A. F. T. Winfield and O. E. Holland, The application of wireless local area network technology to the control of mobile robots, *Microprocessors and Microsystems*, Vol. 23, pp. 597–607 (2000)
[15] M. Zibulevsky and M. Elad, L1-L2 optimization in signal and image processing, *IEEE Signal Processing Mag.*, Vol. 27, 76/88 (2010)

because of the phase delay in the plant $P(s) = 1/(s + 1)^2$. Despite of sparsity in $\boldsymbol{\theta}_{\text{sparse}}^*$ and $\boldsymbol{\eta}_{\text{sparse}}^*$, the performances of the reconstructed signals are comparable to that of the ℓ^2-optimal reconstruction by $\boldsymbol{\theta}_2^*$. To see the difference between these performances more precisely, we draw the reconstruction errors in Fig. 7. We can see that the errors by $\boldsymbol{\theta}_2^*$ and $\boldsymbol{\theta}_{\text{sparse}}^*$ are almost comparable, and the error by $\boldsymbol{\eta}_{\text{sparse}}^*$ is relatively large.

Then we consider quantization. We use the uniform quantizer with step size 0.1 and simulate the output reconstruction. Table II shows the quantized vectors. Fig. 8 shows the reconstruction error under quantization. The errors by the sparse vectors $\boldsymbol{\theta}_{\text{sparse}}^*$ and $\boldsymbol{\eta}_{\text{sparse}}^*$ still remains small while the ℓ^2-optimal reconstruction shows errors affected by quantization. This is because the zero-valued elements in the sparse vectors do not suffer from any quantization distortion.

VII. CONCLUSION

In this paper, we have proposed to use sparse representation for command generation in remote control by ℓ^1-ℓ^2 optimization. An example illustrates the effectiveness of the proposed method. Future work may include the study of