

## Article

# GAMP-Based Low-Complexity Sparse Bayesian Learning Channel Estimation for OTFS Systems in V2X Scenarios

Yuanbing Zheng<sup>1</sup>, Jizhe Wang<sup>1</sup>, Jian Wang<sup>1</sup>, Lu Chen<sup>1</sup>, Chongchong Wu<sup>1</sup>, Xue Li<sup>2</sup>, Yong Liao<sup>2,\*</sup> , Peng Lu<sup>3,\*</sup> and Shaohua Wan<sup>3</sup> 

<sup>1</sup> State Grid Chongqing Information and Telecommunication Company, Chongqing 400012, China

<sup>2</sup> School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China

<sup>3</sup> Key Laboratory of AI and Information Processing, Hechi University, Hechi 546300, China

\* Correspondence: liaoy@cqu.edu.cn (Y.L.); 05068@hcnu.edu.cn (P.L.)

**Abstract:** Vehicle to everything (V2X) is widely regarded as a critical application for future wireless communication networks. In V2X, large relative speeds between vehicles may severely deteriorate the performance of communication between vehicles. Orthogonal time frequency space (OTFS) modulation, which converts time- and frequency-selective channels into non-selective channels in the delay-Doppler (DD) domain, provides a solution for establishing reliable wireless communications in V2X scenarios. However, in the complex multi-scattering scenarios, the channel also suffers from a serious inter-Doppler interference (IDI) problem, which poses a great challenge to the accurate demodulation of OTFS receiver signals. To address the above problems, this paper considers the variation of Doppler sampling points within one symbol when deriving the channel model, which effectively overcomes the IDI problem, and employs a basis expansion model (BEM) to convert the channel estimation into a sparse recovery problem for the basis coefficients. In addition, to better utilize the sparse nature of the OTFS channel, a generalized approximate message passing-sparse Bayesian learning (GAMP-SBL)-based algorithm is employed to estimate the basis coefficients of the channel. The complexity of this algorithm is greatly reduced compared to the conventional SBL algorithm. Finally, system simulation results are reported to verify the superiority of the proposed scheme.

**Keywords:** V2X; OTFS; delay-Doppler; inter-Doppler interference; sparse Bayesian learning; channel estimation



**Citation:** Zheng, Y.; Wang, J.; Wang, J.; Chen, L.; Wu, C.; Li, X.; Liao, Y.; Lu, P.; Wan, S. GAMP-Based Low-Complexity Sparse Bayesian Learning Channel Estimation for OTFS Systems in V2X Scenarios. *Electronics* **2023**, *12*, 4722. <https://doi.org/10.3390/electronics12234722>

Academic Editor: Christos J. Bouras

Received: 20 October 2023

Revised: 15 November 2023

Accepted: 17 November 2023

Published: 21 November 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As a critical component of future wireless communication networks, vehicle to everything (V2X) communication strives to meet stringent standards concerning the volume of connected devices, data rate, latency, and dependability [1,2]. In 2018, the Third Generation Partnership Project (3GPP) released the R15 standard [3], which mentioned the enhancement of long-term evolution-V2X (LTE-V2X), that is, under the premise of compatibility with R14 LTE-V2X, the delay performance, reliability performance, and transmission rate of LTE-V2X are further improved by reducing the resource selection window and adding transmission diversity and carrier aggregation. In 2020, 3GPP officially froze the R16 standard [4], which defined the new ratio-V2X (NR-V2X) based on 5G. NR-V2X supports higher frequency shifts and bandwidth. Meanwhile, ultra-reliable low-latency communication (URLLC) can be achieved with the help of 5G, thus meeting the more advanced business requirements of vehicle formation, advanced driving, and remote driving.

However, V2X channels differ significantly from traditional cellular channels [5]. Because of the relative speed, there may be a higher Doppler shift when communicating between users with high mobility. As a representative of multicarrier modulation techniques, orthogonal frequency division multiplexing (OFDM) [6] is widely employed in 4G/5G

mobile communications due to its benefits of high data transmission rates, high bandwidth efficiency, and significant resistance to multipath fading [7]. In high-speed mobile scenarios, however, the Doppler effect caused by terminal and scatterer movement can seriously damage the orthogonality of subcarriers in OFDM, resulting in intercarrier interference (ICI) [8], which poses a significant challenge for channel estimation and equalization. As a result, creating new modulation waveforms and high-mobility communication techniques for next-generation wireless networks will be a major research area in the future.

In recent years, orthogonal time frequency space (OTFS) has progressively become a hot research issue in academia due to its capacity to successfully offset the channel's Doppler effect and has been discovered to significantly outperform OFDM in doubly-selective channels [9,10]. Furthermore, it inherits the same multipath resistance as OFDM and as a unique modulation scheme based on delay-Doppler (DD) domain, allows each symbol to experience the same channel gain even under high Doppler conditions. Simultaneously, OTFS can be implemented as a pre-processing and post-processing module of a filtered OFDM system, which not only allows for dense and flexible encapsulation of the reference signals to support the critical requirements in large antenna arrays in massive multiple-input multiple-output (MIMO), but also achieves compatibility with 4G and 5G system architecture [11].

Currently, there has been a lot of work to apply OTFS to complex scenarios, such as high-speed mobility. Ref. [12] employs OTFS modulation to construct a 5G-NR transmitting and receiving scheme that can be applied on unmanned aerial vehicles (UAVs) to perform accurate measurements of arrival time even in high-mobility scenarios. In addition, to address the challenges of high mobility in V2X systems, ref. [13] designs a new OTFS-based multi-antenna receiver. Due to the high resolution provided by multi-antennas, the channel sparsity in the delay-Doppler domain can be guaranteed. On this basis, a combined maximum proportion combination (MRC) OTFS iterative detection method is proposed, and the simulation results show the superiority of this scheme. All these prove that OTFS has the characteristics of anti-high Doppler.

Despite the benefits of OTFS listed above, the DD-domain spreading of OTFS symbols generated by 2D channel convolution during transmission provides a significant barrier to channel estimate. Prof. Raviteja [14] of Australia employed a single pulse signal as the pilot, inserting a piece of the guard interval around it, and then used the threshold approach to estimate the channel response in the DD domain. Nevertheless, this channel estimation approach depends on the choice of protection interval, and the threshold is chosen based on experience, which have a significant impact on channel estimation performance. Furthermore, ref. [15] developed a set of transform domain basis functions to model the OTFS channel by constructing a low-dimensional subspace that can accurately estimate the interference caused by inter-Doppler interference (IDI), while the algorithm used in the paper is linear minimum mean square error (LMMSE). The LMMSE-based channel estimation in low-dimensional subspaces uses the prior information of the channel, which is not applicable to real-world scenarios. In addition, there are similar methods in OFDM systems to estimate the phase noise of channels in high-mobility scenarios. Ref. [16] uses the basis expansion model (BEM) to fit the time variation and phase noise of multipath channels and provides an effective approximate solution for the joint estimation problem based on maximum likelihood. Therefore, BEM can effectively and accurately fit and reduce the dimension of channel changes, which is conducive to the subsequent channel estimation.

Indeed, many recent publications have presented compressed sensing-based channel estimate algorithms. Ref. [17] developed an orthogonal matching tracking algorithm for 3D structures that takes advantage of channel sparsity in three domains: the delay domain, the Doppler domain, and the angular domain. Furthermore, ref. [18] adopted three-dimensional Newtonian orthogonal matched tracking to recover the channel parameters of the time-frequency domain, incorporating channel gain, direction of arrival, delay, and Doppler shift.

Reviewing the above channel estimation approaches, it is clear that most simulation environments only evaluate simple channel situations and do not conform to the channels of real scenarios. The Doppler shift in the wireless channel is not constant, but rather conforms to the variation of the U-shaped spectrum, according to the classical Clarke's channel modeling method. Furthermore, unlike OFDM, the size of the grid in OTFS controls the delay-Doppler quantization resolution [19] (the larger the grid, the higher the resolution). However, the grid cannot generally take a larger value. Hence, the Doppler shift cannot be accurately quantified to the grid points, and the above channel estimation methods for this inaccurate channel model will cause certain estimation errors, degrading system performance [20]. To solve the aforementioned issues, we create a BEM-based channel model, transform the OTFS channel estimation into a basis coefficients recovery problem, and suggest generalized approximate message passing (GAMP) to lower the sparse Bayesian learning (SBL) algorithm's complexity. The primary contributions are as follows:

- (1) We first give the input-output expression relationship equation in the DD domain in OTFS system, and based on this, we model the OTFS channel into a DD-domain channel response that can reflect the variation of sampling points in the Doppler domain. We then convert the channel estimation problem into a basis coefficient recovery problem using the BEM.
- (2) In order to estimate the DD-domain channel, including Doppler sampling points, the GAMP-SBL method is additionally proposed to estimate the basis coefficients and channel noise while simultaneously lowering the SBL's complexity.
- (3) The scheme is evaluated by simulation of the OTFS system. The simulation and complexity analysis show that the GAMP-SBL algorithm proposed in this paper has advantages regarding computing complexity and performance under the 3GPP-specified V2X channel model. In summary, both simulation and experimental results demonstrate the effectiveness of the algorithm.

The remainder of the paper is structured as follows. In Section 2, we briefly describe the signals transmitted and receive by the OTFS system and provide a discrete-time formulation of the OTFS channel in V2X scenarios. Then, the channel estimation is formulated as a sparse signal recovery problem, and a GAMP-SBL solution is proposed in Section 3. Simulation results and complexity analysis are given in Section 4. Finally, conclusions are drawn in Section 5.

## 2. System Model

In this section, we first introduce OTFS transmitters and receivers and give mathematical representations of the transmitted and received signals. Then, we analyze the equivalent channel matrix of OTFS in the time, delay, and Doppler domains for the V2X scenario, which is the basis for the subsequent channel estimation. Finally, the sparse channel-based OTFS relation is reformulated in vector form and modeled as a sparse basis coefficient recovery problem. The base-band signal transmission model of the OTFS system is shown in Figure 1.

### 2.1. The Transmitted Signal

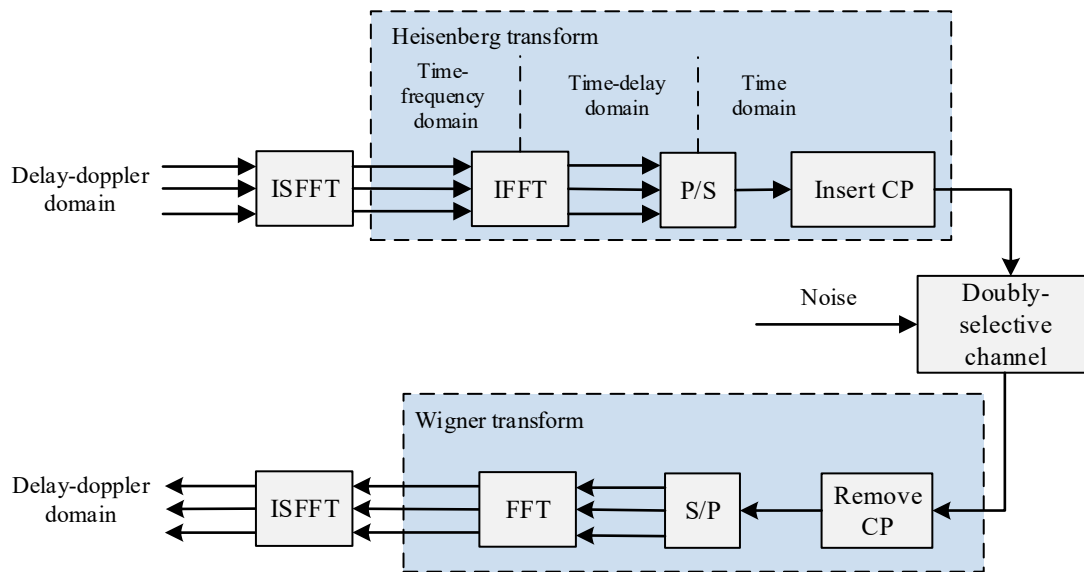
At the transmitter end, the data information that needs to be sent is modulated first, and then it is placed in the resource grid of the DD domain. Here,  $x_{dd}[n, m]$  represents the transmission signal at the  $m, n$  position in the DD domain, and  $x_{tf}[k, l]$  represents the transmission signal at the  $k, l$  position in the time-frequency domain, where  $k = 0, 1, \dots, N - 1$ ,  $l = 0, 1, \dots, M - 1$ . Inverse symmetric finite Fourier transform (ISFFT) transforms the signal from the DD domain to the time-frequency domain with the expression:

$$x_{tf}[l, k] = \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x_{dd}[m, n] e^{j2\pi(\frac{nk}{N} - \frac{ml}{M})} \quad (1)$$

Subsequently, signal  $x_{\text{tf}}[l, k]$  is converted to the time domain using Heisenberg transformation of the time–frequency domain, as follows:

$$s(t) = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} x_{\text{tf}}[l, k] g_t(t - kT) e^{j2\pi l \Delta f (t - kT)} \quad (2)$$

where  $g_t(t)$  denotes the shape filter function in the transmit end. When  $g_t(t)$  is a non-rectangular pulse, it will cause non-orthogonality between subcarriers, which will reduce the bit error rate (BER) performance [21]. Therefore, in this paper, we assume that  $g_t(t)$  is a rectangular pulse, and Equation (2) also degenerates into inverse discrete Fourier transform (IDFT), which is equivalent to traditional OFDM modulation.



**Figure 1.** The transmission model of the OTFS system.

## 2.2. The Received Signal

The time domain signal will be impacted by the channel's fading as it travels across it. Here,  $h(\tau, v)$  represents the complex base-band response of the channel in the DD domain, where  $\tau, v$  represent the delay and the Doppler component, respectively. Then,  $h(\tau, v)$  represents the following:

$$h(\tau, v) = \sum_{p=1}^P h_p \delta(\tau - \tau_p) \delta(v - v_p) \quad (3)$$

where  $P$  is the multipath number;  $h_p$  is the attenuation coefficient of  $p$ -th path;  $\tau_p, v_p$  represent the discretized value of delay and the Doppler of the  $p$ -th path, respectively; and  $\delta(\cdot)$  denotes Dirac delta function. Then, after the time-domain transmitted signal passes through the channel, the receiving signal can represent the two-dimensional convolution of the time-domain transmitted signal and the DD channel, and the expression is as follows:

$$r(t) = \int \int h(\tau, v) s(t - \tau) e^{j2\pi v(t - \tau)} d\tau dv + z(t) \quad (4)$$

where  $z(t)$  is the time domain additive Gaussian white noise. At the receiving end, the Wigner transform is applied to the time domain signal  $r(t)$ , and the expression of the discrete signal  $y_{\text{tf}}[k, l]$  is obtained:

$$y_{\text{tf}}[l, k] = \int r(t) g_r^*(t - kT) e^{-j2\pi l \Delta f (t - kT)} dt \quad (5)$$

In (5),  $g_r(t)$  denotes the matching filter function of the receiving end. Similarly, when  $g_r(t)$  is a rectangular pulse form, Equation (5) is a discrete Fourier transform (DFT). Finally, after the symmetric finite Fourier transform (SFFT) is used, the resource grid  $y_{dd}[m, n]$  of the DD domain is obtained as follows:

$$y_{dd}[m, n] = \frac{1}{\sqrt{MN}} \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} x_{tf}[l, k] e^{-j2\pi(\frac{nk}{N} - \frac{ml}{M})} \quad (6)$$

### 2.3. The Multi-Domain Channel

#### (1) Time domain

It can be seen from the above analysis that the OTFS system is based on the OFDM system with the addition of a pre-module (ISFFT transform) and a post-module (SFFT transform). Similarly, to acquire the time-domain received signal expression, we can begin with the time domain channel. It is known that the received signal in the time domain is obtained by convolution of the transmitted signal and the channel, and the relationship is as follows:

$$r(n) = \sum_{l=0}^{L-1} g(n, l) s(n-l) + z(n), \quad n \in [0, MN-1] \quad (7)$$

where  $r(n)$  is the time-domain receiving symbol, and  $L$  represents the maximum delay of the channel, which is less than the CP length  $N_{CP}$ , that is  $L < N_{CP}$ ,  $g(n, l)$  is the impulse response of the  $n$ -th sampling point in the  $l$ -th tap. It is noted that in order to avoid interference between OTFS transmitted data blocks, we add cyclic prefix (CP) at the vector header when OTFS symbols are converted to serial vectors. Therefore, the matrix relation of the time domain receiving vector can be expressed as follows:

$$\mathbf{r} = \mathbf{G}\mathbf{s} + \mathbf{z} \quad (8)$$

where  $\mathbf{s} \in \mathbb{C}^{MN \times 1}$  and  $\mathbf{r} \in \mathbb{C}^{MN \times 1}$  represent the received signal and the transmitted signal in the time domain, respectively;  $\mathbf{z} \in \mathbb{C}^{MN \times 1}$  represents the additive complex Gaussian white noise vector representing the mean of  $\mathbf{0}$  and the covariance matrix of  $\sigma^2 \mathbf{I}$ , and  $\mathbf{G} \in \mathbb{C}^{MN \times MN}$  represents the cyclic matrix composed of CIR, which is composed as follows:

$$\mathbf{G} = \begin{bmatrix} g_0^0 & 0 & \cdots & 0 & g_{L-1}^0 & g_{L-2}^0 & \cdots & g_1^0 \\ g_1^1 & g_0^1 & 0 & \ddots & 0 & g_{L-1}^1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & g_{L-2}^{k-2} \\ g_{L-2}^{k-1} & \ddots & g_1^{k-1} & g_0^{k-1} & 0 & \ddots & 0 & g_{L-1}^{k-1} \\ g_{L-1}^k & \ddots & \ddots & g_1^k & g_0^k & 0 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & g_{L-1}^{MN-1} & g_{L-2}^{MN-1} & \cdots & g_1^{MN-1} & g_0^{MN-1} \end{bmatrix} \quad (9)$$

For ease of representation, the element of  $\mathbf{G}$  in Equation (9) is denoted as  $g_l^n$ , where  $\mathbf{g} \in \mathbb{C}^{MN \times L}$  is the time domain CIR.

#### (2) Time-delay domain

Let  $\tilde{\mathbf{x}}$  and  $\tilde{\mathbf{y}}$  denote the transmitted and received vectors in the time-delay domain, respectively, and the relationships between them and the time domain vectors are respectively given as follows:

$$\tilde{\mathbf{x}} = \mathbf{P}'\mathbf{s} \quad (10)$$

$$\tilde{\mathbf{y}} = \mathbf{P}'\mathbf{r} \quad (11)$$

Note that  $\mathbf{P} \in \mathbb{C}^{MN \times MN}$  is the row-column interleaving permutation matrix [22] and is defined as follows:

$$\mathbf{P} = \begin{bmatrix} \mathbf{E}_{1,1} & \mathbf{E}_{2,1} & \cdots & \mathbf{E}_{M,1} \\ \mathbf{E}_{1,2} & \mathbf{E}_{2,2} & \cdots & \mathbf{E}_{M,2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{E}_{1,N} & \mathbf{E}_{2,N} & \cdots & \mathbf{E}_{M,N} \end{bmatrix} \quad (12)$$

where  $\mathbf{E}_{i,j} \in \mathbb{C}^{M \times N}$  denotes

$$\mathbf{E}_{i,j}(i',j') = \begin{cases} 1, & \text{if } i' = i \text{ and } j' = j \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

Further, by submitting Equations (8) and (10) into Equation (11), the input-output relation of the time-delay domain can be expressed as follows:

$$\begin{aligned} \tilde{\mathbf{y}} &= \mathbf{P}'\mathbf{r} \\ &= \mathbf{P}'\mathbf{G}\mathbf{s} + \mathbf{z}' \\ &= \mathbf{P}'\mathbf{G}\mathbf{P}\tilde{\mathbf{x}} + \mathbf{z}' \\ &= \tilde{\mathbf{H}}\tilde{\mathbf{x}} + \mathbf{z}' \end{aligned} \quad (14)$$

where  $\tilde{\mathbf{H}} = \mathbf{P}'\mathbf{G}\mathbf{P} \in \mathbb{C}^{MN \times MN}$  is the time-delay domain channel matrix.

### (3) Delay-Doppler domain

The DD domain channel is obtained by applying fast Fourier transform (FFT) to the time domain on the basis of the time-delay domain channel. Hence, the relationships between the DD domain and the time-delay domain are as follows:

$$\mathbf{y} = (\mathbf{I}_M \otimes \mathbf{F}_N) \cdot \tilde{\mathbf{y}} \quad (15)$$

$$\mathbf{x} = (\mathbf{I}_M \otimes \mathbf{F}_N) \cdot \tilde{\mathbf{x}} \quad (16)$$

where  $\mathbf{I}_M \in \mathbb{C}^{M \times M}$  is the identity matrix. In addition,  $\mathbf{F}_N \in \mathbb{C}^{N \times N}$  is Fourier matrix, which consists of the following:

$$F(n,k) = \frac{1}{\sqrt{N}} \exp(-j2\pi nk/N) \quad (17)$$

According to Equations (14)–(16), the input-output relation of the DD domain can be obtained as follows:

$$\begin{aligned} \mathbf{y} &= (\mathbf{I}_M \otimes \mathbf{F}_N) \cdot \tilde{\mathbf{H}} \cdot (\mathbf{I}_M \otimes \mathbf{F}_N^H) \mathbf{x} + \mathbf{w} \\ &= \mathbf{H}\mathbf{x} + \mathbf{w} \end{aligned} \quad (18)$$

In (18),  $\mathbf{H} = (\mathbf{I}_M \otimes \mathbf{F}_N) \cdot \tilde{\mathbf{H}} \cdot (\mathbf{I}_M \otimes \mathbf{F}_N^H) \in \mathbb{C}^{MN \times MN}$  is the DD domain channel matrix.

### 2.4. Problem Formulation

In the DD domain, the two-dimensional convolution of the input signal with the DD domain channel yields the output signal. To facilitate channel estimation, Equation (18) is converted to the following form:

$$\mathbf{y} = \mathcal{X}\mathbf{h} + \tilde{\mathbf{w}} \quad (19)$$

where  $\mathbf{y} = \text{vec}(\mathbf{Y}^T) \in \mathbb{C}^{MN \times 1}$  is serial receiving signal in the DD domain, and  $\mathbf{h} \in \mathbb{C}^{MNL \times 1}$  is the vector form of the DD channel response. In addition,  $\mathcal{X} = [\mathcal{X}_0, \dots, \mathcal{X}_L, \dots, \mathcal{X}_{L-1}] \in \mathbb{C}^{MN \times MNL}$  is composed of the transmission vector of the DD domain, in which  $\mathcal{X}_l \in \mathbb{C}^{MN \times MN}$  is a block diagonal matrix, including  $M$  small matrix  $\Delta_l$  of dimension  $N \times N$ . De-

find a new index  $r_l = \text{circshift}(1 : M, l) \in \mathbb{C}^{1 \times M}$ , and let  $X(r_l(i), :) = [h_i^l(0), \dots, h_i^l(N-1)]$  to obtain the following:

$$\begin{aligned} \Delta_i &= \text{circ}(h_i^l(0), \dots, h_i^l(N-1)) \\ &= \begin{bmatrix} h_i^l(0) & h_i^l(N-1) & \dots & h_i^l(1) \\ h_i^l(1) & h_i^l(0) & \dots & h_i^l(2) \\ \vdots & \vdots & \ddots & \vdots \\ h_i^l(N-1) & h_i^l(N-2) & \dots & h_i^l(0) \end{bmatrix} \end{aligned} \quad (20)$$

From Equation (19), it is found that the dimension of the parameter to be solved  $\mathbf{h}$  is larger than the dimension of the observed value  $\mathbf{y}$ , which suggests that the equation is an underdetermined equation and not amenable to direct solution. To estimate  $\mathbf{h}$ , we employ the BEM to reduce the number of parameters to be solved, which can use a fixed set of basis functions and a linear combination of its basis coefficients to fit the time-domain channel response. In this paper, DPS-BEM [23] is adopted to model the time-domain channel tap as follows:

$$g_l(n) = \sum_{q=0}^{Q-1} b_q(n) c_q(l) + \varepsilon_l(n) \quad (21)$$

where  $\mathbf{b}_q \in \mathbb{C}^{MN \times 1}$ ,  $\mathbf{c}_l \in \mathbb{C}^Q$ , and  $\varepsilon_l \in \mathbb{C}^N$  denote the basis vector, basis coefficients and error vector, respectively. Based on the relationship between the DD domain channel and the time domain, we can obtain the following matrix expression:

$$\begin{aligned} \mathbf{h}_l &= \frac{1}{\sqrt{N}} (\mathbf{F}_N \otimes \mathbf{I}_M) \mathbf{B} \mathbf{c}_l + \frac{1}{\sqrt{N}} (\mathbf{F}_N \otimes \mathbf{I}_M) \varepsilon_l \\ &= \mathbf{A} \mathbf{c}_l + \tilde{\varepsilon}_l \end{aligned} \quad (22)$$

By substituting Equation (22) into Equation (19), the relationship between the observed signal and the basis coefficients can be obtained as follows:

$$\begin{aligned} \mathbf{y} &= \mathcal{X}(\mathbf{I}_L \otimes \mathbf{A}) \mathbf{c} + \tilde{\mathbf{w}} \\ &= \Phi \mathbf{c} + \tilde{\mathbf{w}} \end{aligned} \quad (23)$$

where  $\mathbf{c} = [\mathbf{c}_0, \dots, \mathbf{c}_l, \dots, \mathbf{c}_{L-1}]^T \in \mathbb{C}^{QL}$  denotes the basis coefficients matrix,  $\Phi \triangleq \mathcal{X}(\mathbf{I}_L \otimes \mathbf{A}) \in \mathbb{C}^{MN \times QL}$ .

### 3. Proposed Algorithm

In this section, we propose a low-complexity channel estimation algorithm, the SBL-GAMP algorithm. First, the SBL framework is used to accomplish sparse recovery of the basis coefficients of the created channel model in Equation (23). Then, in order to reduce the computational complexity, GAMP is introduced into the expectation maximization algorithm, and a low-complexity SBL framework based on GAMP is established without degrading the performance. The GAMP framework may successfully tackle the Gaussian message propagation problem in the SBL algorithm due to the Gaussian assumption in the SBL algorithm. Meanwhile, the sum-product method is used to realize belief propagation and track Gaussian messages in the factor graph. In addition, GAMP utilizes a series of scalar operations on the input and output nodes to achieve the approximation of the a posteriori estimates of the joint vectors in the SBL E-step, which greatly reduces the computational effort.

#### 3.1. SBL Algorithm

This section introduces the SBL algorithm based on expectation maximization (EM) iteration to solve the channel model in Equation (23). Assuming that the elements in the

noise vector  $\tilde{\mathbf{w}}$  are independent of each other and obey the complex Gaussian distribution  $\tilde{\mathbf{w}} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$ , the Gaussian likelihood model is obtained as follows:

$$p(\mathbf{y}|\mathbf{c}, \tilde{\mathbf{w}}) = (2\pi\sigma^2)^{-MN/2} \exp\left\{-\frac{1}{2\sigma^2} \|\mathbf{y} - \Phi\mathbf{c}\|_2^2\right\} \quad (24)$$

Making a prior assumption about  $\mathbf{c}$  is a typical operation to avoid overfitting and is equivalent to adding a regular term from the Bayesian perspective [24]. Assuming that each component of the basis coefficients  $\mathbf{c}$  obeys Gaussian distribution and that each variable  $c_i$  corresponds to an A hyperparameter  $\alpha_i$ , the Gaussian prior distribution of  $\mathbf{c}$  can then be calculated as follows:

$$p(\mathbf{c}|\boldsymbol{\alpha}) = \prod_{i=0}^{MN} \mathcal{CN}(c_i|0, \alpha_i^{-1}) \quad (25)$$

where  $\alpha_i$  is the non-negative hyperparameter that controls the sparsity of parameter  $c_i$ . As  $\alpha_i \rightarrow \infty$ ,  $c_i \rightarrow 0$ . Based on the Gaussian prior hypothesis, the hierarchical Bayesian framework is established to ensure the sparsity of the posterior estimation. However, it is difficult to estimate the hyperparameter  $\boldsymbol{\alpha}$  since the multi-layer prior method is adopted to make a prior assumption for  $\boldsymbol{\alpha}$  again. Therefore, define the prior estimate of the hyperparameter to obey the gamma distribution:

$$\begin{aligned} p(\boldsymbol{\alpha}|a, b) &= \prod_{m=0}^{QL-1} \text{Gamma}(\alpha_m|a, b) \\ &= \prod_{m=0}^{QL-1} \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha_m} \end{aligned} \quad (26)$$

In order to make these parameters not informative, we usually set  $a = b = 10^{-4}$ . After defining the prior term for the estimation parameter  $\mathbf{c}$ , the maximum posterior estimate of  $\mathbf{c}$  can be performed using Bayesian inference.

The EM algorithm can be used to learn sparse basis coefficients  $\mathbf{c}$  and hyperparameters  $\{\boldsymbol{\alpha}, \sigma^2\}$ . In the EM frame, the basis coefficients are regarded as hidden variables, and we iteratively maximize the posterior probability to update the hyperparameters. In short, the algorithm alternates between E and M steps.

In the E step, according to Bayesian rule, we can calculate the posterior distribution, i.e.,

$$p(\mathbf{c}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2) \propto p(\mathbf{y}|\mathbf{c}, \sigma^2) p(\mathbf{c}|\boldsymbol{\alpha}) \quad (27)$$

It can be easily proven that the posterior  $\mathbf{c}$  follows a Gaussian distribution and is given by  $p(\mathbf{c}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2) = \mathcal{CN}(\boldsymbol{\mu}^c, \boldsymbol{\Sigma}^c)$ , the mean and covariance of which are respectively given as follows:

$$\boldsymbol{\mu}^c = \sigma^{-2} \boldsymbol{\Sigma}^c \Phi^H \mathbf{y} \quad (28)$$

$$\boldsymbol{\Sigma}^c = \left( \sigma^{-2} \Phi^H \Phi + \boldsymbol{\Lambda} \right)^{-1} \quad (29)$$

where  $\boldsymbol{\mu}, \boldsymbol{\Sigma}$  represent the mean and covariance matrix of  $\mathbf{c}$ , respectively, and  $\boldsymbol{\Lambda} = \text{diag}([\alpha_0, \alpha_1, \dots, \alpha_{M-1}]) \in \mathbb{C}^{M \times M}$ . The Q-function can be computed, i.e.,  $E_{\mathbf{c}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2} [\log p(\boldsymbol{\alpha}, \sigma^2|\mathbf{y})]$ , where  $E_{\mathbf{c}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2}[\cdot]$  represents the expectation of the posterior distribution  $p(\mathbf{c}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ .

In the M step, by maximizing the Q function with respect to hyperparameters  $\{\boldsymbol{\alpha}, \sigma^2\}$ , we can obtain the following updated formula:

$$\alpha_n = \frac{1}{\phi_{n,n}^c + \mu_n^{c2}} \quad (30)$$

$$\sigma^2 = \frac{\|\mathbf{y} - \Phi \boldsymbol{\mu}^c\|_2^2}{MN - \phi_n^c(1 - \alpha_n \Sigma_{n,n}^c)} \quad (31)$$

where  $\Sigma_{n,n}^c$  denotes the  $(i, i)$ -th diagonal element of the covariance matrix  $\boldsymbol{\Sigma}^c$ . The algorithm makes use of the convergence of EM and realizes the global optimization. However, the matrix inversion in Equations (23) and (24) require a lot of computation since the SBL algorithm cannot be applied to large-scale problems. Next, we will develop a low-complexity sparse Bayesian learning algorithm based on GAMP.

### 3.2. GAMP-SBL-Based Channel Estimation

GAMP is a very low-complexity Bayesian iteration technique [25] to approximate the a posteriori distribution  $p(\mathbf{c} | \mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ . As a result, it can be embedded in the EM framework to approximate the a posteriori distribution of the basis coefficients in place of the true a posteriori distribution and greatly reduce the complexity of computation.

Define  $\boldsymbol{\theta} \triangleq \{\boldsymbol{\alpha}, \sigma^2\}$  as hyperparameter and assume a posteriori independence between hidden variables  $\{c_n\}$ . As a result, the true a posteriori distribution can be approximated as follows:

$$\hat{p}(c_n | \mathbf{y}, \hat{\tau}_n^r, \boldsymbol{\theta}) = \frac{p(c_n | \boldsymbol{\theta}) \mathcal{N}(c_n | \hat{\tau}_n^r, \tau_n^r)}{\int_c p(c_n | \boldsymbol{\theta}) \mathcal{N}(c_n | \hat{\tau}_n^r, \tau_n^r)} \quad (32)$$

where  $\hat{\tau}_n$  and  $\tau_n^r$  are the scalars updated during the iteration of the GAMP algorithm. In combination with the Gaussian prior distribution of the basis coefficients, it can be easily shown that the approximate posterior also obeys a Gaussian distribution, and its mean and variance are respectively given as follows:

$$\mu_n^c = \frac{\hat{\tau}_n}{1 + \alpha_n \tau_n^r} \quad (33)$$

$$\phi_n^c = \frac{\tau_n^r}{1 + \alpha_n \tau_n^r} \quad (34)$$

The noiseless output is the other approximation  $z_m = \boldsymbol{\phi}_m^T \mathbf{c}$ , where  $\boldsymbol{\phi}_m^T$  is the  $m$ -th row of  $\Phi$ . The true posterior distribution  $p(z_m | \mathbf{y}, \boldsymbol{\theta})$  can also be approximated as follows:

$$\hat{p}(z_m | \mathbf{y}, \hat{\tau}_m^p, \boldsymbol{\theta}) = \frac{p(y_m | z_m, \boldsymbol{\theta}) \mathcal{N}(z_m | \hat{\tau}_m^p, \tau_m^p)}{\int_z p(y_m | z_m, \boldsymbol{\theta}) \mathcal{N}(z_m | \hat{\tau}_m^p, \tau_m^p)} \quad (35)$$

where  $\hat{\tau}_m$  and  $\tau_m^p$  are the scalars updated during the iteration of the GAMP algorithm. According to the assumption of Gaussian white noise, we can obtain  $p(y_m | z_m, \boldsymbol{\theta}) = \mathcal{N}(y_m - z_m, \sigma^2)$ . Therefore,  $\hat{p}(z_m | \mathbf{y}, \hat{\tau}_m^p, \boldsymbol{\theta})$  also obeys a Gaussian distribution, and its mean and variance are respectively given as follows:

$$\mu_m^z = \frac{\tau_m^p \sigma^{-2} y_m + \hat{\tau}_m^p}{1 + \sigma^{-2} \tau_m^p} \quad (36)$$

$$\phi_m^z = \frac{\tau_m^p}{1 + \sigma^{-2} \tau_m^p} \quad (37)$$

Using the above approximation, we can define the scalar functions:  $g_{\text{in}}(\cdot)$  and  $g_{\text{out}}(\cdot)$ . In the MMSE criterion, the input scalar function  $g_{\text{in}}(\cdot)$  is directly defined as the posterior mean  $\mu_n^c$  [26], i.e.,

$$g_{\text{in}}(\hat{\tau}_n, \tau_n^r, \boldsymbol{\theta}) = \mu_n^c = \frac{\hat{\tau}_n}{1 + \alpha_n \tau_n^r} \quad (38)$$

The partial derivative of  $\tau_n^r g_{\text{in}}(\hat{r}_n, \tau_n^r, \theta)$  with respect to  $\hat{r}_i$  is the posterior variance, i.e.,

$$\tau_n^r \frac{\partial}{\partial \hat{r}_n} g_{\text{in}}(\hat{r}_n, \tau_n^r, \theta) = \phi_n^c = \frac{\tau_n^r}{1 + \alpha_n \tau_n^r} \quad (39)$$

Similarly, the relation of the output scalar function  $g_{\text{out}}(\cdot)$  to the posterior mean  $\mu_m^z$  and its partial derivation are derived as follows:

$$g_{\text{out}}(\hat{p}_m, \tau_m^p, \theta) = \frac{1}{\tau_m^p} (\mu_m^z - \hat{p}_m) \quad (40)$$

$$\tau_m^p \frac{\partial}{\partial \hat{p}_m} g_{\text{out}}(\hat{p}_m, \tau_m^p, \theta) = \frac{\phi_m^z - \tau_m^p}{\tau_m^p} \quad (41)$$

Given the definitions  $g_{\text{in}}(\cdot)$  and  $g_{\text{out}}(\cdot)$ , an efficient algorithm can be derived to generate an approximate posterior distribution of the variables  $\mathbf{c}$  and  $\mathbf{z}$ . In this process, the inverse of the matrix no longer needs to be calculated. The main operation in each iteration is simple matrix multiplication, and then EM is used to update the hyperparameters. The steps of the above GAMP-SBL algorithm are noted as Algorithm 1, where  $\varphi_{mn}$  represents the  $(m, n)$ -th entry, and  $\mu_n^c(k)$  and  $\phi_n^c(k)$  represent the posterior mean and variance of  $c_n$  at the  $k$ -th iteration, respectively.

---

**Algorithm 1** Basis coefficient estimation based on GAMP-SBL

---

**Input:**  $\mathbf{y}, \Phi, \varepsilon$

**Output:**  $\tilde{\mathbf{c}}, \tilde{\sigma}^2$

1. Initialization:  $\mu^c(0) = \phi^c(0) = \mathbf{I}_N, \sigma^2(0) = 1, \hat{s}_m(-1) = 0$
  2. for  $k = 0, 1, \dots, k_{\max}$
  3. E-Step:
    4.  $\forall m \in \{1, \dots, MN\}$  :
      5.  $\hat{z}_m(k) = \sum_n \varphi_{mn} \mu_n^c(k)$
      6.  $\tau_m^p = \sum_n \varphi_{mn}^2 \phi_n^c(k)$
      7.  $\hat{p}_m(k) = \hat{z}_m(k) - \tau_m^p(k) \hat{s}_m(k-1)$
      8.  $\hat{s}_m(k) = g_{\text{out}}(\hat{p}_m(k), \tau_m^p(k), \theta)$
      9.  $\tau_m^s(k) = -\frac{\partial}{\partial \hat{p}_m} g_{\text{out}}(\hat{p}_m(k), \tau_m^p(k), \theta)$
    10.  $\forall n \in \{1, \dots, QL\}$  :
      11.  $\tau_n^r(k) = \left( \sum_m \varphi_{mn}^2 \tau_m^s(k) \right)^{-1}$
      12.  $\hat{r}_n(k) = \mu_n^c(k) + \tau_n^r(k) \sum_m \varphi_{mn} \hat{s}_m(k)$
      13.  $\mu_n^c(k+1) = g_{\text{in}}(\hat{r}_n(k), \tau_n^r(k), \theta)$
      14.  $\phi_n^c(k+1) = \tau_n^r(k) \frac{\partial}{\partial \hat{r}_n} g_{\text{in}}(\hat{r}_n(k), \tau_n^r(k), \theta)$
  15. M-Step:
    16. Update the hyperparameters according to Equations (30) and (31)
    17. If  $\sum_n |\mu_n^c(k+1) - \mu_n^c(k)|^2 \leq \varepsilon$ , break
    18. end
- 

## 4. Simulation Results and Analysis

### 4.1. Complexity Analysis

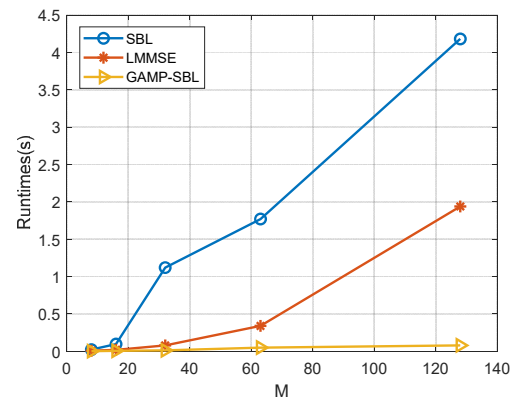
This section analyzes the complexity of the proposed algorithm and traditional basis coefficient estimation algorithms, which is our motivation to improve the SBL-based framework for basis coefficient estimation.

The traditional basis coefficient estimation algorithms based on LS and LMMSE all have the operation of matrix inversion; hence, the complexity is noted as  $(QL)^2 \times MN + (QL)^3$ . As previously stated, matrix inversion operations are present in the SBL algorithm

while solving the a posteriori probability of the basis coefficients. The number of multiplication operations are dominated by mean and variance, i.e.,  $(QL)^2 \times MN + (QL)^3$  and  $(QL)^2 \times MN + (QL)^3$ , respectively, which are comparable to the complexity of the traditional method.

Obviously, the complexity of GAMP-SBL algorithm is mainly determined by the E step in SBL, where the complexity is mainly matrix multiplication. In the GAMP algorithm, each iteration is multiplication between elements and does not involve the operation between matrices; thus, the complexity of the product operation of each iteration is approximately  $(MN) + (QL)$ .

In addition to the analytical results, the simulation runtime is also a useful indicator of complexity. Under the same constraints, when the signal noise ratio (SNR) is 20 dB, we illustrate the runtimes of the different algorithms as the number of subcarriers  $M$  in Figure 2, where the convergence error of both EM and GAMP iterations is  $10^{-6}$ . The runtimes of both SBL and LMMSE increase exponentially as  $M$  increases, which is due to the fact that matrix inversion occurs in both algorithms. The runtime of SBL is higher than that of LMMSE with regard to the presence of the iterative calculation. However, the GAMP-SBL proposed in this paper avoids matrix inversion by decoupling the matrix into scalars using approximation messages. Therefore, as the number of subcarriers  $M$  increases, it does not cause the runtime of this algorithm to increase to a large extent, which implies the superiority of the proposed algorithm.



**Figure 2.** Runtime comparisons when SNR = 20 dB.

#### 4.2. System Performance Comparison

In this section, we run a number of simulations to confirm the suggested algorithm's efficacy in terms of performance and accuracy. We adopt the EVA channel [27], which is specified by 3GPP for V2X scenarios. In addition, the path amplitude follows the Rayleigh distribution, and the average energy is exponentially decreasing over time. Each tap has a Doppler shift, which is randomly generated by Jake's model  $f_i = f_{\max} \cos(\theta_i)$ , where  $\theta_i$  indicates a uniform distribution between  $-\pi$  and  $\pi$ . In this paper, we simulate some scenarios with normalized maximum Doppler shift ranging from 3% to 15%, matching the speeds range of from 137 km/h to 685 km/h. In addition, existing channel estimation schemes are used for simulation comparison, including traditional pulse channel estimation based the threshold method [6] and the basis coefficient estimation algorithm based on LS, MMSE and SBL algorithms. The normalized mean square error (NMSE) and BER are utilized as evaluation criteria to assess the performance of the suggested algorithm and confirm its superiority and efficacy in the V2X channel. NMSE is defined as follows:

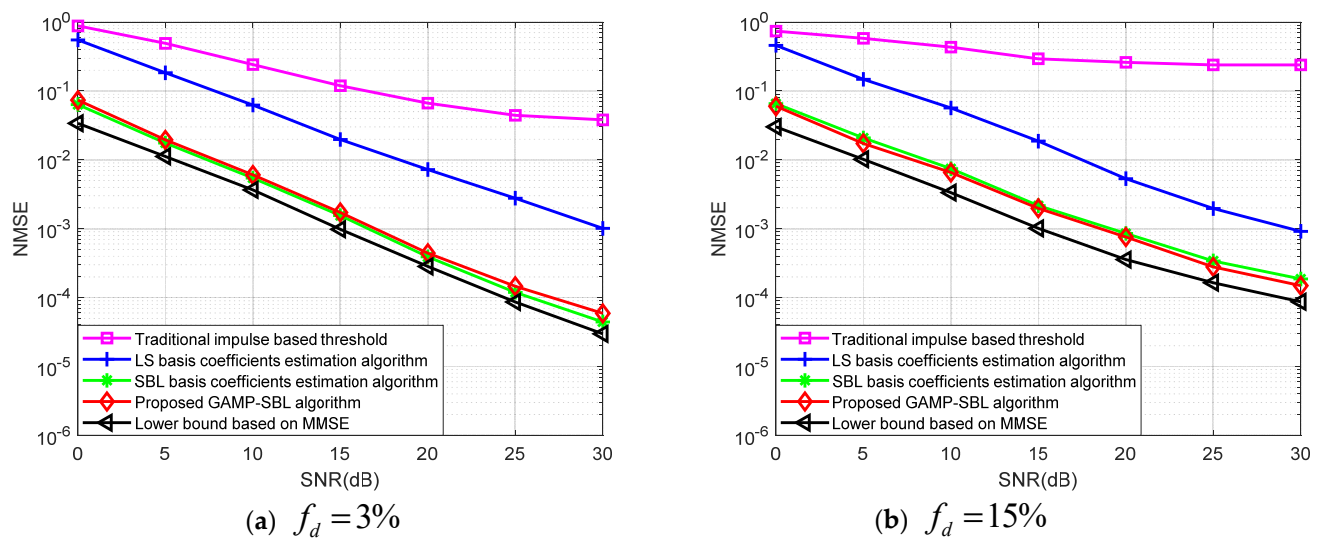
$$\text{NMSE} = 10 \log_{10} \frac{\mathbb{E}[\|\mathbf{h} - \hat{\mathbf{h}}\|^2]}{\mathbb{E}[\|\mathbf{h}\|^2]} \quad (42)$$

where  $\hat{\mathbf{h}}$  denotes the estimated DD domain channel vector. The system simulation parameter for this paper is displayed in Table 1.

**Table 1.** System simulation parameter.

| Simulation Parameters         | Settings         |
|-------------------------------|------------------|
| Carrier frequency             | 5.9 GHz          |
| Subcarrier spacing            | 15 KHz           |
| Modulation method             | QPSK             |
| Number of subcarriers ( $M$ ) | 32               |
| Number of symbols ( $N$ )     | 16               |
| CP length                     | 7                |
| Basis extended dimension $Q$  | 8                |
| Channel model                 | EVA              |
| Maximum user's velocity       | 121.5–607.5 km/h |

Figure 3a presents the NMSE curve of channel estimation when  $f_d = 3\%$ . It is found that the NMSE of the traditional pulse channel estimation-based threshold method curve will produce a platform effect with the increase of SNR and cannot continue to decline. For the basis coefficient estimation model derived in this paper, the NMSE will continue to decline as SNR increases, which proves the superiority of the channel model. At the same time, the NMSE performance of the GAMP-SBL algorithm is basically equal to that of the SBL algorithm and close to the lower bound LMMSE algorithm. It proves that the GAMP-SBL algorithm can not only greatly reduce the complexity, but also guarantee good estimation performance.

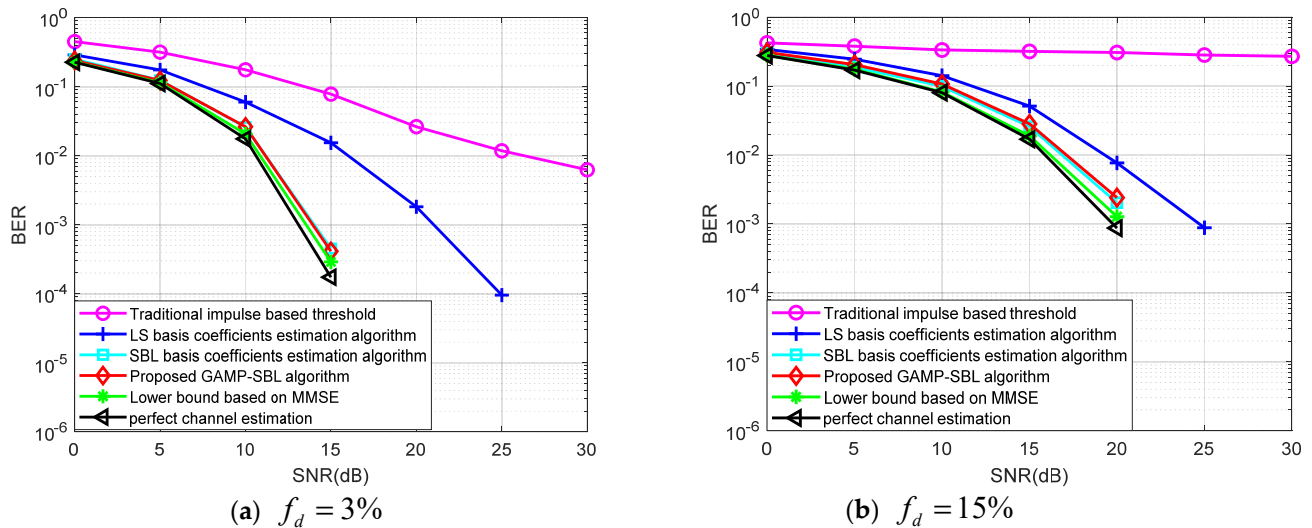


**Figure 3.** The NMSE performance comparison with velocities of 121.5–685 km/h, i.e.,  $f_d = 3$ –15%.

In Figure 3b, we illustrate the performance comparison of NMSE at  $f_d = 15\%$ . In high-speed scenarios, the traditional pulse channel estimation-based threshold method can only estimate the channel response in the DD domain of a single point, but cannot capture the channel change in each Doppler sampling point. Therefore, the platform effect is more serious, and it is not suitable for channel estimation in V2X scenarios. The NMSE performance of GAMP-SBL improves significantly with the increase in the SNR ratio. Therefore, the proposed scheme can achieve accurate channel estimation in complex scenarios, such as V2X.

Figure 4a,b show the performance comparison of BER in different scenarios, where the perfect channel BER serves as the benchmark, and the traditional MMSE algorithm is used as the symbol detector in the OTFS system. It is found that the proposed GAMP-SBL

algorithm achieves satisfactory BER performance, which is close to the performance of the MMSE algorithm in the OTFS system. The channel response obtained using the traditional method is not accurate, which leads to poor BER performance, especially in high-speed scenarios. In conclusion, the channel estimation algorithm proposed in this paper can guarantee good performance in different scenarios and greatly reduce the computational complexity, which proves the superiority of the proposed algorithm.



**Figure 4.** The BER performance comparison with velocities of 121.5–685 km/h, i.e.,  $f_d = 3$ –15%.

## 5. Conclusions

In order to make better use of OTFS channel sparsity and reduce the complexity of channel estimation, a GAMP-SBL channel estimation algorithm is proposed in this paper. We first restate the input-output relationship of the OTFS system in the DD domain and derive a channel model that can reflect the sampling points in the Doppler domain. Secondly, the channel estimation problem is transformed into a sparse basis coefficient recovery problem using BEM, which greatly reduces the parameters of channel estimation. Finally, the GAMP algorithm is used to achieve reliable channel quantization estimation in the OTFS system and avoid matrix inversion in SBL. In particular, (1) under different SNR values, the algorithm is significantly superior to the channel estimation technique based on a single pulse in the simulation, and the matrix operation is converted into scalar computation using the message passing framework, which greatly reduces the algorithm complexity. (2) The channel state information obtained by the model is more accurate. Thus, the proposed algorithm can be applied to multi-scatterer scenarios, such as V2X, and the simulation and experimental results also have superior BER performance in 3GPP-specified vehicle networking scenarios, which verifies the effectiveness of the proposed algorithm.

**Author Contributions:** Conceptualization, Y.L. and P.L.; methodology, Y.Z. and Y.L.; software, X.L.; validation, X.L.; formal analysis, J.W. (Jizhe Wang); investigation, J.W. (Jian Wang); data curation, L.C.; writing—original draft preparation, X.L.; writing—review and editing, Y.L. and C.W.; visualization, X.L.; supervision, Y.Z., P.L., S.W. and Y.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Natural Science Foundation of Chongqing, China (No. CSTB2023NSCQ-MSX0025) and Key Laboratory of AI and Information Processing (Hechi University), Education Department of Guangxi Zhuang Autonomous Region (No. 2022GXZDSY014).

**Data Availability Statement:** Due to the privacy implications of the research project, this article cannot currently provide research data.

**Conflicts of Interest:** Authors Yuanbing Zheng, Jizhe Wang, Jian Wang, Lu Chen, Chongchong Wu were employed by the company State Grid Chongqing Information and Telecommunication. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## References

- Chen, C.; Hu, J.; Qiu, T.; Atiquzzaman, M.; Ren, Z. CVCG: Cooperative V2V-Aided Transmission Scheme Based on Coalitional Game for Popular Content Distribution in Vehicular Ad-Hoc Networks. *IEEE Trans. Mob. Comp.* **2019**, *18*, 2811–2828. [CrossRef]
- Zhao, L.; Zhang, E.; Wan, S.; Hawbani, A.; Al-Dubai, A.Y.; Min, G.; Zomaya, A.Y. MESON: A Mobility-aware Dependent Task Offloading Scheme for Urban Vehicular Edge Computing. *IEEE Trans. Mob. Comp.* **2023**, accepted. [CrossRef]
- 3GPP TR 21.915 (V0.4.0). Technical Specification Group Services and System Aspects; Release 15 Description; Summary of Rel-15 Work Items [S]. 2018, 9. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3389> (accessed on 16 November 2023).
- 3GPP TR 21.916 (V1.0.0). Technical Specification Group Services and System Aspects; Release 16 Description; Summary of Rel-16 Work Items [S]. 2020, 10. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3493> (accessed on 16 November 2023).
- Jiang, P.; Deng, X.; Wan, S.; Qi, H.; Zhang, S. Confidence-Enhanced Mutual Knowledge for Uncertain Segmentation. *IEEE Trans. Intel. Transp. Syst.* **2023**, accepted. [CrossRef]
- Wu, Y.; Zou, W.Y. Orthogonal Frequency Division Multiplexing: A Multi-Carrier Modulation Scheme. *IEEE Trans. Cons. Electr.* **1995**, *4*, 392–399. [CrossRef]
- Chen, H.; Ran, N.; Yuan, L. BER Research of OFDM Systems under QPSK and 16-QAM Modulation. In Proceedings of the 2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Dalian, China, 20–21 August 2022; pp. 14–19. [CrossRef]
- Yeh, H.G. Tutorial for Space-Time ICI Parallel Cancellation Techniques for OFDM Systems. *Intern. J. Interd. Telec. Netw. (IJITN)* **2023**, *15*, 14. [CrossRef]
- Shi, J.; Cong, P.; Zhao, L.; Wang, X.; Wan, S.; Guizani, M. A Two-Stage Strategy for UAV-enabled Wireless Power Transfer in Unknown Environments. *IEEE Trans. Mob. Comp.* **2023**, accepted. [CrossRef]
- Wu, W.; Deng, X.; Jiang, P.; Wan, S.; Guo, Y. CrossFuser: Multi-Modal Feature Fusion for End-to-End Autonomous Driving Under Unseen Weather Conditions. *IEEE Trans. Intel. Transp. Syst.* **2023**, accepted. [CrossRef]
- Reddy, C.S.; Priya, P.; Sen, D.; Singhal, C. Spectral Efficient Modem Design with OTFS Modulation for Vehicular-IoT System. *IEEE Internet Things J.* **2023**, *10*, 2444–2458. [CrossRef]
- Linsalata, F.; Albanese, A.; Sciancalepore, V.; Roveda, F.; Magarini, M.; Costa-Perez, X. OTFS-superimposed PRACH-aided Localization for UAV Safety Applications. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6. [CrossRef]
- Cheng, J.; Jia, C.; Gao, H.; Xu, W.; Bie, Z. OTFS Based Receiver Scheme with Multi-Antennas in High-Mobility V2X Systems. In Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [CrossRef]
- Raviteja, P.; Phan, K.T.; Hong, Y. Embedded Pilot-Aided Channel Estimation for OTFS in Delay-Doppler Channels. *IEEE Trans. Veh. Technol.* **2019**, *68*, 4906–4917. [CrossRef]
- Qu, H.; Liu, G.; Zhang, L.; Mmran, A.; Wen, S. Low-Dimensional Subspace Estimation of Continuous-Doppler-Spread Channel in OTFS Systems. *IEEE Trans. Commun.* **2021**, *69*, 4717–4731. [CrossRef]
- Linsalata, F.; Ksairi, N. On the Joint Estimation of Phase Noise and time-Varying Channels for OFDM under High-Mobility Conditions. In Proceedings of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5. [CrossRef]
- Shen, W.; Dai, L.; An, J.; Fan, P.; Heath, R.W. Channel Estimation for Orthogonal Time Frequency Space (OTFS) Massive MIMO. *IEEE Trans. Signal Proc.* **2019**, *67*, 4204–4217. [CrossRef]
- Li, M.; Zhang, S.; Gao, F.; Fan, P.; Dobre, O.A. A New Path Division Multiple Access for the Massive MIMO-OTFS Networks. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 903–918. [CrossRef]
- Shi, D.; Wang, W.; You, L.; Song, X.; Hong, Y.; Gao, X.; Fettweis, G. Deterministic Pilot Design and Channel Estimation for Downlink Massive MIMO-OTFS Systems in Presence of the Fractional Doppler. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 7151–7165. [CrossRef]
- Dai, J.; Liu, A.; Lau, V.K.N. FDD Massive MIMO Channel Estimation with Arbitrary 2D-Array Geometry. *IEEE Trans. Signal Proc.* **2018**, *66*, 2584–2599. [CrossRef]
- Reddy, C.S.; Sen, D.; Singhal, C. Performance Analysis of NR based Vehicular IoT System with OTFS Modulation. In Proceedings of the 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 27 September–28 October 2021; pp. 1–6. [CrossRef]

22. Thaj, T.; Viterbo, E. Low Complexity Iterative Rake Decision Feedback Equalizer for Zero-Padded OTFS Systems. *IEEE Trans. Veh. Technol.* **2020**, *69*, 15606–15622. [[CrossRef](#)]
23. Zemen, T.; Mecklenbrauker, C.F. Time-variant channel estimation using discrete prolate spheroidal sequences. *IEEE Trans. Signal Proc.* **2005**, *53*, 3597–3607. [[CrossRef](#)]
24. Wei, Z.; Yuan, W.; Li, S.; Yuan, J.; Ng, D.W.K. Off-Grid Channel Estimation with Sparse Bayesian Learning for OTFS Systems. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 7407–7426. [[CrossRef](#)]
25. Yao, J.; Zhang, Z.; Shao, X.; Huang, C.; Zhong, C.; Chen, X. Concentrative Intelligent Reflecting Surface Aided Computational Imaging via Fast Block Sparse Bayesian Learning. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25 April–19 May 2021; pp. 1–6. [[CrossRef](#)]
26. Rangan, S. Generalized Approximate Message Passing for Estimation with Random Linear Mixing. In Proceedings of the 2011 IEEE International Symposium on Information Theory Proceedings, St. Petersburg, Russia, 31 July–5 August 2011; pp. 2168–2172. [[CrossRef](#)]
27. 3GPP TS 36.101 (V17.0.0). Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) Radio Transmission and Reception [S]. 2020, 12. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2411> (accessed on 16 November 2023).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.