A Sparse Bayesian Learning-based OTFS Channel Estimation for 6G Wireless System

Xueping Lan, Xiaoxu Zhang, *Senior Member, IEEE*, Li Zhang, *Senior Member, IEEE*, Zheng Ma, *Member, IEEE*, Qianli Wang, *Member, IEEE*, and Ozgur B. Akan, *Fellow, IEEE*

***Abstract*—Orthogonal time-frequency space (OTFS) modulation is considered a promising candidate for 6G wireless systems due to its superior performance in high mobility scenarios and resilience to Doppler and multipath effects. Reliable channel estimation is crucial to fully realising the potential of OTFS, but most of the existing methods suffer from limited accuracy and high computational complexity. This letter proposes a channel estimation method exploiting generalized approximate message passing sparse Bayesian learning with geometric mean decomposition (GMD-GAMP-SBL). Unlike conventional methods relying on iterative matrix inversion, the proposed approach integrates GAMP and GMD to enhance estimation accuracy and computational efficiency. Specifically, GAMP reduces complexity by replacing expectation-maximization's (EM) matrix inversion, while GMD preconditions the model to reduce the impact of sensing matrix correlations, which in turn improves the estimation accuracy. Simulation results demonstrate that the proposed GMD-GAMP-SBL achieves channel estimation accuracy that is nearly identical to that of the conventional SBL algorithm, while its computational complexity and runtime are substantially lower. This favorable trade-off positions it as a practical and efficient candidate for OTFS systems.**

***Index Terms*—OTFS, channel estimation, GAMP, SBL, geometric mean decomposition.**

I. Introduction

With the growth of high-mobility scenarios, such as high speed rail and vehicular networks, the demand for reliable and efficient wireless communications continues to increase. High mobility intensifies multipath fading and Doppler effects, challenging traditional modulation schemes. Orthogonal time-frequency space (OTFS) mitigates these impairments by mapping signals to the delay-Doppler (DD) domain, which provides a more stable and resilient representation of the wireless channel [1]. Accurate channel estimation is essential for OTFS systems, as it directly affects communication reliability and spectral efficiency, motivating continued efforts to improve estimation accuracy and efficiency [2].

The work of Xiaoxu Zhang was supported by the National Natural Science Foundation of China under Grant 62020106001. The work of Zheng Ma was supported by the National Natural Science Foundation of China under Grant No. U2268201, 62271419. The work of Qianli Wang was supported by the National Natural Science Foundation of China under Grant 62301455.

X. Lan, X. Zhang, Z. Ma, and Q. Wang are with Southwest Jiaotong University, Chengdu 611756, China (e-mail: lanxp998@my.swjtu.edu.cn, xiaoxuzhang@swjtu.edu.cn, zma@home.swjtu.edu.cn, qianli wang@qq.com).

L. Zhang is with School of Electronic and Electrical Engineering, University of Leeds, Leeds, LS2 9JT, U.K (e-mail: L.X.Zhang@leeds.ac.uk).

O. B. Akan is also with the Internet of Everything (IoE) Group, Electrical Engineering Division, Department of Engineering, University of Cambridge, Cambridge CB3 0FA, UK (email: oba21@cam.ac.uk).

Channel estimation in the DD domain is fundamentally a sparse signal recovery problem due to the limited number of dominant propagation paths in high mobility environments. Sparse Bayesian learning (SBL) has been successfully applied to solve the problem, as seen in [3][4], providing high estimation accuracy by modeling channel taps with hierarchical priors. However, its reliance on iterative matrix inversion within the expectation-maximization (EM) framework results in cubic computational complexity, making it prohibitive for real-time systems. To overcome this complexity bottleneck, message passing (MP) frameworks offer a powerful alternative. By decomposing the high-dimensional Bayesian inference into local, scalar operations, MP algorithms achieve significant complexity reduction. This principle has been effectively applied to OTFS, with notable works including MP-based recovery for fractional Doppler channels [5], superimposed-pilot-based designs for joint channel estimation and data detection [6], and approximation message passing (AMP) enhanced data detection [7]. In addition, [8] proposed an improved algorithm, namely the generalized univariate AMP (GUAMP) algorithm, which improves the performance of the MP algorithm by decomposing the matrix. However, it is not combined with the SBL algorithm, and the performance improvement is not ideal. And how to achieve estimation accuracy comparable to SBL while maintaining the low complexity advantage of MP in the presence of highly correlated OTFS sensing matrices remains a challenge.

To address this specific challenge, this letter proposes a sparse Bayesian OTFS channel estimation method with low pilot overhead, exploiting DD domain channel sparsity. By modeling it as sparse signal recovery problem, we integrate GAMP into SBL to simplify covariance updates, avoiding matrix inversion and reducing complexity. Applying geometric mean decomposition (GMD) to the sensing matrix reduces coherence, and enables a two-stage estimation process: GAMP-based sparse recovery followed by AMP refinement.

The novel contributions of this paper are summarised as follows:

- To address the high computational complexity inherent in SBL, this work proposes the GAMP-SBL algorithm. By utilizing the iterative computation of GAMP, the proposed method replaces the matrix inversion process in EM and greatly reduces the computational overhead.
- To improve channel estimation accuracy, the proposed GMD-GAMP-SBL algorithm applies GMD to the sensing matrix to reduce its high mutual coherence.

II. System model

In the OTFS system, the symbol $x[k,l]$ $\{k=0,...,N-1, l=0,..,M-1\}$ is a data sequence of quadrature amplitude modulation (QAM), which is mapped to the $M \times N$ DD domain [9]. N represents the number of Doppler taps, and M represents the number of delay taps. In this paper, the modulation symbols $x[k,l]$ are divided into data symbols $x_d[k,l]$ and pilot symbols $x_p[k,l]$.

$$x[k,l] = \begin{cases} x_p, & k = k_p, l = l_p, \\ 0, & k_p - 2k_v \le k \le k_p + 2k_v, \\ & l_p - l_\tau \le l \le l_p + l_\tau \\ x_d, & \text{otherwise} \end{cases} \tag{1}$$

where k_p and l_p are midpoints of N and M, respectively, and k_v and l_τ represent the maximum Doppler and delay taps. Following guard pilot approach [10], we allocate different power levels to pilot and data symbols.

During transmission, DD domain symbols $x[k,l]$ are transformed into time-frequency domain signals $X[n,m]$ using the inverse symplectic finite Fourier transform. The signals are then converted into continuous time-domain waveforms $s(t)$ through the Heisenberg transform. The signal convolves with the channel response $h(\tau, v)$ to produce the received time domain signal $r(t)$. The $h(\tau, v)$ is expressed as follows:

$$h(\tau, v) = \sum_{i=1}^{K} h_i \delta(\tau - \tau_i)\delta(v - v_i), \tag{2}$$

where K refers to the number of taps of the channel, h_i represents the complex channel gain of the i-th tap, and $\tau_i = \frac{l_i}{M\Delta f}$, $v_i = \frac{k_i + \kappa_i}{NT}$ are the time delay and Doppler frequency shift of the i-th tap, where l_i and k_i are the delay and Doppler taps for the i-th path respectively. Since the model established in this paper is an integer-order Doppler frequency shift, we set $\kappa_i = 0$ [11]. By applying a series of inverse transformations to $r(t)$, we obtain the received signal $y[k,l]$ in the DD domain [12]

$$y[k,l] \approx \sum_{i=1}^{K} \beta_i(k,l) h_i x\left[[k-k_i]_N, [l-l_i]_M\right] + w[k,l], \tag{3}$$

where

$$\beta_i[k,l] = \begin{cases} e^{j2\pi \frac{(l-l_i)k_i}{MN}}, & l_i \le l < M \\ \frac{N-1}{N} e^{j2\pi \frac{(l-l_i)k_i}{MN}} e^{-j2\pi \frac{[k-k_i]_N}{N}}, & 0 \le l < l_i \end{cases} \tag{4}$$

and $w[k,l]$ is the channel noise, $[.]_M$ and $[.]_N$ are the modulo M and modulo N operations.

In high-speed rail and vehicular communications, despite the complex environments, rapid movement limits the number of effective signal paths, resulting in a sparse channel impulse response. According to (3), to achieve channel estimation in the DD domain with $P = (2k_v + 1)(l_\tau + 1)$ received signals, it is essential to embed correlative $Q = (2k_v + 1)(l_\tau + 1)$ pilot symbols according to the location of each received signal, the OTFS channel estimation can be modeled as a compressive sensing problem:

$$\mathbf{Y} = (\mathbf{X_p} \odot \mathbf{A})\mathbf{h} + \mathbf{W} = \mathbf{\Phi h} + \mathbf{W}, \tag{5}$$

the observation vector $\mathbf{Y} \in \mathbb{C}^{P\times 1}$ is formed by collecting all received symbols $y[k,l]$ that fall within the guard region surrounding the pilot, where non-zero energy is received. The unknown sparse channel vector $\mathbf{h} \in \mathbb{C}^{Q\times 1}$ represents the complex gains of all potential paths within the maximum delay-Doppler spread grid. The sensing matrix $\mathbf{\Phi}$ is then built to capture the linear relationship between $\mathbf{h}$ and $\mathbf{Y}$. Each element of $\mathbf{\Phi}$ accounts for the contribution of a specific channel tap to a specific received measurement. A matrix where each element $\mathbf{X_p}[j,i]$ contains the value of the transmitted data or pilot symbol that contributes to the j-th measurement via the i-th channel tap. A matrix where each element $\mathbf{A}[j,i] = conj(\beta_i(k,l))$ compensates for the phase shift derived in (4) for the corresponding measurement and channel tap. $\mathbf{W} \in \mathbb{C}^{P\times 1}$ is the Gaussian noise vector, $\odot$ is the Hadamard product.

III. GMD-GAMP-SBL for OTFS channel estimation

The GAMP-SBL algorithm is utilised for the purpose of channel estimation, thereby circumventing the necessity for matrix inversion [13]. Despite its efficiency, the system is subject to accuracy loss due to a correlated sensing matrix. In order to address this issue, a pre-processing step is required. This pre-processing step involves the implementation of GMD. The purpose of GMD is to address the ill-conditioning that undermines the performance of GAMP [8].

The primary motivation for employing GMD preprocessing lies in its ability to mitigate the inherent ill-posedness of least-squares estimation problems. Taking the standard linear model $\mathbf{Y} = \mathbf{\Phi h} + \mathbf{W}$ as an example, the mean squared error (MSE) of the estimator is closely tied to the properties of the sensing matrix $\mathbf{\Phi}$. High condition numbers typically stem from strong correlations among the columns of matrix $\mathbf{\Phi}$, a characteristic that amplifies noise effects and leads to unstable solutions. The geometric mean decomposition constructs an equivalent system:

$$\mathbf{Y} = \mathbf{GRQ}^H\mathbf{h} + \mathbf{W} = \mathbf{Gb} + \mathbf{W}, \tag{6}$$

$$\mathbf{b} = \mathbf{RQ}^T\mathbf{h} \triangleq \mathbf{Fh}, \tag{7}$$

where $\mathbf{G} \in \mathbb{C}^{P\times r}$ and $\mathbf{Q} \in \mathbb{C}^{r\times Q}$ are semi-unitary matrices ($\mathbf{G}^T\mathbf{G} = \mathbf{I}_r$, $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}_r$), and $\mathbf{R} \in \mathbb{C}^{r\times r}$ is a novel upper-triangular matrix. The defining feature of $\mathbf{R}$ is that all its main diagonal elements are identical and equal to the geometric mean of the positive singular values of $\mathbf{\Phi}$, i.e., $r_{ii} = (\sigma_1\sigma_2\cdots\sigma_r)^{1/r}$ for all i, $\sigma_r$is the maximum singular value of the sensing matrix $\mathbf{\Phi}$. The diagonal elements of the decomposed matrix are equal to the geometric mean of all positive singular values of matrix $\mathbf{\Phi}$. This operation effectively improves the system's condition number by reducing the dynamic range of the diagonal elements in the equivalent

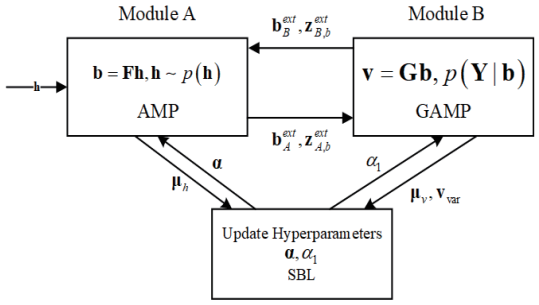


Fig. 1: Modular representation of the GMD-GAMP-SBL algorithm.

channel matrix, resulting in a lower condition number than the original system.

The GMD-GAMP-SBL algorithm improves the channel estmiation process based on the original generalized linear model (GLM) (5) by decomposing the problem into two modules: AMP (7) and GAMP (6) [8]. Module A (AMP) handles standard linear models with pseudo measurement matrices $\mathbf{F}$ and $\mathbf{h}$, while Module B (GAMP) deals with the GLM using pseudo measurement matrix $\mathbf{G}$ and $\mathbf{b}$. Compared to the original GAMP algorithm that performs iterative information updating among variables, our proposed method prioritizes the structural decomposition of the measurement matrix, and cross-domain error correction rather than relying directly on channel sparsity, making GMD-GAMP robust to correlated measurements in GLM inference, thus enhancing the estimation process.

The GMD-GAMP algorithm first performs module B and use $\hat{\mathbf{r}}$ and $\boldsymbol{\tau}^r$ as the external mean and variance passed from module B to module A. Afterwards, it executes module A and return the mean and variance of module A to module B. Therefore, in the GMD-GAMP model, the approximate posterior distribution of the channel matrix $\mathbf{h}$ is

$$p(h_i|\tilde{\mathbf{b}}, \widehat{r}_{A,i}, \tau_{A,i}^r, \boldsymbol{\Theta}) = \mathcal{CN}(h_i|\mu_i^h, \phi_i^h), \tag{8}$$

in which

$$\mu_i^h \triangleq \frac{\widehat{r}_{A,i}}{\alpha_i \tau_{A,i}^r + 1}, \quad \phi_i^h \triangleq \frac{\tau_{A,i}^r}{\alpha_i \tau_{A,i}^r + 1}, \tag{9}$$

where $\widehat{r}_{A,i} = h_i + \tilde{w}_i$, $\tau_{A,i}^r$ is the variance of $\tilde{w}_i$ according to the formula $\tilde{\mathbf{b}} = \mathbf{Fh} + \tilde{\boldsymbol{\xi}}$. Both $\tilde{w}_i$ and $\tilde{\boldsymbol{\xi}}$ are noise variables. This step is performed in the AMP algorithm.

$$p(v_j|\mathbf{Y}, \widehat{p}_{B,j}, \tau_{B,j}^p, \boldsymbol{\Theta}) = \mathcal{CN}(v_j|\mu_j^v, \phi_j^v), \tag{10}$$

where

$$\mu_j^v \triangleq \frac{\tau_{B,j}^p \alpha_1 y_j + \widehat{p}_{B,j}}{\alpha_1 \tau_{B,j}^p + 1}, \quad \phi_j^v \triangleq \frac{\tau_{B,j}^p}{\alpha_1 \tau_{B,j}^p + 1}, \tag{11}$$

where $\widehat{p}_{B,j} = v_j + \tilde{w}_j$, $\tau_{B,j}^p$ is the variance of $\tilde{w}_j$ according to the formula $\mathbf{Y} = \mathbf{Gb} + \mathbf{w}$. Both $\tilde{w}_j$ and $\mathbf{w}$ are noise variables. This step is performed in the GAMP algorithm.

Fig.1 illustrates the block diagram structure of the GMD-GAMP-SBL algorithm. Next, we describe the implementation details of the GMD-GAMP-SBL algorithm, starting with Module B and then Module A.

A. GAMP Module

The external mean $\mathbf{b}_A^{ext}(t) \in \mathcal{R}^r$ and variance $\mathbf{z}_A^{ext,b}(t) \in \mathcal{R}^r$ transmitted from Module A to Module B can be regarded as the prior mean and variance of $\mathbf{b}$. That is:

$$p(\mathbf{b}) = CN\left(\mathbf{b}; \mathbf{b}_A^{ext}(t), \operatorname{diag}\left(\mathbf{z}_A^{ext,b}(t)\right)\right). \tag{12}$$

Additionally, $\mathbf{v} = \mathbf{Gb}$ and $\mathbf{Y}|\mathbf{v} \sim p(\mathbf{Y}|\mathbf{v})$. Therefore, we apply the GAMP algorithm to this equation, treating $\mathbf{b}$ as the unknown signal and $\mathbf{G}$ as the measurement matrix, as shown below.

- Execute the linear step of the output line to obtain $\boldsymbol{\tau}_B^p(t)$ and $\hat{\mathbf{p}}_B(t)$. The detailed calculation formula is shown in Algorithm 1 in the appendix. Subsequent steps are the same.
- Next, continue executing the linear output step to obtain $\hat{\mathbf{s}}_B(t)$ and $\boldsymbol{\tau}_B^s(t)$.
- Execute the input linear step in GAMP to obtain $\hat{\mathbf{r}}_B(t)$ and $\boldsymbol{\tau}_B^r(t)$.
- Perform the input nonlinear step in Module B to obtain the posterior mean $\hat{\mathbf{b}}(t)$ and variance $\boldsymbol{\tau}^b(t)$ of variable $\mathbf{b}$.

After running T_B iterations, the external mean $\mathbf{b}_B^{ext}(t)$ and variance $\mathbf{z}_B^{ext}(t)$ from module B to module A can be obtained as follows [8]:

$$\mathbf{b}_B^{ext}(t) = \hat{\mathbf{r}}_B(t), \mathbf{z}_B^{ext}(t) = \boldsymbol{\tau}_B^r(t). \tag{13}$$

B. AMP Module

The extrinsic mean $\mathbf{b}_B^{ext}(t)$ and variance $\mathbf{z}_B^{ext}(t)$ can be interpreted as pseudo-observations and their corresponding variance for variable $\mathbf{b}$ in Module A:

$$\tilde{\mathbf{b}}(t) = \mathbf{Fh} + \tilde{\boldsymbol{\xi}}(t), \tag{14}$$

where $\tilde{\mathbf{b}}(t) = \mathbf{b}_B^{\text{ext}}(t)$, $\tilde{\boldsymbol{\xi}}(t) \sim CN(\mathbf{0}, \operatorname{diag}(\mathbf{z}_B^{ext}(t)))$. Consequently, we can execute the standard AMP algorithm with T_A iterations on this pseudo-linear model, with the detailed procedural steps outlined as follows.

- Execute the nonlinear step to obtain $\hat{\mathbf{s}}_A(t)$ and $\boldsymbol{\tau}_A^s(t)$.
- Next, continue executing the linear input step to obtain $\hat{\mathbf{r}}_A(t)$ and $\boldsymbol{\tau}_A^r(t)$. These two parameters can be obtained through the following pseudo-model:

$$\hat{\mathbf{r}}_A(t) = \mathbf{h} + \tilde{\mathbf{w}}(t), \tag{15}$$

  where $\tilde{\mathbf{w}}(t) \sim CN(0, \operatorname{diag}(\boldsymbol{\tau}_A^r(t)))$.
- Perform the input nonlinearity step in module A to obtain the posterior mean $\boldsymbol{\mu}^h(t)$ and variance $\boldsymbol{\phi}^h(t)$ of the channel.
- Perform the output linear step to obtain $\hat{\mathbf{p}}_A(t)$ and $\boldsymbol{\tau}_A^p(t)$.

After running the T_A iteration of AMP on module A, obtain the extrinsic mean $\mathbf{b}_A^{ext}(t)$ and variance $\mathbf{z}_A^{ext}(t)$:

$$\mathbf{b}_A^{ext}(t) = \hat{\mathbf{p}}_A(t), \mathbf{z}_A^{ext}(t) = \boldsymbol{\tau}_A^p(t). \tag{16}$$

C. SBL Module

- Update the hyperparameters $\boldsymbol{\alpha}$ of the channel based on the $\boldsymbol{\mu}^h(t)$ obtained from module A.
- Update the hyperparameters α_1 controlling the noise based on the estimated value of $\mathbf{v}$ obtained from module B.

Our approach separates the problem into two modules: Module B (GAMP) for the observation model and Module A (AMP-SBL) for the sparse prior. These modules iteratively pass extrinsic means and variances, with hyperparameters updated after inner loops converge. This structure, enhanced by GMD preprocessing to reduce matrix correlation, maintains high accuracy while significantly lowering complexity.

IV. Performance Evaluation

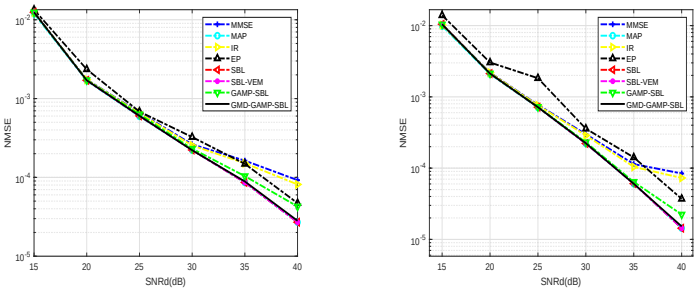
In OTFS systems, normalized mean square error (NMSE) is used to evaluate the system's performance. In the simulation, we set the carrier frequency to 4 GHz and the subcarrier spacing to 15 KHz. 4-QAM and 64-QAM modulation is used. We choose the extended vehicular A (EVA) and extended typical urban (ETU) channel model, and the maximum user speed of 500 Km/h.

We evaluate the NMSE of the GMD-GAMP-SBL in comparison with minimum mean squared error (MMSE), maximum a posteriori (MAP), expectation propagation (EP), iterative reweighting (IR), SBL-variational expectation maximizationand (SBL-VEM) and SBL estimators. It is important to note that the scope of this work focuses primarily on improving the model-driven Bayesian inference framework, particularly addressing the computational complexity and correlation issues within the SBL family of algorithms. Therefore, our comparative analysis is deliberately centered on classical model-based methods that share the same theoretical foundation.

The proposed GMD-GAMP-SBL algorithm demonstrates robust performance across both EVA and ETU channel models, as evidenced in Fig. 2. It effectively bridges the performance gap between GAMP-SBL and the computationally intensive SBL method. At $SNR_d = 40$ dB, it delivers a 2 dB improvement over GAMP-SBL and comes within 0.3 dB of the SBL benchmark. This performance advantage stems from the GMD preprocessing, which successfully mitigates the correlation in the sensing matrix. This is because the proposed GMD preprocessing can reduce the correlation between sensing matrices and maintain the low complexity of GAMP.

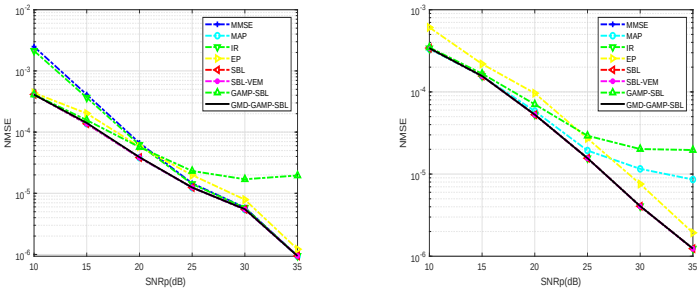
As shown in Fig. 3, the proposed GMD-GAMP-SBL algorithm exhibits near-optimal NMSE performance under both 4-QAM and 64-QAM modulations. Notably, at $SNR_p = 35$ dB, it reduces the performance gap with SBL to less than 0.3 dB while outperforming GAMP-SBL by 11-13 dB. Crucially, it maintains this high accuracy with 64-QAM, demonstrating its robustness and suitability for high-throughput 6G applications. This consistent performance stems from the GMD preprocessing, which effectively decorrelates the sensing matrix and thereby stabilizes the GAMP inference process. As a result, the estimation accuracy becomes inherently less sensitive to variations in modulation schemes, even when using higher-order constellations that are more susceptible to channel estimation errors.

As plotted in Fig. 4, the proposed GMD-GAMP-SBL algorithm demonstrates strong robustness against an increasing condition number of the sensing matrix, outperforming GAMP-SBL by more than 6 dB when the condition number reaches 500. Attributed to the GMD's generation of matrices $\mathbf{G}$, $\mathbf{Q}$, and $\mathbf{R}$ with improved properties, the method effectively mitigates ill-conditioning. Consequently, as shown in Table I, it achieves substantially reduced computational complexity and runtime compared to SBL. The primary reason is that GMD proactively balances singular values prior to channel estimation, thus eliminating ill-conditioned states and ensuring subsequent robustness. Meanwhile, the GAMP framework avoids the costly matrix inversion required by SBL, retaining only low complexity vector operations. The one-time cost of GMD decomposition is very small, thus maintaining low complexity.



(a) (b)

Fig. 2: NMSE versus SNR_d performance of channel estimation under $SNR_p = 18$dB, $N = 64$ and $M = 128$ (a) EVA, (b) ETU.



(a) (b)

Fig. 3: NMSE versus SNR_p performance of channel estimation under $N = 64$, $M = 128$ and $SNR_d = 35$dB. (a) 4-QAM, (b) 64-QAM.

V. Conclusions

This letter presents GMD-GAMP-SBL for OTFS channel estimation, which integrates GAMP to eliminate matrix inversion and GMD preprocessing to counteract performance degradation in correlated channels. Simulation results demonstrate

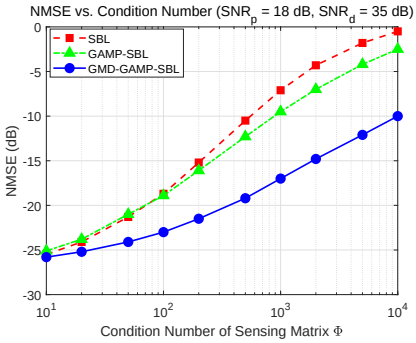


Fig. 4: NMSE versus condition number performance of channel estimation under $N = 64$, $M = 128$, $SNR_d = 35$dB and $SNR_p = 18$dB.

TABLE I: Computational complexity and runtime of channel estimation algorithms under the ETU channel model

Alogirthm	Computational Complexity	Runtime (s)
SBL-VEM	$\mathcal{O}(Q^3 + Q^2P)$	26.9997
SBL	$\mathcal{O}(Q^3)$	24.1844
GAMP-SBL	$\mathcal{O}(Q^2)$	0.7804
GMD-GAMP-SBL	$\mathcal{O}(Q^2)$	0.8914

that our method bridges the accuracy-complexity gap between SBL and GAMP-SBL, achieving near-identical performance to SBL while maintaining $\mathcal{O}(Q^2)$ complexity. By effectively combining message passing with Bayesian learning, this work provides a practical high-precision, low-complexity solution for 6G systems in high-mobility environments. In addition, we plan to explore hybrid approaches that combine the theoretical advantages of model driven methods with the adaptive capabilities of data-driven technologies in the future.

APPENDIX

The detailed process of the GMD-GAMP-SBL algorithm is as described in **Algorithm 1**.

Algorithm 1 Channel estimation based on the GMD-GAMP-SBL

1.**Initialization:** $\mathbf{Y}$, $\mathbf{\Phi}$, P, Q, $\alpha_1(0)$, $\boldsymbol{\alpha}(0)$, $\hat{\mathbf{b}}(0)$, $\tau^b(0)$, $\hat{\mathbf{s}}_B(0)$, $\hat{\mathbf{p}}_A(0)$, $\boldsymbol{\tau}_A^p(0)$, $\mathbf{b}_A^{ext}(0)$, $\mathbf{z}_A^{ext,b}(0)$, $\boldsymbol{\mu}^h(0)$, **set** T_{max}, T_A **and** T_B.
2. **For** $t = 1, ..., T_{max}$**:**
3. **For** $t_B = 1, ..., T_B$**:**
4. Output linear step:
$\boldsymbol{\tau}_B^p(t) = |\mathbf{G}|^2\boldsymbol{\tau}^b(t)$, $\hat{\mathbf{p}}_B(t) = \mathbf{G}\hat{\mathbf{b}}(t-1) - \boldsymbol{\tau}_B^p(t)\hat{\mathbf{s}}_B(t-1)$
5. Output linear step:
$\hat{\mathbf{s}}_B(t) = \frac{1}{\boldsymbol{\tau}_B^p(t)}\left(\frac{\boldsymbol{\tau}_B^p(t)\alpha_1(t-1)\mathbf{Y}+\hat{\mathbf{p}}_B(t)}{\boldsymbol{\tau}_B^p(t)\alpha_1(t-1)+1} - \hat{\mathbf{p}}_B(t)\right)$,
$\boldsymbol{\tau}_B^s(t) = \frac{\alpha_1(t-1)}{\alpha_1(t-1)\boldsymbol{\tau}_B^p(t)+1}$
6. Input linear step:
$\boldsymbol{\tau}_B^r(t) = \left(\left|\mathbf{G}^T\right|^2\boldsymbol{\tau}_B^s(t)\right)^{-1}$,
$\hat{\mathbf{r}}_B(t) = \hat{\mathbf{b}}(t-1) + \boldsymbol{\tau}_B^r(t) \odot \left(\mathbf{G}^T\hat{\mathbf{s}}_B(t)\right)$
7. Input nonlinear step:
$\hat{\mathbf{b}}(t) = \frac{\hat{\mathbf{r}}_B(t)\mathbf{z}_A^{ext}(t-1)+\mathbf{b}_A^{ext}(t-1)\boldsymbol{\tau}_B^r(t)}{\boldsymbol{\tau}_B^r(t)+\mathbf{z}_A^{ext}(t-1)}$,
$\boldsymbol{\tau}^b(t) = \frac{\mathbf{z}_A^{ext}(t-1)\boldsymbol{\tau}_B^r(t)}{\boldsymbol{\tau}_B^r(t)+\mathbf{z}_A^{ext}(t-1)}$
8. **end**
9. Set $\mathbf{b}_B^{ext}(t) = \hat{\mathbf{r}}_B(t)$ and $\mathbf{z}_B^{ext}(t) = \boldsymbol{\tau}_B^r(t)$.
10. **For** $t_A = 1, ..., T_A$**:**
11. Output nonlinear step:
$\hat{\mathbf{s}}_A(t) = \frac{\mathbf{b}_B^{ext}(t)-\hat{\mathbf{p}}_A(t-1)}{\mathbf{z}_B^{ext}(t)+\boldsymbol{\tau}_A^p(t-1)}$, $\boldsymbol{\tau}_A^s(t) = \frac{1}{\mathbf{z}_B^{ext}(t)+\boldsymbol{\tau}_A^p(t-1)}$
12. Input linear step:
$\boldsymbol{\tau}_A^r(t) = \left(\left|\mathbf{F}^T\right|^2\boldsymbol{\tau}_A^s(t)\right)^{-1}$,
$\hat{\mathbf{r}}_A(t) = \boldsymbol{\mu}^h(t-1) + \boldsymbol{\tau}_A^r(t) \odot \left(\mathbf{F}^T\hat{\mathbf{s}}_A(t)\right)$
13. Input nonlinear step:
$\boldsymbol{\mu}^h(t) = \frac{\hat{\mathbf{r}}_A(t)}{\boldsymbol{\alpha}(t-1)\boldsymbol{\tau}_A^r(t)+1}$, $\boldsymbol{\phi}^h(t) = \frac{\boldsymbol{\tau}_A^r(t)}{\boldsymbol{\alpha}(t-1)\boldsymbol{\tau}_A^r(t)+1}$
14. Output nonlinear step:
$\boldsymbol{\tau}_A^p(t) = |\mathbf{F}|^2\boldsymbol{\phi}^h(t)$, $\hat{\mathbf{p}}_A(t) = \mathbf{F}\boldsymbol{\mu}^h(t) - \boldsymbol{\tau}_A^p(t) \odot \hat{\mathbf{s}}_A(t)$
15. **end**
16. Set $\mathbf{b}_A^{ext}(t) = \hat{\mathbf{p}}_A(t)$ and $\mathbf{z}_A^{ext}(t) = \boldsymbol{\tau}_A^p(t)$.
17. Update of $\boldsymbol{\alpha}$: $\alpha_i^{(t+1)} = \frac{2a-1}{\langle\mu^h{}_i^2\rangle+2b}$
18. Update of α_1: $\alpha_1^{(t+1)} = \frac{P+2c-2}{\sum_j\langle(y_j-\mu_{vj})^2+v_{varj}\rangle+2d}$
19. until $\sum_i \left|\mu_i^h(t+1) - \mu_i^h(t)\right|^2 \leq 10^{-6}$
20. **end**

Output: $\mathbf{h}$, $\boldsymbol{\alpha}$, α_1

REFERENCES

[1] S. Doan-Tusha, A. Tusha, S. Althunibat and K. Qaraqe, "Orthogonal Time Frequency Space Multiple Access Using Index Modulation," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 15858-15866, Dec. 2023.

[2] H. Zhang, X. Huang and J. A. Zhang, "Low-Overhead OTFS Transmission With Frequency or Time Domain Channel Estimation," *IEEE Trans. Veh. Technol.*, vol. 73, no. 1, pp. 799-811, Jan. 2024.

[3] Y. Liu, S. Zhang, F. Gao, J. Ma and X. Wang, "Uplink-Aided High Mobility Downlink Channel Estimation Over Massive MIMO-OTFS System," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 9, pp. 1994-2009, Sept. 2020.

[4] M. Li, S. Zhang, F. Gao, P. Fan and O. A. Dobre, "A New Path Division Multiple Access for the Massive MIMO-OTFS Networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 903-918, April 2021.

[5] F. Liu, Z. Yuan, Q. Guo, Z. Wang and P. Sun, "Message Passing-Based Structured Sparse Signal Recovery for Estimation of OTFS Channels With Fractional Doppler Shifts," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 7773-7785, Dec. 2021.

[6] H. B. Mishra, P. Singh, A. K. Prasad and R. Budhiraja, "OTFS Channel Estimation and Data Detection Designs With Superimposed Pilots," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2258-2274, April 2022.

[7] W. Zhuang et al., "Approximate Message Passing-Enhanced Graph Neural Network for OTFS Data Detection," *IEEE Wireless Commun. Lett.*, vol. 13, no. 7, pp. 1913-1917, July 2024.

[8] J. Zhu, X. Meng, X. Lei and Q.Guo, "A Unitary Transform Based Generalized Approximate Message Passing," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, RhodesIsland, Greece, pp.1-5, 2023.

[9] Y. Kanazawa, H. Iimori, C. Pradhan, S. Malomsoky and N. Ishikawa, "Multiple Superimposed Pilots for Accurate Channel Estimation in Orthogonal Time Frequency Space Modulation," *Proc. IEEE 98th Veh. Technol. Conf. (VTC-Fall)*, Hong Kong, pp. 1-5, 2023.

[10] Q. Wang, M. Lei, M. -M. Zhao and M. Zhao, "Variational Bayesian Inference Based Channel Estimation for OTFS System with LSM Prior," *Proc. Int. Symp. Wirel. Commun. Syst. (ISWCS)*, Hangzhou, China, pp. 1-5, 2022.

[11] P. Raviteja, Y. Hong, E. Viterbo and E. Biglieri, "Practical Pulse-Shaping Waveforms for Reduced-Cyclic-Prefix OTFS," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 957-961, Jan. 2019.

[12] P. Raviteja, K. T. Phan, Y. Hong, and E. Viterbo, "Interference cancellation and iterative detection for orthogonal time frequency space modulation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6501-6515, Oct. 2018.

[13] X. Zhang, P. Fan, L. Hao and X. Quan, "Generalized Approximate Message Passing Based Bayesian Learning Detectors for Uplink Grant-Free NOMA," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 15057-15061, Nov. 2023.