Journal of Southeast University (English Edition) Vol. 26, No. 3, pp. 389 – 393 Sept. 2010 ISSN 1003—7985

Digital broadcast channel estimation with compressive sensing

Qi Chenhao Wu Lenan

(School of Information Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: In order to reduce the pilot number and improve spectral efficiency, recently emerged compressive sensing (CS) is applied to the digital broadcast channel estimation. According to the six channel profiles of the European Telecommunication Standards Institute (ETSI) digital radio mondiale (DRM) standard, the subspace pursuit (SP) algorithm is employed for delay spread and attenuation estimation of each path in the case where the channel profile is identified and the multipath number is known. The stop condition for SP is that the sparsity of the estimation equals the multipath number. For the case where the multipath number is unknown, the orthogonal matching pursuit (OMP) algorithm is employed for channel estimation, while the stop condition is that the estimation achieves the noise variance. Simulation results show that with the same number of pilots, CS algorithms outperform the traditional cubic-spline-interpolation-based least squares (LS) channel estimation. SP is also demonstrated to be better than OMP when the multipath number is known as *a priori*.

Key words: channel estimation; compressive sensing (CS); digital radio mondiale (DRM); orthogonal frequency division multiplexing (OFDM)

Compressive sensing (CS) has recently emerged as a collection of principles and methodologies which enables efficient reconstruction of sparse signals from relatively few linear measurements[1-2]. We roughly divide CS algorithms into two classes: greedy algorithms and convex optimization algorithms. Greedy algorithms make a sequential locally optimal choice in an effort to determine a globally optimal solution. They are matching pursuit (MP)[3], orthogonal matching pursuit (OMP)[4], subspace pursuit (SP)[5] and many other variants[6-8]. Compared with convex optimization algorithms, they have a much lower complexity, which indicates that they are more appropriate for practical applications.

Recently, MP has been applied for pilot assisted channel estimation in orthogonal frequency division multiplexing (OFDM) systems[9-10]. It is demonstrated that the pilot number can be substantially reduced compared with the traditional interpolation based least squares (LS) method while the performance is similar. Especially for time-vary-ing channels where channel estimations should be frequently carried out, CS algorithms can save a large number of pilots and thus improve the spectral efficiency. It is also beneficial for the multi-input multi-output (MIMO) system to employ CS algorithms since the pilots increase linearly with the number of transmit antennas.

Received 2010-02-26.

Biographies: Qi Chenhao (1981—), male, doctor, lecturer; Wu Lenan (corresponding author), male, doctor, professor, wuln@seu.edu.cn.

Foundation items: The National Natural Science Foundation of China (No. 60872075), the National High Technology Research and Development Program of China (863 Program) (No. 2008AA01Z227).

Citation: Qi Chenhao, Wu Lenan. Digital broadcast channel estimation with compressive sensing [J]. Journal of Southeast University (English Edition), 2010, 26(3): 389 − 393.

Although MP can find an approximated solution from the overcomplete dictionary with asymptotic convergence, the shortcoming lies in the fact that it may select the same column several times and thus pull down the efficiency. It is necessary to adopt more powerful CS algorithms for different occasions.

In this paper, we apply CS algorithms as OMP and SP for digital broadcast channel estimation. According to the ETSI digital radio mondiale (DRM) standard, six channel profiles are classified. For the case where the multipath number is unknown, we apply the OMP algorithm for channel estimation. For the case where the channel profile is identified and the multipath number is known, we apply the SP algorithm for delay spread and attenuation estimation of each path. In both cases, we formulate pilot assisted OFDM frequency domain channel estimation as a sparse recovery problem. We also compare their performance with the traditional interpolation-based LS method using different numbers of pilots.

1 Problem Formulation

We model channel impulse response (CIR) of multipath propagation as

$$h(\tau, t) = \sum_{p=1}^{S} \alpha_p(t)\delta(\tau - \tau_p(t)) \tag{1}$$

where S, α_p and τ_p are the multipath number, amplitude attenuation and delay spread for path p. According to the ETSI DRM standard[11], we have six channel profiles as listed in Tab. 1. Channel 1 is the pure AWGN channel, which is not common in practice. Channels 2, 4 and 5 are two-path channels with increased delay spreads. Both channels 3 and 6 are four-path channels where eight parameters should be estimated.

With block fading channel assumptions where the channel parameters are constant over each block and supposing perfect symbol synchronization, we model the equivalent discrete CIR as

$$h(m) = \sum_{p=1}^{S} \alpha_p \delta((m - \tau_p)T_s) \tag{2}$$

where T_s is the sampling interval of the receiver. We notice that the elementary time period T of the OFDM symbol in the DRM standard is no more than 0.1 ms, so T_s should be less than 0.05 ms, which is very small compared with the maximum delay spread and results in a channel with relatively few nonzero taps.

Supposing the number of channel taps to be L and S of them nonzero, we define it as an S-sparse channel. Considering an OFDM system, there are totally N subcarriers, among which M subcarriers are selected as pilots, with positions represented by $Q_1, Q_2, \dots, Q_M (1 \leqslant Q_1 < Q_2 < \dots < Q_M$

$\leqslant N$). We denote the transmitted pilots and the received pilots as $X(Q_1)$, $X(Q_2)$, …, $X(Q_M)$ and $Y(Q_1)$, $Y(Q_2)$, …, $Y(Q_M)$, respectively. We have

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{h} + \boldsymbol{\eta} \tag{3}$$

where $\boldsymbol{y} = [Y(Q_1), Y(Q_2), \ldots, Y(Q_M)]^{\mathrm{T}}$, $\boldsymbol{h} = [\boldsymbol{h}(1), \boldsymbol{h}(2), \ldots, \boldsymbol{h}(L)]^{\mathrm{T}}$, $\boldsymbol{\eta} = [\boldsymbol{\eta}(1), \boldsymbol{\eta}(2), \ldots, \boldsymbol{\eta}(M)]^{\mathrm{T}}$. Each component of vector $\boldsymbol{\eta}$ is a dependent complex Gaussian variable. $\boldsymbol{\eta} \sim \mathrm{CN}(0, \sigma I_M)$. I_M is an $M \times M$ identity matrix.

$$\boldsymbol{A} = \boldsymbol{Z}\boldsymbol{F} \tag{4}$$

where $\boldsymbol{F}$ is a submatrix selected by row index $[Q_1, Q_2, \ldots, Q_M]$ and column index $[1, 2, \ldots, L]$. $\boldsymbol{Z} = \mathrm{diag}\{X(Q_1), X(Q_2), \ldots, X(Q_M)\}$ is a diagonal square matrix with each component to be a pilot symbol.

Tab. 1 Six channel profiles in ETSI DRM standard

Channel number	Path 1		Path 2		Path 3		Path 4	
	α_1	τ_1/ms	α_2	τ_2/ms	α_3	τ_3/ms	α_4	τ_4/ms
1	1	0	0	0	0	0	0	0
2	1	0	0.5	1	0	0	0	0
3	1	0	0.7	0.7	0.5	1.5	0.25	2.2
4	1	0	1	2	0	0	0	0
5	1	0	1	4	0	0	0	0
6	0.5	0	1	2	0.25	4	0.062 5	6

If the rows of $\boldsymbol{A}$ are more than its columns, Eq. (3) can be solved by the traditional LS method. However, we are more interested in the under-determined case where the rows of $\boldsymbol{A}$ are fewer than its columns, which means that the pilots are less than unknown channel coefficients. Still we can use

$$\boldsymbol{H}(i) = \frac{\boldsymbol{Y}(i)}{\boldsymbol{X}(i)} \qquad i = Q_1, Q_2, \ldots, Q_M \tag{5}$$

first to figure out the channel transfer function (CTF) at pilot subcarriers and then make interpolations as linear or cubic spline interpolations for data subcarriers. Obviously, this method will yield large deviations because it does not use sparse conditions as *a priori*.

Here we focus on low-complexity greedy CS algorithms and divide the sparse channel estimation problem into two cases. In the first case, the multipath number is unknown, we apply the OMP algorithm for channel estimation. For the case where the channel profile is identified and the multipath number is known, we apply the SP algorithm for delay spread and the attenuation estimation of each path.

2 MP and OMP Algorithm

MP is a sort of algorithm that constructs a sparse solution by iteratively selecting dictionary elements best correlated with the residual part of the signal[3]. If the dictionary is a matrix, the objective of the construction is to find a linear combination of matrix columns which is the closest to the signal. At each step, one column that best correlates with the current residue is added to the current selection. Then, it updates the residue by projecting it onto the new selection. Here we briefly describe the MP and the OMP algorithms based on Eq. (3).

First, we generate a dictionary matrix $\boldsymbol{D}$ from $\boldsymbol{A}$,

$$\boldsymbol{A} = \boldsymbol{D} \cdot \boldsymbol{C} \tag{6}$$

where $\boldsymbol{D} \in \mathbf{R}^{M \times L}$ has the same dimension as $\boldsymbol{A}$ and each column of $\boldsymbol{D}$ is a unit vector. $\boldsymbol{C}$ is a diagonal matrix with each diagonal component corresponding to the normalized coefficient for each column of $\boldsymbol{A}$.

Let $\boldsymbol{d}_i$ denote the i-th column of $\boldsymbol{D}$ and $\boldsymbol{R}_k$ denote the residue at the k-th step. The selected column index at the k-th step is

$$l_k = \arg\max_{i \in \{1, 2, \ldots, L\}} |\langle \boldsymbol{d}_i, \boldsymbol{R}_k \rangle| \tag{7}$$

where $|\langle \boldsymbol{d}_i, \boldsymbol{R}_k \rangle|$ represents the absolute value of inner product between $\boldsymbol{d}_i$ and $\boldsymbol{R}_k$.

Now starting MP with an initial residue $\boldsymbol{R}_1 = \boldsymbol{y}$, the algorithm evolves by

$$\boldsymbol{R}_k = \langle \boldsymbol{d}_{l_k}, \boldsymbol{R}_k \rangle \boldsymbol{d}_{l_k} + \boldsymbol{R}_{k+1} \tag{8}$$

and replaces $\boldsymbol{R}_k$ with $\boldsymbol{R}_{k+1}$. Since $\boldsymbol{R}_{k+1}$ is orthogonal to $\boldsymbol{d}_{l_k}$, we have

$$\|\boldsymbol{R}_k\|_2^2 = |\langle \boldsymbol{d}_{l_k}, \boldsymbol{R}_k \rangle|^2 + \|\boldsymbol{R}_{k+1}\|_2^2 \tag{9}$$

With the increase of $k (k = 1, 2, \ldots)$, we minimize $\|\boldsymbol{R}_{k+1}\|_2$ till it satisfies the stop condition

$$\|\boldsymbol{R}_{k+1}\|_2 \leqslant \sigma \tag{10}$$

Supposing that the iteration ends in K steps, the solution is

$$\boldsymbol{d} = \sum_{i=1}^{K} \langle \boldsymbol{d}_{l_i}, \boldsymbol{R}_i \rangle \boldsymbol{d}_{l_i} \tag{11}$$

which is a linear combination of previous selected columns. Finally, the estimation of $\boldsymbol{h}$ is

$$\hat{\boldsymbol{h}}_{\mathrm{MP}} = \boldsymbol{C}^{-1}\boldsymbol{d} = \boldsymbol{C}^{-1} \sum_{i=1}^{K} \langle \boldsymbol{d}_{l_i}, \boldsymbol{R}_i \rangle \boldsymbol{d}_{l_i} \tag{12}$$

Although MP can rapidly find an approximation from an overcomplete dictionary with asymptotic convergence, the shortcoming lies in the fact that it may select the same columns several times and thus pull down the efficiency. Therefore, OMP is proposed with revision by using the residue's orthogonal component for the next iteration[4]. Only the component that is orthogonal with the space spanned by the previous selected columns is preserved. In most literature, the set containing previous selected columns is called the active set.

Here we denote the active set and its complementary set as I and I^{c}, respectively. $I \cup I^{\mathrm{c}} = \{1, 2, \ldots, L\}$. Unlike MP always selecting candidate column from $\{1, 2, \ldots, L\}$ as in Eq. (7), OMP selects it only from I^{c}, where the selected column index is

$$l_k = \arg\max_{i \in I^{\mathrm{c}}} |\langle \boldsymbol{d}_i, \boldsymbol{R}_k \rangle| \tag{13}$$

Then Gram-Schmidt orthogonalization is implemented to remove the component within the space spanned by I.

$$u_k = \boldsymbol{d}_{l_k} - \sum_{i \in I} \frac{\langle \boldsymbol{d}_{l_k}, \boldsymbol{u}_i \rangle}{\|\boldsymbol{u}_i\|_2^2} \boldsymbol{u}_i \tag{14}$$

where $\{\boldsymbol{u}_k\}$ is an iteratively generated set that can be regarded as the bases of the space spanned by I. We initialize $\boldsymbol{u}_1$ to be $\boldsymbol{d}_{l_1}$ and iteratively update the residue by

$$\boldsymbol{R}_k = \frac{\langle \boldsymbol{R}_k, \boldsymbol{u}_k \rangle}{\| \boldsymbol{u}_k \|_2^2} \boldsymbol{u}_k + \boldsymbol{R}_{k+1} \tag{15}$$

In this way, OMP adds different columns into the active set until the stop condition (10) is satisfied. Supposing it ends in K steps (usually $K \ll L$), the solution is

$$\boldsymbol{d} = \sum_{i=1}^{K} \frac{\langle \boldsymbol{R}_i, \boldsymbol{u}_i \rangle}{\| \boldsymbol{u}_i \|_2^2} \boldsymbol{u}_i \tag{16}$$

Eventually, the estimation of $\boldsymbol{h}$ is

$$\hat{\boldsymbol{h}}_{\mathrm{OMP}} = \boldsymbol{C}^{-1} \boldsymbol{d} = \boldsymbol{C}^{-1} \sum_{i=1}^{K} \frac{\langle \boldsymbol{R}_i, \boldsymbol{u}_i \rangle}{\| \boldsymbol{u}_i \|_2^2} \boldsymbol{u}_i \tag{17}$$

Compared with MP, OMP converges much faster. Even for a large dictionary matrix, the step is countable. It has already been demonstrated in Refs. [12 − 13] that in some circumstances, OMP does succeed in finding the sparsest solution.

But the discussion is going on. OMP works in a greedy way, which determines that the final solution is essentially suboptimal, not globally optimal. Besides, OMP always adds a new column to the active set, but never removes outdated columns from the active set. When a selection error occurs, the iteration will continue till reaching the end without correcting them adaptively. Although several revised versions of OMP have been proposed against these disadvantages[6-8], OMP is still one of the best candidates for practical applications due to its reasonable tradeoff between performance and complexity.

3 SP Algorithm

In the SP algorithm, S columns selection is iteratively refined from $\boldsymbol{A}$ until the stop condition is satisfied[5]. Although it is still developed based on the greedy rule, SP allows new columns to enter as well as to leave the active set. S columns are simultaneously selected rather than only one column in MP and OMP. In this way, the subspace spanned by S columns is tracked down.

The SP algorithm solves the problem of Eq. (3) and works as follows. First, it begins with the same step as Eq. (6) to normalize each column of $\boldsymbol{A}$. Then, it finds S columns from $\boldsymbol{D}$ on which $\boldsymbol{y}$ has the S largest projections, and stores their indices into a set $\hat{I}$. Meanwhile, the residue $\boldsymbol{R}_1$ is also obtained. The projection of $\boldsymbol{y}$ onto each column vector is defined as the absolute value of their inner product which is similar to Eq. (7).

Here we introduce the definition of the residue since it is different from MP and OMP.

Definition 1 If $\boldsymbol{D}^{\mathrm{H}}\boldsymbol{D}$ is invertible, the residue of $\boldsymbol{y}$ on matrix $\boldsymbol{D}$ is defined as

$$\boldsymbol{R} = \boldsymbol{y} - \boldsymbol{D}\boldsymbol{D}^{\dagger}\boldsymbol{y} \tag{18}$$

where

$$\boldsymbol{D}^{\dagger} = (\boldsymbol{D}^{\mathrm{H}}\boldsymbol{D})^{-1}\boldsymbol{D}^{\mathrm{H}} \tag{19}$$

denotes the pseudo inverse of $\boldsymbol{D}$. We simply write it as $\boldsymbol{R} = \mathrm{resid}(\boldsymbol{y}, \boldsymbol{D})$.

Algorithm 1 Subspace pursuit algorithm

Input: $\boldsymbol{A}$, $\boldsymbol{y}$, S;

Initialization: Normalize each column of matrix $\boldsymbol{A}$ with a coefficient diagonal matrix $\boldsymbol{C}$ so that $\boldsymbol{A} = \boldsymbol{D}\boldsymbol{C}$;

$\hat{I} = \{S$ indices corresponding to S columns of $\boldsymbol{D}$ on which $\boldsymbol{y}$ has the largest projections$\}$;

$\boldsymbol{R}_1 = \mathrm{resid}(\boldsymbol{y}, \boldsymbol{D}_{\hat{I}})$;

Iteration: $k = 1, 2, \ldots$

If $\boldsymbol{R}_k = \boldsymbol{0}$, quit the iteration; otherwise continue;

$I' = \hat{I} \cup \{S$ indices corresponding to S columns of $\boldsymbol{D}$ on which $\boldsymbol{R}_k$ has the largest projections$\}$;

Get $\boldsymbol{D}_{I'}^{\dagger}$ and figure out $\boldsymbol{x} = \boldsymbol{D}_{I'}^{\dagger}\boldsymbol{y}$;

$I_k = \{S$ indices corresponding to S components with the largest absolute value in $\boldsymbol{x}\}$;

$\boldsymbol{R}_{k+1} = \mathrm{resid}(\boldsymbol{y}, \boldsymbol{D}_{I_k})$;

If $\| \boldsymbol{R}_{k+1} \|_2 > \| \boldsymbol{R}_k \|_2$, quit the iteration;

Otherwise, let $\hat{I} = I_k$, increase k by one and continue the iteration;

Output: One estimated vector $\hat{\boldsymbol{x}}$ is yielded with a nonzero element indexed by $\hat{I}$ satisfying $\hat{\boldsymbol{x}}_{\hat{I}} = \boldsymbol{D}_{\hat{I}}^{\dagger}\boldsymbol{y}$;

The final result is $\hat{\boldsymbol{h}}_{\mathrm{sp}} = \boldsymbol{C}^{-1}\hat{\boldsymbol{x}}$.

For the last step of initialization, we obtain a submatrix $\boldsymbol{D}_{\hat{I}}$ from $\boldsymbol{D}$ by index set $\hat{I}$ and then figure out the residue $\boldsymbol{R}_1 = \mathrm{resid}(\boldsymbol{y}, \boldsymbol{D}_{\hat{I}})$. During the iterations, another S indices corresponding to S columns of $\boldsymbol{D}$ on which $\boldsymbol{R}_k$ has the largest projections are selected and added to the previous S columns set, forming I'. The number of columns in I' may be less than $2S$ since it is possible to have overlap between two selections. Then, we figure out $\boldsymbol{D}_{I'}^{\dagger}$ and a vector $\boldsymbol{x} = \boldsymbol{D}_{I'}^{\dagger}\boldsymbol{y}$, from which we choose S indices corresponding to S components with the largest absolute value. The active set I_k at step k is acquired and a new residue $\boldsymbol{R}_{k+1} = \mathrm{resid}(\boldsymbol{y}, \boldsymbol{D}_{I_k})$ is also yielded. If

$$\| \boldsymbol{R}_{k+1} \|_2 > \| \boldsymbol{R}_k \|_2 \tag{20}$$

the iteration is terminated as it means that the residue cannot be smaller. Otherwise, we replace $\hat{I}$ with I_k, increase k by one and continue the iteration. It is noticed that at the start of each iteration, current residue $\boldsymbol{R}_k$ is checked whether it is zero. If so, we also terminate the iteration.

Finally, one estimated vector $\hat{\boldsymbol{x}}$ is obtained. The nonzero locations of $\hat{\boldsymbol{x}}$ are indexed by $\hat{I}$ and satisfy $\hat{\boldsymbol{x}}_{\hat{I}} = \boldsymbol{D}_{\hat{I}}^{\dagger}\boldsymbol{y}$. The output is

$$\hat{\boldsymbol{h}}_{\mathrm{sp}} = \boldsymbol{C}^{-1}\hat{\boldsymbol{x}} \tag{21}$$

which is the estimated CIR for Eq. (3).

4 Simulation Results

In our simulations, we set OFDM parameters according to robustness mode B in the DRM standard as shown in Tab. 2. The channel profile is set to be channel 3 as described in Tab. 1. We define the mean square error (MSE) as

$$\mathrm{MSE}\{\hat{\boldsymbol{h}}\} = \frac{\| \hat{\boldsymbol{h}} - \boldsymbol{h} \|_2^2}{\| \boldsymbol{h} \|_2^2} \tag{22}$$

where $\hat{\boldsymbol{h}}$ is the estimate of $\boldsymbol{h}$.

In the first case, we suppose that the multipath number and the channel profile are unknown to the receiver, and we compare MSE performance for the OMP algorithm and the LS method.

The equal-spaced pilot placement and the random pilot placement are employed for LS and OMP, respectively. In Fig. 1, OMP with $M = 12$ is even better than cubic-spline-interpolation-based LS with $M = 69$ for SNR > 20 dB, while their spectral efficiency is 94% and 66%, respectively.

Tab. 2 System parameters

Parameters	Value
FFT length N_{FFT}	256
Used subcarriers N	206
Guard interval N_{g}	64
OFDM symbol period T_{s}/ms	26.7
OFDM sample period T/μs	83.3
Number of pilots M	12
Number of channel multipaths S	4
CIR length L	30
Modulation	QPSK

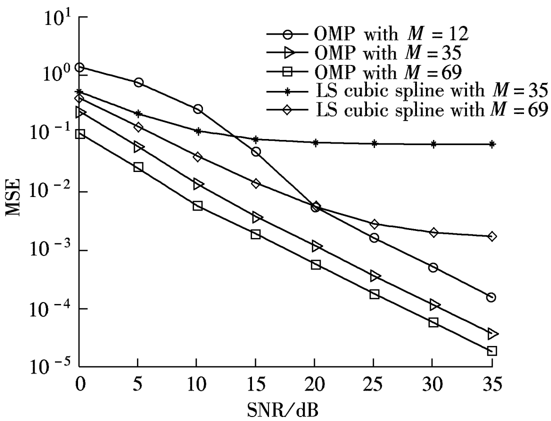

Fig. 1 MSE comparisons for OMP and LS

In the second case, we suppose that the multipath number is known as *a priori* to the receiver where we apply OMP and SP to delay spread and attenuation estimation of each path.

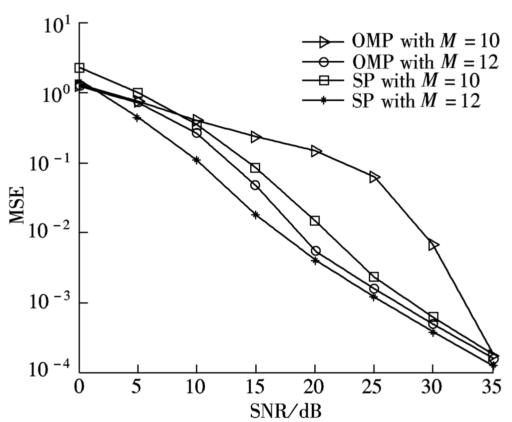

Fig. 2 MSE comparisons for OMP and SP

As shown in Fig. 2, when $M = 10$, SP is clearly superior to OMP. When $M = 12$, SP is still a little better than OMP. The reason is that, at each iterative step OMP always greedily selects one column vector, while SP selects several columns in batch. The possibility to correctly find one column with one selection is much lower than that with batch selection. Besides, once OMP selects one column into the active set, it never removes it, which means that if one error selection occurs, it will never be corrected in the later steps.

5 Conclusion

In this paper, we formulate the OFDM channel estimation as a sparse recovery problem according to the DRM standard. It is shown that CS algorithms are much better than the traditional cubic-spline-interpolation-based LS method. In the case where the multipath number is known, SP outperforms OMP.

References

[1] Donoho D L. Compressed sensing [J]. *IEEE Trans Inf Theory*, 2006, **52**(4): 1289 − 1306.

[2] Candes E J, Tao T. Near-optimal signal recovery from random projections: universal encoding strategies [J]. *IEEE Trans Inf Theory*, 2006, **52**(12): 5406 − 5425.

[3] Mallat S G, Zhang Z. Matching pursuits with time-frequency dictionaries [J]. *IEEE Trans Sig Proc*, 1993, **41**(12): 3397 − 3415.

[4] Tropp J A, Gilbert A C. Signal recovery from random measurements via orthogonal matching pursuit [J]. *IEEE Trans Inf Theory*, 2007, **53**(12): 4655 − 4666.

[5] Dai W, Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction [J]. *IEEE Trans Inf Theory*, 2009, **55**(5): 2230 − 2249.

[6] Neira L R, Lowe D. Optimized orthogonal matching pursuit approach [J]. *IEEE Sig Proc Lett*, 2002, **9**(4): 137 − 140.

[7] Andrle M, Neira L R, Sagianos E. Backward-optimized orthogonal matching pursuit approach [J]. *IEEE Sig Proc Lett*, 2004, **11**(9): 705 − 708.

[8] Qi C H, Wu L N. Sparse recovery algorithms for pilot assisted MIMO OFDM channel estimation [J]. *IEICE Trans Commun*, 2010, **E93-B**(5): 1313 − 1316.

[9] Cotter S F, Rao B D. Sparse Channel estimation via matching pursuit with application to equalization [J]. *IEEE Trans Commun*, 2002, **50**(3): 374 − 377.

[10] Sharp M, Scaglione A. Application of sparse signal recovery to pilot-assisted channel estimation [C]//*IEEE International Conference on Acoustics, Speech, and Signal Processing*. Las Vegas, NV, USA, 2008: 5296 − 5299.

[11] European Telecommunication Standards Institute. ETSI ES 201 980 v2.2.1. Digital radio mondiale (DRM) system specification [S]. European Telecommunication Standards Institute, 2005.

[12] Tropp J A. Greed is good: algorithmic results for sparse approximation [J]. *IEEE Trans Inf Theory*, 2004, **50**(10): 2231 − 2242.

[13] Donoho D L, Elad M, Temlyakov V N. Stable recovery of sparse overcomplete representations in the presence of noise [J]. *IEEE Trans Inf Theory*, 2006, **52**(1): 6 − 18.

采用压缩感知的数字广播信道估计

戚晨皓　　吴乐南

（东南大学信息科学与工程学院，南京 210096）

摘要：为降低导频数目并提高频谱利用率，将最新提出的压缩感知技术应用于数字广播的信道估计．在欧洲电信标准协会 DRM 标准规定的 6 种信道模式中，对于确定了信道模式多径数目的情况，采用子空间跟踪算法估计各径的时延扩展和衰减，设定算法迭代终止条件为估计结果的稀疏度等于多径数目；对于多径数目未知的情况，采用正交匹配追踪算法估计信道参数，设定算法迭代终止条件为估计误差等于噪声方差．仿真结果表明，在相同的导频数目下，压缩感知算法优于传统的基于内插的最小二乘信道估计算法；在多径数目先验的情况下，子空间跟踪算法优于正交匹配追踪算法．

关键词：信道估计；压缩感知；数字广播；正交频分复用

中图分类号：TN911．5