

Low-Complexity Sparse Superimposed Coding for Ultra Reliable Low Latency Communications

Yanfeng Zhang*, Xi'an Fan[§], Xu Zhu[†], Jinkai Zheng*, Hui Liang*, Weiwei Yang*, and Tom H. Luan[#]

*School of Electrical Engineering & Intelligentization, Dongguan University of Technology, Dongguan, China

[§]School of Computer Science and Technology, Dongguan University of Technology, Dongguan, China

[†]School of Electronic and Information Engineering, Harbin Institute of Technology, Shenzhen, China

[#]School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an, China

Email: yfzhang@ieee.org

Abstract—Sparse superimposed coding (SSC) has emerged as a promising technique for short-packet transmission in ultra-reliable low-latency communication scenarios. However, conventional SSC schemes often suffer from high encoding and decoding complexity due to the use of dense codebook matrices. In this paper, we propose a low-complexity SSC scheme by designing a sparse codebook structure, where each codeword contains only a small number of non-zero elements. The decoding is performed using the traditional multipath matching pursuit algorithm, and the overall complexity is significantly reduced by exploiting the sparsity of the codebook. Simulation results show that the proposed scheme achieves a favorable trade-off between BLER performance and computational complexity, and exhibits strong robustness across different transmission block lengths.

Index Terms—sparse superimposed codes, short-packet transmission, ultra-reliable low-latency communication

I. Introduction

Next-generation wireless networks are required to support ultra reliable low latency communication (URLLC), which serves as a fundamental enabler for time-critical applications including autonomous vehicles, industrial IoT systems, and tactile internet services [1]. Traditional transmission strategies that rely on lengthy data packets face substantial challenges when applied to URLLC environments. These conventional approaches demand extensive resource allocation for control signaling, automatic repeat request mechanisms, and error correction procedures to maintain transmission reliability, consequently resulting in elevated latency and diminished spectral efficiency. Furthermore, when packet lengths are reduced, the performance of established coding schemes deteriorates significantly—LDPC codes exhibit reduced redundancy while polar codes cannot achieve complete channel polarization, both leading to compromised reliability. Consequently, developing efficient transmission techniques for short packets represents a fundamental challenge in URLLC implementation.

Sparse vector coding (SVC) has been proposed as a promising approach for short-packet transmission in URLLC scenarios [2]. Unlike conventional channel coding,

SVC encodes information bits into the non-zero indexes of a sparse vector, which is then spread to the time-frequency domain for transmission. At the receiver, a simple energy detection algorithm decodes the signal by identifying these non-zero indexes. Studies have shown that SVC outperforms traditional coding schemes in terms of reliability in short-packet scenarios [2]–[5].

Several variants of SVC have been proposed to enhance its performance. In [3], an enhanced SVC (ESVC) scheme has been proposed to improve spectral efficiency by mapping information bits to both non-zero indices and quadrature amplitude modulation (QAM) symbols. In [4], the SSC leverages multiple constellation alphabets to enhance decoding performance, while further improvements in block error rate (BLER) are achieved by encoding additional information into different constellation alphabets [5]. The sparse regression codes (SPARC), which follow a similar SVC approach, have been explored with efficient approximate message passing (AMP) decoding algorithms [6]. A generalized SPARC (GSPARC) has been introduced to improve codeword construction [7], and block orthogonal sparse superposition codes utilize sequential bit mapping over orthogonal sub-matrices [8]. Our prior work introduced a block SVC scheme with a block sparse mapping pattern to enhance decoding performance [9]. The applicability of SVC has also been extended to various systems, including multi-user multiple-input multiple-output [10], low-resolution analog-to-digital converter systems [11], and high-mobility communications [12].

Most existing SVC schemes [2]–[14] focus on improving reliability through the design of novel encoding and decoding algorithms or optimized codebook matrices. However, these approaches often overlook the computational complexity of SVC, which is a critical factor in resource-constrained IoT scenarios. Moreover, to ensure high decoding accuracy, most SVC schemes employ dense Gaussian or Bernoulli matrices as codebooks, leading to an exponential increase in codebook size with the number of transmitted bits. This results in significantly higher encod-

ing and decoding complexity as well as storage overhead. Consequently, designing SVC schemes with sparse codebook matrices that substantially reduce complexity while maintaining acceptable decoding performance remains an open research challenge.

In this paper, a low-complexity SSC scheme is proposed for short-packet URLLC scenarios. A sparse Bernoulli codebook matrix is designed to substantially reduce the encoding/decoding complexity of SSC schemes with only minimal decoding performance degradation. The proposed sparse codebook matrix enables complexity-performance trade-offs through a sparsity factor that controls the proportion of non-zero elements within the codebook matrix.

The rest of this paper is organized as follows. In Sections II and III, the encoding and decoding process of the proposed low-complexity SSC scheme are described, respectively. In Section IV, the complexity of proposed decoding algorithm is analyzed. Extensive simulation results are shown in Section V, while conclusions are drawn in Section VI.

Notations: Bold symbols represent vectors or matrices. $(\cdot)^T$, $(\cdot)^H$ and $(\cdot)^\dagger$ denote the transpose, conjugate transpose and matrix inversion, respectively. $\|\cdot\|_p$ denotes ℓ_p norm operation. $\lfloor \cdot \rfloor$ is the round-down operation. $\binom{N}{K}$ denotes the number of combinations of selecting K items from N items.

II. Encoding of the Low-Complexity SSC Scheme

In this section, the basic concept of SSC encoding is introduced, including the sparse mapping pattern, sparse codebook spreading, and transmission of spread sequence.

A. Sparse Mapping Pattern of SSC

Consider a short-packet communication system where the information payload comprises b bits per data packet. As shown in Fig. 1, the b bits are partitioned into two segments. The b_I bits are encoded into the non-zero block index \mathcal{B} of a sparse vector $\mathbf{s} \in \mathbb{C}^N$ through sparse mapping. The b_S bits are then mapped onto the values of K non-zero elements of the sparse vector via QAM modulation. Subsequently, the sparse vector is processed through codebook spreading to generate a spread sequence, which is then allocated to time-frequency resources for transmission.

The SSC scheme employs a sparse mapping procedure that allocates the K active entries to N available positions. Therefore, the number of information bits that can be carried by the index resource is determined by

$$b_I = \left\lfloor \log_2 \binom{N}{K} \right\rfloor. \quad (1)$$

In addition, the number of information bits mapped onto the K QAM symbol is

$$b_S = K \log_2(M_{\text{mod}}), \quad (2)$$

where M_{mod} is the modulation order.

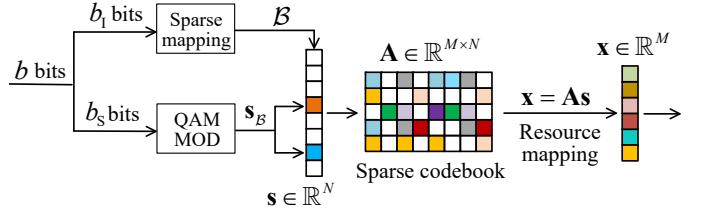


Fig. 1. Diagram of the sparse mapping process of SSC scheme. b bits of information are mapped to a sparse vector of length $N = 8$ and containing $K = 2$ non-zero elements.

B. Sparse Codebook Spreading

Following the sparse encoding procedure, the non-zero components within the sparse vector undergo spreading via K pseudo-random codewords. This transformation process is mathematically formulated as

$$\mathbf{x} = \mathbf{As} = \sum_{m=1}^M \sum_{k=1}^K a_{m,k} s_k, \quad (3)$$

where $\mathbf{x} \in \mathbb{C}^M$ is the spread sequence, s_k and $a_{m,k}$ denote, respectively, the k -th non-zero element and the (m, k) -th element in codebook matrix \mathbf{A} . The elements $a_{m,k}$ in the codebook matrix follow a Bernoulli distribution with a probability of 0.5 [2], taking values of either 1 or -1. An example of \mathbf{A} for $M = 4$ and $N = 6$ is given by

$$\mathbf{A} = \sqrt{\frac{1}{K}} \begin{bmatrix} 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}. \quad (4)$$

According to compressed sensing theory [15], to ensure successful recovery of the sparse vector, the transmission block length M must satisfy:

$$M = \mathcal{O}\left(K \log\left(\frac{N}{K}\right)\right). \quad (5)$$

As indicated in (1), increasing the values of N and K allows the SSC scheme to carry more information bits. However, a larger value of K leads to a degradation of the decoding performance, while a larger value of N leads to a larger size of the codebook matrix. More importantly, according to (5), the required transmission block length M increases with both K and N further enlarging the size of the codebook matrix. A large codebook matrix will lead to a sharp increase in the encoding and decoding complexity and storage overhead of the SSC scheme [4].

To address this issue, a sparse codebook matrix is proposed in this paper to reduce the encoding and decoding complexity of the SSC scheme. Specifically, we first construct a sampling matrix $\mathbf{G} \in \mathbb{R}^{M \times N}$, where each column contains only D ($D \ll M$) non-zero elements with unit values, and the D non-zero elements are randomly distributed. The sparse codebook matrix is then obtained by sampling the original codebook (as defined in (4)) using \mathbf{G} , given by

$$\bar{\mathbf{A}} = \sqrt{\frac{1}{R}} \mathbf{G} \odot \mathbf{A}, \quad (6)$$

where \odot denotes the Hadamard (element-wise) product and $R = D/M$ is a sparsity factor that controls the sparsity of the sparse codebook matrix. An example of $\bar{\mathbf{A}}$ for $M = 4$ and $N = 6$ is given by

$$\bar{\mathbf{A}} = \sqrt{\frac{1}{KR}} \begin{bmatrix} 1 & 0 & -1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 \\ 1 & 0 & 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 1 & 0 & 0 \end{bmatrix}. \quad (7)$$

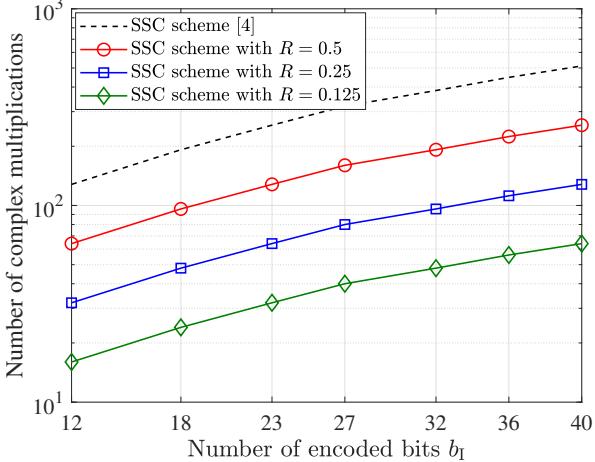


Fig. 2. Encoding complexity vs. the value of b_I .

Replacing the codebook matrix \mathbf{A} in equation (3) with the sparse codebook matrix $\bar{\mathbf{A}}$, the random spreading process can be expressed as:

$$\mathbf{x} = \sum_{k=1}^K \bar{\mathbf{a}}_{\mathcal{S}_k} s_k, \quad (8)$$

where $|\mathcal{S}_k| = D$ and $\bar{\mathbf{a}}_{\mathcal{S}_k}$ denote the index set of non-zero elements and the corresponding non-zero value vector in the k -th codeword of $\bar{\mathbf{A}}$, respectively.

Fig. 2 shows the encoding complexity of the SSC scheme as a function of the value of b_I . For example, when $R = 0.5$, the encoding complexity using the sparse codebook matrix can be reduced by more than 50% compared to that of the original dense codebook matrix.

III. Decoding of the Low-Complexity SSC Scheme

In this section, the decoding algorithm of the low-complexity SSC scheme is introduced and its decoding complexity is analyzed.

A. Transmission of Spread Sequence

The signal \mathbf{x} is transformed into the time domain via an inverse discrete Fourier transform (IDFT), i.e., $\mathbf{x}_T = \mathbf{F}^H \mathbf{x}$, where $\mathbf{F} \in \mathbb{C}^{N \times N}$ is a normalized DFT matrix. To mitigate inter-symbol interference, a cyclic prefix (CP) of length L_{CP} is appended to \mathbf{x}_T . At the receiver, after removing the CP and applying the DFT, the received signal is given by

$$\mathbf{y} = \mathbf{F} \mathbf{H}_T \mathbf{F}^H \bar{\mathbf{A}} \mathbf{s} + \mathbf{w}, \quad \mathbf{y} = \Phi \mathbf{s} + \mathbf{w}, \quad (9)$$

where $\Phi = \mathbf{F} \mathbf{H}_T \mathbf{F}^H \bar{\mathbf{A}}$ is the measurement matrix, $\mathbf{H}_T \in \mathbb{C}^{M \times M}$ denotes the time-domain channel matrix, and $\mathbf{w} \sim \mathcal{CN}(0, \sigma^2 \mathbf{I}_N)$ is the additive white Gaussian noise (AWGN) vector with variance σ^2 . In static scenario, the channel remains unchanged over the duration of a transmission block, thus the time-domain channel matrix \mathbf{H}_T is a cyclic matrix [16].

B. Low-Complexity Decoding Algorithm

Our goal is to recover the indexes and values of non-zero elements in the sparse vector \mathbf{s} from the received signal \mathbf{y} . The maximum likelihood (ML) solution of the system model in (7) can be formulated as

$$\mathbf{s}^* = \arg \max_{\|\mathbf{s}\|_0=K} \Pr(\mathbf{y}|\mathbf{s}, \Phi), \quad (10)$$

where $\|\mathbf{s}\|_0$ is the ℓ_0 -norm of \mathbf{s} . Since our goal is to find out the support of \mathbf{s} , we alternatively have

$$\mathcal{B}^* = \arg \min_{|\mathcal{B}|=K} \|\mathbf{y} - \Phi \mathbf{s}_{\mathcal{B}}\|^2. \quad (11)$$

The existing MMP [2] can be used to solve the sparse recovery problem in (11). Compared to conventional OMP, the Multipath Matching Pursuit (MMP) algorithm significantly improves the robustness and accuracy of sparse signal recovery by exploring multiple candidate support paths in parallel. However, when applied with a dense measurement matrix, the MMP algorithm incurs significantly higher computational cost due to the expensive orthogonal projection

$$\hat{\mathbf{s}}_{\hat{\mathcal{B}}} = (\Phi_{\hat{\mathcal{B}}}^T \Phi_{\hat{\mathcal{B}}})^{-1} \Phi_{\hat{\mathcal{B}}}^T \mathbf{y}, \quad (12)$$

and residual updation

$$\mathbf{r} = \mathbf{y} - \Phi_{\hat{\mathcal{B}}} \hat{\mathbf{s}}_{\hat{\mathcal{B}}}, \quad (13)$$

which need to be performed in each iteration.

Note that the measurement matrix Φ is a sparse matrix due to the diagonal structure of the equivalent channel matrix $\mathbf{F} \mathbf{H}_T \mathbf{F}^H$. Exploiting this sparsity, the Gram matrix $\mathbf{U} = \Phi_{\hat{\mathcal{B}}}^T \Phi_{\hat{\mathcal{B}}}$ can be performed element-wise as

$$\mathbf{U}_{k,l} = \sum_{i \in \mathcal{S}_k \cap \mathcal{S}_l} \Phi_{i,k}^* \Phi_{i,l}, \quad (14)$$

where \mathcal{S}_k is the non-zero index set of the k -th column of $\Phi_{\mathcal{B}^*}$, $\Phi_{i,l}$ denotes the (i, l) -th element of $\Phi_{\mathcal{B}^*}$. Similarly, the correlation calculation in (13) can be formulated as

$$[\Phi_{\hat{\mathcal{B}}}^T \mathbf{y}]_k = \sum_{i \in \mathcal{S}_k} \Phi_{i,k}^* y_i. \quad (15)$$

After obtaining the estimated values of the non-zero elements of the sparse vector \mathbf{s} , the demodulation problem of QAM symbols can be described as

$$s_k^* = \arg \max_{s_k \in \mathcal{A}} \Pr(\hat{s}_k | s_k), \quad (16)$$

where \mathcal{A} is a QAM alphabet with size M_{mod} . The LLR of each bit can be calculated as

$$f(b_{k,m}) = \log \left(\frac{\sum_{k=1}^K \Pr(\hat{s}_k|b_{k,m} = 1)}{\sum_{k=1}^K \Pr(\hat{s}_k|b_{k,m} = 0)} \right), \quad (17)$$

where $b_{k,m} \in \mathcal{A}, \forall m = 1, 2, \dots, \log_2(M_{\text{mod}})$. Note that the conditional probabilities $\Pr(\hat{s}_k|b_{k,m} = 1)$ and $\Pr(\hat{s}_k|b_{k,m} = 0)$ are calculated differently for different modulation orders. The output bit can be given as

$$b_{k,m} = \begin{cases} 0, & f(b_{k,m}) < 0 \\ 1, & f(b_{k,m}) \geq 0 \end{cases}. \quad (18)$$

Finally, the b_I bits can be recovered by sparse demapping of the non-zero index set \mathcal{B}^* , while the b_S bits can be obtained after soft demodulation.

C. Complexity Analysis

The computational complexity of the MMP algorithm varies significantly between dense and sparse codebook matrices. For dense codebook matrices, the MMP decoding complexity is dominated by two operations: least squares estimation for non-zero value recovery with complexity $O(K^3 + MK^2)$, and residual update through matrix-vector multiplication with complexity $O(NM)$. The overall complexity is $O(K^3 + MK^2 + NM)$. In contrast, when employing the proposed sparse codebook matrix with sparsity factor R , the complexity for least squares is given by $O(K^3 + RMK^2)$. The complexity of residual update is substantially reduced to $O(RNM)$ due to the sparse structure of codebook, leading to an overall complexity of $O(K^3 + RMK^2 + RNM)$. This analysis demonstrates that the sparse codebook matrix achieves approximately R -fold complexity reduction in the decoding algorithm. The complexity reduction is particularly significant when $NM \ll K^2M$, which is typical in practical SSC systems with large codebook dimensions. For instance, with $R = 0.5$, the sparse codebook matrix achieves approximately 50% complexity reduction. This substantial complexity reduction makes the proposed sparse codebook matrix highly attractive for resource-constrained URLLC applications while maintaining comparable decoding performance.

IV. Simulation Results

The BLER performance of proposed low-complexity SSC scheme is evaluated in this section. The existing SVC [2] and SSC [4] schemes are selected for performance comparison. QPSK modulation is adopted for the SSC and proposed schemes. The BLER is defined as the ratio of number of packets received in error to total number of transmitted packets.

Fig. 3 shows the BLER performance of different schemes. It can be observed that the BLER performance of the proposed scheme degrades gradually as the sparsity factor R decreases. Notably, for $R \leq 0.25$, the proposed scheme exhibits minimal BLER performance degradation

compared to the conventional SSC scheme [4]. However, when the value of R continues to decrease below this threshold, the BLER performance deteriorates rapidly. At larger R values, the proposed SSC scheme achieves BLER performance very close to that of the conventional SSC scheme. For instance, when $R = 0.5$, the SNR gap between the proposed scheme and the conventional SSC scheme is merely approximately 0.2 dB at $\text{BLER} = 10^{-5}$. This indicates that the proposed scheme can substantially reduce the encoding and decoding complexity of SSC schemes with only marginal BLER performance sacrifice. Moreover, the proposed scheme outperforms the conventional SVC scheme [2] when $R > 0.375$, demonstrating superior performance over a wide range of sparsity factors.

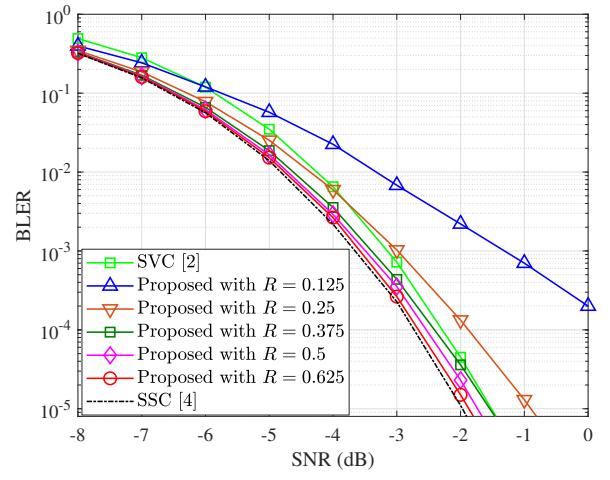


Fig. 3. BLER performances of different schemes vs. SNR over Rayleigh channels with $K = 2$, $N = 257$, $M = 128$ and $b = 19$.

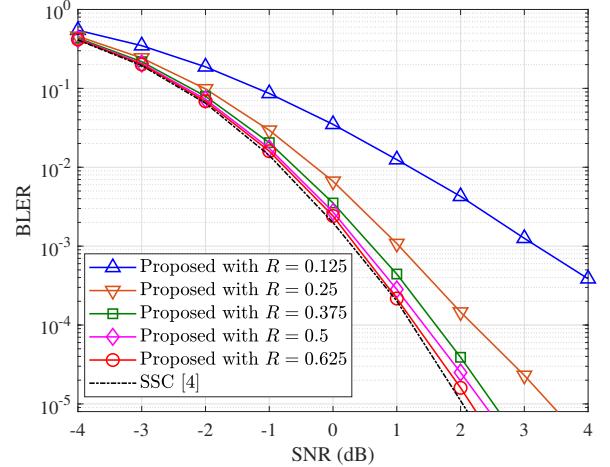


Fig. 4. BLER performances of different schemes vs. SNR over Rayleigh channels with $K = 4$, $N = 240$, $M = 117$ and $b = 35$.

Fig. 4 investigates the BLER performance of different schemes when K takes a larger value. The proposed

scheme exhibits similar advantages compared to conventional SVC and SSC schemes as observed in Fig. 3, confirming the robustness of the performance gains across different system configurations. These results validate that the proposed scheme effectively balances the BLER performance and encoding/decoding complexity for short-packet transmission across varying transmission bit rates.

Fig. 5 shows the BLER performance of the proposed scheme versus the sparsity factor R . It can be observed that the BLER performance deteriorates as the sparsity factor R decreases. Moreover, a sharp decline in BLER is observed when $R < 0.3$, indicating that decoding performance benefits significantly from reduced sparsity in this region. However, when $R > 0.3$, the declining trend of BLER becomes much more gradual. This indicates that the proposed scheme achieves an optimal trade-off between decoding performance and computational complexity around $R = 0.3$.

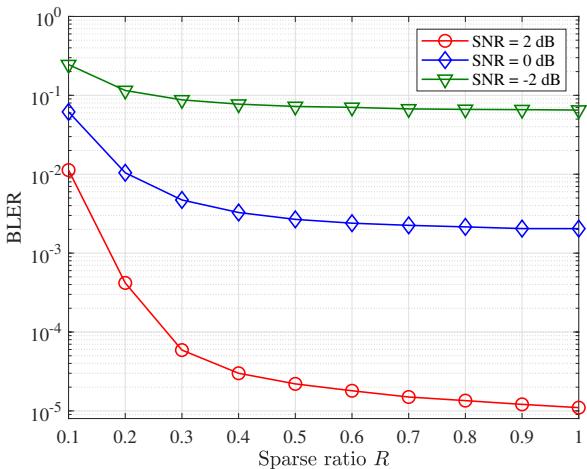


Fig. 5. BLER performances vs. sparsity factor R over Rayleigh channels with $K = 4$, $N = 240$, $M = 117$ and $b = 35$.

Fig. 6 shows the BLER performance of the proposed scheme under different sparsity factors R and various block lengths M . It can be observed that the BLER performance of the proposed scheme degrades as the transmission block length decreases. When $R = 0.5$, the proposed scheme exhibits BLER performance closely comparable to that of the conventional scheme across different transmission block lengths. However, when $R = 0.25$, the BLER performance gap between the proposed and conventional SSC schemes widens as the transmission block length decreases. This indicates that the proposed scheme with $R = 0.5$ demonstrates superior robustness across varying transmission block lengths.

V. Conclusions

In this paper, a low-complexity SSC scheme has been proposed to enhance the efficiency of short-packet transmission. By introducing a sparse codebook structure,

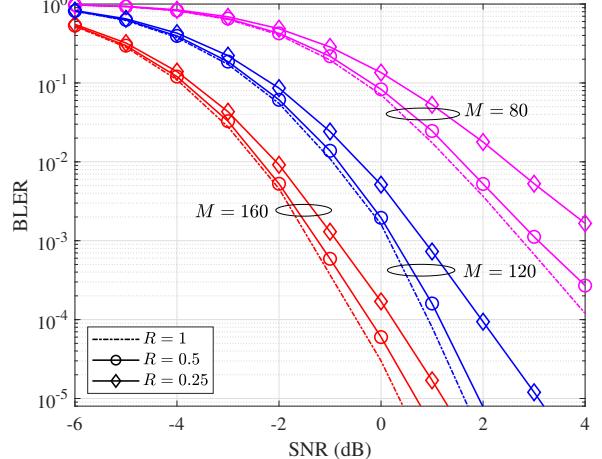


Fig. 6. BLER performances vs. SNR with $K = 4$, $N = 240$, $M = 117$ and $b = 35$.

the proposed scheme significantly reduces the encoding and decoding complexity with minimal BLER performance degradation. Simulation results demonstrate that the proposed approach achieves a favorable trade-off between reliability and complexity. Moreover, the scheme exhibits strong robustness across different transmission block lengths and outperforms conventional SVC under appropriate sparsity configurations, making it a promising candidate for URLLC scenarios.

References

- [1] W. Kim, H. Ji, H. Lee, Y. Kim, J. Lee, and B. Shim, “Sparse vector transmission: An idea whose time has come,” IEEE Veh. Technol. Mag., vol. 15, no. 3, pp. 32–39, 2020.
- [2] H. Ji, S. Park, and B. Shim, “Sparse vector coding for ultra reliable and low latency communications,” IEEE Trans. Wireless Commun., vol. 17, no. 10, pp. 6693–6706, 2018.
- [3] W. Kim, S. Bandari, and B. Shim, “Enhanced sparse vector coding for ultra-reliable and low latency communications,” IEEE Trans. Veh. Technol., vol. 69, no. 5, pp. 5698–5702, 2020.
- [4] X. Zhang, D. Zhang, B. Shim, G. Han, D. Zhang, and T. Sato, “Sparse superimposed coding for short-packet URLLC,” IEEE Internet Things J., vol. 9, no. 7, pp. 5275–5289, 2022.
- [5] L. Yang and P. Fan, “Multiple-mode sparse superposed code with low block error rate for short packet urllc,” IEEE Commun. Lett., vol. 28, no. 2, pp. 248–252, 2024.
- [6] C. Rush, K. Hsieh, and R. Venkataraman, “Capacity-achieving spatially coupled sparse superposition codes with AMP decoding,” IEEE Trans. Inf. Theory, vol. 67, no. 7, pp. 4446–4484, 2021.
- [7] M. K. Sinha and A. P. Kannu, “Generalized sparse regression codes for short block lengths,” IEEE Trans. Commun., vol. 72, no. 5, pp. 2536–2551, 2024.
- [8] D. Han, J. Park, Y. Lee, H. Poor, and N. Lee, “Block orthogonal sparse superposition codes for ultra-reliable low-latency communications,” IEEE Trans. Commun., vol. 71, no. 12, pp. 6884–6897, 2023.
- [9] Y. Zhang, X. Zhu, Y. Liu, X. Fan, Y. Guan, M. L. Wong, and V. K. N. Lau, “Block sparse vector codes for ultra-reliable and low-latency short-packet transmission,” IEEE Trans. Commun., Early Access, 2025.
- [10] R. Zhang, B. Shim, Y. Lou, S. Jia, and W. Wu, “Sparse vector coding aided ultra-reliable and low-latency communications in multi-user massive MIMO systems,” IEEE Trans. Veh. Technol., vol. 70, no. 1, pp. 1019–1024, 2021.

- [11] Y. Zhang, X. Zhu, Z. Xu, Y. Luo, X. Zhan, W. Yang, and H. Liang, “Sparse superimposed codes for vehicular communications with low-resolution ADCs,” in Proc. IEEE/CIC International Conference on Communications in China (ICCC Workshops), 2024, pp. 277–281.
- [12] Y. Zhang, X. Zhu, Y. Liu, Y. Jiang, Y. Guan, D. González G., and V. K. N. Lau, “Sparse superimposed vector transmission for short-packet high-mobility communication,” IEEE Wireless Commun. Lett., vol. 12, no. 11, pp. 1961–1965, 2023.
- [13] X. Zhang, L. Yang, Z. Ding, J. Song, Y. Zhai, and D. Zhang, “Sparse vector coding-based multi-carrier NOMA for in-home health networks,” IEEE J. Sel. Areas Commun., vol. 39, no. 2, pp. 325–337, 2021.
- [14] Y. Zhang, X. Fan, H. Liang, W. Yang, J. Zheng, and T. Luan, “Semi-tensor sparse vector coding for short-packet urllc with low storage overhead,” in 2025 IEEE Wireless Communications and Networking Conference (WCNC), 2025, pp. 1–6.
- [15] M. F. Duarte and Y. C. Eldar, “Structured compressed sensing: From theory to applications,” IEEE Trans. Signal Process., vol. 59, no. 9, pp. 4053–4085, 2011.
- [16] Y. Zhang, X. Zhu, Y. Liu, Y. Jiang, Y. Guan, and V. K. N. Lau, “Hierarchical beam based channel estimation with very low pilot overhead for high mobility MIMO-OFDM systems,” IEEE Trans. Veh. Technol., vol. 71, no. 10, pp. 10543–10558, 2022.