Paper

Dividing the grids of compressed sensing for channel estimation and investigating Markov codes

Dongshin Yang\textsuperscript{1a}) and Yutaka Jitsumatsu\textsuperscript{2b})

\textsuperscript{1} Dept. of Informatics, Grad. Sch. of Information Science and Electrical Engineering, Kyushu University
Motooka 744, Nishi-ku, Fukuoka 819-0395, Japan

\textsuperscript{2} Dept. of Informatics, Kyushu University
Motooka 744, Nishi-ku, Fukuoka 819-0395, Japan

\textsuperscript{a}) yang@me.inf.kyushu-u.ac.jp
\textsuperscript{b}) jitumatu@inf.kyushu-u.ac.jp

Received July 10, 2017; Revised December 1, 2017; Published April 1, 2018

Abstract: Bajwa et al. proposed a channel estimation method based on compressed sensing. This method is markedly superior to the conventional methods. However, there is a problem in the method that multi-path delays may not be resolved if they span between the grids. We study to overcome the drawback of the method. Firstly, we investigate upsampled codes so that we could more accurately estimate the channel. Secondly, we investigate Markov codes. It was shown that Spread Spectrum (SS) codes with negative autocorrelation reduces the Multiple Access Interference (MAI) as well as Bit Error Rate (BER) in chip-asynchronous Spread Spectrum Multiple Access (SSMA) systems. Such SS codes are generated from a Markov chain whose transition probability matrix has a negative eigenvalue $\lambda = -2 + \sqrt{3}$. The mean square error (MSE) of channel estimation using upsampled Markov code is shown to be better than the MSE of channel estimation using upsampled independent and identically distributed code at certain conditions.

Key Words: channel estimation, compressed sensing, Markov chain code, upsampling

1. Introduction

In any communication system, a transmitted signal is distorted by a channel. Such a channel distortion leads to degradation of the bit error rate. Thus, the channel must be estimated and compensated for in some way. As the wireless communication speed increases, accurate channel estimation is an important factor for reliability of communication. Compressed Sensing (CS) can be applied to such channel estimation [1, Section 6.1]. CS is a relatively new signal processing technique which enables sub-Nyquist rate signal acquisition from sparse signal [2].

Bajwa proposed channel estimation methods based on CS. The channel estimation accuracy of
CS is markedly superior to that of the conventional methods based on MUltiple SSignal Classification (MUSIC) and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) algorithms [3]. However, there is a problem in the method that multi-path delays may not be resolved if they span between the grids. This problem leads to inaccurate channel estimation. This is because Bajwa’s method adopted a discretization procedure to reduce the continuous parameter space to a finite set of grid points. Discretization has several significant drawbacks. Firstly, in cases where the true parameters do not fall onto the finite grid, the signal cannot often be sparsely represented by the discrete dictionary. Secondly, although finer grids may improve the reconstruction error in the methods, very fine grids often lead to numerical instability issues [4]. The problem leads to inaccurate channel estimation. Therefore, we try to overcome the drawbacks of discretization.

In this paper, we concentrate our attention on the one using Spread-Spectrum (SS) signal. The received signal is upsampled to overcome the discretization problem and to detect a delay that spans between the grids [5, 6]. A Markov chain model for generating a SS signal is used. It was shown that SS codes with negative autocorrelation reduces the Multiple Access Interference (MAI) as well as Bit Error Rate (BER) in chip-asynchronous Spread Spectrum Multiple Access (SSMA) systems [7–9]. Figure 1 shows MAI concept and our application idea. In the chip-asynchronous SSMA systems, there are several users with different signals. MAI is decreased because of the different signals. Similarly, in the channel estimation with upsampled signal, there are paths with different time delay. Therefore, autocorrelation is expected to be decreased. Such SS codes are generated from a Markov chain whose transition probability matrix has a negative eigenvalue \( \lambda = -2 + \sqrt{3} \) [9]. Interestingly, \( \lambda = -2 + \sqrt{3} \) is optimal if the system is in a chip-asynchronous state, whereas \( \lambda = 0 \) is optimal if it is in a chip-synchronous state. This fact suggests that Markov codes may improve the performance of Bajwa’s channel estimation method using SS codes, if the received signal is upsampled. We will show that SS code generated by a Markov chain improves the MSE of the channel estimation compared with an independent and identically distributed (i.i.d.) code sequence.

2. Compressed sensing overview

In this section, we briefly review Compressed Sensing (CS) theory. In signal processing, the purpose of sampling is to accurately capture the distinct information in a signal of interest using as few samples as possible. CS reconstructs the signal from far fewer samples than required by the Nyquist rate. Also, CS assumes that the original signal has a sparse representation.

CS theory deals with incomplete linear equation systems of the type

\[
y = Xh.
\]

where the \( N \)-dimensional vector \( y \) collects the measurements or observations obtained by a linear sensor, the \( M \)-dimensional coefficient vector \( h \) describes unknown sparse signals, and \( X \) is an \( N \times M \) matrix - the sensing matrix - characterizing how the coefficient vector is mapped to the measurements. CS assumes that \( N < M \). The vectors and matrices may be real or complex valued. A vector \( h \) is called \( K \)-sparse if at most \( K \) of its coefficients are unequal to zero. The goal of CS is to reliably recover \( h \) from the knowledge of \( y \) and \( X \).

The sparsest solution \( l_0 \)-minimization is always a Non-deterministic Polynomial-time hard (NP-hard) problem. However, the sparse signal \( h \) can be reconstructed from measurements by solving via \( l_1 \)-minimization:

\[
\min \| h \|_0 \quad \text{subject to} \quad y = Xh.
\]

\[\text{In this paper, we use boldface upper case letters (i.e., } \mathbf{X} \text{) and boldface lower case letters (i.e., } \mathbf{x} \text{) to denote matrices and vectors, respectively. The superscript } (\cdot)^H \text{ denotes the Hermitian transpose.}
\]

260
where $\hat{h}$ is the reconstructed signal of $h$. There are a variety of CS algorithms. The Dantzig Selector (DS) is one of the few reconstruction methods in the CS literature that are guaranteed to perform near-optimally about stochastic noise [10]. The DS gives an estimated channel vector as follows:

$$h_{\text{DS}} = \min_{h \in \mathbb{C}^M} ||h||_1 \text{ subject to } \|X^H(y - Xh)\|_\infty \leq \varepsilon,$$

where $\varepsilon$ is a constraint parameter which decided by channel environment and $||h||_\infty = \max_i |h_i|$. The $\varepsilon$ in frequency-selective single-antenna environment is determined by

$$\varepsilon = \sqrt{2E(1 + a) \log M},$$

where for any $a \geq 0$, $E$ is energy of transmitted signal and $M$ is maximum delay [3]. The parameter $a$ must be chosen carefully. Otherwise the performance of DS is degraded.

### 3. Channel model and proposed method

In this section, we describe the Bajwa’s method [3] and then extend it to the case where the received signal is upsampled. The purpose of upsampling is to overcome the discretization problem. In addition, Markov chain model for generating spread spectrum code is also explained. Firstly, Fig. 2 shows the system model of continuous-time representation. A single antenna, frequency selective channel is assumed. Let $s(t)$ be a transmitted signal given by

$$s(t) = \sum_{n=0}^{N_0-1} x_n u_{T_c}(t - nT_c),$$

where $x_n \in \{+1, -1\}$ is an SS code, $T_c$ is a chip time, $N_0$ is the code length (or the spreading factor) and

$$u_{T_c}(t) = \begin{cases} \frac{1}{\sqrt{T_c}}, & 0 \leq t \leq T_c, \\ 0, & \text{otherwise.} \end{cases}$$

The received signal is given by

$$r(t) = \int_0^\infty h(\tau)s(t - \tau)d\tau + z(t),$$

where $z(t)$ is complex additive white Gaussian noise (AWGN) signal. For nonupsampled system with $J = 1$, the chip-matched filter output is given by

$$y_n = \int_0^{T_c} r(nT_c + t)u_{T_c}(t)dt.$$

The sampled AWGN signal is given by

$$z_n = \int_0^{T_c} z(nT_c + t)u_{T_c}(t)dt.$$
Fig. 3. The channel impulse response estimated by DS.

\[ h(t) = \sum_{k=1}^{K} c_k \delta(t - t_k), \]  

(10)

where \( t_k \) and \( c_k \) are a real-valued (continuous valued) delay and a complex-valued attenuation factor of the \( k \)-th path, \( K \) is the number of paths, and \( \delta(t) \) is Dirac’s delta function.

In Bajwa’s method, continuous-time signal is discretized. A discrete-time version of \( h(t) \) is represented as

\[ h_d(t) = \sum_{m=0}^{M-1} h[m] \delta(t - mT_c), \]  

(11)

where \( h[m] \) is a discrete-time version of \( h(t) \), and \( M \) is a maximum delay. It is expected that \( h[m] \approx c_k \) if \( t_k \approx mT_c \). However, if \( t_k \)s are located between the grid, we may not estimate \( c_k \) accurately. A matrix-vector form of Eq. (8) for nonupsampled discrete time model is given by

\[ y = Xh + z, \]  

(12)

where, \( y = (y_0, y_1, \ldots, y_{N_0-1})^T \), \( X \) is a Toeplitz matrix whose \((n, m)\) component is \( x_{n-m} \), i.e.,

\[ X = \begin{bmatrix} x_0 & 0 & \cdots & 0 \\ x_1 & x_0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \vdots & \vdots & \ddots & x_0 \\ -x_{N_0-1} & -x_{N_0-2} & \cdots & x_{N_0-M} \end{bmatrix}_{N_0 \times M} \]  

(13)

\[ h = (h[0], \ldots, h[M-1])^T \], and \( z = (z_0, z_1, \ldots, z_{N_0-1})^T \).

We shortly review a well-known phenomenon that we cannot detect the delay spans between the grids.

Figure 3(a) shows an example of DS for a discrete-time model, where the number of paths \( K = 3 \) and true time delays are \( (2.0682, 38.9548, 51.4785) \). The true channel response is shown in the red line and the channel estimation of discrete-time model is shown in the yellow diamond marker. In this example, one path (2.0682) is not estimated. This is because the signal cannot often be sparsely represented by the discrete dictionary. If the true delay spans between the grids, the delay was not accurately estimated.

Let us consider an upsampled model of the matched filter output for estimating multi-path delays spanning between grids. The upsampling factor is denoted by \( J \geq 1 \). The matched filter output is sampled at \( t = nT_c \) for nonupsampling system and at \( t = nT_{cJ} \) for upsampling system. Then \( y_n \) in Eq. (8) is replaced by

\[ y_{up,n} = \int_0^{T_c} r(nT_{cJ} + t)u_T(t)dt. \]  

(14)
2.50 GHz 2.60 GHz, RAM 8.00 GB, and 64 bit system. The energy of transmitted signal (Disciplined Convex Programming). The computer system was Intel® Core™ i7-6500U CPU @ 2.50 GHz 2.60 GHz, RAM 8.00 GB, and 64 bit system. The energy of transmitted signal

\[ \text{SNR} = \frac{||y(t)||^2}{||z(t)||^2}. \]  

(17)

We generate a sequence of the chip-matched filter output \( y_n \) by Eq. (8) and its upsampled output \( y_{\text{up},n} \), where channel impulse response is given by Eq. (10). Then we obtain \( \hat{h}[m] \) and \( \hat{h}_{\text{up}}[m] \) by DS for Bajwa’s model and the upsampled model. For these two case, we have

\[
\begin{cases}
\hat{h}(t) = \sum \hat{h}[m] \delta(t - mT_c) \\
\hat{h}_{\text{up}}(t) = \sum \hat{h}_{\text{up}}[m] \delta(t - m\frac{T}{J}).
\end{cases}
\]  

(18)

4. Spread spectrum and Markov chain codes

Spreading codes are denoted by \( \{x_n\}_{n=0}^{N_0-1} \). Linear Feedback Shift Register (LFSR) sequences, such as M sequence, are often employed as spreading codes. It is also often assumed SS codes are independent and identically distributed (i.i.d.) sequences [3]. In this paper, we compare a Markov chain model with a sequence of i.i.d. random variables. Its purpose is to show the superiority of Markovian codes. Figure 4 shows a simple two-state(+1, -1) Markov chain state transition diagram. Let us consider a simple Markov chain for an SS code sequence \( \{x_n\}_{n=0}^{N_0-1} \in \{+1, -1\} \) with transition probability matrix

\[
P = \{p_{ij}\} = \frac{1}{2} \begin{bmatrix} 1 + \lambda & 1 - \lambda \\ 1 - \lambda & 1 + \lambda \end{bmatrix},
\]  

(16)

where \( p_{ij} = \text{prob}\{x_n = (-1)^j \mid x_n = (-1)^i\} \), \( i, j \in \{1, 2\} \) and \( 0 \leq \lambda \leq 1 \) is an eigenvalue of \( P \). Sensing matrix \( X_{\text{up}} \) whose components \( x_n \)'s are generated by a Markov chain are called Markovian sensing matrix.

5. Simulation and results

In this section, simulation results of the proposed channel estimation method and Markovian sensing matrix are presented. The simulation was performed on MATLAB and CVX (Matlab Software for Disciplined Convex Programming). The computer system was Intel(R) Core(TM) i7-6500U CPU @ 2.50 GHz 2.60 GHz, RAM 8.00 GB, and 64 bit system. The energy of transmitted signal \( E \) is 1. The chip time \( T_c \) is 1. The symbol time interval \( T \) is 1. The delays of continuous value \( t_k \) are generated independently of the number of objects \( k \), following the uniform distribution on \([0, M]\). The attenuation factors \( c_k \) follow Gaussian distribution with zero mean and variance one, also independently of \( K \). Signal-to-Noise Ratio (SNR) is expressed as

\[
\text{SNR} = \frac{||y(t)||^2}{||z(t)||^2}.
\]  

(17)

The matrix-vector representation of the upsampled matched filter output is given by

\[
y_{\text{up}} = X_{\text{up}} h_{\text{up}} + z_{\text{up}},
\]  

(15)

where \( y_{\text{up}} \) is an \( N_0 J \)-dimensional vector, \( X_{\text{up}} \) is a \((N_0 J \times M J)\) Toeplitz matrix whose first column is determined by \( \{x_0, x_0, \ldots, x_0, x_1, \ldots, x_1, \ldots, x_{N_0-1}, \ldots, x_{N_0-1}\} \), i.e., the upsampled spreading code \( \{x_n\} \), first row is determined by \( \{x_0, 0, \ldots, 0\} \), \( h_{\text{up}} \) is an \( M J \)-dimensional vector, and \( z_{\text{up}} \) is an \( N_0 J \)-dimensional vector. Note that upsampled model with \( J = 1 \) is reduced to a discrete-time model without up sampling. Thus, we use \( y_{\text{up}}, X_{\text{up}}, h_{\text{up}}, z_{\text{up}} \) for both discrete-time models with and without up sampling. Figure 3(b) shows upsampling technique. Contrary to discrete-time model, the missed one path (2.0682) can be estimated. Thus, upsampling is effective for detecting the delay spanning between the grids. However, it is known that very fine grids often lead to numerical instability issues [4].

![Fig. 4. A simple two-state(+1, -1) Markov chains state transition diagram.](image-url)
5.1 Compare the upsampled model with the conventional method

Estimated delay by using DS for a discrete-time model shown in Fig. 5, where the number of paths is \( K = 5 \) and true time delays are blue line with circle. The parameter for Dantzig Selector \( a = 80 \). The maximum delay \( M \) is 64. The code length \( N_0 \) is 32. The transmitted signal \( x \) is an i.i.d. code sequence. The channel estimation of nonupsample is shown in yellow line with diamond marker. In this case, all paths are not estimated accurately. This is because the true paths do not fall onto the finite grid. The signal cannot often be sparsely represented by the discrete dictionary. Meanwhile, the delays (green line with \( J = 2 \) and blue line with square \( J = 4 \)) by using the upsampled model are more accurate than the nonupsampled signal.

5.2 Calculation Time

It is known that very fine grids often lead to numerical instability issues [4]. Therefore, we investigate calculation time. Maximum delay \( M \) and the code length \( N_0 \) is 64, the transmitted signal \( x \) is an i.i.d. code sequence. The parameter for Dantzig Selector \( a = 80 \) and the number of paths \( K \) is 5. In Fig. 6, as a comparison, a graph of \( T = 0.05 \times J^3 \) is plotted. This figure implies that the calculation time of the channel estimation using DS under the upsampled signal model increased. This prevents us to employ the upsampled model with very large upsampling factor \( J \).

5.3 Compare Markov chain code with independent and identically distributed code

In this subsection, we investigate the effect of parameter \( \lambda \) for MSE of Markovian sensing matrix. Channel is randomly generated which is denoted by \( h_r \). The code length \( N_0 \) is 32, the number of paths \( K \) is 5, the upsampling parameter \( J \) is 4, the maximum delay \( M \) is 64, SNR is 10 [dB] and eigenvalue parameter for Markov codes \( \lambda \) is \( \{-1, -0.9, -0.8, -0.7, \ldots, 0.2\} \). These methods were repeated 1000 independent Monte Carlo runs. In Fig. 7, MSE is plotted against \( \lambda \). A general MSE is expressed as
Comparing MSE of Markov chain code with MSE of i.i.d. code ($N_0 = 32$).

$$\text{MSE} = \frac{1}{R} \sum_{r=1}^{R} \int |h_r(t) - \hat{h}_r(t)|^2 dt, \quad (19)$$

where $R$ is the number of independent runs. However, it is not appropriate to evaluate $h$ and $\hat{h}$ because of the positional shift of delay. Firstly, in case of $|h_0(t) - \hat{h}_0(t)|^2$, $t_k \neq \hat{t}_k$ is 0 or $t_k = \hat{t}_k$ is $\infty$. Secondly, when the number of $K$ between true delays and estimated delays are different, $t_k - \hat{t}_k$ and $c_k - \hat{c}_k$ could not be used. Otherwise, if $h$ and $\hat{h}$ have width, we can evaluate the positional shift of delay by using MSE. Therefore, we use a filter to evaluate the MSE as

$$\text{MSE} = \frac{1}{R} \sum_{r=1}^{R} \int |h_r(t) * u_T(t) - \hat{h}_r(t) * u_T(t)|^2 dt. \quad (20)$$

In Fig. 7, MSE curves are depicted with respect to $\lambda$. The MSEs of Markov chain codes with $-0.7 \leq \lambda \leq -0.1$ are less than the MSE of i.i.d. codes.

5.4 SNR vs MSE

We investigate the effect of SNR for MSE. The maximum delay $M$ is 64. The code length $N_0$s are 32 and 64. The number of paths is $K = 5$. Parameter for Dantzig selector $a$ is 80. The eigenvalue parameters for Markov codes $\lambda$ are $-0.4$ and $-0.6$. The upsampling factor $J$ is 4. The $t_k$ and $c_k$ were the same at one run. This method was repeated 1000 independent Monte Carlo runs. In Fig. 8, MSE curves are depicted with respect to SNR. The Markov code is shown to be superior to i.i.d. code for SNR from $-10$ [dB] to $30$ [dB].

5.5 Effect of $N_0$ and the parameter for Dantzig selector $a$

We investigate the effect of the code length $N_0$ and the parameter for Dantzig selector $a$. The maximum delay is 64. The code length are 32 and 512. The number of paths is $K = 5$, SNR is 10 [dB] and upsampling factor $J$ is 4. In Figs. 9(a) and 9(b) (The parameter for Dantzig selector $a = 80$) show false estimations that have small values $\hat{c}_k$ appear if $N_0$ increases. The parameter for Dantzig selector $a$ is changed from 80 to 800. In Figs. 9(c) and 9(d) show the small values $\hat{c}_k$ are decreased.
Fig. 9. Different results by $a$ and $N_0$.

Fig. 10. The effect of $N_0$ between Markov chain code and i.i.d. code ($\lambda = -0.4$).

5.6 $N_0$ vs MSE

We evaluate $N_0$ by using MSE. The maximum delay is 64. The code length are \{32, 64, \ldots, 512\}. The number of paths is $K = 5$, the eigenvalue parameter for Markov codes $\lambda$ is $-0.4$, SNR is 10 [dB] and upsampling factor $J$ is 4. The parameter for Dantzig selector $a$ was changed from 80 to 800. This is because false estimations that have small value $\hat{c}_k$ appear if $N_0$ increases. We can control the parameter $a$ to reduce the false estimations. As $N_0$ increases, the estimate for the true value becomes more accurate. However, there are false estimations due to the noise, regardless of the true value. $t_k$ and $c_k$ were the same at one run. This method was repeated 1000 independent Monte Carlo runs.

In Fig. 10, MSE curves are depicted with respect to the code length $N_0$. We observe that the Markov code is better than i.i.d. code for all $N_0 = 32, 48, 64, 96, 128, 256,$ and 512.

6. Conclusion

We have studied a channel estimation based on compressed sensing. The upsampled version of Bajwa’s method is employed and the performance is measured by mean square error (MSE) of the estimated channel coefficients. For improved performance of channel estimation, we investigated spread spectrum codes generated from a Markov chain. Markov codes are characterized by an eigenvalue parameter $\lambda$. This paper showed that MSE of Markov codes with $\lambda \in [-0.7, -0.1]$ is less than that of i.i.d. codes. The calculation time of the channel estimation using DS under the upsampled signal
model increased significantly. This prevents us to employ the upsampled model with very large $J$. We also investigated the effect of signal-to-noise ratio (SNR) for MSE of Markovian sensing matrix and MSE of i.i.d codes. The MSEs decreased as SNR increases. We investigate effect of the code length $N_0$ and the parameter for Dantzig selector $a$. The MSE decreased as $N_0$ increases. However, false estimations that have small values $\hat{c}_k$ appear if $N_0$ increases. We could control the false estimations throughout the parameter for Dantzig selector $a$. The idea of using Markov codes and the upsampled received signal are also immediately applicable to super-resolution target detection using radar [11]. This is because communications and radar exploit similar electromagnetic phenomena and are well described by similar mathematical theories.

Acknowledgments
This research was supported in part by JSPS KAKENHI Grant Number JP16K000333.

References
[1] K. Hayashi, M. Nagahara, and T. Tanaka, “A user’s guide to compressed sensing for communications systems,” IEICE Trans. on Communications, vol. E96-B, no. 3, pp. 685–712, March 2013.
[2] D.L. Donoho, “Compressed sensing,” IEEE Trans. on Information Theory, vol. 52, no. 4, pp. 1289–1306, April 2006.
[3] W.U. Bajwa, et al., “Compressed channel sensing: A new approach to estimating sparse multipath channels,” Proceedings of the IEEE, vol. 98, no. 6, pp. 1058–1076, June 2010.
[4] G. Tang, et al., “Compressed sensing off the grid,” IEEE Trans. on Information Theory, vol. 59, no. 11, pp. 7465–7490, November 2013.
[5] D. Yang and Y. Jitsumatsu, “Parameter discretization problem in channel estimation method based on compressed sensing,” IEICE General Conference, A-8-20, March 2016.
[6] D. Yang and Y. Jitsumatsu, “Discretization in channel estimation using compressed sensing and its performance improvement,” IEEE VTS Asia Pacific Wireless Communications Symposium, August 2016.
[7] G. Mazzini, G. Setti, and R. Rovatti, “Chaotic complex spreading sequences for asynchronous DS-CDMA. I. System modeling and results,” IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications, vol. 44, no. 10 pp. 937–947, October 1997.
[8] H. Watanabe, Y. Narumiya, and M. Hasegawa, “Performance evaluation of chaotic CDMA using an implemented system on software defined radio,” IEICE, Nonlinear Theory and Its Applications, vol. 4, no. 4, pp. 473–481, October 2013.
[9] T. Kohda and H. Fujisaki, “Variances of multiple access interference code average against data average,” Electronics Letters, vol. 36, no. 20, pp. 1717–1719, September 2000.
[10] E. Candès and T. Tao, “The Dantzig selector: statistical estimation when $p$ is much larger than $n$,” The Annals of Statistics, vol. 35, no. 6, pp. 2313–2351, December 2007.
[11] W.U. Bajwa, K. Gedalyahu, and Y.C. Eldar, “Identification of parametric underspread linear systems and super-resolution radar,” IEEE Trans. on Signal Processing, vol. 59, no. 6, pp. 2548–2561, June 2011.