

Pulse Shape-Aided Multipath Delay Estimation for Fine-Grained WiFi Sensing

Ke Xu*, He (Henry) Chen^{*†}, Chenshu Wu[‡]

^{*}Department of Information Engineering, The Chinese University of Hong Kong

[†]Shun Hing Institute of Advanced Engineering, The Chinese University of Hong Kong

[‡]Department of Computer Science, The University of Hong Kong

{xk020, he.chen}@ie.cuhk.edu.hk, chenshu@cs.hku.hk

Abstract—Due to the finite bandwidth of practical wireless systems, one multipath component can manifest itself as a discrete pulse consisting of multiple taps in the digital delay domain. This effect is called *channel leakage*, which complicates the multipath delay estimation problem. In this paper, we develop a new algorithm to estimate multipath delays of leaked channels by leveraging the knowledge of pulse-shaping functions, which can be used to support fine-grained WiFi sensing applications. Specifically, we express the channel impulse response (CIR) as a linear combination of overcomplete basis vectors corresponding to different delays. Considering the limited number of paths in physical environments, we formulate the multipath delay estimation as a sparse recovery problem. We then propose a sparse Bayesian learning (SBL) method to estimate the sparse vector and determine the number of physical paths and their associated delay parameters from the positions of the nonzero entries in the sparse vector. Simulation results show that our algorithm can accurately determine the number of paths, and achieve superior accuracy in path delay estimation and channel reconstruction compared to two benchmarking schemes.

Index Terms—channel leakage, multipath delay estimation, pulse shaping, sparse recovery

I. INTRODUCTION

In the past years, WiFi has evolved beyond its initial role of providing connectivity among wireless devices to also encompass the capability of sensing surrounding environments [1], [2]. This new trend has facilitated various applications such as indoor localization, human gesture recognition, and vital sign detection, making WiFi a key enabling technology in the era of the Internet of things. In current WiFi systems, orthogonal frequency division multiplexing (OFDM) is used to combat frequency-selective fading. The channel frequency response in WiFi OFDM systems is often referred to as channel state information (CSI) in the sensing area. Recently, several tools have been developed to extract the CSI from commodity WiFi devices [3], [4]. These complex-valued CSI can provide fine-grained information of the environment, and has been widely used in WiFi sensing.

In the field of indoor localization, the path delay is a key parameter to determine the position of a target because it

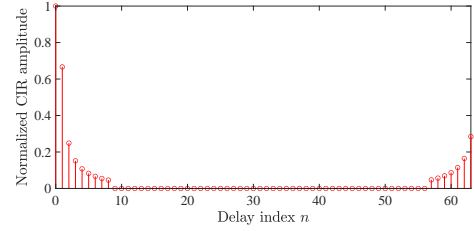


Fig. 1. Illustration of the channel leakage effect with a path delay of 20 ns and a sampling period of 50 ns. A truncated raised-cosine filter with a roll-off factor of 0.05 and a length of 16 is used. In WiFi systems, packet detection is performed by a cross-correlation operation to identify the instance with the greatest signal strength. Therefore, the first tap corresponds to the sample with the largest amplitude in the pulse, and the taps preceding this point are shifted to the end of the CIR. As a result, the absolute delay information is lost due to the packet detection.

reflects the distance between the target and a WiFi device. However, it is nontrivial to obtain an accurate estimate of the path delays from CSI. In wireless systems, pulse shaping and matched filtering are performed at the transmitter and the receiver, respectively. Due to the limited system bandwidth, when the delay of a physical path is a non-integer multiple of the sampling period, the multipath component in the discrete delay domain will manifest itself as a pulse consisting of multiple taps, instead of a single tap. This effect is called *channel leakage* [5]. An example of the channel impulse response (CIR) with the leakage effect is illustrated in Fig. 1, in which the system has a sampling period of $T = 50$ ns while a path arrives at $\tau = 20$ ns. Consequently, a pulse consisting of 16 taps is produced.

Based on the above observation, a significant problem arises if two taps within the same pulse are recognized as two physical paths with distinct delays. This situation can lead to severe degradation of the localization accuracy, especially in multipath-aided applications such as [6]. Although existing subspace-based methods [7] can directly estimate the delay parameters from the frequency domain and circumvent the above issue, these algorithms rely on the underlying assumption of using an ideal pulse shaping filter with a flat frequency response on data subcarriers. However, practical pulse functions have a finite time duration, incurring ripples in the passband and thus breaking the fundamental assumptions of those subspace-based methods. In [8], the authors developed

This research was supported in part by project #MMT 79/22 of the Shun Hing Institute of Advanced Engineering, The Chinese University of Hong Kong. The authors would like to thank Soung Chang Liew for his insightful discussions on channel sparsity and the impacts of pulse shaping.

an atomic norm-based approach to estimate the channel and further obtain the multipath parameters by incorporating the effect of pulse shaping. However, their algorithm focuses on single-carrier systems and also imposes stringent requirements on the pulse parameters of the widely-used raised-cosine filter, largely limiting their practical applications.

In this paper, we leverage the knowledge of the pulse shape and propose a new algorithm for multipath delay estimation. Since the CIR can be regarded as a superposition of multiple pulses shifted by different delays, we discretize the delay parameter into a set of grid points and formulate a sparse recovery problem using overcomplete basis vectors composed of discretized pulses shifted by the delays in the grid. We apply the sparse Bayesian learning (SBL) method [9] to update the weights of the basis vectors. The delay parameters of the physical paths can be automatically identified from the positions of nonzero weights, without requiring additional complex algorithms (e.g., [10]) to determine the number of paths.

It is worth noting that the effect of pulse shaping has been investigated in conventional sparse channel estimation works (e.g., [5], [11]). Indeed, the orthogonal matching pursuit (OMP)-based method proposed in [11] can be adapted to address the problem in this paper. Nevertheless, as shown in Section IV, the method achieves inferior performance for delay estimation due to its greedy nature, even when the number of paths is given.

II. SIGNAL MODEL

We consider a physical multipath channel in the delay domain

$$h_p(\tau) = \sum_{\ell=0}^{L-1} \alpha_\ell \delta(\tau - \tau_\ell), \quad (1)$$

where L is the number of paths, while α_ℓ and τ_ℓ are the complex amplitude and the delay of the ℓ -th path, respectively. A composite channel incorporating the effect of pulse shaping can be expressed as

$$\begin{aligned} h(\tau) &= h_p(\tau) \otimes g_t(\tau) \otimes g_r(\tau) = h_p(\tau) \otimes g(\tau) \\ &= \sum_{\ell=0}^{L-1} \alpha_\ell g(\tau - \tau_\ell), \end{aligned} \quad (2)$$

where \otimes denotes the convolution operation, $g_t(\tau)$ and $g_r(\tau)$ are the pulse shaping filter at the transmitter and the matched filter at the receiver, respectively, and $g(\tau) \triangleq g_t(\tau) \otimes g_r(\tau)$. In this paper, we consider a truncated raised-cosine pulse function, i.e., $g(t) = \text{sinc}\left(\frac{t}{T}\right) \frac{\cos\left(\frac{\pi\beta t}{T}\right)}{1 - \left(\frac{2\beta t}{T}\right)^2} w\left(\frac{t}{L_p T}\right)$, where T is the sampling period, β is the roll-off factor, $\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$ is the sinc function, and $w(t)$ is a window function such that $w(t) = 1$ for $|t| \leq 1$ and $w(t) = 0$ otherwise. Discretizing the composite channel in (2), we have¹

$$h_n = h(nT) = \sum_{\ell=0}^{L-1} \alpha_\ell g(nT - \tau_\ell), \quad n = 0, \dots, N-1, \quad (3)$$

¹Unlike in Fig. 1, we do not account for the delay shift because we can always shift the taps at the end of the CIR back to the beginning.

where N corresponds to the maximum delay spread of the composite channel. The leakage effect is evident from (3) as the energy of a path leaks to multiple taps in a pulse, making it challenging to figure out the exact delay of the physical path. For a WiFi OFDM system with K subcarriers, the frequency response of the composite channel on subcarrier k is given by

$$H_k = \sum_{n=0}^{N-1} h_n e^{-j2\pi kn/N}, \quad k = 0, \dots, K-1. \quad (4)$$

In practical WiFi systems, only a subset of subcarriers are used to transmit data/pilot symbols. For example, in IEEE 802.11a, the 0-th subcarrier is left unused due to the strong DC interference, and subcarriers indexed from 27 to 37 are also excluded to avoid interference from neighboring channels. In this paper, we estimate multipath delays from the CSI on those subcarriers available for data/pilot transmission.

III. SBL-BASED DELAY ESTIMATION

In this section, we first leverage the knowledge of the pulse shape to formulate multipath delay estimation as a sparse recovery problem by discretizing the delay parameters into grid points, and then propose an SBL-based approach to estimate the delay parameters of the physical paths.

A. Problem Formulation

Denote $\mathbf{h}_d = [h_0, \dots, h_{N-1}]^T$ and $\mathbf{h}_f = [H_0, \dots, H_{K-1}]^T$. The relationship between \mathbf{h}_f and \mathbf{h}_d can be written as

$$\mathbf{h}_f = \mathbf{F}_{0:N-1} \mathbf{h}_d, \quad (5)$$

where $\mathbf{F}_{0:N-1}$ is a partial discrete Fourier transform (DFT) matrix composed of the first N columns of a complete DFT matrix \mathbf{F} of size K , with the (m, n) -th entry of \mathbf{F} given by $F_{m,n} = e^{-j2\pi mn/N}$. As mentioned in Section II, only a portion of the frequency response is available. Therefore, the measured CSI can be expressed as

$$\mathbf{y} = \mathbf{F}_{\mathcal{I},0:N-1} \mathbf{h}_d + \mathbf{n}, \quad (6)$$

where \mathbf{y} is the measured CSI, \mathcal{I} is the index set of subcarriers used for data/pilot transmission, $\mathbf{F}_{\mathcal{I},0:N-1}$ is a matrix composed of $|\mathcal{I}|$ rows of $\mathbf{F}_{0:N-1}$ corresponding to the used subcarriers, and \mathbf{n} is the complex Gaussian-distributed measurement error with zero mean and variance σ^2 .

Taking into account the knowledge of the pulse shape, \mathbf{h}_d can be further written as

$$\mathbf{h}_d = \bar{\mathbf{A}} \bar{\boldsymbol{\alpha}}, \quad (7)$$

where $\bar{\mathbf{A}} \in \mathbb{R}^{N \times L}$ and its (n, ℓ) -th entry is given by $\bar{A}_{n,\ell} = g(nT - \tau_\ell)$, and $\bar{\boldsymbol{\alpha}} = [\alpha_0, \dots, \alpha_{L-1}]^T$. Motivated by the expression in (7), we can discretize the delay parameter into a set of fine-grained grid points as $\boldsymbol{\tau} = [0, \dots, (M-1)T_g]^T$, where T_g is the resolution of the grid, M is the number of grid points, and $(M-1)T_g$ is the maximum potential delay spread of the physical paths. Then, we can construct a

dictionary matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ with its (n, m) -th entry given by $A_{n,m} = g(nT - mT_g)$, and (7) can be approximated as

$$\mathbf{h}_d \approx \mathbf{A}\boldsymbol{\alpha}. \quad (8)$$

In (8), when the delay of a path falls on a specific grid point, the corresponding entry in $\boldsymbol{\alpha}$ will be nonzero. Due to the limited number of paths in the physical environment, there are only a small fraction of nonzero entries in $\boldsymbol{\alpha}$. In other words, $\boldsymbol{\alpha}$ is sparse. Substituting (8) into (6), we have

$$\mathbf{y} \approx \mathbf{F}_{\mathcal{I},0:N-1}\mathbf{A}\boldsymbol{\alpha} + \mathbf{n} = \mathbf{B}\boldsymbol{\alpha} + \mathbf{n}, \quad (9)$$

where $\mathbf{B} \triangleq \mathbf{F}_{\mathcal{I},0:N-1}\mathbf{A}$. Next, our objective is to recover $\boldsymbol{\alpha}$ from \mathbf{y} given \mathbf{B} . After $\boldsymbol{\alpha}$ is obtained, we can determine the delay parameters from the positions of its nonzero entries.

B. Delay Estimation

In this part, we estimate $\boldsymbol{\alpha}$ in (9) using an SBL-based method [9], which is a statistical approach for sparse signal estimation, and has been shown to be less susceptible to *structural error* and *convergence error* compared to other sparse recovery algorithms [12].

To proceed, we assign a complex Gaussian prior distribution to $\boldsymbol{\alpha}$:

$$p(\boldsymbol{\alpha}|\boldsymbol{\gamma}) = \prod_{m=0}^{M-1} \mathcal{CN}(\alpha_m; 0, \gamma_m^{-1}) = \prod_{m=0}^{M-1} \frac{\gamma_m}{\pi} e^{-\gamma_m |\alpha_m|^2}, \quad (10)$$

where $\boldsymbol{\gamma} = [\gamma_0, \dots, \gamma_{M-1}]^T$, $\mathcal{CN}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents the probability density function of a complex Gaussian random vector (or a random variable in the scalar case) with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Since $\boldsymbol{\gamma}$ is not known *a priori*, we adopt a Gamma hyperprior model because it is the conjugate prior to the Gaussian distribution in (10):

$$p(\boldsymbol{\gamma}) = \prod_{m=0}^{M-1} \text{Gamma}(\gamma_m; a, b) = \prod_{m=0}^{M-1} \frac{b^a}{\Gamma(a)} \gamma_m^{a-1} e^{-b\gamma_m}, \quad (11)$$

where $\text{Gamma}(\gamma; a, b)$ is the probability density function of a Gamma-distributed variable γ with shape parameter a and rate parameter b , and $\Gamma(\cdot)$ is the Gamma function. In general, the parameters a and b should take small values to make the hyperprior model non-informative. On the other hand, based on (9), the likelihood function is given by

$$p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathcal{CN}(\mathbf{y}; \mathbf{B}\boldsymbol{\alpha}, \boldsymbol{\beta}^{-1}\mathbf{I}) = \frac{\beta^{|\mathcal{I}|}}{\pi^{|\mathcal{I}|}} e^{-\beta \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha}\|_2^2}, \quad (12)$$

where $\beta = \frac{1}{\sigma^2}$, and \mathbf{I} is an identity matrix. We assign another Gamma distribution to β since it is also unknown:

$$p(\beta) = \text{Gamma}(\beta; c, d) = \frac{d^c}{\Gamma(c)} \beta^{c-1} e^{-d\beta}. \quad (13)$$

The joint probability density of all the involved variables can be written as

$$p(\mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, \beta) = p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta})p(\boldsymbol{\alpha}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})p(\beta). \quad (14)$$

With the probabilistic model established, we need to jointly estimate the posterior distributions of $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$ and β given \mathbf{y} from

(14). The sparsity pattern of $\boldsymbol{\alpha}$ can then be reflected from the estimate of $\boldsymbol{\gamma}$. Specifically, when γ_m is large, α_m is forced to be around zero; when γ_m is small, α_m is likely to be large. Therefore, we can find the delay of a path from the position where γ_m takes a small value. It is worth mentioning that the Gaussian prior model in (10) and its hyperprior in (11) are used only to enhance the sparsity, and the actual distribution of the multipath amplitudes is not necessarily identical to the assumed prior.

Since the posterior distribution of the involved variables is difficult to compute directly, we use the variational inference approach [13] to approximate it. The variational inference assumes that the joint posterior distribution can be factorized as

$$p(\boldsymbol{\alpha}, \boldsymbol{\gamma}, \beta|\mathbf{y}) \approx q(\boldsymbol{\alpha})q(\boldsymbol{\gamma})q(\beta). \quad (15)$$

In other words, the variables are mutually independent. By maximizing a lower bound of the likelihood function, the logarithm of the approximated posterior distribution of each variable can be expressed as [13]

$$\ln q(z_i) = \int \ln p(\mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, \beta) \prod_{j \neq i} q(z_j) dz_j + \text{const.}, \quad (16)$$

where z_i represents one of the variables $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$ or β , and “const.” is a constant irrelevant to z_i . The expression in (16) implies an iterative implementation of the variational inference algorithm, because the posterior distribution of one variable depends on the expressions of all the others. Given that explicit expressions of the posterior distributions in the real-valued case have been provided in [13], the results for the complex channel model can be obtained by making modifications to these expressions, as presented below:

- The posterior distribution of $\boldsymbol{\alpha}$ is complex Gaussian with the covariance matrix $\boldsymbol{\Sigma}$ and mean vector $\boldsymbol{\mu}$ given by

$$\boldsymbol{\Sigma} = (\langle \beta \rangle \mathbf{B}^H \mathbf{B} + \mathbf{D})^{-1}, \quad \boldsymbol{\mu} = \langle \beta \rangle \boldsymbol{\Sigma} \mathbf{B}^H \mathbf{y}, \quad (17)$$

where $\langle \cdot \rangle$ is the expectation operation with respect to the posterior distribution, and $\mathbf{D} = \text{diag}\{\langle \gamma \rangle\}$, which is a diagonal matrix with $\langle \gamma \rangle = [\langle \gamma_0 \rangle, \dots, \langle \gamma_{M-1} \rangle]^T$ on its diagonal.

- The posterior distribution of γ_m ($m = 0, \dots, M-1$) is a Gamma distribution with the shape parameter \tilde{a}_m and rate parameter \tilde{b}_m expressed as

$$\tilde{a}_m = a + 1, \quad \tilde{b}_m = b + \langle |\alpha_m|^2 \rangle, \quad (18)$$

where $\langle |\alpha_m|^2 \rangle = |\mu_m|^2 + \Sigma_{m,m}$, with μ_m and $\Sigma_{m,m}$ defined as the m -th entry of $\boldsymbol{\mu}$ and the m -th diagonal entry of $\boldsymbol{\Sigma}$, respectively. The posterior expectation of γ_m is then given by $\langle \gamma_m \rangle = \tilde{a}_m / \tilde{b}_m$.

- The posterior distribution of β is also a Gamma distribution with the shape parameter \tilde{c} and rate parameter \tilde{d} expressed as

$$\tilde{c} = c + |\mathcal{I}|, \quad \tilde{d} = d + \langle \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha}\|_2^2 \rangle, \quad (19)$$

where $\langle \|\mathbf{y} - \mathbf{B}\boldsymbol{\alpha}\|_2^2 \rangle = \|\mathbf{y} - \mathbf{B}\boldsymbol{\mu}\|_2^2 + \text{tr}(\mathbf{B}\boldsymbol{\Sigma}\mathbf{B}^H)$ with $\text{tr}(\cdot)$ denoting the trace of a matrix. Similar to γ_m , the posterior expectation of β is given by $\langle \beta \rangle = \tilde{c} / \tilde{d}$.

The computational complexity of the SBL-based algorithm primarily arises from the matrix inverse in Σ , which is of order $\mathcal{O}(M^3)$ in each iteration. We can use the Woodbury inversion identity to reduce the computational burden [9]:

$$\begin{aligned} & (\langle \beta \rangle \mathbf{B}^H \mathbf{B} + \mathbf{D})^{-1} \\ &= \mathbf{D}^{-1} - \langle \beta \rangle \mathbf{D}^{-1} \mathbf{B}^H (\mathbf{I} + \langle \beta \rangle \mathbf{B} \mathbf{D}^{-1} \mathbf{B}^H)^{-1} \mathbf{B} \mathbf{D}^{-1}. \end{aligned} \quad (20)$$

Consequently, the complexity of the matrix inversion is reduced to $\mathcal{O}(|\mathcal{I}|^3)$. Since $M \gg |\mathcal{I}|$ in this paper, the matrix multiplication will then dominate the complexity, which is $\mathcal{O}(M^2|\mathcal{I}|)$. Furthermore, as the algorithm progresses, we can prune some basis vectors whose corresponding values of $\langle \gamma_m \rangle$ are significantly large. Specifically, we remove the m -th column of \mathbf{B} and the m -th entry of α if $\langle \gamma_m \rangle$ is larger than a threshold.

After the algorithm converges, the grid points corresponding to the remaining basis vectors can be used as the estimate of delay parameters. If the number of the remaining basis vectors exceeds a predetermined value L_{\max} , we select a subset of L_{\max} delay estimates with the smallest values of $\langle \gamma_m \rangle$. Then, we further choose the delay estimates that are separated by at least $\Delta\tau$ from each other. All the above procedures are summarized in Algorithm 1.² If the complex amplitudes of the paths are of interest as well, we can fix the delay estimates and reexecute the algorithm until convergence. The posterior mean of α can then be used as an accurate estimate of the path amplitudes. As a consequence, a “cleaned” version of the CSI data can also be reconstructed.

Algorithm 1 SBL-Based Multipath Delay Estimation

Input: Measured CSI \mathbf{y} , hyperparameters a, b, c, d , the delay grid resolution T_g , the threshold for deleting a basis vector η , and the spacing for delay selection $\Delta\tau$.

Output: The delay estimates \mathcal{T} .

Initialization: The delay estimates $\mathcal{T} = \{0, \dots, (M-1)T_g\} \triangleq \{\hat{\tau}_0, \dots, \hat{\tau}_{M-1}\}$, the matrix \mathbf{B} , $\langle \gamma_m \rangle = a/b$ for all m , and $\langle \beta \rangle = c/d$.

while *stopping criterion not met* **do**

 Compute the posterior distribution of α using (17).

 Compute the posterior distribution of γ using (18).

 Compute the posterior distribution of β using (19).

 Delete $\{\hat{\tau}_m : \langle \gamma_m \rangle > \eta \min\{\langle \gamma_{m'} \rangle\}_{m'=0}^{|\mathcal{T}|-1}\}$ from \mathcal{T} .

 Reconstruct \mathbf{B} and γ using \mathcal{T} .

end while

if $|\mathcal{T}| > L_{\max}$ **then**

 Retain only L_{\max} elements in \mathcal{T} with small values of $\langle \gamma_m \rangle$.

end if

Order $\{\langle \gamma_m \rangle\}_{m=0}^{|\mathcal{T}|-1}$ such that $\langle \gamma_{m_0} \rangle \leq \dots \leq \langle \gamma_{m_{|\mathcal{T}|-1}} \rangle$.

for $i = 1 : |\mathcal{T}| - 1$ **do**

if $|\hat{\tau}_{m_i} - \hat{\tau}_{m_0}| \leq \Delta\tau$ **or** \dots **or** $|\hat{\tau}_{m_i} - \hat{\tau}_{m_{i-1}}| \leq \Delta\tau$ **then**

 Delete $\hat{\tau}_{m_i}$ from \mathcal{T} .

end if

end for

²The stopping criterion for the while loop is defined by either reaching the maximum number of iterations or attaining a relative change in the estimated path amplitudes below a threshold, which will be given in Section IV.

IV. SIMULATIONS

In this section, the performance of the proposed multipath delay estimation algorithm is evaluated through simulations. We use the MATLAB WLAN Toolbox to generate multipath channels. The system bandwidth is 20 MHz, corresponding to a sampling period $T = 50$ ns. Out of a total number of $K = 64$ subcarriers, 52 are used for data/pilot transmission with indices from 1 to 26 and from 38 to 63. The length of the cyclic prefix is set to 32, which is the same as the length of the delay-domain composite channel. We use a raised-cosine pulse function with the roll-off factor $\rho = 0.05$, truncated to a nonzero duration of $16T$. In other words, $L_p = 8$. The number of paths is 3 with delay parameters being set as 24 ns, 65 ns and 103 ns. Note that the delay difference between adjacent paths is smaller than the sampling period. The signal-to-noise ratio (SNR) is defined as the ratio between the power of the first path to the variance of the measurement error σ^2 , and the power of each subsequent path is 5 dB smaller than that of its previous one. For any unspecified parameters, we use the default settings in MATLAB. In the proposed delay estimation algorithm, the grid resolution is set to $T_g = 1$ ns. Hence, the number of grid points is $M = 850$. The hyperparameters in the Gamma distribution are given by $a = b = c = d = 1 \times 10^{-6}$, and we initialize $\langle \alpha_m \rangle = a/b$ and $\langle \beta \rangle = c/d$. A threshold of $\eta = 1 \times 10^5$ is chosen for deleting a basis vector. After the iterations, we set $L_{\max} = 10$ and $\Delta\tau = 5$ ns for further delay selection.

Two benchmarks are considered in this paper: the space-alternating generalized expectation-maximization (SAGE) algorithm [14] and the OMP-based method [11]. The SAGE algorithm is an iterative method to estimate multipath parameters and has been used in [6], [15] for WiFi sensing. However, it neglects the channel leakage effect. For the OMP-based method, it takes into account the pulse shaping function and adopts a greedy approach to find the nonzero positions in α . In theory, all three algorithms can overcome the low-precision limitation resulting from the restricted sampling rate of WiFi systems. We input the ground truth of the number of paths into the two benchmarks, resulting in an upper bound for performance evaluation. For the SAGE and our proposed algorithms, the iteration is terminated when either the relative change of the estimated path amplitudes is less than 1×10^{-4} or the number of iterations reaches 1000, while the number of iterations of OMP is fixed to L . All the results presented below are obtained by averaging over 2000 simulations. The computational complexity of the SAGE, the OMP, and the proposed algorithms are $\mathcal{O}(tLM|\mathcal{I}|)$, $\mathcal{O}(LM|\mathcal{I}|)$ and $\mathcal{O}(tM^2|\mathcal{I}|)$, respectively, where t is the number of iterations actually used. It should be noted that we use an exhaustive search method to find the optimal delay estimate in the maximization step of SAGE, the code of which is released by the authors of [15]. Despite a higher complexity, we will show that the proposed algorithm can achieve markedly better performance than the two benchmarks.

In Table I, we show the probability of correct estimation

TABLE I
PROBABILITY OF CORRECT ESTIMATION OF THE NUMBER OF PATHS

SNR	20	25	30	35	40
Probability	0.7475	0.8630	0.8990	0.9000	0.9145
MAE	0.2535	0.1370	0.1010	0.1020	0.0875

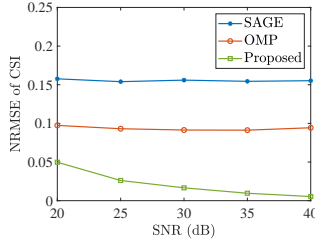


Fig. 2. Comparison of the NRMSE of CSI.

and the mean absolute error (MAE) of the number of paths of the proposed algorithm. As we can see, when the SNR is high, our algorithm can correctly determine the number of paths with a high probability and a low MAE. When the SNR is not large enough, the power of some paths with large delays could be even lower than the noise variance due to the exponential decay. As a result, they cannot be detected and the estimated number of paths is typically smaller than the ground truth. In sensing applications, such weak paths are usually discarded even if detected, because they cannot provide reliable information about a target.

Fig. 2 and Fig. 3 compare the normalized root mean square error (NRMSE) of the CSI reconstruction and the MAE of the delay parameter estimation, respectively. Since the absolute delay information is lost after the packet detection (see Fig. 1), we compute the estimation error of the delay difference between the first and the second paths. We can see that the proposed SBL-based algorithm outperforms the two benchmarks under both metrics. In SAGE, the effect of pulse shaping is neglected, and fake paths could be estimated, resulting in the worst performance. Although the OMP-based method uses the knowledge of the pulse shape, its greedy strategy can produce an inaccurate delay estimate, especially when basis vectors are highly correlated in our sparse model, and the estimation error cannot be corrected in subsequent iterations, leading to an inferior performance compared to the proposed method. The above factors also prevent the performance improvements of the two benchmarks as the SNR increases. In contrast, the proposed algorithm not only takes into account the pulse shaping function, but also adaptively updates the weights of the delay grid points, and thus achieves the best performance.

V. CONCLUSIONS

This paper investigated the multipath delay estimation problem in leaked WiFi channels, in which a physical path may produce multiple taps in the delay domain. We leveraged the knowledge of the pulse shape and formulated a sparse recovery problem using a dictionary matrix composed of discretized pulses shifted by different delays. We used the SBL method to solve this problem, and the path delay parameters can be automatically identified from the nonzero positions of the

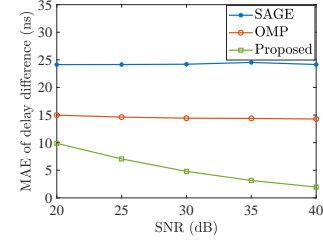


Fig. 3. Comparison of the MAE of $\tau_0 - \tau_1$.

recovered sparse vector. Simulation results have shown that our proposed algorithm can accurately estimate the number of paths and outperforms two counterparts in terms of the RN-MSE of CSI reconstruction and the MAE of delay estimation.

REFERENCES

- [1] Y. Ma, G. Zhou, and S. Wang, "WiFi sensing with channel state information: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 3, pp. 1–36, 2019.
- [2] S. Tan, Y. Ren, J. Yang, and Y. Chen, "Commodity WiFi sensing in ten years: Status, challenges, and opportunities," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 17 832–17 843, 2022.
- [3] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 1, pp. 53–53, 2011.
- [4] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity WiFi," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 53–64.
- [5] G. Taubock, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 255–271, 2010.
- [6] X. Zhang, L. Chen, M. Feng, and T. Jiang, "Toward reliable non-line-of-sight localization using multipath reflections," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 1, pp. 1–25, 2022.
- [7] Z. Li, A. Nimr, P. Schulz, and G. Fettweis, "Superresolution wireless multipath channel path delay estimation for CIR-based localization," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2022, pp. 1940–1945.
- [8] J. Li and U. Mitra, "Improved atomic norm based time-varying multipath channel estimation," *IEEE Transactions on Communications*, vol. 69, no. 9, pp. 6225–6235, 2021.
- [9] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, no. Jun, pp. 211–244, 2001.
- [10] P. Stoica and Y. Selen, "Model-order selection: a review of information criterion rules," *IEEE Signal Processing Magazine*, vol. 21, no. 4, pp. 36–47, 2004.
- [11] A. Scaglione and X. Li, "Compressed channel sensing: Is the restricted isometry property the right metric?" in *2011 17th International Conference on Digital Signal Processing (DSP)*. IEEE, 2011, pp. 1–8.
- [12] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.
- [13] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, 2008.
- [14] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. I. Pedersen, "Channel parameter estimation in mobile radio environments using the SAGE algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 3, pp. 434–450, 1999.
- [15] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu, "Widar2. 0: Passive human tracking with a single Wi-Fi link," in *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, 2018, pp. 350–361.