Receiver design for spread-spectrum communications with a small spread in underwater clustered multipath channels

Xiaoyan Kuai
Key Laboratory of Underwater Acoustic Communication and Marine Information Technology (School of Information Science and Engineering, Xiamen University), Ministry of Education, Xiamen 361005, China

Shengli Zhou
Department of Electrical and Computer Engineering, University of Connecticut, 371 Fairfield Way U-4157, Storrs, Connecticut 06269, USA

Zhaohui Wang
Department of Electrical and Computer Engineering, Michigan Technological University, 1400 Townsend Drive, Houghton, Michigan 49931, USA

En Cheng[a)]
Key Laboratory of Underwater Acoustic Communication and Marine Information Technology (School of Information Science and Engineering, Xiamen University), Ministry of Education, Xiamen 361005, China

(Received 30 August 2016; revised 21 December 2016; accepted 16 February 2017; published online 9 March 2017)

This paper studies a direct-sequence spread-spectrum communication system with a small spreading factor (e.g., a single digit) in underwater acoustic multipath channels. Exploiting the channel characteristics that the propagation paths can be grouped into distinct clusters, a receiver with a set of parallel branches is proposed, where per-survivor processing (PSP) is applied on each branch to deal with the signal from one cluster while explicitly treating the signals from other clusters as structured interference. As such, it overcomes a major limitation of an existing PSP based receiver [Zhou, Morozov, and Preisig, J. Acoust. Soc. Am. **133**, 2746–2754 (2013)], avoiding exponential complexity increase when the spreading factor decreases. On the first branch, joint channel estimation and interference cancellation are performed based solely on the survivor paths. On the other branches, joint channel estimation and interference cancellation are carried out based on the survivor paths and the tentatively decoded data. The bit log-likelihood ratios from different branches are combined for channel decoding. Performance results based on simulations and collected data sets validate the superior performance of the proposed receiver over the conventional RAKE receiver, which is effective only when the spreading factor is large.
[http://dx.doi.org/10.1121/1.4977747]

[KTW] Pages: 1627–1642

NOMENCLATURE

Throughout this paper, bold uppercase letters and lowercase letters are used to denote matrices and column vectors, respectively. We have the following symbol notations.

k	the symbol index used at the transmitter end
m	the symbol index used at the receiver end
L_c	the spreading code length
T_c	the chip period
T_s	the symbol period: $T_s = L_c T_c$
λ	the oversampling factor
T_δ	the sampling interval: $T_\delta = T_c/\lambda$
L_δ	the number of samples per symbol: $L_\delta = \lambda L_c$
Q	the number of clusters
N_{win}	the size of the processing window $[m - w_2, m - w_1]$ in the forward search and $[m + w_1, m + w_2]$ in the backward search: $N_{\text{win}} = w_2 - w_1 + 1$
$\mathbf{y}_m$	the vector of size $L_\delta \times 1$ for branch metric computation
$\mathbf{z}_m$	the vector of size $N_{\text{win}} L_\delta \times 1$ for channel estimation
δ_a	the step size of the Doppler scale on the search grid
δ_τ	the step size of the path delay on the search grid
j	the index of the Doppler scale on the search grid
q	the index of the path delay on the search grid

I. INTRODUCTION

Many emerging underwater applications involve wireless transmission of controlling signals and commands to autonomous underwater vehicles (AUVs) and underwater sensors. Such communication links often require high

[a)]Electronic mail: chengen@xmu.edu.cn

reliability receivers and only a few hydrophones.[2] The direct-sequence spread-spectrum (DSSS) scheme meets such a need, which is one legacy modulation scheme for low-rate robust underwater acoustic communications. One advantage of the DSSS signaling is that it is very effective in low signal-to-noise ratio (SNR) scenarios or in interference-rich environments; in contrast, the recently popular orthogonal frequency division multiplexing (OFDM) scheme is best suited for high data rate communications at high SNR scenarios.[3] Another advantage is that multiple users can be accommodated within one local network with different spreading sequences assigned to different users, thus, facilitating asynchronous code-division multiple access (CDMA). Various receiver designs have been presented in the literature for underwater acoustic DSSS or CDMA.

DSSS systems with long spreading sequences have been studied in underwater acoustic communications providing low probability interception/low probability detection,[4,5] where the symbol duration is greater than the multipath spread to have minimal intersymbol interference (ISI) effects. Two low-complexity receivers have been presented, one based on a cross correlation method,[4] and the other based on transition-code energy detectors.[5] However, the ability to suppress interference based on the cross correlation properties of the *m*-sequences is reduced significantly by the multipath spread.[6] To achieve high processing gain, the *m*-sequences of several hundreds of chips were often used.[4–7] When the ISI cannot be neglected, a decision feedback equalizer (DFE) joint with the phase-locked loop (PLL) approach has been used to overcome the severe frequency-selective distortion, where the precise symbol synchronization and channel equalization require a high SNR level.[9] A RAKE receiver was proposed to combine the multipath arrivals to enhance the signal SNR,[10] and the RAKE receiver was compared with the DFE approach.[11] Working at relative high SNR levels, the spreading factor ranges from several tens to several hundreds.[9–11] Another related work is non-coherent spread spectrum based on the *M*-ary orthogonal signaling for low-rate robust transmissions at low SNR.[7] An additional advantage of this scheme is that it allows for coherent estimation of the channel response function over the entire duration of the signal.[8] A code length of 512 is used for both $M = 2$ and $M = 4$ settings in Ref. 8.

Direct-sequence code-division multiple access (DS-CDMA) has been studied in underwater acoustic networks.[12–19] For example, DS-CDMA can be implemented with adaptive correlation and space diversity techniques.[13] The performance of DS-CDMA and multi-carrier code-division multiple access (MC-CDMA) has been evaluated and compared under different simulation scenarios.[14] Using a DS-CDMA signaling scheme, a multichannel detection receiver was proposed based on the DFE method.[15] A minimum-mean-square-error detector was further studied to reduce the complexity, where the cyclic descent method was used to find a local minimum of the cost function.[16] To provide a higher data rate, a cyclic shifting keying CDMA scheme was proposed,[17] where the passive time reversal receiver is designed to recover the information. A Kalman-filter based blind adaptive multi-user detection algorithm was proposed to effectively improve the system capacity of multi-user communication.[18] Recently, CDMA was combined with multiple-input multiple-output techniques, where simple matched filters were used for demodulation.[19] To support multi-user transmission and provide the necessary quality of performance, a spreading factor at least having double digits was often chosen for the receivers.[12–19]

The interest of this paper is on DSSS transmissions with a small spreading factor (e.g., a single digit) in underwater acoustic communications. Such a system is desirable as the data rate can be improved for DSSS when operating at a low to moderate SNR level. Per-survivor processing (PSP), where channel estimation and data detection are conducted for each survivor branch, provides a general framework for approximation of maximum likelihood sequence estimation (MLSE) whenever the presence of unknown quantities prevents the use of the classic Viterbi algorithm (VA).[20–22] Recognizing that UWA channels are sparse with distinct path arrivals, our previous publication presented a receiver that combines the PSP on the symbol level and sparse channel estimation on the chip level where the number of states in the trellis is moderate due to a large spreading factor.[1,23] However, this receiver is not suitable for a DSSS system with a small spreading factor due to the exponential complexity increase as the spreading factor decreases.

This paper exploits the fact that the time-varying propagation paths tend to be sparsely distributed in the delay-Doppler domain and the paths tend to be clustered.[24–26] Or virtually, a channel with a long multipath spread can be viewed as having several clusters, with a small multipath spread on each cluster. Developing a novel receiver for a DSSS system with a small spreading factor, the major contributions of this work are as follows.

- We propose a receiver with a set of parallel branches, where PSP processing is applied on each branch to deal with the signal from one cluster while explicitly mitigating the structured interference from other clusters. This methodology overcomes a major limitation of the existing PSP based receiver when the spreading factor is small.

Note that all existing publications on DSSS have used a spreading factor of several tens or several hundreds, rather than a single digit as in this work. When the spreading factor is large, a typical receiver (such as the RAKE receiver) involves a despreading operation and then makes decisions on a symbol by symbol basis. Instead, the PSP based receiver relies on sequence estimation, searching for the optimal data sequence along a trellis, and the despreading operation is not explicitly included as a receiver module.

- On the first branch, joint channel estimation and interference cancellation are performed based solely on the survivor paths. On the other branches, joint channel estimation and interference cancellation are carried out based on the survivor paths of the current cluster and the tentatively decoded data obtained from the first cluster. The bit log-likelihood ratios (LLRs) from different branches are combined for channel decoding. Note that interference cancellation and soft information computation for a coded

system were not considered in the existing PSP based receiver.[1]
- Simulations and data sets collected from one field experiment held off the coast of Martha's Vineyard are used to validate the superior performance of the proposed receiver over the conventional RAKE receiver, which is effective only when the spreading factor is large.

The rest of this paper is organized as follows. Section II describes the system model. Section III highlights the key idea of the proposed receiver while Sec. IV presents the signal processing details of various receiver modules. Simulation results are collected in Sec. V, while Sec. VI contains performance results using experimental data sets.

II. SYSTEM MODEL

A typical structure of one transmitted data frame is shown in Fig. 1, where a preamble signal is used for frame detection, N_{tr1} training symbols before data are used for identifying the channel clusters and initialization of forward recursion of VA, N_{tr2} training symbols after data are used for initialization of backward recursion of VA, and N_{s} data symbols carry the information bits.

Let $d[k]$ denote the information symbol to be transmitted at the kth symbol interval, which is taken from a finite alphabet $\mathcal{A}$ with size $|\mathcal{A}|$. Let $c[k;l]$ denote the time-varying spreading sequence for the kth symbol, where L_c is the spreading gain and $l=0,\ldots,L_c-1$. The scenario where a fixed spreading code $c[k;l]=c[l]$ is used for all information symbols is included as a special case. The transmitted passband signal is

$$\tilde{x}(t)=\text{Re}\left\{\left[\sum_k d[k]\sum_{l=0}^{L_c-1}c[k;l]g(t-lT_{\text{c}}-kT_{\text{s}})\right]e^{j2\pi f_c t}\right\}, \tag{1}$$

where T_c is the chip period, $T_s=L_cT_c$ is the symbol period, $g(t)$ is a pulse shaper, and f_c is the carrier frequency. To simplify the notation, define

$$\varphi(k;t):=\sum_{l=0}^{L_c-1}c(k;l)g(t-lT_c). \tag{2}$$

The transmitted signal in Eq. (1) is rewritten as

$$\tilde{x}(t)=\text{Re}\left\{\left[\sum_k d[k]\varphi(k;t-kT_s)\right]e^{j2\pi f_c t}\right\}. \tag{3}$$

We adopt a path-based channel model.[27,28] Consider a channel with N_{pa} discrete paths. Let $A_p(t)$ and $\tau_p(t)$ denote the amplitude and delay of the pth path, respectively. The channel impulse response is expressed as

$$h(t,\tau)=\sum_{p=1}^{N_{\text{pa}}}A_p(t)\delta\big(\tau-\tau_p(t)\big). \tag{4}$$

Assume that within each processing window of size N_{win}, the amplitude variation can be approximated as a constant A_p and the delay variation can be approximated by a first-order polynomial $\tau_p(t)\approx\tau_p-a_pt$. The channel impulse response in Eq. (4) is rewritten as

$$h(t,\tau)=\sum_{p=1}^{N_{\text{pa}}}A_p\delta(\tau-[\tau_p-a_pt]). \tag{5}$$

For a channel with clustered arrivals, the paths can be split into Q clusters. The channel in Eq. (5) can be reformulated as

$$h(t,\tau)=\sum_{i=1}^{Q}\sum_{p\in\Omega_i}A_{i,p}\delta(\tau-[\tau_{i,p}-a_{i,p}t]), \tag{6}$$

where Ω_i is a collection of paths within the ith cluster; $A_{i,p}$, $\tau_{i,p}$, and $a_{i,p}$ denote the amplitude, delay, and the Doppler scale of the pth path within the ith cluster, respectively.

The received signal in passband is then

$$\begin{aligned}\tilde{y}(t)&=\tilde{x}(t)\star h(t,\tau)+\tilde{w}(t)\\&=\sum_{i=1}^{Q}\sum_{p\in\Omega_i}A_{i,p}\tilde{x}((1+a_{i,p})t-\tau_{i,p})+\tilde{w}(t)\\&=\text{Re}\left\{\sum_{i=1}^{Q}\sum_{p\in\Omega_i}A_{i,p}\sum_k d[k]\varphi(k;(1+a_{i,p})t\right.\\&\qquad\left.-\tau_{i,p}-kT_{\text{s}})e^{j2\pi f_c((1+a_{i,p})t-\tau_{i,p})}\right\}+\tilde{w}(t),\end{aligned} \tag{7}$$

where $\tilde{w}(t)$ is the ambient noise. After downshifting the passband waveform to baseband and lowpass filtering (LPF), the baseband waveform $y(t):=\text{LPF}\{\tilde{y}(t)e^{-j2\pi f_c t}\}$ can be formulated as

$$\begin{aligned}y(t)=&\sum_{i=1}^{Q}\sum_{p\in\Omega_i}A_{i,p}e^{-j2\pi f_c\tau_{i,p}}\sum_k d[k]\varphi(k;(1+a_{i,p})t\\&-\tau_{i,p}-kT_s)e^{j2\pi a_{i,p}f_c t}+w(t),\end{aligned} \tag{8}$$

where $w(t)$ is the additive noise in the baseband.

The receiver samples the baseband signal $y(t)$ at a rate of $T_\delta=T_c/\lambda$, with λ being the oversampling factor, which often takes the value of $\lambda=1$ or $\lambda=2$. The discrete-time baseband sample can be expressed as

$$y[n]=y(t)|_{t=nT_\delta}. \tag{9}$$

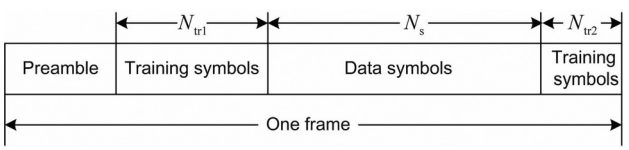

FIG. 1. A typical structure of one data frame.

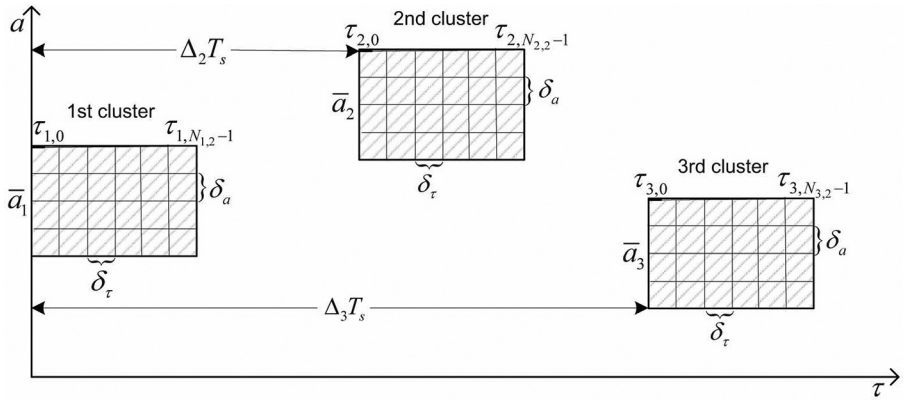

FIG. 2. The search grid on the delay-Doppler plane.

Let m denote the symbol index at the received side. For each symbol interval $\big[(m-1)T_s, mT_s\big)$, there are $L_\delta = \lambda L_c$ samples available. Define

$$n = (m-1)L_\delta + l, \quad l = 0, \ldots, L_\delta - 1. \tag{10}$$

The received signal sample $y[m;l] = y[(m-1)L_\delta + l]$ is represented as

$$y[m;l] = \sum_{i=1}^{Q} \sum_{k} d[k] \sum_{p\in\Omega_i} \xi_{i,p} f(k,m,l;a_{i,p},\tau_{i,p}) + w[m;l], \tag{11}$$

where $w[m;l]$ is the ambient noise sample and

$$\xi_{i,p} = A_{i,p} e^{-j2\pi f_c \tau_{i,p}}, \tag{12}$$

$$f(k,m,l;a,\tau) = \varphi\big(k;(1+a)((m-1)L_\delta + l) \times T_\delta - \tau - kT_s\big) e^{j2\pi a f_c ((m-1)L_\delta + l)T_\delta}. \tag{13}$$

Collect the received samples corresponding to the mth symbol interval as

$$\mathbf{y}_m = \begin{bmatrix} y[m;0] \\ \vdots \\ y[m;L_\delta - 1] \end{bmatrix}, \tag{14}$$

which contains $L_\delta = \lambda L_c$ samples. At the mth symbol time, the recursions on the trellis require the input vector $\mathbf{y}_m$.

III. KEY IDEA FOR THE PROPOSED RECEIVER

The PSP receiver in Ref. 1 is effective for underwater DSSS systems when the spreading factor is large. Let us now discuss its complexity issue when the spreading factor decreases.

For the ith cluster, define the smallest and the largest values on the Doppler scale and delay as

$$a_{i,\min} = \min_{p\in\Omega_i} a_p, \quad a_{i,\max} = \max_{p\in\Omega_i} a_p, \tag{15}$$

$$\tau_{i,\min} = \min_{p\in\Omega_i} \tau_p, \quad \tau_{i,\max} = \max_{p\in\Omega_i} \tau_p. \tag{16}$$

Each input vector $\mathbf{y}_m$ contains the contribution not only from the current symbol $d[m]$ but also from J_0 past symbols $\{d[m-J_0], \ldots, d[m-1]\}$, where

$$J_0 = \left\lceil \frac{\tau_{Q,\max} - \tau_{1,\min}}{T_s} \right\rceil = \left\lceil \frac{\tau_{Q,\max} - \tau_{1,\min}}{L_c T_c} \right\rceil. \tag{17}$$

The number of states in the trellis is thus $|\mathcal{A}|^{J_0}$. A sparse channel estimation algorithm is carried out along with each survivor path, searching a two-dimensional grid on the delay and Doppler plane.[1] Let δ_a denote the search step on the Doppler scale and δ_τ denote the search step on the path delay. The search over the ith cluster has $N_{i,1}N_{i,2}$ grid points, where

$$N_{i,1} = \left\lceil \frac{a_{i,\max} - a_{i,\min}}{\delta_a} \right\rceil, \tag{18}$$

$$N_{i,2} = \left\lceil \frac{\tau_{i,\max} - \tau_{i,\min}}{\delta_\tau} \right\rceil. \tag{19}$$

In total, there are at least $\sum_{i=1}^{Q} N_{i,1}N_{i,2}$ grid points as illustrated in Fig. 2. The orthogonal matching pursuit (OMP) algorithm was used for sparse channel estimation, and its complexity is cubic on the dictionary size.[29] The receiver complexity of the PSP receiver in Ref. 1 is shown in Table I, which includes both the branch metric computation and the channel estimation. As the spreading gain L_c decreases, the number of states increases exponentially. This will prevent the application of the PSP receiver in Ref. 1 in a practical DSSS system with a small spreading factor.

The proposed receiver in this paper overcomes such a limitation through a divide-and-conquer approach, which employs a set of parallel branches, with one PSP processor targeting one cluster. For the ith branch, the signal model is

TABLE I. Complexity comparison of the existing and the proposed PSP receivers.

	Branch metric update	Channel estimation complexity
Existing receiver (Ref. 1)	$\mathcal{O}(\lvert\mathcal{A}\rvert^{J_0})$	$\mathcal{O}\left(\lvert\mathcal{A}\rvert^{J_0}\left(\sum_{i=1}^{Q} N_{i,1}N_{i,2}\right)^3\right)$
The proposed	$\mathcal{O}\left(\sum_{i=1}^{Q} \lvert\mathcal{A}\rvert^{J_i}\right)$	$\mathcal{O}\left(\left(\sum_{i=1}^{Q} \lvert\mathcal{A}\rvert^{J_i}\right)\left(\sum_{i=1}^{Q} N_{i,1}N_{i,2}\right)^3\right)$

$$y[m;l] = \sum_k d[k] \sum_{p\in\Omega_i} \xi_{i,p} f(k,m,l;a_{i,p},\tau_{i,p}) + \sum_{j=1,j\neq i}^{Q} \sum_k d[k] \sum_{p\in\Omega_j} \xi_{j,p} f(k,m,l;a_{j,p},\tau_{j,p}) + w[m;l], \quad (20)$$

where the first part on the right side is viewed as the desirable signal from the ith cluster and the second part is the aggregated interference from other clusters. The branches are processed separately, and soft information from all the branches is then combined for channel decoding.

It is important to note that the aggregated interference from other clusters cannot be simply ignored, otherwise, the noise floor is considerably raised. Instead, the interference is explicitly modeled and mitigated in the proposed receiver along with PSP processing; the detailed algorithms will be presented in Sec. IV. Indeed, should the interference be perfectly canceled, there would be no information loss by treating the branches separately followed by maximum ratio combining. In practice, interference cancellation is not perfect, which leads to performance loss. The price paid for complexity reduction is thus the performance loss due to the existence of residual interference. On the ith branch, the trellis now has $|\mathcal{A}|^{J_i}$ states, where

$$J_i = \left\lceil \frac{\tau_{i,\max} - \tau_{i,\min}}{T_s} \right\rceil. \quad (21)$$

Due to the joint interference cancellation and channel estimation, the channel estimation complexity remains the same as the original PSP receiver because the same delay-Doppler grid points are searched.

The complexity of the proposed receiver is shown in Table I. The complexity reduction is evident by recognizing that for underwater acoustic channels with a large delay spread but having distributed clusters,

$$|\mathcal{A}|^{J_0} \gg |\mathcal{A}|^{J_1} + \cdots + |\mathcal{A}|^{J_Q}, \quad (22)$$

where the difference between the left and right sides of the equation increases as the spreading gain L_c decreases. Note also that Ref. 1 has only considered forward recursion for an uncoded system, and the module of soft information computation based on forward and backward recursions was not provided, which is needed for a coded system.

IV. RECEIVER PROCESSING

On the first branch, the PSP detection with interference cancellation is performed on the signal from the first cluster. Next, with the assistance of the tentatively decoded data from the first branch, joint channel estimation and interference cancellation are carried out on the other branches. Finally, the receiver combines the bit LLRs from all branches and feeds it to the channel decoder.

A. Processing on the first branch

For presentation clarity, we first define the two-dimensional grid on the delay and Doppler plane. The step size on the delay δ_τ is often taken as $\delta_\tau = T_c$ or $\delta_\tau = T_c/2$, depending on the settings. The step size on the Doppler scale is taken as $\delta_a = 0.1/1500$ in this paper, corresponding to a moving speed of 0.1 m/s along the propagation subject to the sound speed of 1500 m/s. For convenience, assume that $N_{i,1}$ is an odd number, e.g., $N_{i,1} = 1,5,7$ in the numerical study. For the first cluster, the tentative Doppler scales $a_{1,j}$ and delays $\tau_{1,q}$ are defined as

$$a_{1,j} = \bar{a}_1 + \left(j - \frac{N_{1,1}-1}{2}\right)\delta_a, \quad (23)$$

$$\tau_{1,q} = q\delta_\tau, \quad (24)$$

respectively, where $\bar{a}_1$ is the middle Doppler scale for the first cluster, $j = 0,\ldots,N_{1,1}-1$, and $q = 0,\ldots,N_{1,2}-1$. Define $\tilde{\tau}_{1i}$ as the inter-cluster delay between the first cluster and the ith cluster, which is the time difference between the first path of the first cluster and the first path of the ith cluster. Define an integer as

$$\Delta_i = \lfloor \tilde{\tau}_{1i}/T_s \rfloor. \quad (25)$$

Without loss of generality, assume that the tentative delays of the ith cluster start on $\Delta_i T_s$; basically, those tentative paths having the delays within the residual of $\tilde{\tau}_{1i} - \Delta T_s$ on the grid will have zero amplitudes. Define the tentative Doppler scales $a_{i,j}$ and delays $\tau_{i,q}$ as

$$a_{i,j} = \bar{a}_i + \left(j - \frac{N_{i,1}-1}{2}\right)\delta_a, \quad (26)$$

$$\tau_{i,q} = \Delta_i T_s + q\delta_\tau, \quad (27)$$

respectively, where $\bar{a}_i$ is the middle Doppler scale for the ith cluster, $i = 2,\ldots,Q$, $j = 0,\ldots,N_{i,1}-1$ and $q = 0,\ldots,N_{i,2}-1$.

With the tentative delay and Doppler values, the discrete-time input-output relationship on $\mathbf{y}_m$ can be decomposed as

$$\mathbf{y}_m = \sum_{k=m-J_1}^{m} d[k] \sum_{j=0}^{N_{1,1}-1} \sum_{q=0}^{N_{1,2}-1} h_{1,j,q} \times \begin{bmatrix} f(k,m,0;a_{1,j},q\delta_\tau) \\ \vdots \\ f(k,m,L_\delta-1;a_{1,j},q\delta_\tau) \end{bmatrix} + \sum_{i=2}^{Q} \sum_{k=m-\Delta_i-J_i}^{m-\Delta_i} d[k] \sum_{j=0}^{N_{i,1}-1} \sum_{q=0}^{N_{i,2}-1} h_{i,j,q} \times \begin{bmatrix} f(k,m-\Delta_i,0;a_{i,j},q\delta_\tau) \\ \vdots \\ f(k,m-\Delta_i,L_\delta-1;a_{i,j},q\delta_\tau) \end{bmatrix} + \boldsymbol{\eta}_m, \quad (28)$$

where $h_{i,j,q}$ are the complex amplitude affiliated with a tentative path with Doppler scale $a_{i,j}$ and delay $\tau_{i,q}$, and $\boldsymbol{\eta}_m$ consists of the ambient noise and the error caused by the channel model mismatch. One can observe from Eq. (28) that $\mathbf{y}_m$ will be affected by the following symbols:

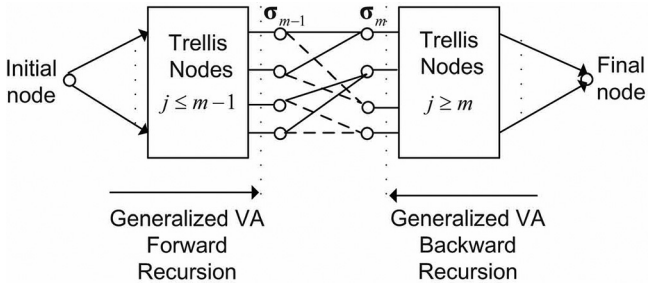

FIG. 3. The forward and backward VA framework, which is reproduced based on Fig. 1 in Ref. 30.

- The symbols $\{d[m-J_1],\ldots,d[m]\}$ due to the multipath effect on the first cluster;
- The interference symbols due to the other $(Q-1)$ clusters $\{d[m-\Delta_i-J_i],\ldots,d[m-\Delta_i]\}_{i=2}^{Q}$.

The receiver builds a trellis on the symbol level based on the first cluster. The state at the mth symbol interval is defined as

$$\boldsymbol{\sigma}_{m-1}=\big(d[m-J_1],\ldots,d[m-1]\big). \tag{29}$$

There are $|\mathcal{A}|^{J_1}$ states in the trellis at each symbol interval, denoted as $S_0,\ldots,S_{|\mathcal{A}|^{J_1}-1}$.

Based on the trellis structure, the soft information is computed depending on the forward recursion and backward recursion of VA. Figure 3 shows the framework about the soft information computation.[30] Let $\alpha(\boldsymbol{\sigma}_{m-1})$ define the accumulated log-likelihood metric from the initial state to the state $\boldsymbol{\sigma}_{m-1}$ by applying VA. For each state, the metric corresponds to the maximum metric over all paths up to that state. Denote the metric at state $\boldsymbol{\sigma}_m$ as $\alpha(\boldsymbol{\sigma}_m)$. A forward recursion relationship is generated by

$$\begin{aligned}&\alpha(\boldsymbol{\sigma}_m)=\max_{\boldsymbol{\sigma}_{m-1}}\big[\alpha(\boldsymbol{\sigma}_{m-1})+\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)\big],\\&\alpha(\mathbf{0})=0,\quad \alpha(\boldsymbol{\sigma}_0)=-\infty\ \text{ for }\boldsymbol{\sigma}_0\neq\mathbf{0},\end{aligned} \tag{30}$$

where $\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$ is the branch metric of the path connecting the state $\boldsymbol{\sigma}_{m-1}$ with state $\boldsymbol{\sigma}_m$. The state metrics for the portion of the trellis beyond the mth symbol interval can similarly be computed recursively by a backward VA starting at the last state. Denote the corresponding metrics at states $\boldsymbol{\sigma}_{m-1}$ and $\boldsymbol{\sigma}_m$ as $\beta(\boldsymbol{\sigma}_{m-1})$ and $\beta(\boldsymbol{\sigma}_m)$, respectively. Thus, the backward recursion can be similarly stated as

$$\begin{aligned}&\beta(\boldsymbol{\sigma}_{m-1})=\max_{\boldsymbol{\sigma}_m}\big[\beta(\boldsymbol{\sigma}_m)+\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)\big],\\&\beta(\mathbf{0})=0,\quad \beta(\boldsymbol{\sigma}_{N_s+J_1})=-\infty\quad\text{for }\boldsymbol{\sigma}_{N_s+J_1}\neq\mathbf{0}.\end{aligned} \tag{31}$$

To update the metrics in Eqs. (30) and (31), the key is on how to compute the branch metric $\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$ in the presence of fast varying channels and inter-cluster interference.

In the sequel, we will discuss channel estimation and branch metric computation in detail using the PSP structure in forward and backward recursions, respectively. The receiver is initialed by the PSP detection in the training symbols duration, where the training symbols are known to the receiver.

1. Forward recursion

a. Channel estimation. To update the metric information at the mth symbol interval, the unknown channel parameters at state $\boldsymbol{\sigma}_{m-1}$ should be first estimated. The joint channel estimate is carried out based on the received samples from duration $\big[(m-1-w_2)T_s,(m-w_1)T_s\big)$, where the samples are collected into a vector

$$\mathbf{z}_{m-1}=\begin{bmatrix}\mathbf{y}_{m-w_2}\\ \vdots\\ \mathbf{y}_{m-w_1}\end{bmatrix}. \tag{32}$$

Based on Eq. (28), the corresponding starting and ending position of the symbols from different clusters can be identified in each processing window, as illustrated in Fig. 4, where the channel estimation is based on $\mathbf{z}_{m-1}$ and the knowledge of the symbols on the survivor path shown in dashed-dotted lines.

The receiver constructs a dictionary of signal templates parameterized by tentative delay and Doppler parameters on the search grid. Suppose that a signal passes through a channel with only one path having unit amplitude, Doppler scale $a_{1,j}$, and delay $\tau_{1,q}$, let $\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1})$ denote the received signal vector from the first cluster associated with the survivor path at state $\boldsymbol{\sigma}_{m-1}$ and $\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1})$ denote the received interference signal propagating along the ith cluster associated with the Doppler scale $a_{i,j}$ and delay $\tau_{i,q}$. Hence, $\mathbf{z}_{m-1}$ can be represented as

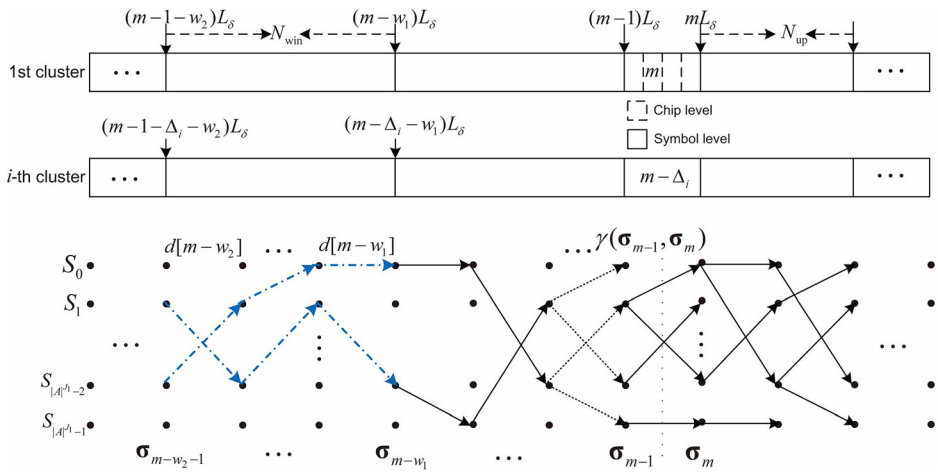

FIG. 4. (Color online) PSP trellis structure at state $\boldsymbol{\sigma}_{m-1}$ for forward recursion.

$$\mathbf{z}_{m-1} = \sum_{j=0}^{N_{1,1}-1}\sum_{q=0}^{N_{1,2}-1} h_{1,j,q}\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1}) + \sum_{i=2}^{Q}\sum_{j=0}^{N_{i,1}-1}\sum_{q=0}^{N_{i,2}-1} h_{i,j,q}\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1}) + \boldsymbol{\eta}. \quad (33)$$

For each tentative path of the first cluster with Doppler scale $a_{1,j}$ and delay $\tau_{1,j}$, the template $\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1})$ is denoted as

$$\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1}) = \sum_{k=m-w_2-J_1}^{m-w_1} d[k] \times \begin{bmatrix} f\left(k, m-w_2, 0; a_{1,j}, q\delta_\tau\right) \\ \vdots \\ f\left(k, m-w_1, L_\delta-1; a_{1,j}, q\delta_\tau\right) \end{bmatrix}. \quad (34)$$

Similarly, for each tentative path of the ith cluster with Doppler scale $a_{i,j}$ and delay $\tau_{i,q}$, the template $\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1})$ is

$$\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1}) = \sum_{k=m-w_2-\Delta_i-J_i}^{m-w_1-\Delta_i} d[k] \times \begin{bmatrix} f\left(k, m-w_2-\Delta_i, 0; a_{i,j}, q\delta_\tau\right) \\ \vdots \\ f\left(k, m-w_1-\Delta_i, L_\delta-1; a_{i,j}, q\delta_\tau\right) \end{bmatrix}, \quad (35)$$

where $f(\cdot)$ in Eqs. (34) and (35) has been defined in Eq. (13).

Define a vector of size $N_{\text{win}} L_\delta \times 1$ as

$$\mathbf{p}_{i,j,q} = \begin{cases} \mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1}), & i = 1 \\ \mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1}), & i \geq 2. \end{cases} \quad (36)$$

The observation vector $\mathbf{z}_{m-1}$ in Eq. (33) can be written as

$$\mathbf{z}_{m-1} = \underbrace{[\mathbf{p}_{1,0,0}, \cdots, \mathbf{p}_{Q,N_{Q,1}-1,N_{Q,2}-1}]}_{\mathbf{A}} \underbrace{\begin{bmatrix} h_{1,0,0} \\ \vdots \\ h_{Q,N_{Q,1}-1,N_{Q,2}-1} \end{bmatrix}}_{\mathbf{h}} + \boldsymbol{\eta} = \mathbf{A}\mathbf{h} + \boldsymbol{\eta}. \quad (37)$$

Imposing the sparsity constraint, the solution is obtained via the following optimization problem:

$$\min \|\mathbf{h}\|_0, \quad \text{subject to } \|\mathbf{z}_{m-1} - \mathbf{A}\mathbf{h}\|^2 \leq \epsilon, \quad (38)$$

where $\|\mathbf{h}\|_0$ denotes the zero norm of vector $\mathbf{h}$ and ϵ is a threshold.

We adopt the OMP algorithm to solve the optimization problem in Eq. (38), which is a suboptimal greedy algorithm based on matching and pursuit. Identifying the significant paths one at a time, the OMP algorithm solves a constrained least-squares (LS) problem at each iteration to measure the fitting error. The residual fitting error is compared with a threshold proportional to the noise power to determine the termination of the algorithm. The obtained channel estimate $\hat{\mathbf{h}}$ is used for metric computation, where there are only a limited number of nonzero entries out of its $\sum_{i=1}^{Q} N_{i,1}N_{i,2}$ elements.

b. Metric computation. The branch metric computation at the mth symbol interval is based on the samples in $\mathbf{y}_m$. For the state $\boldsymbol{\sigma}_{m-1}$,

$$\boldsymbol{\chi}_i(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m) = \sum_{j=0}^{N_{i,1}-1}\sum_{q=0}^{N_{i,2}-1} \hat{h}_{i,j,q} \sum_{k=m-J_i-\Delta_i}^{m-\Delta_i} d[k] \times \begin{bmatrix} f\left(k, m-\Delta_i, 0; a_{i,j}, q\delta_\tau\right) \\ \vdots \\ f\left(k, m-\Delta_i, L_\delta-1; a_{i,j}, q\delta_\tau\right) \end{bmatrix}, \quad (39)$$

where $\{d[k]\}_{k=m-J_i-\Delta_i}^{m-\Delta_i}$ are obtained by looking up the survivor path associated with state $\boldsymbol{\sigma}_{m-1}$, and hence $\boldsymbol{\chi}_i(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m)$ only depends on $\boldsymbol{\sigma}_{m-1}$. Define

$$\mathbf{x}(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m) = \sum_{j=0}^{N_{1,1}-1}\sum_{q=0}^{N_{1,2}-1} \hat{h}_{1,j,q} \sum_{k=m-J_1}^{m} d[k] \times \begin{bmatrix} f\left(k, m, 0; a_{1,j}, q\delta_\tau\right) \\ \vdots \\ f\left(k, m, L_\delta-1; a_{1,j}, q\delta_\tau\right) \end{bmatrix}, \quad (40)$$

where $\{d[k]\}_{k=m-J_1}^{m-1}$ and $d[m]$ are decided by the state $\boldsymbol{\sigma}_{m-1}$ and its transition to $\boldsymbol{\sigma}_m$. The branch metric between the states $\boldsymbol{\sigma}_{m-1}$ and $\boldsymbol{\sigma}_m$ is

$$\gamma(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m) \propto -\left\| \mathbf{y}_m - \mathbf{x}(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m) - \sum_{i=2}^{Q} \hat{\boldsymbol{\chi}}_i(\boldsymbol{\sigma}_{m-1}, \boldsymbol{\sigma}_m) \right\|^2, \quad (41)$$

which is used in Eq. (30) for metrics update.

c. Template approximation. The computation of the dictionary entries $\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1})$ or $\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1})$ can be simplified if the Doppler search range on a cluster ($N_{1,1}\delta_a$ or $N_{i,1}\delta_a$) is small. In such a case, the waveform compression/dilation on the pulse shaping filter can be simplified as $g((1+a_{i,j})t) \approx g((1+\bar{a}_i)t)$, where $\bar{a}_i$ is the middle Doppler scale for the ith cluster. Hence, $f(k, m, l; a, \tau)$ in Eq. (13) can be approximated as

$$\tilde{f}(k, m, l; a_{ij}, \tau) = \varphi\left(k; (1+\bar{a}_i)((m-1)L_\delta + l)T_\delta - \tau - kT_s\right) e^{j2\pi a_{ij} f_c ((m-1)L_\delta + l)T_\delta}. \quad (42)$$

The dictionary entries $\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_{m-1})$ and $\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_{m-1})$ can be obtained by plugging $f(k, m, l; a, \tau) \approx \tilde{f}(k, m, l; a, \tau)$ into Eqs. (34) and (35).

In the special case with $\bar{a}_i = 0$, we have

$$\tilde{f}(k,m,l;a_{ij},\tau)=\varphi\big(k;((m-1)L_\delta+l)T_\delta - \tau - kT_s\big)e^{j2\pi a_{ij}f_c((m-1)L_\delta+l)T_\delta}, \quad (43)$$

where the Doppler effect is only modeled through the frequency shifts and the compression/dilation effect on the waveform is ignored. This template generation is considerably simplified, which corresponds to the narrowband assumption as discussed in Ref. 1.

2. Backward recursion

Consider the metric update from the state $\boldsymbol{\sigma}_m$ to the state $\boldsymbol{\sigma}_{m-1}$ in the backward recursion, where the channel estimation is carried out on the samples in the duration $\big[(m+w_1-1)T_s,(m+w_2)T_s\big)$. We can collect these signal samples into a vector $\mathbf{z}_m$ with size $N_{\text{win}}L_\delta\times 1$ as

$$\mathbf{z}_m=\begin{bmatrix}\mathbf{y}_{m+w_1}\\ \vdots \\ \mathbf{y}_{m+w_2}\end{bmatrix}, \quad (44)$$

which can be represented as Eq. (33) but with the following changes.

- $\mathbf{f}_{1,j,q}(\boldsymbol{\sigma}_m)$ denotes the received signal vector associated with the symbols on the backward survivor path at state $\boldsymbol{\sigma}_m$ from the first cluster, which depends on the symbols $\{d[m+w_1-J_1],\ldots,d[m+w_2]\}$, as the dashed-dotted lines shown in Fig. 5;
- $\mathbf{g}_{i,j,q}(\boldsymbol{\sigma}_m)$ denotes the interference signal vector on the ith cluster, where the involved symbols are $\{d[m+w_1-\Delta_i-J_i],\ldots,d[m+w_2-\Delta_i]\}$. The symbols $\{d[m+1-J_1,\ldots,d[m+w_2-\Delta_i]\}$ are obtained from the survivor path of state $\boldsymbol{\sigma}_m$ from the backward recursion, while the missing symbols $\{d[m+w_1-\Delta_i-J_i],\ldots,d[m-J_1]\}$ are borrowed from the survivor path of state $\boldsymbol{\sigma}_m$ from the forward recursion.

The OMP algorithm is adopted for the channel estimation of different clusters. For the branch metric computation in Eq. (41), $\boldsymbol{\chi}(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$ is constructed by Eq. (39), where the interference symbols are obtained by looking up the forward recursion survivor path at state $\boldsymbol{\sigma}_m$, and $\mathbf{x}(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$ is constructed from Eq. (40). After obtaining the branch metric $\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$, the accumulated metrics can be updated based on Eq. (31).

3. Soft-output information

The lth bit taken by the mth symbol is denoted as $b_m[l]$, where

$$b_m[l]\in\{0,1\},\quad l=1,\ldots,\log_2|\mathcal{A}|. \quad (45)$$

To obtain the soft-output information, we need to find the LLR of the lth bit defined as $\text{LLR}(b_m[l])$,

$$\text{LLR}(b_m[l])=\ln\frac{\sum_{d[m]\in\mathcal{A}}P\Big(b_m[l]=1|\{\mathbf{y}_m\}_{m=1}^{N_s}\Big)}{\sum_{d[m]\in\mathcal{A}}P\Big(b_m[l]=0|\{\mathbf{y}_m\}_{m=1}^{N_s}\Big)}. \quad (46)$$

Here, the max-log approximation $\ln(e^x+e^y)\approx\max(x,y)$ is adopted to obtain[30]

$$\text{LLR}(b_m[l])\propto\max_{\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m:b_m[l]=1}\big[\alpha(\boldsymbol{\sigma}_{m-1})+\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)+\beta(\boldsymbol{\sigma}_m)\big]-\max_{\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m:b_m[l]=0}\big[\alpha(\boldsymbol{\sigma}_{m-1})+\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)+\beta(\boldsymbol{\sigma}_m)\big]. \quad (47)$$

Substitute the updated metrics $\alpha(\boldsymbol{\sigma}_{m-1})$, $\gamma(\boldsymbol{\sigma}_{m-1},\boldsymbol{\sigma}_m)$, and $\beta(\boldsymbol{\sigma}_m)$ into Eq. (47), $\text{LLR}(b_m[l])$ is computed.

The soft information is de-interleaved and passed to the channel decoding. The decoded bits are encoded and modulated to form the tentatively decoded data for the PSP detection on other branches.

B. Multi-cluster combining

On the ith branch, the trellis is constructed based on the signal from the ith cluster with $|\mathcal{A}|^{J_i}$ states. The processing is similar to that on the first branch with the following differences.

- The search grid on the delay is shifted so that the delays for the ith cluster start at zero. Define $\Delta_{ij}=\Delta_j-\Delta_i$. The search grid for the jth cluster starts at $\Delta_{ij}T_s=(\Delta_j-\Delta_i)T_s$. Note that Δ_{ij} could be positive or negative.
- The interference symbols due to the other $(Q-1)$ clusters are assumed available based on the tentatively decoded data, instead of looking into the survivor paths.

Finally, the LLRs from all branches are combined as

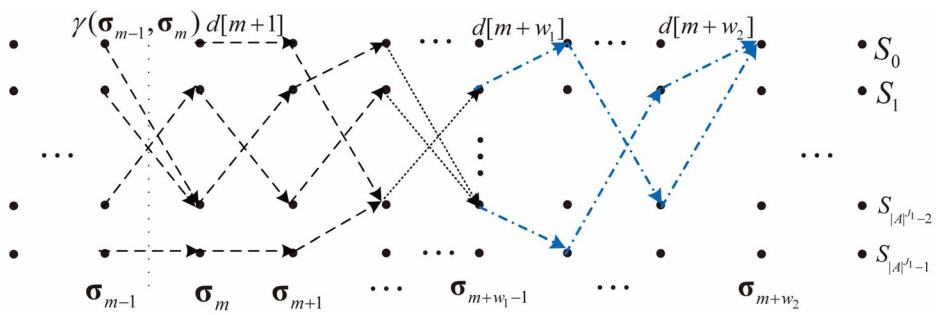

FIG. 5. (Color online) PSP trellis structure at state $\boldsymbol{\sigma}_m$ for backward recursion.

$$\mathrm{LLR}_{\mathrm{comb}}(b_m[l]) = \sum_{i=1}^{Q} \mathrm{LLR}_i(b_m[l]), \tag{48}$$

which is then de-interleaved and passed to the channel decoder.

C. Implementation issues

1. Identification of channel clusters

The proposed receiver needs to set up the search grid on the delay and Doppler plane as illustrated in Fig. 2. Identifying the channel clusters is done prior to data decoding. Here it is done based on the training signal prior to data as illustrated in Fig. 1. Cluster identification is not unique to this paper, and a description is provided here for completeness of the receiver design.

Denote $x_{\mathrm{tr1}}(t)$ as the transmitted training waveform in baseband before the data and T_0 as the time duration of the training signal. Taking $x_{\mathrm{tr1}}(t)$ as a local template, a correlator is used to calculate the inner product between the training signal template and the received waveform for joint delay and Doppler scale estimation

$$r(a,\tau) = \int_0^{T_0} x_{\mathrm{tr1}}^*((1+a)t) y(t+\tau) e^{-j2\pi a f_c t}\, dt, \tag{49}$$

after which its squared modulus is taken as

$$z(a,\tau) = |r(a,\tau)|^2, \quad \tau \in \mathcal{T},\ a \in \mathcal{D} \tag{50}$$

where $\mathcal{T}$ is a time arrival in which the training signal is expected to arrive and $\mathcal{D}$ is the search range of the Doppler scale. The step sizes for the Doppler scale and the delay can have the same or smaller values as those used in the channel estimation. The following processing is used to obtain an estimate of the delay and Doppler scale of each path and then classify the channel paths into different clusters.

The algorithm starts by estimating the delay and Doppler scale associated to the strongest path as follows:

$$(\hat{a}_1, \hat{\tau}_1) = \arg \max_{t\in\mathcal{T}; a\in\mathcal{D}} z(a,\tau), \tag{51}$$

where $\hat{\tau}_1$ and $\hat{a}_1$ represent the estimated delay and Doppler scale of the strongest path, respectively. After identifying the path, it is convenient that a window of length $2T_{\mathrm{corr}}$ centered at the correlation peak is selected to include a few sidelobes,[31] e.g., the parameter T_{corr} is chosen around 0.2 ms for our experiment setting. Define the interval around estimated delay as $\mathcal{Z}_1 = (\hat{\tau}_1 - T_{\mathrm{corr}}, \hat{\tau}_1 + T_{\mathrm{corr}})$.

Continue this process to the pth step and denote $\mathcal{I}_p$ as a set of subintervals of $\mathcal{T}$ around the values of delay already identified through the $(p-1)$th step

$$\mathcal{I}_p = \{\mathcal{Z}_1 \cup \mathcal{Z}_2 \cup \cdots \cup \mathcal{Z}_{p-1}\}. \tag{52}$$

The delay and Doppler scale associated to the pth path is identified by looking for the other local maxima of $z(a,\tau)$, using the following iterative approach:

$$\{\hat{a}_p, \hat{\tau}_p\} = \arg \max_{\substack{a\in D;\tau\in T \\ \tau\notin I_p}} z(a,\tau), \tag{53}$$

where $\mathcal{I}_p$ is excluded from the search. Finally, the search is stopped when the peak is less than the pre-specified threshold (e.g., −10 dB relative to the strongest path) and the paths are orderly identified along with their delays.

Next, the path delays and Doppler scales are assumed to be only slowly varying during a data burst, and hence they can be used for cluster formation. Set the first path as the initial delay of the first cluster, denoted as $\hat{\tau}_{1,0}$. The first cluster Ω_1 then contains all paths that have a delay

$$\hat{\tau}_p < \hat{\tau}_{1,0} + T_{\mathrm{cluster}}, \tag{54}$$

where T_{cluster} is chosen to ensure that the number of states in the trellis for each cluster is moderate; when dealing with the experimental data sets, we set $T_{\mathrm{cluster}} = 3T_s$ so that $J_i \le 3$. Excluding the paths in the first cluster, the same procedure is applied to find the second cluster. Continue this process until all the paths are used. We consider the Doppler scale of the path with maximum correlation peak within Ω_i as the main Doppler scale of ith cluster, denoted as $\hat{\bar{a}}_i$.

2. Multiple-resampling front end

The center Doppler scales for different clusters could be quite different as illustrated in Fig. 2. This motivates the use of a multiple-resampling front end as used in Refs. 32 and 33. Specifically, a separate resampling operation is performed on the incoming signal $\tilde{y}(t)$ before further processing. On the ith branch, the resampling factor is $\hat{\bar{a}}_i$ and the resampled signal is

$$\tilde{y}_i(t) = \tilde{y}\left(\frac{t}{1+\hat{\bar{a}}_i}\right). \tag{55}$$

The signal $\tilde{y}_i(t)$ is being shifted to the baseband and its baseband samples are processed for data demodulation. Clearly, on the ith branch, the search on the Doppler scale for the ith cluster is now centered around zero. With a moderate Doppler spread, the narrowband assumption can then be adopted to simplify the template calculation along the trellis paths for the desired signal, as shown in Eq. (43). For the jth cluster on the ith branch, the search grid on the Doppler scale is now centered at $(\hat{\bar{a}}_j - \hat{\bar{a}}_i)/(1+\hat{\bar{a}}_i) \approx \hat{\bar{a}}_j - \hat{\bar{a}}_i$.

We reiterate that the proposed receiver is functional without the multiple-resampling front end. The multiple-resampling front end can be incorporated to speed up the template construction in both channel estimation and branch metric computation.

3. Reducing the number of channel updates

On the ith branch, the PSP receiver has $|\mathcal{A}|^{J_i}$ states. When $w_1 = 1$, the OMP algorithm will be called $|\mathcal{A}|^{J_i}$ times

at each symbol time. To reduce the complexity, we choose $w_1 > 1$. The survivor paths for different states might be identical during the channel estimation window $[m - w_2, m - w_1]$, and hence the number of channel estimations will be reduced by avoiding the duplicative effort. Further, the receiver can update the channel estimation every N_{up} symbols to reduce the complexity. Combining the above two methods, the channel parameters are viewed approximately constant in the duration $(N_{\text{win}} + N_{\text{up}} + w_1)T_s$. The complexity reduction comes at the cost of reduced accuracy in tracking the channel variations.

4. PSP receiver without interference cancellation

The receiver presented so far explicitly incorporates interference cancellation. One variation is to ignore the interference from other clusters. On the ith branch, the channel estimation complexity now reduces from $\mathcal{O}((\sum_{i=1}^{Q} N_{i,1}N_{i,2})^3)$ to $\mathcal{O}((N_{i,1}N_{i,2})^3)$. For the whole receiver with Q branches, the channel estimation complexity reduces from $\mathcal{O}((\sum_{i=1}^{Q} |\mathcal{A}|^{J_i}) \times (\sum_{i=1}^{Q} N_{i,1}N_{i,2})^3)$ to $\mathcal{O}(\sum_{i=1}^{Q} |\mathcal{A}|^{J_i}(N_{i,1}N_{i,2})^3)$. This complexity reduction will lead to a considerable performance loss to be illustrated in the numerical results.

V. SIMULATION RESULTS

The DSSS system parameters are listed as Table II, and are used in the field experiment as described in Sec. VI: center frequency $f_c = 13$ kHz, bandwidth $B = 4.883$ kHz, spreading factor $L_c = 4$, chip interval $T_c = 1/B = 0.2$ ms, and symbol interval $T_s = L_c T_c = 0.8$ ms. The raised-cosine filter with a roll-off factor 0.4 is used for pulse shaping. For each frame, 486 symbols are used, where we set $N_{\text{tr1}} = 100$ training symbols before the data are used to the initial channel estimation for forward recursion and $N_{\text{tr2}} = 50$ training symbols after the data are used to the initial estimation for the backward recursion. $N_s = 336$ data symbols are encoded with a rate-3/4 convolution code and modulated with binary phase-shift keying (BPSK).

The channel estimation window has the following parameters: $N_{\text{win}} = 50$, $w_1 = 12$, $w_2 = N_{\text{win}} + w_1 - 1$, and $N_{\text{up}} = 20$. For all the simulation results, at least ten frame errors are collected for each SNR level. The bit error rate (BER) after Viterbi decoding will be used as the performance metric.

In the sequel, we first illustrate the performance and complexity tradeoff by comparing the proposed receiver with the existing receiver in Ref. 1 using a baseband channel. Then, the performance of the proposed receiver is investigated in time-invariant and -varying multipath channels with two clusters, where the data frames are generated in the passband. Finally, the performance of the proposed receiver is studied in channels with three clusters, where the last cluster is considerably weaker than the first two clusters.

TABLE II. The system parameters.

Center frequency	f_c	13 kHz
Bandwidth	B	4.883 kHz
Spreading factor	L_c	4
Symbol duration	T_s	0.8 ms
Number of data symbols	N_s	336
Number of training sequence1	N_{tr1}	100
Number of training sequence2	N_{tr2}	50
Convolution code rate	r	3/4

A. Performance and complexity tradeoff using a baseband channel

The system parameters are the same in Table II except that the transmitted and the received sequences are directly generated at the baseband without oversampling ($T_\delta = T_c$). The discrete-time channel is time-invariant and has 24 taps. For simplicity, all the channel taps are independent and identically distributed following the complex Gaussian distribution. We consider three scenarios.

- For the existing receiver[1] treating the channel as having 24 taps, each symbol will introduce interference to the next $J_0 = \lceil (24-1)T_c/T_s \rceil = 6$ symbols. The number of states in the trellis is $2^{J_0} = 64$.
- For the proposed receiver treating the channel as having two clusters of 12 consecutive taps, we have $J_1 = J_2 = \lceil (12-1)T_c/T_s \rceil = 3$, and the total number of states on the two branches is $2^{J_1} + 2^{J_2} = 16$.
- For the proposed receiver treating the channel as having three clusters of eight consecutive taps, we have $J_1 = J_2 = J_3 = \lceil (8-1)T_c/T_s \rceil = 2$. The total number of states on the three branches is $2^{J_1} + 2^{J_2} + 2^{J_3} = 12$.

Figure 6 shows the BER performance of the receivers dividing the channel into different numbers of clusters. At the BER of 10^{-4}, the performance of the proposed receiver with 2 clusters is about 0.8 dB worse than the existing receiver having 1 cluster, while reducing the total number of states from 64 to 16. The performance of the proposed receiver having 3 clusters is about 0.6 dB worse than the receiver having 2 clusters, while reducing the total number of states from 16 to 12. Note that this is a favorable setting for the existing receiver keeping all the channel taps in one cluster because there is no gap between adjacent clusters. This example is provided to

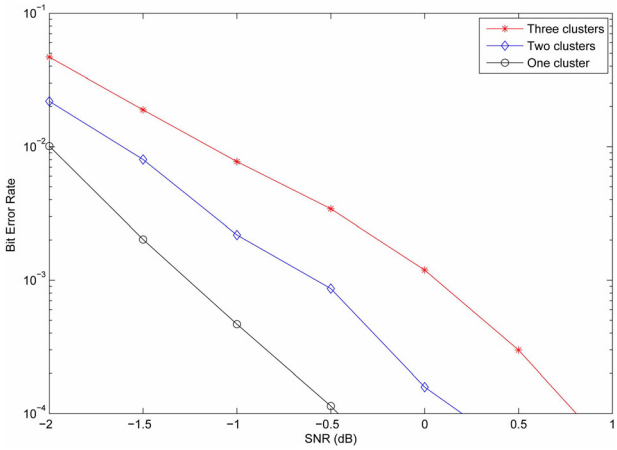

FIG. 6. (Color online) BER performance of the receivers dividing the channel taps into different numbers of clusters.

illustrate the performance and complexity tradeoff by viewing a long channel as having several virtual clusters.

B. Time-invariant channels with two clusters

The simulated data frames are now generated in the passband, according to the mathematical descriptions in Eqs. (1) and (7). We assume that the channel has two clusters and each cluster consists of five discrete paths. Within each cluster, the delays are randomly generated with a uniform distribution. Amplitudes of the paths are Rayleigh distributed. We assume that each cluster has a maximum delay spread $\tau_{\max} = 2$ ms and hence $J_1 = J_2 = \lceil \tau_{\max}/T_s \rceil = 3$. The number of states is $|\mathcal{A}|^{J_1} = |\mathcal{A}|^{J_2} = 8$ with BPSK signaling. We set the average power ratio of the signals arriving along the first cluster to those arriving along the second cluster as 3 dB. The inter-arrival time of the two clusters follows a uniform distribution with $\Delta_2 \sim \mathcal{U}[4, 6]$.

To examine the performance of the proposed receivers in a time-invariant clustered channel, the Doppler rate of each path is set to zero. Chip level sampling with $\lambda = 1$ is used. The search grid on the delay-Doppler plane has $N_{i,1} = 1$ and $N_{i,2} = J_i L_c = 12$, $i = 1,2$, where $\delta_\tau = T_c$. We compare the BER performance of different receivers:

(1) PSP based receiver with interference cancellation.
(2) PSP based receiver without interference cancellation, as described in Sec. IV C 4.
(3) The RAKE receiver: It combines four strongest paths with the maximum ratio combining (MRC) processing.

Two benchmark receivers are added:

(1) Benchmark receiver interference symbols known: PSP based receiver with interference cancellation is carried on the received signal, where the interference symbols are known to the receiver but the channel parameters of interference signal are estimated.
(2) Benchmark receiver interference signal known: PSP based receiver without interference cancellation is carried on the received signal, where the interference signal is known to the receiver and it is subtracted from the received signal before PSP detection. This is not a realistic receiver, but its performance is included here as a performance bound.

Figure 7 depicts the performance of the proposed PSP based receivers, the RAKE receiver, and the benchmark receivers. From Fig. 7, one can observe that:

- Both of the PSP based receivers have significant performance improvements compared with the RAKE receiver. The proposed PSP based receiver with interference mitigation outperforms the receiver without interference cancellation.
- Compared with the benchmark receiver where the interference symbols are known, the proposed receiver with interference mitigation suffers from around 0.5 dB performance loss at the BER level of 10^{-4}. Meanwhile, the performance gap is about 1 dB compared with the benchmark receiver where the interference signal is known.

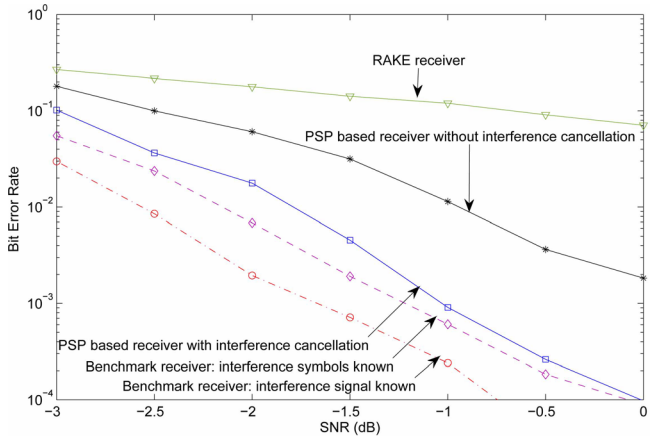

FIG. 7. (Color online) BER performance of different receivers in time-invariant channels.

C. Time-varying channels with two clusters

The path delays are generated as in Sec. V B but each path has a different Doppler scale. We generate five paths with distinct Doppler speeds $v = ac$ for each cluster, where in the first cluster the Doppler speed is drawn from a uniform distribution with a positive mean $\bar{v}_1 = 1$ m/s and the second cluster has a negative mean $\bar{v}_2 = -1$ m/s. The standard deviation is $\sigma_v = 0.12$ m/s. Figure 8 shows one estimated channel scattering function. The multiple resampling front end is adopted. We compare the BER performance of different receivers in this scenario. The RAKE receiver combines the paths that are recognized by OMP algorithm using MRC processing. The symbols for channel estimation are assumed known in each update, which might not be achieved in a practical scenario with decision-directed training. The benchmark receiver with interference symbols known is provided to compare with the proposed PSP based receivers. All the receivers use the approximate templates generated by Eq. (42). The oversampling factor is $\lambda = 2$, and the search grid on the delay-Doppler plane has $N_{i,1} = 5$ and $N_{i,2} = \lambda J_i L_c = 24$, $i = 1,2$, with $\delta_a = 0.1/1500$ and $\delta_\tau = T_c/2$.

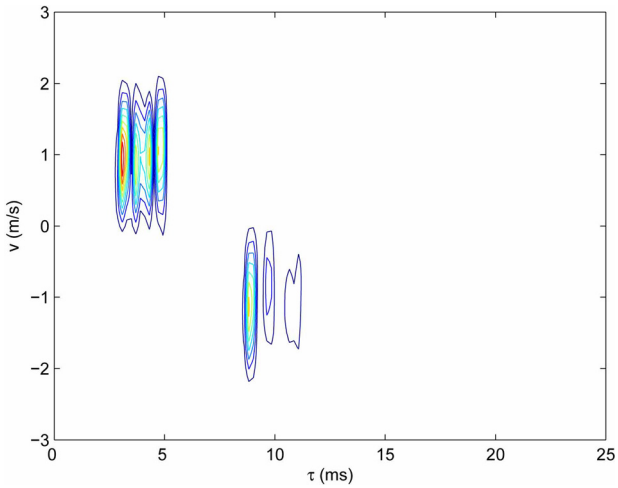

FIG. 8. (Color online) The estimated scattering function of the simulated time-varying channels.

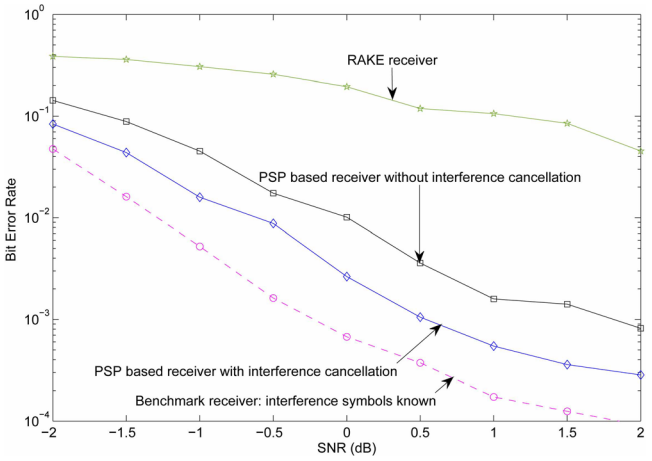

FIG. 9. (Color online) BER performance of different receivers in time-varying channels.

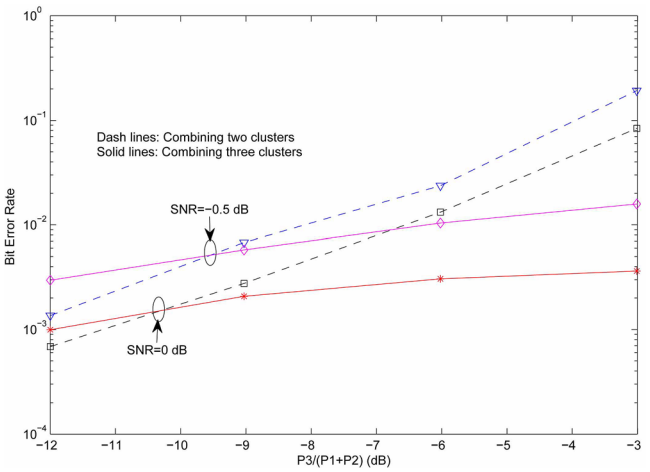

FIG. 10. (Color online) BER performance as a function of the power ratio $P_3/(P_1+P_2)$.

Figure 9 shows the performance of different receivers with/without interference mitigation. One can again observe that both of the PSP based receivers outperform the RAKE receiver. Meanwhile, our proposed receiver with interference mitigation has better performance than the receiver without interference mitigation. There is about 1 dB performance gap between the benchmark receiver and the proposed receiver with interference cancellation.

D. Time invariance channels with three clusters

We now investigate the BER performance of the proposed receiver in channels with three clusters, but the third cluster is considerably weaker than the first two clusters. For simplicity, the channels are time invariant, and the path delays and amplitudes in each cluster are generated as described in Sec. V B. The inter-arrival time of inter-cluster follows a uniform distribution with $\Delta_2 \sim \mathcal{U}[4,6]$ and $\Delta_3 \sim \mathcal{U}[8,12]$. Let P_1,P_2,P_3 denote the average power of the signals arriving along the first, second, and third clusters, respectively. We set $P_1 = P_2$ and vary the power ratio $P_3/(P_1+P_2)$.

Figure 10 shows the BER performance of the proposed receivers combining two or three clusters with a varying power ratio. At SNR $= -0.5$ dB, the performance of the receiver combining two clusters outperforms that combining three clusters when P_3 is -9.5 dB less than (P_1+P_2).

At SNR $= 0$ dB, the performance of the receiver combining two clusters outperforms that combining three clusters when P_3 is -10.1 dB less than (P_1+P_2). Hence, when one cluster is considerably weaker than the rest, it does not contribute positively to the receiver performance, and the corresponding branch can be turned off. The cutoff threshold, however, may not be available analytically.

VI. EXPERIMENTAL RESULTS

We will use some data sets collected from the *mobile acoustic communication experiment* (MACE10), which was carried out off the coast of Martha's Vineyard, MA, June 2010. Some specific descriptions about the experimental environment can be found in Ref. 34, including the sound speed profile and the SNR variations with the range. For our signals, the bandwidth was $B = 4.883$ kHz, the carrier frequency was 13 kHz, and the sampling frequency was 39 062.5 Hz. The receiving array was stationary, while the transmitter was towed slowly away from the receiver and then towed back, at a speed around 1 m/s. The relative distance of transmitter and receiver changed from 500 m to 4.5 km. In this paper, we consider the data collected in the first tow. Figure 11 shows the estimated relative speed within the duration of this tow, which reflects the experimental settings. There were 31 transmissions with 3 blocks in each transmission. The signal was transmitted from a depth of about 80 m and received by a 12-element array. A total of 1116 blocks were received.

An m-sequence block of length 1023 was transmitted, with the chip rate 4.883 kHz. We use this data set as a coded BPSK transmission with time-varying spreading sequences. We set $L_c = 4$, and hence there are a total of 255 symbols, where we set $N_{\text{tr1}} = 100$, $N_{\text{tr2}} = 55$, and $N_{\text{s}} = 100$. Corresponding to each cluster, the trellis has $2^3 = 8$ states. The convolutional code with rate 3/4 was used, and the data rate is

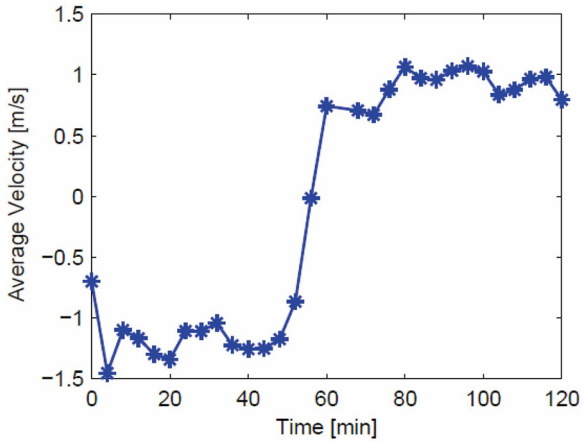

FIG. 11. (Color online) The estimated moving speed in tow 1 from the MACE10 experiment.

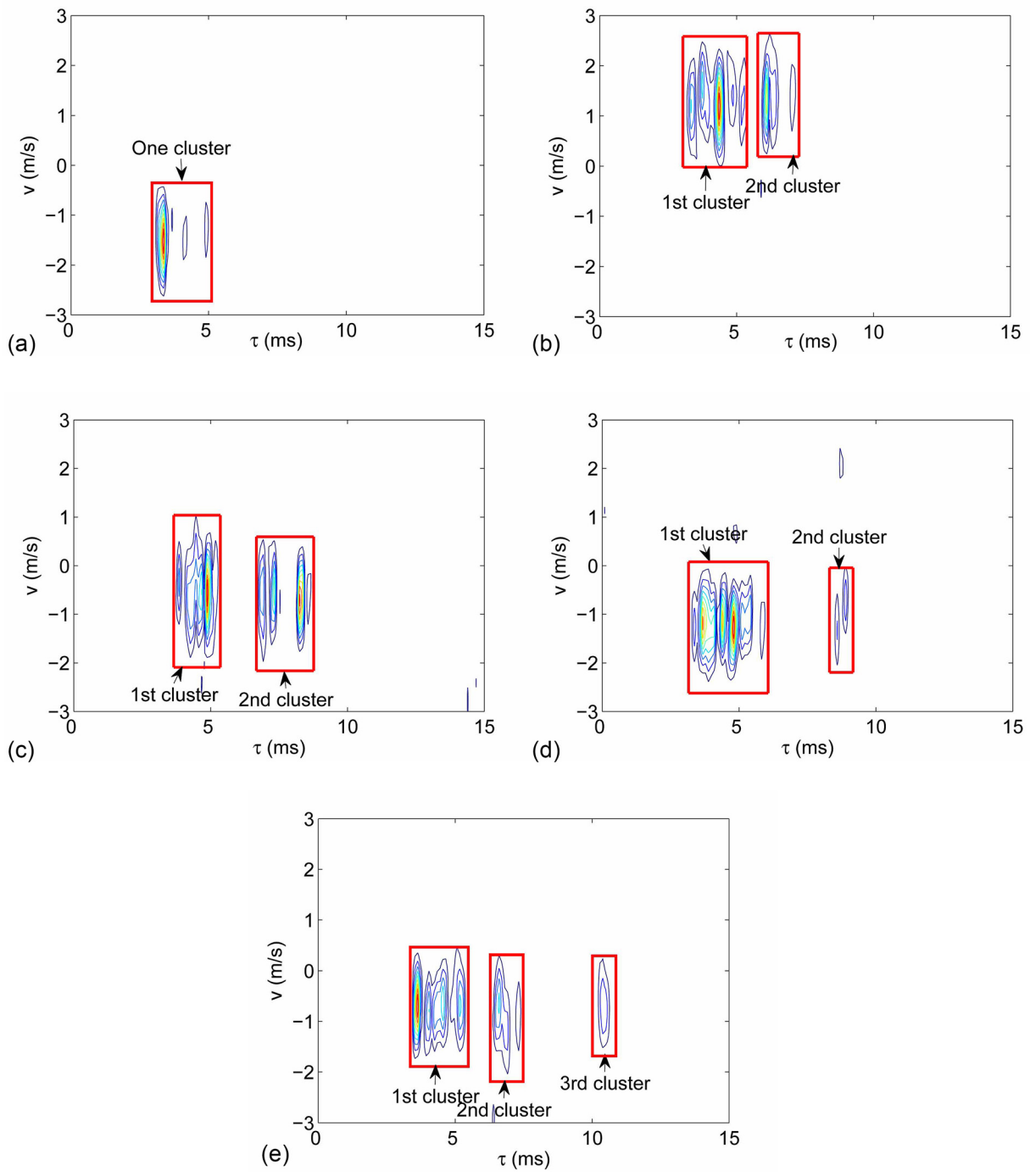

FIG. 12. (Color online) The estimated scattering function of the channels in the experiment where (a) one-cluster channels; (b) two-cluster channels: $\hat{\tilde{a}}_1 > 0$, $\hat{\tilde{a}}_2 > 0$, and $|\hat{v}_1 - \hat{v}_2| < 0.2\,\mathrm{m/s}$; (c) two-cluster channels: $\hat{\tilde{a}}_1 < 0$, $\hat{\tilde{a}}_2 < 0$, and $|\hat{v}_1 - \hat{v}_2| < 0.2\,\mathrm{m/s}$; (d) two-cluster channels: $|\hat{v}_1 - \hat{v}_2| > 0.2\,\mathrm{m/s}$; (e) three-cluster channels.

$$R = \frac{3}{4}\frac{N_s \log_2 Q}{T_{N_{\mathrm{tr1}}} + T_{N_{\mathrm{tr2}}} + T_{N_s}} \simeq 367.65\ \mathrm{bps}. \tag{56}$$

If we just consider the data sequence interval, the data rate would be

$$R = \frac{3}{4}\frac{N_s \log_2 Q}{T_{N_s}} \simeq 937.5\ \mathrm{bps}. \tag{57}$$

Now we classify the received blocks based on the number of clusters and the Doppler scale of each cluster. The receiver carries out the channel classification method in Sec. IV C on the received signal block. For multipath clustering of each block, we can obtain the number of clusters Q and Doppler scale value $\hat{\tilde{a}}_i$ of each cluster. Then the relative speed is estimated as $\hat{v}_i = \hat{\tilde{a}}_i c$, using a normal sound speed of $c = 1500\,\mathrm{m/s}$. Based on the estimated parameters, the 802 good received blocks are separated as the following groups:

(1) The channel has one cluster;
(2) The channel has two clusters, where $\hat{\tilde{a}}_1 > 0$, $\hat{\tilde{a}}_2 > 0$, and $|\hat{v}_1 - \hat{v}_2| < 0.2\,\mathrm{m/s}$;

(3) The channel has two clusters, where $\hat{\bar{a}}_1 < 0$, $\hat{\bar{a}}_2 < 0$, and $|\hat{v}_1 - \hat{v}_2| < 0.2\,\mathrm{m/s}$;
(4) The channel has two clusters, where $|\hat{v}_1 - \hat{v}_2| \geq 0.2\,\mathrm{m/s}$;
(5) The channel has three clusters.

There are 133, 186, 226, 180, 77 blocks for each group, respectively. Figures 12(a)–12(e) show the estimated channel scattering function for each group situation.

In the sequel, the decoding performance of different receivers will be tested with the grouped blocks in three different cases. The SNR of the original received blocks is larger than 15 dB; we do not consider the ambient noise and add artificially generated Gaussian noise to the real data set to have a variable SNR. All the receivers compensate the main Doppler effect by a resampling operation for each block and adopt approximate correlation templates as Eq. (42). For the sparse channel estimation, the step size for the delay estimation is $\delta_\tau = 1/(2B)$, and the step size for the Doppler scale estimation is $\delta_a = 0.1/1500$.

A. Test case 1

In this test case, we consider receivers to decode the 412 received blocks where the multipath channel is divided into 2 clusters and the difference of the estimated Doppler speed between 2 clusters is less than 0.2 m/s. After resampling on the received blocks, we assume the Doppler scaling effect on both clusters is ignored, i.e., $\bar{a}_1 = \bar{a}_2 = 0$, and the search grid has $N_{i,1} = 1$ and $N_{i,2} = 24$. Figure 13 shows the BER performance for the PSP based receiver with interference cancellation, the PSP based receiver without interference cancellation, and the RAKE receiver. The performance of decoding 133 received blocks with 1-cluster channels is also included as a benchmark. One can observe that: (i) The proposed PSP based receivers outperform the RAKE receiver considerably. (ii) The proposed PSP based receiver with interference cancellation has better performance than the receiver without interference cancellation. (iii) Comparing the benchmark receiver with the proposed receiver with interference cancellation, the performance gap is about 1 dB.

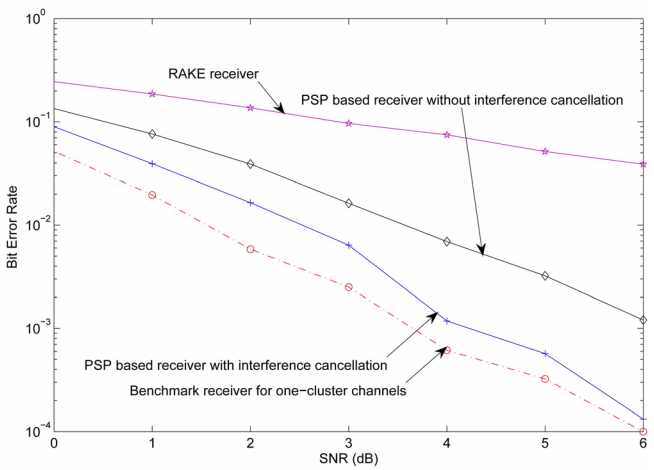

FIG. 13. (Color online) BER performance of different receivers in two-cluster channels where $|\hat{v}_1 - \hat{v}_2| < 0.2$.

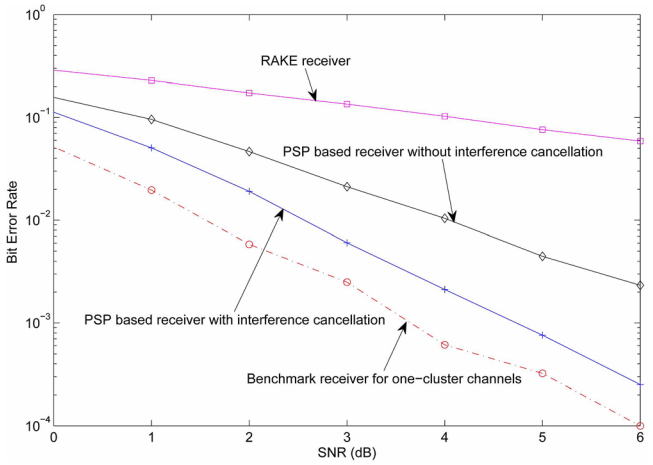

FIG. 14. (Color online) BER performance of different receivers in two-cluster channels where $|\hat{v}_1 - \hat{v}_2| \geq 0.2\,\mathrm{m/s}$.

B. Test case 2

In this test case, we consider receivers to decode the 180 received blocks where the multipath channel is divided into 2 clusters and the difference of the estimated Doppler speed between 2 clusters is large or equal to 0.2 m/s. The multiple-resampling front end is used, and the search grid has $N_{i,1} = 7$ and $N_{i,2} = 24$, $i = 1,2$. Figure 14 shows the BER performance of different receivers. From Fig. 14, one can again observe that the proposed PSP based receivers outperform the RAKE receiver. Moreover, the proposed receiver with interference cancellation has better performance than the receiver without interference cancellation. There is also a performance gap between the proposed receiver and the benchmark receiver about 1 dB.

C. Test case 3

In this test case, we test the performance of different receivers on the blocks with three clusters. The multiple-resampling front end is used and the search grid has $N_{i,1} = 7$ and $N_{i,2} = 24$, $i = 1,2,3$. Figure 15 shows the BER performance of different receivers corresponding to different

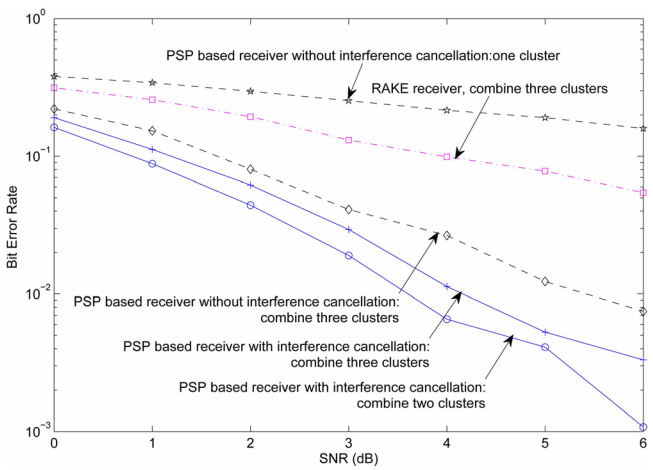

FIG. 15. (Color online) BER performance of different receivers in three-cluster channels.

settings: (i) Treating the channel as having three clusters, and (ii) considering the first and second clusters as useful channels and treating the third cluster as additive noise, and (iii) considering the first cluster as the useful channel, while treating the second and third clusters as additive noise. From Fig. 15, one can see:

- If we just consider the first cluster and treat the other clusters as additive noise [scheme (iii)], the proposed receiver does not work well;
- The performance of the proposed receiver with interference cancellation is improved obviously when the first and second clusters are considered as the useful channel [scheme (ii)];
- When the channels are further divided into three clusters [scheme (i)], the performance of the proposed receiver combing three clusters has no improvement over, and actually is slightly worse than, the performance of that combining the first two clusters. This is because the third cluster appears to be much weaker than the first two clusters. For the 77 blocks used, we treat the data as known and obtain the channel profile via cross correlation, based on which we estimate the power ratio of the third cluster relative to the first 2 clusters. The power ratio varies around -9.1 dB with a standard deviation of 2.5 dB.
- The performance of the proposed receivers with/without interference cancellation outperform the RAKE receiver in three-cluster channels.

VII. CONCLUSIONS

In this paper, we proposed a PSP based receiver for underwater acoustic DSSS systems with a small spreading factor (e.g., a single digit). Exploiting the clustering property of the propagation paths, the proposed receiver consists of multiple branches, where on each branch a trellis was established on the symbol level corresponding to the signal from one cluster, and the signals from other clusters were treated as structured interference. Joint channel estimation and interference mitigation were carried on each branch, and the bit LLRs from all branches were combined for channel decoding. Simulation and experimental data sets were used to validate the performance of the proposed receiver. The results have shown that the proposed receiver has satisfactory performance in channels with different numbers of clusters, which considerably outperforms the conventional RAKE receiver.

ACKNOWLEDGMENTS

This work by X.K. is supported by the Research Fund for the Visiting Scholar Program by the China Scholarship Council Grant No. 201406310129. The work by E.C. and X.K. is partially supported by the National Natural Science Foundation of China under Grant Nos. 61471308 and 61471309. The work by Z.H.W. is partially supported by the National Science Foundation under Grant No. CNS-1551067. The authors are grateful to the anonymous reviewers whose comments have helped to improve the presentation of this paper.

[1]X. Xu, S. Zhou, A. K. Morozov, and J. C. Preisig, "Per-survivor processing for underwater acoustic communications with direct-sequence spread spectrum," J. Acoust. Soc. Am. **133**, 2746–2754 (2013).
[2]F. Qu, L. Yang, and T. C. Yang, "High reliability direct-sequence spread spectrum for underwater acoustic communications," in *Proc. of MTS/IEEE OCEANS*, Biloxi, MI) (2009), pp. 1–6.
[3]S. Zhou and Z.-H. Wang, *OFDM for Underwater Acoustic Communications* (Wiley, New York, 2014), pp. 1–410.
[4]T. C. Yang and W.-B. Yang, "Performance analysis of direct-sequence spread-spectrum underwater acoustic communications with low signal-to-noise-ratio input signals," J. Acoust. Soc. Am. **123**, 842–855 (2008).
[5]T. C. Yang and W.-B. Yang, "Low probability of detection underwater acoustic communications using direct-sequence spread spectrum," J. Acoust. Soc. Am. **124**, 3632–3647 (2008).
[6]T. C. Yang and W.-B. Yang, "Interference suppression for code-division multiple-access communications in an underwater acoustic channel," J. Acoust. Soc. Am. **126**, 220–228 (2009).
[7]P. J. Gendron, "A receiver structure for coherent reception of M-ary orthogonal spread spectrum acoustic communications at very low SNR in shallow water environments," in *Proc. of MTS/IEEE OCEANS*, San Diego, CA (2013), pp. 1–8.
[8]P. J. Gendron, "Shallow water acoustic response and platform motion modeling via a hierarchical Gaussian mixture model," J. Acoust. Soc. Am. **139**, 1923–1937 (2016).
[9]M. Stojanovic and L. Freitag, "Hypothesis-feedback equalization for direct-sequence spread-spectrum underwater communications," in *Proc. of MTS/IEEE OCEANS*, Washington, DC (2000), pp. 123–129.
[10]E. M. Sozer, J. G. Proakis, M. Stojanovic, J. A. Rice, A. Benson, and M. Hatch, "Direct sequence spread spectrum based modem for under water acoustic communication and channel measurements," in *Proc. of MTS/IEEE OCEANS'99, Riding the Crest into the 21st Century*, Seattle, WA (1999), pp. 228-233.
[11]F. Blackmon, E. Sozer, J. Proakis, and M. Stojanovic, "Performance comparison of RAKE and hypothesis feedback direct sequence spread spectrum techniques for underwater communication applications," in *Proc. of MTS/IEEE OCEANS'02*, Biloxi, MI (2002), pp. 594–603.
[12]L. Freitag, M. Stojanovic, S. Singh, and M. Johnson, "Analysis of channel effects on direct-sequence and frequency-hopped spread-spectrum acoustic communication," IEEE J. Ocean. Eng. **26**, 586–593 (2001).
[13]C. C. Tsimenidis, O. R. Hinton, A. E. Adams, and B. S. Sharif, "Underwater acoustic receiver employing direct-sequence spread spectrum and spatial diversity combining for shallow-water multiaccess networking," IEEE J. Ocean. Eng. **26**, 594–602 (2001).
[14]D. P. Konstantakos, C. C. Tsimenidis, A. E. Adams, and B. S. Sharif, "Comparison of DS-CDMA and MC-CDMA techniques for dual-dispersive fading acoustic communication networks," IEE P-Commun. **152**, 1031–1038 (2005).
[15]M. Stojanovic and L. Freitag, "Multichannel detection for wideband underwater acoustic CDMA communications," IEEE J. Ocean. Eng. **31**, 685–695 (2006).
[16]E. Calvo and M. Stojanovic, "Efficient channel-estimation-based multiuser detection for underwater CDMA systems," IEEE J. Ocean. Eng. **33**, 502–512 (2008).
[17]C. He, L. Jing, Q. Zhang, and J. Huang, "Multiuser underwater acoustic communication using cyclic shift keying," in *Proc. of MTS/IEEE OCEANS*, Shanghai, China (2016), pp. 1–6.
[18]G. Yang, J. Yin, D. Huang, L. Jin, and H. Zhou, "A Kalman filter-based blind adaptive multi-user detection algorithm for underwater acoustic networks," IEEE Sens. J. **16**, 4023–4033 (2016).
[19]T. C. Yang, "Spatially multiplexed CDMA multiuser underwater acoustic communications," IEEE J. Ocean. Eng. **41**, 217–231 (2016).
[20]R. Raheli, A. Polydoros, and C.-K. Tzou, "Per-survivor processing: A general approach to MLSE in uncertain environments," IEEE Trans. Commun. **43**, 354–364 (1995).
[21]A. K. Morozov and J. C. Preisig, "Underwater acoustic communications with multi-carrier modulation," in *Proc. of MTS/IEEE OCEANS*, Boston, MA (2006), pp. 1–6.
[22]A. K. Morozov, L. Freitag, and J. C. Preisig, "Joint channel estimation and data recovery for high-rate underwater acoustic communications with multi-carrier modulation," in *Proceedings of the Fifth ACM International Workshop on UnderWater Networks*, Woods Hole, MA (2010), pp. 1–9.
[23]X. Xu, S. Zhou, H. Sun, A. K. Morozov, and Y. Zhang, "Impulsive noise suppression in per-survivor processing based DSSS systems," in *Proc. of MTS/IEEE OCEANS*, St. John's, Canada (2014), pp. 1–5.

[24]X. Geng and A. Zielinski, "An eigenpath underwater acoustic communication channel model," in *Proc. of MTS/IEEE.OCEANS*, San Diego, CA, pp. 1189–1196.
[25]Z. Wang, S. Zhou, J. C. Preisig, K. R. Pattipati, and P. Willett, "Clustered adaptation for estimation of time-varying underwater acoustic channels," IEEE Trans. Signal Process. **60**, 3079–3091 (2012).
[26]J. Li, L. Liao, and Y. V. Zakharov, "Space-time cluster combining for UWA communications," in *MTS/IEEE.OCEANS'2016*, Shanghai, China (2016), pp. 1–6.
[27]C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," IEEE Trans. Signal Process. **58**, 1708–1721 (2010).
[28]X. Xu, Z. Wang, S. Zhou, and L. Wan, "Parameterizing both path amplitude and delay variations of underwater acoustic channels for block decoding of orthogonal frequency division multiplexing," J. Acoustic. Soc. Am. **131**, 4672–4679 (2012).
[29]J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," IEEE Trans. Inform. Theory **53**, 4655–4666 (2007).
[30]A. Viterbi, "An intuitive justification and a simplified implementation of the MAPdecoder for convolutional codes," IEEE J. Sel. Areas Commun. **16**, 260–264 (1998).
[31]L. Marchetti and R. Reggiannini, "An efficient receiver structure for sweep-spread-carrier underwater acoustic links," IEEE J. Ocean. Eng. **41**, 440–449 (2016).
[32]K. Tu, T. M. Duman, M. Stojanovic, and J. G. Proakis, "Multiple-resampling receiver design for OFDM over Doppler-distorted underwater acoustic channels," IEEE J. Ocean. Eng. **38**, 333–346 (2013).
[33]J.-Z. Huang, S. Zhou, and Z.-H. Wang, "Performance results of two iterative receivers for distributed MIMO-OFDM with large Doppler deviations," IEEE J. Ocean. Eng. **38**, 347–357 (2013).
[34]L. Freitag and S. Singh, "Performance of micro-modem psk signaling with a mobile transmitter during the 2010 mace experiment," in *Proc. of MTS/IEEE OCEANS*, Genova, Italy (2015), pp. 1–7.